# 第 5 章：连接到接入点(AP)

## 目标

本章最后，您将了解到以 Wi-Fi Station(STA)模式运行和连接 Wi-Fi 接入点 (AP) 这两部分的基本原理。您将初步了解 TCP/IP 网络协议栈，并对网络协议栈的开放系统互连(OSI)参考模型的前三层（即物理层、数据链路层和网络层）有一个基本理解。您还将了解用来处理连接和加密的 Wi-Fi 数据链路层。最后，您会了解到一些 IP 网络方面的基础知识（地址、网络掩码）以及 WICED 设备配置表(DCT，Device Configuration Table) 的作用。

**最重要的是，您将能够利用 WICED 将物联网 (IoT) 设备连接到 Wi-Fi 网络。**

## 时间：1½ 小时

## 基础

### TCP/IP 网络协议栈

几乎所有复杂系统都是将系统分层，以便管理整体复杂性。“网络协议栈”,或更准确地说，“TCP/IP 网络协议栈”正是这种层级系统，可通过多种网络介质(Wi-Fi、以太网等)实现可靠通信。每层都将本层用户与复杂的下一层隔离开，并简化与上一层的通信。您可能听说过 **OSI 网络模型** ，这是另一种描述网络层的类似方法；然而，用 TCP/IP 模型来想象 IP 网络要更容易一些。

每层都获取上一层的输入，然后再将信息嵌入本层的一个或多个协议数据单元(PDU, Protocol Data Units)中。PDU 是特定层的数据基元单元：例如数据链路层获取 IP 包，并将其分成 1 个或多个 Wi-Fi 数据链路层数据帧。物理层接收数据链路层数据帧，并把它们分成比特位。

|  |  |  |  |
| --- | --- | --- | --- |
| **层** | **协议** | **协议数据单元** | **备注** |
| 第5层 应用层 | [DNS](https://en.wikipedia.org/wiki/Domain_Name_System)、[DHCP](https://en.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol)、[MQTT](https://en.wikipedia.org/wiki/MQTT)、 [HTTP](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol)等。 | 数据 | 应用层下面的几个层可提供交换有用数据的机制。应用层是用来在设备中做有用操作的实际协议，例如 HTTP（获得或发送数据），DNS (由名称找到 IP 地址)，MQTT (发布或订阅)等。 |
| 第 4 层  传输层 | TCP  UDP | (TCP) 分段  (UDP) 数据报 | 经过纠错检查的可靠、有序字节流 – 可将其想象成计算机之间的管道或者电话呼叫。  不可靠的无连接数据报流(数据包) – 可将其想象成把信件投到邮局，但在对方确认之前无法知道对方是否收到信件，而且不能保证递送顺序。 |
| 第 3 层  网络层 | IP | 数据包 | IP 网络可利用源和目标 IP 地址向互联网上的任意位置发送和接收 IP 数据包。IP 层处理数据包的寻址和路由。 |
| 第 2 层  数据链路层 | 802.11 MAC | 帧 | 帧是网络中的传输基元单元。每个帧都不大于数据的最大传输单元 (MTU，Maximum Transmission Unit)，主要取决于每个数据链路层。来自上面几个层的所有数据都被数据链路层分成数据帧。  将比特位转换成未加密的帧。这个层只在局域网通信 |
| 第 1 层  物理层 | 802.11(a,b,g,n,ac) | 比特位 | 通过 Wi-Fi 射频发送和接收比特流；处理载波接入和网络介质的仲裁。 |

### (物理/数据链路) Wi-Fi 基础

Wi-Fi 网络有两个端：端站（Station，例如物联网设备）和接入点（Access Point，例如无线路由器）。为了使端站连接到 Wi-Fi 接入点，它必须具有以下信息：**SSID** 和**加密方案。**WICED 芯片负责选择合适的频带和频道。此外，为了发送数据，所有 Wi-Fi 数据链路帧都用源和目标 **MAC 地址**进行标记。

**SSID（无线网络的名称）**

SSID 是服务集标识符（Service Set Identifier）的简称。SSID 是网络名称，包含 0-32 个字节（也称为八位位组（octet），与 8 位字节一样，但该称呼因种种原因销声匿迹，不过网络界的人士还是这样称呼它们）。这个名称不一定是人类能读懂的（例如 ASCII），但由于它是未编码字节，因此严格区分大小写（要注意）。

**频带（2.4GHz 或 5GHz）**

Wi-Fi 射频根据 Wi-Fi 网络类型 (a、b、g、n、ac、ax) 和工作模式使用不同的调制方案对 1 和 0 编码。编码类型对 IoT 应用是透明的，因为芯片、射频和固件会对此进行虚拟化。然后，数据被发送到 2.4GHz 或 5GHz 频带（使用哪一个频带是很重要的）。注意，5GHz 频带吞吐量更高，时延更短，但范围较小，而 2.4GHz 频带正好相反。

**频道数**

可用频道数与频带(2.4GHz)和地理位置有关。FCC 管理着将哪个频道和频带用于世界上不同的工作区域。在 Wi-Fi 层，通过设置country-code来配置在某区域可用的频道。2.4GHz 非常简单，有频道 1-14，而频道 1-11 在全世界可用。5GHz 具有区域性，监管机构（例如 FCC）会根据您所在的区域强制规定可使用哪个频道。

**然而，从端站角度讲（也就是本课内容）这些都无所谓，因为当您试图加入 SSID 时，WICED SDK 会扫描所有通道以寻找正确的 SSID。**

**加密 (Open、WEP、WPA、WAP2)**

为了保证 Wi-Fi 网络的安全性，经常使用数据链路层加密。网络加密的类型为：Open（即无安全性）；无线等效加密(WEP，Wireless Equivalent Privacy)，不是完全安全（但对于有些有限传统应用来说可以使用）；Wi-Fi 保护访问 (WPA，WiFi Protected Access) 和 WPA2（基本已经取代了 WPA），必须支持 WPA2 才能在产品上使用 Wi-Fi 徽标。有两个版本的 WPA1/2：一个称为“个人”或“预共享秘钥”(PSK，Pre Shared Key)；另一个称为“企业（Enterprise）”。

WEP 和 WPA PSK 都使用密码（称为秘钥）来加密数据。不建议 WEP 加密方案，因为很容易破解（例如使用 Wireshark 和 AirSnort 这样的工具破解）。WPA 的 PSK 秘钥方案则非常安全，因为它使用 AES（Advanced Encryption Standard）。然而，共享秘钥是一种令人头痛的不安全过程，因为每个人都有相同的秘钥。为了解决秘钥分配问题，大部分企业网络解决方案使用 WPA2 Enterprise，该方案使用 RADIUS 服务器来单独处理每个端站的身份验证。

**企业级安全是物联网市场要面对的危机问题，也是 WICED 的差异化特性– WICED 能自动为您解决这个问题！**

**媒体访问控制(MAC)地址**

Wi-Fi MAC 地址是一串唯一的 48 位数字，包含 OUI（Organizationlly Unique ID）和端站 ID。MAC 地址的前三个字节是 OUI 字段，是 IEEE 分配给制造商（例如赛普拉斯）的唯一标识。为使数据链路层能发送数据帧，必须为数据帧添加源和目标 MAC 地址。网络上的其他设备只可将帧发送到能寻址到的协议栈的更高层。记住，数据链路层对于更高的层（例如 IP 层）一无所知。最后，最高有效字节的最高有效位（例如 47 位）指定多播（组）地址；全部位为1 的专用地址（例如 ff:ff:ff:ff:ff:ff）是广播地址（发送到每个人）。

数据链路层需要找出特定 IP 地址的 MAC 地址，以便在 Wi-Fi 网络上将其发送出去。为了找出这个映射，需要用到地址解析协议 (ARP，Address Resolution Protocol)。

**ARP**

每个设备中都有一个 ARP 表，表中包含 MAC 地址到 IP 地址的映射。为了发现 IP 地址的 MAC 地址，需要将“ARP 请求”广播到整个网络。连接到网络的所有设备都会监听 ARP 请求。如果听到一个 ARP 请求中包含自己的 IP 地址，就要用自己的 MAC 地址进行回复。这样，双方都要把信息添加到各自的 ARP 表（其实，如果您听到其他 ARP 请求，您也可更新表格）。这个方案的优势在于如果您发送 ARP 请求的 IP 地址不在您的本地网络上，那么路由器会用它的 MAC 地址进行回复（下部分我们将介绍该主题）。

### IP 网络



**互联网**是一个互连的 **IP 网络的**网格**。云**是您的网络能访问到的**互联网**的所有部分，也表示互联网上能连接到网络的服务器。

互联网上所有**设备**都有一个合法的 **IP 地址** ，并属于一个由**网络掩码**定义的 (IP) **网络**。**路由器**是连接 IP 网络的设备，能够从一个网络获取 IP 数据包，并将数据包转发到下个正确的网络。这是一项复杂的任务，超出了本课程的范畴，不过这也是思科价值 1510 亿美元的原因所在。本课的目的是让您知道，一旦连接到网络，您的数据包就会变魔术似的传送到另一端。

IP 地址是唯一识别 32 位的设备，通常表示为用句号分隔的四个十六进制字节。例如 192.168.15.7。IP 地址分成两个部分：网络地址（前 x 个位）和客户端地址（最后 32-x 个位）。网络掩码定义网络/客户端的分割。例如 192.168.15.\* 的网络掩码为 255.255.255.0

IP 网络（有时称为 IP 子网络）是共享相同网络地址的设备的集合，例如 192.168.15.\* （网络掩码 255.255.255.0）上的所有设备是同一 IP 网络的全部组成部分。

通常，物联网类型的设备的 IP 地址由动态主机控制协议 (DHCP，Dynamic Host Control Protocol) 服务器进行动态分配。要动态分配 DHCP 地址，首先要发送一个请求 IP 地址的 L2 广播数据报(DHREQUEST)。当 DHCP 服务器听到请求，就会用所需的信息进行回复。DHCP 集成到 WICED 中，当启动时能够自动处理信息交换。

### 设备配置表 (DCT，Device Configuration Table)

设备配置表是 WICED 闪存的一部分，采用预定义的格式，可用来存储有关系统的基本信息（客户端 AP SSID、客户端 AP 密码等）。它还可用来存储应用信息。DCT 被 WICED 固件用来“做正确的事”。例如，*wiced\_network\_up()* 可从 DCT 读取网络信息，并连接到指定网络。

该表在make process过程中建立，并与应用一起写入闪存。DCT 也可由您的应用修改（和写入）。

当构建 WICED App 时，既可以使用默认的 DCT，也可以定制一个完整的 DCT 或定制其中一部分。为了预先配置 DCT 表的Wi-Fi部分，您需要使用正确的 #defines 创建一个 .h 文件（通常称为 Wi-Fi\_config\_dct.h）。然后，需要将“WI-FI\_CONFIG\_DCT\_H := Wi-Fi\_config\_dct.h”添加到生成文件（makefile），以创建定制 DCT。

**您可以在目录 “include/default\_Wi-Fi\_config\_dct.h”下获得该文件的模板**



设备可在三种模式下运行，如上表中所示：配置 AP（4-7 行），Soft AP (10-13)，客户端模式（17-23 行）。还可以有多个网络接口并支持 Wi-Fi 和以太网（第 26 行）。配置 AP 模式下的设备允许其他设备与之相连，以通过 Wi-Fi 执行 WICED 系统的配置。Soft AP 模式下的设备可以作为正常的Wi-Fi 接入点。客户端模式的设备会作为端站连接到已有的 Wi-Fi 网络。本章我们只涉及客户端模式，因此只需使用 17-23 行。

要找到 #defines 的定义（或可能的定义），您可以高亮选定，右键点击，并选择“Open declaration”。例如，如果打开“WICED\_SECURITY\_OPEN”的声明，它会把您带到：



WICED SDK 提供一个预定义结构，该结构映射到闪存中的 DCT（在 WICED/platform/include 文件夹中的platform\_dct.h 文件中）。



从上表中看出，DCT 被分成多个段，例如 platform\_dct\_Wi-Fi\_config\_t 包含有关 Wi-Fi 配置的信息，诸如已知接入点等。



“stored\_ap\_list”条目是“wiced\_config\_ap\_entry\_t”类型的数组。数组的第一个元素（即索引 0）包含接入点（连接作为客户端的 STA）的信息。该结构类型如下所示：



结构中的第一个条目(wiced\_ap\_info\_t)包含客户端要连接的接入点的详细信息：



最后，这个结构中的很多条目也是结构类型。查看每个结构的定义，以了解其包含什么值。

DCT 可能在应用处理器的闪存中（如果处理器有内部闪存），或者存在于连接到 Wi-Fi 芯片的串行闪存中。为从 DCT 中读取数据，需要调用 *wiced\_dct\_read\_lock()* 函数；该函数将 DCT 读入 RAM 缓存，然后您可对其进行修改并使用 *wiced\_dct\_write()* 函数写回到闪存。

为 *wiced\_dct\_read\_lock()* 调用提供一个指针指向空结构的指针，这个空结构将用 DCT Wi-Fi 数据填充。结构的类型取决于您想读取 DCT 的哪个段（对 *wiced\_dct\_read\_lock()* 函数，段是一个参数）。例如，您想读取 DCT\_WIFI\_CONFIG\_SECTION，那么指针类型就是 platform\_dct\_Wi-Fi\_config。

您可以在 wiced\_dct\_common.h 文件中找到段名列表，该文件位于 WICED/platform/MCU 里面。以下是可用的段：



处理完 DCT 的 RAM 副本后，需要调用 *wiced\_dct\_read\_unlock()* 函数释放它。

如果闪存是“内部的”，可由处理器直接访问，您可以在调用 *wiced\_dct\_read\_lock()* 时将可写参数设为False，这样 *wiced\_dct\_read\_lock()* 会为您提供一个指向闪存的指针而不是在 RAM 中做一个副本。仅当只读 DCT 时才能这样做。也就是说，如果您想向 DCT 中写入内容，那么可写参数必须设为Ture。

DCT函数的文档位于Components->WICED->Application Framework->DCT



### WICED Wi-Fi SDK

为了连接到 Wi-Fi 网络，必须调用 *wiced\_network\_up()* 函数。这个 API 调用有三个参数：要使用的网络接口；获得 IP 地址等所使用的方法；要使用的静态 IP 参数（或 NULL）。这是 WICED 文档中的对这个API的描述：



wiced\_interface\_t 用于指定使用哪个网络接口。WICED-SDK 支持同时使用多个网络，例如 Wi-Fi 和以太网。为查看定义内容，我转到 SDK 中的 wiced\_interface\_t 定义（高亮选定，点击右键，选择“Open Declaration”）。本课程中，我们始终使用 WICED\_STA\_INTERFACE，这意味着我们总是作为端站（即客户端），不会作为接入点(Access Point)。



*wiced\_network\_up()* 调用中的下一个参数是关于如何配置网络，意味着如何指定 IP 地址、网络掩码、路由器等。您可以静态设置或者使用 DHCP。WICED SDK 可以打开设备内的 DHCP 服务器，以服务来自整个网络的 DHCP 请求。如果作为接入点，这个功能可以派上用场。不过在本课程中，我们使用 “WICED\_USE\_EXTERNAL\_DHCP\_SERVER”，这样您可从 DHCP（在本课程的路由器中运行）获取 IP 信息。以下是此种方法的截屏：



如果不使用 DHCP，则需要通过传递一个名为 wiced\_ip\_setting\_t 的结构来静态指定 IP 网络参数。



### WICED\_RESULT\_T

整个 WICED SDK 中，很多函数都返回一个值，告诉您发生了什么。返回值的类型为“wiced\_result\_t”，这是个巨大的枚举类型。返回的值里包括 WICED\_SUCCESS、WICED\_PENDING 和 WICED\_ERROR。如果观察 wiced\_result\_t，您无法看到这些值，因为枚举类型是分层构建，便于维护。这里是层级结构的顶层：



通过右键点击可以查看子列表。也就是说，如果您点击 WICED\_RESULT\_LIST，您会看到所有 “WICED\_”形式的枚举。那么，对于成功的命令，您会看到 “WICED\_SUCESS”。



### 文档

网络管理函数的相关文档在 WICED SDK 文档中，位于Components->Management->Network Management。



用来与原始 IP 网络连接的函数在Components->IP Communication->Raw IP。



此外，doc 目录中有个名为 WICED-DCT.pdf 的文档，包含 DCT 的介绍。

### 导引器(Introducers)

导引器是一个用来将物联网设备连接到网络的方法。也就是，它们需要知道连接到哪个 Wi-Fi SSID，使用什么密码，使用什么加密秘钥等等。解决这个问题的方法有几种，包括：

* 使用蓝牙连接到物联网设备，然后使用手机 App 配置设备。
* 使用 USB 或串行连接方法将物联网设备连接到计算机，然后用计算机应用程序配置设备。
* 使用物联网设备上的 web 服务器启动一个 Wi-Fi 接入点，然后用计算机或手机连接到物联网设备。DCT 的设备配置部分就是用于这个目的。
* 用所需的信息对设备进行预先编程。

WICED 支持所有这些方法。为了简便起见和节省时间，本课程中我们主要使用预先编程法。后续章节中的一些实例使用物联网设备上 web 服务器的 Wi-Fi 接入点。其他每种方法都会在 SDK 配套提供的实例应用中进行演示介绍。

## 练习

### 01 创建一个连接到开放式网络的 App，连接失败时红色 LED 闪烁，连接成功时绿色 LED 闪烁

1. 建立名为 05 的新文件夹和名为 01\_attach\_open 的子文件夹（或者复制一个以前的项目，例如模板）。
2. 将模板 default\_Wi-Fi\_config\_dct.h 复制到应用程序文件夹中（从步骤 1），并命名为 Wi-Fi\_config\_dct.h。
   1. 提示：记住它在 include 目录中。
3. 修改 Wi-Fi\_config\_dct.h。
   1. 提示：网络名和密码在手册的封底上。
4. 创建和编辑makefile（别忘了添加 #define for WI-FI\_CONFIG\_DCT\_H）。
5. 创建和编辑 01\_attach\_open.c（使用 *wiced\_network\_up()* 函数读取 DCT，并启动网络）。
6. 检查错误代码，并做正确闪烁。
   1. 提示：使用串行终端模拟器在设备引导和连接时查看来自设备的消息。

### 02 连接到 WPA2 AES PSK 网络

1. 复制练习(01)并修改 DCT，以连接到 WPA2 AES PSK 网络。
   1. 提示：网络名和密码在手册的封底上。

### 03 打印网络信息

1. 复制练习 (02) 并添加函数，以打印网络信息：

* 您的 IP 地址 (wiced\_ip\_get\_ipv4\_address)
* 网络掩码 (wiced\_ip\_get\_netmask)
* 路由器网关 (wiced\_ip\_get\_gateway\_address)
* IP 地址 www.cypress.com (wiced\_hostname\_lookup)
* 设备的 MAC 地址 (wwd\_Wi-Fi\_get\_mac\_address)
  1. 提示：查看 API 指南部分“*Components > IP Communication > Raw IP* 和 Components > IP Communication > DNS lookup*”*。
  2. 提示：地址（IP 地址、网络掩码、网关和 Cypress.com）以 wiced\_ip\_address\_t 类型的结构返回。结构中元素（名为 ip.v4）是一个 uint32\_t，包含以 4 个十六进制字节表示的 IPV4 地址。您可以单独屏蔽每个字节，以十进制值打印出来，并用点号分开，获得常见格式。例如，网络掩码 255.255.255.0 以 0xFFFFFF00 返回。
  3. 提示：MAC 地址以 wiced\_mac\_t 类型的结构返回。此结构包含一个名为八位位组 (octet) 的元素。这是一个含有 6 个字节的数组。可以打印这些字节并用“：”分开，以便用典型格式查看 MAC 地址。

### 04（高级）创建一个可在两个不同 SSID 之间切换的应用

1. 从练习 (03) 复制项目。
2. 创建一个函数，该函数可打印 SSID/密码以及当前在 DCT 中存在的安全信息。
3. 创建一个函数，使函数的输入为 ( char\* ssid, char\* passphrase, wiced\_security\_t security )，而且函数可通过执行以下步骤将该信息写入 DCT：
   1. 关闭网络 (*wiced\_network\_down()*)。
   2. 用其他网络的信息编写 DCT：
      1. 使用 *wiced\_dct\_read\_lock()* 获取当前结构。
         1. 提示：要写入值，必须将 ptr\_is\_writable 参数设为 WICED\_TRUE。
      2. 更新所需的信息。
         1. 提示：对于字符串形式的值（即 ssid 和密码）：
            1. 使用 *strcpy()* 将值复制到 RAM 缓冲器。
            2. 要确保更新结构中的字符串长度（可使用 *strlen()* 找到字符串的长度）。
      3. 使用 *wiced\_dct\_write()* 更新闪存中的 DCT。
      4. 使用 *wiced\_dct\_read\_unlock()* 释放内存。
         1. 提示：ptr\_is\_writable 参数必须匹配相应的 *wiced\_dct\_read\_lock()* 函数调用。
      5. 提示：请见实例项目 *snip/dct\_read\_write*。
   3. 重启网络 (*wiced\_network\_up()*)。
4. 使用控制台作为输入。当用户按‘0’或‘1’时，在网络 0/1（例如 0=WA101WPA 与 1=WA101OPEN ）之间切换。如果用户按‘p’，调用步骤 (1) 中编写的打印函数。
   1. 提示：复习第 2 章的 UART 接收练习。
5. 更改所选的网络，然后重启或复位电路板；注意它会以新的 SSID 开始，因为新设置已保存在 DCT 中。

## 推荐阅读

[1] 《TCP/IP详解,卷1：协议》 W.R.Stevens, ISBN 0201633469 – 也被誉为网络权威书籍，如果选一本书来学习 TCP/IP 网络，非它莫属！

[2] 《UNIX网络编程》– W.R.Stevens, ISBN 01394 – 如果您想学习 BSD Socket 编程，这是不二之选——最佳参考书，当今所有网络软件的基础。

[3] RFC 1122 – “互联网主机要求 – 通信层”；互联网工程任务组(IETF) - https://tools.ietf.org/html/rfc1122

[4] RFC 826 – “以太网地址解析协议”；互联网工程任务组(IETF) - https://tools.ietf.org/html/rfc826

[5] RFC 153 – “动态主机配置协议”；互联网工程任务组(IETF) - https://tools.ietf.org/html/rfc1531