

# Wifi P2p连接步骤整理

版权声明：本文为博主原创文章，未经博主允许不得转载。

原文：<http://developer.android.com/training/connect-devices-wirelessly/wifi-direct.html>

标题：Creating P2P Connections with Wi-Fi

刚开始看P2p连接的时候，感觉头绪很多，步骤很乱，弄不清楚来龙去脉。现在用自己的话理了一下。

## P2P连接步骤

设置访问wifi的许可

创建一个Intent过滤器A，接收如下信息

WIFI\_P2P\_STATE\_CHANGED\_ACTION Indicates whether Wi-Fi P2P is enabled

WIFI\_P2P\_PEERS\_CHANGED\_ACTION Indicates that the available peer list has changed.

WIFI\_P2P\_CONNECTION\_CHANGED\_ACTION Indicates the state of Wi-Fi P2P connectivity has changed.

WIFI\_P2P\_THIS\_DEVICE\_CHANGED\_ACTION Indicates this device's configuration details have changed.

创建本线程到WifiP2p框架的通道A

创建一个接收器A，使用前面提及的过滤器A

在Activity的onResume()中挂入接收器A，在onPause()中取消接收器A

~~~~~ 准备工作结束，下面启动扫描 ~~~~~

调用P2pManger的discoverPeers，启动扫描。在其中传入前面的通道，和一个监听器A，用于监听是否成功启动扫描任务（而不是有没有扫描到某个设备）

~~~~~ 下面接收扫描结果 ~~~~~

创建PeerListListener接口的实例A，在接口函数中接收扫描到的设备列表

修改前面的接收器A，使其在收到WIFI\_P2P\_PEERS\_CHANGED\_ACTION时调用requestPeers()，在其中传入实例A和通道A

疑问：不知道何时会调用实例A，需要自己检测嘛？

~~~~~ 准备连接到一个peer ~~~~~

创建一个WifiP2pConfig配置A，在里面填入某个peer的信息

调用connect()函数，在其中传入通道A、配置A、一个监听器B，用于监听是否成功启动p2p握手

任务（而不是有没有最终握到手）

创建WifiP2pManager.ConnectionInfoListener接口的实例B，在接口函数onConnectionInfoAvailable()中获取InetAddress地址和小组的owner信息

再次返回到前面的接收器A，使其在收到WIFI\_P2P\_CONNECTION\_CHANGED\_ACTION时调用requestConnectionInfo()，在其中传入实例B和通道A

坐等接口函数onConnectionInfoAvailable()被调用，其后可以根据InetAddress建立socket等连接  
P2p连接总结

一共是三步：

启动p2p扫描

接收扫描结果，内含附近p2p设备

连接到其中一台设备

注意全是异步操作，注册回调函数来得到结果