课前准备 · 请在开课前完成

手机给 ESP32-S3 配 Wi-Fi

给零基础同学的人工智能编程课 · 课前环境准备指南（约 1~2 小时，可请家长协助）

这节课要做什么?

先从一个真实场景说起。想象你网购了一台智能插座：拆开包装，里面只有设备本体——没有键盘、没有屏幕，连一根数据线都没有。可它必须连上你家 Wi-Fi 才能变"智能"，而你家 Wi-Fi 的密码在它"脑子"里是一片空白。怎么把密码告诉一台没法打字的设备？这是所有智能设备出厂后的第一道坎，行业里叫**配网**。

你在课堂上用的开发板连着电脑、插着数据线，那是**开发者**才有的便利。今天我们要练的，是像真实用户那样——**只用一部手机，把 Wi-Fi 密码"隔空"送进板子。**

这节课的目标，一句话就能说清：让开发板连上你家 Wi-Fi。下课时你会拿到两个看得见的成果：一块 LED 变绿、自己连上网的开发板，和一个你亲手（让 AI）做出来的手机配网 App。今天从头到尾只围绕这一件事，不做别的。

这节课**从零开始、独立成篇**：不依赖任何前期内容，就算之前的内容没跟上，也完全不影响这次动手——跟着课堂节奏走就行。

你手上这块 **ESP32-S3 开发板**就是课堂上的"待配网设备"。我们会用 **AI 编程**的方式实现三种配网方式，每种都有一条不同的"魔法通道"：

1. **SoftAP + 网页**：板子自己开一个 Wi-Fi 热点，手机连上热点，打开网页填密码
2. **BLE 蓝牙**：手机 App 通过蓝牙把密码"写"给板子
3. **SmartConfig**：手机 App 把密码编成一串"信号包"广播出去，板子在空中"偷听"还原密码

全程你只需要做一件事：**把提示词发给 AI，让它写代码。**你不需要会编程，但需要做好下面的课前准备——环境装好、板子点亮，课堂 90 分钟才能顺利跟上。

要准备的物品

ESP32-S3 开发板

课堂主角，型号 N16R8（16MB Flash）

USB 数据线 × 2

要**两根**：一根连 ESP32 开发板，一根连 Android 手机。都要能传数据（只能充电的线不行）

手机

Android 或 iPhone 都可以，用来给板子配网

电脑

Windows 或 Mac，用来写代码和烧录

家里的 Wi-Fi（必须 2.4G）

配网实验的目标网络。**把 Wi-Fi 名称和密码抄在纸上带来**——密码可能只有家长知道，请家长帮忙抄

课前准备三件事

1

装开发环境

ESP-IDF 开发框架 + Android Studio（iPhone 学生再加 Xcode）

2

装 VS Code 插件

ESP-IDF 插件：编译、烧录都在 VS Code 里点按钮完成

3

跑通测试程序

让 AI 写一个 LED 闪烁程序烧进板子——灯亮起来就成功了！

第 1 步：安装开发环境

Windows 版步骤

1 安装 ESP-IDF 开发框架 约 30~60 分钟（下载约 1.6GB）

这个安装包会**一起装好三样东西**：ESP-IDF 框架、**编译工具**（把代码变成板子能跑的程序）、**烧录工具**（把程序传进板子）。不需要另外下载任何工具。

1. 打开下载页面：**https://dl.espressif.com/dl/esp-idf/**
2. 找到 **ESP-IDF v5.5.5 - Offline Installer**（离线安装器，Windows 版）并下载
3. 双击安装，一路点"下一步"即可（保持默认安装位置 `C:\Espressif`）
4. 安装完成后，桌面上会出现 **ESP-IDF 5.5 PowerShell** 快捷方式

> **成功标准：**桌面能看到 "ESP-IDF 5.5 PowerShell" 快捷方式

2 安装 Android Studio 约 20 分钟 课堂做 App 用

1. 打开 **https://developer.android.com/studio**，下载 Windows 版并安装
2. 第一次启动时让它自动下载 SDK（保持默认选项，需要联网，约 10 分钟）

> **成功标准：**能看到 "Welcome to Android Studio" 欢迎页面

Mac 版步骤

1 安装 Xcode 约 1~2 小时（下载约 10GB） 仅 iPhone 学生需要

1. 打开 App Store，搜索 **Xcode** 并安装（比较大，建议提前开始下载）
2. 打开"终端"（Terminal），粘贴运行一行命令安装命令行工具：

```
xcode-select --install
```

3. 打开 Xcode → 菜单 **Settings** → **Accounts** → 点 + 登录你的 **Apple ID**（免费账号即可，真机调试用，不需要付费开发者账号）

> **成功标准：** Xcode 能正常打开，Accounts 页面显示你的 Apple ID 已登录

2 安装 ESP-IDF 开发框架 约 20~40 分钟

1. 打开"终端"（Terminal），把下面三行命令**逐行粘贴**运行：

```
mkdir -p ~/esp && cd ~/esp
git clone --depth 1 --branch v5.5.5 --single-branch https://github.com/espressif/esp-idf.git
cd esp-idf && ./install.sh esp32s3
```

2. 第一行下载 ESP-IDF 框架；**第三行会下载并安装编译工具和烧录工具**（约 10~20 分钟），耐心等待，不需要另外下载任何工具

> **成功标准：** 终端最后出现 **All done**

3 安装 Android Studio 约 20 分钟 课堂做 App 用

1. 打开 **https://developer.android.com/studio**，下载 Mac 版（.dmg），拖进"应用程序"
2. 第一次启动时让它自动下载 SDK（保持默认选项，需要联网）

> **成功标准：** 能看到 "Welcome to Android Studio" 欢迎页面

第 2 步：安装 VS Code 的 ESP-IDF 插件

你的 VS Code 和 Codex 插件已经装好了，现在再加一个"乐鑫官方"插件，负责**编译、烧录、看日志**——以后都靠它点按钮。

1. 打开 VS Code，点左侧"扩展"图标，搜索 **ESP-IDF**（作者 Espressif Systems，官方插件），点安装
2. 安装后按键盘 **F1**，输入并选择：**ESP-IDF: Configure ESP-IDF extension**
3. 配置方式选 **EXPRESS**，然后选择 ESP-IDF 的安装位置：
   - Windows：`C:\Espressif\frameworks\esp-idf-v5.5.x`（v5.5 后面是具体版本号，选里面的 esp-idf 文件夹即可）
   - Mac：`~/esp/esp-idf`（即你的用户目录下 esp 文件夹里的 esp-idf）
4. 等待插件自动安装剩余工具（约 10~20 分钟），期间不要关闭 VS Code

> **成功标准：**VS Code 底部状态栏出现芯片型号 **esp32s3** 和一排小按钮（编译、烧录、监视器）

第 3 步：跑通测试程序（LED 闪烁）

0 连接开发板

用 USB 线把开发板连到电脑。**注意：一定要用"数据线"**（能传数据的线），有的线只能充电。

1 让 AI 生成程序

在 VS Code 里打开一个空文件夹（比如桌面新建一个 `esp32-course` 文件夹），然后打开 AI 对话框，把下面这段话**整段复制粘贴**发给 AI：

```
我要在 ESP32-S3 开发板上做一个"环境自检"程序。
请在当前文件夹创建 ESP-IDF 工程，要求：
1. 目标芯片 ESP32-S3，Flash 16MB
2. 板载 WS2812 彩灯接在 GPIO48
3. 程序效果：LED 每秒切换一次颜色（蓝→绿循环）
4. 告诉我编译、烧录、查看日志的三条命令分别是什么
```

AI 会创建好工程文件并给你命令。接下来在 VS Code 里点按钮即可。

2 编译（把代码变成板子能跑的程序）

先打开 AI 创建的 `led_test` 文件夹（VS Code：文件 → 打开文件夹），然后点底部状态栏的 **编译**按钮。

第一次编译要 **5~15 分钟**，是正常的（它在准备工具）。最终显示 **Build Successful**。

3 烧录（把程序传进板子）

点底部状态栏的 **烧录**按钮。如果提示选串口：

- Windows 选 `COM` 开头数字最大的（如 COM3）
- Mac 选 `/dev/cu.usbserial-` 开头的

看到进度条走完、显示 **Hard resetting** 就成功了。再点 **监视器**按钮可以看到板子打印的日志。

4 验证

> **成功标准：**板载 LED **蓝色、绿色交替亮起** = 环境全部正常，可以来上课了！

如果失败了：把红色报错信息整段复制，粘贴给 AI，问它怎么修：

```
我的 ESP32-S3 编译（或烧录）失败了，报错如下：
（把红色报错文字整个复制到这里）
请告诉我怎么修，用最简单的语言。
```

如果烧录成功但灯不亮：先换一根数据线试试；还不行就打开 AI 问：

```
ESP32-S3 烧录成功但板载 LED 不亮，可能是什么原因？我的板子是 ESP32-S3 N16R8。
```

Android 手机：开启开发者模式和 USB 调试 仅 Android 学生

课堂上要用数据线把手机连到电脑，把做好的 App"装"进手机。这需要先打开手机里藏着的两个开关（iPhone 学生跳过本节）。

1. 打开手机 **设置** → **关于手机**，找到"版本号"一栏，**连续快速点击 7 次**，手机会提示"您已处于开发者模式"
2. 返回 **设置** → **系统** → **开发者选项**（部分品牌在"更多设置"里），找到并打开 **USB 调试** 开关
3. 用数据线把手机连到电脑，手机会弹出 **"是否允许 USB 调试"**，勾选"始终允许"并点"确定"

> **验收办法（30 秒）：**手机连上电脑后弹出"是否允许 USB 调试"窗口 = 开启成功。再确认一次：电脑"设备管理器"（Windows）或"系统信息"（Mac）里能看到手机设备。

找不到"开发者选项"？ 说明第 1 步的 7 连击没生效——不同手机入口略有差别，把手机品牌型号发给 AI，让它告诉你这台手机的具体路径。

常见问题

- **烧录时找不到串口?** 换一根数据线；还不行就装 USB 串口驱动（CH340 或 CP2102，问 AI 要下载地址）。
- **编译特别慢?** 正常！第一次要 5~15 分钟，之后每次只要几十秒。
- **Mac 上插了板子没反应?** 终端运行 `ls /dev/cu.*` 看有没有 `/dev/cu.usbserial-` 开头的设备；没有就换线。
- **ESP-IDF 插件配置到一半卡住了?** 关闭 VS Code 重新打开，再按 F1 重新配置一次。
- **装不上、下载不动?** 网络问题居多。把现象（截图或文字）发给 AI，它会帮你分析。

完成检查清单

逐项打钩确认（打钩状态会自动保存，下次打开还在）。

- [ ] 我有能传数据的数据线（插上电脑能识别出串口）
- [ ] ESP-IDF 已安装（含编译工具和烧录工具；Windows：桌面有 PowerShell 快捷方式；Mac：终端出现 All done）
- [ ] VS Code 的 ESP-IDF 插件配置完成（底部状态栏显示 esp32s3）
- [ ] Android Studio 已安装（能看到欢迎页）
- [ ] Xcode 已安装并登录 Apple ID（仅 iPhone 学生）
- [ ] LED 测试程序编译成功
- [ ] LED 测试程序烧录成功，灯蓝绿交替
- [ ] Android 手机已开启开发者模式和 USB 调试（连电脑弹出"允许 USB 调试"即成功；仅 Android 学生）
- [ ] 上课物品确认（对照"要准备的物品"清单）：开发板、数据线、手机、电脑、Wi-Fi
- [ ] Wi-Fi 名称和密码已抄在纸上带来（密码请家长帮忙抄，课堂做实验要用）
- [ ] 我知道课堂要做什么：用 AI 编程给板子配 Wi-Fi

手机给 ESP32-S3 配 Wi-Fi · 课前准备 · 完成以上清单即可来上课