

[Network Extension](#) / Configuring a Wi-Fi accessory to join a network

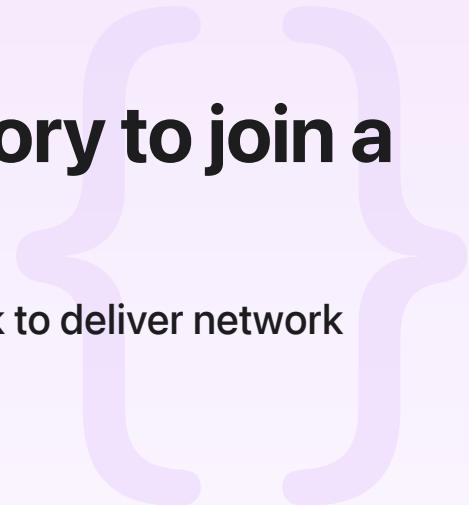
Sample Code

Configuring a Wi-Fi accessory to join a network

Associate an iOS device with an accessory's network to deliver network configuration information.

[Download](#)

iOS 14.2+ | iPadOS 14.2+ | macOS 11.0+ | Xcode 12.4+



Overview

When you're building a Wi-Fi accessory, getting it to join the user's network can be a challenge. The best way to accomplish this is by using [EAWiFiUnconfiguredAccessoryBrowser](#) in the MFi Wireless Accessory Configuration process. If you can't use that process, this sample demonstrates an alternative method.

The sample uses a macOS app to act as an accessory and an iOS app running on an iOS device to configure that accessory. The goal for the iOS app is to deliver network information to the accessory so it can join the user's standard Wi-Fi network. To do that, the iOS app needs to deliver the user's network SSID and its passphrase to the accessory. After receiving this information, the accessory can use it to associate itself with the same network as the iOS device.

Configure the sample code project

After downloading the sample, follow these steps to configure it for use:

1. The macOS target is BrandXAccessory.
- In macOS, set up internet sharing (see [Share the internet connection on Mac with other network users](#)).

- Set the SSID to AccessoryWiFi and the passphrase to embedded1.
 - Configure Internet Sharing to share your connection from Ethernet to computers using Wi-Fi.
2. The iOS target is BrandXAccessorySetup, and requires the use of a physical device.
3. When running the iOS target, you need to change the bundle identifier so that Xcode can create a unique provisioning profile with the project entitlements for your team. Do this in the Xcode Signing & Capabilities pane by altering the Bundle Identifier text field.

Run the sample code project

The macOS app creates a QR code and a network listener. The QR code contains the accessory network information. The iOS app scans the QR code and sends that network information to the BrandXAccessory app for network configuration.

1. Build and run the BrandXAccessory app.
2. Build and run the BrandXAccessorySetup app on an iOS device, and associate the device with the user's network.
3. In the BrandXAccessory app, click Next to embed the accessory network information in a QR code. A QR code appears with the following encoded network information:
 - SSID of the accessory network: AccessoryWiFi
 - Passphrase of the accessory network: embedded1
 - Service URL for NWConnection
 - TLS PSK
 - TLS PSK identity
4. In the iOS app, tap Capture Accessory Network Info to bring up a QR code capture session.
5. Point the iOS device's camera at the QR code in the BrandXAccessory app to capture, decode, and save the QR data. Tap Next.
6. In the iOS app, tap Get Network Info to capture the user's network SSID. Tap Next.
7. In the iOS app, tap Associate to Accessory Network. NEHotspotConfiguration creates a Wi-Fi configuration with the accessory's network data from the QR code, applies it to the device, and updates the device's screen when the association finishes. Tap Next.
8. In the iOS app, tap Confirm Network to confirm the device's association with the accessory network. Tap Next.
9. In the BrandXAccessory app, click Start Listener so the NWListener begins listening for incoming connections from the iOS app.

0. In the iOS app, tap Enter Network Passphrase and enter the passphrase for the user's network.
Tap Set Passphrase.
 1. In the iOS app, tap Send Network Data. The sample sends the passphrase to the Brand XAccessory app, along with the user's network SSID.
 2. In the BrandXAccessory app, the view displays the SSID and passphrase of the user's network.
 3. In the iOS app, tap Next, and then tap Remove Network Configuration to disassociate from the accessory network and rejoin the user's network.
-

See Also

Wi-Fi management

Wi-Fi configuration

Add persistent Wi-Fi configurations, or temporarily move the device to a specific Wi-Fi network.

Hotspot helper

Integrate your app with the iOS hotspot network subsystem.