

将 **Arduino** 快速变成简易的物联网 协议 **LoRa** 包嗅探器

Phodal Huang

September 8, 2017

目录

步骤 1: 选择您的 **Arduino** 3

步骤 2: 下载 **RadioHead** 库 4

步骤 3: 将代码复制到新的 **INO** 中 4

步骤 4: 分析您的网络流量 5

玩点什么: <https://www.wandianshenme.com>

原文链接:<https://www.wandianshenme.com/play/build-an-arduino-to-lora-iot-sniffer>

如果你在通信上工作过一段时间,你就会意识到,如果你能获得一个真正帮助你调试工作的外部视角,它将能大幅地提高我们的工作效率。这就是这个项目所在,把 **Arduino** 变成一个 **LoRa** 无线电包嗅探器。在您的工具箱中,您将能够看到两个或多个 **LoRa** 无线电节点之间的流量,这将帮助您排除那些通过 **LoRa** 网络进行通信时开发中遇到的问题。

例如:我们在开发 **LoRa Gateway** 产品时,我们希望更轻松地检查不同类型的性能调优变化带来的影响。我们可以使用传统的“编程”工具,例如断点或调试、**print** 语句,但是这些技术会对它们的实际效果造成影响。通过使用如此处所示的“网络探测器”或“数据包嗅探器”,您可以在不更改执行通信的软件的情况下,监视设备通信。数据包嗅探方法已经使用了多年,这个项目只是一种便宜的方法,可以添加到您可能已经拥有的部分的工具包中。

对于我想测试的系统,我正在使用 **Raspberry PI 3** 运行网关软件、一个 **Adafruit Feather MO** 与 **RFM95 LoRa Radio** (北美的 **900MHz**) 作为“终端节点”的传感器设备。当我对网关或设备软件进行调整时,**LoRa** 数据包嗅探器允许我查看通过网络的定时 (使用 **millis ()**) 和 **LoRa** 数据包的内容。

步骤 1: 选择您的 **Arduino**

我使用 **Adafruit Feather MO** 与 **RFM95 LoRa Radio** 作为我的数据包嗅探器的基础硬件。我为这个项目选择了这个设备,因为我需要的一切都在一个电路板上。但是,这个项目可以与其他 **Arduino\Arduino** 类开发板一起构建。在 **ino** 文件中,我们已经为其他板卡提供了 **GPIO** 设置 (感谢 **Adafruit**); 它包括了具有集成 **LoRa** 无线电的 **Feather MO**, 带有 **Adafruit** 分线板的 **Arduino** 或使用 **LoRa** 的 **Feather 32u4**, **Feather MO** 或 **Teensy 3.x**。

确保您购买了您选择正确频段的新 **LoRa** 设备; 为您的地理区域选择正确的频率,并匹配您要监视的现有设备:

- **Adafruit Feather MO with RFM95 LoRa Radio**
- **Adafruit LoRa RFM95 Breakout**
- **Adafruit LoRa Radio FeatherWing - RFM95W 900 MHz**

步骤 2: 下载 RadioHead 库

你将需要两个来自互联网的代码，用来构建这个项目，它们既是开源的，也是免费的。

- RadioHead RF 库 (AirSpacye 公司提供)
- 数据包嗅探器应用的 INO 文件

AirSpacye 开发了用于支持 RF95 无线电 (来自 HopeRF) 的嵌入式微处理器的 RadioHead 分组无线电库。这是一个优秀的库，支持『步骤 1』中提到的 Arduino 类型的开发板。它是开源的，根据 GPL 版本 2 许可。您可能已经在您的项目中使用了 Radio Head 库，但如果您还没有使用它。可以从 Radio Head 网站下载这个库：

[Radio Head library site](#)

一旦下载了这个库，要将 Radio Head (RH) 库 (zip 包的形式) 的全部内容解压出来，并放在 Arduino IDE 库文件夹下的文件夹中。

步骤 3: 将代码复制到新的 INO 中

您可以从 LooUQ 的 GitHub 上下载 Arduino 的嗅探器程序。其代码库的链接是：

<https://github.com/LooUQ/Arduino-LoRa-Sniffer>

一旦下载完成，并复制 Arduino 程序到文件夹中，。将需要查看和或更新草图中的某些设置，以满足您自己的需要 (大概在第 18 行的位置)。相关内容如下：

- 输出模式：详细或分隔符 (delimited)
- 如果设置为分隔输出，则可能需要更改分隔符字符 (默认为 '□')

```
1 / *在这里定义你想要的输出格式* /
2 OutputMode mode = delimited;
3 #define DELIMETER_CHAR '~'
4 /* end output format defintion */
5
6 The default frequency for the radio is set to 915 MHz, you may need to
   change this based on your location and the frequency of your other
   radios (at about line 65).
7 // Change to 434.0 or other frequency, must match RX's freq!
8 #define RF95_FREQ 915.0
```

步骤 4: 分析您的网络流量

开始你的网络...

随着您的 LoRa 网络运行, 启动嗅探器程序并开始收集。

如果想保存到磁盘, 您可以使用任何您喜欢的终端应用程序, 但不能使用 **Arduino IDE** 串行监视器。不幸的是, **Arduino IDE** 串行监视器目前无法保存串行端口流。在内部我们使用并可以推荐 **Tera Term**; **Tera Term** 具有许多功能: 包括自动串行端口检测和写入磁盘。

如果您使用分隔符, 则可以使用 **Microsoft Excel**, **Apache OpenOffice** 或 **Google** 表格离线查看网络跟踪。

希望这有助于某些人, 更快地调试一些东西。

Greg @ LooUQ

原文链接: [Turn an Arduino Into a Quick & Easy LoRa Packet Sniffer](#)

原文链接: <https://www.wandianshenme.com/play/build-an-arduino-to-lora-iot-sniffer>