StarPRNT iOS SDK User's Manual

Sep. 4, 2019

Abstract

This paper provides information about the StarPRNT iOS SDK.

It provides guidelines for our customers to build the StarPRNT application.

Version History

Release Date	Update
Apr. 01, 2016	Release
Aug. 11, 2016	Correction of errors
Sep. 12, 2016	Added the uploadData method to the SMCSAllReceipts class.
Apr. 17, 2017	Supports BlackMark and PageMode.
Aug. 16, 2017	Added the API reference of class included in StarIO framework.
	Added the SMBluetoothManagerFactory class.
	Added the API reference of the peripherals (barcode reader / customer display / scale).
Jan. 26, 2018	Added the appendHorizontalTabPosition method to the ISCBBuilder interface.
May. 21, 2018	Added support for mC-Print2, mC-Print3.
	Added list of supported peripherals.
	Added list of printer configuration software.
	Added retry option to portSettings.
	Added the connectedInterface member to the StarPrinterStatus structure.
	Added driveTime parameter and delayTime parameter to appendSound method of the ISCBBuilder interface.
	Added the appendTopMargin method to the ISCBBuilder interface.
	Added the appendPrintableArea method to the ISCBBuilder interface.
	Changed the USB printer debugging method to the Xcode 9 method.
	Added precaution specifically for models which support both Bluetooth and USB (Lightning) for the getPort method and searchPrinter method.
	Added precaution related to readPort compatibility.
	Added information about API supporting Swift.
Jun. 29, 2018	End of support for APS.
Oct. 19, 2018	Added the connectAsync method to the StarIoExtManager class.
	Added the didConnectPort method to the StarIoExtManagerDelegate protocol.
	Added the didFailToConnectPort method to the StarIoExtManagerDelegate protocol.
	Added the SMLogger class.
	Added the SMFileLogger class.
Nov. 20, 2018	Added support for Zebra Technologies (Symbol) DS9208.
	Added support for mC-Sound.
	Added the createMelodySpeakerCommandBuilder method to the StarIoExt class.
	Added the createMelodySpeakerConnectParser method to the StarIoExt class.
	Added the ISMCBBuilder interface.
	Added the StarIoExtMelodySpeakerModel constants.
	Added the SMCBMelodySpeakerSoundStorageArea constants.
	Added the SMSoundSetting class.
Mar. 7, 2019	Added information about Objective-C Bridging Header.
	Added the appendCjkUnifiedIdeographFont method to the ISCBBuilder interface.
	Added the ExtMode to the SCBBarcodeWidth constant.
Sep. 4, 2019	Added how to set Bluetooth Always Usage Description.
	Added information about Objective-C Bridging Header.
	Added information about error handling for the following methods.
	getPort
	beginCheckedBlock
	endCheckedBlock
	writePort
	readPort
	getParsedStatus
	getFirmwareInformation
	searchPrinter
	disconnect
	Added notes on MCP30.

About this manual

This manual is designed to help you understand StarIO and how to build an iOS application to interact with Star Micronics Thermal POS Printers. It is important to understand the basics of the Objective-C or Swift language. Although this SDK is for iOS, there are SDKs available for many different operating systems and programming languages at our website in the Developers section. Check the Developers section of our site for the newest SDKs, technical documentation, FAQs, and many more additional resources.

CAUTION:

- iPad, iPhone, iPod touch and Retina are trademarks of Apple Inc., registered in the U.S. and other countries. iPad Air, iPad mini and Lightning are trademarks of Apple Inc. IOS is a trademark or registered trademark of Cisco in the U.S. and other countries and is used under license.
- Windows is registered trademarks of Microsoft Corporation.
- The Bluetooth® word mark and logos are registered trademarks owned by Bluetooth SIG, Inc.
- The information in this manual is subject to change without notice.
- STAR MICRONICS CO., LTD. has taken every measure to provide accurate information, but assumes no liability for errors or omissions.
- STAR MICRONICS CO., LTD. is not liable for any damages resulting from the use of information contained in this manual.
- Reproduction in whole or in part is prohibited.

Contents

1 Getting Start

1.1 Star Printer Compatibility Chart

	Printer Models	Firmware Version	Interface	Emulation
POS Printer	mC-Print2	1.0 or later	Bluetooth	StarPRNT
			Ethernet	
			USB	
	mC-Print3	1.0 or later	Bluetooth	StarPRNT
			Ethernet	
			USB	
	mPOP	1.0 or later	Bluetooth	StarPRNT
	FVP10	1.2 or later	Bluetooth	StarLine
			Ethernet	
	TSP100IIIW	1.0 or later	Wireless LAN	StarGraphic
	TSP100IIILAN	1.0 or later	Ethernet	StarGraphic
	TSP100IIIBI	1.0 or later	Bluetooth	StarGraphic
	TSP100IIIU	1.0 or later	USB	StarGraphic
	TSP100ECO	1.0 or later	USB *1	StarGraphic
	TSP100U	1.3 or later	USB *1	StarGraphic
	TSP100GT	1.0 or later	USB *1	StarGraphic
	TSP100LAN	2.0 or later	Ethernet	StarGraphic
	TSP650II	1.0 or later	Bluetooth	StarLine
			Ethernet	
	TSP700II	4.0 or later	Bluetooth	StarLine
		3.0 or later	Ethernet	
	TSP800II	2.0 or later	Bluetooth	StarLine
		1.2 or later	Ethernet	
	BSC10 LAN	1.0 or later	Ethernet	ESC/POS

		4.0 or later	Bluetooth (Excluding Jp model)	StarDotImpact
	SP700	3.0 or later	Ethernet	
Portable Printer	SM-S210i	2.4 or later	Bluetooth	StarPRNT *2 / EscPosMobile
	SM-S220i	2.0 or later	Bluetooth	StarPRNT *2 / EscPosMobile
	SM-S230i	1.0 or later	Bluetooth	StarPRNT / EscPosMobile
	SM-T300i	2.4 or later	Bluetooth	StarPRNT *2 / EscPosMobile
	SM-T300	1.1 or later	Wireless LAN	EscPosMobile
Portable Printer	SM-T400i	2.4 or later	Bluetooth	StarPRNT *2 / EscPosMobile
	SM-L200	1.0 or later	Bluetooth Low Energy	StarPRNT / StarPRNTL *3
	SM-L300	1.1 or later	Bluetooth Low Energy	StarPRNT / StarPRNTL

*1 USB interface printer works via an Apple Airport Express.

*2 Support for StarPRNT emulation is firmware version 3.0 or later.

*3 Support for StarPRNTL emulation is firmware version 2.0 or later

1.2 Supported peripherals

The supported peripherals and models which can be used are the following.

Peripherals	Description
BCR-POP1	Barcode reader
DS9208	Zebra Technologies (Symbol) barcode reader
SCD222U	Customer display
mC-Sound	Melody speaker

Peripherals	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i/T300	SM-T400i	BSC10	SM-S210i-StarPRNT	SM-S220i-StarPRNT	SM-S230i-StarPRNT	SM-T300i/T300 StarPRNT	SM-T400i-StarPRNT	SM-L200	SM-L300	SP700
BCR-POP1	✔	✔	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
DS9208	✔	✔	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
SCD222U	✔	✔	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
mC-Sound	-	*1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

- : Not guarantee.
*1: Firmware version 1.3 or later is required.

❖ Portable Printer

When using StarPRNT emulation:

To use the StarPRNT emulation, set the emulation setting of the printer to "StarPRNT". To change the emulation, proceed as follows. (For SM-L200 and SM-L300, it does not need to switch the emulation.)

◆ Switching over between StarPRNT and ESC/POS emulation

1. Turn the printer power and open the printer cover.

2. Press and hold the POWER button and the FEED button simultaneously. As soon as the ERROR lamp flashes five times, release the buttons. The emulation switchover takes place automatically.

3. After setting a paper, close the printer cover. The set emulation is printed out.
   ESC/POS : EMU = ESC/POS
   StarPRNT : EMU = StarPRNT

   If the emulation is not switched correctly, repeat the above steps 1 to 3.
   At that time, in step 2, make sure not to release the buttons until the lamp completes the 5th flash.

4. Please reboot the printer after switching the emulation. *It will be valid after rebooting the printer.

1.3 Connecting a Star POS Printer to an iOS Device

Bluetooth

All Star Bluetooth devices have each initial device name as a factory default setting, such as "Star Micronics" and "DK-AirCash". When using multiple devices, which have same device name, it is useful to change the device name for identifying each device easily.

You can change the LAN/Bluetooth settings of the Star POS device, such as changing the Bluetooth device name. Please use the Printer Configuration Software.

- Pairing of a Star POS device with your iOS Device

   1. Ensure Bluetooth is enabled and the Star POS Device is powered on.
    When the security setting of the Star POS device is set to SSP, press the PAIR button for more than 5 seconds to make it available for pairing.
   2. Tap Settings > Bluetooth.
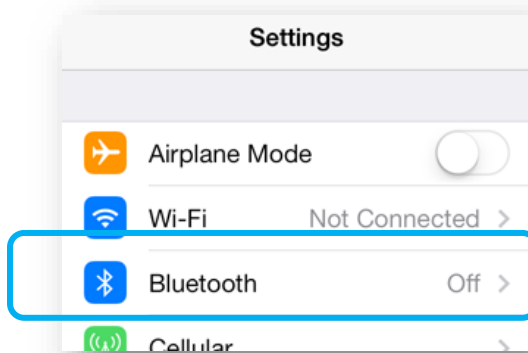
  3. Tap Bluetooth to turn it on. Your iOS device searches and displays the Bluetooth devices in range. Tap the Star portable printer you want to pair with.

4. If a PIN code is used for Star Bluetooth device pairing, enter the PIN and tap Pair.
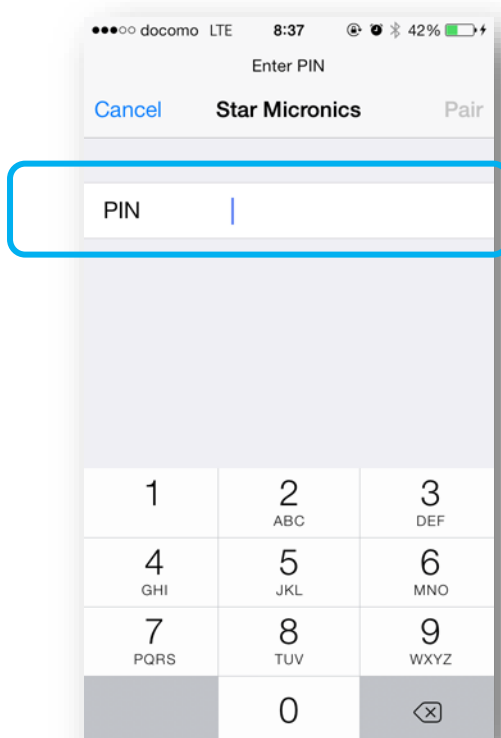
5. When the pairing is complete, you'll see this message.
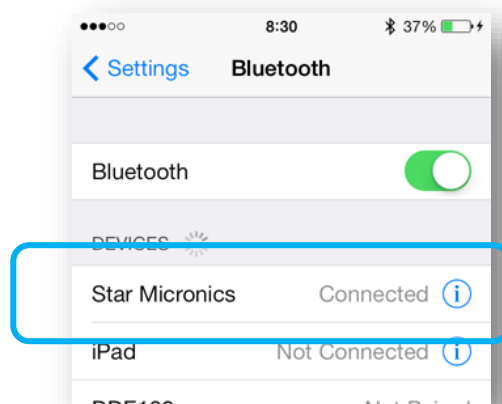
◆**How to change the Bluetooth Device Name**

Please use the Printer Configuration Software to change the iOS Port Name.

To confirm iOS Port Name, select [Settings]-[General]-[About] after Bluetooth pairing is established. The iOS Port Name will be shown under the Bluetooth address.

Ethernet Interface

Star POS Printers ship with DHCP enabled by default. If your network supports DHCP, be sure to make the necessary configurations so that your Star POS Printer will automatically get an IP Address.

Use Star POS Printers with the #9100 Multi Session disabled. The setting can be confirmed by Test Print which can be executed by holding down the printer's feed button while turning the printer on.

Refer to "guidelines-ethernet_en.pdf" for how to confirm and change the #9100 Multi Session setting and how to set the Static IP Address.

You can set up your POS Printers which can connect to TCP/IP. Please use the Printer Configuration Software.

*1 TSP100IIIUtility and Star Setting Utility cannot be used with printers where an IP address has not been assigned (IP address 0.0.0.0).

*2 This software uses TCP communication port 9100, TCP communication port 9100, UDP communication port 2222.

1. Assign an IP Address to the Star POS Devices and connect it to the network.

2. Tap Settings.

3. Ensure Wi-Fi is ON.
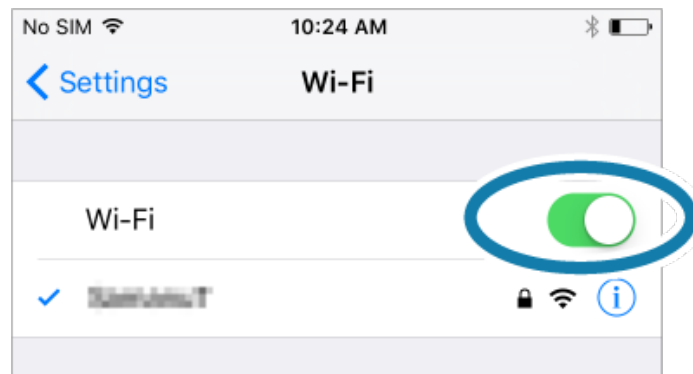
4.Connect to the same network the Star POS Device is on.

USB Interface

Connect the printer and the iOS device with a Lightning cable.

* For usable models, refer to <u>Star Printer Compatibility Chart</u>.

◆**How to debug the USB printer**

- The following descriptions, procedures and URL are subject to changes when Xcode versions upgrade or site updates.   This document is based on our verification using Xcode 9.3 of MacOS 10.13.4 and iOS 11.3 on May 21, 2018.

   Refer to the Apple web site for the most recent information.

When using a USB printer, because the iOS device Lightning port is used for the printer connection, it is not possible to install apps or debug using the Lightning cable. However with Xcode 9 and iOS 11, wireless debugging can be used.

<Setup>

   * The iOS device and mac must be connected to the same network.

   1. Set a pass code for the iOS device using the iOS device "Settings" app.

   2. Connect the iOS device and mac using the Lightning cable.

   3. Launch Xcode and select "Window" - "Devices and Simulators".

   4. Select the iOS device to use in the left pane, and place a checkmark next to "connect via network".

Thereafter, Xcode will display the iOS device in the network even when it is not connected using the Lightning cable, allowing app installation and debugging.

1.4 Printer Configuration Software

Use the software listed below to change the printer settings.

For iOS

Software	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i/T300	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i/T300 StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
Star Setting Utility *1	-	-	-	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔
mPOP Utiltiy	-	-	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
mC-Print Utility	✔	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
TSP100III Utility	-	-	-	-	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Download software from App Store.

- : Not guarantee.
*1 : The memory switch setting cannot be changed.
    To change the memory switch setting, use the software for Windows.

For Android

Software	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i/T300	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i/T300 StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
Star Setting Utility *1	-	-	-	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔
mPOP Utiltiy	-	-	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
mC-Print Utility	✔	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
TSP100III Utility	-	-	-	-	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Download software form [Google play](#).

- : Not guarantee.
*1 : The memory switch setting cannot be changed.
  To change the memory switch setting, use the software for Windows.

For Windows

Software	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i/T300	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i/T300 StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
StarPRNT Intelligence	-	-	-	✔	-	✔	✔	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	✔
StarPRNT Intelligence for mCollection	✔	✔	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
StarPRNT Intelligence for Portable	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✔	✔	✔	✔	✔	✔	✔	-
StarPRNT Intelligence for BSC10	-	-	-	-	-	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-	-	-
Star TSP100 futurePRNT *1	-	-	-	-	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Download software from here.

- : Not guarantee.
*1 : The memory switch setting cannot be changed.
　　To change the memory switch setting, use the software for Android or iOS.

2 How to configure iOS StarPRNT Project

2.1 Add iOS StarPRNT SDK to Project.

- Add StarIO.framework to "Link Binary With Libraries" on the "Build Phases" tab.

- Add StarIO_Extension.framework to "Link Binary With Libraries" on the "Build Phases" tab.

- Add SMCloudServices.framework to "Link Binary With Libraries" on the "Build Phases" tab.

- Add SMCloudServicesResources.bundle in the SMCloudServices.framework folder to "Copy Bundle Resources" on the "Build Phases" tab.

```
-SMCloudServices.framework-
You need to add when using Star Micronics Cloud (AllReceipts, etc.) in the
application.
```

2.2 Add following frameworks to Project.

- CoreBluetooth.framework

- ExternalAccessory.framework

2.3 Edit information property list.

2.3.1 (When using Bluetooth printer) Set Supported external accessory protocols
1. Click on the information property list file (default : "Info.plist").
2. Add the "Supported external accessory protocols" Key.
3. Click the triangle of this key and set the value for the "Item 0" to "jp.star-m.starpro".
   Note: Please do not apply this, if you are not using Bluetooth ineterface.

2.3.2 (When using Bluetooth Low Energy printer on iOS13 or later) Set Bluetooth Always Usage Description
1. Click on the information property list file (default : "Info.plist").
2. Add the "Privacy – Bluetooth Always Usage Description" Key.
3. Set the reason for using Bluetooth in Value (e.g. "Use Bluetooth for communication with the printer.")
4. When communicating with the Bluetooth Low Energy printer on iOS13 or higher, an alert requesting permission to access Bluetooth is displayed. The string set in Value is displayed in the alert as the reason for using Bluetooth.

Swift SDK 〉 Swift SDK 〉 Info.plist 〉 No Selection

Project navigator (left panel):

- Swift SDK
  - ▶ SMCloudServicesResources.bundle
  - ▶ SMCloudServices.framework
  - ▶ StarIO_Extension.framework
  - ▶ StarIO.framework
  - ▶ CoreBluetooth.framework
  - ▼ Swift SDK
    - AllReceiptsExtViewController.swift
    - AllReceiptsViewController.swift
    - ApiViewController.swift
    - AppDelegate.swift
    - Assets.xcassets
    - BarcodeReaderE...iewController.swift
    - BlackMarkPasteViewController.swift
    - BlackMarkViewController.swift
    - CashDrawerExtViewController.swift
    - CashDrawerViewController.swift
    - CombinationExtViewController.swift
    - CombinationViewController.swift
    - CommonViewController.swift
    - ▶ Communications
    - DeviceStatusViewController.swift
    - DisplayExtViewController.swift
    - DisplayViewController.swift
    - ▶ Functions
    - Info.plist
    - ▶ LaunchScreen.storyboard

Key	Type	Value
▼ Information Property List	Dictionary	(24 items)
App Category	String	
Localization native development region	String	en
Bundle display name	String	StarPRNT SDK
Executable file	String	$(EXECUTABLE_NAME)
Bundle identifier	String	$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	$(PRODUCT_NAME)
Bundle OS Type code	String	APPL
Bundle versions string, short	String	5.11.0
Bundle creator OS Type code	String	????
Bundle version	String	1
Application requires iPhone environment	Boolean	YES
▶ App Transport Security Settings	Dictionary	(1 item)
Privacy - Bluetooth Always Usage Description	String	Use Bluetooth for communication with the printer.
Privacy - Bluetooth Peripheral Usage Description	String	Use Bluetooth for communication with the printer.
Privacy - Camera Usage Description	String	Use a Photo file as print data.
Privacy - Photo Library Usage Description	String	Use a Photo file as print data.
Application supports iTunes file sharing	Boolean	YES
Launch screen interface file base name	String	LaunchScreen
Main storyboard file base name	String	Main
▶ Required device capabilities	Array	(1 item)
▼ Supported external accessory protocols	Array	(1 item)
Item 0	String	jp.star-m.starpro
▶ Supported interface orientations	Array	(4 items)
▶ Supported interface orientations (iPad)	Array	(4 items)

2.4 (Development with Swift) Add Objective-C Bridging Header

Frameworks included in the iOS StarPRNT SDK are written in Objective-C. To use them in Swift, you need to add Objective-C Bridging-Header to your project.

1. Select "File" – "New" – "File…" from Xcode menu.

2. Select the "Header File", and then click "Next".

3. Enter a name and click "Create" to create an empty header file.

   In the example shown below, the header file name is "StarIO_Extension-Bridging-Header.h".

4. Write the following import lines in the header file created.

   ```
   #import <StarIO/SMPortSwift.h>

   #import <StarIO_Extension/StarIoExt.h>
   #import <StarIO_Extension/StarIoExtManager.h>
   #import <StarIO_Extension/SMBluetoothManagerFactory.h>
   #import <StarIO_Extension/SMSoundSetting.h>

   // To use SMCloudServices.framework, you also need to write the following lines.
   #import <SMCloudServices/SMCloudServices.h>
   #import <SMCloudServices/SMCSAllReceipts.h>
   ```

5. Specify the path to the created header file at Objective-C Bridging Header of Build Settings.
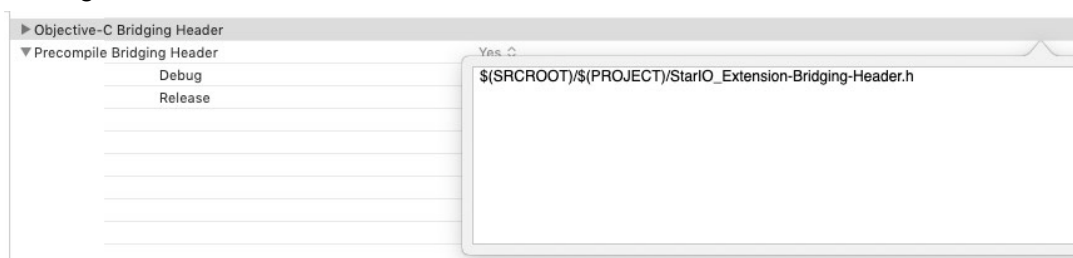
6. Then, you can use frameworks in Swift.

3 SMPort Class (StarIO.framework)

A SMPort class is the class to communicate with a printer.

Method

Name	Description
getPort	Creates a SMPort object and opens a port for communicating with the printer.
releasePort	Closes a connection to the port specified and destroy a SMPort object.
searchPrinter	Search for printers that can connect to Android devices.
writePort	Write data to the printer.
readPort	Read data from the printer.
getParsedStatus	Get printer status.
beginCheckedBlock	Starts to check completion of printing
endCheckedBlock	Terminates to check completion of printing
disconnect	Disconnects the specified Bluetooth device.
getFirmwareInformation	Gets printer model name and firmware version
StarIOVersion	Get the version number of the StarIO library.

Property

Name	Description
portName	Acquires the printer port name.
portSettings	Specifies connection setting information.
timeoutMillis	Acquires and specifies the timeout time for internal control and API. (unit: millisecond)
endCheckedBlockTimeoutMillis	Timeout value for internal control and API. (unit: millisecond)
connected	Connection status with the iOS device of the specified Bluetooth/Bluetooth Low Energy printer.

3.1   Printing flow using a SMPort class

Using the SMPort class, print per the following procedure.
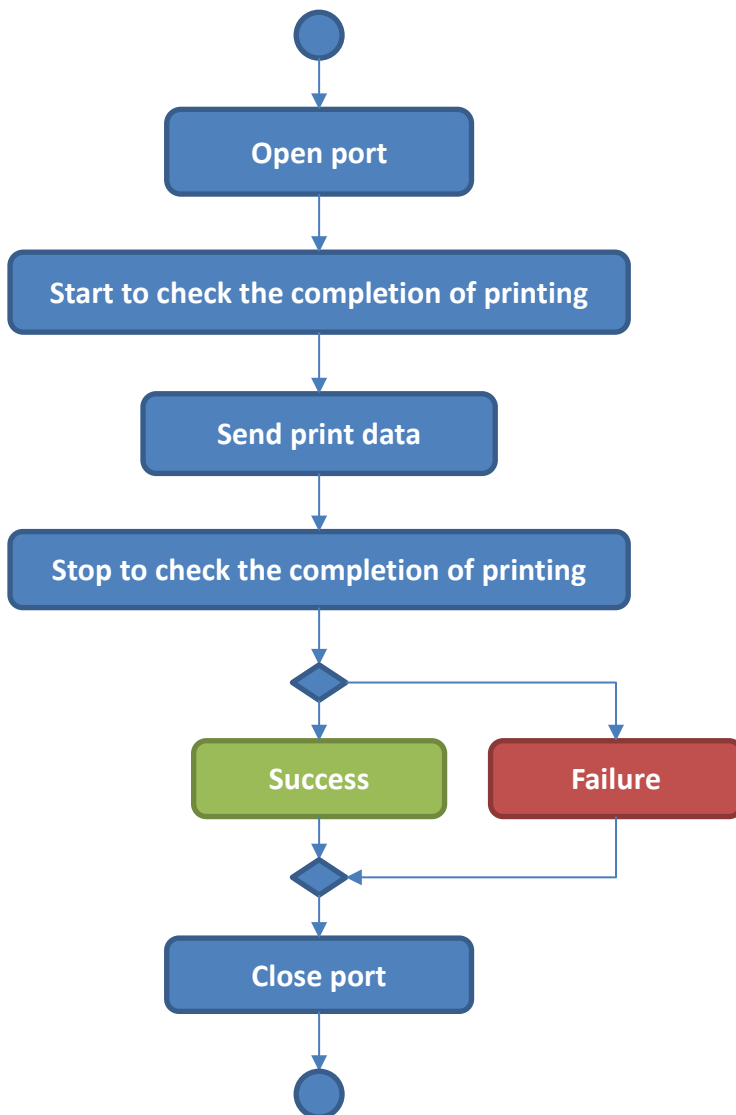
Objective-C

```objc
unsigned char command[] = {0x41, 0x42, 0x43, 0x44, 0x1B, 0x7A, 0x00, 0x1B, 0x64, 0x02};

StarPrinterStatus_2 printerStatus;
SMPort *port = nil;

while (YES) {
    NSError *error = nil;

    // Open port
    port = [SMPort getPort: "BT:Star Micronics" :@"" :10000 :&error];

    if (port == nil) {    break;  /* Some error occurred. */    }

    // Start to check the completion of printing
    [port beginCheckedBlock:&printerStatus :2 :&error];

    if (error != nil) {    break;  /* Some error occurred. */    }

    if (printerStatus.offline == SM_TRUE) {
        break;   // The printer is offline.
    }

    uint32_t total = 0;

    while (total < (uint32_t) commands.length) {
        // Send print data
        uint32_t written = [port writePort:command :total : sizeof(command) - total :&error];

        if (error != nil) {    break;  /* Some error occurred. */    }

        total += written;
    }

    // Stop to check the completion of printing
    [port endCheckedBlock:&printerStatus :2 :&error];

    if (error != nil) {    break;  /* Some error occurred. */    }

    if (printerStatus.offline == SM_TRUE) {
        break;   // The printer is offline.
    }

    // Success
    break;
}

if (port != nil) {
    // Close port
    [SMPort releasePort:port];
    port = nil;
}
```

Refer to Communication.m

Swift

```swift
var command: [UInt8] = [0x41, 0x42, 0x43, 0x44, 0x1b, 0x7a, 0x00, 0x1b, 0x64, 0x02]

while true {
   var port : SMPort

   do {
      // Open port
      port = try SMPort.getPort(portName: "BT:Star Micronics", portSettings: "", ioTimeoutMillis: 10000)

      defer {
         // Close port
         SMPort.release(port)
      }

      var printerStatus: StarPrinterStatus_2 = StarPrinterStatus_2()

      // Start to check the completion of printing
      try port.beginCheckedBlock(starPrinterStatus: &printerStatus, level: 2)

      if printerStatus.offline == sm_true {
         break    // The printer is offline.
      }

      var total: UInt32 = 0

      while total < UInt32(command.count) {
         var written: UInt32 = 0

         // Send print data
         try port.write(writeBuffer: command, offset: total, size: UInt32(command.count) - total,
numberOfBytesWritten: &written)

         total += written
      }

      // Stop to check the completion of printing
      try port.endCheckedBlock(starPrinterStatus: &printerStatus, level: 2)

      if printerStatus.offline == sm_true {
         break    // The printer is offline.
      }

      // Success
      break
   }
   catch let error as NSError {
      break    // Some error occurred.
   }
}
```

Refer to Communication.swift.

.

3.2 Model: SMPort class

Supported method for each model.

Method	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i/T300	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
getPort	*1	*1	*1	*1	*1	*1	*1	*1	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
releasePort	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
searchPrinter	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
writePort	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
readPort	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
getParsedStatus	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
beginCheckedBlock	✔	✔	✔	✔	✔	✔	*2	✔	✔	*4	✔	*4	✔	✔	✔	✔	✔	✔	✔	✔	✔	*2
endCheckedBlock	✔	✔	✔	✔	✔	✔	*2	✔	✔	*4	✔	*4	✔	✔	✔	✔	✔	✔	✔	✔	✔	*2
disconnect *6	✔	✔	✔	✔	-	✔	✔	✔	-	-	-	-	-	-	-	-	-	-	-	-	-	✔
getFirmwareInformation	✔	✔	✔	*5	*3*5	*5	*5	*5	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	*5
StarIOVersion	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔

-: igrored

*1 Support timeout value setting of data timeout function.
   TSP650II, TSP800II: Firmware version 2.0 or later.
   TSP700II: Firmware version 5.0 or later.
*2: Frmware version 3.0 or later is required
*3: The model name of TSP100U, TSP100GT and TSP100ECO is TSP100.
   It is impossible to get the firmware version of TSP100U, TSP100GT, TSP100ECO and TSP100LAN.
*4: When using SM-S200i with firmware Ver 2.1 or earlier, the following limitation exists.
   When using SM-T300 with firmware Ver 2.4 or earlier, the following limitation exists.
   You can confirm the completion of transmission of print data but cannot confirm the completion of printing.
*5: When using Apple AirMac Express with a USB printer, it returns an empty string.
*6: It functions only in Bluetooth interface.

3.3 getPort Method

Creates a SMPort object and opens a port for communicating with the printer.
Refer to the printing process flow using the SMPort class.

3.3.1 For Objective-C

Declaration
+ (SMPort *)getPort:(NSString *)portName

        :(NSString *)portSettings

        :(u_int32_t)ioTimeoutMillis

        :(NSError **)error;

Parameter

Name	Description	Object type
portName	Character string to specify the port for communication with the printer.	NSString *
portSettings	Specifies connection setting information. ・Emulation type ・Data timeout	NSString *
timeoutMillis	Timeout value for internal control and API. [unit: ms]	u_int32_t
error	Connection failure error information An error code is set in the code property.	NSError **

Return value

Description	Object type
An instance of SMPort class. It returns "nil" if it fails to generate communication port.	SMPort *

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.
SMStarIOResultCodeInUseError	Connection was refused by the printer (another host is connected or other reason).

3.3.2 For Swift

Declaration

open class func getPort(portName: String!, portSettings: String!, ioTimeoutMillis: UInt32) throws -> SMPort

open class func getPort(_ portName: String!, _ portSettings: String!, _ ioTimeoutMillis: UInt32, _ error: NSErrorPointer) -> SMPort!

Parameter

Name	Description	Object type
portName	Character string to specify the port for communication with the printer.	String
portSettings	Specifies connection setting information. ・Emulation type ・Data timeout	String
timeoutMillis	Timeout value for internal control and API. [unit: ms]	UInt32
error	Connection failure error information An error code is set in the code property.	NSErrorPointer

Return value

Description	Object type
An instance of SMPort class. It returns "nil" if it fails to generate communication port.	SMPort

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.
SMStarIOResultCodeInUseError	Connection was refused by the printer (another host is connected or other reason).

1. portName Parameter

Interface	portName	Description	Example
Bluetooth*1	BT:iOS Port Name	To specify the iOS Port Name	@"BT:Star Micronics"
	BT:Serial Number	To specify the printer using the serial number	@"BT:2580217090400032"
Bluetooth Low Energy	BLE:Device Name	To specify the Bluetooth Device Name	@"BLE:STAR L200-00001"
	BLE:MAC address	To specify the MAC address in Bluetooth	@"BLE:00:11:62:1b:4d:f4"
Ethernet / Wireless LAN	TCP:IP address	To specify the IP address	@"TCP:192.168.1.130"
USB*1	USB:iOS Port Name	To specify the iOS Port Name	@"USB:TSP100"
	USB:Serial Number	To specify the printer using the serial number	@"USB:2580217090400032"

*1 With models that support both Bluetooth and USB (Lightning), there is no differentiation between Bluetooth and USB, and the same behavior occurs in both cases.
If the Lightning cable is connected while a Bluetooth connection is active, Bluetooth will be automatically disconnected and communication will occur via Lightning. However it is not necessary to change portName from "BT" to "USB" at this time.

2. portSettings Parameter

The portSettings string is specified using the format shown below.
- The identifier that should be used with that model + option specification strings delimited by ";"

The identifier that should be used with each model is as shown below.

Model	Identifier
mC-Print2	""
mC-Print3	""
mPOP	""
FVP10	""
TSP100	""
TSP650II	""
TSP700II	""
TSP800II	""
SM-S210i	"Portable;escpos" or "mini"
SM-S220i	"Portable;escpos" or "mini"
SM-S230i	"Portable;escpos" or "mini"
SM-T300i/T300	"Portable;escpos" or "mini"
SM-T400i	"Portable;escpos" or "mini"
BSC10	"escpos"
SM-S210i StarPRNT	"Portable"
SM-S220i StarPRNT	"Portable"
SM-S230i StarPRNT	"Portable"
SM-T300i/T300 StarPRNT	"Portable"
SM-T400i StarPRNT	"Portable"
SM-L200	"Portable"
SM-L300	"Portable"
SP700	""

The option specification strings are shown below.

Interface	Option type	Connection options
Bluetooth	"d[value]" Ex: "d10"	Data timeout function timeout value setting. For usable models, refer to StarIOPort class. The data timeout function is a function which ignores the remaining print data when data was not sent to the printer within the specified length of time during printing. This function prevents corruption of the next print contents after the Bluetooth connection was disconnected during data sending. The method for specifying the timeout time is the following. (1) If a value from 1 to 255 was specified for [value] : Time that was specified for [value] (units: seconds) (2) If 0 was specified for [value] : Data timeout function is ignored. (3) Other than the above : 3 seconds If the d option is not specified with a model that supports the data timeout function, the data timeout function is enabled and the timeout value is set to 3 seconds.
Ethernet/ Wireless LAN	"l[value]" (lower case "L") Ex: "l10000"	Connection retry will be performed when the target printer is in use by another host. The method for specifying the retry time is the following. (1) If a value will not be specified ("l" only) : Timeout time specified by the getPort method (2) If a value from 0 to 300000 was specified for [value] : Time that was specified for [value] (units: milliseconds) (3) If a value of 300001 or more was specified for [value] : Timeout time specified by the getPort method (4) Other than the above : Retry is not performed.
	"[Port number]" Ex: "9100"	Port number(Only when using AirPort)

Example of portSettings

Printer connection environment	portSettings
Uses TSP650II as the default setting.	""
Uses mC-Print3 with retry enabled.	";l10000"
Uses SM-L200 as the default setting.	"Portable"
If data is not sent for 10 seconds during printing withTSP650II, it operates the data canceling function.	";d10"

3. timeoutMillis Parameter
    timeout is a timeoutMillis timeout controlled internally and is used for communication in the APIs.
    This parameter guarantees that all of the below APIs will complete in a bounded amount of time,
    but does NOT guarantee the exact timeout length.
    Timeout length is 10 seconds if specified less than 10 seconds.
    When [Data timeout function setting] is used in the portSetting parameter, it should be at least 3
    seconds longer than the time specified for the data timeout function. If it is set to less than 3
    seconds, the timeout time of the endCheckedBlock method will be controlled internally to be 3
    seconds longer automatically.

4. Use share printer function with Apple AirPort Express

    Set AirPort Express IP Address for **portName**.

    Ex. @"TCP:192.168.1.2"

    Set port number for **portSettings**.
    Increase the port number in sequential order from 9100 to 9109 until communication is successful.

    Ex. @"9100"

5. Notification in case of SM-L Series
    It could take some time when an iOS device tries to connect to a printer via "Bluetooth Low
    Energy".
    If the connection fails, retry until the connection is successful.
    If the connection time must be reduced, please design your application as the connection to a
    printer always keeps opening. *In this case, the printer cannot be detected by any other
    applications and devices.

3.4 releasePort Method

Closes a connection to the port specified and destroy a SMPort object.

Declaration
+ (void) releasePort: (SMPort *) port;

Parameter

Name	Description	Object type
port	A SMPort object previously created by the getPort method	SMPort *

Return value

Description	Object type
-	-

Refer to the printing process flow using a SMPort class.

1. Notes
After executing getPort, please do not forget releasePort before executing the next getPort. Otherwise the communication may return nil.

3.5 searchPrinter Method

Searches for a printer that can be connected to the iOS device.

3.5.1 For Objective-C

Declaration

+ (NSArray *)searchPrinter:(NSString *)target

:(NSError **)error;

Parameter

Name	Description		Object type
target	All interface types	"ALL:"	NSString *
	Bluetooth	"BT:"	
	Bluetooth Low Energy	"BLE:"	
	Ethernet / Wireless LAN	"TCP:"	
	USB	"USB:"	
error	Connection failure error information An error code is set in the code property.		NSError **

* With models that support both Bluetooth and USB (Lightning), there is no differentiation between Bluetooth and USB.
   Regardless of the actual connection method, when a target is not specified or when "BT:" is specified, the printer is detected as a Bluetooth printer, and when "USB" is specified it is detected as a USB printer.

Return value

Description	Object type
Search result of Star printer. Return value includes instance of PortInfo class. Refer to the PortInfo class for the information that you can get.	NSArray *

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

Example

```
NSArray *portInfoArray;

NSError *error = nil;

portInfoArray = [SMPort searchPrinter:@"ALL:" :&error];

if (error != nil) {
    // Some error occurred.
}

if (portInfoArray == nil) {
    return;
}

for (PortInfo *portInfo in portInfoArray) {
    NSLog(@"Port Name:   %@", portInfo.portName);
    NSLog(@"MAC Address: %@", portInfo.macAddress);
    NSLog(@"Model Name:  %@", portInfo.modelName);
}
```
Refer to SearchPortViewController.m.

3.5.2 For Swift

Declaration
open class func searchPrinter(target: String!) throws -> [Any]

open class func searchPrinter(_ target: String!, _ error: NSErrorPointer) -> [Any]!

Parameter

Name	Description		Object type
target	All interface types	"ALL:"	String
	Bluetooth	"BT:"	
	Bluetooth Low Energy	"BLE:"	
	Ethernet / Wireless LAN	"TCP:"	
	USB	"USB:"	
error	Connection failure error information An error code is set in the code property.		NSErrorPointer

* With models that support both Bluetooth and USB (Lightning), there is no differentiation between Bluetooth and USB.
Regardless of the actual connection method, when a target is not specified or when "BT:" is specified, the printer is detected as a Bluetooth printer, and when "USB" is specified it is detected as a USB printer.

Return value

Description	Object type
Search result of Star printer. Return value includes instance of PortInfo class. Refer to the PortInfo class for the information that you can get.	[Any]

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

Example
```swift
var searchPrinterResult: [PortInfo]? = nil

do {
    searchPrinterResult = try SMPort.searchPrinter(target: "ALL:") as? [PortInfo]
}
catch {
    // do nothing
}

guard let portInfoArray: [PortInfo] = searchPrinterResult else {
    return
}

for portInfo: PortInfo in portInfoArray {
    print("Port Name:  \(portInfo.portName   ?? "")")
    print("MAC Address: \(portInfo.macAddress ?? "")")
    print("Model Name:  \(portInfo.modelName  ?? "")")
}
```

Refer to SearchPortViewController.swift.

1. Search execute time

The time it takes to execute the searchPrinter method differs according to the interface you want to search.

Interface	Execute time
All	7[s]
Bluetooth	Immediately responds with a control.
Bluetooth Low Energy	7[s]
Ethernet / Wireless LAN	6[s]
USB	Immediately responds with a control.

2. Restrictions

This API do not guarantee the discovery of devices.

3. Notes with the Bluetooth Low Energy interface

When getting the printer device name using searchPriner method for the first time, sometimes portName will be @"BLE:". In those cases, please connect the printer using getPort method. Once you have got the Device name, searchPrinter method works correctly.

3.6 writePort Method

This method writes data to the device. Use this to print to the printer, send commands, etc.
To check the completion of printing, run beginCheckedBlock before and endCheckedBlock after this method.

Refer to the printing process flow using a SMPort class.

3.6.1 For Objective-C

Declaration
- (u_int32_t)writePort:(u_int8_t const *)writeBuffer

            :(u_int32_t)offSet

            :(u_int32_t)size

            :(NSError **)error;

Parameter

Name	Description	Object type
writeBuffer	Contains the output data in a byte array.	u_int8_t const *
offset	Specifies where to begin pulling data from writeBuffer.	u_int32_t
size	Number of bytes to write.	u_int32_t
error	Connection failure error information An error code is set in the code property.	NSError **

Return value

Description	Object type
Bluetooth/Ethernet/Wireless LAN/USB I/F The number of bytes that were actually written. The writePort method is successful even when all of the data cannot be written. Your application should call this function a limited number of times until all the data has been written out or until an application determined retry threshold has been reached. **Bluetooth Low Energy I/F** It returns a transmission data size when it succeeded and "0" when it failed.	u_int32_t

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.6.2 For Swift

Declaration

open func write(writeBuffer: UnsafePointer<UInt8>!, offset: UInt32, size: UInt32, numberOfBytesWritten: UnsafeMutablePointer<UInt32>!) throws

open func write(_ writeBuffer: UnsafePointer<UInt8>!, _ offSet: UInt32, _ size: UInt32, _ error: NSErrorPointer) -> UInt32

Parameter

Name	Description	Object type
writeBuffer	Contains the output data in a byte array.	UnsafePointer<UInt8>
offset	Specifies where to begin pulling data from writeBuffer.	UInt32
size	Number of bytes to write.	UInt32
numberOfBytes Written	**Bluetooth/Ethernet/Wireless LAN/USB I/F** The number of bytes that were actually written. The writePort method is successful even when all of the data cannot be written. Your application should call this function a limited number of times until all the data has been written out or until an application determined retry threshold has been reached. **Bluetooth Low Energy I/F** It returns a transmission data size when it succeeded and "0" when it failed.	UnsafeMutablePointer <UInt32>
error	Connection failure error information An error code is set in the code property.	NSErrorPointer

Return value

Description	Object type
Bluetooth/Ethernet/Wireless LAN/USB I/F The number of bytes that were actually written. The writePort method is successful even when all of the data cannot be written. Your application should call this function a limited number of times until all the data has been written out or until an application determined retry threshold has been reached. **Bluetooth Low Energy I/F** It returns a transmission data size when it succeeded and "0" when it failed.	u_int32_t

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.7 readPort Method

Read data from the printer. Please use it only when it is necessary to read Raw byte from the printer.

1. Restrictions
Do not use this method to get Raw Status. Use getParsedStatus Method for getting status.

2. Precaution related to compatibility
Beginning from StarIO.framework v2.4.0 (StarPRNT SDK v5.6.0), the readPort behavior when a LAN printer is used has been changed as shown below.

When the data that should be received when readPort is executed does not exist
- v2.3.3 and before: Throws a PortException.

- v2.4.0 and later: Returns 0.

3.7.1 For Objective-C

Declaration
- (u_int32_t)readPort:(u_int8_t *)readBuffer

                        :(u_int32_t)offset

                        :(u_int32_t)size

                        :(NSError **)error;

Parameter

Name	Description	Object type
readBuffer	A Byte Array buffer into which data is read.	u_int8_t *
offset	specifies where to begin writing data into the readBuffer	u_int32_t
size	Total number of bytes to read.	u_int32_t
error	Connection failure error information An error code is set in the code property.	NSError **

Return value

Description	Object type
The number of bytes that were read. The readPort method will succeed even when no all data was read in. Your application should call this function a limited number of times until the expected data has been read in or until an application determined retry threshold has been reached.	u_int32_t

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.7.2 For Swift

Declaration

open func read(readBuffer: UnsafeMutablePointer<UInt8>!, offset: UInt32, size: UInt32, numberOfBytesRead: UnsafeMutablePointer<UInt32>!) throws

open func read(_ readBuffer: UnsafeMutablePointer<UInt8>!, _ offSet: UInt32, _ size: UInt32, _ error: NSErrorPointer) -> UInt32

Parameter

Name	Description	Object type
readBuffer	A Byte Array buffer into which data is read.	UnsafeMutablePointer<UInt8>
offset	specifies where to begin writing data into the readBuffer	UInt32
size	Total number of bytes to read.	UInt32
numberOfBytes Read	The number of bytes that were read. The readPort method will succeed even when no all data was read in. Your application should call this function a limited number of times until the expected data has been read in or until an application determined retry threshold has been reached.	UnsafeMutablePointer<UInt32>
error	Connection failure error information An error code is set in the code property.	NSErrorPointer

Return value

Description	Object type
The number of bytes that were read. The readPort method will succeed even when no all data was read in. Your application should call this function a limited number of times until the expected data has been read in or until an application determined retry threshold has been reached.	UInt32

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.8 getParsedStatus Method

Get printer status.

3.8.1 For Objective-C

Declaration
- (NSUInteger)getParsedStatus:(void *)starPrinterStatus

:(u_int32_t)level

:(NSError **)error;

Parameter

Name	Description	Object type
starPrinterStatus	StarPrinterStatus structure giving the current device status. For the type of status that can be obtained, refer to the StarPrinterStatus structure.	void *
level	StarPrinterStatus structure level (Possible to specify a value of 0, 1 or 2. Normally 2 is specified.)	u_int32_t
error	Connection failure error information An error code is set in the code property.	NSError **

Return value

Description	Object type
When this method is successful, returns 1. When this method is failure, returns 0.	NSUInteger

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

Example

```
StarPrinterStatus_2 printerStatus;

NSError *error = nil;

[port getParsedStatus: &printerStatus : 2 : &error];

if (error != nil) {
    // Some error occurred.
}

if (printerStatus.offline == SM_TRUE)
{
    if (printerStatus.coverOpen == SM_TRUE) {
        // Cover is open.
    }
    else if (printerStatus.receiptPaperEmpty == SM_TRUE) {
        // Receipt paper is empty.
    }
    else {
        // The printer is offline.
    }
}
else {
    // The printer is online.
}
```
Refer to DeviceStatusViewController.m.

3.8.2 For Swift

Declaration

open func getParsedStatus(starPrinterStatus: UnsafeMutableRawPointer!, level: UInt32) throws

open func getParsedStatus(_ starPrinterStatus: UnsafeMutableRawPointer!, _ level: UInt32, _ error: NSErrorPointer) -> UInt32

Parameter

Name	Description	Object type
starPrinterStatus	StarPrinterStatus structure giving the current device status. For the type of status that can be obtained, refer to the StarPrinterStatus structure.	UnsafeMutableRawPointer
level	StarPrinterStatus structure level (Possible to specify a value of 0, 1 or 2. Normally 2 is specified.)	UInt32
error	Connection failure error information An error code is set in the code property.	NSErrorPointer

Return value

Description	Object type
When this method is successful, returns 1. When this method is failure, returns 0.	UInt32

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

Example

```swift
var printerStatus: StarPrinterStatus_2 = StarPrinterStatus_2()

do {
    try port.getParsedStatus(starPrinterStatus: &printerStatus, level: 2)
}
catch {
    // Some error occurred.
}

if printerStatus.offline == sm_true {
    if printerStatus.coverOpen == sm_true {
        // Cover is open.
    }
    else if printerStatus.receiptPaperEmpty == sm_true {
        // Receipt paper is empty.
    }
    else {
        // The printer is offline.
    }
}
else {
    // The printer is online.
}
```

Refer to DeviceStatusViewController.swift.

3.9 beginCheckedBlock Method

This method is used in combination with endCheckedBlock and checks the completion of printing. beginCheckedBlock must be run just before sending print data.

Refer to the printing process flow using a SMPort class about the procedure of the print end monitoring process by beginCheckedBlock / endCheckedBlock.

3.9.1 For Objective-C

Declaration
- (NSUInteger)beginCheckedBlock:(void *)starPrinterStatus

                        :(u_int32_t)level

                        :(NSError **)error;

Parameter

Name	Description	Object type
starPrinterStatus	A pointer to StarPrinterStatus structure (Possible to specify StarPrinterStatus, StarPrinterStatus_1 of StarPrinterStatus_2. Normally StarPrinterStatus_2 is specified.) When this method is successful, the status of the current printer is stored.	void *
level	The level of StarPrinterStatus structure (Possible to specify a value of 0, 1 or 2. Normally 2 is specified.)	u_int32_t
error	Connection failure error information An error code is set in the code property.	NSError **

Return value

Description	Object type
When this method is successful, returns 1. When this method is failure, returns 0.	NSUInteger

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.9.2 For Swift

Declaration

open func beginCheckedBlock(starPrinterStatus: UnsafeMutableRawPointer!, level: UInt32) throws

open func beginCheckedBlock(_ starPrinterStatus: UnsafeMutableRawPointer!, _ level: UInt32, _ error: NSErrorPointer) -> UInt32

Parameter

Name	Description	Object type
starPrinterStatus	A pointer to StarPrinterStatus structure (Possible to specify StarPrinterStatus, StarPrinterStatus_1 of StarPrinterStatus_2. Normally StarPrinterStatus_2 is specified.) When this method is successful, the status of the current printer is stored.	UnsafeMutableRawPointer
level	The level of StarPrinterStatus structure (Possible to specify a value of 0, 1 or 2. Normally 2 is specified.)	UInt32
error	Connection failure error information An error code is set in the code property.	NSErrorPointer

Return value

Description	Object type
When this method is successful, returns 1. When this method is failure, returns 0.	NSUInteger

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.10 endCheckedBlock Method

This method is used together with the beginCheckedBlock method in a set and checks the completion of printing.
It monitors printer status and when the transferred data is printed completely, returns control. In case of being transferred other kind of data than print data, when its command is processed in the printer, it returns the control.

Refer to the printing process flow using a SMPort class about the procedure of the print end monitoring process by beginCheckedBlock / endCheckedBlock.

3.10.1 For Objective-C

Declaration
- (NSUInteger)endCheckedBlock:(void *)starPrinterStatus

:(u_int32_t)level

:(NSError **)error;

Parameter

Name	Description	Object type
starPrinterStatus	a pointer to StarPrinterStatus structure (Possible to specify StarPrinterStatus, StarPrinterStatus_1 of StarPrinterStatus_2. Normally StarPrinterStatus_2 is specified.) When this method is successful, the status of the current printer is stored.	void *
level	the level of StarPrinterStatus structure (Possible to specify a value of 0,1 or 2. Normally 2 is specified.)	u_int32_t
error	Connection failure error information An error code is set in the code property.	NSError **

Return value

Description	Object type
When this method is successful, returns 1. When this method is failure, returns 0.	NSUInteger

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

1. Timeout value
To timeout value, endCheckedBlockTimeoutMillis property is applied. Default value is the timeout value designated by getPort. Please adjust the endCheckedBlockTimeoutMillis value to be longer than printing time. Timeout length is specified by getPort, endCheckedBlockTimeoutMillis or is 10 seconds if specified less than 10 seconds.

Declaration

open func endCheckedBlock(starPrinterStatus: UnsafeMutableRawPointer!, level: UInt32) throws

open func endCheckedBlock(_ starPrinterStatus: UnsafeMutableRawPointer!, _ level: UInt32, _ error: NSErrorPointer) -> UInt32

Parameter

Name	Description	Object type
starPrinterStatus	a pointer to StarPrinterStatus structure (Possible to specify StarPrinterStatus, StarPrinterStatus_1 of StarPrinterStatus_2. Normally StarPrinterStatus_2 is specified.) When this method is successful, the status of the current printer is stored.	UnsafeMutableRawPointer
level	the level of StarPrinterStatus structure (Possible to specify a value of 0,1 or 2. Normally 2 is specified.)	UInt32
error	Connection failure error information An error code is set in the code property.	NSErrorPointer

Return value

Description	Object type
When this method is successful, returns 1. When this method is failure, returns 0.	UInt32

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

1. Timeout value
To timeout value, endCheckedBlockTimeoutMillis property is applied. Default value is the timeout value designated by getPort. Please adjust the endCheckedBlockTimeoutMillis value to be longer than printing time. Timeout length is specified by getPort, endCheckedBlockTimeoutMillis or is 10 seconds if specified less than 10 seconds.

3.11 disconnect Method

Disconnect the specified Bluetooth device.
After the disconnection, the Bluetooth device can be connected by other iOS terminals.

3.11.1 For Objective-C

Declaration
- (BOOL)disconnect:(NSError **)error;

Parameter

Name	Description	Object type
error	Connection failure error information An error code is set in the code property.	NSError **

Return value

Description	Object type
Disconnect success and fail This method fails in the following cases: - when the disconnection has not been completed within the timeout specified by getPort - when the disconnection function is not supported by a printer (such like portable printers). This method has no effect on Ethernet devices. It always returns YES when it was run with the Ethernet device.	BOOL

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.11.2 For Swift

Declaration

open func disconnectAccessory() throws

open func disconnect(_ error: NSErrorPointer) -> Bool

Parameter

Name	Description	Object type
error	Connection failure error information An error code is set in the code property.	NSErrorPointer

Return value

Description	Object type
Disconnect success and fail This method fails in the following cases: - when the disconnection has not been completed within the timeout specified by getPort - when the disconnection function is not supported by a printer (such like portable printers). This method has no effect on Ethernet devices. It always returns YES when it was run with the Ethernet device.	Bool

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.12 getFirmwareInformation Method

This method gets a model name and firmware Information of the printer.

3.12.1 For Objective-C

Declaration
- (NSDictionary *)getFirmwareInformation:(NSError **)error;

Parameter

Name	Description	Object type
error	Connection failure error information An error code is set in the code property.	NSError **

Return value

Description	Object type
An acquisition result of firmware information The return value can get the model name by setting the NSDictionary object key of the return value to "@ModelName" and get the firmware version by setting "@FirmwareVersion".	NSDictionary *

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.12.2 For Swift

Declaration

open func getFirmwareInformation() throws -> [AnyHashable : Any]

open func getFirmwareInformation(_ error: NSErrorPointer) -> [AnyHashable : Any]!

Parameter

Name	Description	Object type
error	Connection failure error information An error code is set in the code property.	NSErrorPointer

Return value

Description	Object type
An acquisition result of firmware information The return value can get the model name by setting the Dictionary object key of the return value to "ModelName" and get the firmware version by setting "FirmwareVersion".	[AnyHashable : Any]

Error code

Value	Description
SMStarIOResultCodeFailedError	Some kind of error occurred.

3.13 StarIOVersion Method

This method gets the StarIO version.

Declaration
+ (NSString *) StarIOVersion;

Parameter

Name	Description	Name
-	-	-

Return value

Description	Object type
StarIO version	NSString *

3.14 portName Property

Character string to specify the port for communication with the printer.

Declaration
- (NSString *) portName;

3.15 portSettings Property

Specifies connection setting information.

Declaration
- (NSString *) portSettings;

3.16 timeoutMillis Property

Timeout value for internal control and API. [unit: ms]
Timeout length is 10 seconds if specified less than 10 seconds.

Declaration
- (u_int32_t) timeoutMillis;

3.17 endCheckedBlockTimeoutMillis Property

Timeout value for endCheckedBlock Method. [unit: ms]

If it takes long time to print, stand-by time for print completion in endCheckedBlock method can be extended by increasing this value. Default value is the timeout value designated by getPort method. Timeout length is 10 seconds if specified less than 10 seconds.

When [Data timeout function setting] is used in the portSetting parameter of the getPort method, it should be at least 3 seconds longer than the time specified for the data timeout function. If it is set to less than 3 seconds, it will be controlled internally to be 3 seconds longer automatically.

Declaration
@property (assign, readwrite, nonatomic) u_int32_t endCheckedBlockTimeoutMillis;

3.18 connected Property

If the printer is connected to an iOS device, it returns "YES". If the printer is not connected to an iOS device, it returns "NO".
Only Bluetooth/Bluetooth Low Energy interfaces are supported. For non-supported interfaces, it constantly returns "YES".
Due to the restrictions of the iOS device, it takes approximately 5 seconds from Bluetooth/Bluetooth Low Energy communication disconnecting until being reflected in this property.

Declaration
- (BOOL) connected;

4   StarPrinterStatus structure (StarIO.framework)

Class to maintain the printer status.
Maintains the printer status in both the boolean datatype and binary (raw byte array) formats.
For status types that can be acquired, refer to the field list.

Member

Member name	Contents	Type	Detail
blackMarkError	Black Mark Error	SM_BOOLEAN	" SM_TRUE " : Black mark error occurs. " SM_FALSE " : Black mark error does not occur. When you set printer to Black mark, and print to not Black mark paper, this error occurs.
compulsionSwitch	Compulsion SW	SM_BOOLEAN	You can check status of CashDrawer (Open or Close) " SM_TRUE " : Compulsion SW is pressed. " SM_FALSE " : Compulsion SW is not pressed.
coverOpen	Cover Status	SM_BOOLEAN	You can check status of Cover " SM_TRUE " : Cover is opened. " SM_FALSE " : Cover is closed.
cutterError	Auto-cutter Error	SM_BOOLEAN	You can check status of Cutter " SM_TRUE " : Cutter error occurs. " SM_FALSE " : Cutter error does not occur.
etbAvailable	ETB available or not	SM_BOOLEAN	" SM_TRUE " : available to use " SM_FALSE " : not available to use
etbCounter	ETB Counter	UCHAR	You can get current value of ETB
headThermistorError	Head Thermistor Error	SM_BOOLEAN	You can check status of Head Thermistor. " SM_TRUE " : Head thermistor detects an abnormal value. " SM_FALSE " : Head thermistor does not detect an abnormal value.
offline	ONLINE/OFFLINE Status	SM_BOOLEAN	You can check status of Online or offline. " SM_TRUE " : Printer is Offline. " SM_FALSE " : Printer is Online
overTemp	Stopped by high head temperature	SM_BOOLEAN	" SM_TRUE " : Printer is stopped by head temperature. " SM_FALSE " : Printer is not stopped by head temperature.
raw	Byte column of status	UCHAR[63]	Byte column of status (example : HEX 23 86 00 00 00 00 00 00 00)
rawLength	raw length	CHAR	raw length

Member name	Contents	Type	Detail
receiptPaperEmpty	Paper end	SM_BOOLEAN	" SM_TRUE " : Paper end. " SM_FALSE " : Paper exist.
receiptPaperNearEmptyInner	Paper Near-end (Inner Side)	SM_BOOLEAN	" SM_TRUE " : Paper near-end. " SM_FALSE " : Paper does not near-end.
receiveBufferOverflow	Receive Buffer Overflow	SM_BOOLEAN	You can check status of recieved Buffer. " SM_TRUE " : Received buffer is full. " SM_FALSE " : Received buffer is not full.
unrecoverableError	Non-recoverable Error	SM_BOOLEAN	" SM_TRUE " : Unrecoverable error occurs. " SM_FALSE " : Unrecoverable error does not occur. Unrecoverable error : Head Thermistor Error, Auto-cutter Error, Electric Voltage Error and etc.)
voltageError	Electric Voltage Error	SM_BOOLEAN	" SM_TRUE " : Printers detects an abnormal power supply voltage. " SM_FALSE " : Printers does not detect an abnormal power supply voltage.
connectedInterface	Type of interface that is currently connected	UCHAR	You can check type of interface that is currently connected. 0: USB-B 1: USB-A 2: Bluetooth 3: Ethernet

4.1   Model: StarPrinterStatus Structure

Supported member for each models.

Field	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
blackMarkError	–	–	–	✔	–	–	✔	✔	–	–	–	–	–	–	✔	✔	✔	✔	✔	✔	✔	✔
compulsionSwitch	✔	✔	✔	✔	✔	✔	✔	✔	–	–	–	–	–	✔	–	–	–	–	–	–	–	✔
coverOpen	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
cutterError	✔	✔	✔	✔	✔	✔	✔	✔	–	–	–	–	–	–	–	–	–	–	–	–	–	✔
etbAvailable	✔	✔	✔	✔	✔	✔	✔	✔	–	–	–	–	–	–	✔	✔	✔	✔	✔	✔	✔	✔
etbCounter	✔	✔	✔	✔	✔	✔	✔	✔	–	–	–	–	–	–	✔	✔	✔	✔	✔	✔	✔	✔
headThermistorError	✔	✔	✔	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
offline	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
overTemp	✔	✔	✔	✔	✔	✔	✔	✔	–	–	–	–	–	–	✔	✔	✔	✔	✔	✔	✔	✔
raw	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	✔	✔	✔
rawLength	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	✔	✔	✔
receiptPaperEmpty	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
receiptPaperNearEmptyInner	–	*1	–	✔	–	✔	✔	✔	–	–	–	–	–	–	–	–	–	–	–	–	–	✔
receiveBufferOverflow	–	–	–	✔	–	✔	✔	✔	–	–	–	–	–	–	–	–	–	–	–	–	–	✔
unrecoverableError	✔	✔	✔	✔	✔	✔	✔	✔	–	–	–	–	–	–	–	–	–	–	–	–	–	✔
voltageError	✔	✔	✔	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
connectedInterface	✔	✔	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–

- : ignored
*1: MCP30 is not supported

Star

5   PortInfo class (StarIO.framework)

This class holds information and model information for connecting to the Star printer.

Property

Name	Description
portName	Character string to specify the port for communication with the printer.
macAddress	Character string that indicates the printer's MAC address.
modelName	Character string that indicates the printer's model name.

5.1   portName property

Character string to specify the port for communication with the printer.
Can be used to specify the printer connection with the getPort method.

Declaration
@property (retain, readonly) NSString *portName;

5.2   macAddress property

Character string that indicates the printer's MAC address.

Declaration
@property (retain, readonly) NSString *macAddress;

5.3   modelName property

Character string that indicates the printer's model name.

Declaration
@property (retain, readonly) NSString *modelName;

6 SMBluetoothManager class (StarIO.framework)

SMBluetoothManager Class specifies various settings of the Bluetooth interface.
It can not be used with SMPort Class.

Method

Name	Description
initWithName	Creates an instance of SMBluetoothManager.
Open	Open the connection with the Star Bluetooth device.
loadSetting	Get settings from connected Star Bluetooth device.
Close	Close the communication with the Star Bluetooth device.
Apply	Set the value specified for the Star Bluetooth device.

Property

Name	Description
portName	The portName of the device to be connected.
deviceType	The type of the device to be connected.
Opened	Shows whether the port is opened.
deviceName	The current Bluetooth device name.
iOSPortName	The port name to be used with the StarIO.
autoConnect	The setting (Valid or Invalid) of the autoconnection function.
Security	The Bluetooth security setting.
pinCode	The PIN Code to be used for pairing.
deviceNameCapability	The setting enable / disable information of the Bluetooth device name.
iOSPortNameCapability	The setting enable / disable information of the iOSPor name.
autoConnectCapability	The setting enable / disable information of the AutoConnection.
securityTypeCapability	The setting enable / disable information of the Bluetooth Security Type.
pinCodeCapability	The setting enable / disable information of the PIN Code.

Constant

Name	Description
SMDeviceType	Constants of Printer Type
SMBluetoothSecurity	Constants of Bluetooth security type.
SMBluetoothSettingCapability	Constants of enable / disable information of the Bluetooth security type.

6.1 Bluetooth setting change flow using SMBluetoothManager

Using the SMBluetoothManager class, follow the steps below to change the printer's Bluetooth settings.
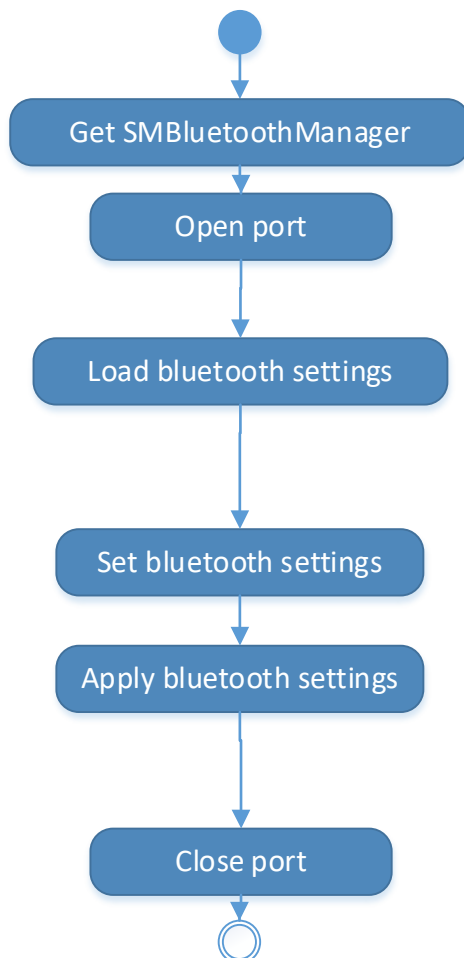
```
// Get SMBluetoothManager
SMBluetoothManager *bluetoothManager =
[SMBluetoothManagerFactory getManager:portName
emulation:emulation];

// Open port
if ([bluetoothManager open] == NO) {
    // Fail to open port.
    return;
}

// Load bluetooth settings
if ([bluetoothManager loadSetting] == NO) {
    // Fail to load settings.
    [bluetoothManager close];
    return;
}

// Set bluetooth settings
// Example) Set the device name to "Star Micronics"
bluetoothManager.deviceName = @"Star Micronics";
// Apply bluetooth settings
if ([bluetoothManager apply] == NO) {
    // Fail to apply settings.
    [bluetoothManager close];
    return;
}

// Close port
[bluetoothManager close];
```

Flow:
- Get SMBluetoothManager
- Open port
- Load bluetooth settings
- Set bluetooth settings
- Apply bluetooth settings
- Close port

Refer to BluetoothSettingViewController.m.

6.2 Model: SMBluetoothManager Class

Supported Method for each models.

Method	mC-Print2	mC-Print3	mPOP *1	FVP10	TSP100 (only TSP100IIIBI)	TSP650II	TSP700II	TSP800II	SM-S210i *2	SM-S220i *2	SM-S230i *2	SM-T300i *2	SM-T400i *2	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
initWithName	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	✔	✔	✔
open	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	✔	✔	✔
loadSetting	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	✔	✔	✔
close	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	✔	✔	✔
apply	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	✔	✔	✔

- : ignored.

*1: F/W Version 1.1 or later is required.

*2: F/W Version 3.0 or later is required.

The properties that can be set (reflected on the Bluetooth device when executing the apply method) in each model are as follows.

Property	mC-Print2	mC-Print3	mPOP *1	FVP10	TSP100 (only TSP100IIIBI)	TSP650II	TSP700II	TSP800II	SM-S210i *2	SM-S220i *2	SM-S230i *2	SM-T300i *2	SM-T400i *2	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
deviceName	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	✔	✔	✔
iOSPortName	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	–	–	✔
pinCode	–	–	–	✔	–	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	✔	✔	✔
autoConnect	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	–	✔	✔	✔	✔	✔	–	–	✔
security	–	–	–	*3	–	*3	*3	*3	*4	*4	*4	*4	*4	–	*4	*4	*4	*4	*4	*4	*4	*3

- : ignored.
*1: F/W Version 1.1 or later is required.
*2: F/W Version 3.0 or later is required.
*3: SecurityType that can be set and acquired is PIN code or SSP
*4: SecurityType that can be set and acquired is PIN code or Disable

6.3 initWithName Method

Creates an instance of SMBluetoothManager.

Declaration
-(id) initWithPortName: (NSString *) portName deviceType: (SMDeviceType) deviceType;

Parameter

Name	Description	Object type
portName	The portName of the device to be connected. It is the same as the portName of the getPort method of the SMPort class.	NSString *
deviceType	The type of the device to be connected.	SMDeviceType

Return value

Description	Object type
It returns Instance of SMBluetoothManager when succeeded. It returns nil when failed.	id

Refer to the printing process flow using a SMBluetoothManager class about the procedure of change the Bluetooth Setting.
Constant of "SMDeviceType" for each model, refer to SMDeviceType Constants.

1. SMBluetoothManagerFactory class
SMBluetoothManagerFactory class can get the appropriate SMBluetoothManager object for the printer emulation.

6.4 open Method

This method is used to open connection to the Bluetooth printer.
Get the current settings by loadSetting method after conducting open method.

Declaration
- (BOOL) open;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
succeeded or NO	BOOL

Refer to the printing process flow using a SMBluetoothManager class about the procedure of change the Bluetooth Setting.

6.5 loadSetting Method

Get settings from connected Star Bluetooth device.

Declaration
- (BOOL) loadSetting;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

Refer to the printing process flow using a SMBluetoothManager class *about the procedure of change the Bluetooth Setting.*

6.6 close Method

Close the communication with the Star Bluetooth device.

Declaration
- (void) close;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

Refer to the printing process flow using a SMBluetoothManager class *about the procedure of change the Bluetooth Setting.*

6.7   apply Method

This method is used to apply the property values of deviceName, iOSPortName, autoConnect, security and pinCode.

Declaration
- (BOOL) apply;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
succeeded or NO	BOOL

1. The values applied with this method are effective after turning the device off and on and paring again.

Refer to the printing process flow using a SMBluetoothManager class *about the procedure of change the Bluetooth Setting.*

6.8   portName Property

The portName of the device to be connected.

Declaration
@property (nonatomic, readonly) NSString *portName;

6.9   deviceType Property

The type of the printer to be connected.

Declaration
@property(nonatomic, readonly) SMDeviceType deviceType;

6.10 opened Property

Shows whether the port is opened.
It returns YES if the open method was successful. Then it will return NO when the close method is called.

Declaration
@property(nonatomic, readonly) BOOL opened;

6.11 deviceName Property

Acquires and specifies the current Bluetooth device name.
This name is displayed when you are pairing via Bluetooth. When using Bluetooth Low Energy, it is used as a connection port name for communication. The current setting is read when the loadSetting method is called. To set it, run the apply method after changing this property.

Declaration
@property(nonatomic, retain) NSString *deviceName;

1. Valid characters
0-9 a-z A-Z ; : ! ? # $ % & , . @ _ - = Space / * + ~ ^ [{ ( ] }) | \

2. Valid number of characters
1 to 16

3. Notes
In case of Bluetooth Low Energy, the changed Bluetooth names are effective after turning the device off and on and connecting again.

6.12 iOSPortName Property

Acquires and specifies the iOS port name to be used with the StarIO for Bluetooth communication.
This function is not used with Bluetooth Low Energy.
The current setting is read when the loadSetting method is called. To set it, run the apply method after changing this property.

Declaration
@property(nonatomic, retain) NSString *iOSPortName;

1. Valid characters
0-9 a-z A-Z ; : ! ? # $ % & , . @ _ - = Space / * + ~ ^ [{ ( ] }) | \

2. Valid number of characters
1 to 16

6.13 autoConnect Property

Acquires and specifies the setting of the auto connection function.
This function is available only for Bluetooth. The current setting is read when the loadSetting method is called. To set it, run the apply method after changing this property.

Declaration
@property(nonatomic, assign) BOOL autoConnect;

1. Notes
Set to NO when the security setting is set to PIN code mode.

6.14 Security Property

Acquires and specifies the Bluetooth security setting.
This function is available only for Bluetooth.
The current setting is read when the open method is called. To set it, run the apply method after changing this property.

Declaration
@property(nonatomic, assign) SMBluetoothSecurity security;

6.15 pinCode Property

Specifies the PIN code of the Bluetooth interface.
This function is available only for Bluetooth.
It can not acquire the current setting.
Set to nil when the PIN code is not changed.

Declaration
@property(nonatomic, retain) NSString *pinCode;

1. Valid characters
0-9 a-z A-Z (except SM-L200 and SM-L300)
0-9 (SM-L200, SM-L300)

2. Valid number of characters
4 to 16 (except SM-L200 and SM-L300)
4 digits (SM-L200, SM-L300)

6.16 deviceNameCapability Property

The setting enable / disable information of the Bluetooth device name.

Declaration
@property (assign, readonly) SMBluetoothSettingCapability deviceNameCapability;

6.17 iOSPortNameCapability Property

The setting enable / disable information of the iOSPort name.

Declaration
@property (assign, readonly) SMBluetoothSettingCapability iOSPortNameCapability;

6.18 autoConnectCapability Property

The setting enable / disable information of the AutoConnection.

Declaration
@property (assign, readonly) SMBluetoothSettingCapability autoConnectCapability;

6.19 securityTypeCapability Property

The setting enable / disable information of the Bluetooth security type.

Declaration
@property (assign, readonly) SMBluetoothSettingCapability securityTypeCapability;

6.20 pinCodeCapability Property

The setting enable / disable information of the PIN code.

Declaration
@property (assign, readonly) SMBluetoothSettingCapability pinCodeCapability;

6.21 SMDeviceType Constants

Constants of Printer Type.

Declaration
typedef enum _SMDeviceType {
    SMDeviceTypeUnknown = 0,
    SMDeviceTypeDesktopPrinter,
    SMDeviceTypePortablePrinter,
    SMDeviceTypeDKAirCash,
} SMDeviceType;

Constants

Name	Description
SMDeviceTypeDesktopPrinter	Desktop Printer
SMDeviceTypePortablePrinter	Portable Printer

Constant of "SMDeviceType" for each model.

Model	SMDeviceType
TSP100	SMDeviceTypeDesktopPrinter
FVP10	
TSP650II	
TSP700II	
TSP800II	
SP700	
mC-Print2	SMDeviceTypePortablePrinter
mC-Print3	
mPOP	
SM-S210i	
SM-S220i	
SM-S230i	
SM-T300i	
SM-T400i	
SM-S210i StarPRNT	
SM-S220i StarPRNT	
SM-S230i StarPRNT	
SM-T300i StarPRNT	
SM-T400i StarPRNT	
SM-L200	
SM-L300	

6.22 SMBluetoothSecurity Constants

Constants of Bluetooth Security type.

Declaration
typedef enum _SMBluetoothSecurity {
   SMBluetoothSecurityDisable,
   SMBluetoothSecuritySSP,
   SMBluetoothSecurityPINcode
} SMBluetoothSecurity;

Constants

Name	Description
SMBluetoothSecurityPINcode	Security by entering PIN code
SMBluetoothSecuritySSP	Security by pressing the interface card button
SMBluetoothSecurityDisable	No security

6.23 SMBluetoothSettingCapability Constants

Constants of enable / disable information of the Bluetooth security type.

Declaration

typedef enum _SMBluetoothSettingCapability {
   SMBluetoothSettingCapabilitySupport,
   SMBluetoothSettingCapabilityNoSupport
} SMBluetoothSettingCapability;

Constants

Name	Description
SMBluetoothSettingCapabilitySupport	Indicates the items and functions can be set for the printer that is currently connected.
SMBluetoothSettingCapabilityNoSupport	Indicates the items and functions can be set for the printer that is currently connected.

7 SMLogger class (StarIO.framework)

This class provides the function which outputs the StarIO.framework and StarIO_Extension.framework logs.
Use of this Log function may have some influences on performance of your application.

Method

Name	Description
start	Starts log output.
stop	Stops log output.
logString	Outputs log of the desired string.

Property

Name	Description
logLevel	Sets or gets the log level.
isLogging	Gets whether log output has started or not.

Constants

Name	Description
SMLogLevel	Constant that specifies the log level.

7.1 start Method

Starts log output.

7.1.1 For Objective-C

Declaration
- (void) start;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

7.1.2 For Swift

Declaration
open func start()

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

7.2 stop Mothod

Stops log output.

7.2.1 For Objective-C

Declaration

- (void) stop;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

7.2.2 For Swift

Declaration

open func stop()

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

7.3   logString Method

Outputs log of the desired string.

7.3.1   For Objective-C

Declaration

- (void) logString:(nonnull NSString *)string;

Parameter

Name	Description	Object type
string	String for log output	NSString

Return value

Description	Object type
-	-

7.3.2   For Swift

Declaration

open func log(_ string: String)

Parameter

Name	Description	Object type
string	String for log output	String

Return value

Description	Object type
-	-

7.4 logLevel Property

Sets or gets the log level.

7.4.1 For Objective-C

Declaration

@property (nonatomic) SMLogLevel logLevel;

Value

Description	Object type
Log level	SMLogLevel

7.4.2 For Swift

Declaration

open var logLevel: SMLogLevel

Value

Description	Object type
Log level	SMLogLevel

7.5 isLogging Property

Gets whether log output has started or not.

7.5.1 For Objective-C

Declaration

@property (nonatomic, readonly) BOOL isLogging;

Value

Description	Object type
Whether log output has started or not	BOOL

7.5.2 For Swift

Declaration

open var isLogging: Bool { get }

Value

Description	Object type
Whether log output has started or not	Bool

7.6 SMLogLevel Constants

Constant that specifies the log output level.

7.6.1 For Objective-C

Declaration
typedef NS_ENUM(NSUInteger, SMLogLevel) {
   SMLogLevelDefault
};

Constants

Name	Description
SMLogLevelDefault	Default level

7.6.2 For Swift

Declaration
public enum SMLogLevel : Int {
      case default
}

Constants

Name	Description
default	Default level

8   SMFileLogger class (StarIO.framework)

This class provides the function which outputs the StarIO.framework and StarIO_Extension.framework logs to files. It extends the [SMLogger class](#).

Property

Name	Description
sharedInstance	The singleton instance of the SMFileLogger class.
maxLogSize	Specifies the maximum log size to be kept.
logDirectoryPath	The directory path of log files to save.

The output log file is saved under the Documents/starlog directory of the application. You can access the saved file by using the file sharing function.

Ex.1: When using iTunes file sharing
It is necessary to set items to Information Property List. The setting procedure is as follows.
1. Select the Information Property List ("Info.plist" by default).
2. Added Key "Application supports iTunes file sharing"
3. Set "YES" to [Value] of "Item 0" displayed by clicking ▽ on the left side of the item name

Ex. 2: Using opening documents in place
It is necessary to set items to Information Property List. The setting procedure is as follows.
1. Select the Information Property List ("Info.plist" by default).
2. Added Key "Supports opening documents in place"
3. Set "YES" to [Value] of "Item 0" displayed by clicking ▽ on the left side of the item name

8.1 sharedInstance Property

The singleton instance of the SMFileLogger class.

8.1.1 For Objective-C

Declaration

@property (class, nonatomic, readonly, nonnull) SMFileLogger *sharedInstance;

Value

Description	Object type
The singleton class of the SMFileLogger instance.	SMFileLogger

8.1.2 For Swift

Declaration

open class var shared: SMFileLogger { get }

Value

Description	Object type
The singleton class of the SMFileLogger instance.	SMFileLogger

8.2 maxLogSize Property

Specifies the maximum log size to be kept.

8.2.1 For Objective-C

Declaration

@property(nonatomic) NSUInteger maxLogSize;

Value

Description	Object type
Maximum log size to be kept (in MB) Default value: 50 (MB) If 0 is specified, no limit on log size is set.	NSUInteger

8.2.2 For Swift

Declaration

open var maxLogSize: UInt

Value

Description	Object type
Maximum log size to be kept (in MB) Default value: 50 (MB) If 0 is specified, no limit on log size is set.	UInt

8.3 logDirectoryPath Property

The directory path of log files to save.

8.3.1 For Objective-C

Declaration

@property (nonatomic, readonly) NSString *logDirectoryPath;

Value

Description	Object type
The directory path of log files to save "Documents/starlog" is set.	NSString

8.3.2 For Swift

Declaration

open var logDirectoryPath: String { get }

Value

Description	Object type
The directory path of log files to save "Documents/starlog" is set.	String

8.4 Sample of log output using the SMFileLogger class

8.4.1 For Objective-C

```
// Get SMFileLogger class singleton instance.
SMFileLogger *logger = SMFileLogger.sharedInstance;

// Set the log level.
logger.logLevel = SMLogLevelDefault;

// Set the maximum log size to be kept (optional).
logger.maxLogSize = 10;

// Start log output.
[logger start];

// Output the desired string (optional).
[logger logString:@"(Log message)"];

// Stop log output.
[logger stop];
```

8.4.2 For Swift

```
// Get SMFileLogger class singleton instance.
let logger = SMFileLogger.shared

// Set the log level.
logger.logLevel = .default

// Set the maximum log size to be kept (optional).
logger.maxLogSize = 10

// Start log output.
logger.start()

// Output the desired character string (optional).
logger.log("(Log message)")

// Stop log output.
logger.stop()
```

9 StarIoExt class (StarIO_Extension.framework)

Method

Name	Description
createCommandBuilder	Creates the ISCBBuilder object.
createDisplayCommandBuilder	Creates the customer display command builder object.
createMelodySpeakerCommandBuilder	Creates the melody speaker command builder object.
createBcrConnectParser	Creates a barcode reader connection status command response analysis object.
createDisplayConnectParser	Creates a customer display connection status command response analysis object.
createMelodySpeakerConnectParser	Creates a melody speaker connection status command response analysis object.

Constants

Name	Description
StarIoExtEmulation	Emulation type constants.
StarIoExtCharacterCode	Character Code constants.
StarIoExtBcrModel	Barcode Reader Model constants.
StarIoExtDisplayModel	Customer Display Model constants.
StarIoExtMelodySpeakerModel	Melody Speaker Model constants.

9.1 createCommandBuilder Method

Creates the ISCBBuilder object.

Declaration

+ (ISCBBuilder *)createCommandBuilder:(StarIoExtEmulation)emulation;

Parameter

Name	Description	Object type
emulation	Emulation type. • StarIoExtEmulationStarPRNT … StarPRNT emulation. • StarIoExtEmulationStarLine … STAR Line Mode emulation. • StarIoExtEmulationStarGraphic … STAR Graphic Mode emulation. • StarIoExtEmulationEscPos … ESC/POS emulation. • StarIoExtEmulationEscPosMobile … ESC/POS Mobile emulation. • StarIoExtEmulationStarDotImpact … STAR Dot Impact emulation. • StarIoExtEmulationStarPRNTL… StarPRNTL emulation	StarIoExtEmulation

Return value

Description	Object type
ISCBBuilder object.	ISCBBuilder

Constant of "StarIoExtEmulation" for each model, refer to StarIoExtEmulation Constants.

Example

```
+ (NSData *)createCommandsImage:(StarIoExtEmulation)emulation image:(UIImage *)image {
    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendBitmap:image diffusion:NO];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to PrinterFunctions.m.

9.2 createDisplayCommandBuilder Method

Creates the customer display command builder object.

Declaration

+ (ISDCBBuilder *)createDisplayCommandBuilder:(StarIoExtDisplayModel)model;

Parameter

Name	Description	Object type
model	Customer Display Model constant ♣ SCD222	StarIoExtDisplayModel

Return value

Description	Object type
ISDCBBuilder object	ISDCBBuilder *

Example

```
- (void)alertView:(UIAlertView *)alertView didDismissWithButtonIndex:(NSInteger)buttonIndex {
    if (buttonIndex != alertView.cancelButtonIndex) {
        ISDCBBuilder *builder = [StarIoExt createDisplayCommandBuilder:StarIoExtDisplayModelSCD222];

        …
}
```

Refer to DisplayViewController.m / DisplayExtViewController.m.

9.3   createMelodySpeakerCommandBuilder Method
Creates the melody speaker command builder object.

Declaration
+ (ISMCBBuilder *)createMelodySpeakerCommandBuilder:(StarIoExtMelodySpeakerModel)model;

Parameter

Name	Description	Object type
model	Melody Speaker Model constant ♣ StarIoExtMelodySpeakerModelMCS10 ♣ StarIoExtMelodySpeakerModelFVP10	StarIoExtMelodySpeakerModel

Return value

Description	Object type
ISMCBBuilder object	ISMCBBuilder *

Example
```
+ (NSData *)createPlayingRegisteredSound:(StarIoExtMelodySpeakerModel)model
              specifySound:(BOOL)specifySound
          soundStorageArea:(SMCBSoundStorageArea)soundStorageArea
              soundNumber:(NSInteger)soundNumber
            specifyVolume:(BOOL)specifyVolume
                    volume:(NSInteger)volume
                    error:(NSError * _Nullable * _Nullable) error {
    ISMCBBuilder *builder = [StarIoExt createMelodySpeakerCommandBuilder:model];

    SMSoundSetting *setting = [SMSoundSetting new];

    if (specifySound == YES) {
        [setting setSoundStorageArea:soundStorageArea];
        [setting setSoundNumber:soundNumber];
    }

    if (specifyVolume == YES) {
        [setting setVolume:volume];
    }

    [builder appendSoundWithSetting:setting
                    error:error];

    if (*error != nil) {
        return nil;
    }

    return [builder.commands copy];
}
```

Refer to MelodySpeakerViewController.m

9.4　createBcrConnectParser Method

Creates a barcode reader connection status command response analysis object.

Declaration

+ (ISCPConnectParser *)createBcrConnectParser:(StarIoExtBcrModel)model;

Parameter

Name	Description	Object type
model	Barcode Reader Model constant ♠　StarIoExtBcrModelPOP1	StarIoExtBcrModel

Return value

Description	Object type
ISCPConnectParser object	ISCPConnectParser *

9.5   createDisplayConnectParser Method

Creates a customer display connection status command response analysis object.

Declaration

+ (ISCPConnectParser *)createDisplayConnectParser:(StarIoExtDisplayModel)model;

Parameter

Name	Description	Object type
model	Customer Display Model constant ♣ SCD222	StarIoExtDisplayModel

Return value

Description	Object type
ISCPConnectParser object	ISCPConnectParser *

Example

```
- (void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)indexPath {
  …

    ISCPConnectParser *parser = [StarIoExt createDisplayConnectParser:StarIoExtDisplayModelSCD222];

    [Communication parseDoNotCheckCondition:parser
                    port:port
              completionHandler:^(CommunicationResult *communicationResult) {
                  if (communicationResult.result == CommResultSuccess) {
                      if (parser.connect == YES) {
                          [self showSimpleAlertWithTitle:@"Check Status"
                                    message:@"Display Connect."
                                     buttonTitle:@"OK"
                                     buttonStyle:UIAlertActionStyleCancel
                                      completion:nil];
                      }
                      else {
                          [self showSimpleAlertWithTitle:@"Check Status"
                                    message:@"Display Disconnect."
                                     buttonTitle:@"OK"
                                     buttonStyle:UIAlertActionStyleCancel
                                      completion:nil];
                      }
                  }
                  ...
              }];
  …
}
```

Refer to DisplayViewController.m / DisplayExtViewController.m.

9.6 createMelodySpeakerConnectParser Method

Creates a melody speaker connection status command response analysis object.

Declaration

+ (ISCPConnectParser
*)createMelodySpeakerConnectParser:(StarIoExtMelodySpeakerModel)model error:(nullable
NSError **)error;

Parameter

Name	Description	Object type
model	Melody Speaker Model constant • StarIoExtMelodySpeakerModelMCS10 • StarIoExtMelodySpeakerModelFVP10 (Not supported)	StarIoExtMelodySpeakerModel
error	Error information when object generation failed	NSError **

Return value

Description	Object type
ISCPConnectParser object	ISCPConnectParser *

Error code

Value	Description
StarIoExtErrorInvalidOperation	StarIoExtMelodySpeakerModelFVP10 is specified

Example

```
if (_melodySpeakerModel == StarIoExtMelodySpeakerModelMCS10) {
    NSError *parserError = nil;

    ISCPConnectParser *parser = [StarIoExt createMelodySpeakerConnectParser:_melodySpeakerModel
                                 error:&parserError];

    [Communication parseDoNotCheckCondition:parser
                    port:port
              completionHandler:^(CommunicationResult *communicationResult) {
                  if (communicationResult.result == CommResultSuccess) {
                      if (parser.connect == YES) {
                          ...
                      }
                      ...
                  }
                  ...
              }];
}
```

Refer to MelodySpeakerViewController.m.

9.7 StarIoExtEmulation Constants

Emulation type constants.

Declaration
typedef NS_ENUM(NSInteger, StarIoExtEmulation) {
    StarIoExtEmulationNone = 0,
    StarIoExtEmulationStarPRNT,
    StarIoExtEmulationStarLine,
    StarIoExtEmulationStarGraphic,
    StarIoExtEmulationEscPos,
    StarIoExtEmulationEscPosMobile,
    StarIoExtEmulationStarDotImpact
    StarIoExtEmulationStarPRNTL
};

Constants

Name	Description
StarIoExtEmulationStarPRNT	StarPRNT emulation.
StarIoExtEmulationStarLine	STAR Line Mode emulation.
StarIoExtEmulationStarGraphic	STAR Graphic Mode emulation.
StarIoExtEmulationEscPos	ESC/POS emulation.
StarIoExtEmulationEscPosMobile	ESC/POS Mobile emulation.
StarIoExtEmulationStarDotImpact	STAR Dot Impact emulation.
StarIoExtEmulationStarPRNTL	StarPRNTL emulation.

1. StarPRNTL emulation
StarPRNTL emulation reduces the command size with the appendBitmap-type method that generates a raster image printing command.
The time it takes to send the raster image command is reduced.
StarPRNTL emulation can only be selected on some supported models. For usable models, refer to the supported printers.
Constant of "StarIoExtEmulation" for each model.

Model	StarIoExtEmulation
mC-Print2	StarIoExtEmulationStarPRNT
mC-Print3	StarIoExtEmulationStarPRNT
mPOP	StarIoExtEmulationStarPRNT
FVP10	StarIoExtEmulationStarLine
TSP100	StarIoExtEmulationStarGraphic
TSP650II	StarIoExtEmulationStarLine
TSP700II	StarIoExtEmulationStarLine
TSP800II	StarIoExtEmulationStarLine
SM-S210i	StarIoExtEmulationEscPosMobile
SM-S220i	StarIoExtEmulationEscPosMobile
SM-S230i	StarIoExtEmulationEscPosMobile
SM-T300i/T300	StarIoExtEmulationEscPosMobile
SM-T400i	StarIoExtEmulationEscPosMobile
BSC10	StarIoExtEmulationEscPos
SM-S210i StarPRNT	StarIoExtEmulationStarPRNT
SM-S220i StarPRNT	StarIoExtEmulationStarPRNT
SM-S230i StarPRNT	StarIoExtEmulationStarPRNT
SM-T300i/T300 StarPRNT	StarIoExtEmulationStarPRNT
SM-T400i StarPRNT	StarIoExtEmulationStarPRNT

9.8 StarIoExtCharacterCode Constants

Character code type constants.

Declaration
typedef NS_ENUM(NSInteger, StarIoExtCharacterCode) {
    StarIoExtCharacterCodeNone = 0,
    StarIoExtCharacterCodeStandard,
    StarIoExtCharacterCodeJapanese,
    StarIoExtCharacterCodeSimplifiedChinese,
    StarIoExtCharacterCodeTraditionalChinese
};

Constants

Name	Description
StarIoExtCharacterCodeStandard	Standard character code.
StarIoExtCharacterCodeJapanese	Japanese character code.
StarIoExtCharacterCodeSimplifiedChinese	Simplified chinese character code.
StarIoExtCharacterCodeTraditionalChinese	Traditional chinese character code.

9.9 StarIoExtBcrModel Constant

Barcode Reader Model constants.

Declaration
typedef NS_ENUM(NSInteger, StarIoExtBcrModel) {
    StarIoExtBcrModelNone = 0,
    StarIoExtBcrModelPOP1,
    StarIoExtBcrModelDS9208
};

Constants

Name	Description
StarIoExtBcrModelPOP1	Barcode reader : BCR-POP1
StarIoExtBcrModelDS9208	Barcode reader : DS9208

9.10 StarIoExtDisplayModel Constant

Customer Display Model constants.

Declaration
typedef NS_ENUM(NSInteger, StarIoExtDisplayModel) {
    StarIoExtDisplayModelNone = 0,
    StarIoExtDisplayModelSCD222
};

Constants

Name	Description
StarIoExtDisplayModelSCD222	Customer display: SCD222U

9.11 StarIoExtMelodySpeakerModel Constant

Melody Speaker Model constants.

Declaration

```
typedef NS_ENUM(NSUInteger, StarIoExtMelodySpeakerModel) {
    StarIoExtMelodySpeakerModelMCS10,
    StarIoExtMelodySpeakerModelFVP10
};
```

Constants

Name	Description
StarIoExtMelodySpeakerModelMCS10	Melody speaker: MCS10
StarIoExtMelodySpeakerModelFVP10	FVP10 internal speaker

10 ISCBBuilder interface (StarIO_Extension.framework)

Method

Name	Description
beginDocument	Begin document command is generated and added to the commands property.
endDocument	End document command is generated and added to the commands property.
appendInitialization	Initialization command is generated and added to the commands property.
appendByte appendData appendBytes	Data is added to the commands property.
appendRawByte appendRawData appendRawBytes	Raw data is added to the commands property.
appendFontStyle	Select command of the font style is generated and added to the commands property.
appendCodePage	Select command of the code page is generated and added to the commands property.
appendInternational	Select command of the international character mode is generated and added to the commands property.
appendLineFeed appendDataWithLineFeed appendBytesWithLineFeed	Line feed command is generated and added to the commands property.
appendUnitFeed appendDataWithUnitFeed appendBytesWithUnitFeed	Unit feed command is generated and added to the commands property.
appendCharacterSpace	Set command of the character space is generated and added to the commands property.
appendLineSpace	Set command of the line space is generated and added to the commands property.
appendTopMargin	Set command of the top margin is generated and added to the commands property.
appendEmphasis appendDataWithEmphasis appendBytesWithEmphasis	Select command of the emphasis mode is generated and added to the commands property.
appendInvert appendDataWithInvert appendBytesWithInvert	Select command of the invert mode is generated and added to the commands property.
appendMultiple appendDataWithMultiple appendBytesWithMultiple appendMultipleHeight appendDataWithMultipleHeight appendBytesWithMultipleHeight appendMultipleWidth appendDataWithMultipleWidth	Select command of the multiple mode is generated and added to the commands property.

Name	Description
appendBytesWithMultipleWidth	
appendUnderLine	Select command of the under line mode is generated and added to the commands property.
appendDataWithUnderLine	
appendBytesWithUnderLine	
appendLogo	Print command of the logo is generated and added to the commands property.
appendAbsolutePosition	Absolute position command is generated and added to the commands property.
appendDataWithAbsolutePosition	
appendBytesWithAbsolutePosition	
appendAlignment	Alignment command is generated and added to the commands property.
appendDataWithAlignment	
appendBytesWithAlignment	
appendHorizonalTabPosition	Horizontal tab set/clear command is generated and added to the commands property.
appendCutPaper	Paper cut command is generated and added to the commands property.
appendPeripheral	Cash drawer command is generated and added to the commands property.
appendSound	Sound/Buzzer command is generated and added to the commands property.
appendBarcodeData	Print command of the barcode is generated and added to the commands property.
appendBarcodeBytes	
appendBarcodeDataWithAbsolutePosition	Print command of the absolute position barcode is generated and added to the commands property.
appendBarcodeBytesWithAbsolutePosition	
appendBarcodeDataWithAlignment	Print command of the alignment barcode is generated and added to the commands property.
appendBarcodeBytesWithAlignment	
appendPdf417Data	Print command of the PDF417 is generated and added to the commands property.
appendPdf417Bytes	
appendPdf417DataWithAbsolutePosition	Print command of the absolute position PDF417 is generated and added to the commands property.
appendPdf417BytesWithAbsolutePosition	
appendPdf417DataWithAlignment	Print command of the alignment PDF417 is generated and added to the commands property.
appendPdf417BytesWithAlignment	
appendQrCodeData	Print command of the QR code is generated and added to the commands property.
appendQrCodeBytes	
appendQrCodeDataWithAbsolutePosition	Print command of the absolute position QR code is generated and added to the commands property.
appendQrCodeBytesWithAbsolutePosition	
appendQrCodeDataWithAlignment	Print command of the alignment QR code is generated and added to the commands property.
appendQrCodeBytesWithAlignment	
appendBitmap	Print command of the bitmap is generated and added to the commands property.
appendBitmapWithAbsolutePosition	Print command of the absolute position bitmap is generated and added to the commands property.

Name	Description
appendBitmapWithAlignment	Print command of the alignment bitmap is generated and added to the commands property.
appendBlackMark	Black mark command is generated and added to the commands property.
beginPageMode	Begin page mode command is generated and added to the commands property.
endPageMode	End page mode command is generated and added to the commands property.
appendPageModeVerticalAbsolutePosition	Vertical absolute position in page mode command is generated and added to the commands property.
appendPageModeRotation	Print direction in page mode command is generated and added to the commands property.
appendPrintableArea	Set command of the printable area is generated and added to the commands property.
appendCjkUnifiedIdeographFont	A command to set the UTF-8 CJK Unified Ideograph font is generated and added to the commands property.

Property

Name	Description
commands	Generated commands.

Constants

Name	Description
SCBInitializationType	Initialization constants.
SCBFontStyleType	Font style constants.
SCBCodePageType	Code Page constants.
SCBInternationalType	International character constants.
SCBLogoSize	Logo size constants.
SCBAlignmentPosition	Alignment position constants.
SCBCutPaperAction	Paper cut constants.
SCBPeripheralChannel	Cash drawer channel constants.
SCBSoundChannel	Sound/Buzzer channel constants.
SCBBarcodeSymbology	Barcode symbology constants.
SCBBarcodeWidth	Barcode width constants.
SCBPdf417Level	PDF417 ECC (security level) constants.
SCBQrCodeModel	QR code model constants.
SCBQrCodeLevel	QR code mistake correction level constants.
SCBBitmapConverterRotation	Bitmap rotation constants.
SCBBlackMarkType	Black mark constants.
SCBPrintableAreaType	Printable area constants.

10.1 Model : ISCBBuilder interface Method

Supported Method for each models.

Function	Method	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
Document control	beginDocument	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	endDocument	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Initialization	appendInitialization	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Data (Text and Command)	appendByte	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendData	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytes	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Raw data (Text and Command)	appendRawByte	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendRawData	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendRawBytes	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Font style	appendFontStyle	✔	✔	✔	✔	-	✔	✔	✔	-	-	-	-	-	✔	✔	✔	✔	✔	✔	✔	✔	✔
Code page	appendCodePage	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
International	appendInternational	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Line feed	appendLineFeed	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendDataWithLineFeed	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithLineFeed	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Unit feed	appendUnitFeed	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendDataWithUnitFeed	✔	✔	✔	✔	*1	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithUnitFeed	✔	✔	✔	✔	*1	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Character space	appendCharacterSpace	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Line space	appendLineSpace	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Top margin	appendTopMargin	-	*2 *3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Emphasis	appendEmphasis	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔

Function	Method	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
	appendDataWithEmphasis	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithEmphasis	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Invert	appendInvert	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendDataWithInvert	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithInvert	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Multiple	appendMultiple	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendDataWithMultiple	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithMultiple	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendMultipleHeight	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendDataWithMultipleHeight	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithMultipleHeight	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendMultipleWidth	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendDataWithMultipleWidth	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithMultipleWidth	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Under line	appendUnderLine	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendDataWithUnderLine	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithUnderLine	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Logo	appendLogo	✔	✔	✔	✔	-	✔	✔	✔	*4	*4	*4	*4	*4	✔	✔	✔	✔	✔	✔	✔	✔	✔
Absolute position	appendAbsolutePosition	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendDataWithAbsolutePosition	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithAbsolutePosition	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Alignment	appendAlignment	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendDataWithAlignment	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBytesWithAlignment	✔	✔	✔	✔	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔

Function	Method	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
Horizontal Tab position	appendHorizontalTabPosition	✔	✔	✔	✔	-	✔	✔	✔	-	-	-	-	-	✔	✔	✔	✔	✔	✔	✔	✔	✔
Cut paper	appendCutPaper	✔	✔	✔	✔	✔	✔	✔	✔	*5	*5	*5	*5	*5	✔	*5	*5	*5	*5	*5	*5	*5	✔
Peripheral	appendPeripheral	✔	✔	✔	✔	✔	✔	✔	✔	-	-	-	-	-	✔	-	-	-	-	-	-	-	✔
Sound	appendSound	✔	✔	✔	✔	✔	✔	✔	✔	-	-	-	-	-	✔	-	-	-	-	-	-	-	✔
Barcode *6	appendBarcodeData	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendBarcodeBytes	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendBarcodeDataWithAbsolutePosition	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendBarcodeBytesWithAbsolutePosition	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendBarcodeDataWithAlignment	✔	✔	✔	✔	*7	✔	✔	✔	*7	*7	*7	*7	*7	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendBarcodeBytesWithAlignment	✔	✔	✔	✔	*7	✔	✔	✔	*7	*7	*7	*7	*7	✔	✔	✔	✔	✔	✔	✔	✔	-
PDF417	appendPdf417Data	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendPdf417Bytes	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendPdf417DataWithAbsolutePosition	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendPdf417BytesWithAbsolutePosition	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendPdf417DataWithAlignment	✔	✔	✔	✔	*7	✔	✔	✔	*7	*7	*7	*7	*7	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendPdf417BytesWithAlignment	✔	✔	✔	✔	*7	✔	✔	✔	*7	*7	*7	*7	*7	✔	✔	✔	✔	✔	✔	✔	✔	-
QR code	appendQrCodeData	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendQrCodeBytes	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendQrCodeDataWithAbsolutePosition	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendQrCodeBytesWithAbsolutePosition	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-

Function	Method	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
	appendQrCodeDataWithAlignment	✔	✔	✔	✔	*7	✔	✔	✔	*7	*7	*7	*7	*7	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendQrCodeBytesWithAlignment	✔	✔	✔	✔	*7	✔	✔	✔	*7	*7	*7	*7	*7	✔	✔	✔	✔	✔	✔	✔	✔	-
Bitmap	appendBitmap	✔	✔	✔	✔	✔	✔	✔	✔	*8	*8	*8	*8	*8	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBitmapWithAbsolutePosition	✔	✔	✔	✔	✔	✔	✔	✔	*8	*8	*8	*8	*8	✔	✔	✔	✔	✔	✔	✔	✔	✔
	appendBitmapWithAlignment	✔	✔	✔	✔	*7	✔	✔	✔	*7 *8	*7 *8	*7 *8	*7 *8	*7 *8	✔	✔	✔	✔	✔	✔	✔	✔	✔
Black mark	appendBlackMark	-	-	-	✔	-	-	✔	✔	-	-	-	*9	*9	-	-	-	-	*10	*10	*10	*9	✔
Page mode	beginPageMode	✔	✔	✔	✔	-	✔	-	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	endPageMode	✔	✔	✔	✔	-	✔	-	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendPageModeVerticalAbsolutePosition	✔	✔	✔	✔	-	✔	-	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
	appendPageModeRotation	✔	✔	✔	✔	-	✔	-	-	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	-
Printable area	appendPrintableArea	✔	✔	✔	✔	✔	✔	✔	✔	–	–	–	–	–	✔	–	–	–	–	–	–	✔	✔
CJK Unified Ideograph Font	appendCjkUnifiedIdeographFont	✔	✔	–	–	–	*11	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–

- : ignored.

*1: Do not append data, append unit feed command only.

*2: When the top margin is set to 10 mm or less, the paper length (cut length) should not exceed 50mm.
   (If the cut paper remains at the paper-exit, a paper jam may occur. There is no limitation when the cut paper is removed.)

*3: MCP30 is not supported.

*4: Always print from normal size.

*5: Without paper cut.

*6: Confirm that the printed barcode can be read properly in real environment.

*7: Always print from left side.

*8: The maximum vertical direction printing area is 30cm (2400 dot).

*9: Need to change printer setting to enable the black mark.
　　Refer to hardware manual for how to change.
*10: Only the front black mark can set enabled or disabled.
　　Need to change printer setting to enable the back black mark (SM-T300i/T300 does not support the back black mark).
　　Refer to hardware manual for how to change.
*11: JP2/TW model with Firmware version 4.0 or later is required.

10.2 beginDocument Method

Begin document command is generated and added to the commands property.

Declaration
- (void)beginDocument;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.3 endDocument Method

End document command is generated and added to the commands property.

Declaration
- (void)endDocument;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.4 appendInitialization Method

Initialization command is generated and added to the commands property.

Declaration
- (void)appendInitialization:(SCBInitializationType)type;

Parameter

Name	Description	Object type
type	Initialization type. • SCBInitializationTypeCommand … Command initialization. • ~~SCBInitializationTypeReset … Reset printer.~~ • ~~SCBInitializationTypeResetWithPrint … Reset printer (execute self print).~~	SCBInitializationType

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendUnderLine:YES];
    [builder appendMultiple:2 height:2];
    [builder appendData:otherData];

    [builder appendInitialization:SCBInitializationTypeCommand];

    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.5 appendData Method

Data (Text and Command) is added to the commands property.

Declaration

- (void)appendByte:(unsigned char)data;

- (void)appendData:(NSData *)otherData;

- (void)appendBytes:(const void *)bytes length:(NSUInteger)length;

Parameter

Name	Description	Object type
data		unsigned char
otherData	Data (Text and Command).	NSData
bytes		const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSASCIIStringEncoding];

    unsigned char bytes[] = {0x48, 0x65, 0x6c, 0x6c, 0x6f, 0x20, 0x57, 0x6f, 0x72, 0x6c, 0x64, 0x2e};

    NSUInteger length = sizeof(bytes);

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];
    [builder appendByte:'\n'];

    [builder appendBytes:bytes length:length];
    [builder appendByte:'\n'];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.6 appendRawData Method

Raw data (Text and Command) is added to the commands property.

Declaration

- (void)appendRawByte:(unsigned char)data;

- (void)appendRawData:(NSData *)otherData;

- (void)appendRawBytes:(const void *)bytes length:(NSUInteger)length;

Parameter

Name	Description	Object type
data		unsigned char
otherData	Raw data (Text and Command).	NSData
bytes		const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation image:(UIImage *)image {
    NSString *urlString = [SMCSAllReceipts uploadBitmap:image completion:nil];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendBitmap:image diffusion:NO];

    NSData *data = [SMCSAllReceipts generateAllReceipts:urlString emulation:emulation info:YES qrCode:YES];

    [builder appendRawData:data];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to AllReceiptsFunctions.m and each Command Specifications.

10.7 appendFontStyle Method

Select command of the font style is generated and added to the commands property.

Declaration
- (void)appendFontStyle:(SCBFontStyleType)type;

Parameter

Name	Description	Object type
type	Font style. • SCBFontStyleTypeA … Font-A (12 x 24 dots) / Specify 7 x 9 font (half dots) • SCBFontStyleTypeB … Font-B (9 x 24 dots) / Specify 5 x 9 font (2P-1)	SCBFontStyleType

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendFontStyle:SCBFontStyleTypeA];
    [builder appendData:otherData];
    [builder appendFontStyle:SCBFontStyleTypeB];
    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.8 appendCodePage Method

Select command of the code page is generated and added to the commands property.

Declaration

- (void)appendCodePage:(SCBCodePageType)type;

Parameter

Name	Description	Object type
type	Code Page. • SCBCodePageTypeCP437 … CodePage437 (USA, Std. Europe). • SCBCodePageTypeCP737 … Codepage 737 (Greek). • SCBCodePageTypeCP772 … Codepage 772 (Lithuanian). • SCBCodePageTypeCP774 … Codepage 774 (Lithuanian). • SCBCodePageTypeCP851 … Codepage 851 (Greek). • SCBCodePageTypeCP852 … Codepage 852 (Latin-2). • SCBCodePageTypeCP855 … Codepage 855 (Cyrillic Bulgarian). • SCBCodePageTypeCP857 … Codepage 857 (Turkey). • SCBCodePageTypeCP858 … Codepage 858 (Multilingual). • SCBCodePageTypeCP860 … Codepage 860 (Portuguese). • SCBCodePageTypeCP861 … Codepage 861 (Icelandic). • SCBCodePageTypeCP862 … Codepage 862 (Israel (Hebrew)). • SCBCodePageTypeCP863 … Codepage 863 (Canadian French). • SCBCodePageTypeCP864 … Codepage 864 (Arabic). • SCBCodePageTypeCP865 … Codepage 865 (Nordic). • SCBCodePageTypeCP866 … Codepage 866 (Cyrillic Russian). • SCBCodePageTypeCP869 … Codepage 869 (Greek). • SCBCodePageTypeCP874 … Codepage 874 (Thai). • SCBCodePageTypeCP928 … Codepage 928 (Greek). • SCBCodePageTypeCP932 … Katakana. • SCBCodePageTypeCP998 … Normal. • SCBCodePageTypeCP999 … Codepage 1252 (Windows Latin-1). • SCBCodePageTypeCP1001 … Codepage 1001 (Arabic). • SCBCodePageTypeCP1250 … Codepage 1250 (Windows Latin-2). • SCBCodePageTypeCP1251 … Codepage 1251 (Windows Cyrillic). • SCBCodePageTypeCP1252 … Codepage 1252 (Windows Latin-1). • SCBCodePageTypeCP2001 … Codepage 2001 (Lithuanian-KBL). • SCBCodePageTypeCP3001 … Codepage 3001 (Estonian-1). • SCBCodePageTypeCP3002 … Codepage 3002 (Estonian-2). • SCBCodePageTypeCP3011 … Codepage 3011 (Latvian-1). • SCBCodePageTypeCP3012 … Codepage 3012 (Latvian-2). • SCBCodePageTypeCP3021 … Codepage 3021 (Bulgarian). • SCBCodePageTypeCP3041 … Codepage 3041 (Maltese). • SCBCodePageTypeCP3840 … Codepage 3840 (IBM-Russian). • SCBCodePageTypeCP3841 … Codepage 3841 (Gost).	SCBCodePageType

Name	Description	Object type
	• SCBCodePageTypeCP3843 … Codepage 3843 (Polish).	
	• SCBCodePageTypeCP3844 … Codepage 3844 (CS2).	
	• SCBCodePageTypeCP3845 … Codepage 3845 (Hungarian).	
	• SCBCodePageTypeCP3846 … Codepage 3846 (Turkish).	
	• SCBCodePageTypeCP3847 … Codepage 3847 (Brazil-ABNT).	
	• SCBCodePageTypeCP3848 … Codepage 3848 (Brazil-ABICOMP).	
	• SCBCodePageTypeUTF8 … UTF-8.	
	• SCBCodePageTypeBlank … User Setting Blank Code Page.	

Return value

Description	Object type
-	-

Example

```objc
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    unsigned char bytes8[] = {0x80, 0x81, 0x82, 0x83, 0x84, 0x85, 0x86, 0x87, 0x88, 0x89, 0x8a, 0x8b, 0x8c, 0x8d, 0x8e, 0x8f, 0x0a};
    unsigned char bytes9[] = {0x90, 0x91, 0x92, 0x93, 0x94, 0x95, 0x96, 0x97, 0x98, 0x99, 0x9a, 0x9b, 0x9c, 0x9d, 0x9e, 0x9f, 0x0a};
    unsigned char bytesA[] = {0xa0, 0xa1, 0xa2, 0xa3, 0xa4, 0xa5, 0xa6, 0xa7, 0xa8, 0xa9, 0xaa, 0xab, 0xac, 0xad, 0xae, 0xaf, 0x0a};
    unsigned char bytesB[] = {0xb0, 0xb1, 0xb2, 0xb3, 0xb4, 0xb5, 0xb6, 0xb7, 0xb8, 0xb9, 0xba, 0xbb, 0xbc, 0xbd, 0xbe, 0xbf, 0x0a};
    unsigned char bytesC[] = {0xc0, 0xc1, 0xc2, 0xc3, 0xc4, 0xc5, 0xc6, 0xc7, 0xc8, 0xc9, 0xca, 0xcb, 0xcc, 0xcd, 0xce, 0xcf, 0x0a};
    unsigned char bytesD[] = {0xd0, 0xd1, 0xd2, 0xd3, 0xd4, 0xd5, 0xd6, 0xd7, 0xd8, 0xd9, 0xda, 0xdb, 0xdc, 0xdd, 0xde, 0xdf, 0x0a};
    unsigned char bytesE[] = {0xe0, 0xe1, 0xe2, 0xe3, 0xe4, 0xe5, 0xe6, 0xe7, 0xe8, 0xe9, 0xea, 0xeb, 0xec, 0xed, 0xee, 0xef, 0x0a};
    unsigned char bytesF[] = {0xf0, 0xf1, 0xf2, 0xf3, 0xf4, 0xf5, 0xf6, 0xf7, 0xf8, 0xf9, 0xfa, 0xfb, 0xfc, 0xfd, 0xfe, 0xff, 0x0a};

    NSUInteger length = sizeof(bytes8);

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendCodePage:SCBCodePageTypeCP932];

    [builder appendBytes:bytes8 length:length];
    [builder appendBytes:bytes9 length:length];
    [builder appendBytes:bytesA length:length];
    [builder appendBytes:bytesB length:length];
    [builder appendBytes:bytesC length:length];
    [builder appendBytes:bytesD length:length];
    [builder appendBytes:bytesE length:length];
    [builder appendBytes:bytesF length:length];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.9 appendInternational Method

Select command of the international character mode is generated and added to the commands property.

Declaration

- (void)appendInternational:(SCBInternationalType)type;

Parameter

Name	Description	Object type
type	International character. • SCBInternationalTypeUSA … USA. • SCBInternationalTypeFrance … France. • SCBInternationalTypeGermany … Germany. • SCBInternationalTypeUK … UK. • SCBInternationalTypeDenmark … Denmark. • SCBInternationalTypeSweden … Sweden. • SCBInternationalTypeItaly … Italy. • SCBInternationalTypeSpain … Spain. • SCBInternationalTypeJapan … Japan. • SCBInternationalTypeNorway … Norway. • SCBInternationalTypeDenmark2 … Denmark ‖. • SCBInternationalTypeSpain2 … Spain ‖. • SCBInternationalTypeLatinAmerica … Latin America. • SCBInternationalTypeKorea … Korea. • SCBInternationalTypeIreland … Ireland. • SCBInternationalTypeLegal … Legal.	SCBInternationalType

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    unsigned char bytes[] = {0x23, 0x24, 0x40, 0x58, 0x5a, 0x5b, 0x5c, 0x5d, 0x5e, 0x60, 0x7b, 0x7c, 0x7d, 0x7e, 0x0a};

    NSUInteger length = sizeof(bytes);

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendInternational:SCBInternationalTypeUSA];
    [builder appendBytes:bytes length:length];

    [builder appendInternational:SCBInternationalTypeJapan];
    [builder appendBytes:bytes length:length];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.10 appendLineFeed Method

Line feed command is generated and added to the commands property.

Declaration

- (void)appendLineFeed;

- (void)appendDataWithLineFeed:(NSData *)otherData;

- (void)appendBytesWithLineFeed:(const void *)bytes length:(NSUInteger)length;

- (void)appendLineFeed:(NSInteger)line;

- (void)appendDataWithLineFeed:(NSData *)otherData line:(NSInteger)line;

- (void)appendBytesWithLineFeed:(const void *)bytes length:(NSUInteger)length line:(NSInteger)line;

Parameter

Name	Description	Object type
unit	Paper feed units. (Units : Lines)	NSInteger
otherData		NSData
bytes	Data (Text and Command).	const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];
    [builder appendLineFeed];

    [builder appendDataWithLineFeed:otherData];

    [builder appendData:otherData];
    [builder appendLineFeed:2];

    [builder appendDataWithLineFeed:otherData line:2];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.11 appendUnitFeed Method

Unit feed command is generated and added to the commands property.

Declaration

- (void)appendUnitFeed:(NSInteger)unit;

- (void)appendDataWithUnitFeed:(NSData *)otherData unit:(NSInteger)unit;

- (void)appendBytesWithUnitFeed:(const void *)bytes length:(NSUInteger)length unit:(NSInteger)unit;

Parameter

Name	Description	Object type
unit	Paper feed units. (Units : Dots)	NSInteger
otherData		NSData
bytes	Data (Text and Command).	const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World."   dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];
    [builder appendUnitFeed:64];

    [builder appendDataWithUnitFeed:otherData unit:64];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.12 appendCharacterSpace Method

Set command of the character space is generated and added to the commands property.

Declaration
- (void)appendCharacterSpace:(NSInteger)space;

Parameter

Name	Description	Object type
space	Character spaces. (Units : Dots)	NSInteger

Return value

Description	Object type
-	-

1. Notes
In Japanese, Simplified Chinese, Traditional Chinese, Korean (DBCS), the character space is to be twice.

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendCharacterSpace:0];
    [builder appendData:otherData];
    [builder appendCharacterSpace:4];
    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.13 appendLineSpace Method

Set command of the line space is generated and added to the commands property.

Declaration

- (void)appendLineSpace:(NSInteger)lineSpace;

Parameter

Name	Description	Object type
lineSpace	Line spaces. (Units : Dots)	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendLineSpace:32];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendLineSpace:24];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.14 appendTopMargin Method

Set command of the top margin is generated and added to the commands property.

Declaration

- (void)appendTopMargin:(NSInteger)margin;

Parameter

Name	Description	Object type
margin	Top margin. (Units : Millimeter)	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createTopMarginData:(StarIoExtEmulation)emulation {
    NSData *data = [@"Hello, World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:StarIoExtEmulationStarPRNT];

    [builder beginDocument];

    [builder appendTopMargin:2];
    [builder appendData:[@"*Top margin:2mm*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendData:data];
    [builder appendData:data];
    [builder appendData:data];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder appendTopMargin:6];
    [builder appendData:[@"*Top margin:6mm*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendData:data];
    [builder appendData:data];
    [builder appendData:data];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder appendTopMargin:11];
    [builder appendData:[@"*Top margin:11mm*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendData:data];
    [builder appendData:data];
    [builder appendData:data];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return builder.commands;
}
```

Refer to ApiFunctions.m.

10.15 appendEmphasis Method

Select command of the emphasis mode is generated and added to the commands property.

Declaration

- (void)appendEmphasis:(BOOL)emphasis;

- (void)appendDataWithEmphasis:(NSData *)otherData;

- (void)appendBytesWithEmphasis:(const void *)bytes length:(NSUInteger)length;

Parameter

Name	Description	Object type
emphasis	Emphasis. • YES … Valid • NO … Invalid	BOOL
otherData		NSData
bytes	Data (Text and Command).	const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherDataHalf0 = [@"Hello " dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherDataHalf1 = [@"World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];
    [builder appendEmphasis:YES];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendEmphasis:NO];
    [builder appendData:otherData];

    [builder appendDataWithEmphasis:otherData];
    [builder appendData:otherData];

    [builder appendDataWithEmphasis:otherDataHalf0];
    [builder appendData:otherDataHalf1];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.16 appendInvert Method

Select command of the invert mode is generated and added to the commands property.

Declaration

- (void)appendInvert:(BOOL)invert;

- (void)appendDataWithInvert:(NSData *)otherData;

- (void)appendBytesWithInvert:(const void *)bytes length:(NSUInteger)length;

Parameter

Name	Description	Object type
invert	Invert. • YES … Valid • NO … Invalid	BOOL
otherData		NSData
bytes	Data (Text and Command).	const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherDataHalf0 = [@"Hello " dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherDataHalf1 = [@"World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];
    [builder appendInvert:YES];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendInvert:NO];
    [builder appendData:otherData];

    [builder appendDataWithInvert:otherData];
    [builder appendData:otherData];

    [builder appendDataWithInvert:otherDataHalf0];
    [builder appendData:otherDataHalf1];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.17 appendMultiple Method

Select command of the multiple mode is generated and added to the commands property.

Declaration

- (void)appendMultiple:(NSInteger)width height:(NSInteger)height;

- (void)appendDataWithMultiple:(NSData *)otherData width:(NSInteger)width
height:(NSInteger)height;

- (void)appendBytesWithMultiple:(const void *)bytes length:(NSUInteger)length
width:(NSInteger)width height:(NSInteger)height;

- (void)appendMultipleHeight:(NSInteger)height;

- (void)appendDataWithMultipleHeight:(NSData *)otherData height:(NSInteger)height;

- (void)appendBytesWithMultipleHeight:(const void *)bytes length:(NSUInteger)length
height:(NSInteger)height;

- (void)appendMultipleWidth:(NSInteger)width;

- (void)appendDataWithMultipleWidth:(NSData *)otherData width:(NSInteger)width;

- (void)appendBytesWithMultipleWidth:(const void *)bytes length:(NSUInteger)length
width:(NSInteger)width;

Parameter

Name	Description	Object type
width	Expanded width.	NSInteger
height	Expanded height.	NSInteger
otherData		NSData
bytes	Data (Text and Command).	const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherDataHalf0 = [@"Hello " dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherDataHalf1 = [@"World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendMultiple:2 height:2];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendMultiple:1 height:1];
    [builder appendData:otherData];

    [builder appendDataWithMultiple:otherData width:2 height:2];
    [builder appendData:          otherData];

    [builder appendDataWithMultiple:otherDataHalf0 width:2 height:2];
    [builder appendData:          otherDataHalf1];

    [builder appendData:otherDataHalf0];
    [builder appendDataWithMultiple:otherDataHalf1 width:2 height:2];

    [builder appendMultipleHeight:2];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendMultipleHeight:1];
    [builder appendData:otherData];

    [builder appendDataWithMultipleHeight:otherDataHalf0 height:2];
    [builder appendData:otherDataHalf1];

    [builder appendData:otherDataHalf0];
    [builder appendDataWithMultipleHeight:otherDataHalf1 height:2];

    [builder appendMultipleWidth:2];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendMultipleWidth:1];
    [builder appendData:otherData];

    [builder appendDataWithMultipleWidth:otherDataHalf0 width:2];
    [builder appendData:otherDataHalf1];

    [builder appendData:otherDataHalf0];
    [builder appendDataWithMultipleWidth:otherDataHalf1 width:2];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.18 appendUnderLine Method

Select command of the under line mode is generated and added to the commands property.

Declaration

- (void)appendUnderLine:(BOOL)underLine;

- (void)appendDataWithUnderLine:(NSData *)otherData;

- (void)appendBytesWithUnderLine:(const void *)bytes length:(NSUInteger)length;

Parameter

Name	Description	Object type
underLine	Under line. • YES … Valid • NO … Invalid	BOOL
otherData		NSData
bytes	Data (Text and Command).	const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherDataHalf0 = [@"Hello " dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherDataHalf1 = [@"World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendUnderLine:YES];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendUnderLine:NO];
    [builder appendData:otherData];

    [builder appendDataWithUnderLine:otherData];
    [builder appendData:otherData];

    [builder appendDataWithUnderLine:otherDataHalf0];
    [builder appendData:otherDataHalf1];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.19 appendLogo Method

Print command of the logo is generated and added to the commands property.

Declaration
- (void)appendLogo:(SCBLogoSize)size number:(NSInteger)number;

Parameter

Name	Description	Object type
size	Logo size. • SCBLogoSizeNormal … Normal. • SCBLogoSizeDoubleWidth … Double Width. • SCBLogoSizeDoubleHeight … Double Height. • SCBLogoSizeDoubleWidthDoubleHeight … Double Width / Height.	SCBLogoSize
number	Logo number.	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendLogo:SCBLogoSizeNormal number:1];

    [builder appendLogo:SCBLogoSizeDoubleWidth number:1];

    [builder appendLogo:SCBLogoSizeDoubleHeight number:1];

    [builder appendLogo:SCBLogoSizeDoubleWidthDoubleHeight number:1];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.20 appendAbsolutePosition Method

Absolute position command is generated and added to the commands property.

Declaration

- (void)appendAbsolutePosition:(NSInteger)position;

- (void)appendDataWithAbsolutePosition:(NSData *)otherData position:(NSInteger)position;

- (void)appendBytesWithAbsolutePosition:(const void *)bytes length:(NSUInteger)length position:(NSInteger)position;

Parameter

Name	Description	Object type
position	Absolute position. (Units : Dots)	NSInteger
otherData		NSData
bytes	Data (Text and Command).	const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendAbsolutePosition:40];
    [builder appendData:otherData];
    [builder appendData:otherData];

    [builder appendDataWithAbsolutePosition:otherData position:40];
    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.21 appendAlignment Method

Alignment command is generated and added to the commands property.

Declaration
- (void)appendAlignment:(SCBAlignmentPosition)position;

- (void)appendDataWithAlignment:(NSData *)otherData position:(SCBAlignmentPosition)position;

- (void)appendBytesWithAlignment:(const void *)bytes length:(NSUInteger)length position:(SCBAlignmentPosition)position;

Parameter

Name	Description	Object type
position	Alignment position. • SCBAlignmentPositionLeft … Left alignment. • SCBAlignmentPositionCenter … Center alignment. • SCBAlignmentPositionRight … Right alignment.	SCBAlignmentPosition
otherData		NSData
bytes	Data (Text and Command).	const void *
length		NSUInteger

Return value

Description	Object type
-	-

1. Notes
"appendAlignment" cannot be used with "appendBitmap".
Therefore, please use "appendBitmapWithAlignment" if you would like to adjust bitmap position.

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendAlignment:SCBAlignmentPositionRight];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendAlignment:SCBAlignmentPositionLeft];
    [builder appendData:otherData];
    [builder appendData:otherData];

    [builder appendDataWithAlignment:otherData position:SCBAlignmentPositionRight];
    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.22 appendHorizontalTabPosition Method

Horizontal tab set/clear command is generated and added to the commands property.

Declaration

- (void)appendHorizontalTabPosition:(NSArray<NSNumber *> *)positions;

Parameter

Name	Description	Object type
position	Array of horizontal tab positions (Units: ANK character pitch) Specifying nil or empty array deletes all currently set horizontal tab positions.	NSArray<NSNumber *> *

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createHorizontalTabPositionData:(StarIoExtEmulation)emulation {
    NSData *otherData1 = [@"QTY\tITEM\tTOTAL\n" dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherData2 = [@"1\tApple\t1.50\n"   dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherData3 = [@"2\tOrange\t2.00\n"  dataUsingEncoding:NSASCIIStringEncoding];
    NSData *otherData4 = [@"5\tBanana\t3.00\n"  dataUsingEncoding:NSASCIIStringEncoding];

    NSArray<NSNumber *> *positions = @[@5, @27];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendHorizontalTabPosition:positions];

    [builder appendData:[@"*Tab Position:5, 27*\n" dataUsingEncoding:NSASCIIStringEncoding]];

    [builder appendData:otherData1];
    [builder appendData:otherData2];
    [builder appendData:otherData3];
    [builder appendData:otherData4];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.23 appendCutPaper Method

Paper cut command is generated and added to the commands property.

Declaration
- (void)appendCutPaper:(SCBCutPaperAction)action;

Parameter

Name	Description	Object type
action	Paper cut action. • SCBCutPaperActionFullCut … Full Cut. • SCBCutPaperActionPartialCut … Partial Cut. • SCBCutPaperActionFullCutWithFeed … Full Cut with Feed. • SCBCutPaperActionPartialCutWithFeed … Partial Cut with Feed.	SCBCutPaperAction

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendData:otherData];
    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.24 appendPeripheral Method

Cash drawer command is generated and added to the commands property.

Declaration

- (void)appendPeripheral:(SCBPeripheralChannel)channel;
- (void)appendPeripheral:(SCBPeripheralChannel)channel time:(NSInteger)time;

Parameter

Name	Description	Object type
channel	Cash drawer channel. • SCBPeripheralChannelNo1 … Channel1. • SCBPeripheralChannelNo2 … Channel2.	SCBPeripheralChannel
time	Drive time. (Units : mSec) * Peripheral channel1 only.	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendPeripheral:SCBPeripheralChannelNo1];
    [builder appendPeripheral:SCBPeripheralChannelNo2];
    [builder appendPeripheral:SCBPeripheralChannelNo1 time:2000];
    [builder appendPeripheral:SCBPeripheralChannelNo2 time:2000];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.25 appendSound Method

Sound/Buzzer command is generated and added to the commands property.

Declaration

- (void)appendSound:(SCBSoundChannel)channel;
- (void)appendSound:(SCBSoundChannel)channel repeat:(NSInteger)repeat;
- (void)appendSound:(SCBSoundChannel)channel repeat:(NSInteger)repeat driveTime:(NSInteger)
driveTime delayTime:(NSInteger)delayTime

Parameter

Name	Description	Object type
channel	Sound/Buzzer channel. • SCBSoundChannelNo1 … Channel1. • SCBSoundChannelNo2 … Channel2.	SCBSoundChannel
repeat	Repeat count.	NSInteger
driveTime	Drive time. (Units : mSec)	NSInteger
delayTime	Delay time. (Units : mSec)	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendSound:SCBSoundChannelNo1];
    [builder appendSound:SCBSoundChannelNo2];
    [builder appendSound:SCBSoundChannelNo1 repeat:3];
    [builder appendSound:SCBSoundChannelNo2 repeat:3];
    [builder appendSound:SCBSoundChannelNo1 repeat:1 driveTime:1000 delayTime:1000];
    [builder appendSound:SCBSoundChannelNo2 repeat:1 driveTime:1000 delayTime:1000];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.26 appendBarcodeData Method

Print command of the barcode is generated and added to the commands property.

Declaration

- (void)appendBarcodeData:(NSData *)otherData symbology:(SCBBarcodeSymbology)symbology width:(SCBBarcodeWidth)width height:(NSInteger)height hri:(BOOL)hri;

- (void)appendBarcodeBytes:(const void *)bytes length:(NSUInteger)length symbology:(SCBBarcodeSymbology)symbology width:(SCBBarcodeWidth)width height:(NSInteger)height hri:(BOOL)hri;

Parameter

Name	Description	Object type
otherData		NSData
bytes	Barcode data.	const void *
length		NSUInteger
symbology	Barcode symbology. • SCBBarcodeSymbologyUPCE … UPC-E. • SCBBarcodeSymbologyUPCA … UPC-A. • SCBBarcodeSymbologyJAN8 … JAN/EAN8. • SCBBarcodeSymbologyJAN13 … JAN/EAN13. • SCBBarcodeSymbologyCode39 … Code39. • SCBBarcodeSymbologyITF … ITF. • SCBBarcodeSymbologyCode128 … Code128. • SCBBarcodeSymbologyCode93 … Code93. • ~~SCBBarcodeSymbologyCodabar … Codabar.~~ • SCBBarcodeSymbologyNW7 … NW7.	SCBBarcodeSymbology
width	Barcode width. • SCBBarcodeWidthMode1 … Mode1. • SCBBarcodeWidthMode2 … Mode2. • SCBBarcodeWidthMode3 … Mode3. • SCBBarcodeWidthMode4 … Mode4. • SCBBarcodeWidthMode5 … Mode5. • SCBBarcodeWidthMode6 … Mode6. • SCBBarcodeWidthMode7 … Mode7. • SCBBarcodeWidthMode8 … Mode8. • SCBBarcodeWidthMode9 … Mode9. • SCBBarcodeWidthExtMode1 … ExtMode1. • SCBBarcodeWidthExtMode2 … ExtMode2. • SCBBarcodeWidthExtMode3 … ExtMode3. • SCBBarcodeWidthExtMode4 … ExtMode4. • SCBBarcodeWidthExtMode5 … ExtMode5. • SCBBarcodeWidthExtMode6 … ExtMode6. • SCBBarcodeWidthExtMode7 … ExtMode7. • SCBBarcodeWidthExtMode8 … ExtMode8. • SCBBarcodeWidthExtMode9 … ExtMode9.	SCBBarcodeWidth

Name	Description	Object type
height	Barcode height. (Units : Dots)	NSInteger
hri	Under-bar characters. • YES … Valid • NO … Invalid	BOOL

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherDataCode128 = [@"{B0123456789" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendBarcodeData:otherDataCode128 symbology:SCBBarcodeSymbologyCode128
width:SCBBarcodeWidthMode1 height:40 hri:YES];
    [builder appendUnitFeed:32];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.27 appendBarcodeDataWithAbsolutePosition Method

Print command of the absolute position barcode is generated and added to the commands property.

Declaration

- (void)appendBarcodeDataWithAbsolutePosition:(NSData *)otherData symbology:(SCBBarcodeSymbology)symbology width:(SCBBarcodeWidth)width height:(NSInteger)height hri:(BOOL)hri position:(NSInteger)position;

- (void)appendBarcodeBytesWithAbsolutePosition:(const void *)bytes length:(NSUInteger)length symbology:(SCBBarcodeSymbology)symbology width:(SCBBarcodeWidth)width height:(NSInteger)height hri:(BOOL)hri position:(NSInteger)position;

Parameter

Name	Description	Object type
position	Absolute position. (Units : Dots)	NSInteger
otherData		NSData
bytes	Barcode data.	const void *
length		NSUInteger
symbology	Barcode symbology. • SCBBarcodeSymbologyUPCE … UPC-E. • SCBBarcodeSymbologyUPCA … UPC-A. • SCBBarcodeSymbologyJAN8 … JAN/EAN8. • SCBBarcodeSymbologyJAN13 … JAN/EAN13. • SCBBarcodeSymbologyCode39 … Code39. • SCBBarcodeSymbologyITF … ITF. • SCBBarcodeSymbologyCode128 … Code128. • SCBBarcodeSymbologyCode93 … Code93. • ~~SCBBarcodeSymbologyCodabar … Codabar.~~ • SCBBarcodeSymbologyNW7 … NW7.	SCBBarcodeSymbology

Name	Description	Object type
width	Barcode width. • SCBBarcodeWidthMode1 … Mode1. • SCBBarcodeWidthMode2 … Mode2. • SCBBarcodeWidthMode3 … Mode3. • SCBBarcodeWidthMode4 … Mode4. • SCBBarcodeWidthMode5 … Mode5. • SCBBarcodeWidthMode6 … Mode6. • SCBBarcodeWidthMode7 … Mode7. • SCBBarcodeWidthMode8 … Mode8. • SCBBarcodeWidthMode9 … Mode9. • SCBBarcodeWidthExtMode1 … ExtMode1. • SCBBarcodeWidthExtMode2 … ExtMode2. • SCBBarcodeWidthExtMode3 … ExtMode3. • SCBBarcodeWidthExtMode4 … ExtMode4. • SCBBarcodeWidthExtMode5 … ExtMode5. • SCBBarcodeWidthExtMode6 … ExtMode6. • SCBBarcodeWidthExtMode7 … ExtMode7. • SCBBarcodeWidthExtMode8 … ExtMode8. • SCBBarcodeWidthExtMode9 … ExtMode9.	SCBBarcodeWidth
height	Barcode height. (Units : Dots)	NSInteger
hri	Under-bar characters. • YES … Valid • NO … Invalid	BOOL

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherDataCode128 = [@"{B0123456789" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendBarcodeDataWithAbsolutePosition:otherDataCode128
symbology:SCBBarcodeSymbologyCode128 width:SCBBarcodeWidthMode1 height:40 hri:YES position:40];
    [builder appendUnitFeed:32];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.28 appendBarcodeDataWithAlignment Method

Print command of the alignment barcode is generated and added to the commands property.

Declaration

- (void)appendBarcodeDataWithAlignment:(NSData *)otherData
symbology:(SCBBarcodeSymbology)symbology width:(SCBBarcodeWidth)width
height:(NSInteger)height hri:(BOOL)hri position:(SCBAlignmentPosition)position;

- (void)appendBarcodeBytesWithAlignment:(const void *)bytes length:(NSUInteger)length
symbology:(SCBBarcodeSymbology)symbology width:(SCBBarcodeWidth)width
height:(NSInteger)height hri:(BOOL)hri position:(SCBAlignmentPosition)position;

Parameter

Name	Description	Object type
position	Alignment position. • SCBAlignmentPositionLeft … Left alignment. • SCBAlignmentPositionCenter … Center alignment. • SCBAlignmentPositionRight … Right alignment.	SCBAlignmentPosition
otherData	Barcode data.	NSData
bytes		const void *
length		NSUInteger
symbology	Barcode symbology. • SCBBarcodeSymbologyUPCE … UPC-E. • SCBBarcodeSymbologyUPCA … UPC-A. • SCBBarcodeSymbologyJAN8 … JAN/EAN8. • SCBBarcodeSymbologyJAN13 … JAN/EAN13. • SCBBarcodeSymbologyCode39 … Code39. • SCBBarcodeSymbologyITF … ITF. • SCBBarcodeSymbologyCode128 … Code128. • SCBBarcodeSymbologyCode93 … Code93. • ~~SCBBarcodeSymbologyCodabar … Codabar.~~ • SCBBarcodeSymbologyNW7 … NW7.	SCBBarcodeSymbology

Name	Description	Object type
width	Barcode width. • SCBBarcodeWidthMode1 … Mode1. • SCBBarcodeWidthMode2 … Mode2. • SCBBarcodeWidthMode3 … Mode3. • SCBBarcodeWidthMode4 … Mode4. • SCBBarcodeWidthMode5 … Mode5. • SCBBarcodeWidthMode6 … Mode6. • SCBBarcodeWidthMode7 … Mode7. • SCBBarcodeWidthMode8 … Mode8. • SCBBarcodeWidthMode9 … Mode9. • SCBBarcodeWidthExtMode1 … ExtMode1. • SCBBarcodeWidthExtMode2 … ExtMode2. • SCBBarcodeWidthExtMode3 … ExtMode3. • SCBBarcodeWidthExtMode4 … ExtMode4. • SCBBarcodeWidthExtMode5 … ExtMode5. • SCBBarcodeWidthExtMode6 … ExtMode6. • SCBBarcodeWidthExtMode7 … ExtMode7. • SCBBarcodeWidthExtMode8 … ExtMode8. • SCBBarcodeWidthExtMode9 … ExtMode9.	SCBBarcodeWidth
height	Barcode height. (Units : Dots)	NSInteger
hri	Under-bar characters. • YES … Valid • NO … Invalid	BOOL

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherDataCode128 = [@"{B0123456789" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendBarcodeDataWithAlignment:otherDataCode128 symbology:SCBBarcodeSymbologyCode128
width:SCBBarcodeWidthMode1 height:40 hri:YES position:SCBAlignmentPositionCenter];
    [builder appendUnitFeed:32];

    [builder appendBarcodeDataWithAlignment:otherDataCode128 symbology:SCBBarcodeSymbologyCode128
width:SCBBarcodeWidthMode1 height:40 hri:YES position:SCBAlignmentPositionRight];
    [builder appendUnitFeed:32];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.29 appendPdf417Data Method

Print command of the PDF417 is generated and added to the commands property.

Declaration

- (void)appendPdf417Data:(NSData *)otherData line:(NSInteger)line column:(NSInteger)column level:(SCBPdf417Level)level module:(NSInteger)module aspect:(NSInteger)aspect;

- (void)appendPdf417Bytes:(const void *)bytes length:(NSUInteger)length line:(NSInteger)line column:(NSInteger)column level:(SCBPdf417Level)level module:(NSInteger)module aspect:(NSInteger)aspect;

Parameter

Name	Description	Object type
otherData		NSData
bytes	PDF417 data.	const void *
length		NSUInteger
line	Number of lines.	NSInteger
column	Number of columns.	NSInteger
level	PDF417 ECC (security level). • SCBPdf417LevelECC0 … Level0. • SCBPdf417LevelECC1 … Level1. • SCBPdf417LevelECC2 … Level2. • SCBPdf417LevelECC3 … Level3. • SCBPdf417LevelECC4 … Level4. • SCBPdf417LevelECC5 … Level5. • SCBPdf417LevelECC6 … Level6. • SCBPdf417LevelECC7 … Level7. • SCBPdf417LevelECC8 … Level8.	SCBPdf417Level
module	Module X direction size.	NSInteger
aspect	Module aspect ratio.	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendPdf417Data:otherData line:0 column:1 level:SCBPdf417LevelECC0 module:2 aspect:2];
    [builder appendUnitFeed:32];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.30 appendPdf417DataWithAbsolutePosition Method

Print command of the absolute position PDF417 is generated and added to the commands property.

Declaration

- (void)appendPdf417DataWithAbsolutePosition:(NSData *)otherData line:(NSInteger)line column:(NSInteger)column level:(SCBPdf417Level)level module:(NSInteger)module aspect:(NSInteger)aspect position:(NSInteger)position;

- (void)appendPdf417BytesWithAbsolutePosition:(const void *)bytes length:(NSUInteger)length line:(NSInteger)line column:(NSInteger)column level:(SCBPdf417Level)level module:(NSInteger)module aspect:(NSInteger)aspect position:(NSInteger)position;

Parameter

Name	Description	Object type
position	Absolute position. (Units : Dots)	NSInteger
otherData		NSData
bytes	PDF417 data.	const void *
length		NSUInteger
line	Number of lines.	NSInteger
column	Number of columns.	NSInteger
level	PDF417 ECC (security level). • SCBPdf417LevelECC0 … Level0. • SCBPdf417LevelECC1 … Level1. • SCBPdf417LevelECC2 … Level2. • SCBPdf417LevelECC3 … Level3. • SCBPdf417LevelECC4 … Level4. • SCBPdf417LevelECC5 … Level5. • SCBPdf417LevelECC6 … Level6. • SCBPdf417LevelECC7 … Level7. • SCBPdf417LevelECC8 … Level8.	SCBPdf417Level
module	Module X direction size.	NSInteger
aspect	Module aspect ratio.	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendPdf417DataWithAbsolutePosition:otherData line:0 column:1 level:SCBPdf417LevelECC0
module:2 aspect:2 position:40];
    [builder appendUnitFeed:32];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.31 appendPdf417DataWithAlignment Method

Print command of the alignment PDF417 is generated and added to the commands property.

Declaration

- (void)appendPdf417DataWithAlignment:(NSData *)otherData line:(NSInteger)line
column:(NSInteger)column level:(SCBPdf417Level)level module:(NSInteger)module
aspect:(NSInteger)aspect position:(SCBAlignmentPosition)position;

- (void)appendPdf417BytesWithAlignment:(const void *)bytes length:(NSUInteger)length
line:(NSInteger)line column:(NSInteger)column level:(SCBPdf417Level)level
module:(NSInteger)module aspect:(NSInteger)aspect position:(SCBAlignmentPosition)position;

Parameter

Name	Description	Object type
position	Alignment position. • SCBAlignmentPositionLeft … Left alignment. • SCBAlignmentPositionCenter … Center alignment. • SCBAlignmentPositionRight … Right alignment.	SCBAlignmentPosition
otherData	PDF417 data.	NSData
bytes		const void *
length		NSUInteger
line	Number of lines.	NSInteger
column	Number of columns.	NSInteger
level	PDF417 ECC (security level). • SCBPdf417LevelECC0 … Level0. • SCBPdf417LevelECC1 … Level1. • SCBPdf417LevelECC2 … Level2. • SCBPdf417LevelECC3 … Level3. • SCBPdf417LevelECC4 … Level4. • SCBPdf417LevelECC5 … Level5. • SCBPdf417LevelECC6 … Level6. • SCBPdf417LevelECC7 … Level7. • SCBPdf417LevelECC8 … Level8.	SCBPdf417Level
module	Module X direction size.	NSInteger
aspect	Module aspect ratio.	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendPdf417DataWithAlignment:otherData line:0 column:1 level:SCBPdf417LevelECC0 module:2
aspect:2 position:SCBAlignmentPositionCenter];
    [builder appendUnitFeed:32];

    [builder appendPdf417DataWithAlignment:otherData line:0 column:1 level:SCBPdf417LevelECC0 module:2
aspect:2 position:SCBAlignmentPositionRight];
    [builder appendUnitFeed:32];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.32 appendQrCodeData Method

Print command of the QR code is generated and added to the commands property.

Declaration

- (void)appendQrCodeData:(NSData *)otherData model:(SCBQrCodeModel)model level:(SCBQrCodeLevel)level cell:(NSInteger)cell;

- (void)appendQrCodeBytes:(const void *)bytes length:(NSUInteger)length model:(SCBQrCodeModel)model level:(SCBQrCodeLevel)level cell:(NSInteger)cell;

Parameter

Name	Description	Object type
otherData		NSData
bytes	QR code data.	const void *
length		NSUInteger
model	QR code model. • SCBQrCodeModelNo1 … Model1. • SCBQrCodeModelNo2 … Model2.	SCBQrCodeModel
level	QR code mistake correction level. • SCBQrCodeLevelL … Level L. • SCBQrCodeLevelM … Level M. • SCBQrCodeLevelQ … Level Q. • SCBQrCodeLevelH … Level H.	SCBQrCodeLevel
cell	QR code cell size.	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendQrCodeData:otherData model:SCBQrCodeModelNo2 level:SCBQrCodeLevelL cell:4];
    [builder appendUnitFeed:32];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.33 appendQrCodeDataWithAbsolutePosition Method

Print command of the absolute position QR code is generated and added to the commands property.

Declaration

- (void)appendQrCodeDataWithAbsolutePosition:(NSData *)otherData model:(SCBQrCodeModel)model level:(SCBQrCodeLevel)level cell:(NSInteger)cell position:(NSInteger)position;

- (void)appendQrCodeBytesWithAbsolutePosition:(const void *)bytes length:(NSUInteger)length model:(SCBQrCodeModel)model level:(SCBQrCodeLevel)level cell:(NSInteger)cell position:(NSInteger)position;

Parameter

Name	Description	Object type
position	Absolute position. (Units : Dots)	NSInteger
otherData		NSData
bytes	QR code data.	const void *
length		NSUInteger
model	QR code model. • SCBQrCodeModelNo1 … Model1. • SCBQrCodeModelNo2 … Model2.	SCBQrCodeModel
level	QR code mistake correction level. • SCBQrCodeLevelL … Level L. • SCBQrCodeLevelM … Level M. • SCBQrCodeLevelQ … Level Q. • SCBQrCodeLevelH … Level H.	SCBQrCodeLevel
cell	QR code cell size.	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendQrCodeDataWithAbsolutePosition:otherData model:SCBQrCodeModelNo2
level:SCBQrCodeLevelL cell:4 position:40];
    [builder appendUnitFeed:32];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.34 appendQrCodeDataWithAlignment Method

Print command of the alignment QR code is generated and added to the commands property.

Declaration

- (void)appendQrCodeDataWithAlignment:(NSData *)otherData model:(SCBQrCodeModel)model
level:(SCBQrCodeLevel)level cell:(NSInteger)cell position:(SCBAlignmentPosition)position;

- (void)appendQrCodeBytesWithAlignment:(const void *)bytes length:(NSUInteger)length
model:(SCBQrCodeModel)model level:(SCBQrCodeLevel)level cell:(NSInteger)cell
position:(SCBAlignmentPosition)position;

Parameter

Name	Description	Object type
position	Alignment position. • SCBAlignmentPositionLeft … Left alignment. • SCBAlignmentPositionCenter … Center alignment. • SCBAlignmentPositionRight … Right alignment.	SCBAlignmentPosition
otherData		NSData
bytes	QR code data.	const void *
length		NSUInteger
model	QR code model. • SCBQrCodeModelNo1 … Model1. • SCBQrCodeModelNo2 … Model2.	SCBQrCodeModel
level	QR code mistake correction level. • SCBQrCodeLevelL … Level L. • SCBQrCodeLevelM … Level M. • SCBQrCodeLevelQ … Level Q. • SCBQrCodeLevelH … Level H.	SCBQrCodeLevel
cell	QR code cell size.	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendQrCodeDataWithAlignment:otherData model:SCBQrCodeModelNo2 level:SCBQrCodeLevelL
cell:4 position:SCBAlignmentPositionCenter];
    [builder appendUnitFeed:32];

    [builder appendQrCodeDataWithAlignment:otherData model:SCBQrCodeModelNo2 level:SCBQrCodeLevelL
cell:4 position:SCBAlignmentPositionRight];
    [builder appendUnitFeed:32];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m and each Command Specifications.

10.35 appendBitmap Method

Print command of the bitmap is generated and added to the commands property.

Declaration

- (void)appendBitmap:(UIImage *)image diffusion:(BOOL)diffusion width:(NSInteger)width bothScale:(BOOL)bothScale rotation:(SCBBitmapConverterRotation)rotation;

- (void)appendBitmap:(UIImage *)image diffusion:(BOOL)diffusion rotation:(SCBBitmapConverterRotation)rotation;

- (void)appendBitmap:(UIImage *)image diffusion:(BOOL)diffusion width:(NSInteger)width bothScale:(BOOL)bothScale;

- (void)appendBitmap:(UIImage *)image diffusion:(BOOL)diffusion;

Parameter

Name	Description	Object type
image	Source bitmap object.	UIImage
diffusion	Random dither. • YES … Valid • NO … Invalid	BOOL
width	Bitmap width after conversion. (Units : Dots)	NSInteger
bothScale	Height is changed according to the conversion rate of the width property. • YES … Valid. • NO … Invalid.	BOOL
rotation	Rotation. • SCBBitmapConverterRotationNormal … Rotated 0. • SCBBitmapConverterRotationRight90 … Rotated 90. • SCBBitmapConverterRotationLeft90 … Rotated 270. • SCBBitmapConverterRotationRotate180 … Rotated 180.	SCBBitmapConverterRotation

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation width:(NSInteger)width {
    UIImage *starLogoImage = [UIImage imageNamed:@"StarLogoImage"];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:[@"\n*Normal*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmap:starLogoImage diffusion:YES];

    [builder appendData:[@"\n*width:Full, bothScale:YES*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmap:starLogoImage diffusion:YES width:width bothScale:YES];
    [builder appendData:[@"\n*width:Full, bothScale:NO*\n"  dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmap:starLogoImage diffusion:YES width:width bothScale:NO];

    [builder appendData:[@"\n*Rotate180*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmap:starLogoImage diffusion:YES rotation:SCBBitmapConverterRotationRotate180];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.36 appendBitmapWithAbsolutePosition Method

Print command of the absolute position bitmap is generated and added to the commands property.

Declaration

- (void)appendBitmapWithAbsolutePosition:(UIImage *)image diffusion:(BOOL)diffusion width:(NSInteger)width bothScale:(BOOL)bothScale rotation:(SCBBitmapConverterRotation)rotation position:(NSInteger)position;

- (void)appendBitmapWithAbsolutePosition:(UIImage *)image diffusion:(BOOL)diffusion rotation:(SCBBitmapConverterRotation)rotation position:(NSInteger)position;

- (void)appendBitmapWithAbsolutePosition:(UIImage *)image diffusion:(BOOL)diffusion width:(NSInteger)width bothScale:(BOOL)bothScale position:(NSInteger)position;

- (void)appendBitmapWithAbsolutePosition:(UIImage *)image diffusion:(BOOL)diffusion position:(NSInteger)position;

Parameter

Name	Description	Object type
position	Absolute position. (Units : Dots)	NSInteger
image	Source bitmap object.	UIImage
diffusion	Random dither. • YES … Valid • NO … Invalid	BOOL
width	Bitmap width after conversion. (Units : Dots)	NSInteger
bothScale	Height is changed according to the conversion rate of the width property. • YES … Valid. • NO … Invalid.	BOOL
rotation	Rotation. • SCBBitmapConverterRotationNormal … Rotated 0. • SCBBitmapConverterRotationRight90 … Rotated 90. • SCBBitmapConverterRotationLeft90 … Rotated 270. • SCBBitmapConverterRotationRotate180 … Rotated 180.	SCBBitmapConverterRotation

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation width:(NSInteger)width {
    UIImage *starLogoImage = [UIImage imageNamed:@"StarLogoImage"];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:[@"\n*Normal, AbsolutePosition:40*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmapWithAbsolutePosition:starLogoImage diffusion:YES position:40];

    [builder appendData:[@"\n*width:Full, bothScale:YES, AbsolutePosition:40*\n"
dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmap:starLogoImage diffusion:YES width:width bothScale:YES position:40];
    [builder appendData:[@"\n*width:Full, bothScale:NO, AbsolutePosition:40*\n"
dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmap:starLogoImage diffusion:YES width:width bothScale:NO position:40];

    [builder appendData:[@"\n*Rotate180, AbsolutePosition:40*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmapWithAbsolutePosition:starLogoImage diffusion:YES
rotation:SCBBitmapConverterRotationRotate180 position:40];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.37 appendBitmapWithAlignment Method

Print command of the alignment bitmap is generated and added to the commands property.

Declaration

- (void)appendBitmapWithAlignment:(UIImage *)image diffusion:(BOOL)diffusion
width:(NSInteger)width bothScale:(BOOL)bothScale rotation:(SCBBitmapConverterRotation)rotation
position:(SCBAlignmentPosition)position;

- (void)appendBitmapWithAlignment:(UIImage *)image diffusion:(BOOL)diffusion
rotation:(SCBBitmapConverterRotation)rotation position:(SCBAlignmentPosition)position;

- (void)appendBitmapWithAlignment:(UIImage *)image diffusion:(BOOL)diffusion
width:(NSInteger)width bothScale:(BOOL)bothScale position:(SCBAlignmentPosition)position;

- (void)appendBitmapWithAlignment:(UIImage *)image diffusion:(BOOL)diffusion
position:(SCBAlignmentPosition)position;

Parameter

Name	Description	Object type
position	Alignment position. • SCBAlignmentPositionLeft … Left alignment. • SCBAlignmentPositionCenter … Center alignment. • SCBAlignmentPositionRight … Right alignment.	SCBAlignmentPosition
image	Source bitmap object.	UIImage
diffusion	Random dither. • YES … Valid • NO … Invalid	BOOL
Width	Bitmap width after conversion. (Units : Dots)	NSInteger
bothScale	Height is changed according to the conversion rate of the width property. • YES … Valid. • NO … Invalid.	BOOL
rotation	Rotation. • SCBBitmapConverterRotationNormal … Rotated 0. • SCBBitmapConverterRotationRight90 … Rotated 90. • SCBBitmapConverterRotationLeft90 … Rotated 270. • SCBBitmapConverterRotationRotate180 … Rotated 180.	SCBBitmapConverterRotation

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation width:(NSInteger)width {
    UIImage *starLogoImage = [UIImage imageNamed:@"StarLogoImage"];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:[@"\n*Normal, Alignment:Center*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmapWithAlignment:starLogoImage diffusion:YES position:SCBAlignmentPositionCenter];

    [builder appendData:[@"\n*width:Full, bothScale:YES, Alignment:Center*\n"
dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmapWithAlignment:starLogoImage diffusion:YES width:width bothScale:YES
position:SCBAlignmentPositionCenter];
    [builder appendData:[@"\n*width:Full, bothScale:NO, Alignment:Center*\n"
dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmapWithAlignment:starLogoImage diffusion:YES width:width bothScale:NO
position:SCBAlignmentPositionCenter];

    [builder appendData:[@"\n*Rotate180, Alignment:Center*\n" dataUsingEncoding:NSASCIIStringEncoding]];
    [builder appendBitmapWithAlignment:starLogoImage diffusion:YES
rotation:SCBBitmapConverterRotationRotate180 position:SCBAlignmentPositionCenter];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.38 appendBlackMark Method

Black mark command is generated and added to the commands property.

Declaration
- (void)appendBlackMark:(SCBBlackMarkType)type;

Parameter

Name	Description	Object type
type	Black mark. • SCBBlackMarkTypeInvalid … Black mark invalid. • SCBBlackMarkTypeValid … Black mark valid. • SCBBlackMarkTypeValidWithDetection … Black mark valid with detection.	SCBBlackMarkType

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendBlackMark:SCBBlackMarkTypeValid];

    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

//  [builder appendBlackMark:SCBBlackMarkTypeInvalid];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.39 beginPageMode Method

Begin page mode command is generated and added to the commands property.

Declaration
- (void)beginPageMode:(CGRect)rect rotation:(SCBBitmapConverterRotation)rotation;

Parameter

Name	Description	Object type
rect	Location and size. (Units : Dots)	CGRect
rotation	Print direction. • SCBBitmapConverterRotationNormal … Rotated 0. • SCBBitmapConverterRotationRight90 … Rotated 90. • SCBBitmapConverterRotationLeft90 … Rotated 270. • SCBBitmapConverterRotationRotate180 … Rotated 180.	SCBBitmapConverterRotation

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation width:(NSInteger)width {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    int height = 30 * 8;     // 30mm!!!

    CGRect rect;

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:[@"\n*Rotate180 Text*\n" dataUsingEncoding:NSASCIIStringEncoding]];

    rect = CGRectMake(0, 0, width,  height);

    [builder beginPageMode:rect rotation:SCBBitmapConverterRotationRotate180];

    [builder appendPageModeVerticalAbsolutePosition:height / 2];

    [builder appendDataWithAbsolutePosition:otherData position:width / 2];

    [builder endPageMode];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.40 endPageMode Method

End page mode command is generated and added to the commands property.

Declaration
- (void)endPageMode;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation width:(NSInteger)width {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    int height = 30 * 8;    // 30mm!!!

    CGRect rect;

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:[@"\n*Rotate180 Text*\n" dataUsingEncoding:NSASCIIStringEncoding]];

    rect = CGRectMake(0, 0, width,  height);

    [builder beginPageMode:rect rotation:SCBBitmapConverterRotationRotate180];

    [builder appendPageModeVerticalAbsolutePosition:height / 2];

    [builder appendDataWithAbsolutePosition:otherData position:width / 2];

    [builder endPageMode];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.41 appendPageModeVerticalAbsolutePosition Method

Vertical absolute position in page mode command is generated and added to the commands property.

Declaration
- (void)appendPageModeVerticalAbsolutePosition:(NSInteger)position;

Parameter

Name	Description	Object type
position	Vertical absolute position in page mode. (Units : Dots)	NSInteger

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation width:(NSInteger)width {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    int height = 30 * 8;     // 30mm!!!

    CGRect rect;

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:[@"\n*Rotate180 Text*\n" dataUsingEncoding:NSASCIIStringEncoding]];

    rect = CGRectMake(0, 0, width,  height);

    [builder beginPageMode:rect rotation:SCBBitmapConverterRotationRotate180];

    [builder appendPageModeVerticalAbsolutePosition:height / 2];

    [builder appendDataWithAbsolutePosition:otherData position:width / 2];

    [builder endPageMode];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.42 appendPageModeRotation Method

Print direction in page mode command is generated and added to the commands property.

Declaration
- (void)appendPageModeRotation:(SCBBitmapConverterRotation)rotation;

Parameter

Name	Description	Object type
rotation	Print direction. • SCBBitmapConverterRotationNormal … Rotated 0. • SCBBitmapConverterRotationRight90 … Rotated 90. • SCBBitmapConverterRotationLeft90 … Rotated 270. • SCBBitmapConverterRotationRotate180 … Rotated 180.	SCBBitmapConverterRotation

Return value

Description	Object type
-	-

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation width:(NSInteger)width {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    int height = 30 * 8;     // 30mm!!!

    CGRect rect;

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:[@"\n*Mixed Text*\n" dataUsingEncoding:NSASCIIStringEncoding]];

    rect = CGRectMake(0, 0, width,  height);

    [builder beginPageMode:rect rotation:SCBBitmapConverterRotationNormal];

    [builder appendPageModeVerticalAbsolutePosition:height / 2];

    [builder appendDataWithAbsolutePosition:otherData position:width / 2];

    [builder appendPageModeRotation:SCBBitmapConverterRotationRotate180];

    [builder appendPageModeVerticalAbsolutePosition:height / 2];

    [builder appendDataWithAbsolutePosition:otherData position:width / 2];

    [builder endPageMode];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.43 appendPrintableArea Method

Set command of the printable area is generated and added to the commands property.

Declaration
- (void)appendPrintableArea:(SCBPrintableAreaType)type;

Parameter

Name	Description	Object type
type	Printable area • SCBPrintableAreaTypeStandard …Standard type • SCBPrintableAreaTypeType1 …Type1 • SCBPrintableAreaTypeType2 …Type2 • SCBPrintableAreaTypeType3 …Type3 • SCBPrintableAreaTypeType4 …Type4	SCBPrintableAreaType

Return value

Description	Object type
-	-

The relationship between the specified PrintableAreaType and the print area to be set is the following.

SCBPrintableAreaType	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S240i	SM-S220i	SM-S230i	SM-T300i/T300	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i/T300 StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
SCBPrintableAreaTypeStandard	48	72 *1	48	72 *1	72 *1	72 *1	72 *1	104 *2	-	-	-	-	-	64 *1*3	-	-	-	-	-	-	72 *1	63
SCBPrintableAreaTypeType1	54	-	54	52.5	51	-	52.5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	48
SCBPrintableAreaTypeType2	-	48	-	-	-	-	80 *1	72	-	-	-	-	-	-	-	-	-	-	-	-	-	60
SCBPrintableAreaTypeType3	50.8	50.8	50.8	50.8	-	50.8	50.8	-	-	-	-	-	-	45 *3	-	-	-	-	-	-	50.8	45
SCBPrintableAreaTypeType4	-	-	-	52	-	-	52	-	-	-	-	-	-	-	-	-	-	-	-	-	-	57

※Units : millimeter

-: Ignored

*1: When 2-inch paper is used, do not set a print area with width 3 inches.

*2: When 3-inch paper is used, do not set a print area with width 4 inches.

*:3: When "Maximum number of lines" is set for the number of memory switch print lines, 72 mm is set for Standard and 50.8 mm is set for Type 3.
   (Bit 4 of memory switch 4 is set to 1.)

Example

```
+ (NSData *)createPrintableAreaDataWithEmulation:(StarIoExtEmulation)emulation
                                type:(SCBPrintableAreaType)type {
    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendPrintableArea:type];

    switch (type) {
        case SCBPrintableAreaTypeStandard:
            [builder appendData:[@"*Standard*\n" dataUsingEncoding:NSASCIIStringEncoding]];
            break;
        case SCBPrintableAreaTypeType1:
            [builder appendData:[@"*Type1*\n" dataUsingEncoding:NSASCIIStringEncoding]];
            break;
        case SCBPrintableAreaTypeType2:
            [builder appendData:[@"*Type2*\n" dataUsingEncoding:NSASCIIStringEncoding]];
            break;
        case SCBPrintableAreaTypeType3:
            [builder appendData:[@"*Type3*\n" dataUsingEncoding:NSASCIIStringEncoding]];
            break;
        case SCBPrintableAreaTypeType4:
            [builder appendData:[@"*Type4*\n" dataUsingEncoding:NSASCIIStringEncoding]];
            break;
    }

    UIImage *image = [UIImage imageNamed:@"PrintableAreaImage.png"];
    [builder appendBitmap:image diffusion:true];

    NSData *data1 = [@"123456789" dataUsingEncoding:NSASCIIStringEncoding];
    NSData *data2 = [@"0" dataUsingEncoding:NSASCIIStringEncoding];

    for (int i = 0; i < 8; i++) {
        [builder appendData:data1];
        [builder appendDataWithInvert:data2];
    }

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return builder.commands;
}
```

Refer to ApiFunctions.m.

10.44 appendCjkUnifiedIdeographFont メソッド

A command to set the UTF-8 CJK Unified Ideograph font is generated and added to the commands property.

Declaration

- (void)appendCjkUnifiedIdeographFont:(NSArray<NSNumber *> *)fonts;

Parameter

Name	Description	Object type
fonts	Array of CJK Unified Ideograph font in UTF-8 • SCBCjkUnifiedIdeographFontJapanese … Japanese font • SCBCjkUnifiedIdeographFontSimplifiedChinese … Simplified Chinese font • SCBCjkUnifiedIdeographFontTraditionalChinese … Traditional Chinese font • SCBCjkUnifiedIdeographFontHangul … Hangul font When two or more fonts are specified, the font specified earlier is printed first. If nil or empty array is specified for the argument, the font setting of the CJK Unified Ideograph is returned to the memory switch setting.	NSArray<NSNumber *> *

Return value

Description	Object type
-	-

Example

```
- (void)append3inchTextReceiptData:(ISCBBuilder *)builder utf8:(BOOL)utf8 {
    NSStringEncoding encoding = NSUTF8StringEncoding;

    // This function is supported by TSP650II(JP2/TW models only) with F/W version 4.0 or later and mC-Print 2/3.
    // Switch Kanji/Hangul font by specifying the font for Unicode CJK Unified Ideographs before each word.

    [builder appendCodePage:SCBCodePageTypeUTF8];

    [builder appendCharacterSpace:0];

    [builder appendAlignment:SCBAlignmentPositionCenter];

    [builder appendData:[@"2017 / 5 / 15 AM 10:00\n" dataUsingEncoding:encoding]];

    [builder appendMultiple:2 height:2];

    [builder appendCjkUnifiedIdeographFont:@[@(SCBCjkUnifiedIdeographFontJapanese)]];
    [builder appendData:[@"受付票 " dataUsingEncoding:encoding]];

    [builder appendCjkUnifiedIdeographFont:@[@(SCBCjkUnifiedIdeographFontTraditionalChinese)]];
    [builder appendData:[@"排號單\n" dataUsingEncoding:encoding]];

    [builder appendCjkUnifiedIdeographFont:@[@(SCBCjkUnifiedIdeographFontSimplifiedChinese)]];
    [builder appendData:[@"排号单 " dataUsingEncoding:encoding]];

    [builder appendCjkUnifiedIdeographFont:@[@(SCBCjkUnifiedIdeographFontHangul)]];
    [builder appendData:[@"접수표\n\n" dataUsingEncoding:encoding]];

    [builder appendMultiple:1 height:1];

    [builder appendCjkUnifiedIdeographFont:@[]];
    [builder appendDataWithMultiple:[@"1\n" dataUsingEncoding:encoding] width:6 height:6];
    [builder appendData:[@"--------------------------------------\n" dataUsingEncoding:encoding]];

    [builder appendCjkUnifiedIdeographFont:@[@(SCBCjkUnifiedIdeographFontJapanese)]];
    [builder appendData:[@"ご本人がお持ちください。\n" dataUsingEncoding:encoding]];
    [builder appendData:[@"※紛失しないようにご注意ください。\n" dataUsingEncoding:encoding]];
}
```

Refer to Utf8MultiLanguageReceiptsImpl.m.

10.45 commands Property

Generated commands.

Declaration

@property (nonatomic, readonly) NSMutableData *commands;

Value

Description	Object type
Generated commands.	NSMutableData

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSASCIIStringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to ApiFunctions.m.

10.46 SCBInitializationType Constants

Initialization constants.

Declaration

typedef NS_ENUM(NSUInteger, SCBInitializationType) {
    SCBInitializationTypeCommand
// SCBInitializationTypeReset,
// SCBInitializationTypeResetWithPrint
};

Constants

Name	Description
SCBInitializationTypeCommand	Command initialization.
~~SCBInitializationTypeReset~~	~~Reset printer.~~
~~SCBInitializationTypeResetWithPrint~~	~~Reset printer (execute self print).~~

Refer to ApiFunctions.m.

10.47 SCBFontStyleType Constants

Font style constants.

Declaration

typedef NS_ENUM(NSUInteger, SCBFontStyleType) {
    SCBFontStyleTypeA,
    SCBFontStyleTypeB
};

Constants

Name	Description
SCBFontStyleTypeA	Font-A (12 x 24 dots) / Specify 7 x 9 font (half dots)
SCBFontStyleTypeB	Font-B (9 x 24 dots) / Specify 5 x 9 font (2P-1)

Refer to ApiFunctions.m and each Command Specifications.

10.48 SCBCodePageType Constants

Code Page constants.

Declaration

```
typedef NS_ENUM(NSUInteger, SCBCodePageType) {
    SCBCodePageTypeCP437,
    SCBCodePageTypeCP737,
    SCBCodePageTypeCP772,
    SCBCodePageTypeCP774,
    SCBCodePageTypeCP851,
    SCBCodePageTypeCP852,
    SCBCodePageTypeCP855,
    SCBCodePageTypeCP857,
    SCBCodePageTypeCP858,
    SCBCodePageTypeCP860,
    SCBCodePageTypeCP861,
    SCBCodePageTypeCP862,
    SCBCodePageTypeCP863,
    SCBCodePageTypeCP864,
    SCBCodePageTypeCP865,
    SCBCodePageTypeCP866,
    SCBCodePageTypeCP869,
    SCBCodePageTypeCP874,
    SCBCodePageTypeCP928,
    SCBCodePageTypeCP932,
    SCBCodePageTypeCP998,
    SCBCodePageTypeCP999,
    SCBCodePageTypeCP1001,
    SCBCodePageTypeCP1250,
    SCBCodePageTypeCP1251,
    SCBCodePageTypeCP1252,
    SCBCodePageTypeCP2001,
    SCBCodePageTypeCP3001,
    SCBCodePageTypeCP3002,
    SCBCodePageTypeCP3011,
    SCBCodePageTypeCP3012,
    SCBCodePageTypeCP3021,
    SCBCodePageTypeCP3041,
    SCBCodePageTypeCP3840,
    SCBCodePageTypeCP3841,
    SCBCodePageTypeCP3843,
    SCBCodePageTypeCP3844,
    SCBCodePageTypeCP3845,
    SCBCodePageTypeCP3846,
    SCBCodePageTypeCP3847,
    SCBCodePageTypeCP3848,
    SCBCodePageTypeUTF8,
    SCBCodePageTypeBlank
};
```

Constants

Name	Description
SCBCodePageTypeCP437	CodePage437 (USA, Std. Europe).
SCBCodePageTypeCP737	Codepage 737 (Greek).
SCBCodePageTypeCP772	Codepage 772 (Lithuanian).
SCBCodePageTypeCP774	Codepage 774 (Lithuanian).
SCBCodePageTypeCP851	Codepage 851 (Greek).
SCBCodePageTypeCP852	Codepage 852 (Latin-2).
SCBCodePageTypeCP855	Codepage 855 (Cyrillic Bulgarian).
SCBCodePageTypeCP857	Codepage 857 (Turkey).
SCBCodePageTypeCP858	Codepage 858 (Multilingual).
SCBCodePageTypeCP860	Codepage 860 (Portuguese).
SCBCodePageTypeCP861	Codepage 861 (Icelandic).
SCBCodePageTypeCP862	Codepage 862 (Israel (Hebrew)).
SCBCodePageTypeCP863	Codepage 863 (Canadian French).
SCBCodePageTypeCP864	Codepage 864 (Arabic).
SCBCodePageTypeCP865	Codepage 865 (Nordic).
SCBCodePageTypeCP866	Codepage 866 (Cyrillic Russian).
SCBCodePageTypeCP869	Codepage 869 (Greek).
SCBCodePageTypeCP874	Codepage 874 (Thai).
SCBCodePageTypeCP928	Codepage 928 (Greek).
SCBCodePageTypeCP932	Katakana.
SCBCodePageTypeCP998	Normal.
SCBCodePageTypeCP999	Codepage 1252 (Windows Latin-1).
SCBCodePageTypeCP1001	Codepage 1001 (Arabic).
SCBCodePageTypeCP1250	Codepage 1250 (Windows Latin-2).
SCBCodePageTypeCP1251	Codepage 1251 (Windows Cyrillic).
SCBCodePageTypeCP1252	Codepage 1252 (Windows Latin-1).
SCBCodePageTypeCP2001	Codepage 2001 (Lithuanian-KBL).
SCBCodePageTypeCP3001	Codepage 3001 (Estonian-1).
SCBCodePageTypeCP3002	Codepage 3002 (Estonian-2).
SCBCodePageTypeCP3011	Codepage 3011 (Latvian-1).
SCBCodePageTypeCP3012	Codepage 3012 (Latvian-2).
SCBCodePageTypeCP3021	Codepage 3021 (Bulgarian).
SCBCodePageTypeCP3041	Codepage 3041 (Maltese).
SCBCodePageTypeCP3840	Codepage 3840 (IBM-Russian).
SCBCodePageTypeCP3841	Codepage 3841 (Gost).
SCBCodePageTypeCP3843	Codepage 3843 (Polish).
SCBCodePageTypeCP3844	Codepage 3844 (CS2).
SCBCodePageTypeCP3845	Codepage 3845 (Hungarian).
SCBCodePageTypeCP3846	Codepage 3846 (Turkish).
SCBCodePageTypeCP3847	Codepage 3847 (Brazil-ABNT).
SCBCodePageTypeCP3848	Codepage 3848 (Brazil-ABICOMP).
SCBCodePageTypeUTF8	UTF-8.
SCBCodePageTypeBlank	User Setting Blank Code Page.

Refer to ApiFunctions.m and each Command Specifications.

10.49 SCBInternationalType Constants

International character constants.

Declaration

typedef NS_ENUM(NSUInteger, SCBInternationalType) {
    SCBInternationalTypeUSA,
    SCBInternationalTypeFrance,
    SCBInternationalTypeGermany,
    SCBInternationalTypeUK,
    SCBInternationalTypeDenmark,
    SCBInternationalTypeSweden,
    SCBInternationalTypeItaly,
    SCBInternationalTypeSpain,
    SCBInternationalTypeJapan,
    SCBInternationalTypeNorway,
    SCBInternationalTypeDenmark2,
    SCBInternationalTypeSpain2,
    SCBInternationalTypeLatinAmerica,
    SCBInternationalTypeKorea,
    SCBInternationalTypeIreland,
    SCBInternationalTypeLegal
};

Constants

Name	Description
SCBInternationalTypeUSA	USA.
SCBInternationalTypeFrance	France.
SCBInternationalTypeGermany	Germany.
SCBInternationalTypeUK	UK.
SCBInternationalTypeDenmark	Denmark.
SCBInternationalTypeSweden	Sweden.
SCBInternationalTypeItaly	Italy.
SCBInternationalTypeSpain	Spain.
SCBInternationalTypeJapan	Japan.
SCBInternationalTypeNorway	Norway.
SCBInternationalTypeDenmark2	Denmark ‖.
SCBInternationalTypeSpain2	Spain ‖.
SCBInternationalTypeLatinAmerica	Latin America.
SCBInternationalTypeKorea	Korea.
SCBInternationalTypeIreland	Ireland.
SCBInternationalTypeLegal	Legal.

Refer to ApiFunctions.m and each Command Specifications.

10.50 SCBLogoSize Constants

Logo size constants.

Declaration
typedef NS_ENUM(NSUInteger, SCBLogoSize) {
    SCBLogoSizeNormal,
    SCBLogoSizeDoubleWidth,
    SCBLogoSizeDoubleHeight,
    SCBLogoSizeDoubleWidthDoubleHeight
};

Constants

Name	Description
SCBLogoSizeNormal	Normal.
SCBLogoSizeDoubleWidth	Double Width.
SCBLogoSizeDoubleHeight	Double Height.
SCBLogoSizeDoubleWidthDoubleHeight	Double Width / Height.

Refer to ApiFunctions.m and each Command Specifications.

10.51 SCBAlignmentPosition Constants

Alignment position constants.

Declaration
typedef NS_ENUM(NSUInteger, SCBAlignmentPosition) {
    SCBAlignmentPositionLeft,
    SCBAlignmentPositionCenter,
    SCBAlignmentPositionRight
};

Constants

Name	Description
SCBAlignmentPositionLeft	Left alignment.
SCBAlignmentPositionCenter	Center alignment.
SCBAlignmentPositionRight	Right alignment.

Refer to ApiFunctions.m and each Command Specifications.

10.52 SCBCutPaperAction Constants

Paper cut constants.

Declaration
typedef NS_ENUM(NSInteger, SCBCutPaperAction) {
    SCBCutPaperActionFullCut,
    SCBCutPaperActionPartialCut,
    SCBCutPaperActionFullCutWithFeed,
    SCBCutPaperActionPartialCutWithFeed
};

Constants

Name	Description
SCBCutPaperActionFullCut	Full Cut.
SCBCutPaperActionPartialCut	Partial Cut.
SCBCutPaperActionFullCutWithFeed	Full Cut with Feed.
SCBCutPaperActionPartialCutWithFeed	Partial Cut with Feed.

Refer to ApiFunctions.m and each Command Specifications.

10.53 SCBPeripheralChannel Constants

Cash drawer channel constants.

Declaration
typedef NS_ENUM(NSInteger, SCBPeripheralChannel) {
    SCBPeripheralChannelNo1,
    SCBPeripheralChannelNo2
};

Constants

Name	Description
SCBPeripheralChannelNo1	Channel1.
SCBPeripheralChannelNo2	Channel2.

Refer to ApiFunctions.m.

10.54 SCBSoundChannel Constants

Sound/Buzzer channel constants.

Declaration
typedef NS_ENUM(NSInteger, SCBSoundChannel) {
   SCBSoundChannelNo1,
   SCBSoundChannelNo2
};

Constants

Name	Description
SCBSoundChannelNo1	Channel1.
SCBSoundChannelNo2	Channel2.

Refer to ApiFunctions.m.

10.55 SCBBarcodeSymbology Constants

Barcode symbology constants.

Declaration
typedef NS_ENUM(NSUInteger, SCBBarcodeSymbology) {
   SCBBarcodeSymbologyUPCE,
   SCBBarcodeSymbologyUPCA,
   SCBBarcodeSymbologyJAN8,
   SCBBarcodeSymbologyJAN13,
   SCBBarcodeSymbologyCode39,
   SCBBarcodeSymbologyITF,
   SCBBarcodeSymbologyCode128,
   SCBBarcodeSymbologyCode93,
// SCBBarcodeSymbologyCodabar,
   SCBBarcodeSymbologyNW7
};

Constants

Name	Description
SCBBarcodeSymbologyUPCE	UPC-E.
SCBBarcodeSymbologyUPCA	UPC-A.
SCBBarcodeSymbologyJAN8	JAN/EAN8.
SCBBarcodeSymbologyJAN13	JAN/EAN13.
SCBBarcodeSymbologyCode39	Code39.
SCBBarcodeSymbologyITF	ITF.
SCBBarcodeSymbologyCode128	Code128.
SCBBarcodeSymbologyCode93	Code93.
~~SCBBarcodeSymbologyCodabar~~	~~Codabar.~~
SCBBarcodeSymbologyNW7	NW7.

Refer to ApiFunctions.m and each Command Specifications.

10.56 SCBBarcodeWidth Constants

Barcode width constants.
By specifying ExtMode, the barcode mode conforming to each command specification is set.

Declaration

typedef NS_ENUM(NSUInteger, SCBBarcodeWidth) {
    SCBBarcodeWidthMode1,
    SCBBarcodeWidthMode2,
    SCBBarcodeWidthMode3,
    SCBBarcodeWidthMode4,
    SCBBarcodeWidthMode5,
    SCBBarcodeWidthMode6,
    SCBBarcodeWidthMode7,
    SCBBarcodeWidthMode8,
    SCBBarcodeWidthMode9,
    SCBBarcodeWidthExtMode1,
    SCBBarcodeWidthExtMode2,
    SCBBarcodeWidthExtMode3,
    SCBBarcodeWidthExtMode4,
    SCBBarcodeWidthExtMode5,
    SCBBarcodeWidthExtMode6,
    SCBBarcodeWidthExtMode7,
    SCBBarcodeWidthExtMode8,
    SCBBarcodeWidthExtMode9
};

Constants

Name	Description
SCBBarcodeWidthMode1	Mode1.
SCBBarcodeWidthMode2	Mode2.
SCBBarcodeWidthMode3	Mode3.
SCBBarcodeWidthMode4	Mode4.
SCBBarcodeWidthMode5	Mode5.
SCBBarcodeWidthMode6	Mode6.
SCBBarcodeWidthMode7	Mode7.
SCBBarcodeWidthMode8	Mode8.
SCBBarcodeWidthMode9	Mode9.
SCBBarcodeWidthExtMode1	ExtMode1.
SCBBarcodeWidthExtMode2	ExtMode2.
SCBBarcodeWidthExtMode3	ExtMode3.
SCBBarcodeWidthExtMode4	ExtMode4.
SCBBarcodeWidthExtMode5	ExtMode5.
SCBBarcodeWidthExtMode6	ExtMode6.
SCBBarcodeWidthExtMode7	ExtMode7.
SCBBarcodeWidthExtMode8	ExtMode8.
SCBBarcodeWidthExtMode9	ExtMode9.

Refer to ApiFunctions.m and each Command Specifications.

10.57 SCBPdf417Level Constants

PDF417 ECC (security level) constants.

Declaration
typedef NS_ENUM(NSUInteger, SCBPdf417Level) {
   SCBPdf417LevelECC0,
   SCBPdf417LevelECC1,
   SCBPdf417LevelECC2,
   SCBPdf417LevelECC3,
   SCBPdf417LevelECC4,
   SCBPdf417LevelECC5,
   SCBPdf417LevelECC6,
   SCBPdf417LevelECC7,
   SCBPdf417LevelECC8
};

Constants

Name	Description
SCBPdf417LevelECC0	Level0.
SCBPdf417LevelECC1	Level1.
SCBPdf417LevelECC2	Level2.
SCBPdf417LevelECC3	Level3.
SCBPdf417LevelECC4	Level4.
SCBPdf417LevelECC5	Level5.
SCBPdf417LevelECC6	Level6.
SCBPdf417LevelECC7	Level7.
SCBPdf417LevelECC8	Level8.

Refer to ApiFunctions.m and each Command Specifications.

10.58 SCBQrCodeModel Constants

QR code model constants.

Declaration
typedef NS_ENUM(NSUInteger, SCBQrCodeModel) {
   SCBQrCodeModelNo1,
   SCBQrCodeModelNo2
};

Constants

Name	Description
SCBQrCodeModelNo1	Model1.
SCBQrCodeModelNo2	Model2.

Refer to ApiFunctions.m and each Command Specifications.

10.59 SCBQrCodeLevel Constants

QR code mistake correction level constants.

Declaration

typedef NS_ENUM(NSUInteger, SCBQrCodeLevel) {
    SCBQrCodeLevelL,
    SCBQrCodeLevelM,
    SCBQrCodeLevelQ,
    SCBQrCodeLevelH
};

Constants

Name	Description
SCBQrCodeLevelL	Level L.
SCBQrCodeLevelM	Level M.
SCBQrCodeLevelQ	Level Q.
SCBQrCodeLevelH	Level H.

Refer to ApiFunctions.m and each Command Specifications.

10.60 SCBBitmapConverterRotation Constants

Bitmap rotation constants.

Declaration

typedef NS_ENUM(NSUInteger, SCBBitmapConverterRotation) {
    SCBBitmapConverterRotationNormal,
    SCBBitmapConverterRotationRight90,
    SCBBitmapConverterRotationLeft90,
    SCBBitmapConverterRotationRotate180
};

Constants

Name	Description
SCBBitmapConverterRotationNormal	Rotated 0.
SCBBitmapConverterRotationRight90	Rotated 90.
SCBBitmapConverterRotationLeft90	Rotated 270.
SCBBitmapConverterRotationRotate180	Rotated 180.

Refer to ApiFunctions.m.

10.61 SCBBlackMarkType Constants

Black mark constants.

Declaration
typedef NS_ENUM(NSInteger, SCBBlackMarkType) {
    SCBBlackMarkTypeInvalid,
    SCBBlackMarkTypeValid,
    SCBBlackMarkTypeValidWithDetection
};

Constants

Name	Description
SCBBlackMarkTypeInvalid	Black mark invalid.
SCBBlackMarkTypeValid	Black mark valid.
SCBBlackMarkTypeValidWithDetection	Black mark valid with detection.

Refer to ApiFunctions.m and each Command Specifications.

10.62 SCBPrintableAreaType Constants

Printable are constants.

Declaration
typedef NS_ENUM(NSInteger, SCBPrintableAreaType) {
    SCBPrintableAreaTypeStandard,
    SCBPrintableAreaTypeType1,
    SCBPrintableAreaTypeType2,
    SCBPrintableAreaTypeType3,
    SCBPrintableAreaTypeType4
}

Constants

Name	Description
SCBPrintableAreaTypeStandard	Standard type
SCBPrintableAreaTypeType1	Type1
SCBPrintableAreaTypeType2	Type2
SCBPrintableAreaTypeType3	Type3
SCBPrintableAreaTypeType4	Type4

Refer to ApiFunctions.m.

10.63 SCBCjkUnifiedIdeographFont Constants

Constants of CJK Unified Ideograph font in UTF-8

Declaration
typedef NS_ENUM(NSInteger, SCBCjkUnifiedIdeographFont) {
    SCBCjkUnifiedIdeographFontJapanese   = 1,
    SCBCjkUnifiedIdeographFontSimplifiedChinese = 2,
    SCBCjkUnifiedIdeographFontTraditionalChinese = 3,
    SCBCjkUnifiedIdeographFontHangul   = 4
};

Constants

Name	Description
SCBCjkUnifiedIdeographFontJapanese	Japanese font
SCBCjkUnifiedIdeographFontSimplifiedChinese	Simplified Chinese font
SCBCjkUnifiedIdeographFontTraditionalChinese	Traditional Chinese font
SCBCjkUnifiedIdeographFontHangul	Hangul font

Refer to Utf8MultiLanguageReceiptsImpl.m.

11 ISDCBBuilder interface (StarIO_Extension.framework)

An interface to provide functions to generate commands for the customer display control.

Method

Name	Description
appendByte	Adds data (text or command) to the commands property.
appendData	
appendBytes	
appendBackSpace	Generates a backspace command, and then adds it to the commands property.
appendHorizontalTab	Generates a horizontal tab command, and then adds it to the commands property.
appendLineFeed	Generates a line feed command, and then adds it to the commands property.
appendCarriageReturn	Generates a carriage-return command, and then adds it to the commands property.
appendBitmap	Generates a graphic display command, and then adds it to the commands property.
appendInternational	Generates an international character specification command, and then adds it to the commands property.
appendCodePage	Generates a code page specification command, and then adds it to the commands property.
appendDeleteToEndOfLine	Generates a command to clear the screen until the end of the line, and then adds it to the commands property.
appendClearScreen	Generates a command to clear the screen, and then adds it to the commands property.
appendHomePosition	Generates a command to move the cursor to the home position, and then adds it to the commands property.
appendTurnOn	Generates a command to turn on/off the backlight, and then adds it to the commands property.
appendSpecifiedPosition	Generates a command to move the cursor to the specified position, and then adds it to the commands property.
appendCursorMode	Generates a command to change the cursor mode, and then adds it to the commands property.
appendContrastMode	Generates a command to change the contrast mode, and then adds it to the commands property.
appendUserDefinedCharacter	Generates a command to register user-defined characters (SBCS), and then adds it to the commands property.
appendUserDefinedDbcsCharacter	Generates a command to register user-defined characters (DBCS), and then adds it to the commands property.

Property

Name	Description
commands	Generated or added commands. * readonly
passThroughCommands	Generated or added commands to which adds a pass-through command. * readonly

Constant

Name	Description
SDCBInternationalType	International character constants.
SDCBCodePageType	Code Page constants.
SDCBCursorMode	Cursor Mode constants.
SDCBContrastMode	ContrastMode constants.

11.1 Model : ISDCBBuilder interface

The ISDCBBuilder interface is only for models which support a customer display. For the supporting models, refer to Supported peripherals.

11.2 appendData Method

Adds data (text or command) to the commands property.

Declaration

- (void)appendByte:(unsigned char)data;

- (void)appendData:(NSData *)otherData;

- (void)appendBytes:(const void *)bytes length:(NSUInteger)length;

Parameter

Name	Description	Object type
data		unsigned char
otherData	Data (Text,Command)	NSData
bytes		const void *
length		NSUInteger

Return value

Description	Object type
-	-

Example

```
+ (void)appendTextPattern:(ISDCBBuilder *)builder number:(int)number {
// [builder appendClearScreen];
   [builder appendCursorMode:SDCBCursorModeOff];
   [builder appendSpecifiedPosition:1 y:1];

   unsigned char pattern1[] =
"\x020\x021\x022\x023\x024\x025\x026\x027\x028\x029\x02a\x02b\x02c\x02d\x02e\x02f\x030\x031\x032\x033"

"\x034\x035\x036\x037\x038\x039\x03a\x03b\x03c\x03d\x03e\x03f\x040\x041\x042\x043\x044\x045\x046\x047";
   ...

   switch (number) {
     default : [builder appendBytes:pattern1 length:sizeof(pattern1)]; break;    // 0
     case 1  : [builder appendBytes:pattern2 length:sizeof(pattern2)]; break;
     case 2  : [builder appendBytes:pattern3 length:sizeof(pattern3)]; break;
     case 3  : [builder appendBytes:pattern4 length:sizeof(pattern4)]; break;
     case 4  : [builder appendBytes:pattern5 length:sizeof(pattern5)]; break;
     case 5  : [builder appendBytes:pattern6 length:sizeof(pattern6)]; break;
   }
}
```

Refer to DisplayFunctions.m.

11.3 appendBackSpace Method

Generates a backspace command, and then adds it to the commands property.

Declaration
void appendBackSpace();

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

11.4 appendHorizontalTab Method

Generates a horizontal tab command, and then adds it to the commands property.

Declaration
- (void)appendHorizontalTab;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

11.5 appendLineFeed Method

Generates a line feed command, and then adds it to the commands property.

Declaration
- (void)appendLineFeed;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

11.6 appendCarriageReturn Method

Generates a carriage-return command, and then adds it to the commands property.

Declaration
- (void)appendCarriageReturn;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

11.7 appendBitmap Method

Generates a graphic display command, and then adds it to the commands property.

Declaration
- (void)appendBitmap:(UIImage *)image diffusion:(BOOL)diffusion;

Parameter

Name	Description	Object type
image	Image you want to display Specify the instance of the 160 × 40 dots monochrome Bitmap class. If a different sized Bitmap is specified, it is resized to 160 × 40 dots. The Bitmap pixels are automatically converted to monochrome value.	UIImage *
diffusion	Random dither. • YES … Valid • NO … Invalid	BOOL

Return value

Description	Object type
-	-

Example

```
+ (void)appendGraphicPattern:(ISDCBBuilder *)builder number:(int)number {
// [builder appendClearScreen];
   [builder appendCursorMode:SDCBCursorModeOff];
// [builder appendSpecifiedPosition:1 y:1];

   UIImage *image;

   switch (number) {
      default : image = [UIImage imageNamed:@"DisplayImage1.png"]; break;    // 0
      case 1  : image = [UIImage imageNamed:@"DisplayImage2.png"]; break;
      case 2  : image = [UIImage imageNamed:@"DisplayImage3.png"]; break;
      case 3  : image = [UIImage imageNamed:@"DisplayImage4.png"]; break;
   }

   [builder appendBitmap:image diffusion:YES];
}
```

Refer to DisplayFunctions.m.

11.8 appendInternational Method

Generates an international character specification command, and then adds it to the commands property.

Declaration

- (void)appendInternational:(SDCBInternationalType)type;

Parameter

Name	Description	Object type
type	International character constants.	SDCBInternationalType

Return value

Description	Object type
-	-

Example

```
+ (void)appendCharacterSet:(ISDCBBuilder *)builder internationalType:(SDCBInternationalType)internationalType
codePageType:(SDCBCodePageType)codePageType {
// [builder appendClearScreen];
   [builder appendCursorMode:SDCBCursorModeOff];
   [builder appendSpecifiedPosition:1 y:1];

   [builder appendInternational:internationalType];
   [builder appendCodePage     :codePageType];

   unsigned char pattern1[] =
"\x02d\x020\x020\x020\x023\x024\x040\x05b\x05c\x05d\x05e\x060\x07b\x07c\x07d\x07e\x020\x020\x020\x02d"

"\x0a0\x0a1\x0a2\x0a3\x0a4\x0a5\x0a6\x0a7\x0a8\x0a9\x0aa\x0ab\x0ac\x0ad\x0ae\x0af\x0b0\x0b1\x0b2\x0b3";

   ...

   switch (codePageType) {
      default                     : [builder appendBytes:pattern1 length:sizeof(pattern1)]; break;     //
CP437,Katakana,CP850,CP860,CP863,CP865,CP1252,CP866,CP852,CP858
      case SDCBCodePageTypeJapanese          : [builder appendBytes:pattern2 length:sizeof(pattern2)]; break;
      case SDCBCodePageTypeSimplifiedChinese  : [builder appendBytes:pattern3 length:sizeof(pattern3)]; break;
      case SDCBCodePageTypeTraditionalChinese : [builder appendBytes:pattern4 length:sizeof(pattern4)]; break;
      case SDCBCodePageTypeHangul            : [builder appendBytes:pattern5 length:sizeof(pattern5)]; break;
   }
}
```

Refer to DisplayFunctions.m.

11.9 appendCodePage Method

Generates a code page specification command, and then adds it to the commands property.

Declaration

- (void)appendCodePage:(SDCBCodePageType)type;

Parameter

Name	Description	Object type
type	Code page constants.	SDCBCodePageType

Return value

Description	Object type
-	-

Example

```
+ (void)appendCharacterSet:(ISDCBBuilder *)builder internationalType:(SDCBInternationalType)internationalType
codePageType:(SDCBCodePageType)codePageType {
//  [builder appendClearScreen];
    [builder appendCursorMode:SDCBCursorModeOff];
    [builder appendSpecifiedPosition:1 y:1];

    [builder appendInternational:internationalType];
    [builder appendCodePage    :codePageType];

    unsigned char pattern1[] =
"\x02d\x020\x020\x020\x023\x024\x040\x05b\x05c\x05d\x05e\x060\x07b\x07c\x07d\x07e\x020\x020\x020\x02d"

"\x0a0\x0a1\x0a2\x0a3\x0a4\x0a5\x0a6\x0a7\x0a8\x0a9\x0aa\x0ab\x0ac\x0ad\x0ae\x0af\x0b0\x0b1\x0b2\x0b3";


    ...

    switch (codePageType) {
        default                          : [builder appendBytes:pattern1 length:sizeof(pattern1)]; break;     //
CP437,Katakana,CP850,CP860,CP863,CP865,CP1252,CP866,CP852,CP858
        case SDCBCodePageTypeJapanese          : [builder appendBytes:pattern2 length:sizeof(pattern2)]; break;
        case SDCBCodePageTypeSimplifiedChinese  : [builder appendBytes:pattern3 length:sizeof(pattern3)]; break;
        case SDCBCodePageTypeTraditionalChinese : [builder appendBytes:pattern4 length:sizeof(pattern4)]; break;
        case SDCBCodePageTypeHangul            : [builder appendBytes:pattern5 length:sizeof(pattern5)]; break;
    }
}
```

Refer to DisplayFunctions.m.

11.10 appendDeleteToEndOfLine Method

Generates a command to clear the screen until the end of the line, and then adds it to the commands property.

Declaration
- (void)appendDeleteToEndOfLine;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

11.11 appendClearScreen Method

Generates a command to clear the screen, and then adds it to the commands property.

Declaration
- (void)appendClearScreen;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

Example
```
+ (void)appendClearScreen:(ISDCBBuilder *)builder {
    [builder appendClearScreen];
}
```

Refer to DisplayFunctions.m.

11.12 appendHomePosition Method

Generates a command to move the cursor to the home position, and then adds it to the commands property.

Declaration
- (void)appendHomePosition;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

11.13 appendTurnOn Method

Generates a command to turn on/off the backlight, and then adds it to the commands property.

Declaration
- (void)appendTurnOn:(BOOL)turnOn;

Parameter

Name	Description	Object type
turnOn	Backlight ON/OFF	BOOL

Return value

Description	Object type
-	-

Example

```
+ (void)appendTurnOn:(ISDCBBuilder *)builder turnOn:(BOOL)turnOn {
    ...
    [builder appendTurnOn:turnOn];
}
```

Refer to DisplayFunctions.m.

11.14 appendSpecifiedPosition Method

Generates a command to move the cursor to the specified position, and then adds it to the commands property.

Declaration
- (void)appendSpecifiedPosition:(int)x y:(int)y;

Parameter

Name	Description	Object type
x	Digit position (Leftmost digit is 1)	int
y	Line position (top line is 1)	int

Return value

Description	Object type
-	-

Example

```
+ (void)appendCursorMode:(ISDCBBuilder *)builder cursorMode:(SDCBCursorMode)cursorMode {
//  [builder appendClearScreen];
    [builder appendCursorMode:SDCBCursorModeOff];
    [builder appendSpecifiedPosition:1 y:1];

    unsigned char pattern[] = "Star Micronics      "
                  "Total :        12345";

    [builder appendBytes:pattern length:sizeof(pattern)];

    [builder appendSpecifiedPosition:20 y:2];

    [builder appendCursorMode:cursorMode];
}
```
Refer to DisplayFunctions.m.

11.15 appendCursorMode Method

Generates a command to change the cursor mode, and then adds it to the commands property.

Declaration
- (void)appendCursorMode:(SDCBCursorMode)cursorMode;

Parameter

Name	Description	Object type
cursorMode	Cursor mode constants.	SDCBCursorMode

Return value

Description	Object type
-	-

Example

```
+ (void)appendCursorMode:(ISDCBBuilder *)builder cursorMode:(SDCBCursorMode)cursorMode {
// [builder appendClearScreen];
    [builder appendCursorMode:SDCBCursorModeOff];
    [builder appendSpecifiedPosition:1 y:1];

    unsigned char pattern[] = "Star Micronics       "
                    "Total :       12345";

    [builder appendBytes:pattern length:sizeof(pattern)];

    [builder appendSpecifiedPosition:20 y:2];

    [builder appendCursorMode:cursorMode];
}
```

Refer to DisplayFunctions.m.

11.16 appendContrastMode Method

Generates a command to change the contrast mode, and then adds it to the commands property.

Declaration
- (void)appendContrastMode:(SDCBContrastMode)contrastMode;

Parameter

Name	Description	Object type
contrastMode	Contrast mode constants.	SDCBContrastMode

Return value

Description	Object type
-	-

Example

```
+ (void)appendContrastMode:(ISDCBBuilder *)builder contrastMode:(SDCBContrastMode)contrastMode {
    ...
    [builder appendContrastMode:contrastMode];
}
```

Refer to DisplayFunctions.m.

11.17 appendUserDefinedCharacter Method

Generates a command to register user-defined characters (SBCS), and then adds it to the commands property.

Declaration

- (void)appendUserDefinedCharacter:(int)index code:(int)code font:(unsigned char *)font;

Parameter

Name	Description	Object type
index	Font number 00h ≦ index ≦ 1Fh	int
code	Character code of user-defined character. 20h ≦ code ≦ 7Fh If CodePageType.Japanese is specified with the appendCharacterSet method and the appendCodePage method, the following range is also supported. A0h ≦ code ≦ DFh	int
font	16-byte font data Refer to the font data format of the user-defined characters. Specifying null deletes the user-defined characters of the specified font No.	unsigned char *

Return value

Description	Object type
-	-

Example

```
+ (void)appendUserDefinedCharacter:(ISDCBBuilder *)builder set:(BOOL)set {
// [builder appendClearScreen];
   [builder appendCursorMode:SDCBCursorModeOff];
   [builder appendSpecifiedPosition:1 y:1];

   [builder appendInternational:SDCBInternationalTypeUSA];
   [builder appendCodePage     :SDCBCodePageTypeJapanese];

   if (set) {
       [builder appendUserDefinedCharacter:0 code:0x20 font:(unsigned char *)
"\x000\x000\x032\x000\x049\x000\x049\x07f\x026\x048\x000\x048\x000\x030\x000\x000"];

       [builder appendUserDefinedDbcsCharacter:0 code:0x8140 font:(unsigned char *)
"\x000\x000\x000\x000\x000\x000\x000\x000\x003\x020\x004\x090\x004\x090\x002\x060"

"\x000\x000\x007\x0f0\x004\x080\x004\x080\x003\x000\x000\x000\x000\x000\x000\x000"];
   }
   else {
       [builder appendUserDefinedCharacter:0 code:0x00 font:nil];

       [builder appendUserDefinedDbcsCharacter:0 code:0x0000 font:nil];
   }

   unsigned char pattern[] =
"\x05b\x020\x020\x053\x074\x061\x072\x020\x04d\x069\x063\x072\x06f\x06e\x069\x063\x073\x020\x020\x05d"

"\x05b\x081\x040\x081\x040\x083\x058\x083\x05e\x081\x05b\x090\x0b8\x096\x0a7\x081\x040\x081\x040\x05d";

   [builder appendBytes:pattern length:sizeof(pattern)];
}
```

Refer to DisplayFunctions.m.

11.18 appendUserDefinedDbcsCharacter Method

Generates a command to register user-defined characters (DBCS), and then adds it to the commands property.

Declaration

- (void)appendUserDefinedDbcsCharacter:(int)index code:(int)code font:(unsigned char *)font;

Parameter

Name	Description	Object type
index	Font number 00h ≦ index ≦ 0Fh	int
code	Character code of user-defined character. 20h ≦ code ≦ 7Fh If CodePageType.Japanese is specified with the appendCharacterSet method and the appendCodePage method, the following range is also supported. 80 00h ≦ code ≦ FF FFh	int
font	32-byte font data Refer to the font data format of the user-defined characters. Specifying null deletes the user-defined characters of the specified font No.	unsigned char *

Return value

Description	Object type
-	-

Example

```
+ (void)appendUserDefinedCharacter:(ISDCBBuilder *)builder set:(BOOL)set {
// [builder appendClearScreen];
   [builder appendCursorMode:SDCBCursorModeOff];
   [builder appendSpecifiedPosition:1 y:1];

   [builder appendInternational:SDCBInternationalTypeUSA];
   [builder appendCodePage     :SDCBCodePageTypeJapanese];

   if (set) {
       [builder appendUserDefinedCharacter:0 code:0x20 font:(unsigned char *)
"\x000\x000\x032\x000\x049\x000\x049\x07f\x026\x048\x000\x048\x000\x030\x000\x000"];

       [builder appendUserDefinedDbcsCharacter:0 code:0x8140 font:(unsigned char *)
"\x000\x000\x000\x000\x000\x000\x000\x000\x003\x020\x004\x090\x004\x090\x002\x060"

"\x000\x000\x007\x0f0\x004\x080\x004\x080\x003\x000\x000\x000\x000\x000\x000\x000"];
   }
   else {
       [builder appendUserDefinedCharacter:0 code:0x00 font:nil];

       [builder appendUserDefinedDbcsCharacter:0 code:0x0000 font:nil];
   }

   unsigned char pattern[] =
"\x05b\x020\x020\x053\x074\x061\x072\x020\x04d\x069\x063\x072\x06f\x06e\x069\x063\x073\x020\x020\x05d"

"\x05b\x081\x040\x081\x040\x083\x058\x083\x05e\x081\x05b\x090\x0b8\x096\x0a7\x081\x040\x081\x040\x05d";

   [builder appendBytes:pattern length:sizeof(pattern)];
}
```
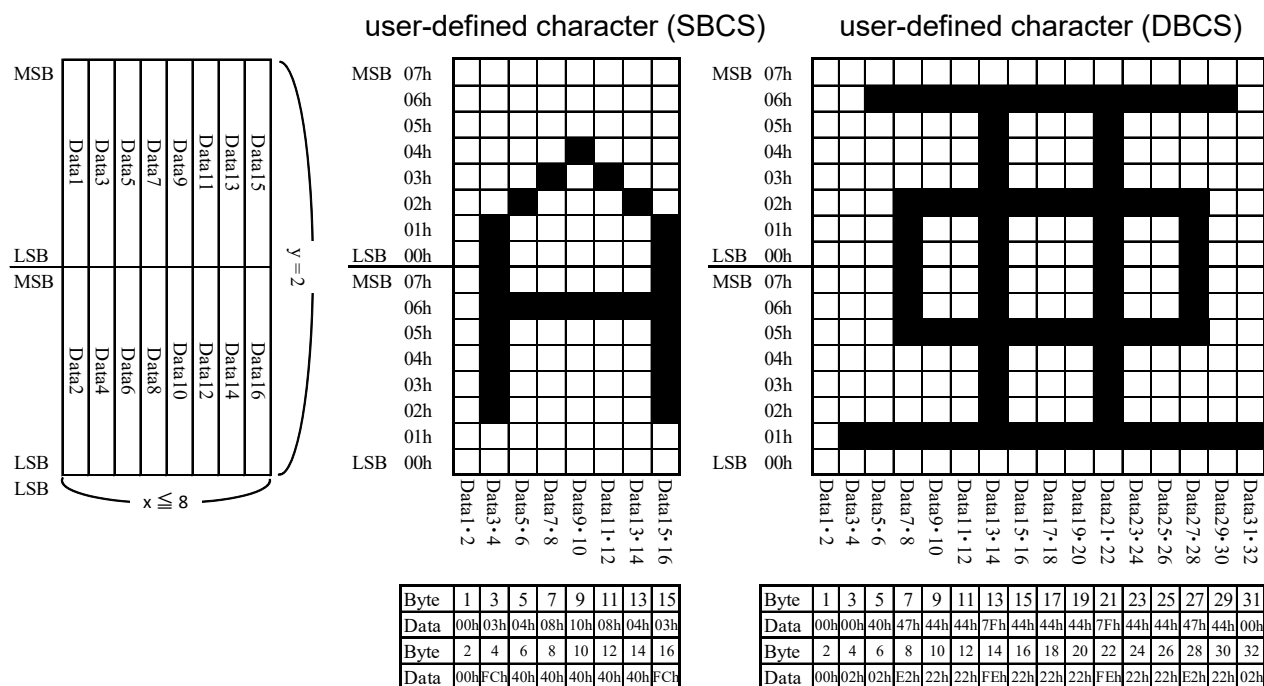
Refer to DisplayFunctions.m.

Font data format of the user-defined characters

user-defined character (SBCS)

Byte	1	3	5	7	9	11	13	15
Data	00h	03h	04h	08h	10h	08h	04h	03h
Byte	2	4	6	8	10	12	14	16
Data	00h	FCh	40h	40h	40h	40h	40h	FCh

user-defined character (DBCS)

Byte	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31
Data	00h	00h	40h	47h	44h	44h	7Fh	44h	44h	44h	7Fh	44h	44h	47h	44h	00h
Byte	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32
Data	00h	02h	02h	E2h	22h	22h	FEh	22h	22h	22h	FEh	22h	22h	E2h	22h	02h

11.19 commands Property

Generated or added commands.

Declaration

@property (nonatomic, readonly) NSMutableData *commands;

Value

Description	Object type
Generated or added commands.	NSMutableData *

11.20 passThroughCommands Property

Generated or added commands to which adds a pass-through command.

Declaration

@property (nonatomic, readonly) NSMutableData *passThroughCommands;

Value

Description	Object type
Generated or added commands to which adds a pass-through command.	NSMutableData *

Example

```
ISDCBBuilder *builder = [StarIoExt createDisplayCommandBuilder:StarIoExtDisplayModelSCD222];

...

NSData *commands = [builder.passThroughCommands copy];
```

Refer to DisplayViewController.m / DisplayExtViewController.m.

11.21 SDCBInternationalType Constant

International character constants.

Declaration

```
typedef NS_ENUM(NSInteger, SDCBInternationalType) {
    SDCBInternationalTypeUSA          = 0x00,
    SDCBInternationalTypeFrance       = 0x01,
    SDCBInternationalTypeGermany      = 0x02,
    SDCBInternationalTypeUK           = 0x03,
    SDCBInternationalTypeDenmark      = 0x04,
    SDCBInternationalTypeSweden       = 0x05,
    SDCBInternationalTypeItaly        = 0x06,
    SDCBInternationalTypeSpain        = 0x07,
    SDCBInternationalTypeJapan        = 0x08,
    SDCBInternationalTypeNorway       = 0x09,
    SDCBInternationalTypeDenmark2     = 0x0a,
    SDCBInternationalTypeSpain2       = 0x0b,
    SDCBInternationalTypeLatinAmerica = 0x0c,
    SDCBInternationalTypeKorea        = 0x0d
};
```

Constants

Name	Description
SDCBInternationalTypeUSA	USA
SDCBInternationalTypeFrance	France
SDCBInternationalTypeGermany	Germany
SDCBInternationalTypeUK	UK
SDCBInternationalTypeDenmark	Denmark
SDCBInternationalTypeSweden	Sweden
SDCBInternationalTypeItaly	Italy
SDCBInternationalTypeSpain	Spain
SDCBInternationalTypeJapan	Japan
SDCBInternationalTypeNorway	Norway
SDCBInternationalTypeDenmark2	Denmark ||
SDCBInternationalTypeSpain2	Spain ||
SDCBInternationalTypeLatinAmerica	Latin America
SDCBInternationalTypeKorea	Korea

11.22 SDCBCodePageType Constant

Code Page constants.

Declaration

```
typedef NS_ENUM(NSInteger, SDCBCodePageType) {
    SDCBCodePageTypeCP437            = 0x00,
    SDCBCodePageTypeKatakana         = 0x01,
    SDCBCodePageTypeCP850            = 0x02,
    SDCBCodePageTypeCP860            = 0x03,
    SDCBCodePageTypeCP863            = 0x04,
    SDCBCodePageTypeCP865            = 0x05,
    SDCBCodePageTypeCP1252           = 0x06,
    SDCBCodePageTypeCP866            = 0x07,
    SDCBCodePageTypeCP852            = 0x08,
    SDCBCodePageTypeCP858            = 0x09,
    SDCBCodePageTypeJapanese         = 0x0a,
    SDCBCodePageTypeSimplifiedChinese  = 0x0b,
    SDCBCodePageTypeTraditionalChinese = 0x0c,
    SDCBCodePageTypeHangul           = 0x0d
};
```

Constants

Name	Description
SDCBCodePageTypeCP437	CodePage437 (USA, Std. Europe).
SDCBCodePageTypeKatakana	Katakana.
SDCBCodePageTypeCP850	PC850 (Multilingual)
SDCBCodePageTypeCP860	PC860 (Portuguese)
SDCBCodePageTypeCP863	PC863 (Canadian-French)
SDCBCodePageTypeCP865	PC865 (Norwegian)
SDCBCodePageTypeCP1252	WPC1252
SDCBCodePageTypeCP866	PC866 [Cyrillic #2]
SDCBCodePageTypeCP852	PC852 [Latin 2]
SDCBCodePageTypeCP858	Page 19 [PC858]
SDCBCodePageTypeJapanese	Japanese font (shift JIS)
SDCBCodePageTypeSimplifiedChinese	Simplified Chinese (GB2312)
SDCBCodePageTypeTraditionalChinese	Traditional Chinese (Big5)
SDCBCodePageTypeHangul	Hangul (KSC5601)

11.23 SDCBCursorMode Constant

Cursor Mode constants.

Declaration
typedef NS_ENUM(NSInteger, SDCBCursorMode) {
   SDCBCursorModeOff   = 0x00,
   SDCBCursorModeBlink = 0x01,
   SDCBCursorModeOn   = 0x02
};

Constants

Name	Description
SDCBCursorModeOff	Cursor off
SDCBCursorModeBlink	Cursor blinking
SDCBCursorModeOn	Cursor lighting

11.24 SDCBContrastMode Constant

Contrast mode constants.

Declaration
typedef NS_ENUM(NSInteger, SDCBContrastMode) {
   SDCBContrastModeMinus3 = 0x00,
   SDCBContrastModeMinus2 = 0x01,
   SDCBContrastModeMinus1 = 0x02,
   SDCBContrastModeDefault = 0x03,
   SDCBContrastModePlus1   = 0x04,
   SDCBContrastModePlus2   = 0x05,
   SDCBContrastModePlus3   = 0x06
};

Constants

Name	Description
SDCBContrastModeMinus3	Contrast: -3
SDCBContrastModeMinus2	Contrast: -2
SDCBContrastModeMinus1	Contrast: -1
SDCBContrastModeDefault	Default
SDCBContrastModePlus1	Contrast: +1
SDCBContrastModePlus2	Contrast: +2
SDCBContrastModePlus3	Contrast: +3

12 ISMCBBuilder interface (StarIO_Extension.framework)

This interface provides the function which creates melody speaker control commands.

Method

Name	Description
appendSoundWithSetting	The command for playback of a sound registered in the melody speaker is generated and added to the commands properties.
appendSoundWithSound	A command for playback of the sound data specified by the argument is generated and added to the commands properties.

Property

Name	Description
commands	Generated or added commands. * *readonly*

Constant

Name	Description
SMCBMelodySpeakerSoundStorageArea	Constant that specifies the sound storage area.

12.1 Model : ISMCBBuilder interface

The ISDCBBuilder interface is only for models which support a melody speaker and for FVP10. For the supporting models, refer to [Supported peripherals](#).

12.2 appendSoundWithSetting Method

The command for playback of a sound registered in the melody speaker is generated and added to the commands properties.

Declaration
- (void)appendSoundWithSetting:(nullable SMSoundSetting *)setting error:(nullable NSError **)error;

Parameter

Name	Description	Object type
setting	Settings information for the playback sound (sound storage area, sound number, etc.)	SMSoundSetting
error	Error information when command generation failed	NSError

The properties and parameters which can be set in the setting argument are the following.

StarIoExtMelody SpeakerModel	Property	Description	Parameters that can be set	Default (operation when nothing is specified)
StarIoExtMelody SpeakerModelMCS10	soundStorageArea	Sound storage area	SMCBMelodySpeakerSoundStorageArea1 SMCBMelodySpeakerSoundStorageArea2	Use the MCS10 unit DIP switch settings.
	soundNumber	Sound number	0 - 7	Use the MCS10 unit DIP switch settings.
	volume	Volume	0 – 15 SMSoundSetting.volumeOff SMSoundSetting.volumeMax SMSoundSetting.volumeMin	Use the MCS10 unit DIP switch settings.
StarIoExtMelody SpeakerModelFVP10	soundStorageArea	Sound storage area	SMCBMelodySpeakerSoundStorageArea1 SMCBMelodySpeakerSoundStorageArea2	SMCBMelodySpeakerSoundStorageArea1
	soundNumber	Sound number	1 - 255	1
	count	Count	1 - 65535	1
	delay	Delay time (1 sec units)	0 - 65535	0
	interval	Interval time (1 sec units)	0 - 65535	0

Sound played according to the combination of parameters specified for the soundStorageArea property and soundNumber property are as follows.

StarIoExtMelodySpeakerModel	soundStorageArea property	soundNumber property	Sound
StarIoExtMelodySpeakerModelMCS10	SMCBMelodySpeakerSoundStorageArea1	0	Sound1
		1	Sound2
		2	Sound3
		3	Sound4
		4	Sound5
		5	Sound6
		6	Sound7
		7	Sound8
	SMCBMelodySpeakerSoundStorageArea2	0	Sound9
		1	Sound10
		2	Sound11
		3	Sound12
		4	Sound13
		5	Sound14
		6	Sound15
		7	Sound16
StarIoExtMelodySpeakerModelFVP10	SMCBMelodySpeakerSoundStorageArea1	1 - 255	Sound stored in the specified sound number of the default area
	SMCBMelodySpeakerSoundStorageArea2	1 - 255	Sound stored in the specified sound number of the user area

Return value

Description	Object type
-	-

Error code

Value	Description
SMResultCodeInvalidArgument	One of the following 1. The property that can not be set on the specified model is set 2. Only one of the soundStorageArea property and the soundNumber property is set
StarIoExtErrorArgumentOutOfRange	Parameter is out of range

Example

```
+ (NSData *)createPlayingRegisteredSound:(StarIoExtMelodySpeakerModel)model
                specifySound:(BOOL)specifySound
              soundStorageArea:(SMCBSoundStorageArea)soundStorageArea
               soundNumber:(NSInteger)soundNumber
              specifyVolume:(BOOL)specifyVolume
                  volume:(NSInteger)volume
                   error:(NSError * _Nullable * _Nullable) error {
    ISMCBBuilder *builder = [StarIoExt createMelodySpeakerCommandBuilder:model];

    SMSoundSetting *setting = [SMSoundSetting new];

    if (specifySound == YES) {
        [setting setSoundStorageArea:soundStorageArea];
        [setting setSoundNumber:soundNumber];
    }

    if (specifyVolume == YES) {
        [setting setVolume:volume];
    }

    [builder appendSoundWithSetting:setting error:error];

    if (*error != nil) {
        return nil;
    }

    return [builder.commands copy];
}
```

Refer to MelodySpeakerFunctions.m

12.3 appendSoundWithSound Method

A command for playback of the sound data specified by the argument is generated and added to the commands properties.

Declaration
- (void)appendSoundWithSound:(nonnull NSData *)data (nullable SMSoundSetting *)setting error:(nullable NSError **)error;

Parameter

Name	Description	Object type
data	Sound data For the supported sound formats, refer to the table of supported sound formats.	NSData
setting	Settings information for the playback sound (sound storage area, sound number, etc.)	SMSoundSetting
error	Error information when command generation failed	NSError **

Table of supported sound formats

StarIoExtMelodySpeakerModel	Sound format
StarIoExtMelodySpeakerModelMCS10	Sound file format: WAV WAV file format: Linear PCM Sampling rate: 12800 Hz Bit depth: 16 bits or 8 bits Channel: Monaural Playback time: Max. 5 sec when bitdepth is 16 　　　　　　　　Max. 10 sec when bitdepth is 8
StarIoExtMelodySpeakerModelFVP10	Not supported

1. It is possible to convert a sound file format to a format in the table of supported sound formats by using the Start Sound Converter application that was provided with this package. For details, refer to "FileFormatList_en.htm" in "Others/StarSoundConverter/(Mac or Windows)/Documents" folder.

The properties and parameters which can be set in the setting argument are the following.

StarIoExtMelody SpeakerModel	Property	Description	Parameters that can be set	Default (operation when nothing is specified)
StarIoExtMelody SpeakerModelMCS10	volume	Volume	0 – 15 SMSoundSetting.volumeOff SMSoundSetting.volumeMax SMSoundSetting.volumeMin	Use the MCS10 unit DIP switch settings.
StarIoExtMelody SpeakerModelFVP10	Not supported			

Return value

Description	Object type
-	-

Error code

Value	Description
StarIoExtErrorFormatError	Data other than a supported sound format is input
SMResultCodeInvalidArgument	The property that can not be set on the specified model is set
StarIoExtErrorArgumentOutOfRange	Parameter is out of range
StarIoExtErrorInvalidOperation	StarIoExtMelodySpeakerModelFVP10 is specified

Example

```
+ (NSData *)createPlayingReceivedData:(StarIoExtMelodySpeakerModel)model
                filePath:(NSString *)filePath
            specifyVolume:(BOOL)specifyVolume
                  volume:(NSInteger)volume
                   error:(NSError * _Nullable * _Nullable)error {
    ISMCBBuilder *builder = [StarIoExt createMelodySpeakerCommandBuilder:model];

    SMSoundSetting *setting = [SMSoundSetting new];

    NSURL *fileUrl = [[NSURL alloc] initWithString:filePath];

    NSData *fileData = [[NSData alloc] initWithContentsOfURL:fileUrl];

    if (specifyVolume == YES) {
        [setting setVolume:volume];
    }

    [builder appendSoundWithSound:fileData
                setting:setting
                 error:error];

    if (*error != nil) {
        return nil;
    }

    return [builder.commands copy];
}
```

Refer to MelodySpeakerFunctions.m

12.4 commands Property

Generated or added commands.

Declaration
@property (nonatomic, readonly) NSMutableData *commands;

Value

Description	Object type
Generated or added commands.	NSMutableData

12.5 SMCBMelodySpeakerSoundStorageArea Constant

Constant that specifies the sound storage area

Declaration
typedef NS_ENUM(NSUInteger, SMCBMelodySpeakerSoundStorageArea) {
   SMCBMelodySpeakerSoundStorageArea1,
   SMCBMelodySpeakerSoundStorageArea2
};

The sound storage areas which correspond to the constants are the following.

StarIoExtMelody SpeakerModel	Constant	Sound storage area
StarIoExtMelody SpeakerModelMCS10	SMCBMelodySpeakerSoundStorageArea1	Sound storage area1
	SMCBMelodySpeakerSoundStorageArea2	Sound storage area2
StarIoExtMelody SpeakerModelFVP10	SMCBMelodySpeakerSoundStorageArea1	Default area
	SMCBMelodySpeakerSoundStorageArea2	User area

Refer to MelodySpeakerFunctions.m

13 SMSoundSetting Class (StarIO_Extension.framework)

This class defines the sound settings.

Property

Name	Description
sounStorageArea	Sound storage area
soundNumber	Sound number
volume	Volume
count	Playback count
delay	Delay time (1 sec units)
interval	Interval time (1 sec units)

Constant

Name	Description
SMSoundSettingVolume	Volume constants.

13.1 soundStorageArea Property

This property specifies the sound storage area.

Declaration
@property (nonatomic) SMCBMelodySpeakerSoundStorageArea soundStorageArea;

Value

Description	Object type
Sound storage area	SMCBMelodySpeakerSoundStorage Area

13.2 soundNumber Property

This property specifies the sound storage number.

Declaration
@property (nonatomic) NSInteger soundNumber;

Value

Description	Object type
Sound number	NSInteger

13.3 volume Property

This property specifies the sound volume.

Declaration
@property (nonatomic) NSInteger volume;

Value

Description	Object type
Volume	NSInteger

13.4 count Property

This property specifies the number of times to play the sound.

Declaration
@property (nonatomic) NSInteger count;

Value

Description	Object type
Number of times to play the sound	NSInteger

13.5 delay Property

This property specifies the delay time.

Declaration

@property (nonatomic) NSInteger delay;

Value

Description	Object type
Delay time (1 sec units)	NSInteger

13.6 interval Property

This property specifies the interval time.

Declaration

@property (nonatomic) NSInteger interval;

Value

Description	Object type
Interval time (1 sec units)	NSInteger

13.7 SMSoundSettingVolume Constant

Volume constants.

Declaration

typedef NS_ENUM(NSInteger, SMSoundSettingVolume) {
    SMSoundSettingVolumeOff = -2,
    SMSoundSettingVolumeMin = -3,
    SMSoundSettingVolumeMax = -4
};

Constants

Name	Description
volumeOff	This constant indicates volume OFF.
volumeMin	This constant indicates volume is minimum.
volumeMax	This constant indicates volume is maximum.

The parameters which are set when used in the SMSoundSetting class properties are the following.

StarIoExtMelody SpeakerModel	Constant	Poperty	Parameter
StarIoExtMelody SpeakerModelMCS10	SMSoundSettingVolumeOff	volume	0
	SMSoundSettingVolumeMin	volume	1
	SMSoundSettingVolumeMax	volume	15
StarIoExtMelody SpeakerModelFVP10	SMSoundSettingVolumeOff	Not supported	
	SMSoundSettingVolumeMin	Not supported	
	SMSoundSettingVolumeMax	Not supported	

14 ISCPParser interface (StarIO_Extension.framework)

An interface to analyze the command response for the peripheral (barcode reader / customer display) control.

Method

Name	Description
createSendCommands	Generates a command to receive the response from the peripheral (barcode reader / customer display).

Property

Name	Description
completionHandler	Analyzes the response of the command (command generated with the createSendCommands) for the peripheral (barcode reader / customer display) control.

Constant

Name	Description
StarIoExtParserCompletionResult	Parse result constants.

14.1 Model : ISCPParser interface

The ISCPParser interface is only for models which support the peripheral (barcode reader / customer display). For the supporting models, refer to Supported peripherals.

14.2 createSendCommands Method

Generates a command to receive the response from the peripheral (barcode reader / customer display).

Declaration
- (NSData *)createSendCommands;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
Generated command.	NSData *

Example

```
+ (BOOL)parseDoNotCheckCondition:(ISCPParser *)parser
                port:(SMPort *)port
       completionHandler:(SendCompletionHandler)completionHandler {
  CommResult result = CommResultErrorOpenPort;
  NSInteger  code   = SMStarIOResultCodeFailedError;

  NSData *commands = [parser createSendCommands];

  while (YES) {
    NSError *error = nil;

    ...

    StarPrinterStatus_2 printerStatus;

    ...

    NSDate *startDate = [NSDate date];

    uint32_t total = 0;

    while (total < (uint32_t) commands.length) {
      uint32_t written = [port writePort:(unsigned char *) commands.bytes :total :(uint32_t) commands.length -
total :&error];

      ...
    }

    ...

    startDate = [NSDate date];     // Restart

    NSMutableData *receivedData = [NSMutableData data];

    while (YES) {
      uint8_t buffer[1024 + 8] = {0};

      …

      uint32_t readLength = [port readPort:buffer :0 :1024 :&error];

      ...

      [receivedData appendBytes:buffer length:readLength];

      int recvDataLength = (int) receivedData.length;

      uint8_t *recvDataBytes = (uint8_t *) receivedData.bytes;

      if (parser.completionHandler(recvDataBytes, &recvDataLength) ==
StarIoExtParserCompletionResultSuccess) {
        result = CommResultSuccess;
        code   = SMStarIOResultCodeSuccess;

        break;
      }
    }

    break;
  }

  ...
}
```

Refer to Communication.m.

14.3 completionHandler Property

Analyzes the response of the command (command generated with the createSendCommands) for the peripheral (barcode reader / customer display) control.

Declaration
typedef StarIoExtParserCompletionResult (^StarIoExtParserCompletionHandler)(uint8_t *buffer, int *length);

@property (nonatomic, copy) StarIoExtParserCompletionHandler completionHandler;

Value

Description	Object type
Parse handler	StarIoExtParserCompletionHandler

Example
Refer to the example of the createSendCommands method.

14.4 StarIoExtParserCompletionResult Constant

Parse result constants.

Declaration
typedef NS_ENUM(NSInteger, StarIoExtParserCompletionResult) {
    StarIoExtParserCompletionResultInvalid = 0,
    StarIoExtParserCompletionResultSuccess,
    StarIoExtParserCompletionResultFailure
};

Constants

Name	Description
StarIoExtParserCompletionResultInvalid	Parse invalid.
StarIoExtParserCompletionResultSuccess	Parse success.
StarIoExtParserCompletionResultFailure	Parse failure.

15 ISCPConnectParser interface (StarIO_Extension.framework)

An interface to provide functions to get the connection/disconnection status of the peripherals (barcode reader / customer display / melody speaker).

Declaration
@interface ISCPConnectParser : ISCPParser

Method

Name	Description
connect	Gets the connection/disconnection status of the peripherals (barcode reader / customer display / melody speaker).

15.1 Model : ISCPConnectParser interface

The ISCPConnectParser interface is only for models which support the peripheral (barcode reader / customer display / melody speaker). For the supporting models, refer to Supported peripherals.

15.2 connect Method

Gets the connection/disconnection status of the peripherals (barcode reader / customer display / melody speaker).
Executes after the completionHandler property returns StarIoExtParserCompletionResultSuccess.

Declaration
- (BOOL)connect;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
The connection/disconnection status of the peripheral.	BOOL

Example
```
ISCPConnectParser *parser = [StarIoExt createDisplayConnectParser:StarIoExtDisplayModelSCD222];

[Communication parseDoNotCheckCondition:parser port:port completionHandler:^(BOOL result, NSString *title, NSString *message) {
   if (result == YES) {
      if (parser.connect == YES) {
         UIAlertView *alertView = [[UIAlertView alloc] initWithTitle:@"Check Status" message:@"Display Connect." delegate:nil cancelButtonTitle:@"OK" otherButtonTitles:nil];

         [alertView show];
      }
      else {
         UIAlertView *alertView = [[UIAlertView alloc] initWithTitle:@"Check Status" message:@"Display Disconnect." delegate:nil cancelButtonTitle:@"OK" otherButtonTitles:nil];

         [alertView show];
      }
   }
   else {
//    UIAlertView *alertView = [[UIAlertView alloc] initWithTitle:@"Failure" message:@"Display Impossible." delegate:nil cancelButtonTitle:@"OK" otherButtonTitles:nil];
      UIAlertView *alertView = [[UIAlertView alloc] initWithTitle:@"Failure" message:@"Printer Impossible." delegate:nil cancelButtonTitle:@"OK" otherButtonTitles:nil];

      [alertView show];
   }
}];
```

Refer to DisplayViewController.m / DisplayExtViewController.m / MelodySpeakerViewController.m.

16 StarPRNT iOS SDK Sample

16.1 Communication

An example of print data transmission of StarIO iOS SDK equivalence and an example of print data transmission with a StarIoExtManager object.

-StarIoExtManager-
When StarIoExtManager is used, a device is always connected.
Therefore, when a device is shared with another application or another terminal, or transition to another application or a sleep state is assumed, consideration of the connection state with the device is required for implementation.

Example (Like a StarIO iOS SDK)

```objc
+ (BOOL)sendCommands:(NSData *)commands portName:(NSString *)portName portSettings:(NSString
*)portSettings timeout:(NSInteger)timeout completionHandler:(SendCompletionHandler)completionHandler {
    …
    SMPort *port = nil;

    while (YES) {
        NSError *error = nil;

        port = [SMPort getPort:portName :portSettings :(uint32_t) timeout :&error];

        if (port == nil) { break; }

        StarPrinterStatus_2 printerStatus;

        [port beginCheckedBlock:&printerStatus :2 :&error];

        if (error != nil) { break; }

        if (printerStatus.offline == SM_TRUE) {
            break;
        }

        NSDate *startDate = [NSDate date];

        uint32_t total = 0;

        while (total < (uint32_t) commands.length) {
            uint32_t written = [port writePort:(unsigned char *) commands.bytes :total :(uint32_t) commands.length -
total :&error];

            if (error != nil) { break; }

            total += written;

            if ([[NSDate date] timeIntervalSinceDate:startDate] >= 30.0) {    // 30000mS!!!
                break;
            }
        }

        if (total < (uint32_t) commands.length) {
            break;
        }

        port.endCheckedBlockTimeoutMillis = 30000;    // 30000mS!!!

        [port endCheckedBlock:&printerStatus :2 :&error];

        if (error != nil) { break; }

        if (printerStatus.offline == SM_TRUE) {
            break;
        }

        break;
    }

    if (port != nil) {
        [SMPort releasePort:port];

        port = nil;
    }
    ...
}
```

Example (Using StarIoExtManager object)

```
+ (BOOL)sendCommands:(NSData *)commands
              port:(SMPort *)port
  completionHandler:(SendCompletionHandler)completionHandler {
  ...

  while (YES) {
    NSError *error = nil;

    if (port == nil) {
      break;
    }

    StarPrinterStatus_2 printerStatus;

    [port beginCheckedBlock:&printerStatus :2 :&error];

    if (error != nil) {
      break;
    }

    if (printerStatus.offline == SM_TRUE) {
      break;
    }

    NSDate *startDate = [NSDate date];

    uint32_t total = 0;

    while (total < (uint32_t) commands.length) {
      uint32_t written = [port writePort:(unsigned char *) commands.bytes :total :(uint32_t) commands.length -
total :&error];

      if (error != nil) {
        break;
      }

      total += written;

      if ([[NSDate date] timeIntervalSinceDate:startDate] >= 30.0) {    // 30000mS!!!
        break;
      }
    }

    if (total < (uint32_t) commands.length) {
      break;
    }

    port.endCheckedBlockTimeoutMillis = 30000;    // 30000mS!!!

    [port endCheckedBlock:&printerStatus :2 :&error];

    if (error != nil) {
      break;
    }

    if (printerStatus.offline == SM_TRUE) {
      break;
    }

    break;
  }

  ...
}
```

17 StarIoExtManager class included in the StarIO_Extension.framework

Method

Name	Description
initWithType	Initializes the StarIoManager object.
connectAsync	Management start.
connect	Management start.
disconnect	Management stop.

Property

Name	Description
port	SMPort object. * *readonly*
lock	Exclusive access control object of communication by port property. * *readonly*
delegate	Delegate of the StarIoExtManager.
printerStatus	Printer status. * *readonly*
printerPaperStatus	Printer paper status. * *readonly*
printerCoverStatus	Printer cover status. * *readonly*
cashDrawerStatus	Cash drawer status. * readonly
barcodeReaderStatus	Barcode reader status. * *readonly*
cashDrawerOpenActiveHigh	Mode of cash drawer open sensor active.

Constants

Name	Description
StarIoExtManagerType	Manager type constants.
StarIoExtManagerPrinterStatus	Printer status constants.
StarIoExtManagerPrinterPaperStatus	Printer paper status constants.
StarIoExtManagerPrinterCoverStatus	Printer cover status constants.
StarIoExtManagerCashDrawerStatus	Cash drawer status constants.
StarIoExtManagerBarcodeReaderStatus	Barcode reader status constants.

-Auto Power Down function with Bluetooth connection-
Please use SM-S and SM-T series in Auto Power Down function "NO USE" setting (default setting).

17.1 initWithType Method

Initializes the StarIoManager object.

Declaration

- (id)initWithType:(StarIoExtManagerType)type portName:(NSString *)portName
portSettings:(NSString *)portSettings ioTimeoutMillis:(NSUInteger)ioTimeoutMillis;

Parameter

Name	Description	Object type
type	Maneger type. • StarIoExtManagerTypeStandard … Management of the printer and the cash drawer. • StarIoExtManagerTypeWithBarcodeReader … Management of the printer, cash drawer and the barcode reader. • StarIoExtManagerTypeOnlyBarcodeReader … Management of the barcode reader.	StarIoExtManagerType
portName	Printer port name.	NSString
portSettings	Port settings.	NSString
ioTimeoutMillis	Timeout value for internal control and API.	NSUInteger

Return value

Description	Object type
StarIoExtManager object.	StarIoExtManager

Example

```
- (void)viewDidLoad {
    [super viewDidLoad];
    // Do any additional setup after loading the view.

    _starIoExtManager = [[StarIoExtManager alloc] initWithType:StarIoExtManagerTypeStandard
                                portName:[AppDelegate getPortName]
                                portSettings:[AppDelegate getPortSettings]
                                ioTimeoutMillis:10000];                    // 10000mS!!!

    _starIoExtManager.delegate = self;
}
```

Refer to PrinterExtViewController.m, CashDrawerExtViewController.m, BarcodeReaderExtViewController.m and CombinationExtViewController.m.

17.2 connectAsync Method

Starts management.
This method is executed asynchronously, and the connection result is sent to the object that was set in the delegate property. If the connection with the printer was successful, the didConnectPort method is called. If the connection failed, the didFailToConnectPort method is called.

Declaration
- (void)connectAsync;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
-	-

Example

```
- (void)refreshPrinter {
    self.blind = YES;

    [self.starIoExtManager disconnect];

    [self.starIoExtManager connectAsync];
}
```

Refer to PrinterExtWithConnectAsyncViewController.m.

17.3 connect Method

Management start.
This method is executed synchronously.

Declaration
- (BOOL)connect;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
Result. • YES … Success • NO … Failure	BOOL

Example

```objc
- (void)viewWillAppear:(BOOL)animated {
  [super viewWillAppear:animated];

  if (_starIoExtManager.port != nil) {
    [_starIoExtManager disconnect];
  }

  if ([_starIoExtManager connect] == NO) {
    UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Fail to Open Port."
                                 message:@""
                                 delegate:self
                           cancelButtonTitle:@"OK"
                           otherButtonTitles:nil];

    [alert show];
  }
}
```

Refer to PrinterExtViewController.m, CashDrawerExtViewController.m, BarcodeReaderExtViewController.m and CombinationExtViewController.m.

17.4 disconnect Method

Management stop.

Declaration
- (BOOL)disconnect;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
Result. • YES … Success • NO … Failure	BOOL

Example

```
- (void)viewWillDisappear:(BOOL)animated {
   [super viewWillDisappear:animated];

   [_starIoExtManager disconnect];
}
```

Refer to PrinterExtViewController.m, CashDrawerExtViewController.m, BarcodeReaderExtViewController.m and CombinationExtViewController.m.

17.5 port Property

SMPort object. * *readonly*

Declaration
@property (readonly, nonatomic) SMPort *port;

17.6 lock Property

Exclusive access control object of communication by port property. ** readonly*

By using the lock property, an application can acquire a right to use the port property of StarIoExtManager class.

StarIoExtManager class starts monitoring printer status automatically after connect method is successfully complete.The printer status monitoring process uses the port property of StarIoExtManager class to communicate with a printer.If an application executes method of the port property during monitoring, the port property is used by two at the same time and method of the port property cannot work properly.

When an application executes lock() method of the lock property, StarIoExtManager class stops monitoring printer status, and the application regains a right to use the port property from StarIoExtManager class. As a result, the application can execute method of the port property properly.

After an application finishes using method of the port property, execute unlock() method of the lock property.StarIoExtManager class regains a right to use the port property from the application and resumes monitoring printer status.

Declaration
@property (readonly, nonatomic) NSRecursiveLock *lock;

Example
```
- (IBAction)touchUpInsidePrintButton:(id)sender {

    …

    [_starIoExtManager.lock lock];

    [Communication sendCommands:commands port:[_starIoExtManager port]];

    [_starIoExtManager.lock unlock];

    …

}
```

Refer to PrinterExtViewController.m, CashDrawerExtViewController.m and CombinationExtViewController.m.

17.7 delegate Property

Delegate of the StarIoExtManager.

Declaration

@property (weak, nonatomic) id<StarIoExtManagerDelegate> delegate;

Example

```
- (void)viewDidLoad {
    [super viewDidLoad];
    // Do any additional setup after loading the view.

    _starIoExtManager = [[StarIoExtManager alloc] initWithType:StarIoExtManagerTypeOnlyBarcodeReader
                                    portName:[AppDelegate getPortName]
                                    portSettings:[AppDelegate getPortSettings]
                                    ioTimeoutMillis:10000];                     // 10000mS!!!

    _starIoExtManager.delegate = self;
}
```

Refer to PrinterExtViewController.m, CashDrawerExtViewController.m, BarcodeReaderExtViewController.m and CombinationExtViewController.m.

17.8 printerStatus Property

Printer Online status. * *readonly*

Declaration

@property (readonly, nonatomic) StarIoExtManagerPrinterStatus printerStatus;

Value

Description	Object type
Printer status. • StarIoExtManagerPrinterStatusInvalid … Invalid. • StarIoExtManagerPrinterStatusImpossible … Impossible to use printer. • StarIoExtManagerPrinterStatusOnline … Detect online. • StarIoExtManagerPrinterStatusOffline … Detect offline.	StarIoExtManagerPrinterStatus

17.9 printerPaperStatus Property

Printer paper status. * *readonly*

Declaration

@property (readonly, nonatomic) StarIoExtManagerPrinterPaperStatus printerPaperStatus;

Value

Description	Object type
Printer paper status. • StarIoExtManagerPrinterPaperStatusInvalid … Invalid. • StarIoExtManagerPrinterPaperStatusImpossible … Impossible to use Equipment. • StarIoExtManagerPrinterPaperStatusReady … Detect paper ready. • StarIoExtManagerPrinterPaperStatusNearEmpty … Detect paper near end. • StarIoExtManagerPrinterPaperStatusEmpty … Detect paper empty.	StarIoExtManagerPrinterPaperStatus

17.10 printerCoverStatus Property

Printer cover status. * *readonly*

Declaration

@property (readonly, nonatomic) StarIoExtManagerPrinterCoverStatus printerCoverStatus;

Value

Description	Object type
Printer cover status. • StarIoExtManagerPrinterCoverStatusInvalid … Invalid. • StarIoExtManagerPrinterCoverStatusImpossible … Ipossible to use Equipment. • StarIoExtManagerPrinterCoverStatusOpen … Detect cover open. • StarIoExtManagerPrinterCoverStatusClose … Detect cover close.	StarIoExtManagerPrinterCoverStatus

17.11 cashDrawerStatus Property

Cash drawer status. * *readonly*

Declaration

@property (readonly, nonatomic) StarIoExtManagerCashDrawerStatus cashDrawerStatus;

Value

Description	Object type
Cash drawer status. • StarIoExtManagerCashDrawerStatusInvalid … Invalid. • StarIoExtManagerCashDrawerStatusImpossible … Impossible to use Equipment. • StarIoExtManagerCashDrawerStatusOpen … Detect Cash drawer open. • StarIoExtManagerCashDrawerStatusClose … Detect Cash drawer close.	StarIoExtManagerCashDrawerStatus

17.12 barcodeReaderStatus Property

Barcode reader status. *readonly*

Declaration

@property (readonly, nonatomic) StarIoExtManagerBarcodeReaderStatus barcodeReaderStatus;

Value

Description	Object type
Barcode reader status. • StarIoExtManagerBarcodeReaderStatusInvalid … Invalid. • StarIoExtManagerBarcodeReaderStatusImpossible … Impossible to use Equipment. • StarIoExtManagerBarcodeReaderStatusConnect … Detect Barcode reader connection. • StarIoExtManagerBarcodeReaderStatusDisconnect … Detect Barcode reader disconnection.	StarIoExtManagerBarcodeReaderStatus

17.13 cashDrawerOpenActiveHigh Property

Mode of cash drawer open sensor active.

Declaration

@property (nonatomic) BOOL cashDrawerOpenActiveHigh;

Value

Description	Object type
Mode of a cash drawer open sensor active. • YES … Active high. • NO … Active low.	BOOL

17.14 StarIoExtManagerType Constants

Manager type constants.

Declaration

typedef NS_ENUM(NSUInteger, StarIoExtManagerType) {
    StarIoExtManagerTypeStandard = 0,
    StarIoExtManagerTypeWithBarcodeReader,
    StarIoExtManagerTypeOnlyBarcodeReader,
};

Constants

Name	Description
StarIoExtManagerTypeStandard	Management of the printer and the cash drawer.
StarIoExtManagerTypeWithBarcodeReader	Management of the printer, cash drawer and the barcode reader.
StarIoExtManagerTypeOnlyBarcodeReader	Management of the barcode reader.

17.15 StarIoExtManagerPrinterStatus Constants

Printer status constants.

Declaration

typedef NS_ENUM(NSInteger, StarIoExtManagerPrinterStatus) {
    StarIoExtManagerPrinterStatusInvalid = 0,
    StarIoExtManagerPrinterStatusImpossible,
    StarIoExtManagerPrinterStatusOnline,
    StarIoExtManagerPrinterStatusOffline
};

Constants

Name	Description
StarIoExtManagerPrinterStatusInvalid	Invalid.
StarIoExtManagerPrinterStatusImpossible	Impossible to use printer.
StarIoExtManagerPrinterStatusPrinterOnline	Detect Printer online.
StarIoExtManagerPrinterStatusPrinterOffline	Detect Printer offline.

17.16 StarIoExtManagerPrinterPaperStatus Constants

Printer paper status constants.

Declaration
typedef NS_ENUM(NSInteger, StarIoExtManagerPrinterPaperStatus) {
   StarIoExtManagerPrinterPaperStatusInvalid = 0,
   StarIoExtManagerPrinterPaperStatusImpossible,
   StarIoExtManagerPrinterPaperStatusReady,
   StarIoExtManagerPrinterPaperStatusNearEmpty,
   StarIoExtManagerPrinterPaperStatusEmpty
};

Constants

Name	Description
StarIoExtManagerPrinterPaperStatusInvalid	Invalid.
StarIoExtManagerPrinterPaperStatusImpossible	Impossible to use Printer.
StarIoExtManagerPrinterPaperStatusReady	Detect Printer paper ready.
StarIoExtManagerPrinterPaperStatusNearEmpty	Detect Printer paper near end.
StarIoExtManagerPrinterPaperStatusEmpty	Detect Printer paper empty.

17.17 StarIoExtManagerPrinterCoverStatus Constants

Printer cover status constants.

Declaration
typedef NS_ENUM(NSInteger, StarIoExtManagerPrinterCoverStatus) {
   StarIoExtManagerPrinterCoverStatusInvalid = 0,
   StarIoExtManagerPrinterCoverStatusImpossible,
   StarIoExtManagerPrinterCoverStatusOpen,
   StarIoExtManagerPrinterCoverStatusClose
};

Constants

Name	Description
StarIoExtManagerPrinterCoverStatusInvalid	Invalid.
StarIoExtManagerPrinterCoverStatusImpossible	Impossible to use Printer.
StarIoExtManagerPrinterCoverStatusOpen	Detect Printer cover open.
StarIoExtManagerPrinterCoverStatusClose	Detect Printer cover close.

17.18 StarIoExtManagerCashDrawerStatus Constants

Cash drawer status constants.

Declaration
typedef NS_ENUM(NSInteger, StarIoExtManagerCashDrawerStatus) {
    StarIoExtManagerCashDrawerStatusInvalid = 0,
    StarIoExtManagerCashDrawerStatusImpossible,
    StarIoExtManagerCashDrawerStatusOpen,
    StarIoExtManagerCashDrawerStatusClose
};

Constants

Name	Description
StarIoExtManagerCashDrawerStatusInvalid	Invalid.
StarIoExtManagerCashDrawerStatusImpossible	Impossible to use Cash drawer.
StarIoExtManagerCashDrawerStatusOpen	Detect Cash drawer open.
StarIoExtManagerCashDrawerStatusClose	Detect Cash drawer close.

17.19 StarIoExtManagerBarcodeReaderStatus Constants

Barcode reader status constants.

Declaration
typedef NS_ENUM(NSInteger, StarIoExtManagerBarcodeReaderStatus) {
    StarIoExtManagerBarcodeReaderStatusInvalid = 0,
    StarIoExtManagerBarcodeReaderStatusImpossible,
    StarIoExtManagerBarcodeReaderStatusConnect,
    StarIoExtManagerBarcodeReaderStatusDisconnect
};

Constants

Name	Description
StarIoExtManagerBarcodeReaderStatusInvalid	Invalid.
StarIoExtManagerBarcodeReaderStatusImpossible	Impossible to use Barcode reader.
StarIoExtManagerBarcodeReaderStatusConnect	Detect Barcode reader connection.
StarIoExtManagerBarcodeReaderStatusDisconnect	Detect Barcode reader disconnection.

18 StarIoExtManagerDelegate class included in the StarIO_Extension.framework

Method

Name	Description
didConnectPort	Sends notice of connection success. * *optional*
didFailToConnectPort	Sends notice of connection failure. * *optional*
didPrinterImpossible	Tells the delegate the printer impossible. * *optional*
didPrinterOnline	Tells the delegate the printer has come online. * *optional*
didPrinterOffline	Tells the delegate the printer has come offline. * *optional*
didPrinterPaperReady	Tells the delegate the printer has come paper ready. * *optional*
didPrinterPaperNearEmpty	Tells the delegate the printer has come paper near end. * *optional*
didPrinterPaperEmpty	Tells the delegate the printer has come paper empty. * *optional*
didPrinterCoverOpen	Tells the delegate the printer has come cover open. * *optional*
didPrinterCoverClose	Tells the delegate the printer has come cover close. * *optional*
didCashDrawerOpen	Tells the delegate the cash drawer has come open. * *optional*
didCashDrawerClose	Tells the delegate the cash drawer has come close. * *optional*
didBarcodeReaderImpossible	Tells the delegate the barcode reader impossible. * *optional*
didBarcodeReaderConnect	Tells the delegate the barcode reader connected. * *optional*
didBarcodeReaderDisconnect	Tells the delegate the barcode reader disconnected. * *optional*
didBarcodeDataReceive	Tells the delegate the barcode data received. * *optional*
didAccessoryConnectSuccess	Tells the delegate the bluetooth accessory connection succeeded. * *optional*
didAccessoryConnectFailure	Tells the delegate the bluetooth accessory connection failed. * *optional*
didAccessoryDisconnect	Tells the delegate the bluetooth accessory disconnected. * *optional*
didStatusUpdate	Tells the delegate the status updated. * *optional*

18.1 didConnectPort Method

Sends notice of connection success. * *optional*

Declaration

- (void)manager:(nonnull StarIoExtManager *)manager
didConnectPort:(nonnull NSString *)portName;

Parameter

Name	Description	Object type
manager	Call source StarIoExtManager object	StarIoExtManager *
portName	Printer port name	NSString *

Return value

Description	Object type
-	-

Example

```
- (void)manager:(StarIoExtManager *)manager didConnectPort:(NSString *)portName {
    dispatch_async(GlobalQueueManager.sharedManager.serialQueue, ^{
        dispatch_async(dispatch_get_main_queue(), ^{
            if ((self.didAppear == NO) && (self.starIoExtManager.port != nil)) {
                [self.printButton sendActionsForControlEvents:UIControlEventTouchUpInside];
            } else {
                self.blind = NO;
            }

            self.didAppear = YES;
        });
    });
}
```

Refer to PrinterExtWithConnectAsyncViewController.m.

18.2 didFailToConnectPort Method

Sends notice of connection failure. * *optional*

Declaration

- (void)manager:(nonnull StarIoExtManager *)manager
didFailToConnectPort:(nonnull NSString *)portName
error:(nullable NSError *)error;

Parameter

Name	Description	Object type
manager	Call source StarIoExtManager object	StarIoExtManager *
portName	Printer port name	NSString *
error	Connection failure error information An error code is set in the code property.	NSError *

Error code

Value	Description
SMResultCodeFailed	Some kind of error occurred.
SMResultCodeInUse	Connection was refused by the printer (another host is connected or other reason).
SMResultCodeInProcess	Connection process is already in progress.

Return value

Description	Object type
-	-

Example

```objc
- (void)manager:(StarIoExtManager *)manager didFailToConnectPort:(NSString *)portName error:(NSError *)error
{
    if (error != nil) {
        [self showSimpleAlertWithTitle:@"Fail to Open Port."
                    message:nil
                buttonTitle:@"OK"
                buttonStyle:UIAlertActionStyleCancel
                 completion:^(UIAlertController *alertController) {
                    self.commentLabel.text = [NSString stringWithFormat:@"%@\n"
                                    "\n"
                                    "Check the device. (Power and Bluetooth pairing)\n"
                                    "Then touch up the Refresh button.\n", error.localizedDescription];

                    self.commentLabel.textColor = UIColor.redColor;

                    [self beginAnimationCommantLabel];

                    self.blind = NO;
                }];
    }

    self.didAppear = YES;
}
```

Refer to PrinterExtWithConnectAsyncViewController.m.

18.3 didPrinterImpossible Method

Tells the delegate the printer impossible. ** optional*

Declaration

- (void)didPrinterImpossible;

- (void)didPrinterImpossible:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didPrinterImpossible:(StarIoExtManager *)manager {
    _commentLabel.text = @"Printer Impossible.";

    _commentLabel.textColor = [UIColor redColor];
}
```

Refer to PrinterExtViewController.m, CashDrawerExtViewController.m and CombinationExtViewController.m.

18.4 didPrinterOnline Method

Tells the delegate the printer has come online. * *optional*

Declaration
- (void)didPrinterOnline;

- (void)didPrinterOnline:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didPrinterOnline:(StarIoExtManager *)manager {
    _commentLabel.text = @"Printer Online.";

    _commentLabel.textColor = [UIColor blueColor];
}
```

Refer to PrinterExtViewController.m and CombinationExtViewController.m.

18.5 didPrinterOffline Method

Tells the delegate the printer has come offline. * *optional*

Declaration
- (void)didPrinterOffline;

- (void)didPrinterOffline:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didPrinterOffline:(StarIoExtManager *)manager {
    _commentLabel.text = @"Printer Offline.";

    _commentLabel.textColor = [UIColor redColor];
}
```

Refer to PrinterExtViewController.m and CombinationExtViewController.m.

18.6 didPrinterPaperReady Method

Tells the delegate the printer has come paper ready. * *optional*

Declaration

- (void)didPrinterPaperReady;

- (void)didPrinterPaperReady:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didPrinterPaperReady:(StarIoExtManager *)manager {
    _commentLabel.text = @"Printer Paper Ready.";

    _commentLabel.textColor = [UIColor blueColor];
}
```

Refer to PrinterExtViewController.m and CombinationExtViewController.m.

18.7 didPrinterPaperNearEmpty Method

Tells the delegate the printer has come paper near end. * *optional*

Declaration

- (void)didPrinterPaperNearEmpty;

- (void)didPrinterPaperNearEmpty:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didPrinterPaperNearEmpty:(StarIoExtManager *)manager {
    _commentLabel.text = @"Printer Paper Near Empty.";

    _commentLabel.textColor = [UIColor orangeColor];
}
```

Refer to PrinterExtViewController.m and CombinationExtViewController.m.

18.8 didPrinterPaperEmpty Method

Tells the delegate the printer has come paper empty. * *optional*

Declaration

- (void)didPrinterPaperEmpty;

- (void)didPrinterPaperEmpty:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didPrinterPaperEmpty:(StarIoExtManager *)manager {
    _commentLabel.text = @"Printer Paper Empty.";

    _commentLabel.textColor = [UIColor redColor];
}
```

Refer to PrinterExtViewController.m and CombinationExtViewController.m.

18.9 didPrinterCoverOpen Method

Tells the delegate the printer has come cover open. * *optional*

Declaration

- (void)didPrinterCoverOpen;

- (void)didPrinterCoverOpen:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didPrinterCoverOpen:(StarIoExtManager *)manager {
    _commentLabel.text = @"Printer Cover Open.";

    _commentLabel.textColor = [UIColor redColor];
}
```

Refer to PrinterExtViewController.m and CombinationExtViewController.m.

18.10 didPrinterCoverClose Method

Tells the delegate the printer has come cover close. * *optional*

Declaration

- (void)didPrinterCoverClose;

- (void)didPrinterCoverClose:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didPrinterCoverClose:(StarIoExtManager *)manager {
    _commentLabel.text = @"Printer Cover Close.";

    _commentLabel.textColor = [UIColor blueColor];
}
```

Refer to PrinterExtViewController.m and CombinationExtViewController.m.

18.11 didCashDrawerOpen Method

Tells the delegate the cash drawer has come open. * *optional*

Declaration

- (void)didCashDrawerOpen;

- (void)didCashDrawerOpen:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didCashDrawerOpen:(StarIoExtManager *)manager {
    _commentLabel.text = @"Cash Drawer Open.";

//  _commentLabel.textColor = [UIColor redColor];
    _commentLabel.textColor = [UIColor magentaColor];
}
```

Refer to CashDrawerExtViewController.m and CombinationExtViewController.m.

18.12 didCashDrawerClose Method

Tells the delegate the cash drawer has come close. *optional*

Declaration

- (void)didCashDrawerClose;

- (void)didCashDrawerClose:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didCashDrawerClose:(StarIoExtManager *)manager {
    _commentLabel.text = @"Cash Drawer Close.";

    _commentLabel.textColor = [UIColor blueColor];
}
```

Refer to CashDrawerExtViewController.m and CombinationExtViewController.m.

18.13 didBarcodeReaderImpossible Method

Tells the delegate the barcode reader impossible. *optional*

Declaration

- (void)didBarcodeReaderImpossible;

- (void)didBarcodeReaderImpossible:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didBarcodeReaderImpossible:(StarIoExtManager *)manager {
    _commentLabel.text = @"Barcode Reader Impossible.";

    _commentLabel.textColor = [UIColor redColor];
}
```

Refer to BarcodeReaderExtViewController.m and CombinationExtViewController.m.

18.14 didBarcodeReaderConnect Method

Tells the delegate the barcode reader connected. * *optional*

Declaration

- (void)didBarcodeReaderConnect;

- (void)didBarcodeReaderConnect:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didBarcodeReaderConnect:(StarIoExtManager *)manager {
    _commentLabel.text = @"Barcode Reader Connect.";

    _commentLabel.textColor = [UIColor blueColor];
}
```

Refer to BarcodeReaderExtViewController.m and CombinationExtViewController.m.

18.15 didBarcodeReaderDisconnect Method

Tells the delegate the barcode reader disconnected. * *optional*

Declaration

- (void)didBarcodeReaderDisconnect;

- (void)didBarcodeReaderDisconnect:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didBarcodeReaderDisconnect:(StarIoExtManager *)manager {
    _commentLabel.text = @"Barcode Reader Disconnect.";

    _commentLabel.textColor = [UIColor redColor];
}
```

Refer to BarcodeReaderExtViewController.m and CombinationExtViewController.m.

18.16 didBarcodeDataReceive Method

Tells the delegate the barcode data received. * *optional*

Declaration

- (void)didBarcodeDataReceive:(NSData *)data;

- (void)didBarcodeDataReceive:(StarIoExtManager *)manager data:(NSData *)data;

Parameter

Name	Description	Object type
data	Received barcode data.	NSData
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didBarcodeDataReceive:(StarIoExtManager *)manager data:(NSData *)data {
    NSMutableString *text = [NSMutableString stringWithString:@""];

    const uint8_t *p = [data bytes];

    for (int i = 0; i < data.length; i++) {
        uint8_t ch = *(p + i);

        if(ch >= 0x20 && ch <= 0x7f) {
            [text appendFormat:@"%c", (char) ch];
        }
        else if (ch == 0x0d) {

            …

            text = [NSMutableString stringWithString:@""];
        }
    }
}
```

Refer to BarcodeReaderExtViewController.m and CombinationExtViewController.m.

18.17 didAccessoryConnectSuccess Method

Tells the delegate the bluetooth accessory connection succeeded. * *optional*

Declaration

- (void)didAccessoryConnectSuccess;

- (void)didAccessoryConnectSuccess:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didAccessoryConnectSuccess:(StarIoExtManager *)manager {
    _commentLabel.text = @"Accessory Connect Success.";

    _commentLabel.textColor = [UIColor blueColor];
}
```

Refer to PrinterExtViewController.m, CashDrawerExtViewController.m, BarcodeReaderExtViewController.m and CombinationExtViewController.m.

18.18 didAccessoryConnectFailure Method

Tells the delegate the bluetooth accessory connection failed. * *optional*

Declaration

- (void)didAccessoryConnectFailure;

- (void)didAccessoryConnectFailure:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
Manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didAccessoryConnectFailure:(StarIoExtManager *)manager {
    _commentLabel.text = @"Accessory Connect Failure.";

    _commentLabel.textColor = [UIColor redColor];
}
```

Refer to PrinterExtViewController.m, CashDrawerExtViewController.m, BarcodeReaderExtViewController.m and CombinationExtViewController.m.

18.19 didAccessoryDisconnect Method

Tells the delegate the bluetooth accessory disconnected. * *optional*

Declaration

- (void)didAccessoryDisconnect;

- (void)didAccessoryDisconnect:(StarIoExtManager *)manager;

Parameter

Name	Description	Object type
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didAccessoryDisconnect:(StarIoExtManager *)manager {
    _commentLabel.text = @"Accessory Disconnect.";

    _commentLabel.textColor = [UIColor redColor];
}
```

Refer to PrinterExtViewController.m, CashDrawerExtViewController.m, BarcodeReaderExtViewController.m and CombinationExtViewController.m.

18.20 didStatusUpdate Method

Tells the delegate the status updated. * *optional*

Declaration

- (void)didStatusUpdate:(NSString *)status;

- (void)didStatusUpdate:(StarIoExtManager *)manager status:(NSString *)status;

Parameter

Name	Description	Object type
status	Updated status. Hexdecimal digit sequence of 3rd to 6th bytes in Automatic Status (*) (ex. "28000000") -Status update timing- • When the status changes. • Even no status change, every 5 min. * Please refer to STAR Line Mode and StarPRNT command manual for details of Automatic Status.	NSString
manager	A StarIoExtManager object informing the delegate.	StarIoExtManager

Return value

Description	Object type
-	-

Example

```
- (void)didStatusUpdate:(StarIoExtManager *)manager status:(NSString *)status {
    _commentLabel.text = status;

    _commentLabel.textColor = [UIColor greenColor];
}
```

Refer to AllReceiptsExtViewController.m.

19 SMBluetoothManagerFactory class (StarIO_Extension.framework)

SMBluetoothManagerFactory class can get the appropriate SMBluetoothManager object for the printer emulation.

Method

Name	Description
getManager	Gets the object of the SMBluetoothManager class for the emulation to be passed in the parameter.

19.1 getManager Method

Gets the object of the SMBluetoothManager class for the emulation to be passed in the parameter.

Declaration

- (SMBluetoothManager *)getManager:(NSString *)portName
emulation:(StarIoExtEmulation)emulation;

Parameter

Name	Description	Object type
portName	Printer port name	NSString *
emulation	emulation	StarIoExtEmulation

Return value

Description	Object type
SMBluetoothManager object	SMBluetoothManager *

Refer to the printing process flow using a SMBluetoothManager *about the procedure of change the Bluetooth Setting.*
Constant of "StarIoExtEmulation" for each model, refer to StarIoExtEmulation Constants.

20 SMCloudServices class included in the SMCloudServices.framework

Method

Name	Description
showRegistrationView	It displays a view of the Star Micronics Cloud registration.
isRegistered	Checks whether the application is registered with the Star Micronics Cloud.

20.1 showRegistrationView Method

It displays a view of the Star Micronics Cloud registration.
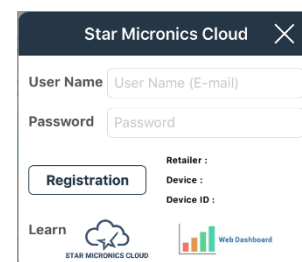
Declaration

+ (void)showRegistrationView:(void (^)(BOOL isRegistered))completion;

Parameter

Name	Description	Object type
completion	When the View is closed, check application registration status as following parameter. -isRegistered- • YES … Registered. • NO … Unregistered.	void (^)(BOOL isRegistered)

Return value

Description	Object type
-	-

Example

```
- (void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)indexPath {

    …

    [SMCloudServices showRegistrationView:^(BOOL isRegistration) {
        [_tableView reloadData];
    }];

    …

}
```

Refer to AllReceiptsViewController.m.

20.2 isRegistered Method

Checks whether the application is registered with the Star Micronics Cloud.

Declaration

+ (BOOL)isRegistered;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
Result. • YES … Registered. • NO … Unregistered.	BOOL

Example

```
- (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath {

    …

    BOOL userInteractionEnabled = YES;

    if ([SMCloudServices isRegistered] == NO) {
        userInteractionEnabled = NO;
    }

    …

    return cell;
}
```

Refer to AllReceiptsViewController.m.

21 SMCSAllReceipts class included in the SMCloudServices.framework

Method

Name	Description
uploadBitmap	Uploads bitmap to the Star Micronics Service.
uploadData	Uploads data to the Star Micronics Service.
updateStatus	Updates the device status information on the Star Micronics Cloud.
generateAllReceipts	Generates the print data for the AllReceipts™ use.

21.1 Model : SMCSAllReceipts class Method

Supported Method for each models.

Function	Method	mC-Print2	mC-Print3	mPOP	FVP10	TSP100	TSP650II	TSP700II	TSP800II	SM-S210i	SM-S220i	SM-S230i	SM-T300i	SM-T400i	BSC10	SM-S210i StarPRNT	SM-S220i StarPRNT	SM-S230i StarPRNT	SM-T300i StarPRNT	SM-T400i StarPRNT	SM-L200	SM-L300	SP700
Upload	uploadBitmap	✔	✔	✔	✔	✔	✔	✔	✔	-	-	-	-	-	✔	✔	✔	✔	✔	✔	✔	✔	-
	uploadData	✔	✔	✔	✔	-	✔	✔	✔	-	-	-	-	-	✔	✔	✔	✔	✔	✔	✔	✔	-
Update	updateStatus	✔	✔	✔	✔	✔	✔	✔	✔	-	-	-	-	-	✔	✔	✔	✔	✔	✔	✔	✔	-
Generate	generateAllReceipts	✔	✔	✔	✔	✔	✔	✔	✔	-	-	-	-	-	✔	✔	✔	✔	✔	✔	✔	✔	-

- : Not guarantee.

21.2 uploadBitmap Method

Uploads bitmap (UIImage object) to the Star Micronics Service.

Declaration

+ (NSString *)uploadBitmap:(UIImage *)image completion:(void (^)(NSInteger statusCode, NSError *error))completion;

Parameter

Name	Description	Object type
image	Bitmap to upload.	UIImage
completion	When the upload is completed, check upload result as following status. -statusCode- • HTTP status code. -error- • Error information when fails to upload data.	void (^)(NSInteger statusCode, NSError *error)

Return value

Description	Object type
Uploaded URL.	NSString

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation image:(UIImage *)image {
    NSString *urlString = [SMCSAllReceipts uploadBitmap:image completion:nil];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendBitmap:image diffusion:NO];

    NSData *data = [SMCSAllReceipts generateAllReceipts:urlString emulation:emulation info:YES qrCode:YES];

    [builder appendRawData:data];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to AllReceiptsFunctions.m.

21.3 uploadData Method

Uploads data (NSData object) to the Star Micronics Service.

Declaration

+ (NSString *)uploadData:(NSData *)data emulation:(StarIoExtEmulation)emulation characterCode:(StarIoExtCharacterCode)characterCode width:(NSInteger)width completion:(void (^)(NSInteger statusCode, NSError *error))completion;

Parameter

Name	Description	Object type
data	Data to upload.	NSData
emulation	Emulation type. • StarIoExtEmulationStarPRNT … StarPRNT emulation. • StarIoExtEmulationStarLine … STAR Line Mode emulation. • ~~StarIoExtEmulationStarGraphic … STAR Graphic Mode emulation.~~ • StarIoExtEmulationEscPos … ESC/POS emulation. • ~~StarIoExtEmulationEscPosMobile … ESC/POS Mobile emulation.~~ • ~~StarIoExtEmulationStarDotImpact … STAR Dot Impact emulation.~~	StarIoExtEmulation
characterCode	Character code type constants. • StarIoExtCharacterCodeStandard … Standard character code. • StarIoExtCharacterCodeJapanese … Japanese character code. • ~~StarIoExtCharacterCodeSimplifiedChinese … Simplified chinese character code.~~ • ~~StarIoExtCharacterCodeTraditionalChinese … Traditional chinese character code.~~	StarIoExtCharacterCode
width	Printable width. (Units : Dots)	NSInteger
completion	When the upload is completed, check upload result as following status. -statusCode- • HTTP status code. -error- • Error information when fails to upload data.	void (^)(NSInteger statusCode, NSError *error)

Return value

Description	Object type
Uploaded URL.	NSString

Constant of "StarIoExtEmulation" for each model, refer to StarIoExtEmulation Constants.

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation characterCode:(StarIoExtCharacterCode)characterCode
data:(NSData *)data width:(NSInteger)width {
    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:data];

    [builder endDocument];

    NSData *receiptData = [builder.commands copy];

    NSString *urlString = [SMCSAllReceipts uploadData:receiptData emulation:emulation
characterCode:characterCode width:width completion:nil];

    builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:data];

    NSData *allReceiptsData = [SMCSAllReceipts generateAllReceipts:urlString emulation:emulation info:YES
qrCode:YES];

    [builder appendRawData:allReceiptsData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to AllReceiptsFunctions.m.

21.4 updateStatus Method

Updates the device status information on the Star Micronics Cloud.

Declaration

+ (void)updateStatus:(NSString *)status completion:(void (^)(NSInteger statusCode, NSError *error))completion;

Parameter

Name	Description	Object type
status	Updated status. Hexdecimal digit sequence of 3rd to 6th bytes in Automatic Status (*) (ex. "28000000") * Please refer to STAR Line Mode and StarPRNT command manual for details of Automatic Status.	NSString
completion	When the update is completed, check update result as following parameter. -statusCode- • HTTP status code. -error- • If the request fails, the error parameter contains information about the failure.	void (^)(NSInteger statusCode, NSError *error)

Return value

Description	Object type
-	-

Example

```
- (void)didStatusUpdate:(StarIoExtManager *)manager status:(NSString *)status {
    [SMCSAllReceipts updateStatus:status completion:^(NSInteger statusCode, NSError *error) {

        ...

    }];
}
```

Refer to AllReceiptsExtViewController.m.

21.5 generateAllReceipts Method

Generates the print data for the AllReceipts™ use.

Declaration

+ (NSData *)generateAllReceipts:(NSString *)urlString emulation:(StarIoExtEmulation)emulation info:(BOOL)info qrCode:(BOOL)qrCode;
+ (NSData *)generateAllReceipts:(NSString *)urlString emulation:(StarIoExtEmulation)emulation info:(BOOL)info qrCode:(BOOL)qrCode width:(NSInteger)width;

Parameter

Name	Description	Object type
urlString	Uploaded URL	NSString
emulation	Emulation type. • StarIoExtEmulationStarPRNT … StarPRNT emulation. • StarIoExtEmulationStarLine … STAR Line Mode emulation. • StarIoExtEmulationStarGraphic … STAR Graphic Mode emulation. • StarIoExtEmulationEscPos … ESC/POS emulation. • ~~StarIoExtEmulationEscPosMobile … ESC/POS Mobile emulation.~~ • ~~StarIoExtEmulationStarDotImpact … STAR Dot Impact emulation.~~	StarIoExtEmulation
info	Generates information logo. • YES … Valid. • NO … Invalid.	BOOL
qrCode	Generates QR code. • YES … Valid. • NO … Invalid.	BOOL
width	Printable width. (Units : Dots)	NSInteger

Return value

Description	Object type
Generated print data.	NSData

Constant of "StarIoExtEmulation" for each model, refer to .

Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation image:(UIImage *)image {
    NSString *urlString = [SMCSAllReceipts uploadBitmap:image completion:nil];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendBitmap:image diffusion:NO];

    NSData *data;

    if (emulation == StarIoExtEmulationStarGraphic) {
        data = [SMCSAllReceipts generateAllReceipts:urlString emulation:emulation info:info qrCode:qrCode width:width];    // Support to centering in Star Graphic.
    }
    else {
        data = [SMCSAllReceipts generateAllReceipts:urlString emulation:emulation info:info qrCode:qrCode];    // Non support to centering in Star Graphic.
    }

    [builder appendRawData:data];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

    return [builder.commands copy];
}
```

Refer to AllReceiptsFunctions.m.

252 / 262

22 Deprecated API

22.1 SMPort class

22.1.1 getPort Method
This method has been deprecated. Please use the getPort method instead.

Creates a SMPort object and opens a port for communicating with the printer.

Declaration
+ (SMPort *) getPort: (NSString *) portName : (NSString *) portSettings : (u_int32_t) timeoutMillis;

Parameter

Name	Description	Object type
portName	Character string to specify the port for communication with the printer.	NSString *
portSettings	Specifies connection setting information. ・Emulation type ・Data timeout	NSString *
timeoutMillis	Timeout value for internal control and API. [unit: ms]	u_int32_t

Return value

Description	Object type
An instance of SMPort class. It returns "nil" if it fails to generate communication port.	SMPort *

22.1.2 searchPrinter Method

This method has been deprecated. Please use the searchPrinter method instead.

Searches for a printer that can be connected to the iOS device.

Declaration

+ (NSArray *) searchPrinter;

+ (NSArray *) searchPrinter: (NSString *) target;

Parameter

Name	Description		Object type
target	Specify the interface type of the Star printer to be searched. If the target is not specified, it searches all interface types.		NSString *
	Bluetooth	"BT:"	
	Bluetooth Low Energy	"BLE:"	
	Ethernet / Wireless LAN	"TCP:"	
	USB	"USB:"	

Return value

Description	Object type
Search result of Star printer. NSArray of return value includes instance of PortInfo class. Refer to the PortInfo class for the information that you can get.	NSArray *

22.1.3 writePort Method

This method has been deprecated. Please use the writePort method instead.

This method writes data to the device. Use this to print to the printer, send commands, etc.
To check the completion of printing, run beginCheckedBlock before and endCheckedBlock after this method.

Declaration
- (u_int32_t) writePort:(u_int8_t const *) writeBuffer :(u_int32_t) offset :(u_int32_t) size;

Parameter

Name	Description	Object type
writeBuffer	Contains the output data in a byte array.	u_int8_t const *
offset	Specifies where to begin pulling data from writeBuffer.	u_int32_t
size	Number of bytes to write.	u_int32_t

Return value

Description	Object type
Bluetooth/Ethernet/Wireless LAN/USB I/F The number of bytes that were actually written. The writePort method is successful even when all of the data cannot be written. Your application should call this function a limited number of times until all the data has been written out or until an application determined retry threshold has been reached.	u_int32_t
Bluetooth Low Energy I/F It returns a transmission data size when it succeeded and "0" when it failed.	

Exception

Description	Object type
When a communication failure occurs	PortException

22.1.4 readPort Method

This method has been deprecated. Please use the <u>readPort method</u> instead.

Read data from the printer. Please use it only when it is necessary to read Raw byte from the printer.

Declaration

- (u_int32_t) readPort:(u_int8_t *) readBuffer :(u_int32_t) offSet :(u_int32_t) size;

Parameter

Name	Description	Object type
readBuffer	A Byte Array buffer into which data is read.	u_int8_t *
offset	specifies where to begin writing data into the readBuffer	u_int32_t
size	Total number of bytes to read.	u_int32_t

Return value

Description	Object type
The number of bytes that were read. The readPort method will succeed even when no all data was read in. Your application should call this function a limited number of times until the expected data has been read in or until an application determined retry threshold has been reached.	u_int32_t

Exception

Description	Object type
when a communication failure occurs	PortException

22.1.5 getParsedStatus Method

This method has been deprecated. Please use the getParsedStatus method instead.

Get printer status.

Declaration
- (void) getParsedStatus:(void *) starPrinterStatus :(u_int32_t) level;

Parameter

Name	Description	Object type
starPrinterStatus	StarPrinterStatus structure giving the current device status. For the type of status that can be obtained, refer to the StarPrinterStatus structure.	void *
level	StarPrinterStatus structure level (Possible to specify a value of 0, 1 or 2. Normally 2 is specified.)	u_int32_t

Return value

Description	Object type
-	-

Exception

Description	Object type
when a communication failure occurs	PortException

22.1.6 beginCheckedBlock Method

This method has been deprecated. Please use the beginCheckedBlock method instead.

This method is used in combination with endCheckedBlock and checks the completion of printing. beginCheckedBlock must be run just before sending print data.

Declaration
- (void) beginCheckedBlock:(void *) starPrinterStatus :(u_int32_t) level;

Parameter

Name	Description	Object type
starPrinterStatus	A pointer to StarPrinterStatus structure (Possible to specify StarPrinterStatus, StarPrinterStatus_1 of StarPrinterStatus_2. Normally StarPrinterStatus_2 is specified.) When this method is successful, the status of the current printer is stored.	void *
level	The level of StarPrinterStatus structure (Possible to specify a value of 0, 1 or 2. Normally 2 is specified.)	u_int32_t

Return value

Description	Object type
-	-

Exception

Description	Object type
when a communication failure occurs	PortException

22.1.7 endCheckedBlock Method

This method has been deprecated. Please use the endCheckedBlock method instead.

This method is used together with the beginCheckedBlock method in a set and checks the completion of printing.
It monitors printer status and when the transferred data is printed completely, returns control. In case of being transferred other kind of data than print data, when its command is processed in the printer, it returns the control.

Declaration
- (void) endCheckedBlock:(void *) starPrinterStatus :(u_int32_t) level;

Parameter

Name	Description	Object type
starPrinterStatus	a pointer to StarPrinterStatus structure (Possible to specify StarPrinterStatus, StarPrinterStatus_1 of StarPrinterStatus_2. Normally StarPrinterStatus_2 is specified.) When this method is successful, the status of the current printer is stored.	void *
level	the level of StarPrinterStatus structure (Possible to specify a value of 0,1 or 2. Normally 2 is specified.)	u_int32_t

Return value

Description	Object type
-	-

Exception

Description	Object type
-When a communication failure (An error sending the command such as Off-Line) occurs - No response for the completion of printing from a printer within the timeout	PortException

22.1.8 disconnect Method

This method has been deprecated. Please use the disconnect method instead.

Disconnect the specified Bluetooth device.
After the disconnection, the Bluetooth device can be connected by other iOS terminals.

Declaration
- (BOOL) disconnect;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
Disconnect success and fail This method fails in the following cases: - when the disconnection has not been completed within the timeout specified by getPort - when the disconnection function is not supported by a printer (such like portable printers). This method has no effect on Ethernet devices. It always returns YES when it was run with the Ethernet device.	BOOL

22.1.9 getFirmwareInformation Method

This method has been deprecated. Please use the getFirmwareInformation method instead.

This method gets a model name and firmware Information of the printer.

Declaration
- (NSDictionary *) getFirmwareInformation;

Parameter

Name	Description	Object type
-	-	-

Return value

Description	Object type
An acquisition result of firmware information The return value can get the model name by setting the NSDictionary object key of the return value to "@ModelName" and get the firmware version by setting "@FirmwareVersion".	NSDictionary *

Exception

Description	Object type
If it failed to get information, it returns an empty string.	PortException

Appendix A. How to use AllReceipts™ (Guides for Retailers)

Please visit the following URL for details about "AllReceipts™", FREE digital receipts service from Star Micronics.

www.allreceipts.com

You can start using our services immediately after you complete your registration from the following website.

<< Star Micronics Cloud Retailer Registration site >>
www.starmicronicscloud.com
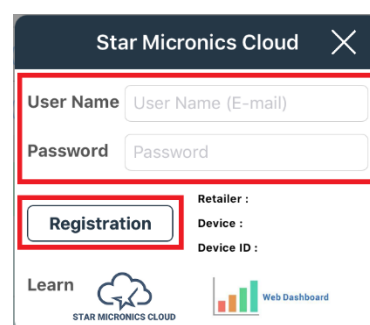
1) Device Registration

Register the device using your registered Star Micronics Cloud account.
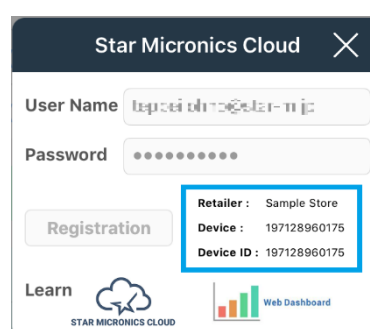
Once you have registered the device, you can use the service of uploading print data to the cloud server and manage the registered device from the Star Micronics Cloud dashboard.

2) Enter the username and the password of your registered Star Micronics Cloud account and click "Registeration".

3) When the device is registered successfully, the account information of the connected Star Micronics Cloud appears.