

低廉的亚马逊 **Amazon Alexa** 与
Raspberry Pi、**Ngrok** 进行家庭自
动化灯开关

Phodal Huang

September 8, 2017

目录

步骤 1: 材料	3
步骤 2: 设置 Raspberry Pi	3
步骤 3: 安装和依赖	4
步骤 4: Ngrok	4
步骤 5: Python 脚本	5
步骤 6: Alexa Skill	6
步骤 7: 继电器	6
步骤 8: 试试吧!	7

玩点什么: <https://www.wandianshenme.com>

原文链接:<https://www.wandianshenme.com/play/build-home-automation-with-amazon-echo-rasp>

我刚刚得到了一个亚马逊 **Echo Dot** 智能音箱, 并且一直想扩大自己的家庭自动化的可能性。但是, 对于诸多现成的 **IOT** 解决方案的价格不满意, 我想我可以自己做——用合理的功能并省下一些钱。

我已经有了一堆 **Raspberry Pi**, 所以我认为它这是一个很好的开始。**Raspberry Pi** 绝对是这个项目中最为贵的部分, 但是对我来说是免费的, 你还能直接使用 **\$10** 美元的 **Pi Zero Wireless**。

我的是这个想法是: 让 **Raspberry Pi** 可用于 **Alexa** (来自 **Echo** 上的), 并使用它来控制继电器。这个继电器开启了很多事情的可能性, 但是我想用它来控制我客厅里的灯。

步骤 1: 材料

我们需要的材料有:

- **Raspebrry Pi**
- **5V** 继电器模块
- 跳线
- **Amazon Echo**
- 带导线的公电源插头
- 带导线的母电源插头

步骤 2: 设置 **Raspberry Pi**

设置 **Raspberry Pi** 有这样做的各种方法。如果您有一个 **HDMI** 显示器和 **Usb** 键盘将会方便, 我认为这更容易一些。否则你只能做做一个“无头” (**headless**) 的设置。

这是一个很好的教程, 在 **Windows** 上进行“无头” (**headless**) 设置:

<http://www.circuitbasics.com/raspberry-pi-basics-setup-without-monitor-keyboard-headless-mode>

对于 **Mac OS** 的用户来说, 则可以用:

<https://www.instructables.com/id/My-Raspberry-Pi-Setup/>

但最简单的方法是, 加载带有 **NOOBS** 系统的 **SD** 卡, 在连接到显示器和键盘的同时启动 **Raspberry Pi**, 然后进行配置。相关教程如下:

<http://lifehacker.com/the-always-up-to-date-guide-to-setting-up-your-raspberr-1781419054>

如果你没有做一个“无头” (**headless**) 设置, 你应该仍然可以使用已经准备好的

SSH, 其余部分是必要的。为了做到这一点, 我用 Putty。Putty 的网站是:

<http://www.putty.org/>

如果您不知道您的 Raspberry Pi 的 IP 地址, 您可以使用高级 IP 扫描仪:

<https://www.advanced-ip-scanner.com/>

然后在 Putty 中输入 Raspberry Pi 中输入 Pi 的主机名和 IP 地址, 将端口设置为在 22, 然后单击打开。系统将提示您进行登录。

步骤 3: 安装和依赖

对于我来说, 让这一切可以工作的主要原因是, 它很好玩。亚马逊提供了一个名为“flask-ask”的 python 库来接收来自 Alexa 的来电。有关 flask-ask 的问题, 可以其创建者 John Wheeler 创建的网站: <https://alexatutorial.com/flask-ask/>。一旦它的工作正常, 这个功能真的很好用。但是唯一痛苦的是: 从哪里获取它, 并让它工作。

我的第一个注意事项是, 我最终不得不使用 python 3 来进行下一步。

在这里是我整理的依赖关系:

```
1 apt-get update
2 apt-get install libffi-dev
3 apt-get install python3-cffi
4 apt-get install libssl-dev
5 pip3 install python-dateutil
6 pip3 install click
7 pip3 install cryptography
8 pip3 install flask-ask
```

我认为这是已知的所有库, 但我可能漏掉了几个。不过, 我希望它至少能为您节省查找包的时间。

步骤 4: Ngrok

Ngrok 是一个实用工具, 可以从您的 Raspberry Pi 打开一个隧道到 HTTPS 结点, 以便 Alexa 能够找到它。你需要在你的 Raspberry Pi 上一直运行这个服务, 以保证服务一直可用。你在这里下载相应的 zip 文件:

<https://ngrok.com/download> (PS: 获取稳定的 ARM Linux 版)

并使用 **SCP** 将其复制到您的 **Raspberry Pi** 上。我喜欢使用 **winSCP**。你可以在这里找到它：

<https://winscp.net/eng/download.php>

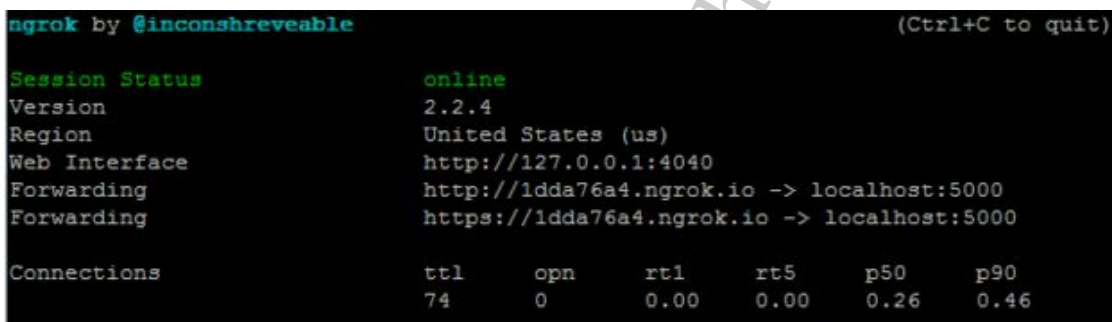
一旦复制到你的 **Raspberry Pi** 上，请解压之：

```
1 unzip ngrok-stable-linux-arm.zip
```

然后运行：

```
1 sudo ./ngrok http 5000
```

如下图所示：



```
ngrok by @inconshreveable (Ctrl+C to quit)
Session Status      online
Version             2.2.4
Region             United States (us)
Web Interface       http://127.0.0.1:4040
Forwarding          http://1dda76a4.ngrok.io -> localhost:5000
Forwarding          https://1dda76a4.ngrok.io -> localhost:5000
Connections
  ttl    opn    rt1    rt5    p50    p90
   74     0    0.00  0.00  0.26  0.46
```

注意：

在 **SSH** 断开连接后，要保持此运行状态，有几种方法可以执行。我发现一个名为 **tmux** 的实用程序，我很喜欢它。安装它可以用：

```
1 sudo apt-get install tmux
```

使用 **tmux** 命令启动会话。在会话中可以让进程运行，在这种情况下是 **ngrok** 进程。按 **ctrl + b**，然后 **d**。当你离开它仍然会运行。要回到该会话使用：

```
1 tmux attach -t [会话的名字] (您创建的第一个会话将为“0”)
```

要查看正在运行的会话的列表，请使用：

```
1 tmux list-sessions
```

步骤 5: Python 脚本

除了 **ngrok**，您的脚本还需要在后台开运行另外一个服务。继续打开另一个 **putty** 实例，来打开另一个 **SSH** 终端。

你可以使用我附带的脚本

它导入了所有需要的模块（基本上是 `flask` 和 `GPIO`），并设置了应用程序。然后，它为名为“TurnLights”的 Intent 设置一个端点，并将“slot”（如变量）“status”映射到“status”。随后，它定义一个函数“turn”来充当处理程序。在这里，我们将引脚 3 设置为输出，如果语音命令的状态为“on”，则将该引脚置为高电平；如果为“关闭”，则将引脚设置为低电平。接着，我们返回一些关于做了什么的信息。

脚本下载: [light.py](#)

使用命令运行脚本:

```
1 python3 light.py
```

步骤 6: Alexa Skill

下一步是使 Alexa Skill。需要，于是亚马逊开发商门户网站。如果没有帐户，请注册一个帐户:

<https://developer.amazon.com/edw/home.html#/skills>

添加新的 skill，并把它命名为我们能辨认的东西，如我的是“RPi control”。然后决定一个调用名称，我取的名字是“jeoffrey”。

intent 是一个有点棘手的東西。其语法非常具体，而且在确保 Alexa 能实际理解的情况下，示例语句非常重要。您可以使用我的，或使用它作为参照:

[intent.txt](#)

在配置中，选择 `https` 并输入您的 `ngrok` 的网址。在证书中选择『My development endpoint is a sub-domain of a domain that has a wildcard certificate from a certificate authority』

现在，一旦完成建造（可能需要几分钟），您可以进入测试页面，并输入类似于我们用于示例话语（“TurnLights to On”）的内容。如果所有设置正确，您应该看到它在 Raspberry Pi 上注册，并获得正确的响应。

步骤 7: 继电器

最后一步是连接继电器。继电器将打开电路，并允许电流流向您的灯。继电器有两侧，控制侧和开关侧。在控制端你会发现 3 个端口:

- DC+
- DC-

- IN

其控制端全部连接到 Pi。参考 Raspberry Pi GPIO 引脚排列来匹配连接, DC + 连接到 5V, DC- 连接到 GND 和 IN 到使用的 GPIO 引脚 3。

切换到另外一边, 还有另外三个终端:

- NO
- COM
- NC

这些是“常开”, 意味着在继电器被触发之前, 没有电流流向该口。“常闭”意味着电流流向此端子, 直到继电器触发。而“COM”是输入。通常继电器将以高信号触发 IN 引脚。这个特定的继电器有一个 H/L 跳线, 意味着你可以选择是在高电平还是低电平触发, 我将其置于高电平触发。

我们只会使用其中两个口, 您需要将一根电线从交流电源线连接到 COM 口; 接着, 将一条电线从延长线并入 NO 端子。; 然后, 将剩余的电线从延长线, 连接到电源插头上的剩余电线上。

步骤 8: 试试吧!

是时候, 试试吧!

原文链接: <http://www.instructables.com/id/Cheap-Alexa-Home-Automation/>

原文链接: <https://www.wandianshenme.com/play/build-home-automation-with-amazon-echo-rasp>