使用 **Amazon Echo** 和 **Python** 控制 **Raspberry Pi GPIO**

玩

September 8, 2017

目录

原文链接:https://www.wandianshenme.com/play/□□amazon-echo□python□□raspberry-pi-gpio

此项目的主要目的在于使用 Amazon Echo 的 Alexa’s 语音命令和 Python 来控制 Raspberry Pi 的 GPIO。此文改编自记忆游戏项目 (链接：Flask-Ask: A New Python Framework for Rapid Alexa Skills Kit Development.)

Amazon Echo 可有可无，但我强烈推荐使用。本文使用一个非常简单的例子轻松的演示如何只使用几十行 Python 代码将 Raspberry Pi 的功能开放给 Alexa。

我写此文的主要原因是因为我在别处找不到类似的指南，觉得功能很酷。目前有几种不同的方式将定制的 Smart Home 编程添加到 Alexa，但据我所知，这是使用 Raspberry Pi 和 Python 的第一种方法。

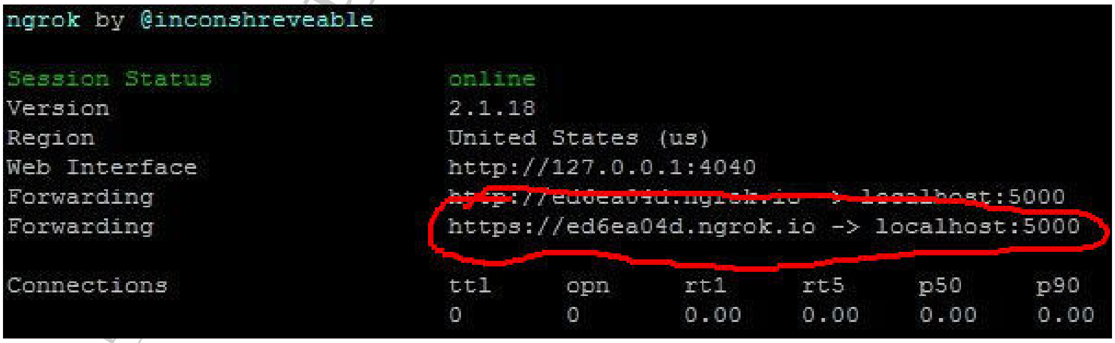
步骤 1：初始化

我使用一块 Raspberry Pi 3 开发板和最新的 Raspbian Jessie-lite 镜像 (镜像来自于:https://www.raspberrypi.org/downloads/raspbian/，此外还需要使用 SSH 的两个终端会话访问 Pi。登录后，输入以下命令来安装所需的软件包和 python 库：

```
sudo apt-get update && sudo apt-get upgrade -y
sudo apt-get install python2.7-dev python-dev python-pip
sudo pip install Flask flask-ask
```

步骤二：安装 **Ngrok**

命令如下图:



```
ngrok by @inconshreveable

Session Status                online
Version                       2.1.18
Region                        United States (us)
Web Interface                 http://127.0.0.1:4040
Forwarding                    http://ed6ea04d.ngrok.io -> localhost:5000
Forwarding                    https://ed6ea04d.ngrok.io -> localhost:5000

Connections                   ttl     opn     rt1     rt5     p50     p90
                              0       0       0.00    0.00    0.00    0.00
```

对于 ngrok 我想不到更好的解释，因此这里是一个引用我之前链接的亚马逊的指南：

> ngrok 是一个命令行程序，打开到 localhost 的安全通道，并将该通道暴露在 HTTPS 端点后面。ngrok 使得 Alexa 可以立即与您的代码对话。按照以下三

个步骤生成一个公共 HTTPS 端点到 127.0.0.1:5000。

查看 https://ngrok.com/download 并下载最新的 Linux ARM 发布版本压缩包并解压到主目录:

```
unzip /home/pi/ngrok-stable-linux-arm.zip
```

之后运行以下命令行:

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

您的屏幕应该能看见类似像上面的图像。请注意以 https 开头的 'Forwarding' URL，稍后将使用。

注意：不幸的是，每次启动服务时，ngrok URL 都会更改，因此如果您尝试长时间运行，这不是永久的解决方案。如果您需要一个更为永久的 URL 用于您的项目，我会推荐像 Yaler 或 Page Kite 这样的服务。

步骤三：**Python** 脚本

打开一个新的终端会话并创建一个 gpio_control.py 文件：

```
nano gpio_control.py
```

复制/粘贴以下的代码到刚建立的 gpio_control.py 文件：

```
from flask import Flask
from flask_ask import Ask, statement, convert_errors
import RPi.GPIO as GPIO
import logging

GPIO.setmode(GPIO.BCM)

app = Flask(__name__)
ask = Ask(app, '/')

logging.getLogger("flask_ask").setLevel(logging.DEBUG)

@ask.intent('GPIOControlIntent', mapping={'status': 'status', 'pin': 'pin'})
def gpio_control(status, pin):
```

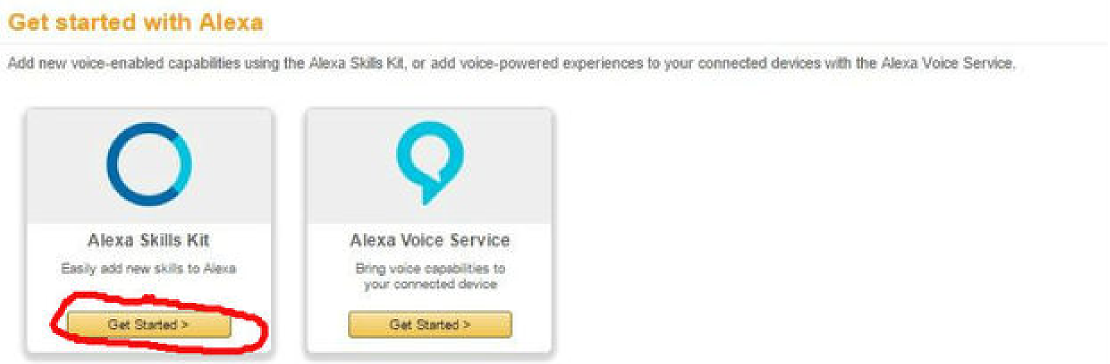
```

try:
pinNum = int(pin)except Exception as e:
    return statement('Pin number not valid.')

GPIO.setup(pinNum, GPIO.OUT)

if status in ['on', 'high']:    GPIO.output(pinNum, GPIO.HIGH)
if status in ['off', 'low']:    GPIO.output(pinNum, GPIO.LOW)

return statement('Turning pin {} {}'.format(pin, status))
```

保存并关闭文件。使用以下命令运行 python 脚本:

```
sudo python gpio_control.py
```

ngrok 和 gpio_control.py 运行的同时我们设置 AWS...

步骤四：**AWS** 账户



Get started with Alexa

Add new voice-enabled capabilities using the Alexa Skills Kit, or add voice-powered experiences to your connected devices with the Alexa Voice Service.

Alexa Skills Kit

Easily add new skills to Alexa

Get Started >

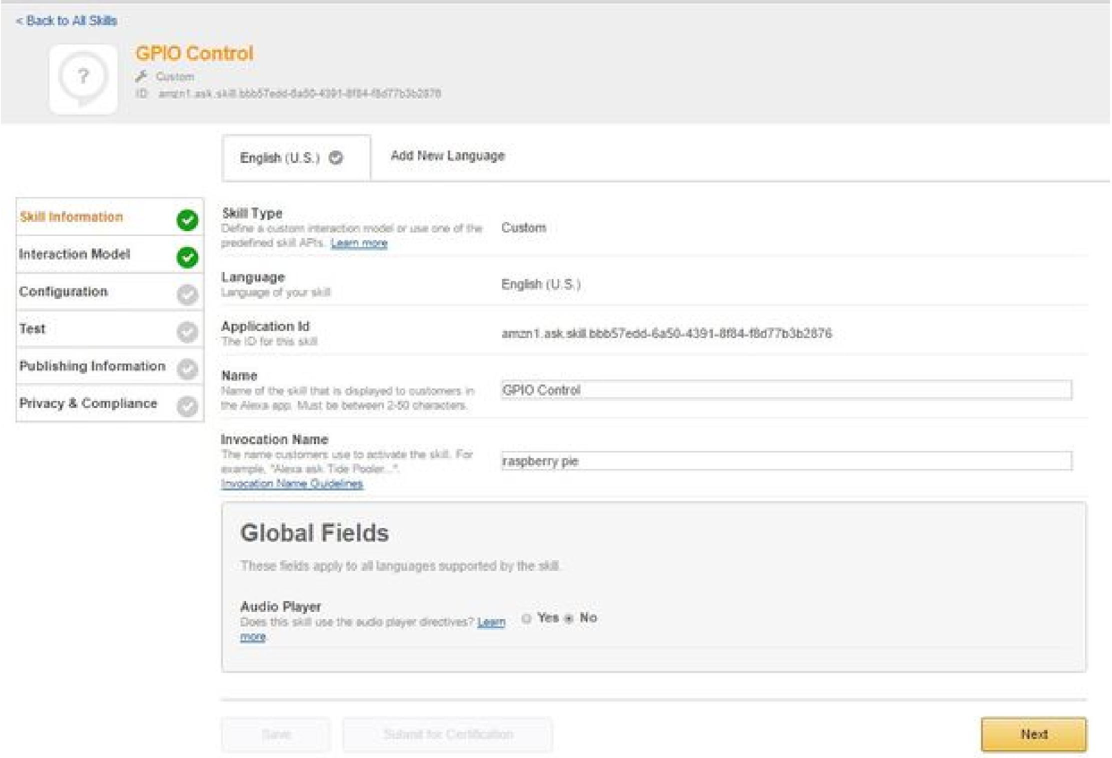
Alexa Voice Service

Bring voice capabilities to your connected device

Get Started >

先创建或者登陆 AWS 开发者账户并打开Alexa Skill 列表。

步骤五：**Alexa Skill** - 信息



< Back to All Skills

GPIO Control

Custom

ID: amzn1.ask.skill.bbb57edd-6a50-4391-8f84-f8d77b3b2876

English (U.S.) | Add New Language

- Skill Information ✓
- Interaction Model ✓
- Configuration
- Test
- Publishing Information
- Privacy & Compliance

Skill Type
Define a custom interaction model or use one of the predefined skill APIs. Learn more
Custom

Language
Language of your skill
English (U.S.)

Application Id
The ID for this skill
amzn1.ask.skill.bbb57edd-6a50-4391-8f84-f8d77b3b2876

Name
Name of the skill that is displayed to customers in the Alexa app. Must be between 2-50 characters.
GPIO Control

Invocation Name
The name customers use to activate the skill. For example, "Alexa ask Tide Pooler...".
Invocation Name Guidelines
raspberry pie

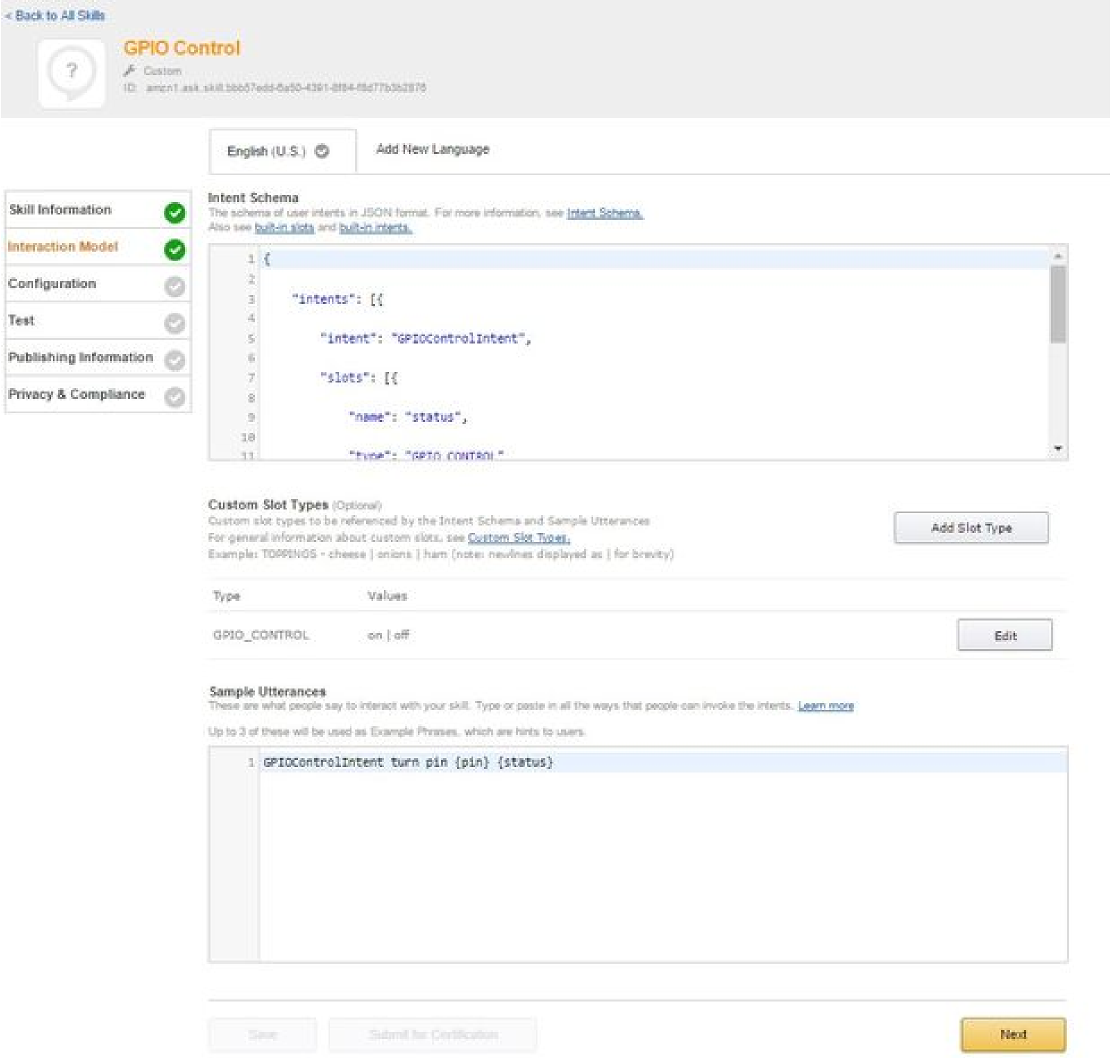
Global Fields

These fields apply to all languages supported by the skill.

Audio Player
Does this skill use the audio player directives? Learn more
○ Yes ● No

将 Skill 名称设置为’GPIO Control’，将 Invocation Name 设置为要用于激活 Skill 的词。点击’Next’ 继续。

步骤六：**Alexa Skill** - 交互模式



< Back to All Skills

GPIO Control

Custom

ID: amzn1.ask.skill.bbb57edd-6a50-4391-8f84-f6d77b5b2878

Intent Schema

The schema of user intents in JSON format. For more information, see Intent Schema.
Also see built-in slots and built-in intents.

```
{

    "intents": [{

        "intent": "GPIOControlIntent",

        "slots": [{

            "name": "status",

            "type": "GPIO_CONTROL"
```

Custom Slot Types (Optional)

Custom slot types to be referenced by the Intent Schema and Sample Utterances
For general information about custom slots, see Custom Slot Types.
Example: TOPPINGS - cheese | onions | ham (note: newlines displayed as | for brevity)

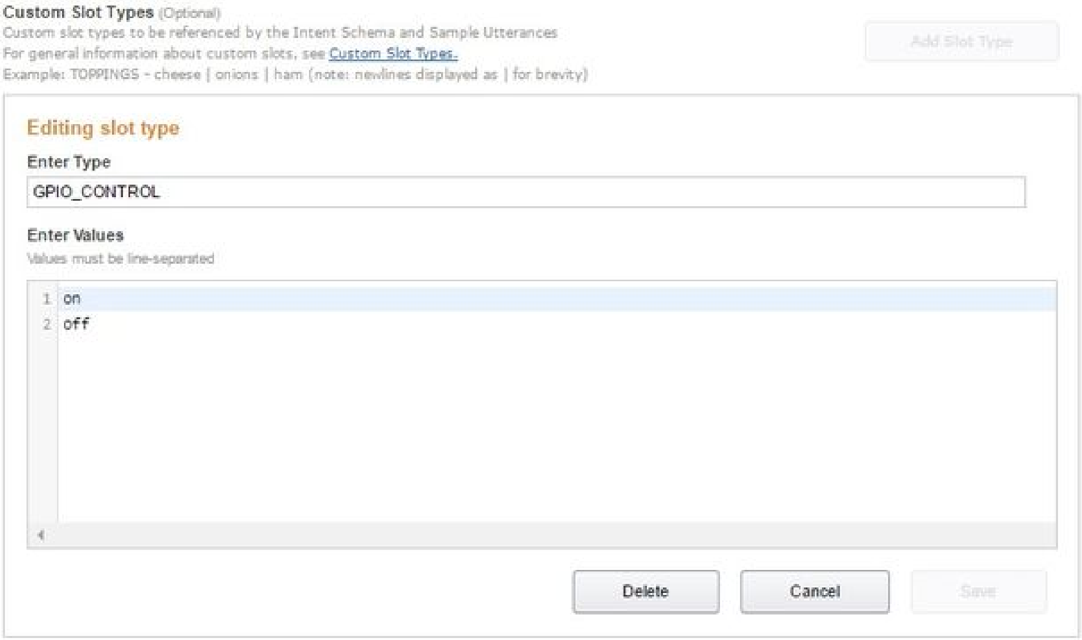
Type	Values
GPIO_CONTROL	on | off

Sample Utterances

These are what people say to interact with your skill. Type or paste in all the ways that people can invoke the intents. Learn more

Up to 3 of these will be used as Example Phrases, which are hints to users.

```
GPIOControlIntent turn pin {pin} {status}
```



Custom Slot Types (Optional)

Custom slot types to be referenced by the Intent Schema and Sample Utterances
For general information about custom slots, see Custom Slot Types.
Example: TOPPINGS - cheese | onions | ham (note: newlines displayed as | for brevity)

Editing slot type

Enter Type

GPIO_CONTROL

Enter Values

Values must be line-separated

```
on
off
```

将以下内容复制/粘贴到’Intent Schema’ 框中：

```
{

    "intents": [{

        "intent": "GPIOControlIntent",

        "slots": [{

            "name": "status",

            "type": "GPIO_CONTROL"

        },
                  {

            "name": "pin",

            "type": "AMAZON.NUMBER"

        }]

    }]

}
```

接下来，点击’Add Slot Type’ 并在’Enter Type’ 下写入’GPIO_CONTROL’，在’Enter Values’ 下写入：

```
on
off
```

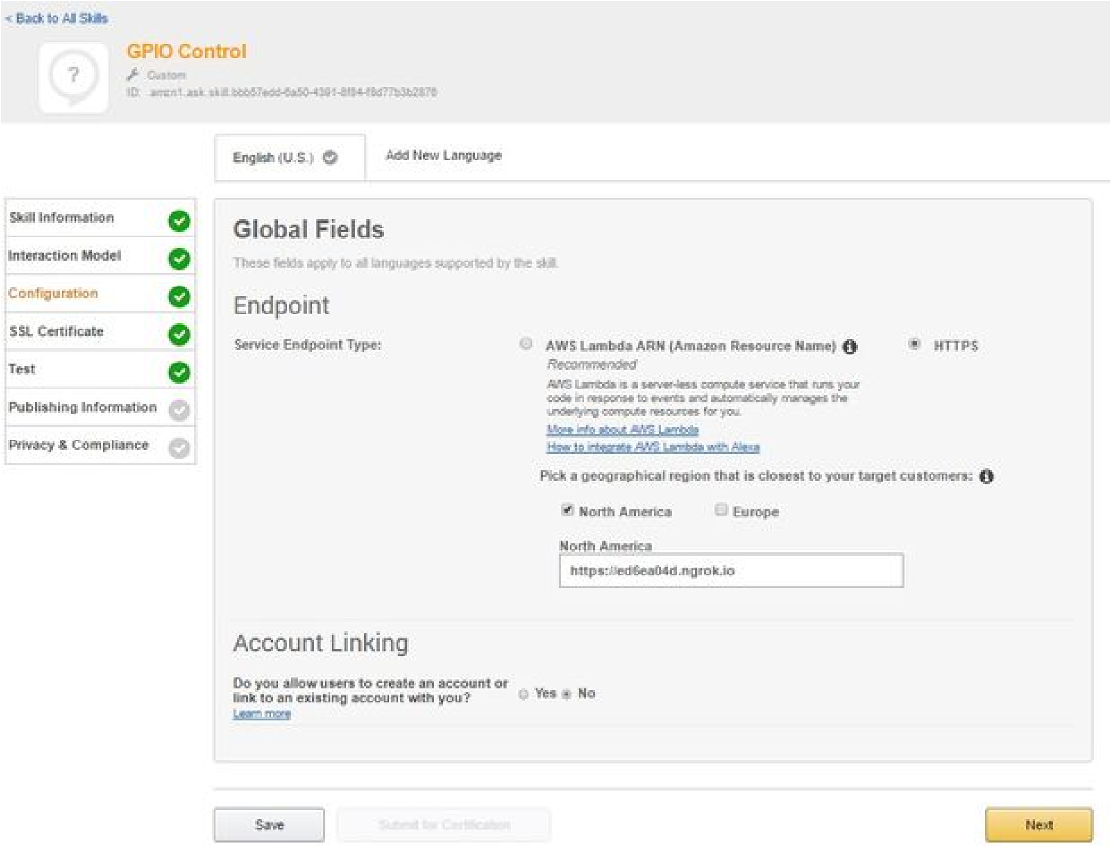
这只是一个简单的例子，’high’,’low’ 或者其他更丰富的词都可以添加。

将以下内容复制/粘贴到’Sample Utterances’ 框中：

```
GPIOControlIntent to turn pin {pin} {status}
```

点击’Save’ 然后’Next’。

步骤七：**Alexa Skill** - 配置



< Back to All Skills

GPIO Control

Custom

ID: amzn1.ask.skill.bbb57edd-6a50-4391-8f84-f8d77b3b2878

English (U.S.) | Add New Language

Skill Information ✓
Interaction Model ✓
Configuration ✓
SSL Certificate ✓
Test ✓
Publishing Information
Privacy & Compliance

Global Fields

These fields apply to all languages supported by the skill.

Endpoint

Service Endpoint Type: ○ AWS Lambda ARN (Amazon Resource Name) ● HTTPS

Recommended

AWS Lambda is a server-less compute service that runs your code in response to events and automatically manages the underlying compute resources for you.

More info about AWS Lambda

How to integrate AWS Lambda with Alexa

Pick a geographical region that is closest to your target customers:

☑ North America □ Europe

North America: https://ed6ea04d.ngrok.io

Account Linking

Do you allow users to create an account or link to an existing account with you? ○ Yes ● No

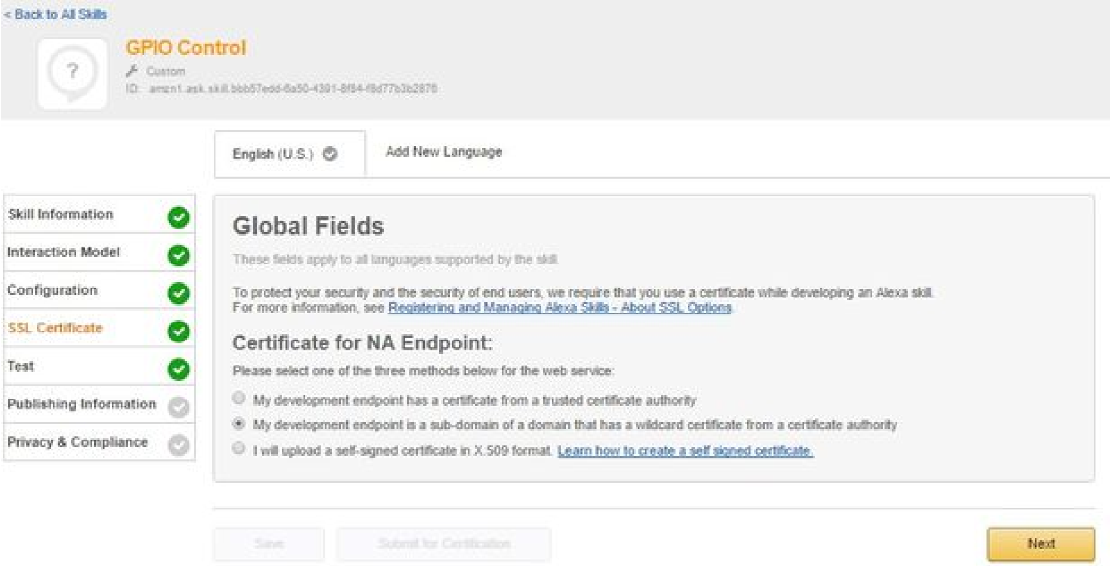
Learn more

选择’HTTPS’ 作为服务端点类型并选择一个区域。

从步骤 2 输入 ngrok URL，然后单击 ‘Next’。URL 应该如下：

```
https://ed6ea04d.ngrok.io
```

步骤八：**Alexa Skill - SSL** 证书 **#**



Global Fields

These fields apply to all languages supported by the skill.

To protect your security and the security of end users, we require that you use a certificate while developing an Alexa skill.
For more information, see Registering and Managing Alexa Skills - About SSL Options.

Certificate for NA Endpoint:

Please select one of the three methods below for the web service:

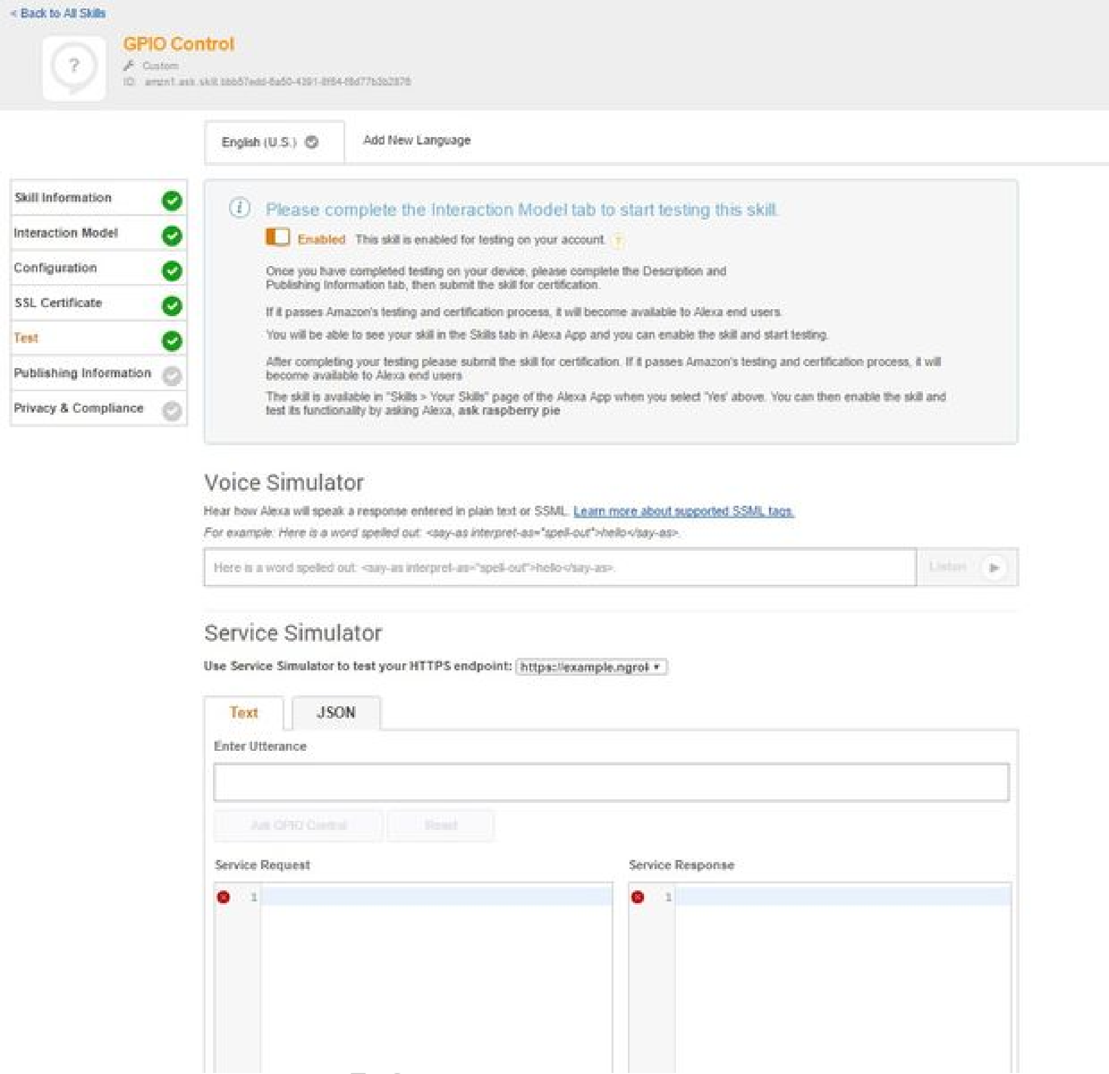
○ My development endpoint has a certificate from a trusted certificate authority

● My development endpoint is a sub-domain of a domain that has a wildcard certificate from a certificate authority

○ I will upload a self-signed certificate in X.509 format. Learn how to create a self signed certificate.

选择 ‘My development endpoint is a sub-domain of a domain that has a wildcard certificate from a certificate authority’ 选项并点击 ‘Next’。

步骤九：**Alexa Skill** - 测试



< Back to All Skills

GPIO Control

Custom

ID: amzn1.ask.skill.bbb57edd-8a50-4391-8f54-f8d77b3b2878

English (U.S.) | Add New Language

- Skill Information ✓
- Interaction Model ✓
- Configuration ✓
- SSL Certificate ✓
- Test ✓
- Publishing Information
- Privacy & Compliance

Please complete the Interaction Model tab to start testing this skill.

Enabled This skill is enabled for testing on your account.

Once you have completed testing on your device, please complete the Description and Publishing Information tab, then submit the skill for certification.

If it passes Amazon's testing and certification process, it will become available to Alexa end users.

You will be able to see your skill in the Skills tab in Alexa App and you can enable the skill and start testing.

After completing your testing please submit the skill for certification. If it passes Amazon's testing and certification process, it will become available to Alexa end users

The skill is available in "Skills > Your Skills" page of the Alexa App when you select 'Yes' above. You can then enable the skill and test its functionality by asking Alexa, **ask raspberry pie**

Voice Simulator

Hear how Alexa will speak a response entered in plain text or SSML. Learn more about supported SSML tags.

For example: Here is a word spelled out: <say-as interpret-as="spell-out">hello</say-as>.

Here is a word spelled out: <say-as interpret-as="spell-out">hello</say-as>. | Listen ▶

Service Simulator

Use Service Simulator to test your HTTPS endpoint: https://example.ngrok ▾

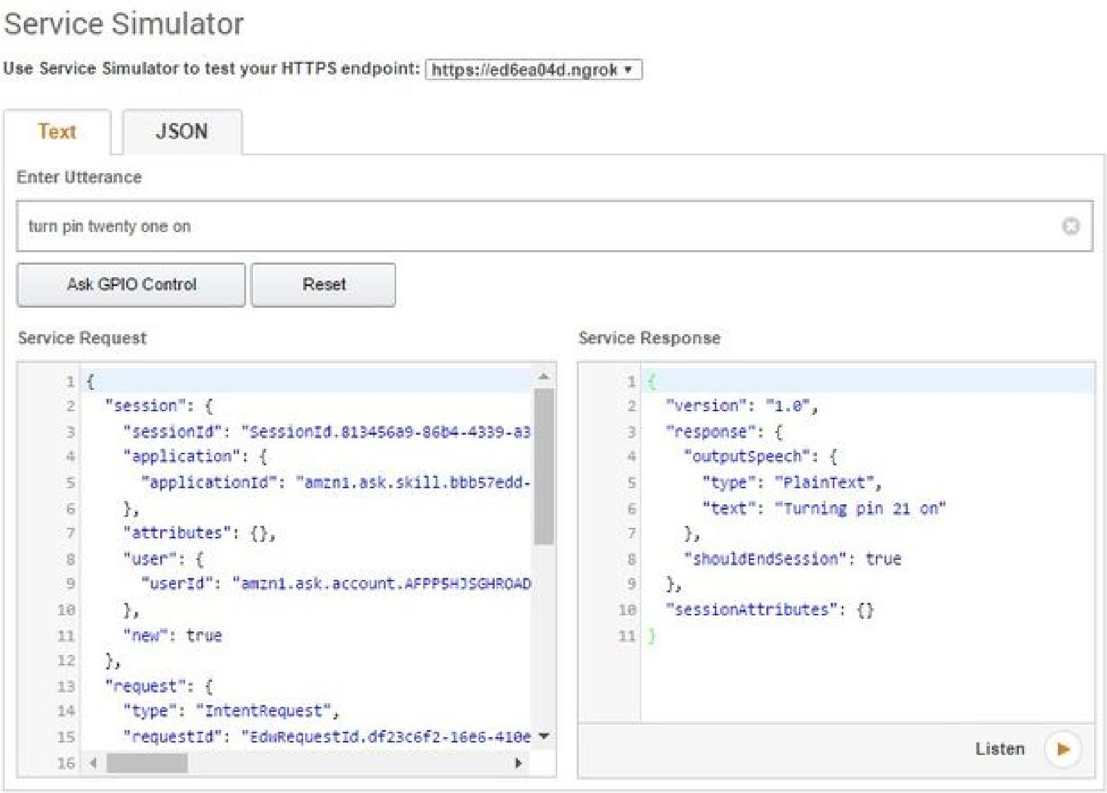
Text | JSON

Enter Utterance

Ask GPIO Control | Reset

Service Request

Service Response



Service Simulator
Use Service Simulator to test your HTTPS endpoint: https://ed6ea04d.ngrok ▾
Text
JSON
Enter Utterance

turn pin twenty one on

```
{
  "session": {
    "sessionId": "SessionId.813456a9-86b4-4339-a3
    "application": {
      "applicationId": "amzn1.ask.skill.bbb57edd-
    },
    "attributes": {},
    "user": {
      "userId": "amzn1.ask.account.AFPPSHJSGHROAD
    },
    "new": true
  },
  "request": {
    "type": "IntentRequest",
    "requestId": "EdwRequestId.df23c6f2-16e6-410e
```

```
{
  "version": "1.0",
  "response": {
    "outputSpeech": {
      "type": "PlainText",
      "text": "Turning pin 21 on"
    },
    "shouldEndSession": true
  },
  "sessionAttributes": {}
}
```

如果一切设置正确，您现在应该看到一个类似于上面第一个图像的屏幕。技能现已启用，可以通过使用以下语法连接到 AWS 开发人员帐户的任何 Amazon Echo 设备（或 http://echosim.io/）进行访问：

```
Alexa, tell Raspberry Pi to turn pin {pin number} {on or off}
```

例如：

```
Alexa, tell Raspberry Pi to turn pin twenty one on
```

如果您有任何问题，测试 Skill 的最简单的方法是使用测试页面上的语音或服务模拟器。尝试在服务模拟器下的'Enter Utterance' 框中输入以下内容：

```
turn pin twenty one on
```

然后点击 'Ask GPIO Control' 运行测试。

步骤十：包装

测试完成后，无需再继续进行 Skill 设置，因为剩下的已公开发布。据我所知，没有什么可以阻止使用者继续以“开发”模式进行个人使用，尽管如此，我的最终目标是使用此功能将 Alexa 整合到我现有的 Raspberry Pi 智能家居设置中，以便一切都可以自动控制。

这是我的第一个指导，所以任何反馈是赞赏。如果您有任何问题或麻烦设置，我将尽全力帮助。祝你好运！

步骤十一：优点 - 使用单词而非引脚号

如果您已遵循上述所有步骤，并希望添加更多功能，您可以通过以下步骤创建一个可以使用 'fan' 或 'light' 等实际单词进行调用的功能。

将以下代码添加到 gpio_control.py :

```
@ask.intent('LocationControlIntent', mapping={'status': 'status',
    'location': 'location'})
def location_control(status, location):

locationDict = {
    'fan': 12,
    'light': 21
}

targetPin = locationDict[location]

GPIO.setup(targetPin, GPIO.OUT)

if status in ['on', 'high']:        GPIO.output(targetPin, GPIO.HIGH)
if status in ['off', 'low']:        GPIO.output(targetPin, GPIO.LOW)

return statement('Turning {} {}!'.format(location, status))
```

复制/粘贴以下代码到 'Intent Schema' 框中:

```
{
    "intents": [{
            "intent": "LocationControlIntent",
            "slots": [{
                "name": "status",
                "type": "GPIO_CONTROL"
            },
                      {
```

```
                "name": "location",
"type": "LOCATION"            }]
     }]
}
```

接下来，点击 ‘Add Slot Type’ 并在’Enter Type’ 写入’LOCATION’。在 ‘Enter Values’ 下，记下您想要用来实现此目的的任何字词，例如：

```
fan
light
television
```

复制/粘贴以下代码到 ‘Sample Utterances’ 框中：

```
LocationControlIntent to turn {location} {status}
LocationControlIntent to change the {location} to {status}
```

点击 ‘Save’，它将更新您的设置，之后新的短语将可用。例：

```
Alexa, tell Raspberry Pi to turn fan on
Alexa, tell Raspberry Pi to change the light to off
```

原文链接：https://www.wandianshenme.com/play/□□amazon-echo□python□□raspberry-pi-gpio