

***Package na***

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

# ESP8266实现网状网络(Mesh Network)

## 我自己对8266Mesh组网的理解

ESP8266基础的AT指令集能够进行点对点（两台设备）、多对点的连接（一个AP节点对应最多5个站点），但是没有实现应用层中的mesh组网，形成一个树状的网络通信，表示的类型是，一个AP节点\站点（双模式的路由节点，特殊情况直接链接路由器上云数据）链接另一个AP\站点（双模式的路由节点），接着这个路由节点可以继续链接路由节点，或者链接终端。



## 乐鑫官方对wifi的mesh定义

### 重点小节

1）网状网络(Mesh Network) 特点：

网状网络(Mesh Network)在wiki上有了很详细的介绍，其特点可以简单描述为：

|  |
| --- |
| * 所有节点都可与拓扑中所有节点进行连线而形成一个“局域网路” * 所有节点可以透过多次跳跃进行数据通信 * **即使在拓扑中有节点无法服务或过于忙碌，网络还是可以正常运作** |

2）在网状网络中，每个节点都是一个AP(AccessPopint)，可以被其他节点连接，也可以连接一个其他的节点。

3）ESP8266单个节点最多支持4个连接，ESP32可以支持到10个。

4）ESP-WIFI-MESH 是一种无线通信网络，采用***树状拓扑结构***，其节点具有乐鑫 SoC 的 ***AP-STA 特性***。

5）ESP-WIFI-MESH 能够提供一个***易于部署、自动组网并自我修复的网络***。

6）***根据节点信号的强度值自动连接信号强的节点***

7）

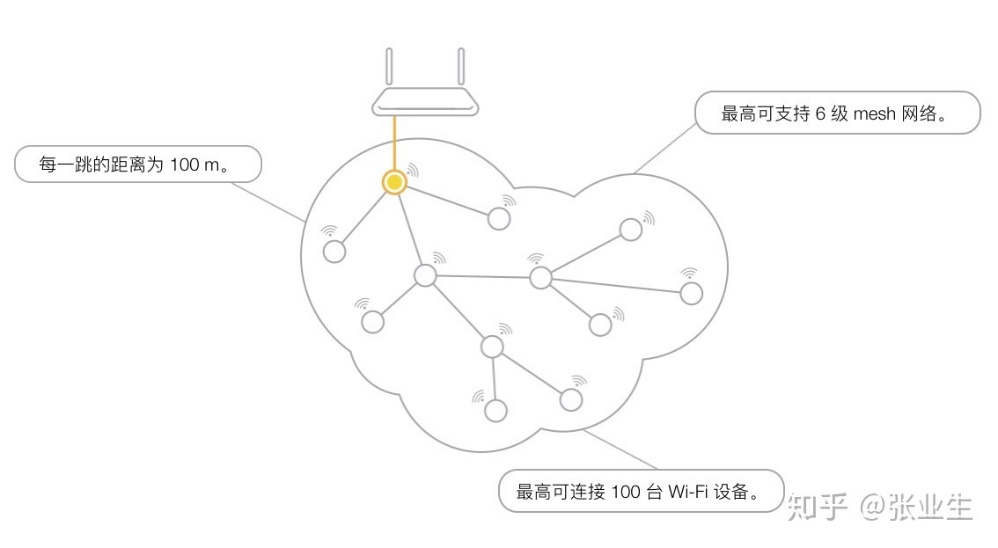
8）

9）

### 基本介绍

对于esp8266来说，网状网络的意义是什么呢？官方有个简单的介绍：

|  |
| --- |
| 随着物联网的发展，物联网节点规模迅速扩张，但路由器可供直接接入的节点数有限（通常少于 32 个）。针对此问题，乐鑫开发了 ESP-MESH 协议。在 Mesh 网络中，节点之间可以组成网络并转发数据包。这样不需要改变路由即可实现大量节点连接到网络。 |



在网状网络中，每个节点都是一个AP(AccessPopint)，可以被其他节点连接，也可以连接一个其他的节点；

ESP8266单个节点最多支持4个连接，ESP32可以支持到10个。



### ESP-WIFI-MESH

<https://www.espressif.com.cn/zh-hans/products/sdks/esp-wifi-mesh/overview>

ESP-WIFI-MESH 是一种无线通信网络，采用***树状拓扑结构***，其节点具有乐鑫 SoC 的 ***AP-STA 特性***。

ESP-WIFI-MESH 能够提供一个***易于部署、自动组网并自我修复的网络***。它无需部署 Wi-Fi 基础设施，就能在更大范围内部署节点，基于这种网络拓扑结构，它最多可以扩展到 1000 个节点。这样一来，ESP-WIFI-MESH 也可用于在家庭部署中覆盖 Wi-Fi 信号无法到达的盲点。

#### 简单和安全的设置

|  |
| --- |
| 我们为 Mesh 网络提供使用方便且可定制的***手机 APP***，这些 APP 可以快速发现新节点，并支持通过 Bluetooth LE 的方式轻松配置它们，确保将配置数据安全地分层推送到网络中的节点，这使得 ESP-WIFI-MESH 管理员能够清晰地管理节点分组，并为现有的部署选择合适的路由策略。 |

#### 自组网络和网络自愈

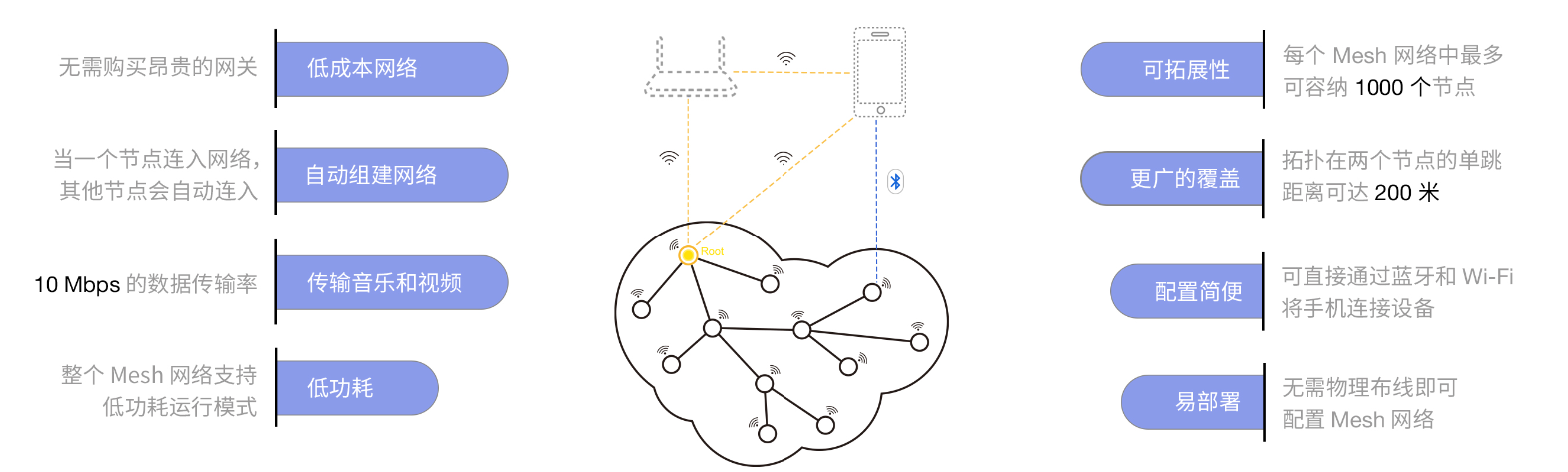
设备能够***根据节点信号的强度值自动连接信号强的节点***，形成 ESP-WIFI-MESH 网络，有助于在父节点断开时，实现不同节点之间的自动重新连接。这一特性能够自动修复网络，在网络故障时仍保证网络的稳定性。

#### 易于部署和拓展

ESP-WIFI-MESH 网络无需额外网关，***即可通过根节点直接连接路由器***，即使是小容量的 AP，也能支持大量设备，无需额外安装 Mesh 基础设施和设备就能覆盖更广的区域。

#### IP 连接

ESP Mesh 网络中的所有节点都可以进行 ***IP 连接***，不仅可以相互通信，还可以与网络外部进行通信。充当 NAT 或桥接器的根节点可以对这些节点提供 Internet 访问。



|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

#### github代码

<https://github.com/espressif/esp-mdf>

##### readme

该版本的 MDF 基于 IDF master 分支，不建议用在产品开发上，***如果寻求稳定版 MDF，建议使用 release/v1.0 分支***。

MDF master 分支已支持 ESP32S2。但以下示例暂时无法在 ESP32S2 上使用，后续将会逐步为 ESP32S2 适配

|  |
| --- |
| wanghao@DESKTOP-KQMGM9O MINGW64 ~/Desktop/新建文件夹/esp-mdf (master)  $ **git checkout origin/release/v1.0**  Note: switching to 'origin/release/v1.0'.  You are in 'detached HEAD' state. You can look around, make experimental  changes and commit them, and you can discard any commits you make in this  state without impacting any branches by switching back to a branch.  If you want to create a new branch to retain commits you create, you may  do so (now or later) by using -c with the switch command. Example:  git switch -c <new-branch-name>  Or undo this operation with:  git switch -  Turn off this advice by setting config variable advice.detachedHead to false  HEAD is now at e5ec634 Merge branch 'bugfix/mupgrade\_memory\_leak\_v1.0' into 'release/v1.0' |

##### 官方编程手册

<https://docs.espressif.com/projects/esp-mdf/zh_CN/latest/get-started/index.html#id4>

##### release/v1.0 readme

<https://github.com/espressif/esp-mdf/blob/release/v1.0/README_cn.md>

ESP-WIFI-MESH 开发框架

ESP-MDF (Espressif Mesh Development Framework) 是基于 ***ESP32*** 芯片的 ESP-WIFI-MESH 开发框架。ESP-WIFI-MESH 是一种基于 Wi-Fi 构建的 MESH 网络通信协议。

###### 概述

ESP-MDF 在 ESP-WIFI-MESH 协议栈的基础上增加了配网、升级、调试机制及应用示例。使用 ESP-MDF 您可以快速上手 ESP-WIFI-MESH 开发。其主要特点如下：

* 快速配网：在 app 配网的基础上增加了设备间链式配网，以实现大范围快速配网；
* 稳定升级：通过断点续传、数据压缩、版本回退和固件检查等机制达到高效升级；
* 高效调试：支持指令终端、通过无线进行日志传输和调试等多种调试方式；
* 局域网控制：支持 app 控制、传感器控制等；
* 丰富的示例：提供了基于 ESP-WIFI-MESH 的照明等综合解决方案。

|  |
| --- |
|  |

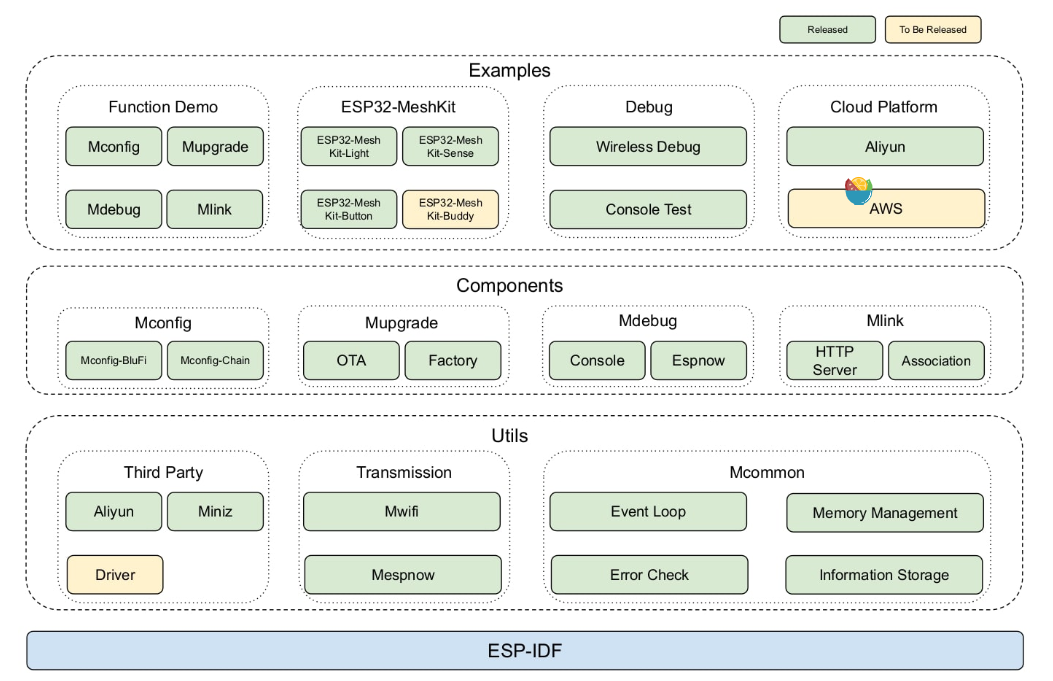
###### 框架

ESP-MDF 共分为 Utils、Components 和 Examples 三个部分（如下图所示），他们之间的关系：

Utils 是 ESP-IDF APIs 的抽象封装以及第三方库，

Components 是基于 Utils APIs 组成的 ESP-MDF 功能模块，

Examples 是基于 Components 完成的 ESP-WIFI-MESH 解决方案。



Utils：

* Third Party：第三方的组件

Driver：常用的按键、LED 等驱动

Miniz：无损高性能数据压缩库

Aliyun：阿里云物联网套件

* Transmission：设备间数据通信方式

Mwifi：对 ***ESP-WIFI-MESH 的封装***，在其基础上增加了重包过滤、数据压缩、分包传输和 P2P 组播

Mespnow：对 ESP-NOW 的封装，在其基础上增加了重包过滤、CRC 校验、数据分包

* Mcommon：ESP-MDF 各组件之间的共用模块

Event loop：ESP-MDF 的事件处理

Error Check：ESP-MDF 的错误码管理

Memory Management：ESP-MDF 的内存管理

Information Storage：将配置信息存储到 flash 中

Components

Mconfig：配网模块

Mupgrade：升级模块

Mdebug：调试模块

Mlink：局域网控制模块

Examples

* Function demo：各个功能模块的使用示例

Mwifi：常见组网方式的示例：无路由器、有路由器。先基于此示例进行开发,而后在其基础上添加配网、升级、无线测试等功能

Mupgrade：设备的升级示例

Mconfig：设备的配网示例

Mcommon：通用模块示例,事件处理 内存管理 信息存储的使用示例

* Debug：性能测试和调试工具

Console Test：通过串口输入命令的方式，测试 ESP-WIFI-MESH 吞吐量、网络配置、发包时延。

Wireless Debug：通过无线的方式进行 ESP-MDF 调试

* Development Kit：ESP32-MeshKit 使用示例, 用于调研和了解 ESP-WIFI-MESH

ESP32-MeshKit-Light：板载 ESP32 芯片的智能灯，用于 ESP-WIFI-MESH 作为主干网络进行长供电的场景。支持 BLE + ESP-WIFI-MESH, 可实现BLE网关, iBeacon 和 BLE 扫描

ESP32-MeshKit-Sense：ESP-WIFI-MESH 在 Deep-sleep + Light-sleep 模式下的低功耗方案，可用于：监测 MeshKit 外设功耗和根据传感器数据控制 MeshKit 外设

ESP32-MeshKit-Button：ESP-WIFI-MESH 在超低功耗的场景下使用，平常处于断电状态，仅在唤醒时工作，并通过 ESP-NOW 给 ESP-WIFI-MESH 设备发包。

* 云平台: ESP-MDF 对接云平台

Aliyun Linkkit：ESP-MDF 接入阿里飞燕平台示例

AWS：ESP-MDF 接入 AWS 平台示例

###### 使用 ESP-MDF 进行开发

您首先需要详细阅读 ESP-WIFI-MESH 通信协议和ESP-MDF 编程指南，并通过 ESP32-MeshKit 开发套件调研和了解 ESP-WIFI-MESH。其次基于 Function demo进行您的项目开发，当您可以在开发中遇到问题，首先可在官方论坛和官方 GitHub 上寻找是否已存在类似问题，若不存在类似问题，您也可直接在网站中提问。

ESP-WIFI-MESH 通信协议

<https://docs.espressif.com/projects/esp-idf/zh_CN/v4.4.2/esp32/api-guides/esp-wifi-mesh.html>

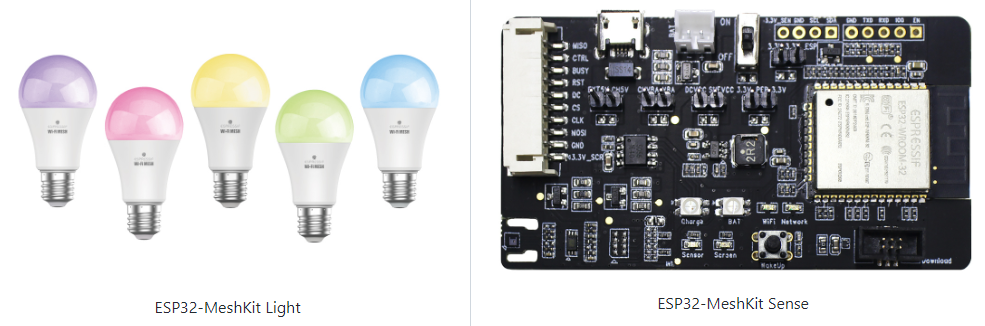
ESP-MDF 编程指南

<https://docs.espressif.com/projects/esp-mdf/zh_CN/latest/index.html>

###### 开发板指南

ESP32-MeshKit 开发套件

ESP32-MeshKit 包含一整套完整的 ESP-WIFI-MESH 的照明解决方案（如下图所示），可配套 ESP-Mesh App（iOS 版和安卓版）使用，既可以调研和了解 ESP-WIFI-MESH，也可以进行二次开发。



产品：

ESP32-MeshKit-Light：RGBCW 智能灯，直观反应控制结果，可用于测试组网时间、响应速度、距离测试、稳定性测试等。

ESP32-MeshKit-Sense：带有光强传感器和温湿度传感器，可用于功耗测量和低功耗应用的开发，可配套使用 ESP-Prog 进行固件烧录和 Debug。

ESP32-MeshKit-Button：作为开关控制，用于低功耗应用的开发，可配套使用 ESP-Prog 进行固件烧录和调试。

ESP32-Buddy 开发板

ESP32-Buddy 是专为 ESP-WIFI-MESH 开发测试而设计的开发板。体积小，采用 USB 供电，方便做大数量设备的测试及距离测试。

功能：

16 MB 的 flash：存储运行日志

OLED 屏：显示当前设备所在的层级、连接状态等信息

LED：运行状态指示

温湿度传感器：数据采集

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

#### 论坛

<https://www.esp32.com/viewforum.php?f=21>

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

<https://cloud.tencent.com/developer/article/1764431>

# C语言10大基础算法，学C语言必会源码（珍藏版）