

# StarPRNT iOS SDK ユーザーズマニュアル

2016 年 04 月 01 日

## 要約

---

このマニュアルは StarPRNT iOS SDK に関する情報を提供するものです。

ユーザー向けに StarPRNT アプリケーションの構築に関するガイドラインを示します。

#### ドキュメントの履歴

日付	変更内容
2016 年 04 月 01 日	発行

## 目次

1	iOS StarPRNT プロジェクト構成方法 .....	6
1.1	iOS StarPRNT SDK をプロジェクトに追加 .....	6
1.2	必要なフレームワークをプロジェクトに追加 .....	6
1.3	Information Property List へ項目を設定 .....	6
2	StarIoExt class included in the StarIO_Extension.framework.....	7
2.1	createCommandBuilder メソッド .....	7
2.2	StarIoExtEmulation コンスタント .....	8
2.3	Model : portSetting and StarIoExtEmulation .....	9
3	ISCBBuilder interface included in the StarIO_Extension.framework.....	10
3.1	モデル : ISCBBuilder interface メソッド .....	13
3.2	beginDocument メソッド .....	17
3.3	endDocument メソッド .....	18
3.4	appendInitialization メソッド .....	19
3.5	appendData メソッド .....	20
3.6	appendRawData メソッド .....	21
3.7	appendFontStyle メソッド .....	22
3.8	appendCodePage メソッド .....	23
3.9	appendInternational メソッド .....	26
3.10	appendLineFeed メソッド .....	28
3.11	appendUnitFeed メソッド .....	29
3.12	appendCharacterSpace メソッド .....	30
3.13	appendLineSpace メソッド .....	31
3.14	appendEmphasis メソッド .....	32
3.15	appendInvert メソッド .....	33
3.16	appendMultiple メソッド .....	34
3.17	appendUnderLine メソッド .....	36
3.18	appendLogo メソッド .....	37
3.19	appendAbsolutePosition メソッド .....	38
3.20	appendAlignment メソッド .....	39
3.21	appendCutPaper メソッド .....	40
3.22	appendPeripheral メソッド .....	41
3.23	appendSound メソッド .....	42
3.24	appendBarcodeData メソッド .....	43
3.25	appendBarcodeDataWithAbsolutePosition メソッド .....	45
3.26	appendBarcodeDataWithAlignment メソッド .....	47
3.27	appendPdf417Data メソッド .....	49
3.28	appendPdf417DataWithAbsolutePosition メソッド .....	51
3.29	appendPdf417DataWithAlignment メソッド .....	53
3.30	appendQrCodeData メソッド .....	55
3.31	appendQrCodeDataWithAbsolutePosition メソッド .....	56
3.32	appendQrCodeDataWithAlignment メソッド .....	57
3.33	appendBitmap メソッド .....	59
3.34	appendBitmapWithAbsolutePosition メソッド .....	61
3.35	appendBitmapWithAlignment メソッド .....	63
3.36	commands プロパティ .....	65
3.37	SCBInitializationType コンスタント .....	66
3.38	SCBFontStyleType コンスタント .....	66

3.39	SCBCodePageType コンスタント .....	67
3.40	SCBInternationalType コンスタント .....	69
3.41	SCBLogoSize コンスタント .....	70
3.42	SCBAlignmentPosition コンスタント .....	70
3.43	SCBCutPaperAction コンスタント .....	71
3.44	SCBPeripheralChannel コンスタント .....	71
3.45	SCBSoundChannel コンスタント .....	72
3.46	SCBBarcodeSymbology コンスタント .....	72
3.47	SCBBarcodeWidth コンスタント .....	73
3.48	SCBPdf417Level コンスタント .....	74
3.49	SCBQRCodeModel コンスタント .....	74
3.50	SCBQRCodeLevel コンスタント .....	75
3.51	SCBBitmapConverterRotation コンスタント .....	75
4	StarIoExtManager class included in the StarIO_Extension.framework .....	76
4.1	initWithType メソッド .....	77
4.2	connect メソッド .....	78
4.3	disconnect メソッド .....	79
4.4	port プロパティ .....	79
4.5	lock プロパティ .....	80
4.6	delegate プロパティ .....	80
4.7	printerOnlineStatus プロパティ .....	81
4.8	printerPaperStatus プロパティ .....	81
4.9	printerCoverStatus プロパティ .....	82
4.10	cashDrawerStatus プロパティ .....	82
4.11	barcodeReaderStatus プロパティ .....	83
4.12	cashDrawerOpenActiveHigh プロパティ .....	83
4.13	StarIoExtManagerType コンスタント .....	84
4.14	StarIoExtManagerPrinterStatus コンスタント .....	84
4.15	StarIoExtManagerPrinterPaperStatus コンスタント .....	85
4.16	StarIoExtManagerPrinterCoverStatus コンスタント .....	85
4.17	StarIoExtManagerCashDrawerStatus コンスタント .....	86
4.18	StarIoExtManagerBarcodeReaderStatus コンスタント .....	86
5	StarIoExtManagerDelegate class included in the StarIO_Extension.framework .....	87
5.1	didPrinterImpossible メソッド .....	87
5.2	didPrinterOnline メソッド .....	88
5.3	didPrinterOffline メソッド .....	88
5.4	didPrinterPaperReady メソッド .....	89
5.5	didPrinterPaperNearEmpty メソッド .....	89
5.6	didPrinterPaperEmpty メソッド .....	90
5.7	didPrinterCoverOpen メソッド .....	90
5.8	didPrinterCoverClose メソッド .....	91
5.9	didCashDrawerOpen メソッド .....	91
5.10	didCashDrawerClose メソッド .....	92
5.11	didBarcodeReaderImpossible メソッド .....	92
5.12	didBarcodeReaderConnect メソッド .....	93
5.13	didBarcodeReaderDisconnect メソッド .....	93
5.14	didBarcodeDataReceive メソッド .....	94
5.15	didAccessoryConnectSuccess メソッド .....	95

5.16	didAccessoryConnectFailure メソッド .....	95
5.17	didAccessoryDisconnect メソッド .....	96
5.18	didStatusUpdate メソッド .....	97
6	StarPRNT iOS SDK Sample .....	98
6.1	Communication .....	98
7	API expanded for Swift support included in the StarIO.framework .....	101
8	SMCloudServices class included in the SMCloudServices.framework .....	102
8.1	showRegistrationView メソッド .....	102
8.2	isRegistered メソッド .....	103
9	SMCSAllReceipts class included in the SMCloudServices.framework .....	104
9.1	モデル : SMCSAllReceipts class メソッド .....	104
9.2	uploadBitmap メソッド .....	105
9.3	updateStatus メソッド .....	106
9.4	generateAllReceipts メソッド .....	107
Appendix A. AllReceipt <sup>TM</sup> 利用方法案内 (配布用) .....		109

## 1 iOS StarPRNT プロジェクト構成方法

### 1.1 iOS StarPRNT SDK をプロジェクトに追加

- “Build Phases”タブの”Link Binary With Libraries”に StarIO.framework を追加する
- “Build Phases”タブの”Link Binary With Libraries”に StarIO\_Extension.framework を追加する
- “Build Phases”タブの”Link Binary With Libraries”に SMCloudServices.framework を追加する
- “Build Phases”タブの”Copy Bundle Resources”に SMCloudServices.framework フォルダ配下にある SMCloudServicesResources.bundle を追加する

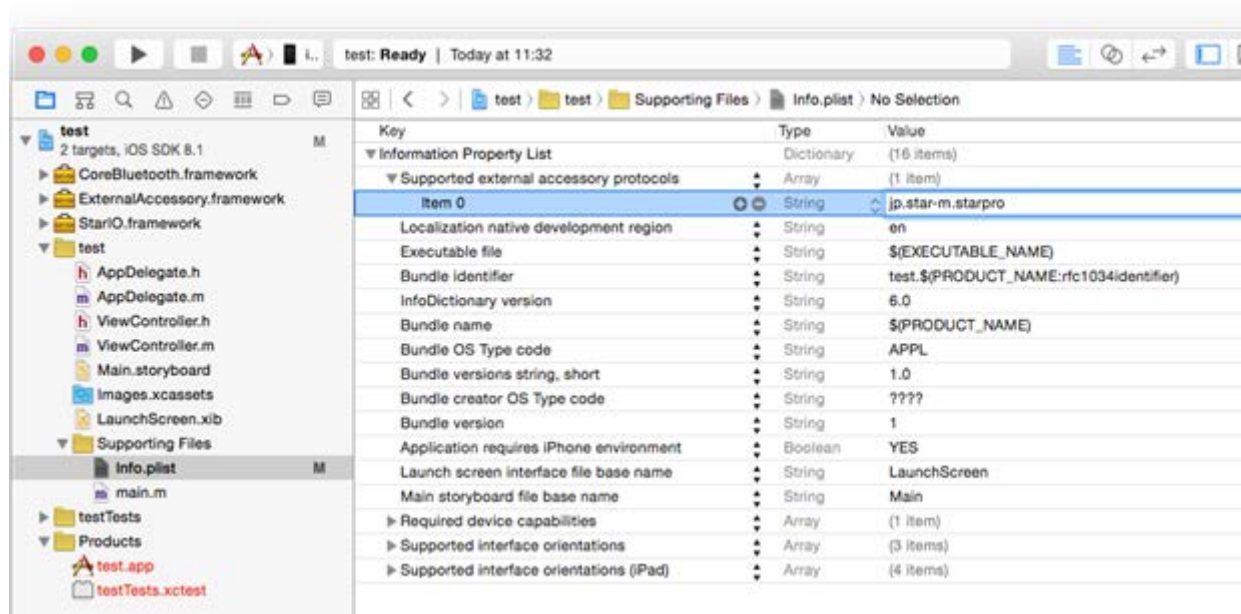
### 1.2 必要なフレームワークをプロジェクトに追加

- CoreBluetooth.framework
- ExternalAccessory.framework

### 1.3 Information Property List へ項目を設定

※Bluetooth プリンターを使用しない場合は、この設定を行わないでください。

1. Information Property List（デフォルトでは”Info.plist”）を選択します。
2. Key ”Supported external accessory protocols”を追加
3. 項目名左側の▽をクリックして表示される”Item 0”の[Value]に ”jp.star-m.starpro” を設定



## 2 StarloExt class included in the StarIO\_Extension.framework

### メソッド

名称	説明
createCommandBuilder	コマンド生成オブジェクトを生成します。

### コンスタント

名称	説明
StarloExtEmulation	エミュレーション指定定数。

### 2.1 createCommandBuilder メソッド

コマンド生成オブジェクトを生成します。

### 宣言

```
+ (ISCBuilder *)createCommandBuilder:( StarloExtEmulation)emulation;
```

### 引数

名称	説明	型
emulation	エミュレーション指定 <ul style="list-style-type: none"> <li>StarloExtEmulationStarPRNT ... StarPRNT エミュレーション</li> <li>StarloExtEmulationStarLine ... STAR Line Mode エミュレーション</li> <li>StarloExtEmulationStarGraphic ... STAR Graphic Mode エミュレーション</li> <li>StarloExtEmulationEscPos ... ESC/POS エミュレーション</li> <li>StarloExtEmulationEscPosMobile ... ESC/POS Mobile エミュレーション</li> <li>StarloExtEmulationStarDotImpact ... STAR ドットインパクトエミュレーション</li> </ul>	StarloExtEmulation

### 戻り値

説明	型
ISCBuilder オブジェクト	ISCBuilder

## Example

```
+ (NSData *)createCommandImage:(StarloExtEmulation)emulation image:(UIImage *)image {
    ISCBBuilder *builder = [StarloExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendBitmap:image diffusion:NO];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

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

PrinterFunctions.m を参照ください。

## 2.2 StarloExtEmulation コンスタント

エミュレーション指定定数。

### 宣言

```
typedef NS_ENUM(NSUInteger, StarloExtEmulation) {
    StarloExtEmulationNone = 0,
    StarloExtEmulationStarPRNT,
    StarloExtEmulationStarLine,
    StarloExtEmulationStarGraphic,
    StarloExtEmulationEscPos,
    StarloExtEmulationEscPosMobile,
    StarloExtEmulationStarDotImpact
};
```

### 定数

名称	説明
StarloExtEmulationStarPRNT	StarPRNT エミュレーション
StarloExtEmulationStarLine	STAR Line Mode エミュレーション
StarloExtEmulationStarGraphic	STAR Graphic Mode エミュレーション
StarloExtEmulationEscPos	ESC/POS エミュレーション
StarloExtEmulationEscPosMobile	ESC/POS Mobile エミュレーション
StarloExtEmulationStarDotImpact	STAR ドットインパクトエミュレーション



## 2.3 Model : portSetting and StarloExtEmulation

各モデルに対して適用すべき portSettings 文字列と StarloExtEmulation 定数は以下のとおりです。

Model	portSettings	StarloExtEmulation
mPOP	""	StarloExtEmulationStarPRNT
FVP10	""	StarloExtEmulationStarLine
TSP100	""	StarloExtEmulationStarGraphic
TSP650II	""	StarloExtEmulationStarLine
TSP700II	""	StarloExtEmulationStarLine
TSP800II	""	StarloExtEmulationStarLine
SM-S210i	"mini"	StarloExtEmulationEscPosMobile
SM-S220i	"mini"	StarloExtEmulationEscPosMobile
SM-S230i	"mini"	StarloExtEmulationEscPosMobile
SM-T300i	"mini"	StarloExtEmulationEscPosMobile
SM-T400i	"mini"	StarloExtEmulationEscPosMobile
BSC10	"escpos"	StarloExtEmulationEscPos
SM-S210i StarPRNT	"Portable"	StarloExtEmulationStarPRNT
SM-S220i StarPRNT	"Portable"	StarloExtEmulationStarPRNT
SM-S230i StarPRNT	"Portable"	StarloExtEmulationStarPRNT
SM-T300i StarPRNT	"Portable"	StarloExtEmulationStarPRNT
SM-T400i StarPRNT	"Portable"	StarloExtEmulationStarPRNT
SM-L200	"Portable"	StarloExtEmulationStarPRNT
SP700	""	StarloExtEmulationStarDotImpact

portSettings は StarIO.framework に含まれている SMPort クラスの getPort メソッドもしくは StarIO\_Extension.framework に含まれている StarloExtManager クラスの initWithType メソッドに使用します。

StarloExtEmulation は StarIO\_Extension.framework に含まれている StarloExt クラスの createCommandBuilder メソッドに使用します。

### 3 ISCBBuilder interface included in the StarIO\_Extension.framework

#### メソッド

名称	説明
beginDocument	ドキュメント開始コマンドを生成し、commands プロパティに追加します。
endDocument	ドキュメント終了コマンドを生成し、commands プロパティに追加します。
appendInitialization	初期化コマンドを生成し、commands プロパティに追加します。
appendByte	データ（テキストやコマンド）を commands プロパティに追加します。
appendData	
appendBytes	
appendRawByte	生のデータ（テキストやコマンド）を commands プロパティに追加します。
appendRawData	
appendRawBytes	
appendFontStyle	フォントスタイル指定コマンドを生成し、commands プロパティに追加します。
appendCodePage	コードページ指定コマンドを生成し、commands プロパティに追加します。
appendInternational	国際文字指定コマンドを生成し、commands プロパティに追加します。
appendLineFeed	行単位の用紙送りコマンドを生成し、commands プロパティに追加します。
appendDataWithLineFeed	
appendBytesWithLineFeed	
appendUnitFeed	1 ドットライン単位の用紙送りコマンドを生成し、commands プロパティに追加します。
appendDataWithUnitFeed	
appendBytesWithUnitFeed	
appendCharacterSpace	文字間スペース設定コマンドを生成し、commands プロパティに追加します。
appendLineSpace	行間スペース設定コマンドを生成し、commands プロパティに追加します。
appendEmphasis	強調指定コマンドを生成し、commands プロパティに追加します。
appendDataWithEmphasis	
appendBytesWithEmphasis	
appendInvert	白黒反転指定コマンドを生成し、commands プロパティに追加します。
appendDataWithInvert	
appendBytesWithInvert	
appendMultiple	拡大指定コマンドを生成し、commands プロパティに追加します。
appendDataWithMultiple	
appendBytesWithMultiple	
appendMultipleHeight	
appendDataWithMultipleHeight	
appendBytesWithMultipleHeight	
appendMultipleWidth	
appendDataWithMultipleWidth	
appendBytesWithMultipleWidth	
appendUnderLine	アンダーライン指定コマンドを生成し、commands プロパティに追加します。
appendDataWithUnderLine	
appendBytesWithUnderLine	
appendLogo	ロゴ印刷コマンドを生成し、commands プロパティに追加します。

名称	説明
appendAbsolutePosition	絶対位置指定コマンドを生成し、commands プロパティに追加します。
appendDataWithAbsolutePosition	
appendBytesWithAbsolutePosition	
appendAlignment	位置揃え指定コマンドを生成し、commands プロパティに追加します。
appendDataWithAlignment	
appendBytesWithAlignment	
appendCutPaper	用紙カットコマンドを生成し、commands プロパティに追加します。
appendPeripheral	ペリフェラル駆動コマンドを生成し、commands プロパティに追加します。
appendSound	サウンド駆動コマンドを生成し、commands プロパティに追加します。
appendBarcodeData	バーコード印刷コマンドを生成し、commands プロパティに追加します。
appendBarcodeBytes	
appendBarcodeDataWithAbsolutePosition	絶対位置指定バーコード印刷コマンドを生成し、commands プロパティに追加します。
appendBarcodeBytesWithAbsolutePosition	
appendBarcodeDataWithAlignment	位置揃えバーコード印刷コマンドを生成し、commands プロパティに追加します。
appendBarcodeBytesWithAlignment	
appendPdf417Data	PDF417 印刷コマンドを生成し、commands プロパティに追加します。
appendPdf417Bytes	
appendPdf417DataWithAbsolutePosition	絶対位置指定 PDF417 印刷コマンドを生成し、commands プロパティに追加します。
appendPdf417BytesWithAbsolutePosition	
appendPdf417DataWithAlignment	位置揃え PDF417 印刷コマンドを生成し、commands プロパティに追加します。
appendPdf417BytesWithAlignment	
appendQRCodeData	QR コード印刷コマンドを生成し、commands プロパティに追加します。
appendQRCodeBytes	
appendQRCodeDataWithAbsolutePosition	絶対位置指定 QR コード印刷コマンドを生成し、commands プロパティに追加します。
appendQRCodeBytesWithAbsolutePosition	
appendQRCodeDataWithAlignment	位置揃え QR コード印刷コマンドを生成し、commands プロパティに追加します。
appendQRCodeBytesWithAlignment	
appendBitmap	ビットマップ印刷コマンドを生成し、commands プロパティに追加します。
appendBitmapWithAbsolutePosition	絶対位置指定ビットマップ印刷コマンドを生成し、commands プロパティに追加します。
appendBitmapWithAlignment	位置揃えビットマップ印刷コマンドを生成し、commands プロパティに追加します。

## プロパティ

名称	説明
commands	生成、追加されたコマンド列です。

## コンスタント

名称	説明
SCBInitializationType	初期化指定定数。
SCBFontStyleType	フォントスタイル指定定数。
SCBCodePageType	コードページ指定定数。
SCBInternationalType	国際文字指定定数。
SCBLogoSize	ロゴサイズ指定定数。
SCBAlignmentPosition	位置揃え指定定数。
SCBCutPaperAction	用紙カット指定定数。
SCBPeripheralChannel	ペリフェラルチャネル指定定数。
SCBSoundChannel	サウンドチャネル指定定数。
SCBBarcodeSymbology	バーコードシンボル指定定数。
SCBBarcodeWidth	バーコード幅指定定数。
SCBPdf417Level	PDF417 ECC（セキュリティレベル）指定定数。
SCBQRCodeModel	QRコードモデル指定定数
SCBQRCodeLevel	QRコード誤り訂正レベル指定定数。
SCBBitmapConverterRotation	ビットマップ回転指定定数。

### 3.1 モデル : ISCBBUILDER interface メソッド

各モデルにおいて機能するメソッドは以下のとおりです。

Function	Method	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	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	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

- : 無視

\*1 : データ（テキストやコマンド）の追加は行わず、1 ドットライン単位用の紙送りコマンドの追加のみ行う

Function	Method	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	SP700
Emphasis	appendEmphasis	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendDataWithEmphasis	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBytesWithEmphasis	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Invert	appendInvert	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendDataWithInvert	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBytesWithInvert	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Multiple	appendMultiple	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendDataWithMultiple	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBytesWithMultiple	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendMultipleHeight	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendDataWithMultipleHeight	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBytesWithMultipleHeight	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendMultipleWidth	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendDataWithMultipleWidth	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBytesWithMultipleWidth	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Under line	appendUnderLine	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendDataWithUnderLine	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBytesWithUnderLine	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Logo	appendLogo	✓	✓	-	✓	✓	✓	*2	*2	*2	*2	*2	✓	✓	✓	✓	✓	✓	✓	✓
Absolute position	appendAbsolutePosition	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendDataWithAbsolutePosition	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBytesWithAbsolutePosition	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Alignment	appendAlignment	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendDataWithAlignment	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBytesWithAlignment	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

- : 無視

\*2 : 常にノーマルサイズ

Function	Method	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	SP700
Cut paper	appendCutPaper	✓	✓	✓	✓	✓	✓	*3	*3	*3	*3	*3	✓	*3	*3	*3	*3	*3	*3	✓
Peripheral	appendPeripheral	✓	✓	✓	✓	✓	✓	-	-	-	-	-	✓	-	-	-	-	-	-	✓
Sound	appendSound	✓	✓	✓	✓	✓	✓	-	-	-	-	-	✓	-	-	-	-	-	-	✓
Barcode	appendBarcodeData	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendBarcodeBytes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendBarcodeDataWithAbsolutePosition	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendBarcodeBytesWithAbsolutePosition	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendBarcodeDataWithAlignment	✓	✓	*4	✓	✓	✓	*4	*4	*4	*4	*4	✓	✓	✓	✓	✓	✓	✓	-
	appendBarcodeBytesWithAlignment	✓	✓	*4	✓	✓	✓	*4	*4	*4	*4	*4	✓	✓	✓	✓	✓	✓	✓	-
PDF417	appendPdf417Data	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendPdf417Bytes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendPdf417DataWithAbsolutePosition	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendPdf417BytesWithAbsolutePosition	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendPdf417DataWithAlignment	✓	✓	*4	✓	✓	✓	*4	*4	*4	*4	*4	✓	✓	✓	✓	✓	✓	✓	-
	appendPdf417BytesWithAlignment	✓	✓	*4	✓	✓	✓	*4	*4	*4	*4	*4	✓	✓	✓	✓	✓	✓	✓	-

- : 無視

\*3 : 用紙カットを行わない

\*4 : 常に左揃え

Function	Method	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	SP700
QR code	appendQrCodeData	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendQrCodeBytes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendQrCodeDataWithAbsolutePosition	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendQrCodeBytesWithAbsolutePosition	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	appendQrCodeDataWithAlignment	✓	✓	*4	✓	✓	✓	*4	*4	*4	*4	*4	✓	✓	✓	✓	✓	✓	✓	-
	appendQrCodeBytesWithAlignment	✓	✓	*4	✓	✓	✓	*4	*4	*4	*4	*4	✓	✓	✓	✓	✓	✓	✓	-
Bitmap	appendBitmap	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBitmapWithAbsolutePosition	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	appendBitmapWithAlignment	✓	✓	*4	✓	✓	✓	*4	*4	*4	*4	*4	✓	✓	✓	✓	✓	✓	✓	✓

- : 無視

\*4 : 常に左揃え



## 3.2 beginDocument メソッド

ドキュメント開始コマンドを生成し、commands プロパティに追加します。

### 宣言

- (void)beginDocument;

### 引数

名称	説明	型
-	-	-

### 戻り値

説明	型
-	-

### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

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

ApiFunctions.m を参照ください。

### 3.3 endDocument メソッド

ドキュメント終了コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)endDocument;

#### 引数

名称	説明	型
-	-	-

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarIoExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

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

ApiFunctions.m を参照ください。

### 3.4 appendInitialization メソッド

初期化コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendInitialization:(SCBInitializationType)type;

#### 引数

名称	説明	型
type	初期化タイプ <ul style="list-style-type: none"> <li>SCBInitializationTypeCommand ... コマンド初期化</li> <li>SCBInitializationTypeReset ... プリンタリセット</li> <li>SCBInitializationTypeResetWithPrint ... プリンタリセット（自己印字実行）</li> </ul>	SCBInitializationType

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    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];
}
```

ApiFunctions.m を参照ください。

### 3.5 appendData メソッド

データ（テキストやコマンド）を commands プロパティに追加します。

#### 宣言

- (void)appendByte:(unsigned char)data;
- (void)appendData:(NSData \*)otherData;
- (void)appendBytes:(const void \*)bytes length:(NSUInteger)length;

#### 引数

名称	説明	型
data	データ（テキストやコマンド）	unsigned char
otherData		NSData
bytes		const void *
length		NSUInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSUTF8StringEncoding];

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

    NSUInteger length = sizeof(bytes);

    ISCBBuilder *builder = [StarloExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];
    [builder appendByte:0x0a];

    [builder appendBytes:bytes length:length];
    [builder appendByte:0x0a];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

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

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.6 appendRawData メソッド

生のデータ（テキストやコマンド）を commands プロパティに追加します。

#### 宣言

- (void)appendRawByte:(unsigned char)data;
- (void)appendRawData:(NSData \*)otherData;
- (void)appendRawBytes:(const void \*)bytes length:(NSUInteger)length;

#### 引数

名称	説明	型
data	生のデータ（テキストやコマンド）	unsigned char
otherData		NSData
bytes		const void *
length		NSUInteger

#### 戻り値

説明	型
-	-

#### Example

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

    ISCBBuilder *builder = [StarloExt 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];
}
```

AllReceiptsFunctions.m /各コマンド仕様書を参照ください。

### 3.7 appendFontStyle メソッド

フォントスタイル指定コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendFontStyle:(SCBFontStyleType)type;

#### 引数

名称	説明	型
type	フォントスタイル <ul style="list-style-type: none"> <li>SCBFontStyleTypeA ... フォント A (12 x 24 ドット) / 7 x 9 フォント (ハーフドット)</li> <li>SCBFontStyleTypeB ... フォント B (9 x 24 ドット) / 5 x 9 フォント (2P-1)</li> </ul>	SCBFontStyleType

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.8 appendCodePage メソッド

コードページ指定コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (void)appendCodePage:(SCBCodePageType)type;
```

#### 引数

名称	説明	型
type	<p>コードページ</p> <ul style="list-style-type: none"> <li>• SCBCodePageTypeCP437 ... CodePage437 (USA, Std. Europe).</li> <li>• SCBCodePageTypeCP737 ... Codepage 737 (Greek).</li> <li>• SCBCodePageTypeCP772 ... Codepage 772 (Lithuanian).</li> <li>• SCBCodePageTypeCP774 ... Codepage 774 (Lithuanian).</li> <li>• SCBCodePageTypeCP851 ... Codepage 851 (Greek).</li> <li>• SCBCodePageTypeCP852 ... Codepage 852 (Latin-2).</li> <li>• SCBCodePageTypeCP855 ... Codepage 855 (Cyrillic Bulgarian).</li> <li>• SCBCodePageTypeCP857 ... Codepage 857 (Turkey).</li> <li>• SCBCodePageTypeCP858 ... Codepage 858 (Multilingual).</li> <li>• SCBCodePageTypeCP860 ... Codepage 860 (Portuguese).</li> <li>• SCBCodePageTypeCP861 ... Codepage 861 (Icelandic).</li> <li>• SCBCodePageTypeCP862 ... Codepage 862 (Israel (Hebrew)).</li> <li>• SCBCodePageTypeCP863 ... Codepage 863 (Canadian French).</li> <li>• SCBCodePageTypeCP864 ... Codepage 864 (Arabic).</li> <li>• SCBCodePageTypeCP865 ... Codepage 865 (Nordic).</li> <li>• SCBCodePageTypeCP866 ... Codepage 866 (Cyrillic Russian).</li> <li>• SCBCodePageTypeCP869 ... Codepage 869 (Greek).</li> <li>• SCBCodePageTypeCP874 ... Codepage 874 (Thai).</li> <li>• SCBCodePageTypeCP928 ... Codepage 928 (Greek).</li> </ul>	SCBCodePageType

名称	説明	型
type	<ul style="list-style-type: none"> <li>• SCBCodePageTypeCP932 ... Katakana.</li> <li>• SCBCodePageTypeCP998 ... Normal.</li> <li>• SCBCodePageTypeCP999 ... Codepage 1252 (Windows Latin-1).</li> <li>• SCBCodePageTypeCP1001 ... Codepage 1001 (Arabic).</li> <li>• SCBCodePageTypeCP1250 ... Codepage 1250 (Windows Latin-2).</li> <li>• SCBCodePageTypeCP1251 ... Codepage 1251 (Windows Cyrillic).</li> <li>• SCBCodePageTypeCP1252 ... Codepage 1252 (Windows Latin-1).</li> <li>• SCBCodePageTypeCP2001 ... Codepage 2001 (Lithuanian-KBL).</li> <li>• SCBCodePageTypeCP3001 ... Codepage 3001 (Estonian-1).</li> <li>• SCBCodePageTypeCP3002 ... Codepage 3002 (Estonian-2).</li> <li>• SCBCodePageTypeCP3011 ... Codepage 3011 (Latvian-1).</li> <li>• SCBCodePageTypeCP3012 ... Codepage 3012 (Latvian-2).</li> <li>• SCBCodePageTypeCP3021 ... Codepage 3021 (Bulgarian).</li> <li>• SCBCodePageTypeCP3041 ... Codepage 3041 (Maltese).</li> <li>• SCBCodePageTypeCP3840 ... Codepage 3840 (IBM-Russian).</li> <li>• SCBCodePageTypeCP3841 ... Codepage 3841 (Gost).</li> <li>• SCBCodePageTypeCP3843 ... Codepage 3843 (Polish).</li> <li>• SCBCodePageTypeCP3844 ... Codepage 3844 (CS2).</li> <li>• SCBCodePageTypeCP3845 ... Codepage 3845 (Hungarian).</li> <li>• SCBCodePageTypeCP3846 ... Codepage 3846 (Turkish).</li> <li>• SCBCodePageTypeCP3847 ... Codepage 3847 (Brazil-ABNT).</li> <li>• SCBCodePageTypeCP3848 ... Codepage 3848 (Brazil-ABICOMP).</li> <li>• SCBCodePageTypeUTF8 ... UTF-8.</li> <li>• SCBCodePageTypeBlank ... User Setting Blank Code Page.</li> </ul>	SCBCodePageType



## 戻り値

説明	型
-	-

## Example

```
+ (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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.9 appendInternational メソッド

国際文字指定コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (void)appendInternational:(SCBInternationalType)type;
```

#### 引数

名称	説明	型
type	国際文字 <ul style="list-style-type: none"> <li>• SCBInternationalTypeUSA ... USA</li> <li>• SCBInternationalTypeFrance ... フランス</li> <li>• SCBInternationalTypeGermany ... ドイツ</li> <li>• SCBInternationalTypeUK ... イギリス</li> <li>• SCBInternationalTypeDenmark ... デンマーク</li> <li>• SCBInternationalTypeSweden ... スウェーデン</li> <li>• SCBInternationalTypeItaly ... イタリア</li> <li>• SCBInternationalTypeSpain ... スペイン</li> <li>• SCBInternationalTypeJapan ... 日本</li> <li>• SCBInternationalTypeNorway ... ノルウェー</li> <li>• SCBInternationalTypeDenmark2 ... デンマーク II</li> <li>• SCBInternationalTypeSpain2 ... スペイン II</li> <li>• SCBInternationalTypeLatinAmerica ... ラテンアメリカ</li> <li>• SCBInternationalTypeKorea ... 韓国</li> <li>• SCBInternationalTypeIreland ... アイルランド</li> <li>• SCBInternationalTypeLegal ... Legal</li> </ul>	SCBInternationalType

#### 戻り値

説明	型
-	-

### 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.10 appendLineFeed メソッド

行単位用の紙送りコマンドを生成し、commands プロパティに追加します。

#### 宣言

- (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;

#### 引数

名称	説明	型
line	紙送り量（行数単位）	NSInteger
otherData	データ（テキストやコマンド）	NSData
bytes		const void *
length		NSUInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m を参照ください。

### 3.11 appendUnitFeed メソッド

1 ドットライン単位用の紙送りコマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendUnitFeed:(NSInteger)unit;
- (void)appendDataWithUnitFeed:(NSData \*)otherData unit:(NSInteger)unit;
- (void)appendBytesWithUnitFeed:(const void \*)bytes length:(NSUInteger)length unit:(NSInteger)unit;

#### 引数

名称	説明	型
unit	紙送り量（ドット数単位）	NSInteger
otherData	データ（テキストやコマンド）	NSData
bytes		const void *
length		NSUInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSUTF8StringEncoding];

    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];
}
```

ApiFunctions.m を参照ください。

### 3.12 appendCharacterSpace メソッド

文字間スペース設定コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendCharacterSpace:(NSInteger)space;

#### 引数

名称	説明	型
space	文字間スペース（ドット数単位）	NSInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m を参照ください。

### 3.13 appendLineSpace メソッド

行間スペース設定コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendLineSpace:(NSInteger)lineSpace;

#### 引数

名称	説明	型
lineSpace	行間スペース（ドット数単位）	NSInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.14 appendEmphasis メソッド

強調指定コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendEmphasis:(BOOL)emphasis;
- (void)appendDataWithEmphasis:(NSData \*)otherData;
- (void)appendBytesWithEmphasis:(const void \*)bytes length:(NSUInteger)length;

#### 引数

名称	説明	型
emphasis	強調 <ul style="list-style-type: none"> <li>YES ... 強調設定</li> <li>NO ... 強調解除</li> </ul>	BOOL
otherData	データ (テキストやコマンド)	NSData
bytes		const void *
length		NSUInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];
    NSData *otherDataHalf0 = [@"Hello " dataUsingEncoding:NSUTF8StringEncoding];
    NSData *otherDataHalf1 = [@"World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m を参照ください。



### 3.15 appendInvert メソッド

白黒反転指定コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendInvert:(BOOL)invert;
- (void)appendDataWithInvert:(NSData \*)otherData;
- (void)appendBytesWithInvert:(const void \*)bytes length:(NSUInteger)length;

#### 引数

名称	説明	型
invert	白黒反転 <ul style="list-style-type: none"> <li>YES ... 白黒反転設定</li> <li>NO ... 白黒反転解除</li> </ul>	BOOL
otherData	データ (テキストやコマンド)	NSData
bytes		const void *
length		NSUInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];
    NSData *otherDataHalf0 = [@"Hello " dataUsingEncoding:NSUTF8StringEncoding];
    NSData *otherDataHalf1 = [@"World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m を参照ください。

### 3.16 appendMultiple メソッド

拡大指定コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (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;

#### 引数

名称	説明	型
width	横倍率	NSInteger
height	縦倍率	NSInteger
otherData	データ（テキストやコマンド）	NSData
bytes		const void *
length		NSUInteger

#### 戻り値

説明	型
-	-

## Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];
    NSData *otherDataHalf0 = [@"Hello " dataUsingEncoding:NSUTF8StringEncoding];
    NSData *otherDataHalf1 = [@"World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.17 appendUnderLine メソッド

アンダーライン指定コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendUnderLine:(BOOL)underLine;
- (void)appendDataWithUnderLine:(NSData \*)otherData;
- (void)appendBytesWithUnderLine:(const void \*)bytes length:(NSUInteger)length;

#### 引数

名称	説明	型
underLine	アンダーライン <ul style="list-style-type: none"> <li>YES ... アンダーライン設定</li> <li>NO ... アンダーライン解除</li> </ul>	BOOL
otherData	データ (テキストやコマンド)	NSData
bytes		const void *
length		NSUInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];
    NSData *otherDataHalf0 = [@"Hello " dataUsingEncoding:NSUTF8StringEncoding];
    NSData *otherDataHalf1 = [@"World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m を参照ください。

### 3.18 appendLogo メソッド

ロゴ印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

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

#### 引数

名称	説明	型
number	ロゴサイズ <ul style="list-style-type: none"> <li>SCBLogoSizeNormal ... ノーマルモード</li> <li>SCBLogoSizeDoubleWidth ... 横 2 倍モード</li> <li>SCBLogoSizeDoubleHeight ... 縦 2 倍モード</li> <li>SCBLogoSizeDoubleWidthDoubleHeight ... 縦 2 倍、横 2 倍モード</li> </ul>	SCBLogoSize

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.19 appendAbsolutePosition メソッド

絶対位置指定コマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendAbsolutePosition:(NSInteger)position;
- (void)appendDataWithAbsolutePosition:(NSData \*)otherData position:(NSInteger)position;
- (void)appendBytesWithAbsolutePosition:(const void \*)bytes length:(NSUInteger)length position:(NSInteger)position;

#### 引数

名称	説明	型
position	絶対位置（ドット数単位）	NSInteger
otherData	データ（テキストやコマンド）	NSData
bytes		const void *
length		NSUInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    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];
}
```

ApiFunctions.m を参照ください。

## 3.20 appendAlignment メソッド

位置揃え指定コマンドを生成し、commands プロパティに追加します。

### 宣言

- (void)appendAlignment:(SCBAlignmentPosition)position;
- (void)appendDataWithAlignment:(NSData \*)otherData position:(SCBAlignmentPosition)position;
- (void)appendBytesWithAlignment:(const void \*)bytes length:(NSUInteger)length position:(SCBAlignmentPosition)position;

### 引数

名称	説明	型
position	位置揃え <ul style="list-style-type: none"> <li>SCBAlignmentPositionLeft ... 左揃え</li> <li>SCBAlignmentPositionCenter ... 中心揃え</li> <li>SCBAlignmentPositionRight ... 右揃え</li> </ul>	SCBAlignmentPosition
otherData	データ (テキストやコマンド)	NSData
bytes		const void *
length		NSUInteger

### 戻り値

説明	型
-	-

### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.21 appendCutPaper メソッド

用紙カットコマンドを生成し、commands プロパティに追加します。

#### 宣言

- (void)appendCutPaper:(SCBCutPaperAction)action;

#### 引数

名称	説明	型
action	用紙カット <ul style="list-style-type: none"> <li>SCBCutPaperActionFullCut ... フルカット</li> <li>SCBCutPaperActionPartialCut ... パーシャルカット</li> <li>SCBCutPaperActionFullCutWithFeed ... 用紙送り付きフルカット</li> <li>SCBCutPaperActionPartialCutWithFeed ... 用紙送り付きパーシャルカット</li> </ul>	SCBCutPaperAction

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m を参照ください。



## 3.22 appendPeripheral メソッド

ペリフェラル駆動コマンドを生成し、commands プロパティに追加します。

### 宣言

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

### 引数

名称	説明	型
channel	ペリフェラルチャネル <ul style="list-style-type: none"> <li>SCBPeripheralChannelNo1 ... チャネル 1</li> <li>SCBPeripheralChannelNo2 ... チャネル 2</li> </ul>	SCBPeripheralChannel
time	駆動時間 (1 ミリ秒単位) ※チャネル 1 のみ有効	NSInteger

### 戻り値

説明	型
-	-

### 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];
}
```

ApiFunctions.m.m /各コマンド仕様書を参照ください。

### 3.23 appendSound メソッド

サウンド駆動コマンドを生成し、commands プロパティに追加します。

#### 宣言

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

#### 引数

名称	説明	型
channel	サウンドチャンネル <ul style="list-style-type: none"> <li>SCBSoundChannelNo1 ... チャンネル 1</li> <li>SCBSoundChannelNo2 ... チャンネル 2</li> </ul>	SCBSoundChannel
repeat	繰り返し回数	NSInteger

#### 戻り値

説明	型
-	-

#### 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 endDocument];

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

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.24 appendBarcodeData メソッド

バーコード印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
(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;
```

#### 引数

名称	説明	型
otherData	バーコードデータ	NSData
bytes		const void *
length		NSUInteger
symbology	バーコードシンボル <ul style="list-style-type: none"> <li>SCBBarcodeSymbologyUPCE ... UPC-E.</li> <li>SCBBarcodeSymbologyUPCA ... UPC-A.</li> <li>SCBBarcodeSymbologyJAN8 ... JAN/EAN8.</li> <li>SCBBarcodeSymbologyJAN13 ... JAN/EAN13.</li> <li>SCBBarcodeSymbologyCode39 ... Code39.</li> <li>SCBBarcodeSymbologyITF ... ITF.</li> <li>SCBBarcodeSymbologyCode128 ... Code128.</li> <li>SCBBarcodeSymbologyCode93 ... Code93.</li> <li>SCBBarcodeSymbologyCodabar ... Codabar.</li> <li>SCBBarcodeSymbologyNW7 ... NW7.</li> </ul>	SCBBarcodeSymbology
width	バーコード幅 <ul style="list-style-type: none"> <li>SCBBarcodeWidthMode1 ... モード 1</li> <li>SCBBarcodeWidthMode2 ... モード 2</li> <li>SCBBarcodeWidthMode3 ... モード 3</li> <li>SCBBarcodeWidthMode4 ... モード 4</li> <li>SCBBarcodeWidthMode5 ... モード 5</li> <li>SCBBarcodeWidthMode6 ... モード 6</li> <li>SCBBarcodeWidthMode7 ... モード 7</li> <li>SCBBarcodeWidthMode8 ... モード 8</li> <li>SCBBarcodeWidthMode9 ... モード 9</li> </ul>	SCBBarcodeWidth
height	バーコードの高さ（ドット数単位）	NSInteger
hri	バー下文字列 <ul style="list-style-type: none"> <li>YES ... 印刷あり</li> <li>NO ... 印刷なし</li> </ul>	BOOL

#### 戻り値

説明	型
-	-

### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherDataCode128 = [@"{B0123456789" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.25 appendBarcodeDataWithAbsolutePosition メソッド

絶対位置指定バーコード印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
position	絶対位置（ドット数単位）	NSInteger
otherData	バーコードデータ	NSData
bytes		const void *
length		NSUInteger
symbology	バーコードシンボル <ul style="list-style-type: none"> <li>SCBBarcodeSymbologyUPCE ... UPC-E.</li> <li>SCBBarcodeSymbologyUPCA ... UPC-A.</li> <li>SCBBarcodeSymbologyJAN8 ... JAN/EAN8.</li> <li>SCBBarcodeSymbologyJAN13 ... JAN/EAN13.</li> <li>SCBBarcodeSymbologyCode39 ... Code39.</li> <li>SCBBarcodeSymbologyITF ... ITF.</li> <li>SCBBarcodeSymbologyCode128 ... Code128.</li> <li>SCBBarcodeSymbologyCode93 ... Code93.</li> <li>SCBBarcodeSymbologyCodabar ... Codabar.</li> <li>SCBBarcodeSymbologyNW7 ... NW7.</li> </ul>	SCBBarcodeSymbology
width	バーコード幅 <ul style="list-style-type: none"> <li>SCBBarcodeWidthMode1 ... モード 1</li> <li>SCBBarcodeWidthMode2 ... モード 2</li> <li>SCBBarcodeWidthMode3 ... モード 3</li> <li>SCBBarcodeWidthMode4 ... モード 4</li> <li>SCBBarcodeWidthMode5 ... モード 5</li> <li>SCBBarcodeWidthMode6 ... モード 6</li> <li>SCBBarcodeWidthMode7 ... モード 7</li> <li>SCBBarcodeWidthMode8 ... モード 8</li> <li>SCBBarcodeWidthMode9 ... モード 9</li> </ul>	SCBBarcodeWidth
height	バーコードの高さ（ドット数単位）	NSInteger
hri	バー下文字列 <ul style="list-style-type: none"> <li>YES ... 印刷あり</li> <li>NO ... 印刷なし</li> </ul>	BOOL

## 戻り値

説明	型
-	-

## Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherDataCode128 = [@"{B0123456789" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.26 appendBarcodeDataWithAlignment メソッド

位置揃えバーコード印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
position	位置揃え指定 <ul style="list-style-type: none"> <li>SCBAlignmentPositionLeft ... 左揃え</li> <li>SCBAlignmentPositionCenter ... 中心揃え</li> <li>SCBAlignmentPositionRight ... 右揃え</li> </ul>	SCBAlignmentPosition
otherData		NSData
bytes	バーコードデータ	const void *
length		NSUInteger
symbology	バーコードシンボル <ul style="list-style-type: none"> <li>SCBBarcodeSymbologyUPCE ... UPC-E.</li> <li>SCBBarcodeSymbologyUPCA ... UPC-A.</li> <li>SCBBarcodeSymbologyJAN8 ... JAN/EAN8.</li> <li>SCBBarcodeSymbologyJAN13 ... JAN/EAN13.</li> <li>SCBBarcodeSymbologyCode39 ... Code39.</li> <li>SCBBarcodeSymbologyITF ... ITF.</li> <li>SCBBarcodeSymbologyCode128 ... Code128.</li> <li>SCBBarcodeSymbologyCode93 ... Code93.</li> <li>SCBBarcodeSymbologyCodabar ... Codabar.</li> <li>SCBBarcodeSymbologyNW7 ... NW7.</li> </ul>	SCBBarcodeSymbology
width	バーコード幅 <ul style="list-style-type: none"> <li>SCBBarcodeWidthMode1 ... モード 1</li> <li>SCBBarcodeWidthMode2 ... モード 2</li> <li>SCBBarcodeWidthMode3 ... モード 3</li> <li>SCBBarcodeWidthMode4 ... モード 4</li> <li>SCBBarcodeWidthMode5 ... モード 5</li> <li>SCBBarcodeWidthMode6 ... モード 6</li> <li>SCBBarcodeWidthMode7 ... モード 7</li> <li>SCBBarcodeWidthMode8 ... モード 8</li> <li>SCBBarcodeWidthMode9 ... モード 9</li> </ul>	SCBBarcodeWidth
height	バーコードの高さ (ドット数単位)	NSInteger

名称	説明	型
hri	バー下文字列 <ul style="list-style-type: none"> <li>YES ... 印刷あり</li> <li>NO ... 印刷なし</li> </ul>	BOOL

## 戻り値

説明	型
-	-

## Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherDataCode128 = [@"{B0123456789" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。



### 3.27 appendPdf417Data メソッド

PDF417 印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
otherData	PDF417 データ	NSData
bytes		const void *
length		NSUInteger
line	ライン数	NSInteger
column	カラム数	NSInteger
level	PDF417 ECC（セキュリティレベル） <ul style="list-style-type: none"> <li>SCBPdf417LevelECC0 ... セキュリティレベル 0</li> <li>SCBPdf417LevelECC1 ... セキュリティレベル 1</li> <li>SCBPdf417LevelECC2 ... セキュリティレベル 2</li> <li>SCBPdf417LevelECC3 ... セキュリティレベル 3</li> <li>SCBPdf417LevelECC4 ... セキュリティレベル 4</li> <li>SCBPdf417LevelECC5 ... セキュリティレベル 5</li> <li>SCBPdf417LevelECC6 ... セキュリティレベル 6</li> <li>SCBPdf417LevelECC7 ... セキュリティレベル 7</li> <li>SCBPdf417LevelECC8 ... セキュリティレベル 8</li> </ul>	SCBPdf417Level
module	モジュール X 方向サイズ	NSInteger
aspect	モジュールアスペクト比	NSInteger

#### 戻り値

説明	型
-	-

### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.28 appendPdf417DataWithAbsolutePosition メソッド

絶対位置指定 PDF417 印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
position	絶対位置（ドット数単位）	NSInteger
otherData	PDF417 データ	NSData
bytes		const void *
length		NSUInteger
line	ライン数	NSInteger
column	カラム数	NSInteger
level	PDF417 ECC（セキュリティレベル） <ul style="list-style-type: none"> <li>• SCBPdf417LevelECC0 ... セキュリティレベル 0</li> <li>• SCBPdf417LevelECC1 ... セキュリティレベル 1</li> <li>• SCBPdf417LevelECC2 ... セキュリティレベル 2</li> <li>• SCBPdf417LevelECC3 ... セキュリティレベル 3</li> <li>• SCBPdf417LevelECC4 ... セキュリティレベル 4</li> <li>• SCBPdf417LevelECC5 ... セキュリティレベル 5</li> <li>• SCBPdf417LevelECC6 ... セキュリティレベル 6</li> <li>• SCBPdf417LevelECC7 ... セキュリティレベル 7</li> <li>• SCBPdf417LevelECC8 ... セキュリティレベル 8</li> </ul>	SCBPdf417Level
module	モジュール X 方向サイズ	NSInteger
aspect	モジュールアスペクト比	NSInteger

#### 戻り値

説明	型
-	-

### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.29 appendPdf417DataWithAlignment メソッド

位置揃え PDF417 印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
position	位置揃え指定 <ul style="list-style-type: none"> <li>SCBAlignmentPositionLeft ... 左揃え</li> <li>SCBAlignmentPositionCenter ... 中心揃え</li> <li>SCBAlignmentPositionRight ... 右揃え</li> </ul>	SCBAlignmentPosition
otherData	PDF417 データ	NSData
bytes		const void *
length		NSUInteger
line	ライン数	NSInteger
column	カラム数	NSInteger
level	PDF417 ECC (セキュリティレベル) <ul style="list-style-type: none"> <li>SCBPdf417LevelECC0 ... セキュリティレベル 0</li> <li>SCBPdf417LevelECC1 ... セキュリティレベル 1</li> <li>SCBPdf417LevelECC2 ... セキュリティレベル 2</li> <li>SCBPdf417LevelECC3 ... セキュリティレベル 3</li> <li>SCBPdf417LevelECC4 ... セキュリティレベル 4</li> <li>SCBPdf417LevelECC5 ... セキュリティレベル 5</li> <li>SCBPdf417LevelECC6 ... セキュリティレベル 6</li> <li>SCBPdf417LevelECC7 ... セキュリティレベル 7</li> <li>SCBPdf417LevelECC8 ... セキュリティレベル 8</li> </ul>	SCBPdf417Level
module	モジュール X 方向サイズ	NSInteger
aspect	モジュールアスペクト比	NSInteger

#### 戻り値

説明	型
-	-

### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

*ApiFunctions.m* /各コマンド仕様書を参照ください。

### 3.30 appendQrCodeData メソッド

QR コード印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
otherData	QR コードデータ	NSData
bytes		const void *
length		NSUInteger
model	QR コードモデル • SCBQrCodeModelNo1 ... モデル 1 • SCBQrCodeModelNo2 ... モデル 2	SCBQrCodeModel
level	QR コード誤り訂正レベル • SCBQrCodeLevelL ... 誤り訂正レベル L • SCBQrCodeLevelM ... 誤り訂正レベル M • SCBQrCodeLevelQ ... 誤り訂正レベル Q • SCBQrCodeLevelH ... 誤り訂正レベル H	SCBQrCodeLevel
cell	QR コードセルサイズ	NSInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.31 appendQrCodeDataWithAbsolutePosition メソッド

絶対位置指定 QR コード印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
position	絶対位置（ドット数単位）	NSInteger
otherData	QR コードデータ	NSData
bytes		const void *
length		NSUInteger
model	QR コードモデル <ul style="list-style-type: none"> <li>SCBQrCodeModelNo1 ... モデル 1</li> <li>SCBQrCodeModelNo2 ... モデル 2</li> </ul>	SCBQrCodeModel
level	QR コード誤り訂正レベル <ul style="list-style-type: none"> <li>SCBQrCodeLevelL ... 誤り訂正レベル L</li> <li>SCBQrCodeLevelM ... 誤り訂正レベル M</li> <li>SCBQrCodeLevelQ ... 誤り訂正レベル Q</li> <li>SCBQrCodeLevelH ... 誤り訂正レベル H</li> </ul>	SCBQrCodeLevel
cell	QR コードセルサイズ	NSInteger

#### 戻り値

説明	型
-	-

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt 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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。



### 3.32 appendQrCodeDataWithAlignment メソッド

位置揃え QR コード印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
position	位置揃え指定 <ul style="list-style-type: none"> <li>SCBAlignmentPositionLeft ... 左揃え</li> <li>SCBAlignmentPositionCenter ... 中心揃え</li> <li>SCBAlignmentPositionRight ... 右揃え</li> </ul>	SCBAlignmentPosition
otherData	QR コードデータ	NSData
bytes		const void *
length		NSUInteger
model	QR コードモデル <ul style="list-style-type: none"> <li>SCBQrCodeModelNo1 ... モデル 1</li> <li>SCBQrCodeModelNo2 ... モデル 2</li> </ul>	SCBQrCodeModel
level	QR コード誤り訂正レベル <ul style="list-style-type: none"> <li>SCBQrCodeLevelL ... 誤り訂正レベル L</li> <li>SCBQrCodeLevelM ... 誤り訂正レベル M</li> <li>SCBQrCodeLevelQ ... 誤り訂正レベル Q</li> <li>SCBQrCodeLevelH ... 誤り訂正レベル H</li> </ul>	SCBQrCodeLevel
cell	QR コードセルサイズ	NSInteger

#### 戻り値

説明	型
-	-

### Example

```
+ (NSData *)createData:(StarIoExtEmulation)emulation {
    NSData *otherData = [@"Hello World." dataUsingEncoding:NSUTF8StringEncoding];

    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];
}
```

ApiFunctions.m /各コマンド仕様書を参照ください。

### 3.33 appendBitmap メソッド

ビットマップ印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
image	元となるビットマップオブジェクト	UIImage
diffusion	誤差拡散 <ul style="list-style-type: none"> <li>YES ... 誤差拡散を行う</li> <li>NO ... 誤差拡散を行わない</li> </ul>	BOOL
width	変換後のビットマップ幅（ドット数単位）	NSInteger
bothScale	width プロパティによる変換率に合わせた高さ変換 <ul style="list-style-type: none"> <li>YES ... 高さ変換を行う</li> <li>NO ... 高さ変換を行わない</li> </ul>	BOOL
rotation	変換後の回転 <ul style="list-style-type: none"> <li>SCBBitmapConverterRotationNormal ... 回転なし</li> <li>SCBBitmapConverterRotationRight90 ... 右 90 度回転</li> <li>SCBBitmapConverterRotationLeft90 ... 左 90 度回転</li> <li>SCBBitmapConverterRotationRotate180 ... 180 度回転</li> </ul>	SCBBitmapConverterRotation

#### 戻り値

説明	型
-	-

## Example

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

    ISCBBuilder *builder = [StarloExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:@"\nNormal\n" dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmap:starLogoImage diffusion:YES];

    [builder appendData:@"\nwidth:Full, bothScale:YES\n" dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmap:starLogoImage diffusion:YES width:width bothScale:YES];
    [builder appendData:@"\nwidth:Full, bothScale:NO\n" dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmap:starLogoImage diffusion:YES width:width bothScale:NO];

    [builder appendData:@"\nRotate180\n" dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmap:starLogoImage diffusion:YES rotation:SCBBitmapConverterRotationRotate180];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

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

*ApiFunctions.m* を参照ください。

### 3.34 appendBitmapWithAbsolutePosition メソッド

絶対位置指定ビットマップ印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
position	絶対位置（ドット数単位）	NSInteger
image	元となるビットマップオブジェクト	UIImage
diffusion	誤差拡散 <ul style="list-style-type: none"> <li>YES ... 誤差拡散を行う</li> <li>NO ... 誤差拡散を行わない</li> </ul>	BOOL
width	変換後のビットマップ幅（ドット数単位）	NSInteger
bothScale	width プロパティによる変換率に合わせた高さ変換 <ul style="list-style-type: none"> <li>YES ... 高さ変換を行う</li> <li>NO ... 高さ変換を行わない</li> </ul>	BOOL
rotation	変換後の回転 <ul style="list-style-type: none"> <li>SCBBitmapConverterRotationNormal ... 回転なし</li> <li>SCBBitmapConverterRotationRight90 ... 右 90 度回転</li> <li>SCBBitmapConverterRotationLeft90 ... 左 90 度回転</li> <li>SCBBitmapConverterRotationRotate180 ... 180 度回転</li> </ul>	SCBBitmapConverterRotation

#### 戻り値

説明	型
-	-

## Example

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

    ISCBBuilder *builder = [StarloExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:@"\nNormal, AbsolutePosition:40*\n" dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmapWithAbsolutePosition:starLogoImage diffusion:YES position:40];

    [builder appendData:@"\nwidth:Full, bothScale:YES, AbsolutePosition:40*\n"
dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmap:starLogoImage diffusion:YES width:width bothScale:YES position:40];
    [builder appendData:@"\nwidth:Full, bothScale:NO, AbsolutePosition:40*\n"
dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmap:starLogoImage diffusion:YES width:width bothScale:NO position:40];

    [builder appendData:@"\nRotate180, AbsolutePosition:40*\n" dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmapWithAbsolutePosition:starLogoImage diffusion:YES
rotation:SCBBitmapConverterRotationRotate180 position:40];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

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

*ApiFunctions.m* を参照ください。

### 3.35 appendBitmapWithAlignment メソッド

位置揃えビットマップ印刷コマンドを生成し、commands プロパティに追加します。

#### 宣言

```
- (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;
```

#### 引数

名称	説明	型
position	位置揃え指定 <ul style="list-style-type: none"> <li>SCBAlignmentPositionLeft ... 左揃え</li> <li>SCBAlignmentPositionCenter ... 中心揃え</li> <li>SCBAlignmentPositionRight ... 右揃え</li> </ul>	SCBAlignmentPosition
image	元となるビットマップオブジェクト	UIImage
diffusion	誤差拡散 <ul style="list-style-type: none"> <li>YES ... 誤差拡散を行う</li> <li>NO ... 誤差拡散を行わない</li> </ul>	BOOL
width	変換後のビットマップ幅（ドット数単位）	NSInteger
bothScale	width プロパティによる変換率に合わせた高さ変換 <ul style="list-style-type: none"> <li>YES ... 高さ変換を行う</li> <li>NO ... 高さ変換を行わない</li> </ul>	BOOL
rotation	変換後の回転 <ul style="list-style-type: none"> <li>SCBBitmapConverterRotationNormal ... 回転なし</li> <li>SCBBitmapConverterRotationRight90 ... 右 90 度回転</li> <li>SCBBitmapConverterRotationLeft90 ... 左 90 度回転</li> <li>SCBBitmapConverterRotationRotate180 ... 180 度回転</li> </ul>	SCBBitmapConverterRotation

#### 戻り値

説明	型
-	-

## Example

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

    ISCBBuilder *builder = [StarloExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:@"\nNormal, Alignment:Center*\n" dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmapWithAlignment:starLogoImage diffusion:YES position:SCBAlignmentPositionCenter];

    [builder appendData:@"\nwidth:Full, bothScale:YES, Alignment:Center*\n"
    dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmapWithAlignment:starLogoImage diffusion:YES width:width bothScale:YES
    position:SCBAlignmentPositionCenter];

    [builder appendData:@"\nwidth:Full, bothScale:NO, Alignment:Center*\n"
    dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmapWithAlignment:starLogoImage diffusion:YES width:width bothScale:NO
    position:SCBAlignmentPositionCenter];

    [builder appendData:@"\nRotate180, Alignment:Center*\n" dataUsingEncoding:NSUTF8StringEncoding];
    [builder appendBitmapWithAlignment:starLogoImage diffusion:YES
    rotation:SCBBitmapConverterRotationRotate180 position:SCBAlignmentPositionCenter];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

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

ApiFunctions.m を参照ください。



### 3.36 commands プロパティ

生成、追加されたコマンド列です。

#### 宣言

@property (nonatomic, readonly) NSMutableData \*commands;

#### 値

説明	型
生成、追加されたコマンド列	NSMutableData

#### Example

```
+ (NSData *)createData:(StarloExtEmulation)emulation {
    NSData *otherData = [@"Hello World.\n" dataUsingEncoding:NSUTF8StringEncoding];

    ISCBBuilder *builder = [StarloExt createCommandBuilder:emulation];

    [builder beginDocument];

    [builder appendData:otherData];

    [builder appendCutPaper:SCBCutPaperActionPartialCutWithFeed];

    [builder endDocument];

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

ApiFunctions.m を参照ください。

### 3.37 SCBInitializationType コンスタント

初期化指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBInitializationTypeCommand	コマンド初期化
SCBInitializationTypeReset	プリンタリセット
SCBInitializationTypeResetWithPrint	プリンタリセット（自己印字実行）

*ApiFunctions.m* を参照ください。

### 3.38 SCBFontStyleType コンスタント

フォントスタイル指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBFontStyleTypeA	フォント A（12 x 24 ドット） / 7 x 9 フォント（ハーフトット）
SCBFontStyleTypeB	フォント B（9 x 24 ドット） / 5 x 9 フォント（2P-1）

*ApiFunctions.m* / 各コマンド仕様書を参照ください。

### 3.39 SCBCodePageType コンスタント

コードページ指定定数。

#### 宣言

```
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
};
```

## 定数

名称	説明
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.

ApiFunctions.m / 各コマンド仕様書を参照ください。

### 3.40 SCBInternationalType コンスタント

国際文字指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBInternationalTypeUSA	USA
SCBInternationalTypeFrance	フランス
SCBInternationalTypeGermany	ドイツ
SCBInternationalTypeUK	イギリス
SCBInternationalTypeDenmark	デンマーク
SCBInternationalTypeSweden	スウェーデン
SCBInternationalTypeItaly	イタリア
SCBInternationalTypeSpain	スペイン
SCBInternationalTypeJapan	日本
SCBInternationalTypeNorway	ノルウェー
SCBInternationalTypeDenmark2	デンマーク II
SCBInternationalTypeSpain2	スペイン II
SCBInternationalTypeLatinAmerica	ラテンアメリカ
SCBInternationalTypeKorea	韓国
SCBInternationalTypeIreland	アイルランド
SCBInternationalTypeLegal	Legal

ApiFunctions.m / 各コマンド仕様書を参照ください。

### 3.41 SCBLogoSize コンスタント

ロゴサイズ指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBLogoSizeNormal	ノーマルモード
SCBLogoSizeDoubleWidth	横 2 倍モード
SCBLogoSizeDoubleHeight	縦 2 倍モード
SCBLogoSizeDoubleWidthDoubleHeight	縦 2 倍、横 2 倍モード

*ApiFunctions.m / 各コマンド仕様書を参照ください。*

### 3.42 SCBAlignmentPosition コンスタント

位置揃え指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBAlignmentPositionLeft	左揃え
SCBAlignmentPositionCenter	中心揃え
SCBAlignmentPositionRight	右揃え

*ApiFunctions.m / 各コマンド仕様書を参照ください。*

### 3.43 SCBCutPaperAction コンスタント

用紙カット指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBCutPaperActionFullCut	フルカット
SCBCutPaperActionPartialCut	パーシャルカット
SCBCutPaperActionFullCutWithFeed	用紙送り付きフルカット
SCBCutPaperActionPartialCutWithFeed	用紙送り付きパーシャルカット

*ApiFunctions.m* / 各コマンド仕様書を参照ください。

### 3.44 SCBPeripheralChannel コンスタント

ペリフェラルチャネル指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBPeripheralChannelNo1	チャンネル 1
SCBPeripheralChannelNo2	チャンネル 2

*ApiFunctions.m* を参照ください。

### 3.45 SCBSoundChannel コンスタント

サウンドチャンネル指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBSoundChannelNo1	チャンネル 1
SCBSoundChannelNo2	チャンネル 2

*ApiFunctions.m* を参照ください。

### 3.46 SCBBarcodeSymbology コンスタント

バーコードシンボル指定定数。

#### 宣言

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

#### 定数

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

*ApiFunctions.m* / 各コマンド仕様書を参照ください。



### 3.47 SCBBarcodeWidth コンスタント

バーコード幅指定定数。

#### 宣言

```
typedef NS_ENUM(NSUInteger, SCBBarcodeWidth) {
    SCBBarcodeWidthMode1,
    SCBBarcodeWidthMode2,
    SCBBarcodeWidthMode3,
    SCBBarcodeWidthMode4,
    SCBBarcodeWidthMode5,
    SCBBarcodeWidthMode6,
    SCBBarcodeWidthMode7,
    SCBBarcodeWidthMode8,
    SCBBarcodeWidthMode9
};
```

#### 定数

名称	説明
SCBBarcodeWidthMode1	モード 1
SCBBarcodeWidthMode2	モード 2
SCBBarcodeWidthMode3	モード 3
SCBBarcodeWidthMode4	モード 4
SCBBarcodeWidthMode5	モード 5
SCBBarcodeWidthMode6	モード 6
SCBBarcodeWidthMode7	モード 7
SCBBarcodeWidthMode8	モード 8
SCBBarcodeWidthMode9	モード 9

*ApiFunctions.m* / 各コマンド仕様書を参照ください。

### 3.48 SCBPdf417Level コンスタント

PDF417 ECC（セキュリティレベル）指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBPdf417LevelECC0	セキュリティレベル 0
SCBPdf417LevelECC1	セキュリティレベル 1
SCBPdf417LevelECC2	セキュリティレベル 2
SCBPdf417LevelECC3	セキュリティレベル 3
SCBPdf417LevelECC4	セキュリティレベル 4
SCBPdf417LevelECC5	セキュリティレベル 5
SCBPdf417LevelECC6	セキュリティレベル 6
SCBPdf417LevelECC7	セキュリティレベル 7
SCBPdf417LevelECC8	セキュリティレベル 8

*ApiFunctions.m / 各コマンド仕様書を参照ください。*

### 3.49 SCBQRCodeModel コンスタント

QR コードモデル指定定数

#### 宣言

```
typedef NS_ENUM(NSUInteger, SCBQRCodeModel) {
    SCBQRCodeModelNo1,
    SCBQRCodeModelNo2
};
```

#### 定数

名称	説明
SCBQRCodeModelNo1	モデル 1
SCBQRCodeModelNo2	モデル 2

*ApiFunctions.m / 各コマンド仕様書を参照ください。*

### 3.50 SCBQRCodeLevel コンスタント

QR コード誤り訂正レベル指定定数。

#### 宣言

```
typedef NS_ENUM(NSUInteger, SCBQRCodeLevel) {
    SCBQRCodeLevelL,
    SCBQRCodeLevelM,
    SCBQRCodeLevelQ,
    SCBQRCodeLevelH
};
```

#### 定数

名称	説明
SCBQRCodeLevelL	誤り訂正レベル L
SCBQRCodeLevelM	誤り訂正レベル M
SCBQRCodeLevelQ	誤り訂正レベル Q
SCBQRCodeLevelH	誤り訂正レベル H

*ApiFunctions.m* / 各コマンド仕様書を参照ください。

### 3.51 SCBBitmapConverterRotation コンスタント

ビットマップ回転指定定数。

#### 宣言

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

#### 定数

名称	説明
SCBBitmapConverterRotationNormal	回転なし
SCBBitmapConverterRotationRight90	右 90 度回転
SCBBitmapConverterRotationLeft90	左 90 度回転
SCBBitmapConverterRotationRotate180	180 度回転

*ApiFunctions.m* を参照ください。

## 4 StarIoExtManager class included in the StarIO\_Extension.framework

### メソッド

名称	説明
initWithType	StarIoExtManager を初期化します。
connect	マネージメントを開始します。
disconnect	マネージメントを停止します。

### プロパティ

名称	説明
port	StarIoPort オブジェクト。* readonly
lock	port プロパティによる通信権利の排他制御オブジェクト。* readonly
delegate	StarIoExtManager のデリゲートオブジェクト。
printerStatus	プリンタステータス。* readonly
printerPaperStatus	プリンタ用紙ステータス。* readonly
printerCoverStatus	プリンタカバーステータス。* readonly
cashDrawerStatus	キャッシュドロアーステータス。* readonly
barcodeReaderStatus	バーコードリーダーステータス。* readonly
cashDrawerOpenActiveHigh	キャッシュドロア開閉センサのモード。

### コンスタント

名称	説明
StarIoExtManagerType	マネージャタイプ定数。
StarIoExtManagerPrinterStatus	プリンタステータス定数。
StarIoExtManagerPrinterPaperStatus	プリンタ用紙ステータス定数。
StarIoExtManagerPrinterCoverStatus	プリンタカバーステータス定数。
StarIoExtManagerCashDrawerStatus	キャッシュドロアーステータス定数。
StarIoExtManagerBarcodeReaderStatus	バーコードリーダーステータス定数。

## 4.1 initWithType メソッド

StarIoExtManager を初期化します。

### 宣言

```
-(id)initWithType:(StarIoExtManagerType)type portName:(NSString *)portName
portSettings:(NSString *)portSettings ioTimeoutMillis:(NSUInteger)ioTimeoutMillis;
```

### 引数

名称	説明	型
type	マネージャタイプ <ul style="list-style-type: none"> <li>StarIoExtManagerTypeStandard ... プリンタ、キャッシュドロワのマネージメント</li> <li>StarIoExtManagerTypeWithBarcodeReader ... プリンタ、キャッシュドロワ、バーコードリーダーのマネージメント</li> <li>StarIoExtManagerTypeOnlyBarcodeReader ... バーコードリーダーのマネージメント</li> </ul>	StarIoExtManagerType
portName	プリンタポート名	NSString
portSettings	ポートセッティング	NSString
ioTimeoutMillis	内部制御および API のタイムアウト値	NSUInteger

### 戻り値

説明	型
StarIoExtManager オブジェクト	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;
}
```

PrinterExtViewController.m / CashDrawerExtViewController.m / BarcodeReaderExtViewController.m / CombinationExtViewController.m を参照ください。

◆portName、portSettings、ioTimeoutMills 詳細を知りたい場合、StarIO iOS SDK ドキュメントを参照してください。

## 4.2 connect メソッド

マネージメントを開始します。

### 宣言

- (BOOL)connect;

### 引数

名称	説明	型
-	-	-

### 戻り値

説明	型
リザルト	
<ul style="list-style-type: none"> <li>YES ... 成功</li> <li>NO ... 失敗</li> </ul>	BOOL

### Example

```
- (void)viewWillAppear:(BOOL)animated {
    [super viewWillAppear:animated];

    if (_starloExtManager.port != nil) {
        [_starloExtManager disconnect];
    }

    if ([_starloExtManager connect] == NO) {
        UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Fail to Open Port."
                                                             message:@""
                                                             delegate:self
                                                             cancelButtonTitle:@"OK"
                                                             otherButtonTitles:nil];

        [alert show];
    }
}
```

*PrinterExtViewController.m / CashDrawerExtViewController.m / BarcodeReaderExtViewController.m / CombinationExtViewController.m* を参照ください。

### 4.3 disconnect メソッド

マネージメントを停止します。

#### 宣言

- (BOOL)disconnect;

#### 引数

名称	説明	型
-	-	-

#### 戻り値

説明	型
リザルト <ul style="list-style-type: none"> <li>YES ... 成功</li> <li>NO ... 失敗</li> </ul>	BOOL

#### Example

```
- (void)viewWillDisappear:(BOOL)animated {
    [super viewWillDisappear:animated];

    [starIoExtManager disconnect];
}
```

*PrinterExtViewController.m / CashDrawerExtViewController.m / BarcodeReaderExtViewController.m / CombinationExtViewController.m* を参照ください。

### 4.4 port プロパティ

StarIOPort オブジェクト。\* *readonly*

#### 宣言

@property (readonly, nonatomic) SMPort \*port;

◆StarIO オブジェクトの詳細を知りたい場合、StarIO iOS SDK ドキュメントを参照してください。

## 4.5 lock プロパティ

port プロパティによる通信権利の排他制御オブジェクト。\* *readonly*

### 宣言

@property (readonly, nonatomic) NSRecursiveLock \*lock;

### Example

```
- (IBAction)touchUpInsidePrintButton:(id)sender {
    ...

    [_starloExtManager.lock lock];

    [Communication sendCommands:commands port:[_starloExtManager port]];

    [_starloExtManager.lock unlock];

    ...
}
```

*PrinterExtViewController.m / CashDrawerExtViewController.m / CombinationExtViewController.m* を参照ください。

## 4.6 delegate プロパティ

StarloExtManager のデリゲートオブジェクト。

### 宣言

@property (weak, nonatomic) id<StarloExtManagerDelegate> delegate;

### Example

```
- (void)viewDidLoad {
    [super viewDidLoad];
    // Do any additional setup after loading the view.

    _starloExtManager = [[StarloExtManager alloc] initWithType:StarloExtManagerTypeOnlyBarcodeReader
        portName:[AppDelegate getPortName]
        portSettings:[AppDelegate getPortSettings]
        ioTimeoutMillis:10000]; // 10000mS!!!

    _starloExtManager.delegate = self;
}
```

*PrinterExtViewController.m / CashDrawerExtViewController.m / BarcodeReaderExtViewController.m / CombinationExtViewController.m* を参照ください。



## 4.7 printerOnlineStatus プロパティ

プリンタオンラインステータス。\* *readonly*

### 宣言

@property (readonly, nonatomic) StarloExtManagerStatus printerOnlineStatus;

### 値

説明	型
プリンタステータス <ul style="list-style-type: none"> <li>StarloExtManagerPrinterStatusInvalid ... 不定</li> <li>StarloExtManagerPrinterStatusImpossible ... プリンタ使用不能</li> <li>StarloExtManagerPrinterStatusOnline ... プリンタオンライン検出</li> <li>StarloExtManagerPrinterStatusOffline ... プリンタオフライン検出</li> </ul>	StarloExtManagerStatus

## 4.8 printerPaperStatus プロパティ

プリンタ用紙ステータス。\* *readonly*

### 宣言

@property (readonly, nonatomic) StarloExtManagerPrinterPaperStatus printerPaperStatus;

### 値

説明	型
プリンタ用紙ステータス <ul style="list-style-type: none"> <li>StarloExtManagerPrinterPaperStatusInvalid ... 不定</li> <li>StarloExtManagerPrinterPaperStatusImpossible ... 機器使用不能</li> <li>StarloExtManagerPrinterPaperStatusReady ... プリンタ用紙レディ検出</li> <li>StarloExtManagerPrinterPaperStatusNearEmpty ... プリンタ用紙ニアエンド検出</li> <li>StarloExtManagerPrinterPaperStatusEmpty ... プリンタ用紙エンpty検出</li> </ul>	StarloExtManagerPrinterPaperStatus

## 4.9 printerCoverStatus プロパティ

プリンタカバーステータス。\* *readonly*

### 宣言

@property (readonly, nonatomic) StarloExtManagerPrinterCoverStatus printerCoverStatus;

### 値

説明	型
プリンタカバーステータス <ul style="list-style-type: none"> <li>StarloExtManagerPrinterCoverStatusInvalid ... 不定</li> <li>StarloExtManagerPrinterCoverStatusImpossible ... 機器使用不能</li> <li>StarloExtManagerPrinterCoverStatusOpen ... プリンタカバーオープン検出</li> <li>StarloExtManagerPrinterCoverStatusClose ... プリンタカバークローズ検出</li> </ul>	StarloExtManagerPrinterCoverStatus

## 4.10 cashDrawerStatus プロパティ

キャッシュドローステータス。\* *readonly*

### 宣言

@property (readonly, nonatomic) StarloExtManagerCashDrawerStatus cashDrawerStatus;

### 値

説明	型
キャッシュドローステータス <ul style="list-style-type: none"> <li>StarloExtManagerCashDrawerStatusInvalid ... 不定</li> <li>StarloExtManagerCashDrawerStatusImpossible ... 機器使用不能</li> <li>StarloExtManagerCashDrawerStatusOpen ... キャッシュドロアーオープン検出</li> <li>StarloExtManagerCashDrawerStatusClose ... キャッシュドロアークローズ検出</li> </ul>	StarloExtManagerCashDrawerStatus

## 4.11 barcodeReaderStatus プロパティ

バーコードリーダーステータス。\* *readonly*

### 宣言

@property (readonly, nonatomic) StarloExtManagerBarcodeReaderStatus barcodeReaderStatus;

### 値

説明	型
バーコードリーダーステータス <ul style="list-style-type: none"> <li>StarloExtManagerBarcodeReaderStatusInvalid ... 不定</li> <li>StarloExtManagerBarcodeReaderStatusImpossible ... 機器使用不能</li> <li>StarloExtManagerBarcodeReaderStatusConnect ... バーコードリーダー接続検出</li> <li>StarloExtManagerBarcodeReaderStatusDisconnect ... バーコードリーダー切断検出</li> </ul>	StarloExtManagerBarcodeReaderStatus

## 4.12 cashDrawerOpenActiveHigh プロパティ

キャッシュドロアー開閉センサのモード。

### 宣言

@property (nonatomic) BOOL cashDrawerOpenActiveHigh;

### 値

説明	型
キャッシュドロアー開閉センサモード <ul style="list-style-type: none"> <li>YES ... High アクティブ</li> <li>NO ... Low アクティブ</li> </ul>	BOOL

## 4.13 StarloExtManagerType コンスタント

マネージャタイプ定数。

### 宣言

```
typedef NS_ENUM(NSUInteger, StarloExtManagerType) {
    StarloExtManagerTypeStandard = 0,
    StarloExtManagerTypeWithBarcodeReader,
    StarloExtManagerTypeOnlyBarcodeReader,
};
```

### 定数

名称	説明
StarloExtManagerTypeStandard	プリンタ、キャッシュドロワのマネージメント
StarloExtManagerTypeWithBarcodeReader	プリンタ、キャッシュドロワ、バーコードリーダーのマネージメント
StarloExtManagerTypeOnlyBarcodeReader	バーコードリーダーのマネージメント

## 4.14 StarloExtManagerPrinterStatus コンスタント

プリンタステータス定数。

### 宣言

```
typedef NS_ENUM(NSUInteger, StarloExtManagerPrinterStatus) {
    StarloExtManagerPrinterStatusInvalid = 0,
    StarloExtManagerPrinterStatusImpossible,
    StarloExtManagerPrinterStatusOnline,
    StarloExtManagerPrinterStatusOffline
};
```

### 定数

名称	説明
StarloExtManagerPrinterStatusInvalid	不定
StarloExtManagerPrinterStatusImpossible	プリンタ使用不能
StarloExtManagerPrinterStatusPrinterOnline	プリンタオンライン検出
StarloExtManagerPrinterStatusPrinterOffline	プリンタオフライン検出

## 4.15 StarloExtManagerPrinterPaperStatus コンスタント

プリンタ用紙ステータス定数。

### 宣言

```
typedef NS_ENUM(NSInteger, StarloExtManagerPrinterPaperStatus) {
    StarloExtManagerPrinterPaperStatusInvalid = 0,
    StarloExtManagerPrinterPaperStatusImpossible,
    StarloExtManagerPrinterPaperStatusReady,
    StarloExtManagerPrinterPaperStatusNearEmpty,
    StarloExtManagerPrinterPaperStatusEmpty
};
```

### 定数

名称	説明
StarloExtManagerPrinterPaperStatusInvalid	不定
StarloExtManagerPrinterPaperStatusImpossible	プリンタ使用不能
StarloExtManagerPrinterPaperStatusReady	プリンタ用紙レディ検出
StarloExtManagerPrinterPaperStatusNearEmpty	プリンタ用紙ニアエンド検出
StarloExtManagerPrinterPaperStatusEmpty	プリンタ用紙エンプティ検出

## 4.16 StarloExtManagerPrinterCoverStatus コンスタント

プリンタカバーステータス定数。

### 宣言

```
typedef NS_ENUM(NSInteger, StarloExtManagerPrinterCoverStatus) {
    StarloExtManagerPrinterCoverStatusInvalid = 0,
    StarloExtManagerPrinterCoverStatusImpossible,
    StarloExtManagerPrinterCoverStatusOpen,
    StarloExtManagerPrinterCoverStatusClose
};
```

### 定数

名称	説明
StarloExtManagerPrinterCoverStatusInvalid	不定
StarloExtManagerPrinterCoverStatusImpossible	プリンタ使用不能
StarloExtManagerPrinterCoverStatusOpen	プリンタカバーオープン検出
StarloExtManagerPrinterCoverStatusClose	プリンタカバークローズ検出

## 4.17 StarloExtManagerCashDrawerStatus コンスタント

キャッシュドロアステータス定数。

### 宣言

```
typedef NS_ENUM(NSInteger, StarloExtManagerCashDrawerStatus) {
    StarloExtManagerCashDrawerStatusInvalid = 0,
    StarloExtManagerCashDrawerStatusImpossible,
    StarloExtManagerCashDrawerStatusOpen,
    StarloExtManagerCashDrawerStatusClose
};
```

### 定数

名称	説明
StarloExtManagerCashDrawerStatusInvalid	不定
StarloExtManagerCashDrawerStatusImpossible	キャッシュドロア使用不能
StarloExtManagerCashDrawerStatusOpen	キャッシュドロアオープン検出
StarloExtManagerCashDrawerStatusClose	キャッシュドロアクローズ検出

## 4.18 StarloExtManagerBarcodeReaderStatus コンスタント

バーコードリーダーステータス定数。

### 宣言

```
typedef NS_ENUM(NSInteger, StarloExtManagerBarcodeReaderStatus) {
    StarloExtManagerBarcodeReaderStatusInvalid = 0,
    StarloExtManagerBarcodeReaderStatusImpossible,
    StarloExtManagerBarcodeReaderStatusConnect,
    StarloExtManagerBarcodeReaderStatusDisconnect
};
```

### 定数

名称	説明
StarloExtManagerBarcodeReaderStatusInvalid	不定
StarloExtManagerBarcodeReaderStatusImpossible	バーコードリーダー使用不能
StarloExtManagerBarcodeReaderStatusConnect	バーコードリーダー接続検出
StarloExtManagerBarcodeReaderStatusDisconnect	バーコードリーダー切断検出

## 5 StarloExtManagerDelegate class included in the StarIO\_Extension.framework

### メソッド

名称	説明
didPrinterImpossible	プリンタ使用不能を通知。 * <i>optional</i>
didPrinterOnline	プリンタオンラインを通知。 * <i>optional</i>
didPrinterOffline	プリンタオフラインを通知。 * <i>optional</i>
didPrinterPaperReady	プリンタ用紙レディを通知。 * <i>optional</i>
didPrinterPaperNearEmpty	プリンタ用紙ニアエンプティを通知。 * <i>optional</i>
didPrinterPaperEmpty	プリンタ用紙エンプティを通知。 * <i>optional</i>
didPrinterCoverOpen	プリンタカバーオープンを通知。 * <i>optional</i>
didPrinterCoverClose	プリンタカバークローズを通知。 * <i>optional</i>
didCashDrawerOpen	キャッシュドロアーオープンを通知。 * <i>optional</i>
didCashDrawerClose	キャッシュドロアークローズを通知。 * <i>optional</i>
didBarcodeReaderImpossible	バーコードリーダー使用不能を通知。 * <i>optional</i>
didBarcodeReaderConnect	バーコードリーダー接続を通知。 * <i>optional</i>
didBarcodeReaderDisconnect	バーコードリーダー切断を通知。 * <i>optional</i>
didBarcodeDataReceive	バーコードデータ受信を通知。 * <i>optional</i>
didAccessoryConnectSuccess	Bluetooth アクセサリー接続成功を通知。 * <i>optional</i>
didAccessoryConnectFailure	Bluetooth アクセサリー接続失敗を通知。 * <i>optional</i>
didAccessoryDisconnect	Bluetooth アクセサリー切断を通知。 * <i>optional</i>
didStatusUpdate	ステータスアップデートを通知。 * <i>optional</i>

### 5.1 didPrinterImpossible メソッド

プリンタ使用不能を通知。 \* *optional*

#### 宣言

- (void)didPrinterImpossible;
- (void)didPrinterImpossible:(StarloExtManager \*)manager;

#### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

#### 戻り値

説明	型
-	-

#### Example

```
- (void)didPrinterImpossible:(StarloExtManager *)manager {
    _commentLabel.text = @"Printer Impossible.";

    _commentLabel.textColor = [UIColor redColor];
}
```

PrinterExtViewController.m / CashDrawerExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.2 didPrinterOnline メソッド

プリンタオンラインを通知。\* optional

### 宣言

- (void)didPrinterOnline;
- (void)didPrinterOnline:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
-(void)didPrinterOnline:(StarloExtManager *)manager {
    _commentLabel.text = @"Printer Online.";
    _commentLabel.textColor = [UIColor blueColor];
}
```

PrinterExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.3 didPrinterOffline メソッド

プリンタオフラインを通知。\* optional

### 宣言

- (void)didPrinterOffline;
- (void)didPrinterOffline:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
-(void)didPrinterOffline:(StarloExtManager *)manager {
    _commentLabel.text = @"Printer Offline.";
    _commentLabel.textColor = [UIColor redColor];
}
```

PrinterExtViewController.m / CombinationExtViewController.m を参照ください。



## 5.4 didPrinterPaperReady メソッド

プリンタ用紙レディを通知。\* optional

### 宣言

- (void)didPrinterPaperReady;
- (void)didPrinterPaperReady:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didPrinterPaperReady:(StarloExtManager *)manager {
    _commentLabel.text = @"Printer Paper Ready.";
    _commentLabel.textColor = [UIColor blueColor];
}
```

PrinterExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.5 didPrinterPaperNearEmpty メソッド

プリンタ用紙ニアエンプティを通知。\* optional

### 宣言

- (void)didPrinterPaperNearEmpty;
- (void)didPrinterPaperNearEmpty:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didPrinterPaperNearEmpty:(StarloExtManager *)manager {
    _commentLabel.text = @"Printer Paper Near Empty.";
    _commentLabel.textColor = [UIColor orangeColor];
}
```

PrinterExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.6 didPrinterPaperEmpty メソッド

プリンタ用紙エンプティを通知。\* *optional*

### 宣言

- (void)didPrinterPaperEmpty;
- (void)didPrinterPaperEmpty:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didPrinterPaperEmpty:(StarloExtManager *)manager {
    _commentLabel.text = @"Printer Paper Empty.";
    _commentLabel.textColor = [UIColor redColor];
}
```

PrinterExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.7 didPrinterCoverOpen メソッド

プリンタカバーオープンを通知。\* *optional*

### 宣言

- (void)didPrinterCoverOpen;
- (void)didPrinterCoverOpen:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didPrinterCoverOpen:(StarloExtManager *)manager {
    _commentLabel.text = @"Printer Cover Open.";
    _commentLabel.textColor = [UIColor redColor];
}
```

PrinterExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.8 didPrinterCoverClose メソッド

プリンタカバークローズを通知。\* *optional*

### 宣言

- (void)didPrinterCoverClose;
- (void)didPrinterCoverClose:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didPrinterCoverClose:(StarloExtManager *)manager {
    _commentLabel.text = @"Printer Cover Close.";
    _commentLabel.textColor = [UIColor blueColor];
}
```

PrinterExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.9 didCashDrawerOpen メソッド

キャッシュドロアーオープンを通知。\* *optional*

### 宣言

- (void)didCashDrawerOpen;
- (void)didCashDrawerOpen:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didCashDrawerOpen:(StarloExtManager *)manager {
    _commentLabel.text = @"Cash Drawer Open.";
    // _commentLabel.textColor = [UIColor redColor];
    _commentLabel.textColor = [UIColor magentaColor];
}
```

CashDrawerExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.10 didCashDrawerClose メソッド

キャッシュドロアークローズを通知。\* optional

### 宣言

- (void)didCashDrawerClose;
- (void)didCashDrawerClose:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didCashDrawerClose:(StarloExtManager *)manager {
    _commentLabel.text = @"Cash Drawer Close.";
    _commentLabel.textColor = [UIColor blueColor];
}
```

CashDrawerExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.11 didBarcodeReaderImpossible メソッド

バーコードリーダー使用不能を通知。\* optional

### 宣言

- (void)didBarcodeReaderImpossible;
- (void)didBarcodeReaderImpossible:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didBarcodeReaderImpossible:(StarloExtManager *)manager {
    _commentLabel.text = @"Barcode Reader Impossible.";
    _commentLabel.textColor = [UIColor redColor];
}
```

BarcodeReaderExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.12 didBarcodeReaderConnect メソッド

バーコードリーダー接続を通知。\* *optional*

### 宣言

- (void)didBarcodeReaderConnect;
- (void)didBarcodeReaderConnect:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didBarcodeReaderConnect:(StarloExtManager *)manager {
    _commentLabel.text = @"Barcode Reader Connect.";
    _commentLabel.textColor = [UIColor blueColor];
}
```

BarcodeReaderExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.13 didBarcodeReaderDisconnect メソッド

バーコードリーダー切断を通知。\* *optional*

### 宣言

- (void)didBarcodeReaderDisconnect;
- (void)didBarcodeReaderDisconnect:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didBarcodeReaderDisconnect:(StarloExtManager *)manager {
    _commentLabel.text = @"Barcode Reader Disconnect.";
    _commentLabel.textColor = [UIColor redColor];
}
```

BarcodeReaderExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.14 didBarcodeDataReceive メソッド

バーコードデータ受信を通知。\* optional

### 宣言

- (void)didBarcodeDataReceive:(NSData \*)data;
- (void)didBarcodeDataReceive:(StarloExtManager \*)manager data:(NSData \*)data;

### 引数

名称	説明	型
data	受信したバーコードデータ	NSData
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didBarcodeDataReceive:(StarloExtManager *)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:@""];
        }
    }
}
```

BarcodeReaderExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.15 didAccessoryConnectSuccess メソッド

Bluetooth アクセサリー接続成功を通知。\* optional

### 宣言

- (void)didAccessoryConnectSuccess;
- (void)didAccessoryConnectSuccess:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didAccessoryConnectSuccess:(StarloExtManager *)manager {
    _commentLabel.text = @"Accessory Connect Success.";
    _commentLabel.textColor = [UIColor blueColor];
}
```

PrinterExtViewController.m / CashDrawerExtViewController.m / BarcodeReaderExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.16 didAccessoryConnectFailure メソッド

Bluetooth アクセサリー接続失敗を通知。\* optional

### 宣言

- (void)didAccessoryConnectFailure;
- (void)didAccessoryConnectFailure:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didAccessoryConnectFailure:(StarloExtManager *)manager {
    _commentLabel.text = @"Accessory Connect Failure.";
    _commentLabel.textColor = [UIColor redColor];
}
```

PrinterExtViewController.m / CashDrawerExtViewController.m / BarcodeReaderExtViewController.m / CombinationExtViewController.m を参照ください。

## 5.17 didAccessoryDisconnect メソッド

Bluetooth アクセサリー切断を通知。\* *optional*

### 宣言

- (void)didAccessoryDisconnect;
- (void)didAccessoryDisconnect:(StarloExtManager \*)manager;

### 引数

名称	説明	型
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didAccessoryDisconnect:(StarloExtManager *)manager {
    _commentLabel.text = @"Accessory Disconnect.";
    _commentLabel.textColor = [UIColor redColor];
}
```

PrinterExtViewController.m / CashDrawerExtViewController.m / BarcodeReaderExtViewController.m / CombinationExtViewController.m を参照ください。



## 5.18 didStatusUpdate メソッド

ステータスアップデートを通知。\* *optional*

### 宣言

- (void)didStatusUpdate:(NSString \*)status;
- (void)didStatusUpdate:(StarloExtManager \*)manager status:(NSString \*)status;

### 引数

名称	説明	型
status	<p>アップデートステータス</p> <p>自動ステータス（※）の3バイト目～6バイト目を16進数文字列化したもの（例：“28000000”）</p> <p>-ステータス通知タイミング-</p> <ul style="list-style-type: none"> <li>• ステータス変化時</li> <li>• ステータス未変化時、5分毎</li> </ul> <p>※STAR Line Mode コマンド仕様書もしくはStarPRNT コマンド仕様書参照ください</p>	NSString
manager	呼び出し元の StarloExtManager オブジェクト	StarloExtManager

### 戻り値

説明	型
-	-

### Example

```
- (void)didStatusUpdate:(StarloExtManager *)manager status:(NSString *)status {
    _commentLabel.text = status;
    _commentLabel.textColor = [UIColor greenColor];
}
```

AllReceiptsExtViewController.m を参照ください。

## 6 StarPRNT iOS SDK Sample

### 6.1 Communication

StarIO iOS SDK 相当の印刷データ送信の一例と StarIoExtManager オブジェクトを使用した印刷データ送信の一例が用意されています。

#### Example (StarIO iOS SDK 相当)

```
+ (BOOL)sendCommands:(NSData *)commands portName:(NSString *)portName portSettings:(NSString
*)portSettings timeout:(NSInteger)timeout {
    BOOL result = NO;

    ...

    SMPort *port = nil;

    @try {
        while (YES) {
            port = [SMPort getPort:portName :portSettings :(uint32_t) timeout];

            if (port == nil) {

                ...

                break;
            }

            StarPrinterStatus_2 printerStatus;

            [port beginCheckedBlock:&printerStatus :2];

            if (printerStatus.offline == SM_TRUE) {

                ...

                break;
            }

            NSDate *startDate = [NSDate date];

            uint32_t total = 0;

            while (total < commandLength) {
                uint32_t written = [port writePort:commandsBytes :total :commandLength - total];

                total += written;

                if ([NSDate date] timeIntervalSinceDate:startDate) >= 30.0 { // 30000mS!!!
                    break;
                }
            }

            if (total < commandLength) {

                ...

                break;
            }

            port.endCheckedBlockTimeoutMillis = 30000; // 30000mS!!!

            [port endCheckedBlock:&printerStatus :2];
        }
    }
}
```

```

        if (printerStatus.offline == SM_TRUE) {
            ...
            break;
        }

        result = YES;
        break;
    }
}
}catch (PortException *exc) {

    ...

}
@finally {
    if (port != nil) {
        [SMPort releasePort:port];
    }
}

...

return result;
}

```

### Example (StarIoExtManager オブジェクトを使用)

```

+ (BOOL)sendCommands:(NSData *)commands port:(SMPort *)port {
    BOOL result = NO;

    ...

    @try {
        while (YES) {
            ...

            StarPrinterStatus_2 printerStatus;

            [port beginCheckedBlock:&printerStatus :2];

            if (printerStatus.offline == SM_TRUE) {

                ...

                break;
            }

            NSDate *startDate = [NSDate date];

            uint32_t total = 0;

            while (total < commandLength) {
                uint32_t written = [port writePort:commandsBytes :total :commandLength - total];

                total += written;

                if ([NSDate date] timeIntervalSinceDate:startDate] >= 30.0) { // 30000mS!!!
                    break;
                }
            }
        }
    }
}

```

```

if (total < commandLength) {
    ...
    break;
}

port.endCheckedBlockTimeoutMillis = 30000; // 30000mS!!!
[port endCheckedBlock:&printerStatus :2];

if (printerStatus.offline == SM_TRUE) {
    ...
    break;
}

result = YES;
break;
}
}
@catch (PortException *exc) {
    ...
}

...

return result;
}

```

**StarIO.framework** 詳細を知りたい場合、StarIO iOS SDK ドキュメントを参照してください。

## 7 API expanded for Swift support included in the StarIO.framework

NSError によってメソッドの終了結果を知ることのできる API が用意されました。

### Swift 宣言

```
public func writePort(writeBuffer: UnsafePointer<UInt8>, _ offSet: UInt32, _ size: UInt32, _ error:
NSErrorPointer) -> UInt32

public func readPort(readBuffer: UnsafeMutablePointer<UInt8>, _ offSet: UInt32, _ size: UInt32, _ error:
NSErrorPointer) -> UInt32

public func getParsedStatus(starPrinterStatus: UnsafeMutablePointer<Void>, _ level: UInt32, _ error:
NSErrorPointer) -> UInt32

public func getFirmwareInformation(error: NSErrorPointer) -> [NSObject : AnyObject]!

public func getDipSwitchInformation(error: NSErrorPointer) -> [NSObject : AnyObject]!

public func getOnlineStatusWithError(error: NSErrorPointer) -> UInt32

public func beginCheckedBlock(starPrinterStatus: UnsafeMutablePointer<Void>, _ level: UInt32, _ error:
NSErrorPointer) -> UInt32

public func endCheckedBlock(starPrinterStatus: UnsafeMutablePointer<Void>, _ level: UInt32, _ error:
NSErrorPointer) -> UInt32
```

### Objective-c 宣言

```
- (u_int32_t)writePort:(u_int8_t const *)writeBuffer :(u_int32_t)offSet :(u_int32_t)size :(NSError **)error;
- (u_int32_t)readPort:(u_int8_t *)readBuffer :(u_int32_t)offSet :(u_int32_t)size :(NSError **)error;
- (NSUInteger)getParsedStatus:(void *)starPrinterStatus :(u_int32_t)level :(NSError **)error;
- (NSDictionary *)getFirmwareInformation:(NSError **)error;
- (NSDictionary *)getDipSwitchInformation:(NSError **)error;
- (NSUInteger)getOnlineStatus:(NSError **)error;
- (NSUInteger)beginCheckedBlock:(void *)starPrinterStatus :(u_int32_t)level :(NSError **)error;
- (NSUInteger)endCheckedBlock:(void *)starPrinterStatus :(u_int32_t)level :(NSError **)error;
```

**StarIO.framework 詳細を知りたい場合、StarIO iOS SDK ドキュメントを参照してください。**

## 8 SMCloudServices class included in the SMCloudServices.framework

### メソッド

名称	説明
showRegistrationView	スター精密クラウドサービス登録のビューを表示します。
isRegistered	アプリケーションがスター精密クラウドサービスに登録されているか確認します。

### 8.1 showRegistrationView メソッド

スター精密クラウドサービス登録のビューを表示します。



### 宣言

```
+ (void)showRegistrationView:(void (^)(BOOL isRegistered))completion;
```

### 引数

名称	説明	型
completion	ビューが閉じられた際、アプリケーションの登録状況を以下の引数によって捉えることができます。 -isRegistered- • YES ... 登録されている • NO ... 登録されていない	void (^)(BOOL isRegistered)

### 戻り値

説明	型
-	-

### Example

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

    [SMCloudServices showRegistrationView:^(BOOL isRegistration) {
        [_tableView reloadData];
    }];

    ...
}
```

AllReceiptsViewController.m を参照ください。

## 8.2 isRegistered メソッド

アプリケーションがスター精密クラウドサービスに登録されているか確認します。

### 宣言

```
+ (BOOL)isRegistered;
```

### 引数

名称	説明	型
-	-	-

### 戻り値

説明	型
リザルト <ul style="list-style-type: none"> <li>YES ... 登録されている</li> <li>NO ... 登録されていない</li> </ul>	BOOL

### Example

```
- (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath {
    ...

    BOOL userInteractionEnabled = YES;

    if ([SMCloudServices isRegistered] == NO) {
        userInteractionEnabled = NO;
    }

    ...

    return cell;
}
```

*AllReceiptsViewController.m* を参照ください。

## 9 SMCSAllReceipts class included in the SMCloudServices.framework

### メソッド

名称	説明
uploadBitmap	ビットマップをスター精密クラウドサービスへアップロードします。
updateStatus	スター精密クラウドサービス上のデバイスステータス情報をアップデートします。
generateAllReceipts	AllReceipts™ 利用のための印刷データを生成します。

### 9.1 モデル : SMCSAllReceipts class メソッド

各モデルにおいて機能するメソッドは以下のとおりです。

Function	Method	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	SP700
Upload	uploadBitmap	✓	✓	✓	✓	✓	✓	-	-	-	-	-	✓	✓	✓	✓	✓	✓	✓	-
Update	updateStatus	✓	✓	✓	✓	✓	✓	-	-	-	-	-	✓	✓	✓	✓	✓	✓	✓	-
Generate	generateAllReceipts	✓	✓	✓	✓	✓	✓	-	-	-	-	-	✓	✓	✓	✓	✓	✓	✓	-

-: 対象外



## 9.2 uploadBitmap メソッド

ビットマップ(UImage オブジェクト)をスター精密クラウドサービスへアップロードします。

### 宣言

```
+ (NSString *)uploadBitmap:(UImage *)image completion:(void (^)(NSInteger statusCode, NSError *error))completion;
```

### 引数

名称	説明	型
image	アップロードするビットマップ	UImage
completion	アップロードが完了した際、アップロード結果を以下の引数によって捉えることができます。  -statusCode- • HTTP ステータスコード  -error- • アップロードに失敗した際のエラー情報	void (^)(NSInteger statusCode, NSError *error)

### 戻り値

説明	型
アップロード URL	NSString

### Example

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

    ISCBBuilder *builder = [StarloExt 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];
}
```

*AllReceiptsFunctions.m* を参照ください。

### 9.3 updateStatus メソッド

スター精密クラウドサービス上のデバイスステータス情報をアップデートします。

#### 宣言

```
+ (void)updateStatus:(NSString *)status completion:(void (^)(NSInteger statusCode, NSError *error))completion;
```

#### 引数

名称	説明	型
status	<p>アップデートステータス</p> <p>自動ステータス（※）の3バイト目～6バイト目を16進数文字列化したもの（例：“28000000”）</p> <p>※STAR Line Mode コマンド仕様書もしくはStarPRNT コマンド仕様書参照ください</p>	NSString
completion	<p>アップデートが完了した際、アップデート結果を以下の引数によって捉えることができます。</p> <p>-statusCode-</p> <ul style="list-style-type: none"> <li>HTTP ステータスコード</li> </ul> <p>-error-</p> <ul style="list-style-type: none"> <li>アップデートに失敗した際のエラー情報</li> </ul>	void (^)(NSInteger statusCode, NSError *error)

#### 戻り値

説明	型
-	-

#### Example

```
- (void)didStatusUpdate:(StarIoExtManager *)manager status:(NSString *)status {
    [SMCSAllReceipts updateStatus:status completion:^(NSInteger statusCode, NSError *error) {
        ...
    }];
}
```

AllReceiptsExtViewController.m を参照ください。

## 9.4 generateAllReceipts メソッド

AllReceipts™ 利用のための印刷データを生成します。

### 宣言

```
+ (NSData *)generateAllReceipts:(NSString *)urlString emulation:(StarloExtEmulation)emulation
info:(BOOL)info qrCode:(BOOL)qrCode;
```

### 引数

名称	説明	型
urlString	アップロード URL	NSString
emulation	エミュレーション <ul style="list-style-type: none"> <li>StarloExtEmulationStarPRNT ... StarPRNT エミュレーション</li> <li>StarloExtEmulationStarLine ... STAR Line Mode エミュレーション</li> <li>StarloExtEmulationStarGraphic ... STAR Graphic Mode エミュレーション</li> <li>StarloExtEmulationEscPos ... ESC/POS エミュレーション</li> <li>StarloExtEmulationEscPosMobile ... ESC/POS Mobile エミュレーション</li> <li>StarloExtEmulationStarDotImpact ... STAR ドットインパクトエミュレーション</li> </ul>	StarloExtEmulation
info	インフォメーションロゴ <ul style="list-style-type: none"> <li>YES ... 印刷データに含める</li> <li>NO ... 印刷データに含めない</li> </ul>	BOOL
qrCode	QR コード <ul style="list-style-type: none"> <li>YES ... 印刷データに含める</li> <li>NO ... 印刷データに含めない</li> </ul>	BOOL

### 戻り値

説明	型
生成された印刷データ	NSData

### Example

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

    ISCBBuilder *builder = [StarloExt 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];
}
```

*AllReceiptsFunctions.m* を参照ください。

## Appendix A. AllReceipt™ 利用方法案内 (配布用)

無料の電子レシートサービス『AllReceipts™』について、詳しくは下記 URL をご参照ください。

[www.allreceipts.com](http://www.allreceipts.com)

下記サイトにて店舗アカウントを登録いただくだけで、その日からサービスの利用が可能です。

＜＜スター精密クラウドサービス 店舗アカウント登録サイト＞＞

[www.starcloudservices.com](http://www.starcloudservices.com)

### 1) デバイスの登録

ご登録いただいたスター精密クラウドサービスのアカウントでデバイスを登録します。

デバイスの登録をすると、スター精密クラウドサービスのダッシュボードから登録したデバイスの管理が可能となります。

2) あらかじめご登録いただいた、スター精密クラウドサービスアカウントのユーザー名（Eメール）とパスワードを入力して [登録] をタップします。



3) デバイスの登録が成功すると、接続したスター精密クラウドサービスのアカウント情報が表示されます。

