

**METAPACE**

**API Reference Guide**

**Metapace SDK for UPOS  
Compliant  
iOS**

---

**Ver. 2.00**

## Copyright

© METAPACE Co., Ltd. All rights reserved.

This user manual and all property of the product are protected under copyright law. It is strictly prohibited to copy, store, and transmit the whole or any part of the manual and any property of the product without the prior written approval of METAPACE Co., Ltd.

The information contained herein is designed only for use with this METAPACE product.

METAPACE is not responsible for any direct or indirect damages, arising from or related to use of this information.

- The METAPACE logo is the registered trademark of METAPACE Co., Ltd.
- All other brand or product names are trademarks of their respective companies or organizations.

METAPACE Co., Ltd. maintains ongoing efforts to enhance and upgrade the functions and quality of all our products.

In the following, product specifications and/or user manual content may be changed without prior notice.

## Caution

Some semiconductor devices are easily damaged by static electricity. You should turn the printer "OFF", before you connect or remove the cables on the rear side, in order to guard the printer against the static electricity. If the printer is damaged by the static electricity, you should turn the printer "OFF".

# Table of Contents

<b>Copyright .....</b>	<b>2</b>
<b>1. About This Manual .....</b>	<b>6</b>
<b>2. Support OS and Interface .....</b>	<b>7</b>
2-1 Operating System .....	7
2-2 Supported Printers and Interfaces.....	7
<b>3. Development Environment .....</b>	<b>8</b>
3-1 System Requirements.....	8
3-1-1 Xcode.....	8
3-1-2 Project (for Bluetooth connection).....	8
3-1-3 Project (for discovering printers).....	9
3-2 Connecting iOS Device.....	10
3-2-1 Bluetooth.....	10
3-2-2 Network .....	12
3-2-3 Network-Ad Hoc Mode .....	13
<b>4. Package Contents .....</b>	<b>16</b>
4-1 Manual.....	16
4-2 Library .....	16
4-3 Sample source code.....	16
<b>5. Defines .....</b>	<b>17</b>
5-1 UPOS Define.h .....	17
5-1-1 Printer Search Results Notification Key Name .....	17
5-1-2 INTERFACE_TYPE .....	17
5-2 UPOS Defines Printer.h .....	18
5-2-1 UPOS_PRINTER_STATION.....	18
5-2-2 UPOS_PRINTER_BARCODE_ALIGNMENT .....	18
5-2-3 UPOS_PRINTER_BARCODE_TEXT_POSITION .....	18
5-2-4 UPOS_PRINTER_BARCODE_SYBBOLOGIES.....	19
5-2-5 UPOS_PRINTER_BITMAP_ALIGNMENT.....	19
5-2-6 UPOS_PRINTER_TRANSACTION.....	20
5-2-7 PAGE_MODE_DIRECTION .....	20
5-2-8 UPOS_PRINTER_STATUS.....	21
5-2-9 UPOS_PRINTER_BITMAP_WIDTH.....	21
<b>6. UPOS Devices Reference.....</b>	<b>22</b>
6-1 UPOS Device .....	23
6-1-1 init With Model Name .....	23
6-2 UPOS Devices.....	24
6-2-1 get List .....	24

6-2-2 add Device .....	24
6-2-3 remove Device .....	25
6-2-4 remove All Devices .....	25
6-3 UPOS Printer .....	26
6-4 UPOS Printers .....	26
6-5 UPOS Cash Drawer.....	26
6-6 UPOS Cash Drawers.....	26
<b>7. UPOS Device Controller Reference .....</b>	<b>27</b>
7-1 get Support Device Strings .....	27
7-2 refresh Wifi Lookup.....	28
7-3 refresh Ethernet Lookup .....	28
7-4 refresh BT Lookup.....	28
7-5 refresh BLE Lookup.....	28
7-6 get Registered Device.....	29
7-7 open.....	30
7-8 open.....	31
7-9 claim .....	32
7-10 set Enable Device .....	32
7-11 release Device.....	33
7-12 close.....	33
7-13 connect.....	34
<b>8. UPOS Printer Controller Reference .....</b>	<b>35</b>
8-1 UPOS Printer Controller .....	36
8-1-1 set Character Set .....	36
8-1-2 print Normal.....	38
8-1-3 cut Paper.....	39
8-1-4 print Barcode .....	40
8-1-5 print Raw Data .....	41
8-1-6 print Bitmap .....	42
8-1-7 print PDF.....	43
8-1-8 transaction Print .....	44
8-2 UPOS Printer Controller (Pagemode) .....	45
8-2-1 set Page Area .....	45
8-2-2 set Page Mode Direction.....	46
8-2-3 set Left Position .....	47
8-2-4 set Vertical Position .....	48
8-2-5 print Data In Page Mode .....	48
<b>9. UPOS CD Controller Reference.....</b>	<b>49</b>
9-1 UPOS CD Controller .....	49
9-1-1 Open Drawer.....	49

**10. UPOS Device Control Delegate Reference ..... 50**  
10-1 Output Complete Event.....50  
10-2 Status Update Event .....50

## **1. About This Manual**

This SDK manual describes the library required for developing applications for iOS.

It additionally describes how to use SDK, specifications, and restrictions.

## 2. Support OS and Interface

### 2-1 Operating System

- iOS 12.0 or later

### 2-2 Supported Printers and Interfaces

Model	Resolution	Communication Interface	Max Print Width(dot)
T-3II	180 dpi	Bluetooth / WLAN / Ethernet	512 dots
T-3III	180 dpi	Bluetooth / WLAN / Ethernet	512 dots

## 3. Development Environment

### 3-1 System Requirements

#### 3-1-1 Xcode

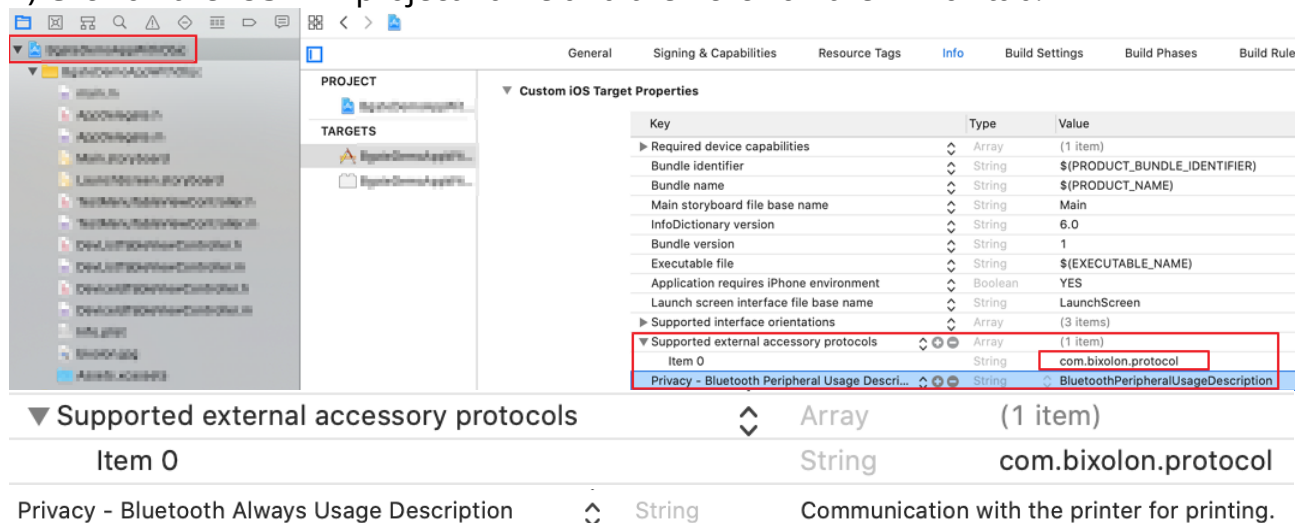
Reference: <http://developer.apple.com/devcenter/ios/index.action>

#### 3-1-2 Project (for Bluetooth connection)

To communicate with a Bluetooth printer, the following two properties must be added.

No.	Property Name	Description
1	UISupportedExternalAccessoryProtocols	Protocol for communicating with the device (printer)
2	NSBluetoothAlwaysUsageDescription	Prompt message for user why iOS APP needs Bluetooth access

1) Click on the iOS APP project name and then click on the “Info” tab.



2) In “Custom iOS Target Properties”, add the key “UISupportedExternalAccessoryProtocols” and enter the value “com.bixolon.protocol”.

3) In “Custom iOS Target Properties”, add the “NSBluetoothAlwaysUsageDescription” key and enter a string about why Bluetooth access is needed.



### 3-1-3 Project (for discovering printers)

If the iOS project deployment target version is 12 or higher and the iPhone/iPad OS version is 14 or higher, you must need to request multicast permission from Apple to discover printer devices. See the link below.

1) <https://developer.apple.com/news/?id=0oi77447>

2) <https://developer.apple.com/contact/request/networking-multicast>



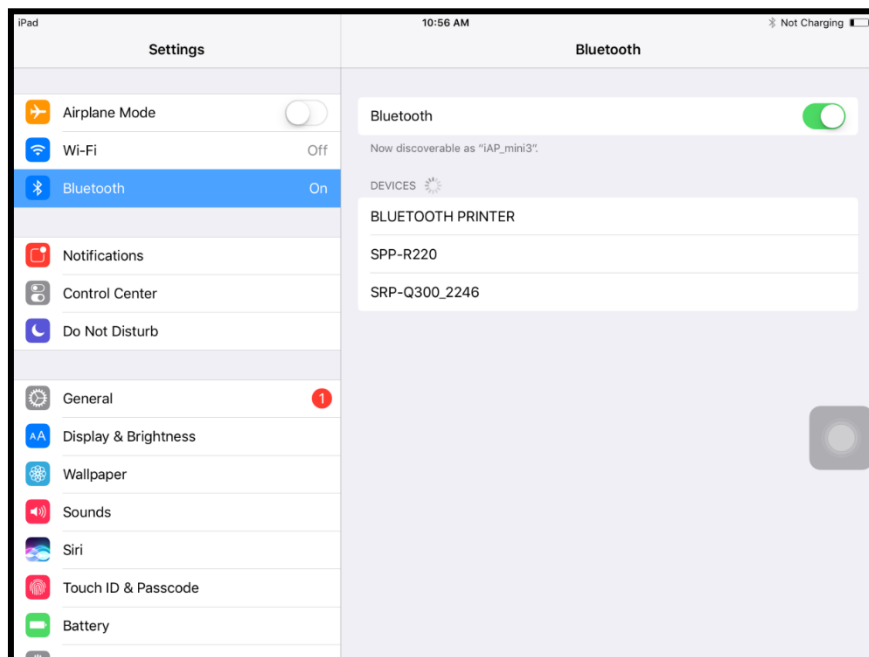
- IP address used for discovering devices: 255.255.255.255
- Port numbers used for discovering devices:  
48780, 48781, 9000, 3337

### **3-2 Connecting iOS Device**

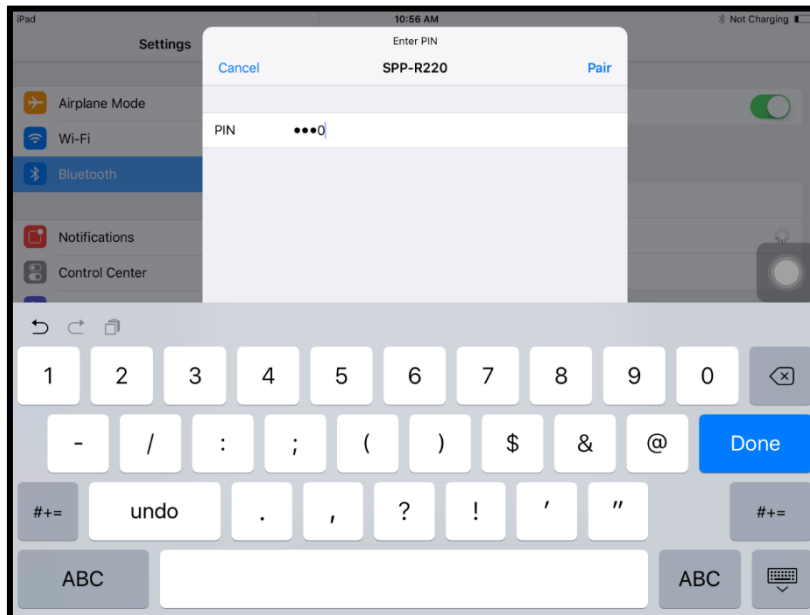
- Some details and names of specific items could be different depending on the iOS version or device.

#### 3-2-1 Bluetooth

1. Select [Settings].
2. Bluetooth should be enabled and the printer power should be on.
3. Select [Bluetooth] for settings.

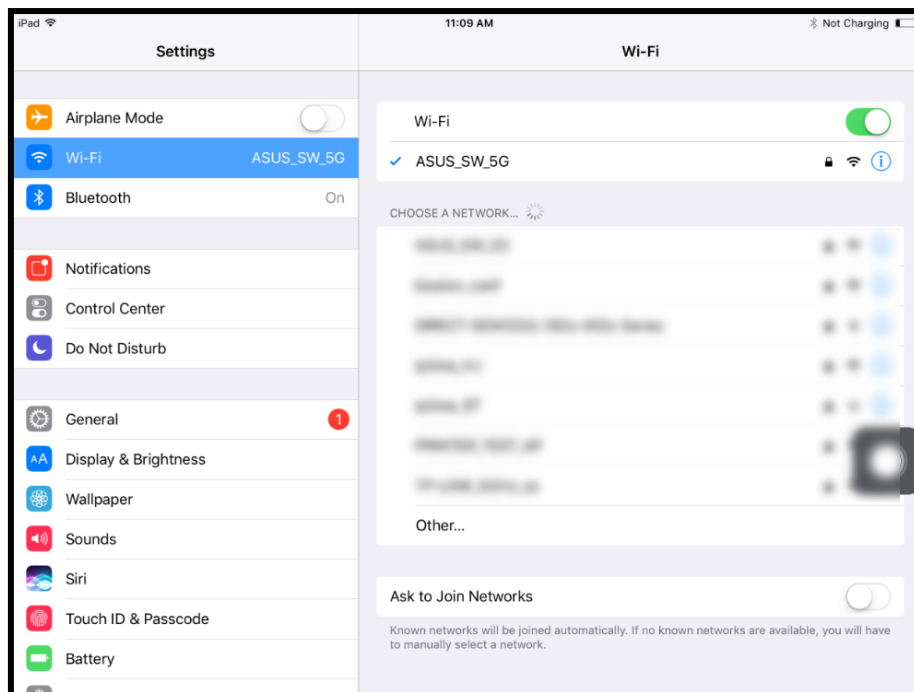


4. Select [Scan] to search the printer to connect and perform pairing.
5. Enter PIN code. Default PIN code is "0000".



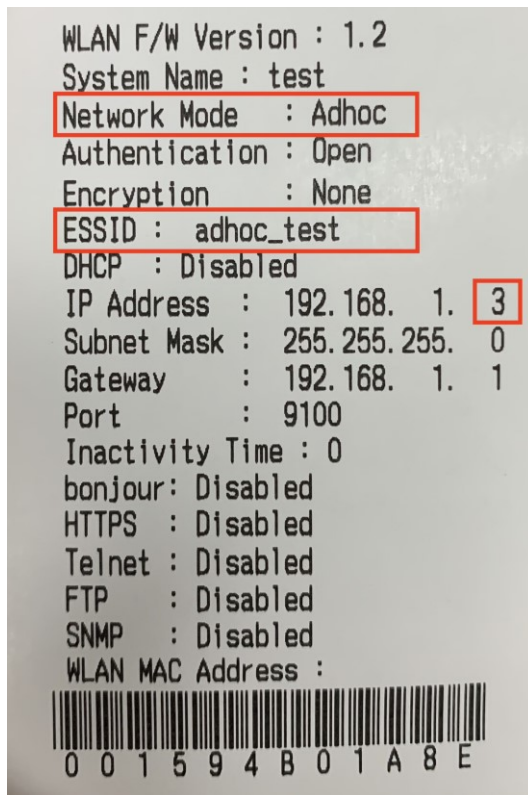
### **3-2-2 Network**

1. Connect the printer to the network AP (Access Point) and assign an IP address or obtain one using DHCP. As METAPACE's printer is initially set to Ad-hoc or SoftAP, the printer needs to be set up first with Net Configuration Tool to communicate with the device. The Net Configuration Tool can be downloaded from the METAPACE Website. (Refer to the Net Configuration Tool manual for details on settings)
2. Run Settings (located on the home screen) to configure Wi-Fi for the device.
3. Turn Wi-Fi on by clicking the switch on the top of the sub menus.
4. Connect the device to the same network that the printer is connected to.



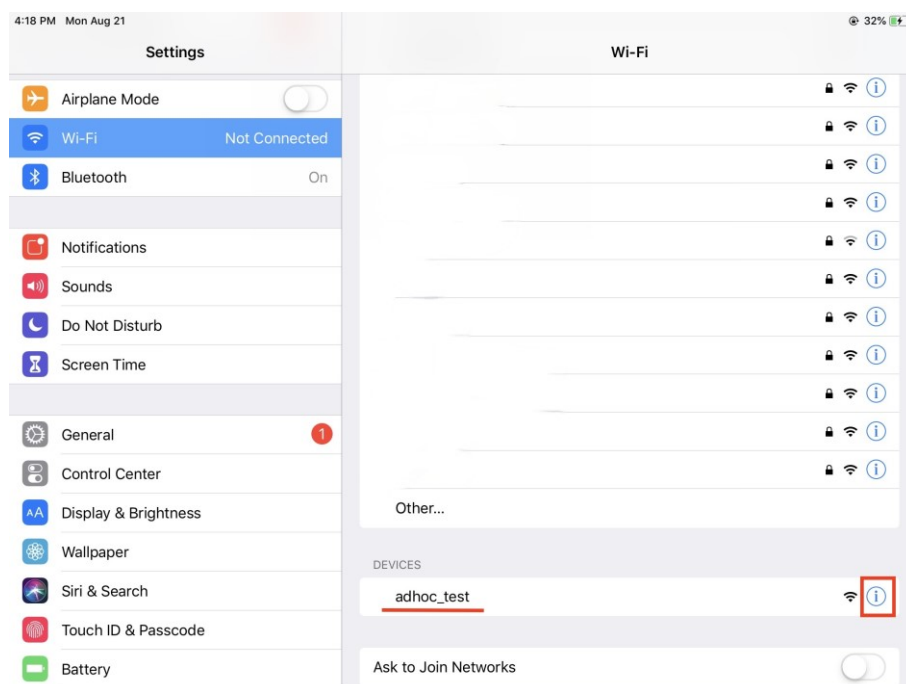
## 3-2-3 Network-Ad Hoc Mode

1. If the "Network Mode" of the printer is specified as "ADHOC", it is needed to configure IP Address".



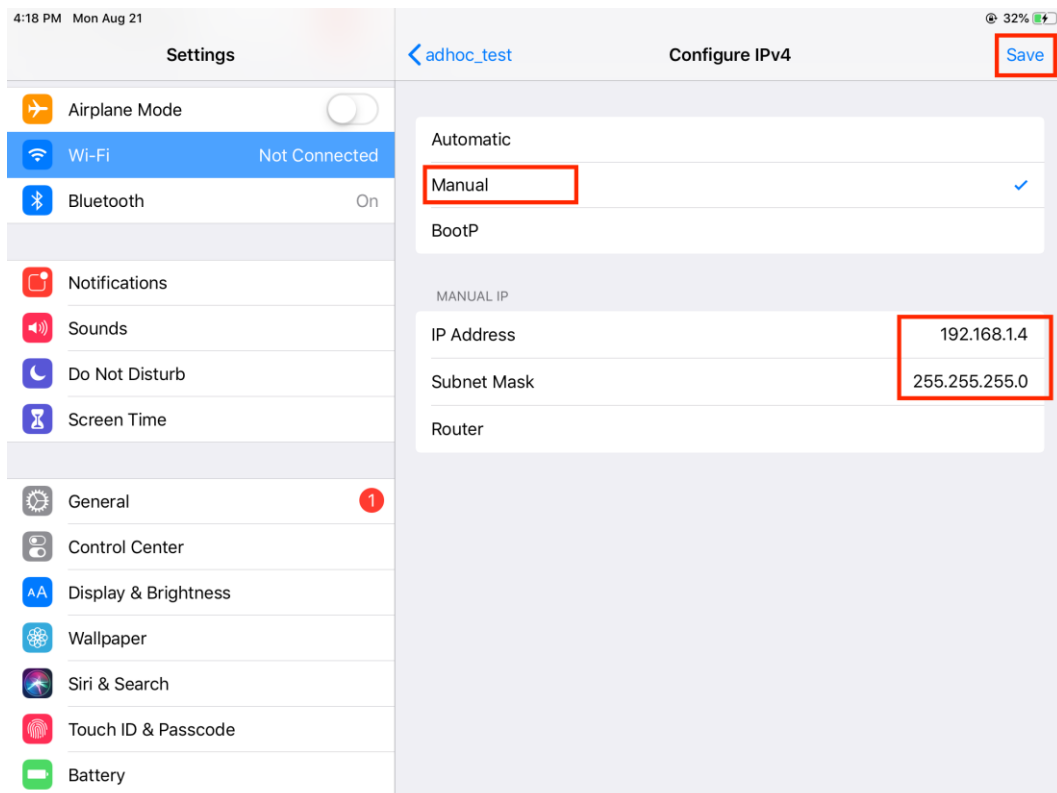
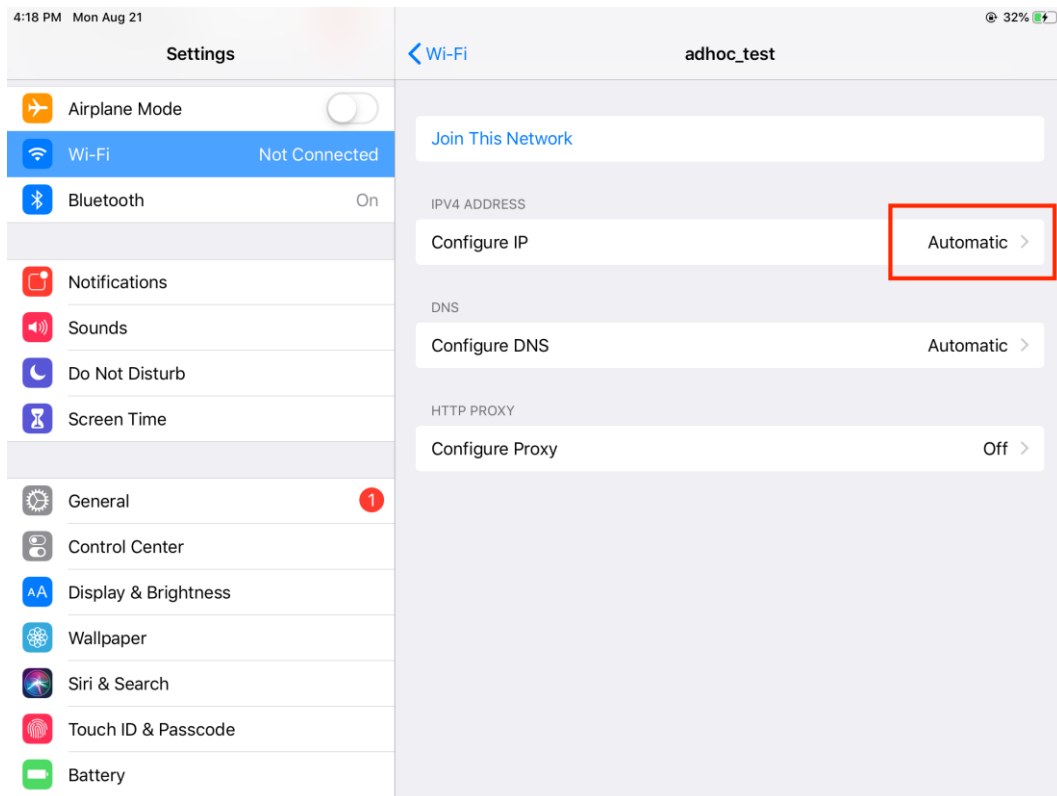
※ self-test print out

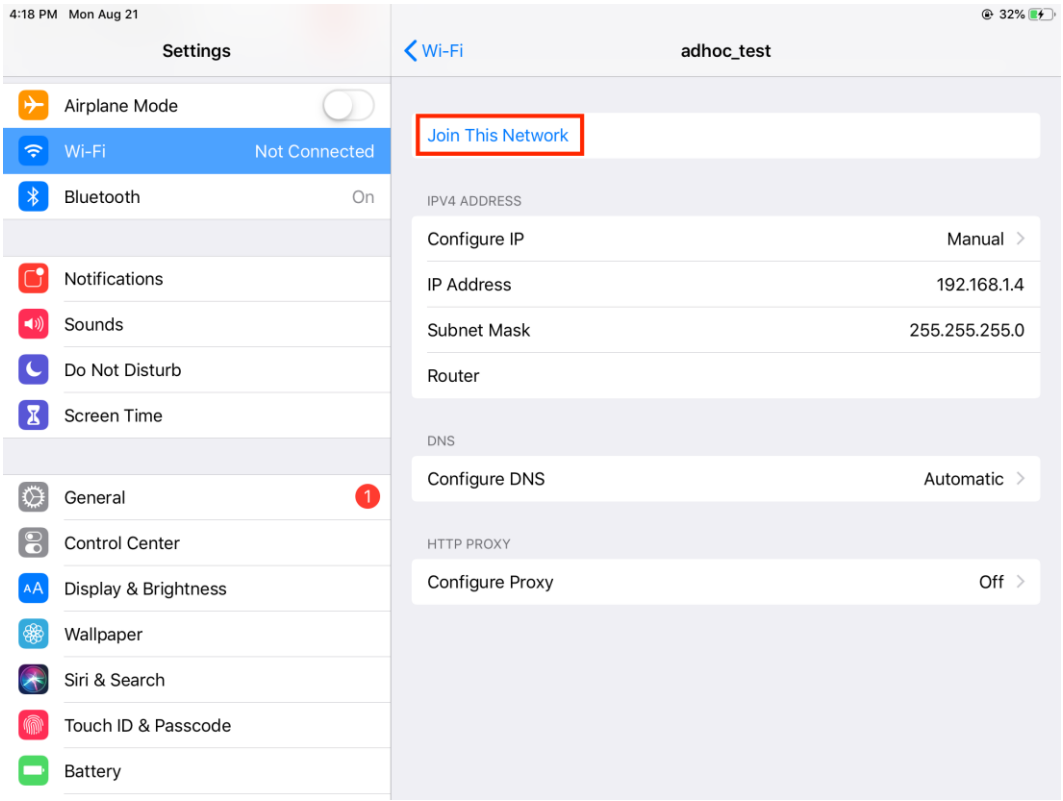
2. Click the " ⓘ " button next to SSID.



3. Assign IP address which consists of four decimal numbers as below.

※ Modify the fourth part if the ip address starts with "192.168.1.x".





## 4. Package Contents

### 4-1 Manual

Manual location/ Name	Description
Manual_METAPACE_SDK_for_iOS_UPOS_compliant_API_Reference Guide_english_Rev_x_xx	Manual in English

### 4-2 Library

Name	Description	Library location
libMetapaceUPOS.a	Printer Controller Library	UPOSSDK/StaticLibrary
frmMetapaceUPOS.xcframework	Printer Controller XCFramework	UPOSSDK/XCFramework

### 4-3 Sample source code

Name	Language
Sample_MultiConnection_ObjC	Objective C
Sample_ MultiConnection_Swift	Swift



## 5. Defines

### 5-1 UPOS Define.h

#### 5-1-1 Printer Search Results Notification Key Name

When calling the printer search API, the following notification occurs.

name	Interface	Description
__NOTIFICATION_NAME_WIFI_WILL_LOOKUP__	WiFi	Start
__NOTIFICATION_NAME_WIFI_FOUND_PRINTER__		Find
__NOTIFICATION_NAME_WIFI_LOOKUP_COMPLETE__		End
__NOTIFICATION_NAME_ETHERNET_WILL_LOOKUP__	Ethernet	Start
__NOTIFICATION_NAME_ETHERNET_FOUND_PRINTER__		Find
__NOTIFICATION_NAME_ETHERNET_LOOKUP_COMPLETE__		End
__NOTIFICATION_NAME_BT_WILL_LOOKUP__	Bluetooth	Start
__NOTIFICATION_NAME_BT_FOUND_PRINTER__		Find
__NOTIFICATION_NAME_BT_LOOKUP_COMPLETE__		End
__NOTIFICATION_NAME_BLE_WILL_LOOKUP__	Bluetooth LE	Start
__NOTIFICATION_NAME_BLE_FOUND_PRINTER__		Find
__NOTIFICATION_NAME_BLE_LOOKUP_COMPLETE__		End

#### 5-1-2 INTERFACE TYPE

Provides a constant value for the target Device's communication interface type.

Name	description
_INTERFACETYPE_WIFI	WiFi
_INTERFACETYPE_ETHERNET	Ethernet
_INTERFACETYPE_BLUETOOTH	Bluetooth
_INTERFACETYPE_BLE	Bluetooth LE

**5-2 UPOS Defines Printer.h****5-2-1 UPOS\_PRINTER\_STATION**

Name	description
PTR_S_RECEIPT	Receipt Printer Type

**5-2-2 UPOS\_PRINTER\_BARCODE\_ALIGNMENT**

Provides a constant value for the Print Barcode API's alignment Param.

Name	description
PTR_BC_LEFT	Left
PTR_BC_CENTER	Center
PTR_BC_RIGHT	Right

**5-2-3 UPOS\_PRINTER\_BARCODE\_TEXT\_POSITION**

Provides a constant value for the Print Barcode API's text Position Param.

Name	description
PTR_BC_TEXT_NONE	Text none
PTR_BC_TEXT_ABOVE	Text Above
PTR_BC_TEXT_BELOW	Text Below



1D Barcode Type Only

**5-2-4 UPOS PRINTER BARCODE SYMBOLOGIES**

Provides a constant value for the Print Barcode API's symbology Param.

Name	description
PTR_BCS_UPCA	UPC-A
PTR_BCS_UPCE	UPC-E
PTR_BCS_EAN8(PTR_BCS_JAN8)	EAN8
PTR_BCS_EAN13(PTR_BCS_JAN13)	EAN13
PTR_BCS_ITF	ITF
PTR_BCS_Codabar	Codabar
PTR_BCS_Code39	Code39
PTR_BCS_Code93	Code93
PTR_BCS_Code128	Code128
PTR_BCS_PDF417	PDF417(2D Type)
PTR_BCS_QRCODE	QRCODE(2D Type)

**5-2-5 UPOS PRINTER BITMAP ALIGNMENT**

Provides a constant value for the Print Bitmap API's alignment Param.

Name	description
PTR_BM_LEFT	Left
PTR_BM_CENTER	Center
PTR_BM_RIGHT	Right

**5-2-6 UPOS PRINTER TRANSACTION**

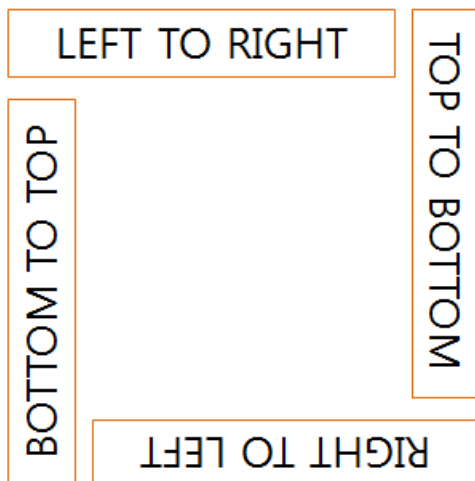
Provides a constant value for the transaction Print API's Param.

Name	description
PTR_TP_TRANSACTION	Transaction Mode Enable
PTR_TP_NORMAL	Transaction Mode Disable

**5-2-7 PAGE\_MODE\_DIRECTION**

Provides a constant value for the set Page mode Direction API's Param.

Name	description
LEFT_TO_RIGHT	No rotation
BOTTOM_TO_TOP	270 degrees
RIGHT_TO_LEFT	180 degrees
TOP_TO_BOTTOM	90 degrees



**5-2-8 UPOS PRINTER STATUS**

Provides a constant value for the status Update Event delegate's Param.

Name	description
PTR_SUE_COVER_OPEN	Printer Cover Open
PTR_SUE_COVER_OK	Printer Cover Close
PTR_SUE_REC_EMPTY	Printer No paper
PTR_SUE_REC_NEAREMPTY	Printer Insufficient paper
PTR_SUE_REC_PAPEROK	Printer Paper OK
PTR_SUE_OFFLINE	Unable to print

**5-2-9 UPOS PRINTER BITMAP WIDTH**

Provides a constant value that sets Print Width Param as the printer width.

Name	description
PTR_BM_ASIS	Automatically set the printer width

## 6. UPOS Devices Reference

This is a reference to use and manage the target device in order to use the printer.

Class	Description
UPOS Device	UPOS Printer/UPOS Cash Drawer Super class
UPOS Devices	UPOS Printers/UPOS Cash Drawers Super class
UPOS Printer	Printer class
UPOS Printers	Printers class
UPOS Cash Drawer	Cash Drawer class
UPOS Cash Drawers	Cash Drawers class



UPOS Devices.h

**6-1 UPOS Device**

@property	Description
NSString *modelName	Device Name
NSString *ldn	Logical Device name
NSNumber *interfaceType	Interface Type(5-1-3_INTERFACE_TYPE)
NSString *address	ip address (case in. wi-fi, Ethernet) Mac address (case in. Bluetooth LE)
NSString *port	Port Number
NSString *bluetoothDeviceName	Bluetooth Device Name



ldn and bluetoothDeviceName are optional. (Not required)

**6-1-1 init With Model Name**

Function to initialize the Wifi / Ethernet / Bluetooth LE device objects.

**[Syntax]**

```
- (instancetype)initWithModelName: (NSString*) modelName
                      interfaceType: (NSNumber*) interfaceType
                      address: (NSString*) address
                      port: (NSString*) port;
```

**[Parameters]**

Type	Name	Description
NSString*	modelName	Printer Name
NSNumber*	interfaceType	Interface Type(5-1-3_INTERFACE_TYPE)
NSString*	address	Printer IP Address (Mac address for Bluetooth LE)
NSString*	port	Printer Port Number

**[Returns]**

Self(UPOSPrinter / UPOSCashDrawer)

**6-2 UPOS Devices****6-2-1 get List**

Returns a list of added devices.

**[Syntax]**

-(NSArray\*)getList;

**[Returns]**

Type	Description
NSArray*	List of added devices

**6-2-2 add Device**

Adds devices.

**[Syntax]**

-(BOOL) addDevice: (id) device;

**[Parameters]**

Type	Name	Description
Id	device	Object containing the information about the devices to add  Adding Printer: UPOSPrinter Type Object Adding Cash Drawer: UPOSCashDrawer Type Object

**[Returns]**

Type	Value	Description
BOOL	NO	Only



**6-2-3 remove Device**

Removes a device registered.

**[Syntax]**

-(BOOL) removeDevice: (id) device;

**[Parameters]**

Type	Name	Description
Id	Device	Object containing the information about added devices that are to be removed.  Removing Printer: UPOSPrinter Type Object Removing Cash Drawer: UPOSCashDrawer Type Object

**[Returns]**

Type	Value	Description
BOOL	YES	Returns if successful.
BOOL	NO	Returns if it fails.

**6-2-4 remove All Devices**

Removes all the devices registered.

**[Syntax]**

-(void) removeAllDevice;

**6-3 UPOS Printer****6-4 UPOS Printers****6-5 UPOS Cash Drawer**

@property	Description
NSNumber *pinNumber	Cash Drawer Pin Number (default value: 2)
NSNumber *pinLevel	Cash Drawer Pin Level (default value: 1)
NSNumber *pulseOnTime	Cash Drawer pluseOnTime (default value: 50)
NSNumber *pulseOffTime	Cash Drawer pluseOffTime (default value: 200)

**6-6 UPOS Cash Drawers**

## 7. UPOS Device Controller Reference

A parent class of the Printer and Cash Drawer device controllers

@property	Description
id<UPOSDeviceControlDelegate> delegate	Delegate
(readonly) UPOSDevice *targetDevice	Connected device object
BOOL DeviceEnabled	Enabled/disabled status (Enabled after Open -> Claim)



UPOSDeviceController.h

### 7-1 get Support Device Strings

- Returns a list of supported devices.

#### [Syntax]

```
-(NSMutableArray*) getSupportDeviceStrings;
```

#### [Returns]

Type	Description
NSMutableArray*	Supported printers

### **7-2 refresh Wifi Lookup**

- Allows searching of available Wi-Fi devices.

#### **[Syntax]**

-(void)refreshWifiLookup;



- UPOS Printer Controller Type Only Supported
- Refer to 5-1-1 for the search results.

### **7-3 refresh Ethernet Lookup**

- Allows searching of available Ethernet devices.

#### **[Syntax]**

-(void) refreshEthernetLookup;



- UPOS Printer Controller Type Only Supported
- Refer to 5-1-1 for the search results.

### **7-4 refresh BT Lookup**

- Allows searching of available Bluetooth devices.

#### **[Syntax]**

-(void) refreshBTLookup;



- UPOS Printer Controller Type Only Supported
- Refer to 5-1-1 for the search results.

### **7-5 refresh BLE Lookup**

- Allows searching of available Bluetooth LE devices.

#### **[Syntax]**

-(void) refreshBLELookup;



- UPOS Printer Controller Type Only Supported
- Refer to 5-1-1 for the search results.

### **7-6 get Registered Device**

- Allows return of a list of added devices.

#### **[Syntax]**

-(UPOSDevices\*) getRegisteredDevice;

#### **[Returns]**

Type	Description
UPOSDevices*	Added device information



Search added devices.

### **7-7 open**

- This method initiates the use of the printer class, and it includes the initialization process such as memory allocation. This method should be called first before calling the claim and other subsequent methods.

#### **[Syntax]**

-(NSInteger)open: (NSString\*) logicalDeviceName;

#### **[Parameters]**

Type	Name	Description
NSString*	logicalDeviceName	the name of the device to be opened

#### **[Returns]**

Name	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.



Only added devices can be opened.

### 7-8 open

- This method initiates the use of the printer class, and it includes the initialization process such as memory allocation. This method should be called first before calling the claim and other subsequent methods.

### [Syntax]

```
-(NSInteger)open: (NSString*) logicalDeviceName  
               address: (NSString*) address ;
```

### [Parameters]

Type	Name	Description
NSString*	logicalDeviceName	the name of the device to be opened
NSString*	Address	the address of the printer to be connected  wifi, ethernet: ip address Bluetooth LE: mac address

### [Returns]

Name	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.



- Only added devices can be opened.
- Supported interfaces: WiFi, Ethernet, Bluetooth LE

### **7-9 claim**

- Claims successfully opened devices.

#### **[Syntax]**

-(NSInteger)claim: (NSInteger) timeout;

#### **[Parameters]**

Type	Name	Description
NSInteger	timeout	Sets the max. time to connection. (ex. 5 seconds, claim:5000)

#### **[Returns]**

Constant Name	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enable.
UPOS_E_ILLEGAL	The requested operation failed.



Only open devices can be claimed.

### **7-10 set Enable Device**

- Enable/disable successfully claimed devices.



setter API for the Device Enabled Property



### **7-11 release Device**

- Releases the claimed devices.

#### **[Syntax]**

-(NSInteger)releaseDevice;

#### **[Returns]**

Constant Name	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enable.
UPOS_E_ILLEGAL	The requested operation failed.



Only claimed devices can be released.

### **7-12 close**

- Closes released devices.

#### **[Syntax]**

-(NSInteger)close;

#### **[Returns]**

Constant Name	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enable.
UPOS_E_ILLEGAL	The requested operation failed.



Only released devices can be closed.

### **7-13 connect**

- Supports Open -> Claim -> setEnableDevice: YES.

#### **[Syntax]**

-(NSInteger)connect: (NSString\*) logicalDeviceName;

#### **[Parameters]**

Type	Name	Description
NSString*	logicalDeviceName	modelName or ldn of the devices to be connected

#### **[Returns]**

Constant Name	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enable.
UPOS_E_ILLEGAL	The requested operation failed.

## 8. UPOS Printer Controller Reference

This is a printer controller class.

Category	Description
UPOSPrinterController	Basic printing functions e.g. text, image, and PDF
UPOSPrinterController(Pagemode)	Page mode function

@property	Description
(readonly) BOOL CapRecPapercut	Cutter support
(readonly) NSInteger RecLineWidth	Width Length
NSInteger RecLineChars	Number of texts that can be printed in a line
(readonly) NSString* RecLineCharsList	Supported RecLineChars List
BOOL AsyncMode	Support Outputcomplete Event when enabled



UPOS Printer Controller.h

## 8-1 UPOS Printer Controller

- Provides basic printing functions.

### 8-1-1 set Character Set

Changes the encoding type of the text.

#### [Syntax]

-(NSInteger)setCharacterSet: (NSInteger) characterSet

#### [Parameters]

Type	Name	Description
NS Integer	Character Set	Code pages for printing characters
		ValueDescription
		437PC437 (USA, Standard Europe)
		737PC737 (Greek)
		775PC775 (Baltic)
		850PC850 (Western Europe, Multilingual)
		852PC852 (Latin 2)
		855PC855 (Cyrillic)
		857PC857 (Turkce)
		858PC858 (Euro)
		860PC860 (Portuguese)
		862PC862 (Hebrew DOS code)
		863PC863 (Canadian-French)
		864PC864 (Arabic)
		865PC865 (Nordic)
		866PC866 (Cyrillic#2)
		874THAI (PC874)
		912PC912 (Albania)
		928PC928 (Greek)
		1250WPC1250 (Czech)
		1251WPC1251 (Cyrillic)
		1252WPC1252 (Latin 1)
		1253WPC1253 (Greek)
		1254WPC1254 (Turkce)
		1255WPC1255 (Hebrew new code)
		1256WPC1256 (Arabic)
		1257WPC1257 (Baltic)
		1258VIETNAM (WPC1258)
		7065FARSI
		7565KATAKANA

## **SDK for iOS UPOS**

		7572	KHMER	
		8411	THAI 11	
		8414	THAI 14	
		8416	THAI 16	
		8418	THAI 18	
		8442	THAI 42	
		3031	TCVN3 (VIETNAM)	
		3032	TCVN3 Capital (VIETNAM)	
		3033	VISCII (VIETNAM)	
		28605	Latin 9	
		5601	KS5601 (Korean)	
		6605	BIG5 (Chinese)	
		2312	GB2312 (Chinese)	
		8374	ShiftJ S (Japanese)	
		54936	GB18030 (Chinese)	
		54937	UTF-8 (Unicode)	
		54938	UTF-16 (Unicode)	
		54939	UTF-32 (Unicode)	

### **[Returns]**

Constant Name	Description
UPOS_SUCCESS	Operation is successful.

**8-1-2 print Normal**

Prints texts.

**[Syntax]**

```
-(NSInteger)printNormal: (NSInteger) station
                    data: (NSString*) data;
```

**[Parameters]**

Type	Name	Description
NSInteger	station	Fixed value PTR_S_RECEIPT
NSString *	data	Specify the data to print. ESC Sequences, carriage returns, line feeds Data and printable characters are permitted.

**ESC Sequences**

Refer to commands that start with ESC(0x1B) + '|' (0x7C).

In the list of commands below, "#" represents a decimal number value and there is a range value for each command. "[" Indicates a command that is optional. If you omit a numeric value, it will become "0". "!" disables the setting.

Any ESC Sequences not supported by the printer are ignored.

Escape Sequence	Support Status & Range of Settings	Description
[#]P	O (# : 0~100)	Cut
[#]fP	O (# : 0~100)	Cut after feeding
[#]lF	O (# : 0~50)	# indicates the number of lines
#fT	O (# : 0, 1, 2)	# sets the font type 0: default font (A), 1 = Font B, 2 = Font C
[!]bC	O	Bold
[!][#]uC	O (# : 1~2)	Underline
[!]rvC	O	Inverted
1C	O	Resize font width by 100% and height by 100%
2C	O	Resize font width by 200% and height by 100%
3C	O	Resize font width by 100% and height by 200%
4C	O	Resize font width by 200% and height by 200%
#hC	O	# times the font width
#vC	O	# times the font height
cA	O	Align center
rA	O	Align right
lA	O	Align left

N	O	Resets the value that can be set as ESC Sequence. <b>Font type A, font width by 100% and height by 100%</b> <b>Disable bold, disable underline, disable inversion</b>
---	---	---

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

8-1-3 cut Paper

Cuts the paper.

**[Syntax]**

-(NSInteger) cutPaper: (NSInteger) percentage feedCut(BOOL)feedCut;

**[Parameters]**

Type	Name	Description
NS Integer	percentage	Full cut or Partial cut. Only models with a cutter support this function.
		Value      Description
		100      Full cut
		90      Partial cut
BOOL	feedCut	Value      Description
		YES      Feed paper to the cutting position and cut it.
		NO      No feed paper.

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**8-1-4 print Barcode**

Prints barcodes.

**[Syntax]**

```
-(NSInteger) printBarcode: (NSInteger) station  
                        data: (NSString*) data  
            symbology: (NSInteger) symbology  
              height: (NSInteger) height  
            barWidth: (NSInteger) barWidth  
            alignment: (NSInteger) alignment  
            textPostion: (NSInteger) textPosition;
```

**[Parameters]**

Type	Name	Description
NS Integer	station	Fixed value PTR_S_RECEIPT
NS String*	data	The data to be included in the barcode may be different from the one permitted for each barcode type.
NS Integer	symbology	Selects barcode type. (5-2-4 UPOS_PRINTER_BARCODE_SYMBOLOGIES Note)
NS Integer	height	Sets the barcode height.
NS Integer	barWidth	Sets the barcode width.
NS Integer	alignment	Selects barcode alignment. (5-2-2 UPOS_PRINTER_BARCODE_ALIGNMENT Note)
NS Integer	textPosition	Sets the position of the text to be printed with the barcode. (5-2-3 UPOS_PRINTER_BARCODE_TEXT_POSITION Note)

**[Return Values]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.



**8-1-5 print Raw Data**

Sends user-defined data to the printer.

**[Syntax]**

```
-(NSInteger) printRawData: (NSInteger) station  
                        data: (NSData*) data;
```

**[Parameters]**

Type	Name	Description
NSInteger	station	Fixed value PTR_S_RECEIPT
NSData *	data	Inputs user-defined data.

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**8-1-6 print Bitmap**

Prints images. (Image data)

**[Syntax]**

```
-(NSInteger) printBitmap: (NSInteger) station  
                        image: (UIImage*) image  
                        width: (NSInteger) width  
                        alignment: (NSInteger) alignment  
                        brightness: (NSInteger) brightness  
compressType: (NSInteger) compressType  
dithering: (BOOL) dithering;
```

**[Parameters]**

Type	Name	Description
NSInteger	station	Fixed value PTR_S_RECEIPT
NSString*	image	Sets the path of an image file.
NSInteger	width	Sets the width of an image.
NSInteger	alignment	Selects the alignment of an image.
NSInteger	brightness	Sets the brightness level of an image. (Range: 0 ~ 100)
NSInteger	compressType	Compression type (None:0, RLE:1, LZMA: 2)
BOOL	dithering	Applies an image dithering or not

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**8-1-7 print PDF**

Prints PDF files.

**[Syntax]**

```
-(NSInteger) printPDF: (NSInteger) station  
                    fileName: (NSString*) fileName  
                    width: (NSInteger) width  
                    alignment: (NSInteger) alignment  
                    page: (NSInteger) page  
                    brightness: (NSInteger) brightness  
                    compressType: (NSInteger) compressType  
                    dithering: (BOOL) dithering;
```

**[Parameters]**

Type	Name	Description
NSInteger	station	Fixed value PTR_S_RECEIPT
NSString *	fileName	Sets the path of a PDF file.
NSInteger	width	Sets the width of a PDF file.
NSInteger	alignment	Selects the alignment of an image.
NSInteger	page	Sets the page number of a PDF file to print. (starts from 1)
NSInteger	brightness	Sets the brightness level. (0 ~ 100)
NSInteger	compressType	Compression type (None:0, RLE:1, LZMA: 2)
BOOL	dithering	Applies an image dithering or not

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**8-1-8 transaction Print**

Prints in Transaction Mode.

**[Syntax]**

```
-(NSInteger)transactionPrint: (NSInteger) station  
                        control: (NSInteger) control;
```



When changing from transaction mode to normal mode, print data is sent to the printer, and 'OutputCompleteEvent' occurs when the transfer is complete.

**[Parameters]**

Type	Name	Description
NSInteger	station	Fixed value PTR_S_RECEIPT
NSInteger	control	Transaction Mode (5-2-6 UPOS_PRINTER_TRANSACTION Note)

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**8-2 UPOS Printer Controller (Pagemode)**

- Categorize API in Page Mode.

**8-2-1 set Page Area**

Sets the Page Mode area.

**[Syntax]**

- (NSInteger)setPageArea: (NSInteger) startingX  
startingY: (NSInteger) startingY  
width: (NSInteger) width  
height: (NSInteger) height;

**[Parameters]**

Type	Name	Description
NSInteger	startingX	Sets the X coordinate.
NSInteger	startingY	Sets the Y coordinate.
NSInteger	width	Sets the width.
NSInteger	height	Sets the height.

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**8-2-2 set Page Mode Direction**

Sets a print direction in Page Mode.

**[Syntax]**

- (NSInteger)setPageModeDirection: (PAGE\_MODE\_DIRECTION) direction;

**[Parameters]**

Type	Name	Description
PAGE_MODE_DIRECTION	Direction	Sets a print direction. (5-2-7 PAGE_MODE_DIRECTION Note)

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**8-2-3 set Left Position**

Sets the starting position (X coordinate).

**[Syntax]**

- (NSInteger)setLeftPosition: (NSInteger) positionX

**[Parameters]**

Type	Name	Description
NSInteger	positionX	Sets the starting position (X coordinate).

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**8-2-4 set Vertical Position**

Sets the starting position (Y coordinate).

**[Syntax]**

- (NSInteger)setVerticalPosition: (NSInteger) positionY;

**[Parameters]**

Type	Name	Description
NSInteger	positionY	Sets the starting position (Y coordinate).

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**8-2-5 print Data In Page Mode**

Sets the starting position (Y coordinate).

Prints the data saved to the buffer in Page Mode.

**[Syntax]**

- (NSInteger)printDataInPageMode;

**[Returns]**

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.



## 9. UPOS CD Controller Reference

This is a Cash Drawer controller class.

Category	Description
UPOSCDController	Cash Drawer Open function

@property	Description
(readonly) BOOL IsHighSignal	Cash drawer signal : Low or High



UPOSCDController.h

### 9-1 UPOS CD Controller



- Cash Drawer can be opened after Open-Claim-Enable the UPOS Printer Controller.
- Cash Drawer can be controller after Add Device-Open-Claim-Enable in the UPOS CD Controller.

#### 9-1-1 Open Drawer

Opens the Cash Drawer.

#### [Syntax]

-(NSInteger) openDrawer;

#### [Returns]

Constant Value	Description
UPOS_SUCCESS	Operation is successful.
UPOS_E_CLOSED	Device to access is closed.
UPOS_E_NOTCLAIMED	Claim method should be called first.
UPOS_E_DISABLED	Not enabled.
UPOS_E_ILLEGAL	The requested operation failed.

**10. UPOS Device Control Delegate Reference**

This is a class that provides delegate occurred in the printer.



UPOSDeviceControlDelegate.h

**10-1 Output Complete Event**

Triggers the print complete event. In this case, it should be used in Async Mode.

Constant Value (when an event is triggered)
(NSNumber*) 0

**10-2 Status Update Event**

The event is triggered whenever the printer changes its status.  
(Note 5-2-8 UPOS\_PRINTER\_STATUS)