

Class

XWIFIManager

The `XWIFIManager` class handles network communication and provides callback blocks for printer-related statuses and operations.

```
import Foundation
import Cocoa
class NetProfile
```

XWIFIManagerDelegate

Delegate Methods

xWifiConnected(toHost:port:)

Callback for successful connection

```
func xWifiConnected(
    toHost host: String!,
    port: UInt16
)
```

- **host**
The host address of the connected device.
- **port**
The port number used for the connection.

xWifiDisconnectWithError(_:host:)

Callback for disconnection error

```
func xWifiDisconnectWithError(
    _ error: Error!,
    host: String!
)
```

- **error**
The error information related to the disconnection.
- **host**
The host address of the connected device.

xWifiWriteValue(withTag:host:)

Callback for successful data transmission

```
func xWifiWriteValue(  
    withTag tag: Int,  
    host: String!  
)
```

- **tag**
The tag value associated with the data transmission.
- **host**
The host address of the connected device.

xWifiReceiveValue(for:host:)

Callback for receiving data from the printer

```
func xWifiReceiveValue(  
    for data: Data?,  
    host: String?  
)
```

- **data**
The received data.
- **host**
The host address of the connected device.

Properties

Network Communication Class

```
class XWIFIManager: NSObject
```

- **hostStr**
The host address string of the printer.

```
var hostStr: String?
```

- **port**
The port number used for communication.

```
var port: UInt16 = 0
```

- **isConnect**

The connection status of the printer.

```
var isConnect: Bool = false
```

- **delegate**

The delegate to receive WiFi manager events.

```
weak var delegate: XWiFiManagerDelegate?
```

- **receiveBlock**

The callback block called when data is received.

```
var receiveBlock: XWiFiManagerReceiveBlock?
```

- **writeBlock**

The callback block called when data is written.

```
var writeBlock: XWiFiManagerWriteBlock?
```

- **statusPOSBlock**

The callback block called when reporting POS printer status.

```
var statusPOSBlock: XWiFiPOSPrinterStatusBlock?
```

- **statusLabelBlock**

The callback block called when reporting label printer status.

```
var statusLabelBlock: XWiFiLabelPrinterStatusBlock?
```

- **snBlock**

The callback block called when reporting the printer's serial number.

```
var snBlock: XWiFiPrinterSNBlock?
```

- **checkBlock**

The callback block called when checking POS printer status.

```
var checkBlock: XWiFiPrinterCheckBlock?
```

- **foundPrinterBlock**

The callback block called when a printer is discovered.

```
var foundPrinterBlock: XWiFiManagerFoundPrinterBlock?
```

- **cashBoxBlock**

The callback block called when reporting cash drawer status.

```
var cashBoxBlock: XWIFICashBoxBlock?
```

- **connectedPrinter**

The configuration profile of the connected printer.

```
var connectedPrinter: NetProfile?
```

Methods

Singleton Object

```
static let sharedInstance = XWIFIManager()
```

Remove a Delegate Object

```
func removeDelegate(_ delegate: XWIFIManagerDelegate!)
```

- **delegate**

The delegate object to be removed.

Remove All Delegate Objects

```
func removeAllDelegates()
```

Connect to the Printer Address

```
func connect(  
    withHost hostStr: String!,  
    port: UInt16  
)
```

- **hostStr**

The host address of the printer.

- **port**

The port number to connect to.

Disconnect from the Printer

```
func disconnect()
```

Write Commands to the Printer

```
func writeCommand(with data: Data!)
```

- **data**
The command data to be sent.

Write Commands with a Receive Callback

```
func writeCommand(  
    with data: Data!,  
    receiveCallBack receiveBlock: XWIFIManagerReceiveBlock!  
)
```

- **data**
The command data to be sent.
- **receiveBlock**
The callback block when data is received.

Write Commands with a Write Completion Callback

```
func writeCommand(  
    with data: Data!,  
    writeCallBack writeBlock: XWIFIManagerWriteBlock!  
)
```

- **data**
The command data to be sent.
- **writeBlock**
The callback block when the write operation is complete.

Create a UDP Socket

```
func createUdpSocket() -> Bool
```

- **return**
Whether the socket creation was successful.

Close the UDP Socket

```
func closeUdpSocket()
```

Search for Printers

```
func sendFindCmd(_ foundPrinterBlock: XWIFIManagerFoundPrinterBlock!)
```

- **foundPrinterBlock**

A callback block that is triggered when a printer is found. It returns a `NetProfile` object containing the following printer network configuration details:

- `printerName`: The name of the printer. The default is the printer's MAC address.
- `printerDesc`: A description of the printer, typically used to provide basic information about the printer.
- `modelName`: The printer's model name. Some printers may not provide this, so it might be `nil` or unset.
- `MAC`: The printer's MAC address, used to uniquely identify the device on the network.
- `IP`: The printer's IP address, used for network communication to locate the device.
- `mask`: The printer's subnet mask, used for defining the network's range.
- `gateway`: The printer's gateway address, used for routing network traffic.
- `DHCP`: The printer's DHCP setting, indicating whether dynamic IP assignment is enabled. `1` means enabled, `0` means disabled.

Set IP Configuration

```
func setIPConfigWithIP(  
    _ ip: String!,  
    mask: String!,  
    gateway: String!,  
    dhcp: Bool  
)
```

- **ip**
The IP address of the printer.
- **mask**
The subnet mask.
- **gateway**
The gateway address.
- **dhcp**
Whether to enable DHCP.

Set WiFi Configuration

```
func setWiFiConfigWithIP(  
    _ ip: String!,  
    mask: String!,  
    gateway: String!,  
    ssid: String!,  
    password: String!,  
    encrypt: UInt  
)
```

- **ip**
The IP address of the printer.
- **mask**
The subnet mask.
- **gateway**
The gateway address.
- **ssid**
The WiFi SSID.
- **password**
The WiFi password.
- **encrypt**
The encryption method.

Check Printer Connection Status

```
func printerIsConnect() -> Bool
```

- **return**
Whether the printer is connected.

Get Printer Serial Number

```
func printerSN(_ snBlock: XWIFIPrinterSNBlock!)
```

- **snBlock**
The callback block for the serial number.

Check Printer Status (POS Printers)

```
func printerPOSStatus(_ statusBlock: XWIFIPOSPrinterStatusBlock!)
```

- **statusBlock**
The callback block for the POS printer status.

Check Printer Status (Label Printers)

```
func printerLabelStatus(_ statusBlock: XWIFILabelPrinterStatusBlock!)
```

- **statusBlock**
The callback block for the label printer status.

Query Printer Status

```
func printerPOSCheck(  
    _ type: Int32,  
    check checkBlock: XWIFIPrinterCheckBlock!  
)
```

- **type**
The query type (1: common status, 2: offline status, 3: error status, 4: paper feed status).
- **checkBlock**
The callback block for the query result.

Check Cash Drawer Status

```
func cashBoxCheck(_ cashBoxBlock: XWIFICashBoxBlock!)
```

- **cashBoxBlock**
The callback block for the cash drawer status.

Get Copyright Information

```
class func getCopyRight() -> String!
```

- **return**
The copyright information string.