

1. 概述

该文档用于介绍联想图像打印机蓝牙配网的总体流程，以及对应API的使用说明。联想图像打印机蓝牙配网解决方案以SDK的方式提供，可供上层应用开发工程师快速熟悉蓝牙配网流程，从而快速集成到产品中。

2. 适用人员

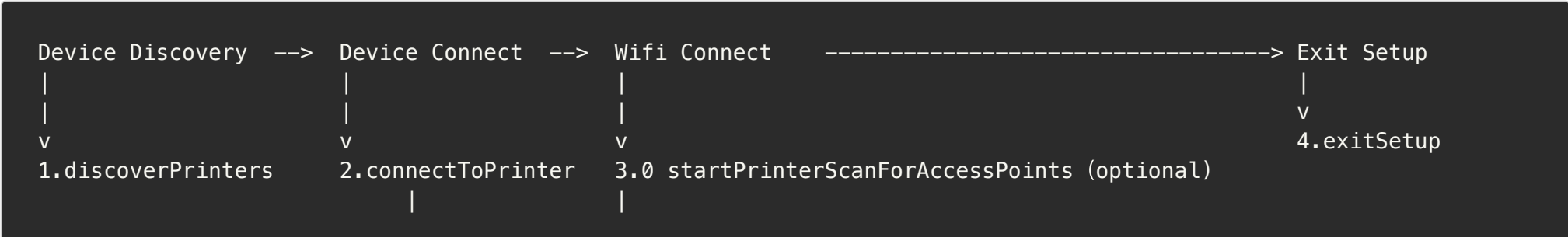
- Android应用开发工程师和测试人员
- iOS应用开发工程师和测试人员
- 小程序蓝牙配网开发工程师和测试人员
- 产品经理

2. 流程介绍

打印机蓝牙配网主要包括下面四部分内容

1. 设备发现 - 查找待配网打印机。
2. 设备连接 - 蓝牙连接到待配网打印机。
3. 设备配网 - 连接打印机到指定网络（仅支持2.4G, 暂不支持5G）。
4. 退出配网 - 断开与打印机的蓝牙连接并释放持有的资源。

总体流程如下图所示：



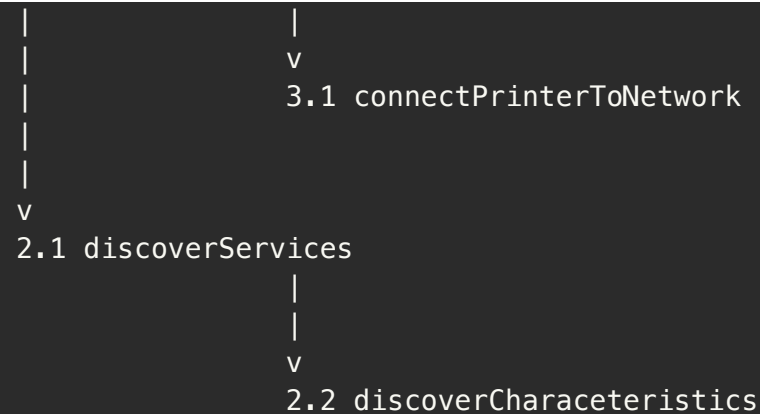


图1. 蓝牙配网总体流程

3. 配网实现

配网解决方案在实现过程中总体遵循1-2-4的原则，预期达到可用、易用、高可扩展、低耦合的目标。

- 1, 一致的API接口。
- 2, 支持iOS和Android两个平台。
- 4, 兼容四种开发语言，Swift、Objective-C, Kotlin、Java。

SDK使用LeBlePrinterSetupManager抽象配网总体流程并对外暴露配网接口供应用层调用。配网结果基于异步回调的事件驱动方式实现，应用层只需注册相应回调接口并监听相应的回调事件以刷新界面，即可完成配网。

3.1 初始化

接口说明

```
public class func create(  
    queue: DispatchQueue                                = .main,  
    deviceScanCallback: OnLeBleDeviceScanCallback?    = nil,  
    deviceConnectCallback: OnLeBleDeviceConnectCallback? = nil,  
    wifiScanCallback: OnLeBleAccessPointsScanCallback? = nil,  
    wifiConnectCallback: OnLeBleWifiConnectCallback?  = nil
```

参数说明

- `queue`, 可选, 回调事件处理消息队列, 用于指定回调事件所在的线程, 默认为主线程。
- `deviceScanCallback`, 可选, 设备扫描事件回调接口, 定义见 `OnLeBleDeviceScanCallback`。
- `deviceConnectCallback`, 可选, 设备连接事件回调接口, 定义见 `OnLeBleDeviceConnectCallback`。
- `wifiScanCallback`, 可选, 配网结果事件回调接口, 定义见 `OnLeBleAccessPointsScanCallback`。
- `wifiConnectCallback`, 可选, 设备连接事件回调接口, 定义见 `OnLeBleWifiConnectCallback`。

初始化 `LeBlePrinterSetupManager` 并注册相关的回调接口。

```
setupManager = LeBlePrinterSetupManagerFactory.create(.main, nil, nil, nil, nil)
```

事件回调接口也可以通过 `setupManager` 接口设置。适用于多页面实现, 单个页面实现不同的回调接口。

```
setupManager.setDeviceScanCallback(delegate: self)  
setupManager.setDeviceConnectCallback(delegate: self)  
setupManager.setWifiScanCallback(delegate: self)  
setupManager.setWifiConnectCallback(delegate: self)
```

3.2 设备发现

接口说明

`discoverPrinters`用于发现待配网打印机，接口定义如下：

```
func discoverPrinters(options: LeBleDeviceScanOptions)
```

参数说明

- `options`, 设备搜索可选配置项。

`LeBleDeviceScanOptions`定义如下：

- `timeoutInSeconds`, 设备搜索超时时间，默认为5秒。
- `matchers`, 设备过滤器列表，用于过滤满足指定条件的设备，默认为空。
- `showBluetoothAlert`, (iOS Only)是否弹窗提示用户开启蓝牙，默认为`true`。若配置为`false`，可在`onDeviceScanFailed`错误回调中处理相应错误。

设备扫描结果通过注册的`deviceScanCallback`接口返回，接口定义如下：

```
public protocol OnLeBleDeviceScanCallback {  
    @objc optional func onDeviceScanChanged(newDevice: LeBlePrinter)  
    @objc optional func onDeviceScanFinished(discoveredDevices: Array<LeBlePrinter>)  
    @objc optional func onDeviceScanFailed(error: LeBleDeviceScanError)  
    @objc optional func onDeviceScanStateChanged(from oldState: LeBleDeviceScanState, to newState:  
LeBleDeviceScanState)  
}
```

各回调函数描述如下：

- `onDeviceScanChanged`, 发现新的设备。
- `onDeviceScanFinished`, 设备发现完成。
- `onDeviceScanFailed`, 设备发现失败，失败原因见`LeBleDeviceScanError`
- `onDeviceScanStateChanged`, 设备发现状态改变，状态描述见`LeBleDeviceScanState`

LeBleDeviceScanError

错误类别	错误代码	错误原因	备注
Timeout	1	扫描超时	扫描请求超时，未找到设备
BtUnknown	10	未知错误	未知初始化错误
BtUnsupported	12	不支持错误	不支持BLE设备扫描
BtUnauthorized	13	未授权错误	蓝牙权限未开启
BtPoweredOff	14	蓝牙未打开	蓝牙开关未打开

LeBleDeviceScanState

状态类别	状态码	描述
Initailized	0	初始化完成
Idle	1	扫描就绪，可接收新的扫描请求
Accepted	2	接收到了新的扫描请求但扫描尚未启动
Waiting	3	等待扫描就绪
Scanning	4	扫描进行中
Failed	5	扫描失败
Finished	6	扫描完成

示例代码

1. 搜索并返回所有可配网打印机。



```
setupManager.discoverPrinters(options: LeBleDeviceScanOptions())
```

2. 搜索并返回名称为M7360DNW-000017, 信号强度高于-50dbm的待配网打印机。

```
class RssiMatcher: LeBleDeviceMatcher {  
    func matches(printer: LeBlePrinter) -> Bool {  
        return printer.rssi > -50  
    }  
}  
  
class PrinterNameMatcher: LeBleDeviceMatcher {  
    func matches(printer: LeBlePrinter) -> Bool {  
        return printer.name == "M7360DNW-000017"  
    }  
}  
  
let options = LeBleDeviceScanOptions()  
options.matchers = [RssiMatcher(), PrinterNameMatcher()]  
setupManager.discoverPrinters(options: options)
```

3.3 连接到设备

接口说明

connectToPrinter用于连接到目标打印机, 接口定义如下:

```
func connectToPrinter(_ printer: LeBlePrinter, with options: LeBleDeviceConnectOptions)
```

参数说明

- `printer`, 目标打印机, 为设备扫描时返回的打印机。
- `options`, 设备连接可选配置项。

`LeBleDeviceConnectOptions`定义如下:

- `timeoutInSeconds`, 设备搜索超时时间, 默认为5秒。
- `maxRetryTimes`, 连接失败最大重试次数, 默认为3次。

设备连接结果在注册的`deviceConnectCallback`回调接口中返回, 接口定义如下:

```
@objc public protocol OnLeBleDeviceConnectCallback {
    @objc optional func onDeviceConnected(reconnect: Bool)
    @objc optional func onDeviceConnectFailed(error: LeBleDeviceConnectError)
    @objc optional func onDeviceDisconnected()
    @objc optional func onServicesDiscoveryFinished(services: Array<CBService>)
    @objc optional func onCharacteristicsDiscoveryFinished(service: CBService, characteristics:
Array<CBCharacteristic>)
    @objc optional func onServiceDiscoveryCompleted()
}
```

- `onDeviceConnected`, 成功连接到打印机。参数 `reconnect` 为 `false` 表示初次连接, 否则表示再次连接 (在已连接的基础上再次发起连接请求, 这种情况下会快速返回)。
- `onDeviceConnectFailed`, 连接打印机失败, 失败原因见 `LeBleDeviceConnectError`。
- `onDeviceDisconnected`, 与打印机的连接已断开。
- `onServicesDiscoveryFinished`, 服务发现完成, 进一步发现服务关联的属性。
- `onCharacteristicsDiscoveryFinished`, 属性发现完成。
- `onServiceDiscoveryCompleted`, 服务和属性全部发现完成。此后可进行设备配网, 应用需在此回调函数之后调用 `connectPrinterToNetwork`(或 `startPrinterScanForAccessPoints`)。

注: 设备连接回调接口方法标记为 `optional` 表示非必须实现 (SDK提供了默认实现), 应用可根据需要选择是否重载。

设备连接错误 - `LeBleDeviceConnectError`

错误类别	错误代码	错误原因	备注
Timeout	0	连接超时	无
Undefined	1	未定义	无

示例代码

```
setupManager.connectToPrinter(printer, with: LeBleDeviceConnectOptions())
```

3.4 设备网络扫描

接口说明

`startPrinterScanForAccessPoints`用于启动打印机网络扫描。

```
func startPrinterScanForAccessPoints(options: LeBleWifiScanOptions)
```

参数说明

- `options`, 设备AP扫描可选配置项。

`LeBleWifiScanOptions`定义如下：

- `timeoutInSeconds`, 扫描请求超时时间，默认为30秒。
- `maxNumOfScanResults`, 最大接收AP数目，默认为20。

设备AP扫描结果在注册的`wifiScanCallback`回调接口中返回, 接口定义如下：


```

@objc
public protocol OnLeBleAccessPointsScanCallback {
    @objc optional func onNewAccessPointAvailable(ap: LeAccessPointScanRecord)
    @objc optional func onAccessPointsDiscovered(discoveredAps: Array<LeAccessPointScanRecord>)
    @objc optional func onAccessPointsScanFailed(error: LeBlePrinterSetupError)
}

```

各回调函数描述如下:

- `onNewAccessPointAvailable`, AP扫描结果发生了变化, 收到一条新的AP结果。
- `onAccessPointsDiscovered`, AP扫描完成, 收到全部的AP结果。
- `onAccessPointsScanFailed`, AP扫描失败, 错误信息如`LeBleApScanError`描述。

LeBleApScanError

错误类别	错误代码	错误原因	备注
PrinterNoConnected	0	打印机未连接	需要首先调用 <code>connectToPrinter</code> 接到目标打印机
WifiCfgNoSupported	1	设备不支持配网	
EnableNtfFailed	2	开启蓝牙通知失败	
Failed	3	打印机测扫描失败	
TimedOut	4	请求超时	

示例代码

```

setupManager.startPrinterScanForAccessPoints(options: LeBleWifiScanOptions())

```

3.5 设备配网

接口说明

`connectPrinterToNetwork`用于配置网络信息，连接打印机到指定的网络。

```
func connectPrinterToNetwork(options: LeBleWifiConnectOptions)
```

参数说明

- `options`, 设备配网可选配置项。

`LeBleWifiScanOptions`定义如下：

- `timeoutInSeconds`, 连接超时时间，默认为30秒。
- `ssidName`, 目标网络ssid名称。
- `password` , 目标网络ssid密码。

设备配网结果在注册的`wifiConnectCallback`回调接口中返回，接口定义如下：

```
@objc
public protocol OnLeBleWifiConnectCallback {
    func onWifiConnected(info: LeWifiConnectResultInfo)
    func onWifiConnectFailed(error: LeBleWifiConnectError)
}
```

- `onWifiConnected`, 设备已连接到目标网络，可通过参数`info`查看打印机相关信息，包括`ssid`, `SN`和`IP`地址。
- `onWifiConnectFailed`, 设备配网失败，原因如`LeBleWifiConnectError`描述。

LeBleWifiConnectError

错误类别	错误代码	错误原因	备注
------	------	------	----

错误类别	错误代码	错误原因	备注
PrinterNoConnected	0	打印机未连接	需要首先调用 <code>connectToPrinter</code> 接到目标打印机
WifiCfgNoSupported	1	设备不支持配网	
EnableNtfFailed	2	开启蓝牙通知失败	
AccessPointNotFound	3	没找到目标网络	
TimedOut	4	连接超时	
IncorrectPassword	5	密码错误	
Unknown	0xFF	未知错误	

示例代码

```
let options = LeBleWifiConnectOptions();
options.ssidName = "YOUR_SSID"
options.password = "YOUR_PSWD"
setupManager.connectPrinterToNetwork(options: options)
```

3.6 退出配网

`exitSetup`用于断开与打印机的蓝牙连接并释放相关资源。

```
func exitSetup()
```

注：为避免连接相关的问题和资源泄漏，每次配网完成后需调用该接口以释放相关资源。

4. 权限声明

蓝牙设备发现和连接需要申请蓝牙相关的具体权限。

- iOS

```
<key>NSBluetoothAlwaysUsageDescription</key>
```

- Android

```
// 1. 用于启动蓝牙设备扫描
- android.permission.BLUETOOTH_ADMIN
- android.permission.android.permission.BLUETOOTH

// 2. 用于获取设备扫描结果
android.permission.ACCESS_COARSE_LOCATION
android.permission.ACCESS_FINE_LOCATION

// 3. 用于连接到蓝牙设备进行信息收发
android.Manifest.permission.BLUETOOTH_CONNECT
```

5. 附录

除了[LeBlePrinterSetupManager](#), SDK还提供了[LeBleSetupHelper](#)配网助手, 以简化设备发现和配网流程。

1. 注册回调接口



```
LeBleSetupHelper.shared.setDeviceScanCallback(delegate: self)
LeBleSetupHelper.shared.setDeviceConnectCallback(delegate: self)
LeBleSetupHelper.shared.setWifiScanCallback(delegate: self)
LeBleSetupHelper.shared.setWifiConnectCallback(delegate: self)
```

2. 发现设备

```
LeBleSetupHelper.shared.discoverPrinters(options: LeBleDeviceScanOptions())
```

3. 连接设备

```
LeBleSetupHelper.shared.connectToPrinter(printer, with: LeBleDeviceConnectOptions())
```

4. 设备网络扫描

```
LeBleSetupHelper.shared.startPrinterScanForAccessPoints(options: LeBleWifiScanOptions())
```

5. 设备配网

```
let opts = LeBleWifiConnectOptions()
opts.ssidName = TARGET_NETWORK_SSID
opts.password = TARGET_NETWORK_PSWD
LeBleSetupHelper.shared.connectPrinterToNetwork(options: opts)
```

6. 退出配网

```
LeBleSetupHelper.shared.exitSetup()
```