

OPN-2002i SERIES/OPN-3002i SERIES

DATA collector

iOS SDK ユーザーズガイド

改版履歴

資料管理番号：SI12057

発行番号：Rev.1.1

製品名：OPN-2002i Series / OPN-3002i Series

版	日付	変更箇所	内容
Rev.1.0	2012/10/30	—	新規作成
Rev.1.1	2013/02/08	表紙	OPN-3002 Series を追加
		3.1-3.4	OPN-3002 固有の設定項目を追加

はじめに

本書は、「OPN-2002i」「OPN-3002i」等 MFi 認証されている弊社データコレクタを利用した iOS 向けアプリケーションの開発に利用されることを目的としています。

データコレクタの操作方法や、iOS 開発で必要となる一般的な技術情報につきましては、それぞれの説明書などを参照してください。

本書内の解説は「OPN-3002i 固有設定」項目を除き、全て「OPN-2002i」「OPN-3002i」に共通となります。

本書の構成

1. 開発環境

本 SDK を利用できる条件や手順について、解説しています。

2. APIの利用

クラスライブラリの主な API について、概要と利用方法をサンプルコードと合わせて解説しています。

3. コマンドリファレンス

クラスライブラリの API を使って、データコレクタに送信できるコマンドの一覧です。

4. 配布

作成したアプリケーションの配布する際の注意点を解説しています。

API の詳細については、本書には記載していません。「OPNBluetoothKit API リファレンス」を参照してください。

目次

はじめに	2
本書の構成	2
1. 開発環境	2
2. APIの利用	2
3. コマンドリファレンス	2
4. 配布	2
目次	3
1. 開発環境	5
1.1 実行環境	5
1.2 開発環境でのSDK使用手順	5
■ frameworkを開発環境に取り込む	6
■ Info.plistにSupported external accessory protocolsを追加する	7
2. APIの利用	8
■ OPN2002iBluetoothServiceを生成・取得する	8
■ データコレクタと接続する	8
■ セッションをオープンする	9
■ セッションをクローズする	9
■ API標準のコマンドを実行する	9
■ 任意のコマンドを実行する	10
■ 通信エラーを検知する	10
■ データコレクタと接続中か確認する	11
3. コマンドリファレンス	12
3.1 デコーダ設定	12
3.2 Bluetooth設定	17
3.3 プリフィックス	23

3.4 サフィックス	25
3.5 文字列設定	26
4. 配布	32

1. 開発環境

1.1 実行環境

本 SDK は iOS4.3 以上を対象としています。

1.2 開発環境でのSDK使用手順

本 SDK のクラスライブラリを利用するには、以下の 2 点を対応する必要があります。

- `OPNBluetoothKit.framework` と `ExternalAccessory.framework` を開発環境に取り込む
- `Info.plist` に `Supported external accessory protocols` を追加し、`jp.opto.opnprotocol` を指定する

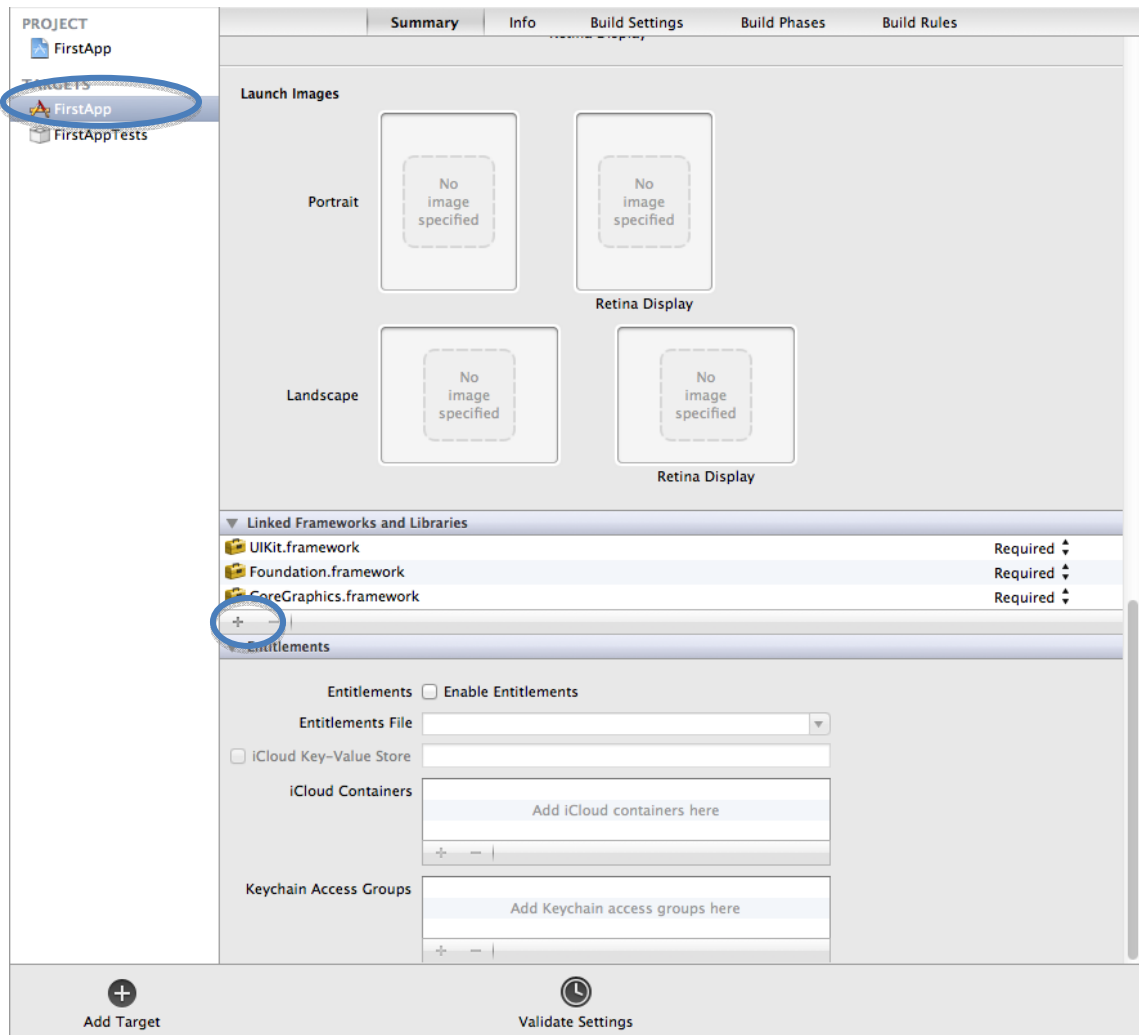
(前提)

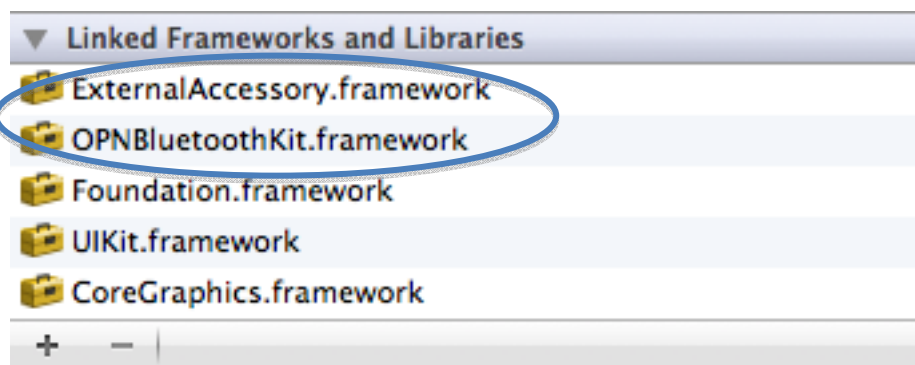
Xcode 上で iOS アプリケーションを開発するための一般的な環境は整っていて、Xcode プロジェクトが既に作成されている状態を前提とします。以下の例では、プロジェクト「FirstApp」が作成されている状態からの手順を解説します。

また、本書では、開発環境として Xcode 4.5.1 を使った場合の手順を解説します。

■frameworkを開発環境に取り込む

- ① FirstApp プロジェクトの TARGETS の Summary タブにある Link Binary With Libraries から本 SDK と ExternalAccessory.framework を追加します。





■Info.plistにSupported external accessory protocolsを追加する

- ① FirstApp プロジェクトにある FirstApp-Info.plist を開きます。

	Key	Type	Value
	Localization native development region	String	en
	Bundle display name	String	\$(PRODUCT_NAME)
	Executable file	String	\$(EXECUTABLE_NAME)
	Bundle identifier	String	jp.opto.{\$(PRODUCT_NAME)rfc1034identifier}
	InfoDictionary version	String	6.0
	Bundle name	String	\$(PRODUCT_NAME)
	Bundle OS Type code	String	APPL
	Bundle versions string, short	String	1.0
	Bundle creator OS Type code	String	???
	Bundle version	String	1.0
	Application requires iPhone environment	Boolean	YES
	Main storyboard file base name	String	MainStoryboard_iPhone
	Main storyboard file base name (iPad)	String	MainStoryboard_iPad
	Required device capabilities	Array	(1 item)
	Supported interface orientations	Array	(3 items)
	Supported interface orientations (iPad)	Array	(4 items)

- ② Supported external accessory protocols キーを追加します。

▶ Supported interface orientations	Array	(3 items)
▶ Supported interface orientations (iPad)	Array	(4 items)
▶ Supported external accessory protocols	Array	(1 item)

- ③ 値として jp.opto.opnprotocol を指定します。

▶ Supported interface orientations	Array	(3 items)
▶ Supported interface orientations (iPad)	Array	(4 items)
▼ Supported external accessory protocols	Array	(1 item)
Item 0	String	jp.opto.opnprotocol

2. APIの利用

主な API の使用方法について、サンプルコードを用いて解説します。

なお、フレームワークの使用には `import` が必要です。

```
#import <OPNBluetoothKit/OPN2002iBluetoothService.h>
```

■ OPN2002iBluetoothServiceを生成・取得する

```
OPN2002iBluetoothService *sessionController =  
    [OPN2002iBluetoothService sharedController];
```

データコレクタとの Bluetooth 通信は、全て OPN2002iBluetoothService クラスを介して行います。

OPN2002iBluetoothService の `sharedController()` メソッドを実行することで、インスタンスを生成、取得できます。

OPN2002iBluetoothService のインスタンスは実行中ただ 1 つしか存在しないため、まだインスタンスが生成されていなければ生成後返却され、既に生成されていればそのインスタンスが返却されます。

■ データコレクタと接続する

```
NSArray *accessoryList = [[NSMutableArray alloc] initWithArray:  
    [[EAAccessoryManager sharedAccessoryManager]  
        connectedAccessories]];  
  
if(accessoryList == nil || accessoryList.count == 0) return;  
EAAccessory *accessory = [accessoryList lastObject];  
  
OPN2002iBluetoothService *service =  
    [OPN2002iBluetoothService sharedController];  
  
NSArray *protocolStrings = [accessory protocolStrings];  
if(protocolStrings == nil || [protocolStrings count]==0) return;  
NSString *protocolString = [protocolStrings objectAtIndex:0];  
  
[service setupControllerForAccessory:accessory  
    withProtocolString:protocolString];
```

PN2002iBluetoothService の `setupControllerForAccessory ()` メソッドを実行

することで、データコレクタと接続することができます。

本サンプルでは、`setupControllerForAccessory()`メソッドの実行に必要な引数であるアクセサリとプロトコル文字列を取得し、接続を行っています。

■ セッションをオープンする

```
OPN2002iBluetoothService *sessionController =  
    [OPN2002iBluetoothService sharedController];  
[sessionController openSession];
```

接続済みの場合、`OPN2002iBluetoothService` の `openSession()`メソッドを実行することで、データコレクタとのセッションをオープンできます。

■ セッションをクローズする

```
OPN2002iBluetoothService *sessionController =  
    [OPN2002iBluetoothService sharedController];  
[sessionController closeSession];
```

`OPN2002iBluetoothService` の `closeSession()`メソッドを実行することで、データコレクタとのセッションをクローズできます。

■ API標準のコマンドを実行する

```
OPN2002iBluetoothService *service =  
    [OPN2002iBluetoothService sharedController];  
[service getFirmwareVersion:self];
```

データコレクタ関連のコマンドの中には、`OPN2002iBluetoothService` クラスのメソッドとして、提供されているものがあります。これらのコマンドを実行するためのメソッドは、通常、以下のような形をしています。

- (void)メソッド名:(id<`OPN2002iBluetoothServiceDelegate`>)delegate;

本サンプルでは、`OPN2002iBluetoothService` のインスタンスを取得し、`getFirmwareVersion()`メソッドの引数に自分自身をデリゲートとして渡しています。

■ 任意のコマンドを実行する

```
OPN2002iBluetoothService *service =  
    [OPN2002iBluetoothService sharedController];  
[service writeData:@"W8" dataCommand] target:self];
```

OPN2002iBluetoothService クラスのメソッドとして提供されていないコマンドを実行する場合は、OPN2002iBluetoothService の writeData() メソッドを実行します。writeData() メソッドの引数に、実行したいコマンド文字列とデリゲートを指定することで、コマンドが実行されます。

本サンプルでは、OPN2002iBluetoothService のインスタンスを取得し、writeData() メソッドの引数に、コマンド文字列の dataCommand と、自分自身をデリゲートとして、それぞれ渡しています。

利用できるコマンドについては、「3.5 文字列設定」を参照してください。

■ 通信エラーを検知する

```
@interface AppViewController <OPN2002iBluetoothServiceDelegate> {  
}
```

OPN2002iBluetoothService とデータコレクタとの通信結果を処理するためには、まず、通信結果を処理するクラスが OPN2002iBluetoothServiceDelegate プロトコルを採用する必要があります。

本サンプルでは、AppViewController クラスが当該プロトコルを採用しています。

```
OPN2002iBluetoothService *sessionController =  
    [OPN2002iBluetoothService sharedController];  
[service setDelegate:self];
```

次に、そのクラスを OPN2002iBluetoothService のインスタンスのデリゲートに指定します。

```

- (void)bluetoothService:(OPN2002iBluetoothService *)service
    withError:(NSError *)error
{
    NSLog(@"error: %@", error);
}

```

最後に、そのクラスで以下のメソッドを実行します。

```

- (void)bluetoothService:(OPNBluetoothService *)service
    withError:(NSError *)error;

```

本サンプルでは、検知したエラーの内容をログに出力しています。

■ データコレクタと接続中か確認する

```

isConnected =
    [[[OPN2002iBluetoothService
        sharedController] accessory] isConnected];

```

OPN2002iBluetoothService の accessory プロパティである EAAccessory の isConnected プロパティで、データコレクタと接続中か確認できます。YES なら接続中です。

3. コマンドリファレンス

3.1 デコーダ設定

設定項目	パラメータ	設定値	コマンド
UPC-A/E 読み取り	許可	1	R1
	禁止	0	[X4B
UPC-A/E Addon2 読み取り	許可	1	R2
	禁止	0	[X4C
UPC-A/E Addon5 読み取り	許可	1	R3
	禁止	0	[X4D
JAN/EAN-13/8 読み取り	許可	1	R4
	禁止	0	[X4E
JAN/EAN-13/8 Addon2 読み取り	許可	1	R5
	禁止	0	[X4F
JAN/EAN-13/8 Addon5 読み取り	許可	1	R6
	禁止	0	[X4G
Code-39 読み取り	許可	1	B2
	禁止	0	VB
Tri-optic 読み取り	許可	1	JZ
	禁止	0	[DDJ
NW-7 読み取り	許可	1	B3
	禁止	0	VC
Industrial 2 of 5 読み取り	許可	1	R7
	禁止	0	V7
Interleaved 2 of 5 読み取り	許可	1	R8
	禁止	0	V8
S-Code 読み取り	許可	1	R9
	禁止	0	[DDK
Matrix2of5 読み取り	許可	1	BB
	禁止	0	[DDL

設定項目	パラメータ	設定値	コマンド
Code 93 読み取り	許可	1	B5
	禁止	0	VD
Code 128 読み取り	許可	1	B6
	禁止	0	VE
EAN-128 読み取り	許可 if Possible	2	OG
	許可 EAN-128 Only	1	JF
	禁止 Code 128 として出力	0	OF
MSI/Plessey 読み取り	許可	1	B7
	禁止	0	VF
IATA	許可	1	B8
	禁止	0	VH
UK/Plessey 読み取り	許可	1	B1
	禁止	0	VA
Telepen 読み取り	許可	1	B9
	禁止	0	VG
GS1 DataBar(RSS-14)	許可	1	JX
	禁止	0	SJ
GS1 DataBar Limited(RSS-Limited)	許可	1	JY
	禁止	0	SK
GS1 DataBar Expanded(RSS-Expanded)	許可	1	DR
	禁止	0	SL
Code-11 読み取り	許可	1	[BLC
	禁止	0	[BLA
Code3of5 読み取り	許可	1	WH
	禁止	0	WI
UPC-A フォーマット (C/D 転送)	1 3 桁 (先頭 0 と C D 付き)	10	E2

設定項目	パラメータ	設定値	コマンド
	1 2桁(先頭0なし)	2	E3
	1 2桁 (CDなし)	8	E4
	1 1桁 (先頭0とCDなし)	0	E5
UPC-E フォーマット (C/D 転送)	8桁 (先頭0とCD付き)	10	E6
	7桁 (先頭なし)	2	E7
	7桁 (CDなし)	8	E8
	6桁 (先頭0とCDなし)	0	E9
JAN/EAN-13 フォーマット (C/D 転送)	CD 転送する	1	6K
	CD 転送しない	0	6J
JAN/EAN-8 フォーマット (C/D 転送)	CD 転送する	1	6I
	CD 転送しない	0	6H
Code 39 C/D 転送	CD 転送する	1	D9
	CD 転送しない	0	D8
NW-7 C/D 転送	CD 転送する	1	H8
	CD 転送しない	0	H9
Industrial 2 of 5/ Interleaved 2 of 5 C/D 転送	CD 転送する	1	E0
	CD 転送しない	0	E1
MSI/Plessey C/D 転送	CD 転送する CD1	1	4E
	CD 転送する CD1 and CD2	2	4F
	CD 転送しない	0	4G
IATA C/D 転送	CD 転送する	1	4L
	CD 転送しない	0	4M
GS1 DataBar family C/D 転送	CD 転送する	1	DL

設定項目	パラメータ	設定値	コマンド
	CD 転送しない	0	DM
WPC(UPC/EAN/JAN) C/D 計算	C/D 計算する	1	[XEE
	C/D 計算しない	0	[XEF
Code 39 C/D 計算	C/D 計算する	1	C0
	C/D 計算しない	0	C1
NW-7 C/D 計算	C/D 計算する Mod10/W1,2spec1	1	[XF8
	C/D 計算する Mod16	2	H6
	C/D 計算する 7 check	3	[XFB
	C/D 計算する Mod11	4	[XFC
	C/D 計算しない	0	H7
Industrial 2 of 5/ Interleaved 2 of 5 C/D 計算	C/D 計算する	1	G1
	C/D 計算しない	0	G0
Code 93 C/D 計算	C/D 計算する	1	AC
	C/D 計算しない	0	9Q
Code 128/EAN-128 C/D 計算	C/D 計算する	1	ME
	C/D 計算しない	0	MF
MSI/Plessey C/D 計算	C/D 計算する CD1 only(Mod10)	1	4B
	C/D 計算する CD's (Mod10/Mod10)	2	4C
	C/D 計算する CD's (Mod10/Mod11)	3	4D
	C/D 計算する CD's (Mod11/Mod10)	4	4R
	C/D 計算しない	0	4A

設定項目	パラメータ	設定値	コマンド
IATA C/D 計算	C/D 計算する (CPN+FORM SERIAL)	1	4J
	C/D 計算する (FORM SERIAL)	2	4I
	C/D 計算する (ALL DATA)	3	4K
	C/D 計算しない	0	4H
Code 39 スタートストップ 転送	転送する	1	D0
	転送しない	0	D1
NW-7 スタートストップ転送	転送する ABCD/TN*E	1	F1
	転送する abcd/tn*e	2	F2
	転送する ABCD	3	F3
	転送する abcd	4	F4
	転送する DC1DC2DC3DC4	5	FA
	転送しない	0	F0
OPN-3002i 固有設定			
Intelligent Mail 読み取り	許可	1	[D5F
	禁止	0	[D5G
Postnet 読み取り	許可	1	[D6A
	禁止	0	[D6B
Japanese postal 読み取り	許可	1	[D5P
	禁止	0	[D5Q
CodablockF 読み取り	許可	1	[D4P
	禁止	0	[D4Q
Data Matrix(ECC200) 読み取り	許可	1	[BCC
	禁止	0	[BCO
Data	許可	1	[BG0

設定項目	パラメータ	設定値	コマンド
Matrix(ECC000-140) 読み取り	禁止	0	[BG1
Aztec code 読み取り	許可	1	[BCH
	禁止	0	[BCT
Aztec runes 読み取り	許可	1	[BF2
	禁止	0	[BF3
Chinese Sensible code 読み取り	許可	1	[D4L
	禁止	0	[D4M
QR code 読み取り	許可	1	[BCD
	禁止	0	[BCP
MicroQR 読み取り	許可	1	[D2U
	禁止	0	[D2V
Maxi Code 読み取り	許可	1	[BCE
	禁止	0	[BCQ
Composite on GS1Databar 読み取り	許可	1	[BHE
	禁止	0	[BHF
Composite on UPC/EAN 読み取り	許可	1	[D1V
	禁止	0	[D1W

3.2 Bluetooth設定

設定項目	パラメータ	設定値	コマンド
読み取りモード	単発読み	1	S0
	複数読み	2	S1
	連続読み	3	S2
読み取り時間	無限	0	YM
	0 秒	-1	Y0
	1 秒	50	Y1
	2 秒	100	Y2
	3 秒	150	Y3
	4 秒	200	Y4
	5 秒	250	Y5

設定項目	パラメータ	設定値	コマンド
	6 秒	300	Y6
	7 秒	350	Y7
	8 秒	400	Y8
	9 秒	450	Y9
	読み取り時間 10 倍	-	YL
照合回数	1 回読取 0 回照合	0	X0
	2 回読取 1 回照合	1	X1
	3 回読取 2 回照合	2	X2
	4 回読取 3 回照合	3	X3
	5 回読取 4 回照合	4	BS
	6 回読取 5 回照合	5	BT
	7 回読取 6 回照合	6	BU
	8 回読取 7 回照合	7	BV
	9 回読取 8 回照合	8	BW
	10 回読取 9 回照合	9	[XBT
	11 回読取 10 回照合	10	[XBU
	12 回読取 11 回照合	11	[XBV
	13 回読取 12 回照合	12	[XBW
	14 回読取 13 回照合	13	[XBX
	15 回読取 14 回照合	14	[XBY
	16 回読取 15 回照合	15	[XBZ
二度読み防止タイマ	無限	0	AG
	50ms	3	AH
	100ms	5	AI
	200ms	10	AJ
	300ms	15	AK
	400ms	20	AL
	500ms	25	AM
	600ms	30	AN
アドオンタイマ	なし	0	XA
	250ms	13	XB
	500ms	25	XC
	750ms	38	XD
ブザーボリューム	最大	127	T0

設定項目	パラメータ	設定値	コマンド
	大	32	T1
	中	8	T2
	小	1	T3
LED 点灯時間	無効	0	T4
	200ms	10	T5
	400ms	20	T6
	600ms	30	T7
トリガモード	トリガ無効	0	S7
	トリガ有効	1	S8
トリガリピート	無効	0	/K
	有効	1	/M
ブザー音	無効	0	W0
	有効	1	W8
ブザートーン	単音	0	W1
	高低	1	W2
	低高	2	W3
	4.5kHz	3	[XTS
	2.2kHz-2kHz	4	[X%Q
ブザー鳴動時間	100ms	5	W4
	200ms	10	W5
	400ms	20	W6
	50ms	2	W7
ブザー鳴動タイミング	転送前ブザー	0	VY
	転送後ブザー	1	VZ
接続相手アドレス		char× 12¥0	
	アドレス設定開始	-]BDAS
	アドレス設定終了	-]BDAE
コマンド応答	有り (ACK/NAK)	1	WC
	BT33(0-7)の設定に従う	0	WD
PIN コード		char×	

設定項目	パラメータ	設定値	コマンド
		16¥0	
	PIN 設定開始	-]PINS
	PIN 設定終了	-]PINE
トリガ接続	無効	0]TSCD
	有効	1]TSCE
アドレスバーコード読み取りによる接続処理実行有無	無効	0]DIAU
	有効	1]ENAU
ACK/NAK 制御	無し	0	[XP5
	有り	1	P3
	有り (NoResponse)	2	P4
接続モード	SPP マスター	0]BCMA
	SPP スレーブ	1]BCSA
	HID	2	[C02
スレーブ接続待ち時間	30 秒	1500]SWT0
	1 分	3000]SWT1
	2 分	6000]SWT2
	3 分	9000]SWT3
	4 分	12000]SWT4
自動再接続有効時間	無効	0]CA00
	1 分	3000]CA01
	2 分	6000]CA02
	3 分	9000]CA03
	4 分	12000]CA04
	5 分	15000]CA05
	6 分	18000]CA06
	7 分	21000]CA07
	8 分	24000]CA08
	9 分	27000]CA09
	10 分	30000]CA10
	11 分	33000]CA11
	12 分	36000]CA12

設定項目	パラメータ	設定値	コマンド
	13 分	39000]CA13
	14 分	42000]CA14
	15 分	45000]CA15
自動切断時間	無効	0]AD00
	10 分	30000]AD01
	20 分	60000]AD02
	30 分	90000]AD03
	40 分	120000]AD04
	50 分	150000]AD05
	60 分	180000]AD06
	1 分	3000]ADM1
	2 分	6000]ADM2
	3 分	9000]ADM3
	4 分	12000]ADM4
	5 分	15000]ADM5
	6 分	18000]ADM6
	7 分	21000]ADM7
	8 分	24000]ADM8
	9 分	27000]ADM9
	10 秒	500]ADS1
	20 秒	1000]ADS2
	30 秒	1500]ADS3
	40 秒	2000]ADS4
	50 秒	2500]ADS5
トリガ接続長押し時間	トリガ接続無効	0]PC00
	1 秒	50]PC01
	2 秒	100]PC02
	3 秒	150]PC03
	4 秒	200]PC04
	5 秒	250]PC05
	6 秒	300]PC06
	7 秒	350]PC07
	8 秒	400]PC08
	9 秒	450]PC09

設定項目	パラメータ	設定値	コマンド
トリガ切断長押し時間	トリガ切断無効	0]PD00
	1 秒	50]PD01
	2 秒	100]PD02
	3 秒	150]PD03
	4 秒	200]PD04
	5 秒	250]PD05
	6 秒	300]PD06
	7 秒	350]PD07
	8 秒	400]PD08
	9 秒	450]PD09
ACK/NAK 待ち時間	1s	50	[XI7
	2s	100	[XI8
	5s	250	[XI9
圏外メモリー	無効	0]DTMD
	有効	1]DTME
データコレクト	有効	0	[BM0
	無効	1	[BM1
バーコード読取時自動接続	無効	0]ARCD
	有効	1]ARCE
データコレクタからの切断時ブザー	無効	0]DSSD
	有効	1]DSSE
接続相手からの切断時ブザー	無効	0]DSPD
	有効	1]DSPE
メモリーデータの出力方法	接続時即出力	0	[EBB
	ファンクションキー押下 かコマンドで出力	1	[EBC
	データ出力コマンド		[EBD
USB 接続時 COM 通信	無効	0	[C10
	有効	1	[C11
ファンクション押下時出力	HT	0x09	[\$09
	LF	0x0A	[\$0A
	CR	0x0D	[\$0D

設定項目	パラメータ	設定値	コマンド
	CAN	0x18	[\$18
	ESC	0x1B	[\$1B
	iphone ソフトキーボード 表示	0xA6	[\$A6
	ENTER	0xB2	[\$B2
Bluetooth デバイス名 (自機)	デバイス名設定開始	-	[E65
	デバイス名設定終了	-	[E66
前回のスレーブ接続時の相手アドレス(12 文字 + NUL)			
OPN-3002i 固有設定			
グッドリードバイブレーター	有効	1	[EBI
	無効	0xFF	[EBH

3.3 プリフィックス

設定項目	-	-	コマンド
UPC-A のプリフィックス			N1
UPC-A アドオンのプリフィックス			M0
UPC-E のプリフィックス			N2
UPC-E アドオンのプリフィックス			M1
JAN/EAN-13 のプリフィックス			N3
JAN/EAN-13 アドオンのプリフィックス			M2
JAN/EAN-8 のプリフィックス			N4
JAN/EAN-8 アドオンのプリフィックス			M3
Code-39 のプリフィックス			M4
Tri-Optic のプリフィックス			MC
NW-7 のプリフィックス			M5
Industrial2of5 のプリフィックス			M6
Interleaved2of5 のプリフィックス			M7
S Code のプリフィックス			MB
Matrix 2of5 のプリフィックス			GL

設定項目	-	-	コマンド
Code-93 のプリフィクス			M8
Code-128 のプリフィクス			M9
MSI/Plessey のプリフィクス			N0 (ゼロ)
IATA のプリフィクス			I8
UK/Plessey のプリフィクス			MA
Telepen のプリフィクス			L8
RSS のプリフィクス			OE
Code-11 のプリフィクス			[BLD
Code 3of5 のプリフィクス			*\$
EAN-128 のプリフィクス			[XMX
コモンプリフィックス設定			MZ
全コードのプリフィクス設定			RY
OPN-3002i 固有設定			
Intelligent Mail のプリフィクス			[D5I
Postnet のプリフィクス			[D6D
Japanese postal のプリフィクス			[D5S
CodablockF のプリフィクス			[D4S
Data Matrix(ECC200, ECC000-140) のプリフィクス			MD
Aztec code/Aztec Runes のプリフィクス			[BF0
Chinese Sensible code のプリフィクス			[D4N
QR/MicroQR のプリフィクス			MK
Maxi code のプリフィクス			ML
Composite(on GS1Databar,UPC/EAN) のプリフィクス			RR

3.4 サフィックス

設定項目	-	-	コマンド
UPC-A のサフィックス			N6
UPC-A アドオンのサフィックス			O0 (オーゼロ)
UPC-E のサフィックス			N7
UPC-E アドオンのサフィックス			O1
JAN/EAN-13 のサフィックス			N8
JAN/EAN-13 アドオンのサフィックス			O2
JAN/EAN-8 のサフィックス			N9
JAN/EAN-8 アドオンのサフィックス			O3
Code-39 のサフィックス			O4
Tri-Optic のサフィックス			PN
NW-7 のサフィックス			O5
Industrial 2of5 のサフィックス			O6
Interleaved 2of5 のサフィックス			O7
S Code のサフィックス			OB
Matrix 2of5 のサフィックス			GM
Code-93 のサフィックス			O8
Code-128 のサフィックス			O9
MSI/Plessey のサフィックス			N5
IATA のサフィックス			I9
UK/Plessey のサフィックス			OA
Telepen のサフィックス			L9
RSS のサフィックス			PQ
Code-11 のサフィックス			[BLE
Code 3of5 のサフィックス			*%
EAN-128 のサフィックス			[XOX
コモンサフィックス			PS
全コードのサフィックス設定			RZ
OPN-3002i 固有設定			
Intelligent Mail のサフィックス			[D5J

設定項目	-	-	コマンド
ス			
Postnet のサフィックス			[D6E
Japanese postal のサフィックス			[D5T
CodablockF のサフィックス			[D4T
Data Matrix(ECC200, ECC000-140) のサフィックス			PO
Aztec code/Aztec Runes のサフィックス			[BF1
Chinese Sensible code のサフィックス			[D4O
QR/MicroQR のサフィックス			PW
Maxi code のサフィックス			PX
Composite(on GS1Databar,UPC/EAN) のサフィックス			RS

3.5 文字列設定

設定文字	コマンド	利用可能な設定	XZ5 出力
0	Q0	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3000
1	Q1	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3100
2	Q2	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3200
3	Q3	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3300
4	Q4	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3400
5	Q5	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3500

設定文字	コマンド	利用可能な設定	XZ5 出力
		イックス、サフィックス	
6	Q6	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3600
7	Q7	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3700
8	Q8	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3800
9	Q9	時刻、PIN、接続相手アドレス、デバイス名、プリフィックス、サフィックス	3900
A	0A	接続相手アドレス、デバイス名、プリフィックス、サフィックス	4100
B	0B	接続相手アドレス、デバイス名、プリフィックス、サフィックス	4200
C	0C	接続相手アドレス、デバイス名、プリフィックス、サフィックス	4300
D	0D	接続相手アドレス、デバイス名、プリフィックス、サフィックス	4400
E	0E	接続相手アドレス、デバイス名、プリフィックス、サフィックス	4500
F	0F	接続相手アドレス、デバイス名、プリフィックス、サフィックス	4600
G	0G	デバイス名、プリフィックス、サフィックス	4700
H	0H	デバイス名、プリフィックス、サフィックス	4800
I	0I	デバイス名、プリフィックス、サフィックス	4900
J	0J	デバイス名、プリフィックス、サフィックス	4A00
K	0K	デバイス名、プリフィックス、サフィックス	4B00
L	0L	デバイス名、プリフィックス、サフィックス	4C00
M	0M	デバイス名、プリフィックス、サフィックス	4D00
N	0N	デバイス名、プリフィックス、サフィックス	4E00
O	0O	デバイス名、プリフィックス、サフィックス	4F00
P	0P	デバイス名、プリフィックス、サフィックス	5000
Q	0Q	デバイス名、プリフィックス、サフィックス	5100
R	0R	デバイス名、プリフィックス、サフィックス	5200

設定文字	コマンド	利用可能な設定	XZ5 出力
S	0S	デバイス名、プリフィックス、サフィックス	5300
T	0T	デバイス名、プリフィックス、サフィックス	5400
U	0U	デバイス名、プリフィックス、サフィックス	5500
V	0V	デバイス名、プリフィックス、サフィックス	5600
W	0W	デバイス名、プリフィックス、サフィックス	5700
X	0X	デバイス名、プリフィックス、サフィックス	5800
Y	0Y	デバイス名、プリフィックス、サフィックス	5900
Z	0Z	デバイス名、プリフィックス、サフィックス	5A00
a	\$A	デバイス名、プリフィックス、サフィックス	6100
b	\$B	デバイス名、プリフィックス、サフィックス	6200
c	\$C	デバイス名、プリフィックス、サフィックス	6300
d	\$D	デバイス名、プリフィックス、サフィックス	6400
e	\$E	デバイス名、プリフィックス、サフィックス	6500
f	\$F	デバイス名、プリフィックス、サフィックス	6600
g	\$G	デバイス名、プリフィックス、サフィックス	6700
h	\$H	デバイス名、プリフィックス、サフィックス	6800
i	\$I	デバイス名、プリフィックス、サフィックス	6900
j	\$J	デバイス名、プリフィックス、サフィックス	6A00
k	\$K	デバイス名、プリフィックス、サフィックス	6B00
l	\$L	デバイス名、プリフィックス、サフィックス	6C00
m	\$M	デバイス名、プリフィックス、サフィックス	6D00
n	\$N	デバイス名、プリフィックス、サフィックス	6E00
o	\$O	デバイス名、プリフィックス、サフィックス	6F00
p	\$P	デバイス名、プリフィックス、サフィックス	7000
q	\$Q	デバイス名、プリフィックス、サフィックス	7100
r	\$R	デバイス名、プリフィックス、サフィックス	7200
s	\$S	デバイス名、プリフィックス、サフィックス	7300
t	\$T	デバイス名、プリフィックス、サフィックス	7400
u	\$U	デバイス名、プリフィックス、サフィックス	7500
v	\$V	デバイス名、プリフィックス、サフィックス	7600
w	\$W	デバイス名、プリフィックス、サフィックス	7700
x	\$X	デバイス名、プリフィックス、サフィックス	7800
y	\$Y	デバイス名、プリフィックス、サフィックス	7900

設定文字	コマンド	利用可能な設定	XZ5 出力
z	\$Z	デバイス名、プリフィックス、サフィックス	7A00
<SPACE>	5A	デバイス名、プリフィックス、サフィックス	2000
!	5B	デバイス名、プリフィックス、サフィックス	2100
"	5C	デバイス名、プリフィックス、サフィックス	2200
#	5D	デバイス名、プリフィックス、サフィックス	2300
\$	5E	デバイス名、プリフィックス、サフィックス	2400
%	5F	デバイス名、プリフィックス、サフィックス	2500
&	5G	デバイス名、プリフィックス、サフィックス	2600
'	5H	デバイス名、プリフィックス、サフィックス	2700
(5I	デバイス名、プリフィックス、サフィックス	2800
)	5J	デバイス名、プリフィックス、サフィックス	2900
*	5K	デバイス名、プリフィックス、サフィックス	2A00
+	5L	デバイス名、プリフィックス、サフィックス	2B00
,	5M	デバイス名、プリフィックス、サフィックス	2C00
-	5N	デバイス名、プリフィックス、サフィックス	2D00
.	5O	デバイス名、プリフィックス、サフィックス	2E00
/	5P	デバイス名、プリフィックス、サフィックス	2F00
:	6A	デバイス名、プリフィックス、サフィックス	3A00
;	6B	デバイス名、プリフィックス、サフィックス	3B00
<	6C	デバイス名、プリフィックス、サフィックス	3C00
=	6D	デバイス名、プリフィックス、サフィックス	3D00
>	6E	デバイス名、プリフィックス、サフィックス	3E00
?	6F	デバイス名、プリフィックス、サフィックス	3F00
@	6G	デバイス名、プリフィックス、サフィックス	4000
[7A	デバイス名、プリフィックス、サフィックス	5B00
¥	7B	デバイス名、プリフィックス、サフィックス	5C00
]	7C	デバイス名、プリフィックス、サフィックス	5D00
^	7D	デバイス名、プリフィックス、サフィックス	5E00
_	7E	デバイス名、プリフィックス、サフィックス	5F00
`	7F	デバイス名、プリフィックス、サフィックス	6000
{	9T	デバイス名、プリフィックス、サフィックス	7B00
	9U	デバイス名、プリフィックス、サフィックス	7C00
}	9V	デバイス名、プリフィックス、サフィックス	7D00

設定文字	コマンド	利用可能な設定	XZ5 出力
~	9W	デバイス名、プリフィックス、サフィックス	7E00
^@ (NULL)	9G	プリフィックス、サフィックス	00FF
^A (SOH)	1A	プリフィックス、サフィックス	01FF
^B (STX)	1B	プリフィックス、サフィックス	02FF
^C (ETX)	1C	プリフィックス、サフィックス	03FF
^D (EOT)	1D	プリフィックス、サフィックス	04FF
^E (ENQ)	1E	プリフィックス、サフィックス	05FF
^F (ACK)	1F	プリフィックス、サフィックス	06FF
^G (BEL)	1G	プリフィックス、サフィックス	07FF
^H (BS)	1H	プリフィックス、サフィックス	08FF
^I (HT)	1I	プリフィックス、サフィックス	09FF
^J (LF)	1J	プリフィックス、サフィックス	0AFF
^K (VT)	1K	プリフィックス、サフィックス	0BFF
^L (FF)	1L	プリフィックス、サフィックス	0CFF
^M (CR)	1M	プリフィックス、サフィックス	0DFF
^N (SO)	1N	プリフィックス、サフィックス	0EFF
^O (SI)	1O	プリフィックス、サフィックス	0FFF
^P (DLE)	1P	プリフィックス、サフィックス	1FF0
^Q (DC1)	1Q	プリフィックス、サフィックス	11FF
^R (DC2)	1R	プリフィックス、サフィックス	12FF
^S (DC3)	1S	プリフィックス、サフィックス	13FF
^T (DC4)	1T	プリフィックス、サフィックス	14FF
^U (NAK)	1U	プリフィックス、サフィックス	15FF
^V (SYN)	1V	プリフィックス、サフィックス	16FF
^W (ETB)	1W	プリフィックス、サフィックス	17FF
^X (CAN)	1X	プリフィックス、サフィックス	18FF
^Y (EM)	1Y	プリフィックス、サフィックス	19FF
^Z (SUB)	1Z	プリフィックス、サフィックス	1AFF
^[(ESC)	9A	プリフィックス、サフィックス	1BFF
^\\ (FS)	9B	プリフィックス、サフィックス	1CFF
^] (GS)	9C	プリフィックス、サフィックス	1DFF
^^ (RS)	9D	プリフィックス、サフィックス	1EFF
^_ (US)	9E	プリフィックス、サフィックス	1FFF

設定文字	コマンド	利用可能な設定	XZ5 出力
DEL (ASCII 127)	9F	プリフィックス、サフィックス	7FFF
年	[\$YR	プリフィックス、サフィックス	2559
月	[\$MO	プリフィックス、サフィックス	254D
日	[\$DY	プリフィックス、サフィックス	2544
時	[\$HR	プリフィックス、サフィックス	2568
分	[\$MI	プリフィックス、サフィックス	256D
秒	[\$SC	プリフィックス、サフィックス	2573
スキャン カウント	[\$CT	プリフィックス、サフィックス	2543
バーコード 種別	[\$BT	プリフィックス、サフィックス	2554
バーコード データ長	[\$BL	プリフィックス、サフィックス	254C
電池電圧	[\$BV	プリフィックス、サフィックス	2556
BD アドレス	[\$AR	プリフィックス、サフィックス	2541
端末 ID	[\$ID	プリフィックス、サフィックス	2549
端末名	[\$NM	プリフィックス、サフィックス	254E

4. 配布

MFi 認証機器へ接続するアプリケーションを iTunes Store から配布するには、MFi 認証機器の製造メーカー（弊社）から Apple Inc. へそのソフトウェアの情報を登録する必要があります。

Ad hoc や Enterprise での配布は登録の必要はありません。

iTunes Store での配布が必要な場合は弊社へご連絡ください。

弊社製品名: OPN-2002i Series
OPN-3002i Series
発行番号: Rev.1.1
管理番号: SI12057

株式会社オプトエレクトロニクス

E-Mail: sales@opto.co.jp URL: <http://www.opto.co.jp>