

# iOS Program Interface Manual

## Bluetooth:

### 1. Defined in the IHRBLEManager Class

#### 1.1 sharedInstance

Singleton.

- (void) sharedInstance

#### 1.2 startScan

Search for bluetooth printers.

- (void) startScan

#### 1.3 stopScan

Stop searching for bluetooth printers.

-(void) stopScan

#### 1.4 connectDevice

Connect a Printer.

- (void) connectDevice:(CBPeripheral \*) peripheral

[Parameter]

-peripheral: Device.

#### 1.5 disconnectRootPeripheral

Disconnect a Printer.

- (void) disconnectRootPeripheral

#### 1.6 writeCommandWithData

Data printing.

- (void) writeCommandWithData: (NSData \*) data

[Parameter]

- data: Data to print.

## 1.7 getPrinterData

Get printer status.

- (void) getPrinterData: (IHRBLEManagerReceiveCallbackBlock \*) receiveBlock  
[Parameter]
- receiveBlock: callback function return characteristic.

## IHRBLEManagerDelegte

### 1.7 IHRbleUpdatePeripheralList

Discover peripherals.

- (void) IHRbleUpdatePeripheralList: (NSArray \*) peripherals RSSIList:  
(NSArray \*) rssiList  
[Parameter]
- peripherals: Peripheral List.
- rssiList: Rssi List.

### 1.8 IHRbleConnectPeripheral

Peripheral connected succeeded.

- (void) IHRbleConnectPeripheral: (CBPeripheral \*) peripheral  
[Parameter]
- peripheral: Device.

### 1.9 IHRbleFailToConnectPeripheral

Peripheral connected failed.

- (void) IHRbleFailToConnectPeripheral: (CBPeripheral \*) peripheral error:  
(NSError \*)error  
[Parameter]
- peripheral: Device.
- error: Connect error.

### 1.10 IHRbleDisconnectPeripheral

Peripheral disconnected.

- (void) IHRbleDisconnectPeripheral: (CBPeripheral \*) peripheral isAutoDisconnect:  
(BOOL \*) isAutoDisconnect  
[Parameter]
- peripheral: Device.
- isAutoDisconnect: Whether to automatically disconnect.

## 1.11 IHRbleWriteValueForCharacteristic

Send data succeeded.

- (void) IHRbleWriteValueForCharacteristic: (CBCharacteristic \*) character error: (NSError \*) error  
[Parameter]
- character: Device attributes.
- error: Send error.

## 1.12 IHRbleReceiveValueForCharacteristic

receive data from printer.

- (void) IHRbleReceiveValueForCharacteristic: (CBCharacteristic \*) character error: (NSError \*) error  
[Parameter]
- character: Device attributes.
- error: Receive error.

# WiFi:

## 2. Defined in the IHRWIFIManager Class

### 2.1 sharedInstance

Singleton.

- (void) sharedInstance

### 2.2 connectWithHost

Connect a Printer.

- (void) connectWithHost: (NSString \*) hostStr port: (UInt16 \*) port  
[Parameter]
- hostStr: Connect printer ip address.
- port: Port Number.

### 2.3 disconnect

Disconnect a Printer.

- (void) disconnect

### 2.4 writeCommandWithData

Data printing.

- (void) writeCommandWithData: (NSData \*) data

[Parameter]

- data: Data to print.

## 2.5 getPrinterData

Get printer status.

- (void) getPrinterData: (IHRWIFIManagerReceiveCallbackBlock \*) receiveBlock

[Parameter]

- receiveBlock: callback function return characteristic.

## IHRWIFIManagerDelegate

### 2.5 IHRwifiConnectedToHost

Connected a Printer.

- (void) IHRwifiConnectedToHost: (NSString \*) host port: (UInt16) port

[Parameter]

- host: IP Address.
- port: Port Number.

### 2.6 IHRwifiDisconnectWithError

Disconnected a Printer.

- (void) IHRwifiDisconnectWithError: (NSError \*) error

[Parameter]

- error: error.

### 2.7 IHRwifiDisconnectForManual

Disconnected a Printer for manual.

- (void) IHRwifiDisconnectForManual

### 2.8 IHRwifiWriteValueWithTag

Print successful.

- (void) IHRwifiWriteValueSuccessWithTag:(long)tag

### 2.9 IHRwifiReceiveValueForData

receive data from printer.

- (void) IHRwifiReceiveValueForData:(NSData \*)data

[Parameter]

- data: Printer data.

## 3. Defined in the IHRCommand Class

### 3.1 initializePrinter

Initialize the printer.

- (NSData \*) initializePrinter

[Return value]

- NSData \*: Send data.

### 3.2 printAndFeedLine

Print and wrap.

- (NSData \*) printAndFeedLine

[Return value]

- NSData \*: Send data.

### 3.3 printAndBackStandardModel

Print and go back to standard mode.

- (NSData \*) printAndBackStandardModel

[Return value]

- NSData \*: Send data.

### 3.4 printAndTabs

Print and tab.

- (NSData \*) printAndTabs

[Return value]

- NSData \*: Send data.

### 3.5 cancelPrintDataByPageModel

Cancel print data in page mode.

- (NSData \*) cancelPrintDataByPageModel

[Return value]

- NSData \*: Send data.

### 3.6 printUnderPageModel

Printing in page mode.

- (NSData \*) printUnderPageModel

[Return value]

- NSData \*: Send data.

### 3.7 setDefaultLineSpace

Set default line spacing.

- (NSData \*) setDefaultLineSpace

[Return value]

- NSData \*: Send data.

### 3.8 setLeftSpaceWithnL

Set left margin.

- (NSData \*) setLeftSpaceWithnL:(int) nL andnH:(int) nH

[Parameter]

- nL: 0 ~ 255

- nH: 0 ~ 255

(nL+nH\*256) print area width

[Return value]

- NSData \*: Send data.

### 3.9 horizontalPosition

Horizontal positioning.

- (NSData \*) horizontalPosition

[Return value]

- NSData \*: Send data.

### 3.10 selectPagemodel

Select page mode.

- (NSData \*) selectPagemodel

[Return value]

- NSData \*: Send data.

### 3.11 selectFont

Select page mode.

- (NSData \*) selectFont: (int) n

[Parameter]

- n:

0, 48	ASCII (12 * 24)
-------	-----------------

1, 49	ASCII (9 * 17)
-------	----------------

[Return value]

- NSData \*: Send data.

### 3.12 selectInternationCharacterSets

Select international character set.

- (NSData \*) selectInternationCharacterSets: (int) n

[Parameter]

- n:

0	America
1	France
2	Germany
3	England
4	Denmark 1
5	Sweden
6	Italy
7	Spain 1
8	Japan
9	Norway
10	Denmark 2
11	Spain 2
12	Latin America
13	Korea
14	Slovenia / Croatia
15	China

[Return value]

- NSData \*: Send data.

### 3.13 selectStabdardModel

Select standard mode

- (NSData \*) selectStabdardModel

[Return value]

- NSData \*: Send data.

### 3.14 printAndFeedForwardWhitN

Print and advance paper n lines.

- (NSData \*) printAndFeedForwardWhitN: (int) n

[Parameter]

- n: 0~255.

[Return value]

- NSData \*: Send data.

### 3.15 selectOrCancleInvertPrintModel

Select or cancel the black and white reverse print mode.

- (NSData \*) selectOrCancleInvertPrintModel: (int) n

[Parameter]

- n: 1: select; 0: cancel.

[Return value]

- NSData \*: Send data.

### 3.16 setHorizontalAndVerticalMoveUnitWithX

Set horizontal and vertical movement units.

- (NSData \*) setHorizontalAndVerticalMoveUnitWithX: (int) x andY:(int) y

[Parameter]

- x: horizontal, 0~255.

- y: vertical, 0~255

[Return value]

- NSData \*: Send data.

### 3.17 setBarcodeHeight

Set barcode height.

- (NSData \*) setBarcodeHeight: (int) n

[Parameter]

- n: 1 ~ 255, default: 162.

[Return value]

- NSData \*: Send data.

### 3.18 setBarcodeWidth

Set barcode width.

- (NSData \*) setBarcodeWidth: (int) n

[Parameter]



- n: 2~6, default: 3.

[Return value]

- NSData \*: Send data.

### 3.19 printBarcodeWithM

Print barcode.

- (NSData \*) printBarcodeWithM: (int) m andContent: (NSString \*) content

useEnCodeing: (NSStringEncoding) strEncoding

[Parameter]

- m:

0	UPC-A
1	UPC-E
2	JAN13 (EAN13)
3	JAN8 (EAB13)
4	CODE39
5	ITF
6	CODABAR

- content: barcode content.

- strEncoding: Character Encoding.

[Return value]

- NSData \*: Send data.

### 3.20 printRasteBmpWithM

Print raster bitmap.

- (NSData \*) printRasteBmpWithM: (PrintRasterType) m andImage: (UIImage \*) image

andType: (BmpType) type andPaperHeight: (int) paperheight

[Parameter]

- m:

RasterNolmorWH	Normal
RasterDoubleWidth	Double width
RasterDoubleHeight	Double height
RasterDoubleWH	Double width and height

- image: image

- type: Dithering: Binary method.

Threshold: Dithering method.

- paperheight: Paper height.

[Return value]

- NSData \*: Send data.

### 3.21 printerOrderBuzzingHintWithRes

Printer beeps for print.

- (NSData \*) printerOrderBuzzingHintWithRes: (int) n andTime: (int) t

[Parameter]

- n: Beeps count, 1~9.

- t: Beeps time, (1~9) \* 50ms.

[Return value]

- NSData \*: Send data.

### 3.22 selectAlignment

Set alignment.

- (NSData \*) selectAlignment: (int) n

[Parameter]

- n:

0, 48	left alignment
1, 49	center alignment
2, 50	right alignment

[Return value]

- NSData \*: Send data.

### 3.23 selectHRICharactersPrintPosition

Set HRI Character print position.

- (NSData \*) selectHRICharactersPrintPosition: (int) n

[Parameter]

- n:

0, 48	none
1, 49	above
2, 50	below
3, 51	above and below

[Return value]

- NSData \*: Send data.

### 3.24 setPrintAreaWidthWithnL

Set the print area width.

- (NSData \*) setPrintAreaWidthWithnL: (int) nL andnH:(int)nH

[Parameter]

- nL:  $0 \leq nL \leq 255$

- nH:  $0 \leq nH \leq 255$

(nL+nH\*256) print area width

[Return value]

- NSData \*: Send data.

### 3.25 printQRCode

Print QR code.

- (NSData \*) printQRCode: (int) n level:(int)errLevel code:(NSString \*)code useEnCodeing:

(NSStringEncoding)strEncoding

[Parameter]

- n: unit size 1~16

- errLevel:error correct level 48~51

48	error correct level L (7%)
49	error correct level M (15%)
50	error correct level Q (25%)
51	error correct level H (30%)

- strEncoding: Character Encoding.

[Return value]

- NSData \*: Send data.

### 3.26 selectCutPageModelAndCutpage

Select the cut paper mode and cut the paper.

- (NSData \*) selectCutPageModelAndCutpage: (int) m

[Parameter]

- m:

0, 48	full cut
1, 49	half cut

[Return value]

- NSData \*: Send data.

### 3.27 creatCashBoxContorPulseWithM

Generate cash box control pulses.

- (NSData \*) creatCashBoxContorPulseWithM: (int) m andT1:(int) t1 andT2:(int) t2  
[Parameter]

- m:

0, 48	pin 2
1, 49	pin 5

- t1: 0 ~ 255, default 30

- t2: 0 ~ 255, default 255

Cash drawer open pulse high level time [t1 \* 2ms], low level [t2 \* 2ms],

if t2 < t1, low level [t1 \* 2ms]

[Return value]

- NSData \*: Send data.