

# 如何用 **NodeMCU** 和 **Amazon Alexa** 自制智能家居系统

Phodal Huang

September 8, 2017

## 目录

步骤 1: Showcase .....	3
步骤 2: 材料清单 .....	3
步骤 3: 必要的软件和文档 .....	3
步骤 4: 什么是 NodeMCU .....	3
步骤 5: NodeMCU + Arduino IDE 智能家居 .....	4
步骤 6: 设置 Arduino IDE 来为 NodeMCU 编程 .....	4
步骤七: 电路及引脚图 .....	5
步骤八: 代码 .....	6
步骤九: 演示 .....	7

玩点什么: <https://www.wandianshenme.com>

原文链接:<https://www.wandianshenme.com/play/diy-home-automation-with-nodemcu-amazon-ec>

如果你读过我的一篇文章: [DIY Home Automation With ESP8266 \(Linknode R4\) and Amazon Alexa](#), 那么你就知道这次的教程的基础知识了。这次不是使用一个专用的板子, 而是要使用一个 **NodeMCU** 和一个 **4 通道** 的继电器, 来完成相似的工作。与我写的每个家庭自动化教程一样, 这些教程的目的: 不是向您展示, 如何将系统完全集成到家庭的基础设施中。而是, 为您提供如何使用低电压 **DC** 控制市电交流供电设备的一个起点。

## 步骤 1: Showcase

本视频将向您展示, 如果您遵循本教程可以做到什么程度。首先观看它, 然后阅读指导, 就能了解如何使用 **Amazon Alexa**, 来控制您家中使用 **NodeMCU** 的继电器。

## 步骤 2: 材料清单

要构建此项目, 您将需要以下硬件:

- Amazon Echo Dot
- NodeMCU 开发板
- 4 通道继电器
- 20x4 I2C 字符 LCD
- 杜邦型跳线
- 面包板

## 步骤 3: 必要的软件和文档

要完成此项目, 您将需要以下软件和文档:

- Arduino IDE
- [FauxMos 源码](#)
- [NodeMCU 文档](#)

## 步骤 4: 什么是 NodeMCU

NodeMCU 是一个开源的 IoT 平台, 其包括运行在 **Espressif ESP8266 Wi-Fi SoC** 上的固件, 以及基于 **ESP-12** 模块的硬件。默认情况下, NodeMCU 固件使用基于 **eLua** 项

目的 Lua 脚本语言，并基于 ESP8266 的 Espressif Non-OS SDK。它使用许多开源项目，如 lua-cjson 和 spiffs。

## 步骤 5: NodeMCU + Arduino IDE 智能家居

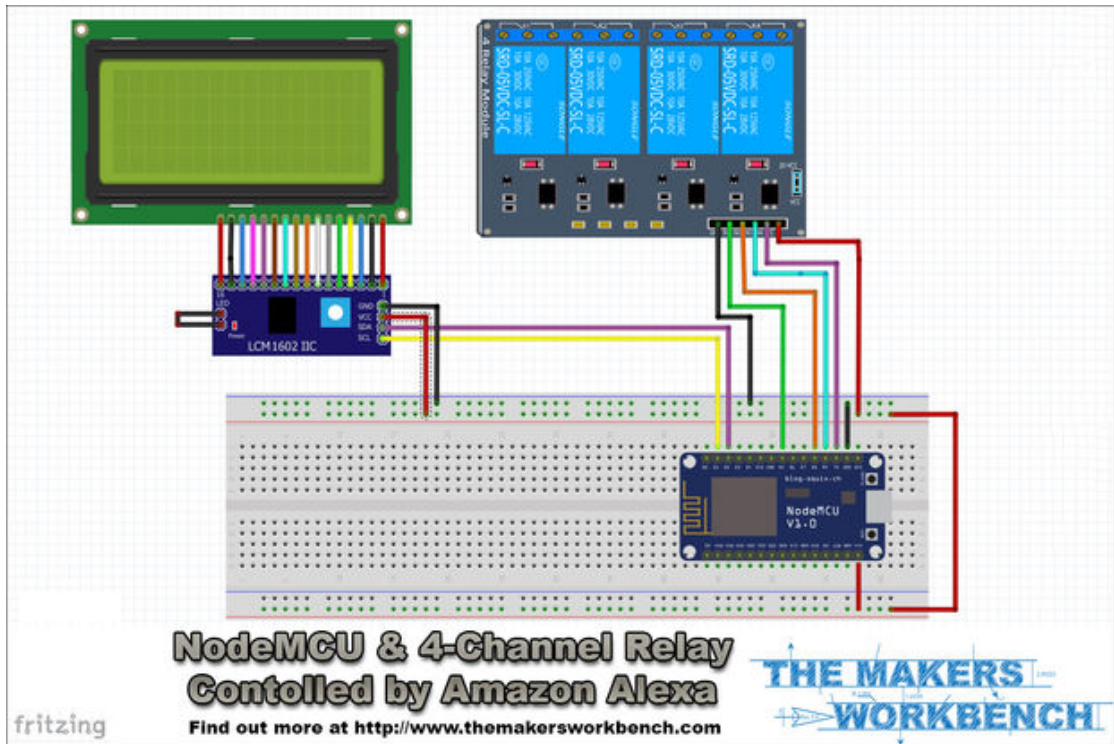
由于 Arduino.cc 开始开发基于非 AVR 处理器（如 ARM / SAM MCU）的新 MCU 板，并先在 Arduino Due 中使用。因此需要修改 Arduino IDE，以便 IDE 以支持额外的工具链，它允许将 Arduino C/C++ 编译到这些新的处理器上。它们通过 Board Manager 和 SAM Core 来做到这一点。一个“Core”是 Board Manager 和 Arduino IDE 所需的软件组件的集合，用于将 Arduino C/C++ 源文件编译到目标 MCU 的机器语言。几个创新的 ESP8266 爱好者在 [GitHub ESP8266 Core 网页上](#) 提供了 ESP8266 WiFi SoC 的 Arduino 内核。它通过被称为“ESP8266 Core for the Arduino IDE”，并已成为各种基于 ESP8266 的模块，和开发板（包括 NodeMCUs）的领先软件开发平台之一。

如果您以前没有阅读过关于此主题的指导，要开始使用 NodeMCU，您将需要在 Arduino IDE 中做一些额外的工作，才能够使用 Arduino IDE 能对 NodeMCU 进行编程。按照以下步骤使 Arduino IDE 设置为使用 ESP8266 和 NodeMCU。

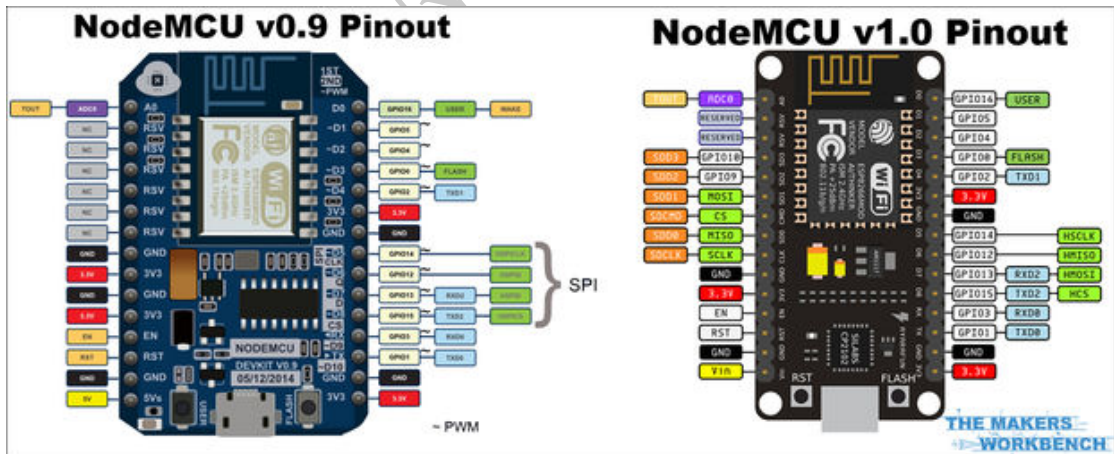
## 步骤 6: 设置 Arduino IDE 来为 NodeMCU 编程

1. 在开始之前，您需要在下载最新版本的 Arduino IDE。安装 Arduino 后，运行该程序，并导航到“文件 (File)”>“首选项 (Preferences)””，将地址 [http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json) 添加到 Additional Board Manager URL 字段中。
2. 打开位于 Tools > Board Menu > Boards Manager 中的 Boards manager
3. 搜索并安装 esp8266 平台（安装后不要忘了从 Tools -> Board 中选择您的 ESP8266 板）。
4. 最后，将鼠标悬停在“工具 (Tool)”菜单上，选择对应的开发板。如如果您使用的是 NodeMCU V1.0 板，则选择 NodeMCU 1.0 (ESP-12 Module) 板。

步骤七：电路及引脚图



这个项目的原理图很简单，直截了当。按照上面的图表以及下一步中列出的引脚连接引脚连接，来复制我的设置。请注意，我使用的 20x4 字符液晶显示屏，已经焊接了 I2C 模块。如果您想使用单独的 LCD 和 I2C 背包，请按照特定 LCD 和 I2C 模块的引脚排列，以确保它们正确连接。



I2C 模块与 NodeMCU 连接:

- I2C GND -> NodeMCU GND
- I2C VSS -> NodeMCU 5Vin/5V
- I2C SCL -> NodeMCU D1
- I2C SDA -> NodeMCU D2

NodeMCU 与四通道继电器连接：

- 继电器 GND -> NodeMCU GND
- 继电器 VCC -> NodeMCU 5Vin/5V
- 继电器引脚 1 -> NodeMCU D5/GPIO14
- 继电器引脚 2 -> NodeMCU D8/GPIO15
- 继电器引脚 3 -> NodeMCU D9/GPIO03
- 继电器引脚 4 -> NodeMCU D10/GPIO01

## 步骤八：代码

我将不会分享所有需要的代码文件，为了让 Amazon Alexa 在此处运行，它需要有很多代码。请访问我的 [GitHub](#) 来下载需要的所有需要的文件，其使用的是 zip 存档。请注意，下面的代码将无法编译或者上传，没有这些附加文件，所以请转到我的 [GitHub](#) 来获取它们。

我将分享我使用的主要 Arduino 程序，以便于你可以了解一切如何联系在一起。我将不会分解整个代码，但我会指出，我正在创建四个具有唯一名称的回调，四个具有唯一名称的开关，以及定义继电器的四个整数。我还定义了用于 Alexa 理解的四个唯一的调用名称，这些名称映射到 Web 服务器上的独立端口，每个唯一的调用名称都有两个设置：On 和 Off。

最后，我将四个继电器引脚中的每一个的引脚模式设置为输出。那么，它就像为每个继电器设置八个不同的功能一样简单。除了我正在使用的继电器板的类型，我还必须在设置代码中，将所有的四个继电器引脚设置为 HIGH 状态。这是必需的，因为线圈被连接在板上所使用的继电器中。高电平信号使继电器断开，低电平信号使其接通。

例如，当 Alexa 听到“Turn on lightOne”命令时，NodeMCU 上的 ESP8266 运行 lightOneOn 方法。这里还有更多的处理 FauxMo Belkin WeMo 仿真的代码，坦率的说，我不完全确定我了解它的每一个方面的代码。但是我知道它的工作原理，添加新设备（WeMo 和 Alexa 最多可以支持 14 个）就像复制上面提到的那样简单，并给每个新设备一个唯一的名称。所以不用多说，下面是 Arduino 代码，它可以使您实现上述视频中看到的魔法工作。请记住，如果没有其他五个文件，这个代码是没用的。[GitHub](#) 相关项目地址：[GitHub](#)。

## 步骤九：演示

将所有连接的代码和代码上传到 **NodeMCU** 后，您可以将某些主电源设备连接到继电器，然后请求 **Alexa** 打开相应的设备。

原文链接：<http://www.instructables.com/id/How-To-DIY-Home-Automation-With-NodeMCU-and->

原文链接：<https://www.wandianshenme.com/play/diy-home-automation-with-nodemcu-amazon-ec>

玩点什么：<https://www.wandianshenme.com>