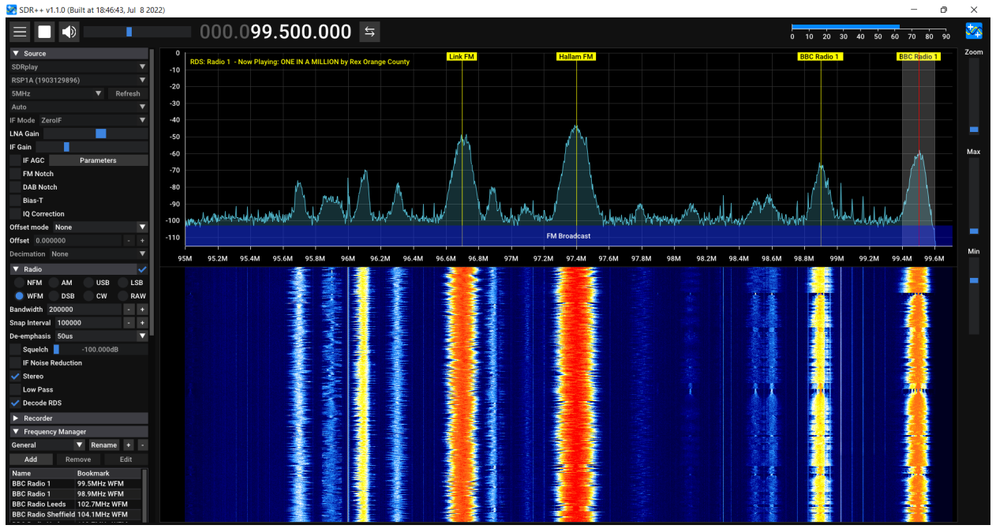
SDR++ USER GUIDE

For SDR++ up to version 1.1. December 2022

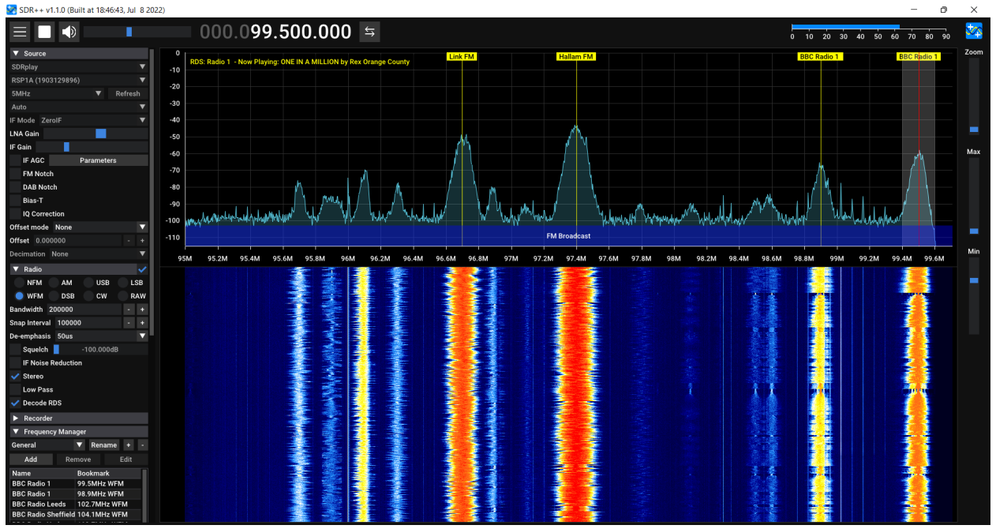
Download from https://sdrpp.org



SDR++ 用户指南

适用于 SDR++，直到版本 1.1。2022 年 12 月

从 https://sdrpp.org 下载



Contents

Click to select links

目录

点击以选择链接

Introduction

SDR++ is a cross-platform and open source software defined radio (SDR) program that will work with many different SDR receivers. Its aim is to be bloat-free and simple to use.

In this user guide we take the same approach - we'll not only tell you about its features, but also explain what these do and how it can help your SDR listening experience.

You'll also find explanations of some of the more advanced features, so you can move on to those as you get more experienced.

Features

- Wide hardware support
- Cross-platform (Windows, Linux, Mac, BSD - plus Android in beta)
- Wide hardware support including version for the Raspberry Pi
- Full waterfall
- Modular design with community plugins
- Multiple VFOs so you can listen to more than one frequency at the same time - within the receive bandwidth - using a single SDR receiver
- Built in server to operate your SDR receiver remotely
- Range Scanner

Supported SDR receivers

- Adalm Pluto SDR
- Airspy - all models
- BladeRF
- CaribouLite (planned)
- Ettus USRPs (in beta)
- HackRF (see troubleshooting section about removing soapy source)
- Hermes Lite 2 (experimental)
- Lime SDRs
- Per Vices Noctar (planned)

介绍

SDR++ 是一个跨平台的开源软件定义无线电（SDR）程序，可与多种不同的 SDR 接收器配合使用。它的目标是无负担且易于使用。

在本用户指南中，我们采取相同的方法——我们不仅会告诉您它的功能，还会解释这些功能的作用以及它如何帮助您的SDR聆听体验。

您还会找到一些更高级功能的解释，这样当您经验增加时就可以开始使用这些功能。

功能

- 广泛的硬件支持
- 跨平台（Windows、Linux、Mac、BSD——以及测试中的 Android）
- 广泛的硬件支持，包括适用于树莓派的版本
- 全瀑布
- 具有社区插件的模块化设计
- 多个VFO，允许您在接收带宽范围内使用单个SDR接收机同时监听多个频率
- 内置服务器，可远程操作您的SDR接收器
- 范围扫描器

支持的 SDR 接收器

- Adalm Pluto SDR
- Airspy - 所有型号
- BladeRF
- CaribouLite（计划中）
- Ettus USRP（测试版）
- HackRF（请参阅故障排除部分，了解如何移除含皂源）
- Hermes Lite 2（实验版）
- 酸橙 SDRs
- 按照夜恶习（计划中）

- PicoSDR (coming soon)
- RFSpace devices (in beta)
- RTL-SDR dongles
- SDRPlay receivers - providing the API is installed and running
- SPECTRAN V6 (Coming soon)

Other receivers supported by SoapySDR, when installed, are supported but may need additional modules added to Soapy and further configuration e.g. the FUNcube dongle pro+.

SDR++ will also connect to remote receivers using SDR++ Server or SpyServer.

You may need to install the correct drivers for your SDR receiver, but once that's done SDR++ should pick it up when you select it as the source.

New in version 1.1.0

- Complete rewrite of the digital signal processing (DSP) code for clearer audio, better CPU and memory performance and higher selectivity
- Range scanner - scans a frequency range for active signals (experimental)
- Invert IQ option - when operating SDR++ as a panadapter on radios with IF out connections that require this
- Rigctl client - allows use of SDR++ as panadapter (not yet in nightly releases but may be built)
- Radio Data System (RDS) decoding on broadcast FM signals
- IF noise reduction to pull out weak FM signals
- Improvements to noise blanker
- Ability to adjust more automatic gain control parameters
- FM low pass filter is now selectable, rather than set as ‘on’

- PicoSDR（即将推出）
- RFSpace 设备（测试版）
- RTL-SDR 加密狗
- SDRPlay 接收器 - 前提是 API 已安装并运行
- SPECTRAN V6（即将推出）

其他由 SoapySDR 支持的接收器，在安装后也可以使用，但可能需要向 Soapy 添加额外模块并进行进一步配置，例如 FUNcube Dongle Pro+。

SDR++ 还将使用SDR++ 服务器或SpyServer连接到远程接收器。

您可能需要为您的 SDR 接收器安装正确的驱动程序，但一旦完成，当您将其选为源时，SDR++ 应该能够识别它。

版本 1.1.0 新增功能

- 完全重写数字信号处理（DSP）代码，以实现更清晰的音频、更好的 CPU 和内存性能以及更高的选择性
- 范围扫描器 - 扫描特定频率范围内的活动信号（实验性）
- 反转 IQ 选项 - 在将 SDR++ 作为具有 IF 输出连接且需要此功能的收音机的泛波器操作时
- Rigctl 客户端 - 允许将 SDR++ 用作全景适配器（尚未在夜间版本中提供，但可以构建）
- 广播调频信号的无线电数据系统 (RDS) 解码
- 如果降噪以提取微弱的 FM 信号
- 噪声消除器的改进
- 能够调整更多自动增益控制参数
- 现在可以选择 FM 低通滤波器，而不是默认设置为“开启”

Installing SDR++

Hardware requirements for PCs and MACs

If you are installing on a PC, it needs to have a graphics card or chip - and up to date drivers - that support OpenGL 2.1. Without it SDR++ will not run in graphical mode - only as a server.

You can check what version of OpenGL you are running on

- Linux by using the following command in a terminal:

```
glxinfo | grep "OpenGL version"
```

- On MAC and Windows by downloading and installing the free OpenGL Extensions Viewer

The processing performed by SDR++ can be CPU intensive, so some older and less powerful hardware may struggle. You should have no problems, though, with a PC or laptop manufactured in the last 5 years, and SDR++ has been known to run on some machines that are 10 years old. To minimise the load, you can

- close other programs
- reduce the bandwidth you are sampling
- use decimation where available
- turn off other SDR++ features that use the CPU, such as stereo sound, noise reduction or the noise blanker
- run a single VFO only

Downloading

You can download SDR++ from sdrpp.org. You can use the 'release' version, but it's best to use the nightly build - it has the latest features and is just as stable. The latest release may be an earlier version number but this manual covers additional features only found in the nightly release.

正在安装 SDR++

电脑和 Mac 的硬件要求

如果你在 PC 上安装，它需要有支持 OpenGL 2.1 的显卡或芯片，并且驱动程序需要是最新版本。没有这些，SDR++ 将无法以图形模式运行——只能以服务器模式运行。

你可以检查你正在运行的 OpenGL 版本

- 在终端中使用以下命令在 Linux 上操作：

```
glxinfo | grep "OpenGL 版本"
```

- 在 MAC 和 Windows 上，通过下载并安装免费的 OpenGL 扩展查看器

SDR++ 执行的处理可能对 CPU 要求较高，因此一些较旧和性能较弱的硬件可能会运行困难。不过，使用过去五年内制造的 PC 或笔记本电脑通常没有问题，而已知 SDR++ 甚至可以在一些十年前的机器上运行。为了减少负载，你可以

- 关闭其他程序
- 降低你正在采样的带宽
- 在可用情况下使用抽样
- 关闭使用 CPU 的其他 SDR++ 功能，例如立体声、降噪或噪声消除器
- 仅运行一个 VFO

正在下载

您可以从 sdrpp.org 下载 SDR++。您可以使用“正式版”，但最好使用夜间构建版——它具有最新功能，而且同样稳定。最新的正式版可能版本号较低，但本手册介绍的功能仅在夜间版本中提供。

If you want to go direct to download SDR++, then you can now download a bang up to date nightly build of SDR++ for any supported operating system, without needing to create a GitHub account. Go to https://www.sdrpp.org/nightly.

This release is automatically updated every time there's a new nightly build.

Alternatively, you can download the nightly build of SDR++ from the main SDR++ Github page If you have a Github account. Github accounts are free - join Github. On the nightly builds page, scroll down to 'Artifacts' and download the correct file for your operating system. The nightly builds have the latest features, and are usually as stable as the last release.

Windows

Download `sdrpp_windows_x64.zip`

Extract it to the directory of your choice. We'd recommend you create a folder on your main hard drive such as `C:\SDRPP` or one in your user folders.

To create a desktop shortcut, right click on the sdrpp.exe file and

- In Windows 10, select Send to -> Desktop (create shortcut)
- In Windows 11, select 'more options', then Send to -> Desktop (create shortcut),

You can then rename the shortcut on the desktop to whatever you want.

Linux

Debian-based (Ubuntu, Mint, etc)

There are different packages for different versions, so you need to know, for instance, which version of Ubuntu you are running.

You can find this out by opening a terminal and typing:

```
lsb_release -a
```


如果你想直接下载 SDR++，那么你现在可以下载适用于任何支持操作系统的最新夜间版 SDR++，无需创建 GitHub 帐户。访问 https://www.sdrpp.org/nightly。

每当有新的夜间构建版本时，此版本会自动更新 .

或者，如果你有 Github 帐户，可以从主要的 SDR++ Github 页面下载 SDR++ 的夜间构建版本。Github 帐户是免费的——加入 Github。在夜间构建页面上，向下滚动到“Artifacts”，然后下载适合你操作系统的文件。夜间构建版本拥有最新功能，并且通常与上一个发布版本一样稳定。

窗口

下载 sdrpp_windows_x64.zip

将其解压到您选择的目录。我们建议您在主硬盘上创建一个文件夹，例如 C:\SDRPP，或者在您的用户文件夹中创建一个文件夹。

要创建桌面快捷方式，请右键单击 sdrpp.exe 文件并

- 在 Windows 10 中，选择“发送到 - 桌面（创建快捷方式）”
- 在 Windows 11 中，选择“更多选项”，然后选择 发送到 - 桌面（创建快捷方式）

然后你可以将桌面上的快捷方式重命名为你想要的任何名称。

Linux

基于 Debian 的（Ubuntu、Mint 等）

不同版本有不同的软件包，因此你需要知道，例如，你正在运行的是哪个版本的 Ubuntu。

你可以通过打开终端并输入以下命令来了解这个信息：

```
lsb_release -a
```

Extract the compressed file to the directory of your choice.

When you have downloaded the correct release, in a terminal, run:

```
sudo apt install libfftw3-dev libglfw3-dev libglew-dev libvolk2-dev
libsoapysdr-dev libairspyhf-dev libiio-dev libad9361-dev librtaudio-dev
libhackrf-dev zstd
```

If libvolk2-dev is not available, use libvolk1-dev. This installs some dependencies that you might not have.

Then you can install SDR++. You can do this from a terminal (where you'd replace the .deb file with the one you've chosen):

```
sudo dpkg -i sdrpp_debian_amd64.deb
```

This step can be left out if you want to use a package installer to install the program file. For instance you could use Gdebi to install it.

Alternatively, if you have a spare PC or laptop, you can install Dragon OS as its operating system. This is an SDR focused Ubuntu-based distribution that comes with a nightly release of SDR++ already loaded and configured.

Arch-based

Install the latest release from the sdrpp-git AUR package

Raspberry Pi

SDR++ has a package file for the Raspberry Pi operating system 32 bit OS.

It will operate best on the Pi 4 and 400, and satisfactorily on the Pi 3. It is not recommended for earlier models or the Pi Zero.

将压缩文件解压到您选择的目录。

当你下载了正确的版本后，在终端中运行：

```
sudo apt 安装 libfftw3-dev libglfw3-dev libglew-dev libvolk2-dev libsoapysdr-dev
libairspyhf-dev libiio-dev libad9361-dev librtaudio-dev libhackrf-dev zstd
```

如果 libvolk2-dev 不可用，请使用 libvolk1-dev。这将安装一些你可能没有的依赖项。

然后你可以安装 SDR++。你可以通过终端来进行安装（在这里你需要将 .deb 文件替换为你选择的文件）：

```
sudo dpkg -i sdrpp_debian_amd64.deb
```

如果您想使用软件包安装程序来安装程序文件，这一步可以省略。例如，您可以使用 Gdebi 来安装它。

或者，如果你有一台备用的电脑或笔记本电脑，你可以将 Dragon OS 安装为其操作系统。这是一个以 SDR 为重点的基于 Ubuntu 的发行版，内置并配置了 nightly 版本的 SDR++ 。

基于Arch的

从 sdrpp-git AUR 软件包安装最新版本

树莓派

SDR++ 有一个适用于树莓派操作系统32位的安装包文件。

它在 Pi 4 和 Pi 400 上运行最佳，在 Pi 3 上运行也令人满意。不推荐用于更早的型号或 Pi Zero。

Download `sdrpp_raspios_bullseye_armhf` from the nightly releases. This contains a '.deb' file that you can extract from the file and install. In a terminal, navigate to the folder where you extracted it, and install it with the command

```
sudo apt install ./sdrpp_debian_armhf.deb
```

This will also create a shortcut to the program in the menu under 'Other'.

Other distributions

There are currently no existing packages for other Linux distributions. For these systems you'll have to build from source.

You may find issues with some distros when self-building. In particular, on Mint based distros there is an audio problem. You can find support for some common issues in the discussions or issues sections of the SDR++ Github pages.

SDR++ will compile and run on the Pine phone although support is experimental at the moment.

MacOS

Some MacOS devices may be able to use the installer downloadable from the Releases page. You should install the zstd library first. To do this, first install Brew and then, in a terminal:

```
brew install zstd
```

For instructions on building SDR++ yourself, go to the SDR++ Github instructions. This is not straightforward, so we would not recommend it except for developers.

Android

For the Android version you need

从每夜版本中下载 sdrpp_raspios_bullseye_armhf。该版本包含一个可以从文件中提取并安装的 '.deb' 文件。在终端中，导航到你提取文件的文件夹，然后使用命令进行安装

```
sudo apt 安装 ./sdrpp_debian_armhf.deb
```

这还将在菜单的“其他”下创建该程序的快捷方式。

其他发行版

目前没有适用于其他 Linux 发行版的现成软件包。对于这些系统，您需要从源代码构建。

在自行构建时，你可能会遇到一些发行版的问题。特别是在基于 Mint 的发行版上，会存在音频问题。你可以在 SDR++ 的 Github 页面上的讨论或问题部分找到对一些常见问题的支持。

SDR++ 可以在Pine手机上编译和运行，尽管目前支持仍处于实验阶段。

MacOS

一些 MacOS 设备可能可以使用可从 Releases 页面下载的安装程序。你应该先安装 zstd 库。为此，首先安装 Brew，然后在终端中执行：

```
使用 brew 安装 zstd
```

有关自行构建 SDR++ 的说明，请访问 SDR++ 的 GitHub 说明。这并不简单，因此我们不推荐，除非您是开发者。

安卓

对于安卓版本，您需要

- **Android 9.0 or later**
- OpenGL 2.1or later - most phones with Android 9 or better should have this or a later version. You can check by installing OpenGL Extensions Viewer from the Play Store. This will also tell you what version of Android you are running.

Currently, only these SDR receivers and protocols are supported under the SDR++ Android app:

Airspy	Airspy HF+	HackRF	PlutoSDR (network only)
RFspace	RTL-SDR	RTL-TCP	SDR++ Server
SpyServer			

Instructions:

1. Download and install the SDR++ Android version. It should have a name like `sdrpp-1.1.0a.apk`. If the file has '.zip' after this then rename it to delete .zip so Android can recognise and install it. It's not yet available from the Play Store. When you select the downloaded file, your device should prompt you to open it with the Package Installer. You may have to change your Android settings to allow installing from this source.
2. Download and install the Android drivers if you have an SDRPlay device. Drivers for RTL-SDR devices are built into SDR++.
3. Connect your SDR receiver before starting SDR++. As most SDR receivers are connected through full size USB connectors, you will need an 'On the Go' (OTG) adapter to connect it. If you use a USB hub with this, you'll get best performance on a powered hub. If you are just connecting to an SDR++ server (or Spyserver) you don't need to do anything.

- **安卓 9.0 或更高版本**
- OpenGL 2.1 或更高版本 - 大多数搭载 Android 9 或更高版本的手机应具备此版本或更高版本。您可以通过从应用商店安装 OpenGL 扩展查看器来检查。它还会告诉您正在运行的 Android 版本。

目前，SDR++ Android 应用程序仅支持以下 SDR 接收器和协议：

空中间谍	Airspy HF+	HackRF	PlutoSDR（仅限网络）
RF空间	RTL-SDR	RTL-TCP	SDR++ 服务器
间谍服务器			

说明：

1. 下载并安装 SDR++ 的 Android 版本。文件名应类似于 sdrpp-1.1.0a.apk。如果文件名后面有“.zip”，请将其重命名以删除 .zip，以便 Android 可以识别并安装它。目前尚未在 Play 商店提供。当您选择下载的文件时，设备应提示您使用“安装包程序”打开它。您可能需要更改 Android 设置以允许从此来源安装。

2. 如果您有 SDRPlay 设备，请下载并安装 Android 驱动程序。RTL-SDR 设备的驱动程序已内置于 SDR++ 中。

3. 在启动 SDR++ 之前，请先连接您的 SDR 接收器。由于大多数 SDR 接收器通过标准 USB 接口连接，您需要使用‘On the Go’（OTG）适配器来连接它。如果您使用 USB 集线器连接，建议使用带电源的集线器以获得最佳性能。如果您只是连接到 SDR++ 服务器（或 Spyserver），则无需进行任何操作。

4. You may need to permit Android to load the drivers on first use. SDR++ should have necessary permissions, including network access, but you can grant or adjust these if your device settings are different.

5. You may be asked to allow SDR++ to access the receiver. If so, approve this and ensure you 'refresh' in the SDR source menu in SDR++. You may need to re-start some drivers or grant access again if you restart your phone or SDR++ crashes.

6. If you are reconnecting your SDR receiver you MUST press 'refresh' in the SDR source before pressing play or SDR++ may crash.

The SDR++ app works in the same way as other versions, in almost all respects. Resizing the menu and waterfall may, of course, take more precision with a finger or stylus on smaller touch screens.

Since phones usually have a high screen resolution, set the High DPI scaling in the 'Display' section of the menu to an appropriate level for your device. You'll need to restart SDR++ after this for the rescale to take effect.

Because you are using a power intensive app, and connecting an SDR receiver which draws power from your phone or Android device, this can draw a lot of power. This will reduce your battery level more quickly than usual.

If you have them, you can connect a powered usb hub and mouse to the OTG cable. You'll get more accurate control of the menu items and controls in this way. Also, battery drain will be less because the SDR receiver will draw power from the hub.

Known issues:

- the user interface has some problems
- there is no easy way to select a path for recording or a file for playback, and importing bookmarks to (and exporting them from) frequency manager is not possible
- the sink on Android may have higher latency
- SDR++ doesn't suspend power saving and, for instance, audio may become choppy if the display is blanked

4. 您可能需要允许 Android 在首次使用时加载驱动程序。SDR++ 应具有必要的权限，包括网络访问权限，但如果您的设备设置不同，您可以授予或调整这些权限。

5. 系统可能会要求您允许 SDR++ 访问接收器。如果是这样，请批准此操作，并确保在 SDR++ 的源菜单中“刷新”。如果您重启手机或 SDR++ 崩溃，可能需要重新启动某些驱动程序或再次授予访问权限。

6. 如果您正在重新连接 SDR 接收器，您必须在按播放之前在 SDR 源中按“刷新”，否则 SDR++ 可能会崩溃。

SDR++ 应用在几乎所有方面的工作方式与其他版本相同。当然，在较小的触摸屏上用手指或手写笔调整菜单和瀑布图的大小可能需要更多的精确操作。

由于手机通常具有较高的屏幕分辨率，请在菜单的“显示”部分将高 DPI 缩放设置为适合您设备的合适级别。完成此操作后，您需要重启 SDR++ 才能生效。

由于您正在使用一个高耗电的应用程序，并连接了一个从您的手机或安卓设备获取电力的 SDR 接收器，这可能会消耗大量电力。这会比平时更快地降低您的电池电量。

如果你有它们，你可以将带电的USB集线器和鼠标连接到OTG线。通过这种方式，你可以更准确地控制菜单项和控件。此外，电池消耗会更少，因为SDR接收器将从集线器获取电源。

已知问题：

- 用户界面存在一些问题
- 没有简单的方法来选择录音的路径或播放的文件，并且无法将书签导入到频率管理器（或从中导出）

- 安卓上的接收端可能延迟更高
- SDR++ 不会暂停省电模式，例如，如果显示屏被关闭，音频可能会出现断断续续的情况

- menus sometimes close when the app is in the background
- at some menu sizes, the app may crash - if this happens, start SDR++ with the Android device in landscape
- on Samsung devices, the keyboard doesn't always work.

Building it yourself

The SDR++ page on Github has instructions on building SDR++ from source on different platforms. SDR++ has been built on an older Chromebook using the rpi_install.sh script and changing libvolk1-dev to libvolk2-dev.

- 当应用程序在后台时，菜单有时会关闭
- 在某些菜单尺寸下，应用可能会崩溃——如果发生这种情况，请在Android设备横屏状态下启动SDR++
- 在三星设备上，键盘并不总是能正常工作。

自己建造

Github上的SDR++ 页面有关于在不同平台上从源码构建SDR++ 的说明。SDR++ 已在一台较旧的Chromebook上使用rpi_install.sh脚本构建，并将libvolk1-dev更改为libvolk2-dev。

Installing your SDR receiver and selecting Source

Install your SDR receiver

SDR++ works with a wide range of SDR receivers (often called 'dongles'). If your SDR receiver is supported then, once you've installed the driver, you can simply connect it and select it as the source.

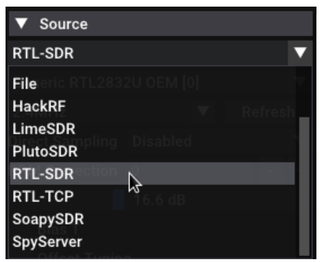
In some cases you may need to take other steps to make your receiver work. Doing this is beyond the scope of this manual, and you should look for help or drivers on the website of the manufacturer. For instance in the Linux operating system, though many receivers are supported without additional drivers, you may need to add modules to SoapySDR for some receivers to be recognised.

Get started with SDR++

Next, you need to launch SDR++. On Android you should have your SDR receiver plugged in before you start SDR++. On other operating systems, if you don't have it plugged in you can press the 'Refresh' button for SDR to detect it. You can launch SDR++ from any shortcut you've created for it, there may be a menu entry (in Linux) or you can launch it directly from the sdrpp.exe program in the folder you created in Windows.

'Source' - Selecting the SDR receiver you use

You'll then have to select the type of SDR receiver you are using from the '**Source**' module tab in the menu. If you can't see the menu, you can toggle it on and off by clicking on the menu icon in the top left. By default this will be on the top left hand side.



First select the make or type of SDR receiver by clicking on the arrow to see the drop down list.

If the SDR receiver is set up properly, you should also see it appear, if it's been detected by SDR++. Look for the model name and/or receiver serial number. If you have more than

安装您的 SDR 接收器并选择信号源

安装您的 SDR 接收器

SDR++ 兼容多种 SDR 接收器（通常称为“加密狗”）。如果你的 SDR 接收器受支持，那么在安装驱动程序后，你只需连接它并将其选为信号源即可。

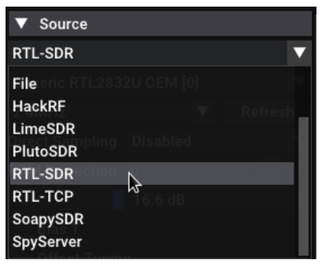
在某些情况下，您可能需要采取其他步骤使接收器工作。执行此操作超出了本手册的范围，您应当在制造商的网站上寻求帮助或查找驱动程序。例如在 Linux 操作系统中，虽然许多接收器无需额外驱动程序即可支持，但对于某些接收器，您可能需要为 SoapySDR 添加模块才能被识别。

开始使用 SDR++

接下来，您需要启动 SDR++。在 Android 上，您应该在启动 SDR++ 之前将 SDR 接收器插入。在其他操作系统上，如果未插入，您可以按下“刷新”按钮让 SDR 检测它。您可以通过为 SDR 创建的任何快捷方式启动 SDR++，在 Linux 中可能有菜单项，或者您可以直接从 Windows 中创建的文件夹中的 sdrpp.exe 程序启动它。

‘来源’ - 选择您使用的SDR接收器

然后，您需要在菜单中的“Source”模块标签中选择您正在使用的SDR接收器类型。如果看不到菜单，可以通过点击左上角的菜单图标来切换显示。默认情况下，它将显示在左上角。



首先通过点击箭头选择SDR接收机的品牌或类型，以查看下拉列表。

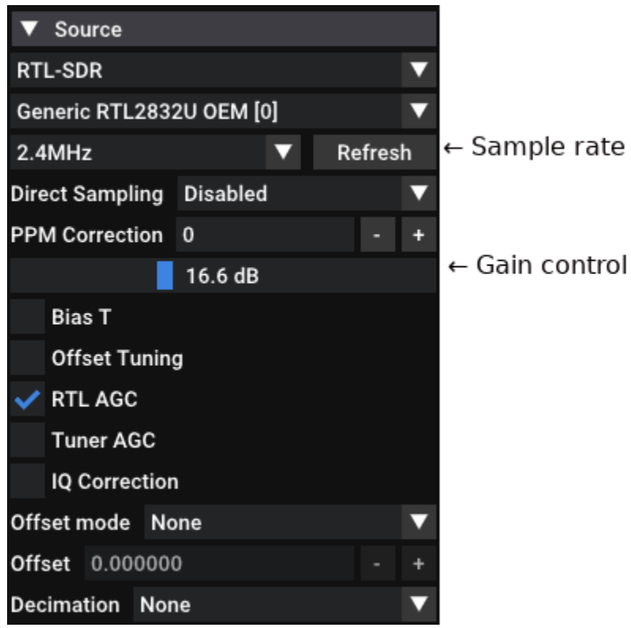
如果SDR接收器设置正确，如果它已被SDR++检测到，你也应该能看到它出现。请查找型号名称和/或接收器序列号。如果你有多个

one SDR receiver of a particular make then you need to select the one you wish to use. For instance for SDRPlay you might need to choose between an RSP1A and RSPduo or between two RSP1As.

In most cases that should be enough to get the SDR receiver working. You can now use the play button to start receiving. On its first run, SDR++ should open in the FM broadcast band. Adjust the gain - if your receiver has a manual gain control (see below) - and get listening!

There's more information on using SDR++ to connect to an SDR++ server, to an Airspy SpyServer, or to one on the SpyServer network in the Advanced Features section.

The source menu



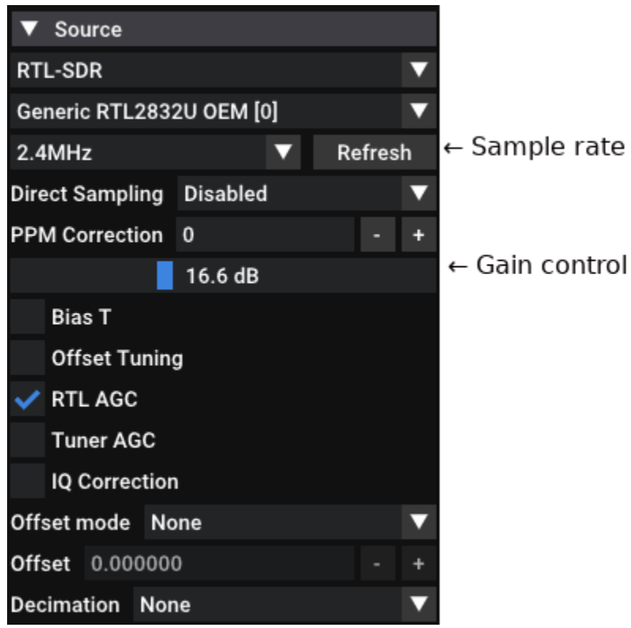
The options in this menu will vary according to which SDR receiver you use.

如果你有一台特定型号的 SDR 接收器，那么你需要选择你想使用的那一台。例如，对于 SDRPlay，你可能需要在 RSP1A 和 RSPduo 之间选择，或者在两台 RSP1A 之间选择。

在大多数情况下，这应该足够让SDR接收器工作。你现在可以使用播放按钮开始接收。在首次运行时，SDR++ 应该会在调频广播频段打开。如果你的接收器有手动增益控制（见下文），请调整增益，然后开始收听！

在高级功能部分，有关使用SDR++ 连接到SDR++ 服务器、Airspy SpyServer或SpyServer网络上的服务器的更多信息。

源菜单



此菜单中的选项将根据您使用的*SDR*接收器而有所不同。

Source options

To adjust some of these options you may need to stop SDR++ from receiving. These options will be ‘greyed out’ whilst receiving.

Gain control (or LNA Gain)

SDR receivers have different ways of controlling the ‘gain’ applied to a signal. Some will have a manual control, others will have an automatic gain control (AGC -see below), and others may have both. If you have AGC switched on, you won’t be able to adjust the manual gain.

With manual gain, you need to adjust the ‘gain control’ to get the right signal level. This adjusts the amount by which the signal is boosted and shows a reading in decibels - dB. Drag the blue bar to the right to increase the gain, and to the left to reduce gain.

If you tune to a known clear signal on the FM broadcast band, you should make a rough adjustment to the gain control to get a hiss free signal. Sometimes, surprisingly, reducing the gain may bring up a signal and make it clearer.

When you have listened to more signals you should adjust it again. Using a weak signal, adjust the gain to get the best ‘signal to noise ratio’ - where there is the biggest difference between the signal you want and the background hiss or other noise.

Some SDR receivers will also have an IF gain control which adjusts the gain at a different stage of the receive circuitry.

If SDR++ is working then you might want to skip ahead to the section on using SDR++ and the main panel. The rest of this section looks at the settings you can adjust to get the best from your SDR receiver. It includes some that you’ll only see if you’ve got a particular model of SDR receiver.

AGC (“automatic gain control”)

Whilst your gain control adjusts the level of radio signal (plus noise) coming into your SDR, the AGC automatically adjusts this to make weak signals stronger and to reduce strong signals so the overall volume is comfortable.

源选项

要调整其中的一些选项，您可能需要停止 SDR++ 的接收。在接收时，这些选项将显示为“灰色”。

增益控制（或低噪声放大器增益）

SDR 接收器有不同的方法来控制施加到信号上的“增益”。有些接收器有手动控制，有些则有自动增益控制（AGC——见下文），还有些可能两者兼有。如果你开启了 AGC，就无法调整手动增益。

在手动增益模式下，您需要调整“增益控制”以获得合适的信号水平。这会调节信号的增益量，并显示以分贝（dB）为单位的读数。将蓝色滑块向右拖动以增加增益，向左拖动以降低增益。

如果你调到调频广播频段上已知的清晰信号，你应该大致调节增益控制以获得无噪声的信号。有时，令人惊讶的是，降低增益可能会增强信号并使其更清晰。

当你听过更多信号后，应再次进行调整。使用弱信号时，调整增益以获得最佳“信噪比”——即所需信号与背景嘶嘶声或其他噪声之间差异最大的状态。

一些软件定义无线电接收器还会有一个中频增益控制，用于调节接收电路不同阶段的增益。

如果SDR++ 正在工作，那么你可能想直接跳到关于使用SDR++ 和主面板的章节。本节的其余部分将介绍你可以调整的设置，以充分发挥你的SDR接收机的性能。其中包括一些只有特定型号的SDR接收机才会显示的设置。

AGC（“自动增益控制”）

当你的增益控制调整进入SDR的无线电信号（加上噪声）水平时，AGC会自动进行调整，使微弱信号更强，并减弱强信号，从而使整体音量保持在舒适范围。

Depending on your SDR receiver you may have additional options to either choose between AGC built into the hardware or to specify the level. Some SDR receivers will have fixed levels of AGC, for instance: off, low and high on the Airspy HF+ Discovery.

Sample rate

The 'sample rate' determines how much of the radio frequency spectrum your SDR receiver can show at the same time on the main screen. This is limited by your SDR receiver's hardware.

For instance, the SDRplay RSP1A can show signals across up to 10 MHz on the spectrum display, whilst RTL-SDR dongles can manage 2 MHz, and the Airspy HF+ Discovery can show just 768kHz. The useful amount of the radio spectrum may be less than this - for instance about 610kHz for the HF+ Discovery.

A lower figure is not necessarily bad - with the HF+ Discovery it means that the hardware can reject strong singles on a nearby frequency that could swamp weak singles within the sample width.

It also takes a more powerful computer to process a higher sample rate. If you reduce the sample rate it can help a less powerful computer cope without interruption or glitches. Not all sample rates shown may be supported by the SDR receiver.

Bias-T

A Bias-T can generate an electric current that can travel down coaxial cable to power a remote RF amplifier, without interfering with the radio signal travelling the other way. These are available as a separately powered unit - such as with the MLA30+ magnetic loop - but some SDR receivers have their own Bias-T built in. Examples are the SDRplay range, BladeRF and the RTL-SDR Blog v3.

With those receivers you can tick the box in SDR++ to switch on their Bias-T.

You should refer to the manual and specifications of your SDR receiver to ensure that their internal Bias-T is compatible and has sufficient power for the attached amplifier. Incorrect use may damage your SDR receiver.

根据您的SDR接收器，您可能有额外选项，可以在硬件内置的AGC之间选择，或指定AGC的级别。一些SDR接收器将具有固定的AGC级别，例如：在Airspy HF+ Discovery上，AGC可设置为关闭、低或高。

采样率

“采样率”决定了您的 SDR 接收器在主屏幕上同时显示多少射频频谱。这受到您的 SDR 接收器硬件的限制。

例如，SDRplay RSP1A 可以在频谱显示上显示高达 10 MHz 的信号，而 RTL-SDR 加密狗只能处理 2 MHz，Airspy HF+ Discovery 只能显示 768 kHz。实际可用的无线电频谱可能比这更少——例如 HF+ Discovery 大约为 610 kHz。

较低的数值不一定是坏事——对于 HF+ Discovery 来说，这意味着硬件可以拒绝邻近频率上的强信号，从而防止这些信号淹没样本宽度内的弱信号。

处理更高的采样率也需要更强大的计算机。如果您降低采样率，可以帮助性能较低的计算机在不间断或无故障的情况下运行。并非所有显示的采样率都可能被SDR接收器支持。

偏置-T

偏置-T（Bias-T）可以产生电流，通过同轴电缆传输，为远程射频放大器供电，同时不会干扰沿相反方向传输的无线电信号。这类装置可以作为独立供电单元使用，例如 MLA30+ 磁环天线，但一些 SDR 接收器内置了自己的偏置-T。示例包括 SDRplay 系列、BladeRF 以及 RTL-SDR Blog v3。

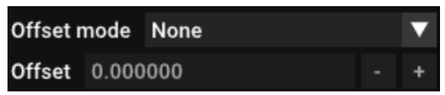
使用那些接收器，你可以在SDR++ 中勾选复选框以开启它们的偏置T。

您应参考 SDR 接收器的手册和规格，以确保其内部偏置-T 兼容并为所连接的放大器提供足够的功率。不正确的使用可能会损坏您的 SDR 接收器。

Offset tuning for up-converters and down-converters

You can have additional hardware connected to your SDR receiver that shifts a band of frequencies into the range of frequencies the SDR receiver can receive.

Select and unselect the 'Offset Tuning' box to activate or deactivate this once you have set the correct offset mode.



For instance, if your SDR receiver only covers VHF (30-300 MHz) and UHF frequencies (300 MHz upwards) and you want to listen to HF (3-30 MHz) or MF (300KHz to 3 MHz) frequencies, then you might use the Ham it Up converter. This is an 'up converter' and delivers signals raised in frequency so your SDR receiver can listen to them within its normal frequency coverage. It raises the signals up by 125 MHz so you'd hear, for instance, 7MHz by tuning to 132MHz in SDR++.

Of course you want to see the true frequency of the signal displayed in SDR++ so it actually displays (in the example given) 7MHz in the frequency readout. You can do this by selecting the type of converter in '**Offset Mode**' which covers:

- Spyverter
- Ham it Up
- DK5AV X band converter
- Ku band satellite LNA's (down converters)

This will set the right offset for those converters.

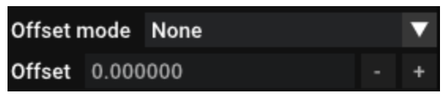
If you are not using a converter, the offset mode should be unticked or set to 'None'.

You can also set a custom offset. The figure in **'Offset'** can be set or adjusted by the *minus* '-' and *plus* '+' signs, or by typing in the offset figure in hertz. So '125000000' is for a 125 MHz *up*converter. Precede this figure with a *minus* if you are using a *down* converter.

上变频器和下变频器的偏移调节

你可以将额外的硬件连接到你的 SDR 接收器，将一段频率范围转换到 SDR 接收器可以接收的频率范围内。

在设置好正确的偏移模式后，选中或取消选中“偏移调谐”框以激活或停用此功能。



例如，如果您的SDR接收机仅覆盖VHF（30-300 MHz）和UHF频段（300 MHz及以上），而您想监听HF（3-30 MHz）或MF（300 kHz到3 MHz）频段，那么您可能会使用Ham it Up转换器。这是一种“上变频器”，可以将信号频率提高，使您的SDR接收机能够在其正常频率覆盖范围内接收这些信号。它将信号提升125 MHz，因此例如，您调谐SDR到132 MHz就可以接收7 MHz的信号。

当然，你希望在SDR++ 中看到信号的真实频率，因此它实际显示（在给出的例子中）频率读数为7MHz。你可以通过在“偏移模式”中选择转换器类型来实现，这包括：

- 间谍转换器
- 卖弄表演
- DK5AV X 波段转换器
- Ku 频段卫星低噪声放大器（下变频器）

这将为那些转换器设置正确的偏移量。

如果您没有使用转换器，偏移模式应取消勾选或设置为“无”。

您也可以设置自定义偏移量。“偏移”中的数值可以通过减号“-”和加号“+”进行设置或调整，也可以直接输入赫兹值的偏移量。因此，“125000000”是用于125 MHz上变频器。如果您使用下变频器，请在此数值前加上负号。

If the frequency reading is still not correct, use a transmission on a known frequency to adjust it until it's correct. You could use a time signal like WWV or a strong VOLMET signal, or compare it with another receiver. On some less stable SDR receivers you may also need to correct the frequency using the PPM Correction setting.

IQ Correction

This setting will correct imbalances in the radio hardware that can lead to effects such as a 'DC spike' showing on the frequency display, and imbalances that can particularly affect the audio or the reception of data modes. Adjust the value until the spike or imbalance disappears or moves outside the VFO's passband.

Decimation

Decimation is a way of reducing the number of times a sample is taken of the radio signals coming into your SDR receiver. A setting of 2 means that 1 in every 2 samples will be taken, 4 means 1 in 4 and so on. To start with, leave it at zero.

Introducing decimation can increase the 'dynamic range' of the SDR receiver - the range of signal strengths over which it can successfully receive the required signals. Try different settings with your receiver and watch out for it stuttering.

Decimation reduces the frequency range sampled. For instance changing decimation to 4 on an SDRplay RSP1A reduces the frequency width you can view from 10 MHz to approximately 1.2 MHz. Increasing decimation may also reduce the quality of the received signal.

Source settings for specific SDR receivers

There are settings options which are unique, or significantly different, for various supported SDR receivers.

The source settings for SDR++ also covers

- listening to recordings of the RF Spectrum (also called the IQ signal): see the part of this guide on the 'Recorder' to learn more

如果频率读数仍然不正确，请使用已知频率的信号进行调整，直到读数正确。您可以使用像 WWV 这样的时间信号或强信号 VOLMET，也可以与另一台接收机进行对比。在一些稳定性较差的软件定义无线电（SDR）接收机上，您可能还需要使用 PPM 校正设置来修正频率。

智商矫正

此设置将校正无线电硬件中的不平衡，这些不平衡可能导致频率显示上出现“直流峰值”等现象，以及可能特别影响音频或数据模式接收的不平衡。调整该值，直到峰值或不平衡消失或移出 VFO的通带范围。

大规模毁灭

抽取是减少从进入 SDR 接收器的无线电信号中采样次数的一种方法。设置为 2 意味着每 2 个样本中取 1 个，设置为 4 意味着每 4 个样本中取 1 个，依此类推。初始时，将其保持为零即可。

引入抽取可以增加SDR接收器的“动态范围”——即其能成功接收所需信号的信号强度范围。尝试在接收器上使用不同的设置，并注意其是否出现卡顿。

抽取会减少采样的频率范围。例如，将 SDRplay RSP1A 的抽取设置为 4，会将可视频宽从 10 MHz 降至约 1.2 MHz。增加抽取倍数可能还会降低接收信号的质量。

特定SDR接收器的源设置

对于各种支持的SDR接收器，有些设置选项是独特的，或者有显著差异。

SDR++ 的源设置也涵盖

- 听RF频谱（也称为IQ信号）的录音：请参阅本指南中关于“录音器”的部分以了解更多信息

- connecting to an SDR++ server at a remote location: see the part of this guide on 'SDR++ Server'
- connecting to an Airspy remote server: see the part of this guide on 'Connecting to remote SDRs using SpyServer'.

RTL-SDR only source settings

Direct sampling

RTL-SDR's mostly cover VHF and UHF frequencies. Some RTL-SDR (and other) dongles can go to lower frequencies by using 'direct sampling'. This works but you can expect:

- poorer performance than on it's design frequencies
- poorer performance than an up converter like Ham it Up
- less resistance to strong signals on nearby frequencies
- radio transmissions - 'images' - appearing at an apparently different frequency from their actual frequency. For instance on the RTL-SDR Blog v3, you might hear a USB signal below 14 MHz appearing above 14 MHz, and in LSB.

To receive the lower frequencies - usually 500 kHz – 24 MHz - the RTL-SDR dongle must use direct sampling. Use the Source menu setting to do this - usually by selecting the **Q branch**. You can then use frequencies in the HF range, but VHF and UHF signals will no longer be present.

RTL AGC and Tuner AGC

These apply automatic gain control at the sampler stage or the tuner stage of the RTL-SDR dongle respectively. These hardware controls tend to be more broadband and less effective than using software gain control for these dongles.

PPM Correction

Inexpensive RTL SDR dongles sometimes do not have a stable or well aligned oscillator that is used for setting the frequency accurately. Some better ones such as the RTL-SDR blog v3 will have a temperature controlled crystal oscillator (TCXO) in the circuit which can prevent this.

- 连接到位于远程位置的SDR++ 服务器：请参阅本指南中关于“SDR++ 服务器”的部分-
连接到Airspy远程服务器：请参阅本指南中关于“使用SpyServer连接远程SDR”的部分。

仅 RTL-SDR 来源设置

直接采样

RTL-SDR 主要覆盖 VHF 和 UHF 频段。一些 RTL-SDR（以及其他）加密狗可以通过使用“直接采样”来接收更低频率。这种方法可行，但你可以预期：

- 性能比设计频率下更差
- 性能不如像 Ham it Up 这样的升频转换器
- 对附近频率的强信号阻抗较小
- 无线电传输——‘图像’——以表面上与实际频率不同的频率出现。例如，在 RTL-SDR Blog v3 上，你可能会听到一个低于 14 MHz 的 USB 信号出现在 14 MHz 以上，并且以 LSB 形式出现。

要接收较低的频率——通常为 500 kHz 至 24 MHz——RTL-SDR 加密狗必须使用直接采样。使用“源”菜单设置来执行此操作——通常选择 Q 分支。然后你可以使用 HF 范围的频率，但 VHF 和 UHF 信号将不再存在。

RTL 自动增益控制和调谐器自动增益控制

这些分别在RTL-SDR加密狗的采样器阶段或调谐器阶段应用自动增益控制。这些硬件控制往往更宽带，并且不如使用软件增益控制对于这些加密狗那样有效。

PPM 校正

廉价的 RTL SDR 加密狗有时没有稳定或良好对准的振荡器用于精确设定频率。一些较好的型号，例如 RTL-SDR 博客 v3，会在电路中使用温控晶体振荡器 (TCXO)，可以防止这个问题。

Other generic ones do not, which can mean the frequency passed to the SDR software is incorrect. If this is the case you will need to calibrate the receiver through SDR++ by entering a figure for the correction in 'PPM' - parts per million.

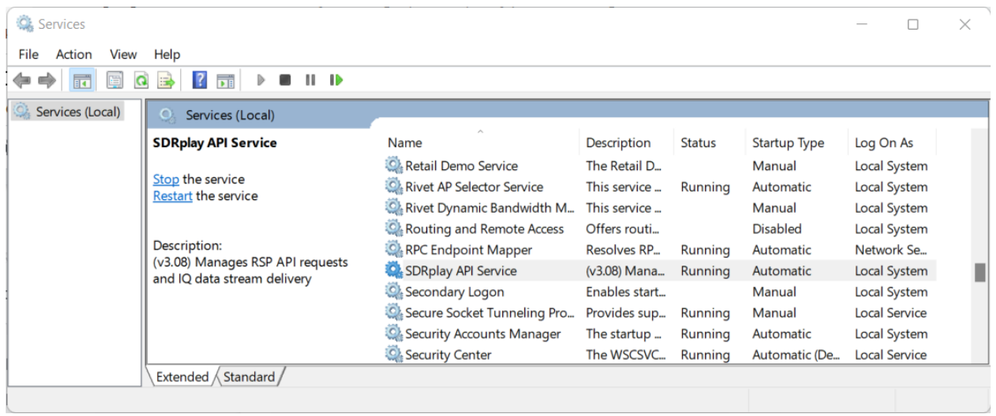
To calibrate your dongle and find the PPM figure use the `rtl_test` utility. This will have been included when you installed the drivers for your rtl-sdr dongle, and is available in Windows, MAC and Linux.

More about using rtl_test.

SDR Play only source settings

On Windows the SDRPlay API service that is required for SDR++ to recognise an SDR Play device may sometimes fail to start or have stopped. If this happens SDRPlay will not appear in the list of sources, or you may get a message that no device was found. First, wait a few seconds then press 'Refresh' in the source menu. If this does not work, you can re-start the service directly either by

- starting, then closing, SDRUno before you run SDR++
- changing the services settings to start the SDRplay API service, and change it to run automatically. You can find these settings by typing 'services' into the search function. If it's already set to automatic, re-start the service.



其他通用型号则不具备此功能，这可能意味着传递给SDR软件的频率不正确。如果是这种情况，您需要通过SDR对接收器进行校准，通过在“PPM”（百万分之一）中输入修正数值来实现。

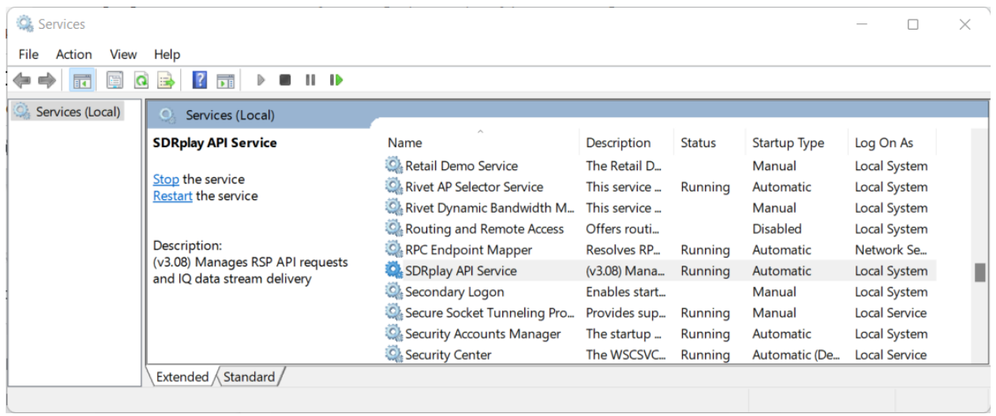
要校准您的加密狗并找到 PPM 数值，请使用 rtl_test 工具。安装 rtl-sdr 加密狗驱动程序时会包含该工具，并且在 Windows、Mac 和 Linux 上均可使用。

更多关于使用 rtl_test 的信息。

仅限SDR Play的来源设置

在 Windows 上，SDRPlay API 服务是 SDR++ 识别 SDR Play 设备所必需的，但有时该服务可能无法启动或已停止。如果发生这种情况，SDRPlay 将不会出现在源列表中，或者您可能会收到未找到设备的消息。首先，等待几秒钟，然后在源菜单中按“刷新”。如果这不起作用，您可以直接重新启动该服务，方法如下：

- 在运行 SDR++ 之前，先启动然后关闭 SDRUno
- 更改服务设置以启动 SDRplay API 服务，并将其设置为自动运行。您可以在搜索功能中输入“services”来找到这些设置。如果它已经设置为自动，请重新启动该服务。



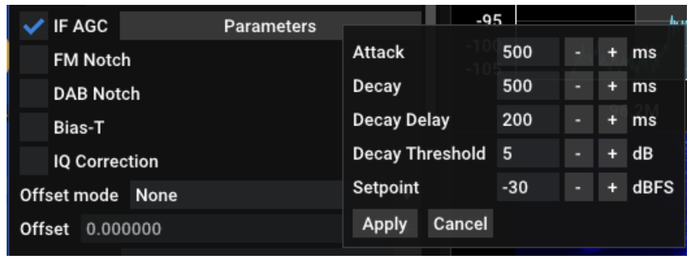
IF mode

This sets the ‘intermediate frequency’ used by an SDRPlay receiver. ZeroIF should be sufficient for most purposes. The ‘Low IF’ mode can be used for specialist uses but may generate an artificial ‘spike’ and signal on the spectrum display.

IF AGC

This controls the gain or amplification at the IF stage of the receiver automatically. There’s more about how ‘automatic gain control’ (AGC) works in the section above .

You can alter how the IF AGC works on an SDRPlay receiver by selecting ‘parameters’. You can control how quickly AGC kicks in to reduce strong signals or boost weak ones through the attack and decay settings, and adjust the signal level at which it will operate through the setpoint. Usually, the default settings will work well.



FM Notch / DAB Notch

These switch the notch filters for these bands on and off. This is to prevent your receiver being swamped by strong signals from these broadcast bands.

Antenna

This allows you to choose between the antenna inputs (Ports A, B, C) in the RSPdx and RSPduo.

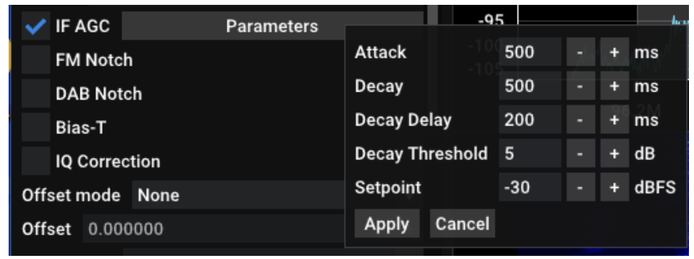
IF 模式

这设置了SDRPlay接收器使用的“中频”。对于大多数用途，ZeroIF就足够了。“低中频”模式可用于特殊用途，但可能在频谱显示上产生人为的“峰值”和信号。

如果
AGC

这会自动控制接收机中中频(IF)阶段的增益或放大。关于“自动增益控制”(AGC)的工作原理，前面的章节有更多说明。

您可以通过选择“参数”来改变SDRPlay接收器上IF AGC的工作方式。您可以通过调整攻击和衰减设置来控制AGC启动的速度，从而减弱强信号或增强弱信号，并通过设定点调整其工作的信号电平。通常，默认设置就能很好地工作。



FM 陷波 / DAB 陷波

这些开关可以打开或关闭这些频段的陷波滤波器。这样可以防止您的接收器被这些广播频段的强信号淹没。

天线

这使您可以在 RSPdx 和 RSPduo 中选择天线输入（端口 A、B、C）。

Airspy HF+ Discovery-only source settings

Attenuation

This switches in the receiver's internal attenuation which reduces the received signal strength. It works only when the AGC is switched off. Attenuation can be useful where you have a very strong signal. You can also increase the attenuation where you are receiving a lot of noise, as this can reduce it and will sometimes allow you to hear the wanted signal better.

HF LNA

The HF+ Discovery has an LNA or 'low noise amplifier'. The LNA will amplify (increase) the received signal (but also the noise).

Lime SDR only source settings

Antenna

The Lime SDR is normally sold with between 2 and 10 RF inputs. This control allows you to select the appropriate input.

Pluto SDR

IP

The USB connection to the Adalm Pluto is used as a serial console for sending commands and configuring the Pluto, but SDR programs access the receiver through the network connection it creates. By default this is 192.168.2.1, but you may have changed this when you configured the Pluto or when adding a USB ethernet adaptor. You insert the IP address in this field to access the Pluto.

Gain mode and PGA Gain

This allows you to set a fixed level of semi-automatic gain control. Choose between: manual, fast attack, slow attack and hybrid. If you choose manual, the gain will be set by the programmable-gain amplifier setting - the 'PGA' slider. The PGA slider is not accessible with the other settings.

仅限 Airspy HF+ Discovery 源设置

衰减

这会切换接收机内部的衰减，从而降低接收到的信号强度。只有在关闭自动增益控制（AGC）时才有效。当信号非常强时，衰减是有用的。您还可以在接收到大量噪声时增加衰减，因为这可以减少噪声，有时还能让您更清楚地听到所需信号。

高频低噪声放大器

HF+ Discovery 配备了 LNA 或“低噪声放大器”。LNA 会放大（增强）接收到的信号（但同时也会放大噪声）。

仅 Lime SDR 源设置

天线

Lime SDR 通常配备 2 到 10 个射频输入。此控制选项允许您选择合适的输入。

冥王星 SDR

IP

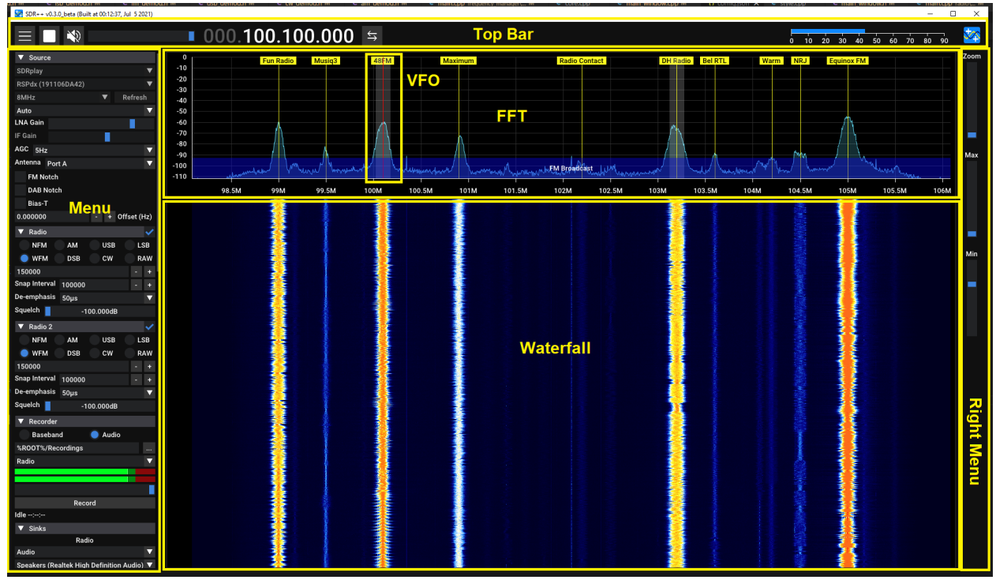
连接到 Adalm Pluto 的 USB 用作串行控制台，用于发送命令和配置 Pluto，但 SDR 程序通过其创建的网络连接访问接收器。默认情况下，该地址为 192.168.2.1，但在配置 Pluto 或添加 USB 以太网适配器时，你可能已更改了此地址。你需要将 IP 地址输入此字段以访问 Pluto。

增益模式和PGA增益

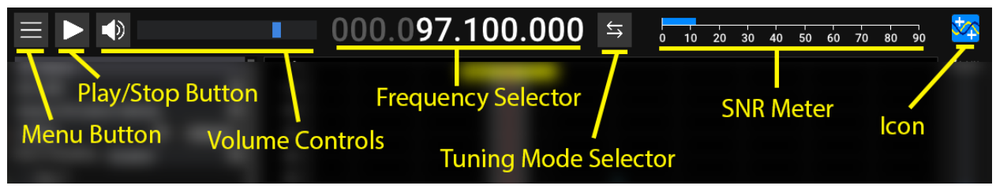
这允许您设置固定水平的半自动增益控制。可选择：手动、快速响应、慢速响应和混合模式。如果选择手动，增益将由可编程增益放大器（PGA）设置的滑块控制。在其他设置下，PGA 滑块无法使用。

Using SDR++

The main screen



Top Bar



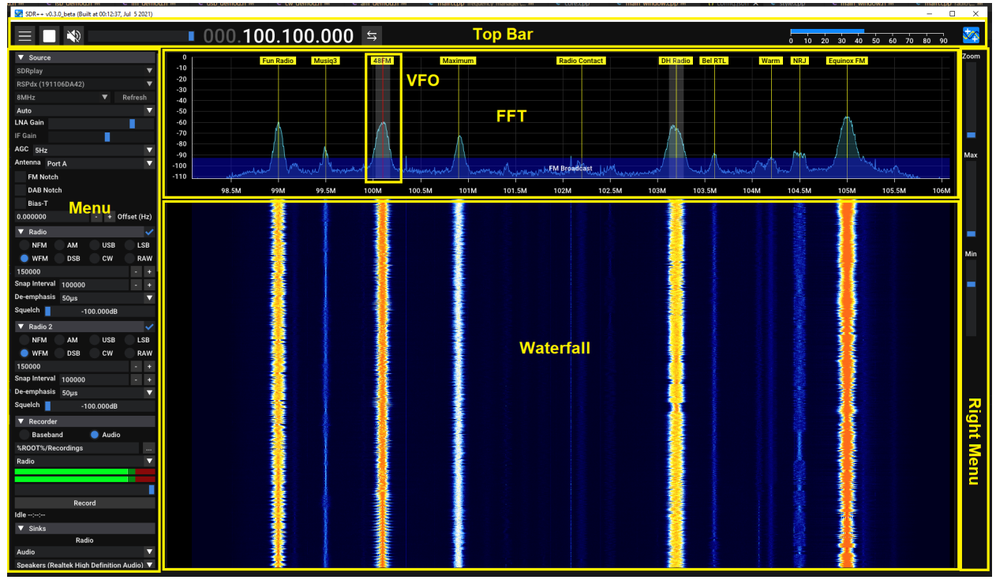
The top bar contains the most important controls.

Menu button and menu

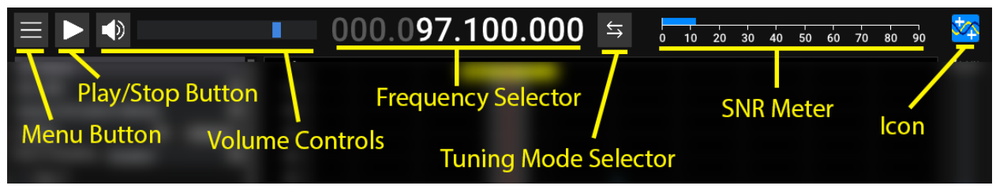
The menu button reveals the side menu to show all the modules and allow you to make changes, or hides it so you have more screen space for the spectrum display and waterfall.

使用SDR++

主屏幕



顶部栏



顶部栏包含最重要的控件。

菜单按钮和菜单

菜单按钮可以显示侧边菜单以查看所有模块并进行更改，或者隐藏侧边菜单，以便为频谱显示和瀑布图腾出更多屏幕空间。

The menu shows all the different settings and modules in SDR++. If you want the modules in a different order you can drag and drop them into a different place in the menu.

Start button

Play button - use this to start the receiver.

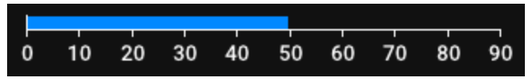
When started, a white square stop button replaces it. Press this to stop receiving.

Volume control

Drag the cursor to adjust the volume level or click on the bar at the level you want. You can click on the speaker symbol to mute and un-mute the audio. If you have multiple VFOs this will adjust the volume for the selected VFO. Otherwise you can adjust the volume for each VFO independently using the control for it's 'sink'.

SNR Meter

The signal meter provides a measure of the relative strength of signals above the background noise - the 'signal to noise ratio' or 'SNR'. It is not a calibrated S-meter like you would find on an amateur transceiver or professional receivers, as it is designed to work with a wide range of SDRs, with varying settings.



Frequency Selector and tuning

The frequency selector or display shows the frequency of the received signal in hertz. The decimal points distinguish between (from left to right):

菜单显示了 SDR++ 中的所有不同设置和模块。如果你想将模块按不同顺序排列，可以将它们拖放到菜单中的其他位置。

开始按钮

播放按钮 - 使用此按钮启动接收器。

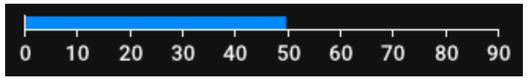
当 s　　一旦开始，一个白色方形停止按钮会取代它。按下此按钮可停止接收。

音量控制

拖动光标以调整音量级别，或点击您想要的音量位置的进度条。您可以点击扬声器图标来静音或取消静音。如果您有多个VFO，这将调整所选VFO的音量。否则，您可以使用其“接收端”的控制单独调整每个VFO的音量。

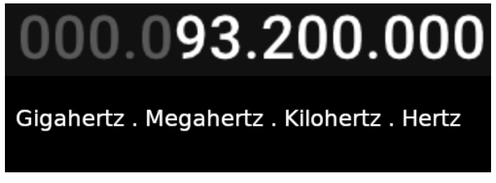
信噪比计

信号表提供了信号相对于背景噪声的相对强度度量——即“信噪比”或“SNR”。它并不是像业余无线电收发机或专业接收机上的校准S表，因为它设计用于与各种不同设置的SDR一起使用。



频率选择器与调谐

频率选择器或显示屏显示接收信号的频率（以赫兹为单位）。小数点用于区分（从左到右）：



So, in the illustration above, it shows 93.2 MHz or 93,200 kHz or 93,200,000 Hertz.

The frequency being received by SDR++ is shown by the VFO band on the FFT spectrum display and waterfall. You can have multiple VFOs.

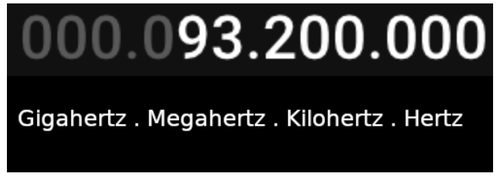
In some circumstances you might have no VFOs running - for instance if you have unticked the box for the radio VFO. In these circumstances the centre frequency of the SDR will be shown.

You can tune your receiver in several different ways:

Using the frequency selector

- Place your mouse's cursor over the figure in the frequency display that you want to change and use the **scroll wheel** to move it up or down
- Click (or tap if you have a touch screen) on the upper part of the figure to increase it by 1, or on the lower part to decrease it by 1
- To input a specific frequency hover your mouse or tap on the largest digit you need to change then type in the whole frequency. The cursor will move across to each digit you change as you do this. So, to change 93.2 to 101.5 in the picture above you would place the mouse over the '0' before the '9' and type '1015'.
- Hover the mouse cursor over the spectrum display or waterfall and use the scroll wheel. One 'notch' of the wheel will increase or decrease the frequency by the amount set as the 'Snap interval' in the radio menu.

Other ways of tuning include:



所以，在那上图显示为 93.2 兆赫、93,200 千赫或 93,200,000 赫兹。

SDR++ 接收到的频率显示在FFT频谱显示和瀑布图的VFO频段上。你可以使用多个VFO。

在某些情况下，你可能没有任何VFO在运行——例如，如果你取消了无线电VFO的复选框。在这些情况下，将显示SDR的中心频率。

您可以通过几种不同方式调整接收器：

使用频率选择器

● 将鼠标光标移动到频率显示中要更改的数字上，然后使用滚轮向上或向下移动它

● 点击（如果使用触摸屏则轻触）图形的上部将其增加1，点击下部将其减少1

● 要输入特定频率，将鼠标悬停或点击你需要更改的最大数字，然后输入整个频率。光标会随着你更改的每个数字移动。因此，要将上图中的 93.2 改为 101.5，你需要将鼠标放在‘9’前面的‘0’上，然后输入‘1015’。

● 将鼠标光标悬停在频谱显示或瀑布图上，并使用滚轮。滚轮的每一次‘刻度’会按无线电菜单中设置的“捕捉间隔”增加或减少频率。

调优的其他方法包括：

- Click on the FFT spectrum display or waterfall on the frequency you want. Usually this will be when it shows a signal of interest - more about these below.
- Drag the frequency scale at the bottom of the FFT spectrum display.
- Drag the VFO band to the frequency you want.

You will also have to set the correct modulation, for instance ‘AM’, in the Radio module (see below). The Frequency Manager module can also tune the receiver directly to bookmarked frequencies you have saved.

Tuning Mode

Next to the frequency display is a button that changes the behaviour of the VFO band. The button toggles between:

‘Centre tuning mode’ - if you click or tap on the waterfall or spectrum to tune then the selected frequency ‘snaps’ to the centre of the display, and the spectrum width is centred on this frequency

‘Normal tuning’ mode - you can click or tap to tune anywhere on the spectrum display and waterfall and they stay where they are. You will then see a continuous line on the waterfall for the chosen signal and other signals that you might be monitoring.

Icon

The icon shows the credits and build version of the software when clicked.

- 点击 FFT 频谱显示或瀑布图上你想要的频率。通常是在显示感兴趣信号的时候——下面会详细说明这些信号。
- 拖动 FFT 频谱显示底部的频率刻度。
- 将VFO频段拖动到您想要的频率。

您还需要在无线电模块中设置正确的调制方式，例如“AM”（见下文）。频率管理模块也可以直接将接收器调至您保存的书签频率。

调谐模式

在频率显示旁边有一个按钮，用于改变VFO频段的行为。该按钮可以在以下模式之间切换：

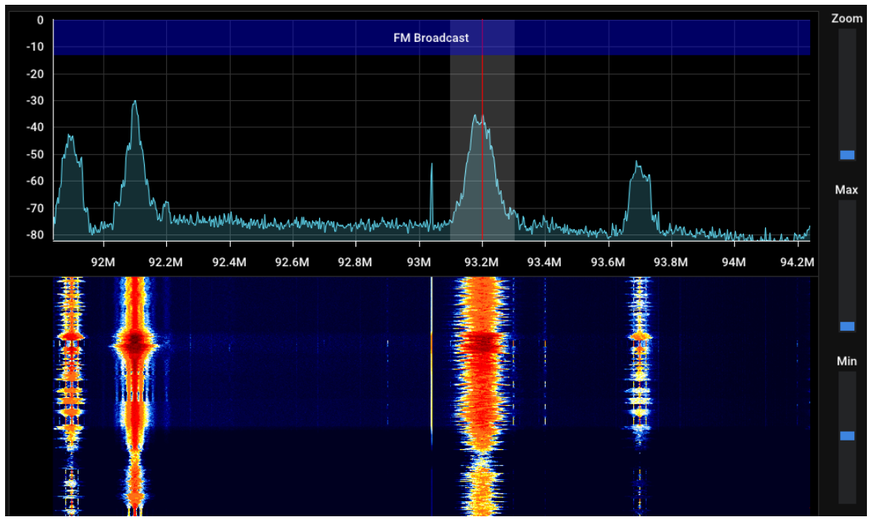
‘中心调谐模式’——如果您点击或轻触瀑布图或频谱进行调谐，所选频率会‘吸附’到显示屏的中心，并且频谱宽度会以该频率为中心

“正常调谐”模式——您可以点击或轻触频谱显示和瀑布图上的任意位置进行调谐，它们会保持在原来的位置。然后，您将在瀑布图上看到所选信号以及您可能正在监控的其他信号的连续线。

Icon

点击该图标会显示软件的版权信息和构建版本。

Spectrum display (FFT) and waterfall



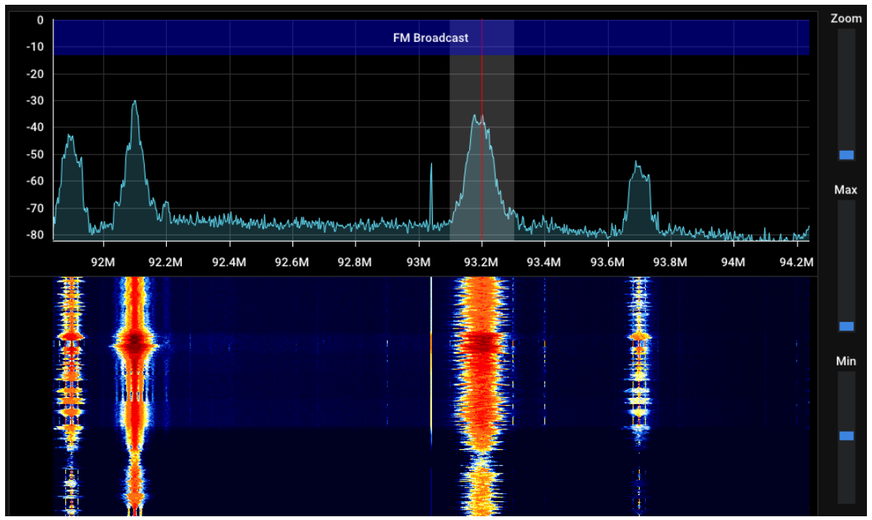
Using a spectrum display and waterfall is the best way that a modern SDR receiver can show a chunk of the radio spectrum. You can see several transmissions at once and select which one to investigate.

You can see the strength of each signal through the depth of the color and the width of the transmission. You can see some of the characteristics of the transmission such as the sidebands and even recognise visually whether a signal is CW, upper or lower sideband or data.

The FFT or spectrum display

This is at the top. When receiving, it will show a part of the radio frequency spectrum. It can show the width of the radio frequency spectrum that your SDR receiver is set to receive. For instance, on an SDRplay SDR this can be up to 10 MHz. This can usually be set in the ‘source’ module.

频谱显示（FFT）和瀑布图



使用频谱显示和瀑布图是现代软件定义无线电接收机展示一段无线电频谱的最佳方式。你可以同时看到多个传输信号，并选择其中一个进行分析。

你可以通过颜色的深浅和传输的宽度看到每个信号的强度。你还可以看到传输的一些特征，比如边带，甚至可以通过视觉识别信号是连续波(CW)、上边带、下边带还是数据。

FFT 或频谱显示

这是在最上方。接收时，它将显示一部分射频频谱。它可以显示 SDR 接收器设置接收的射频频谱宽度。例如，在 SDRplay SDR 上，这可以达到 10 MHz。通常可以在“源”模块中设置。

The vertical axis shows the strength of the signal - or noise[1].

The horizontal axis shows the frequency.

Signals - or indeed some man made or natural noise and interference - will create peaks because they are louder or stronger than the base noise level.

You can click or tap on the peaks to tune to that frequency and hear what the signals are.

As already mentioned, further ways of tuning the receiver include

- dragging the frequency scale at the base of the spectrum display so that the cursor is over the desired frequency.
- dragging the VFO band to the desired frequency

The waterfall

This shows the same information as the FFT/spectrum display but over time - hence it moves. You can see this on a CW (morse) signal where the peaks of the dots and dashes will create a track on the waterfall, as though somebody is writing them out on paper but vertically.

The waterfall helps you identify signals of interest because it will catch intermittent transmissions like 2-way voice traffic. These show up as a track of interrupted lines on the waterfall. You can click on the line to tune to it and hear the transmission.

Using the FFT/spectrum display and waterfall together

You can adjust the size of each by dragging the dividing line up or down to make one bigger and the other smaller.

Zoom

This sliding control on the right menu allows you to zoom in and out. It narrows or widens the frequency range that the spectrum and waterfall covers. This means the peak and

[1] this is in dBFS and not dBm and is not calibrated. It is just relative to the maximum level of the digitization

纵轴显示信号强度或噪声。

横轴显示频率。

信号——或者确实是某些人为或自然的噪声和干扰——会产生峰值，因为它们比基础噪声水平更响或更强。

你可以点击　　点击峰值以调到该频率，听信号内容　　.

如前所述，调整接收器的其他方法包括

- 拖动频谱显示底部的频率刻度，使光标位于所需的频率上。
- 将 VFO 频段拖动到所需频率

瀑布

这显示了与 FFT/频谱显示相同的信息，但随时间变化——因此它会移动。你可以在连续波（莫尔斯电码）信号上看到这一点，点和划的峰值会在瀑布图上形成一条轨迹，就像有人在纸上垂直书写它们一样。

瀑布图可以帮助你识别感兴趣的信号，因为它能够捕捉间歇性的传输，例如双向语音流量。这些信号会显示为瀑布图上断续的线条。你可以点击这些线条来调频收听传输内容。

同时使用FFT/频谱显示和瀑布图

你可以通过向上或向下拖动分隔线来调整每个的大小，使一个变大，另一个变小。

Zoom

右侧菜单上的这个滑动控制可以让你放大或缩小视图。它可以缩小或扩大频谱和瀑布图所覆盖的频率范围。这意味着峰值和

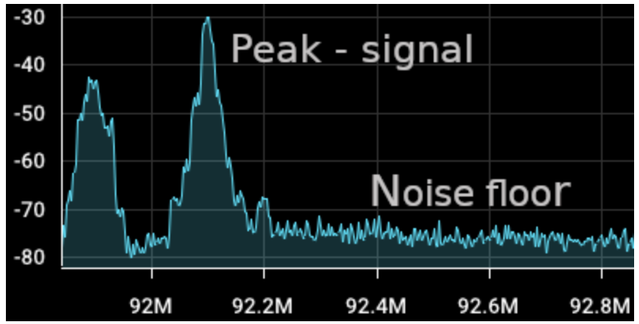
[1] 这是以 dBFS 为单位，而不是 dBm，且没有经过校准。它只是相对于数字化的最大电平的相对值。

track of the signal is bigger. You'll be able to see more details of the sidebands and click more accurately on the signal so it is 'in tune'

Min and Max

These sliding controls select the high and low points for the signal strength shown on the spectrum - effectively the top and bottom range.

Start by adjusting the 'min' slider so that the base noise floor - the valleys and plains below any peaks - is brought down to the bottom of the spectrum display like this:



By doing this you'll see that the waterfall background - except where there are signals - changes from an intense color to a more or less uniform dull or black area. This means that you can better see the brighter tracks from weak signals.

The 'max' slider will be at 0dB when fully down. You can move it upwards if you need to show weaker signals on the waterfall.

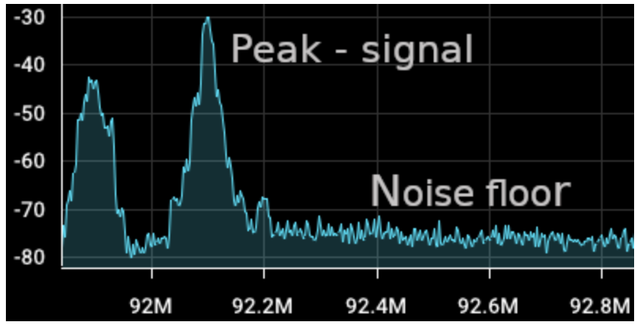
Play with these controls until you have the right contrast on the waterfall that works for you. You may need to adjust these when you change frequency because the noise floor and signal strengths will be different.

信号的轨迹更大。你将能够看到更多边带的细节，并且可以更准确地点击信号，使其“调谐”正确

最小值和最大值

这些滑动控制用于选择频谱上显示的信号强度的高点和低点——实际上就是顶部和底部范围。

首先调整“最小值”滑块，使基底噪声地板——任何峰值下方的低谷和平原——下降到频谱显示的底部，如下所示：



通过这样做，你会发现瀑布图的背景——除了有信号的地方——会从强烈的颜色变为或多或少均匀的暗色或黑色区域。这意味着你可以更清楚地看到弱信号发出的较亮痕迹。

当“最大”滑块完全下调时，将处于 0dB。 如果您需要在瀑布图上显示较弱的信号，可以将其向上移动。

调整这些控制，直到你找到适合你的瀑布图的正确对比度。当你改变频率时，可能需要再次调整这些设置，因为噪声底和信号强度会有所不同。

Tuning using the FFT/spectrum and waterfall

You can tune to frequencies shown on the spectrum and waterfall by pointing and clicking. This will set the VFO to that frequency and the VFO band will be shown. You can tune up and down by using the scroll wheel on your mouse, if it is hovering over either the FFT/spectrum display or waterfall. It will tune in increments set by the 'snap interval' set in the Radio module, for instance 100kHz for FM broadcast or 100Hz when in single sideband (LSB or USB).

The VFO (variable frequency oscillator) tunes the SDR to a particular frequency. It's shown here as a band or line going straight through both the spectrum display and the waterfall. This is like the tuning needle or cursor you'd get on an old radio. The band/line is as wide as the filter width the VFO is set to. You can increase or decrease the filter width by dragging the edge of the VFO band outwards or inwards.

If you hover your mouse pointer over the VFO band it will show 'radio' or whatever name you have given to that VFO. If you do the same with the ctrl key pressed it will show the frequency, bandwidth, whether the bandwidth is locked and the signal to noise ratio (which will change dynamically).

Other uses for the spectrum and waterfall

If you have a manual **antenna tuner**, you can adjust it to maximise the signal strength. Technically it's matching the antenna impedance with that of the receiver to produce more signal into your SDR.

As you adjust the inductance and capacitance to tune the antenna you'll see the noise floor rise and sweep across the spectrum. You can adjust one control (usually capacitance) and then the other to get the peak at the frequency you want to hear.

You can create **bookmarks** at particular frequencies using the **Frequency Manager module**. These will display on the spectrum display. The **Bandplan** module will display the allocation of blocks of frequencies on the spectrum. More about these below.

使用 FFT/频谱和瀑布图进行调谐

您可以通过指向并点击频谱和瀑布图上显示的频率来进行调谐。这将把VFO设置为该频率，并显示VFO频段。如果鼠标悬停在FFT/频谱显示或瀑布图上，您也可以使用滚轮向上或向下调谐。调谐的增量由无线电模块中设置的“锁定间隔”决定，例如FM广播时为100kHz，单边带（LSB或USB）模式下为100Hz。

VFO（可变频振荡器）将SDR调谐到特定频率。在这里，它显示为一条横穿频谱显示和瀑布图的带状或线条。这就像老式收音机上的调谐指针或光标。该带状/线条的宽度与VFO设置的滤波器宽度相同。你可以通过将VFO带的边缘向外或向内拖动来增大或减小滤波器宽度。

如果将鼠标指针悬停在VFO频段上，它会显示“radio”或您为该VFO设置的任何名称。如果在按住Ctrl键的同时执行相同操作，它将显示频率、带宽、带宽是否锁定以及信噪比（会动态变化）。

频谱和瀑布图的其他用途

如果你有一个手动天线调谐器，你可以调整它以最大化信号强度。从技术上讲，这就是将天线阻抗与接收器阻抗匹配，以便向你的SDR输入更多信号。

当你调整电感和电容来调谐天线时，你会看到噪声底升高并在频谱上扫描。你可以先调整一个控制（通常是电容），然后再调整另一个，以在你想听的频率上达到峰值。

您可以使用频率管理器模块在特定频率创建书签。这些书签将在频谱显示上显示。频段规划模块将显示频谱上频率块的分配情况。更多内容如下所示。

Keyboard controls

F11 Toggle fullscreen.

Home Toggle waterfall.

End Toggles between play and stop. In Linux this may close SDR++.

< > Left Arrow / Right arrow - Tunes SDR++ up or down in frequency by selected step ('snap interval'). If you have your mouse over the frequency selector it will instead move between the digits of the frequency, which you can then adjust with the up/down arrows.

tab Moves down the menu between variable slider controls, so that you can then alter them instead by directly entering the desired figure in the box. For instance, it might move between the source gain control to the Radio VFO bandwidth and with another press to the snap interval below it. Type the figure you want to use e.g. 2400 followed by either enter (which will us the chosen figure) or another press on tab to move you to the next box. shift + tab will instead move you up the menu between boxes.

page up and page down keys - move from on VFO to another, if multiple VFOs are enabled.

You can find a full list of user interface key controls on the SDR++ github pages.

SDR++ command line related parameters

You can use these in the command line interface e.g. in Windows after sdrpp.exe.

Short	Verbose	Description
-a	--addr	In server mode this specifies the IP address of the network interface accepting connections
	--autostart	Autostart the SDR when SDR++ starts up. This may be useful for those using SDR++ as a permanent display. It will only work for sources that do not require connection prior to starting. Only available from later version 1.06 nightly builds.
-c	--con	Show console (-s on earlier versions)
-h	--help	Show help. Only available from version 1.06 or nightly builds.

键盘控制

F11 切换全屏模式。

首页 切换瀑布流。

End 键在播放和停止之间切换。在 Linux 系统中，这可能会关闭 SDR++。

< > 左箭头 / 右箭头 - 调整 SDR 的频率++ 按选定步长（“捕捉间隔”）向上或向下。如果鼠标悬停在频率选择器上，它将改为在频率的各位数字之间移动，然后您可以使用上下箭头进行调整。

Tab 键在菜单的变量滑块控制之间向下移动，这样您就可以通过直接在框中输入所需的数值来更改它们。例如，它可以在源增益控制和无线电 VFO 带宽之间切换，再按一次可以移动到下面的捕捉间隔。输入您想使用的数值，例如 2400，然后按 Enter（将使用所选数值），或者再次按 Tab 键以移动到下一个框。Shift+ + Tab 将向上在菜单的框之间移动。

向上翻页键和向下翻页键 - 在多个 VFO 启用时，从一个 VFO 移动到另一个 VFO。

您可以在 SDR++ 的 GitHub 页面上找到完整的用户界面键控列表。

SDR++ 命令行相关参数

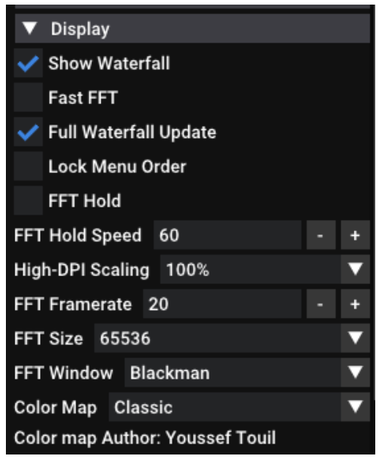
你可以 在命令行界面中使用这些，例如在 Windows 中运行 sdrpp.exe 后。

短	冗长的	描述
-a	--地址	在服务器模式下，这指定了网络接口的IP地址 接受连接
	--自动启动	在SDR++ 启动时自动启动SDR。这可能会很有用，对于 那些将SDR++ 作为永久显示使用的人。它只适用于 无需在启动前连接的来源。仅 可从稍后的 1.06 夜间构建版本获得。
-c	--con	显示控制台（早期版本使用 -s）
-h	--help	显示帮助。仅适用于版本 1.06 或测试版。

Short	Verbose	Description
`-p`	`--port`	Server mode port. Insert port address after parameter.
`-r`	`--root`	Root directory where all files are saved. Insert path after the parameter.
`-s`	`--server`	Run in server mode. More details on using these commands with SDR++ Server.

Display menu

This controls the configuration of the waterfall. 'FFT' stands for 'fast fourier transform'. FFT is the algorithm used by the computer to convert the data from the signals in the SDR into visual representations - the spectrum display and waterfall.



Show waterfall - switches the waterfall on or off

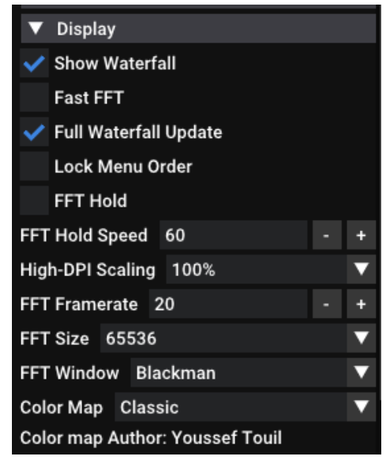
Fast FFT - using this will mean that a faster but less precise rendering will be used. This will show less detail on the waterfall and spectrum display. [Removed from release version 1.1.0 onwards].

Full waterfall update - when enabled, the history of the waterfall is updated when viewing parameters like zoom and min/max are changed.

短	冗长的	描述
`-p`	`--端口`	服务器模式端口。在参数后插入端口地址。
`-r`	`--根`	保存所有文件的根目录。在之后插入路径 参数。
`-s`	`--服务器`	以服务器模式运行。有关使用这些命令的更多详细信息 与 SDR++ 服务器。

显示菜单

这控制瀑布图的配置。‘FFT’代表‘快速傅里叶变换’。FFT 是计算机用来将 SDR 中的信号数据转换为可视化表示的算法——频谱显示和瀑布图。

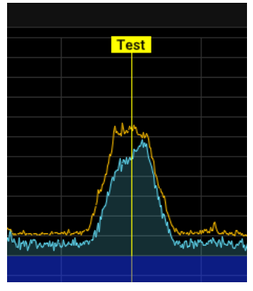


显示瀑布 - 开启或关闭瀑布

快速 FFT - 使用此选项意味着将使用更快速但精度较低的渲染。这将在瀑布和频谱显示中显示较少的细节。[从 1.1.0 版本开始已从发行版中移除]。

完整瀑布图更新 - 启用时，当查看诸如缩放和最小/最大值等参数发生变化时，瀑布图的历史记录也会更新。

FFT Hold - this provides a trace across the spectrum display showing the maximum signal level that has been reached.



You can adjust the amount of time for which the trace is displayed using **FFT Hold Speed**. A larger value means the trace reduces quicker.

Lock menu order - enabling this prevents you from moving the menu section. It is enabled by default in Android to prevent accidental moves whilst dragging scroll bars. If you try to move a menu section whilst this is set, an error message will be displayed

High-DPI scaling - this allows you to adjust the size of the controls, borders, scroll bars and text in SDR++. You might use it if you have a very high resolution monitor, or an Android device like a phone where you need these to be bigger so they can be seen, or to make the controls large enough to be easily moved by a finger or stylus on a touch screen. You'll need to close and then re-open SDR++ for the settings to take effect.

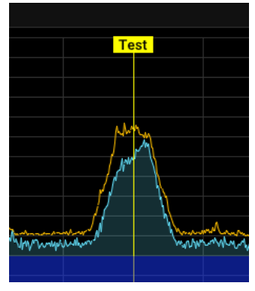
FFT Frame rate - SDR++ runs an algorithm to generate the FFT/spectrum. When this is run more times each second it produces a more detailed FFT/Spectrum display, but at the expense of frequency resolution. As a result the waterfall descends faster. Higher is quicker. The default is 20.

FFT size - This increases or decreases resolution - the number of signal data points that are shown on the spectrum display and waterfall. Higher is better, and will show weaker signal traces, but will take more processing power. A slower CPU may not handle the higher rates and produce some clipping or stuttering in the audio output.

FFT Window - This setting determines the algorithms applied to data before being run through the FFT, and will affect the look of display signal peaks on the spectrum display and waterfall. You have 2 choices - 'Rectangular' or 'Blackman' and, from version 1.1.0, Nuttall. Nuttall is the new default.

Color Map - You can select one of a number of 'maps' that change the color and intensity shown on the waterfall for the range of signals from weak to strong.

FFT保持——这在频谱显示中提供了一条轨迹，显示已达到的最大信号水平。



您可以使用 FFT 保持速度调整显示迹线的时间。数值越大，迹线消退得越快。

锁定菜单顺序 - 启用此功能可以防止移动菜单部分。在 Android 中默认启用此功能，以防止在拖动滚动条时意外移动。如果在此设置启用的情况下尝试移动菜单部分，将显示错误信息

高DPI缩放——这允许您调整SDR++中控件、边框、滚动条和文本的大小。如果您使用的是高分辨率显示器，或者像手机这样的安卓设备，需要将这些元素放大以便看清，或者为了在触摸屏上用手指或手写笔轻松操作控件，您可能会使用此功能。您需要关闭然后重新打开SDR++ ，设置才能生效。

FFT 帧率 - SDR++ 运行一个算法来生成 FFT/频谱。当每秒运行次数增加时，它会生成更详细的 FFT/ 频谱显示，但会牺牲频率分辨率。因此，瀑布图下降得更快。数值越高速度越快。默认值是 20。

FFT 大小 - 这会增加或减少分辨率——即在频谱显示和瀑布图上显示的信号数据点数量。数值越高越好，可以显示更弱的信号痕迹，但会占用更多的处理能力。较慢的 CPU 可能无法处理较高的速率，可能会在音频输出中产生一些削波或卡顿。

FFT 窗口 - 此设置决定在数据通过 FFT 处理前应用的算法，并会影响频谱显示和瀑布图中信号峰的表现。您有两种选择 —— “矩形（Rectangular）” 或 “布莱克曼（Blackman）”，从版本 1.1.0 起，增加了 Nuttall。Nuttall 是新的默认选项。

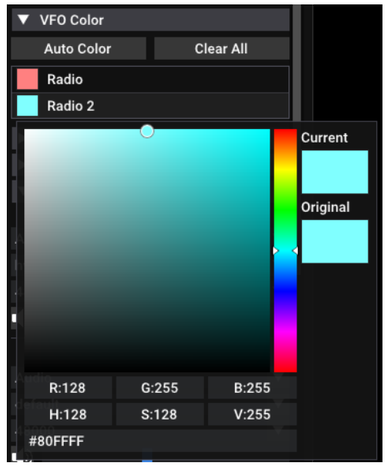
颜色映射 - 您可以选择多种“映射”之一，这些映射会根据信号从弱到强的范围改变瀑布图显示的颜色和强度。

This allows you to change the color of the VFO ‘band’. The VFO band shows the frequency to which the SDR is tuned across the spectrum display and waterfall, and is as wide as the filter width to which it is set.

A color can be chosen which best contrasts with the background.

If you have set up multiple VFOs, you can set the colors of each to help you identify them.

VFO Color



This simple control allows you to select a color to use for the VFO band. This is particularly useful if you have multiple VFOs.

Click on the colored square next to the VFO that you want to change, then either select the color from the tool or specify the color in hex code.

You can reset the colors by selecting ‘Clear All’ or ‘Auto Color’ to have SDR++ allocate colors automatically.

Bandplans

The allocation of bands of radio frequencies, particularly at VHF and UHF, varies between different countries and regions. For instance the allocation of frequencies on the 60 meter amateur band varies, as does the existence of a 4 metre (70 MHz) amateur band.

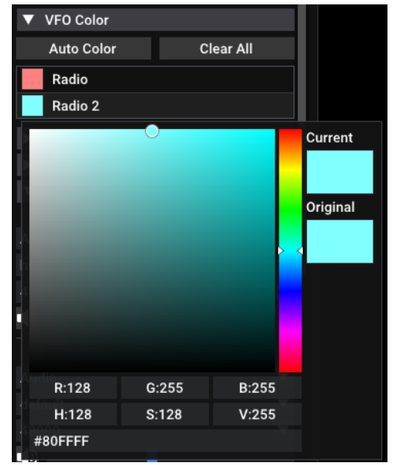
SDR++ contains a number of bandplans for different geographical areas e.g. UK or USA. At a minimum they will show the blocks of frequencies used by radio amateurs and broadcasters. Some, such as the UK band plan, will also cover HF air and marine allocations, and major usage on VHF and UHF, such as private mobile radio.

这允许您更改VFO“频段”的颜色。VFO频段显示SDR在频谱显示和瀑布图中调谐的频率，并且宽度与所设置的滤波器宽度相同。

可以选择一种与背景形成最佳对比的颜色。

如果您已经设置了多个VFO，您可以为每个VFO设置颜色，以帮助您识别它们。

VFO 颜色



这个简单的控制允许您选择用于 VFO 频段的颜色。如果您有多个 VFO，这特别有用。

点击您想要更改的 VFO 旁边的彩色方块，然后从工具中选择颜色或以十六进制代码指定颜色。

您可以通过选择“全部清除”或“自动颜色”来重置颜色，以便 SDR++ 自动分配颜色。

频段计划

不同国家和地区对无线电频段的分配，尤其是甚高频（VHF）和超高频（UHF），有所不同。例如，60米业余无线电频段的频率分配各不相同，4米（70 MHz）业余无线电频段的存在情况也各有差异。

SDR++ 包含适用于不同地理区域的多个频段计划，例如英国或美国。至少，它们会显示业余无线电和广播频率的分配块。一些频段计划，例如英国的频段计划，还会涵盖HF空中和海上分配，以及VHF和UHF的主要使用情况，例如私人移动无线电。

By default SDR++ will open with the 'Worldwide bandplan'. You can change it from the drop down list to another available bandplan.

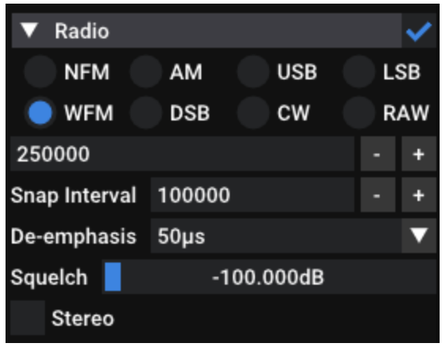
You can choose whether the bandplan is displayed, and whether it is shown at the top or bottom of the Spectrum monitor.

You can alter the bandplan by editing the appropriate file .json file [in the /res subfolder] in a text editor. You should arrange the entries in frequency order and follow the same format. There are preset 'types' of allocations, such as "amateur" that will each be shown in the same color. If you want to create a new type or alter the colors used, then you will need to edit the entries under "bandColors" in the main `config.json` file in the root folder.

Themes

There are a number of themes which you can select to choose the overall look of SDR++. It will change the colors of the background, spectrum and waterfall, text and menu boxes.

Radio module



←Filter width

This is the main module for setting the type of 'modulation' you can receive.

You can remove the tick in the box next to 'Radio' to disable this module. SDR++ will continue to show signals in the spectrum display and waterfall but there will be no sound sent to the speakers or other outputs.

默认情况下，SDR++ 将以“全球频段计划”打开。您可以从下拉列表中更改为其他可用的频段计划。

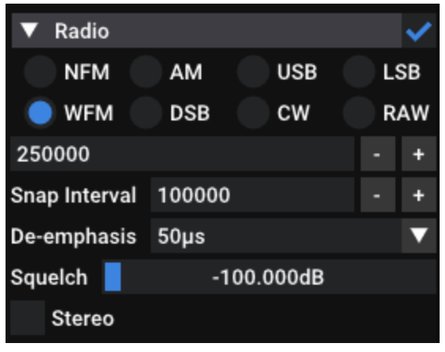
您可以选择是否显示频段图，以及是否显示在频谱监视器的顶部或底部。

您可以通过在文本编辑器中编辑位于 /res 子文件夹中的相应 .json 文件 [] 来更改频段计划。您应按频率顺序排列条目，并遵循相同的格式。系统有预设的分配“类型”，例如“业余”，每种类型都将显示为相同的颜色。如果您想创建新的类型或更改使用的颜色，则需要编辑位于根文件夹 main config.json 文件中的“bandColors”条目。

主题

您可以选择多个主题来决定 SDR++ 的整体外观。它会更改背景、频谱和瀑布图、文本以及菜单框的颜色。

无线电模块



←滤波器宽度

这是用于设置您可以接收的“调制”类型的主要模块。

您可以取消选中“Radio”旁边的复选框以禁用此模块。SDR++ 仍将继续在频谱显示和瀑布图中显示信号，但不会将声音发送到扬声器或其他输出设备。

Modes

SDR++ can receive all the main modes of modulation applied to signals. You can select these by clicking on the button for the mode.

Mode	Name	Use	Default Filter
NFM	Narrow band Frequency Modulation	From 27 MHz upward for Citizens Band, most business radio and on amateur radio bands. Voice, and V/UHF data transmissions such as DMR, NOAA APT and POCSAG/Flex are modulated in NFM	12.5 khz
WFM	Wideband Frequency Modulation	Broadcast band 87-108 MHz. You can also choose the de-emphasis used, and switch stereo reception on and off.	150 kHz
AM	Amplitude Modulation	Long, medium and short wave broadcast stations, Citizens Band, Airband.	10 kHz
DSB	Double Sideband	Used for dual sideband suppressed AM carrier transmissions.	4.6 kHz
USB	Upper sideband	Amateur bands above 10MHz and on 60m for voice, and for most data modes on all bands; most HF utility stations (air, marine, military)	2.8 kHz
LSB	Lower sideband	Mostly used for voice transmissions on Amateur bands below 10 MHz and for rtty (radioteletype)	2.8 kHz
CW	Continuous Wave	Morse code transmissions. Basically LSB with a narrow filter	200 Hz
RAW		This outputs the IF of the VFO as audio. Left channel is I branch, right is Q branch	

模式

SDR++ 可以接收所有应用于信号的主要调制模式。您可以通过点击模式按钮来选择这些模式。

Mode	Name	Use	默认筛选器
NFM	窄带 频率 调制	从 27 MHz 起用于市民 频段，大多数商业无线电及开启 业余无线电频段。语音，和 甚高/超高频数据传输，例如 DMR、NOAA APT 和 POCSAG/Flex 是 在NFM中调制	12.5 千赫
WFM	宽带频率 调制	广播频段 87-108 MHz。你可以 还可以选择使用的去强调方式， 并打开立体声接收 off.	150 千赫
AM	振幅 调制	长波、中波和短波 广播电台，市民频段 航空频段	10 千赫
DSB	双边带	用于双边带抑制 AM 载波传输。	4.6 千赫
USB	上边带	业余乐队在10MHz以上及 语音为60米，大多数数据也是如此 所有频段的模式；大多数短波实用功能 站点（航空站、海事站、军事站）	2.8 千赫
LSB	下边带	主要用于语音传输 在10 MHz以下的业余频段上以及 用于 RTTY（无线电电传打字）	2.8 千赫
CW	连续波	摩尔斯电码传输。基本上 带窄滤波器的下边带	200 赫兹
RAW		这会输出 VFO 的中频（IF）为 音频。左声道是 I 分支，右声道是 Q 部门	

Filter width

Different modes of transmission use different bandwidths. SDR++ has default filters to match the bandwidth (see table above), but you can alter these by typing in the value or clicking on the '-' and '+' symbols to narrow or widen the filter. You can also adjust this by dragging the edge of the VFO band - shown on the spectrum display - in or out. You might do this to make the audio more pleasant, cut out interference from signals that are close in frequency or to ensure all the information on a data signal is received.

Snap interval

This is used to set the frequency step (or 'increment') that the receiver will move when tuning.

For instance, for FM broadcast stations you'd want this set as 100 kHz. It would then move from, for instance 93.1 to 93.2, then to 93.3 MHz for each movement of the scroll wheel on your mouse (when the cursor is over the frequency display or waterfall).

The snap interval is retained by SDR++ for each mode. So, if you alter it to 2800Hz on USB all other USB signals will use this filter width until you alter it again.

See the section on the frequency display and tuning for more on tuning.

De-emphasis (NFM and WFM only)

On FM signals, audio frequencies - particularly in the high range - are boosted (emphasised) at the transmitter then reduced at the receiver (de-emphasis).

You can switch off the de-emphasis or adjust the amount applied. In Europe 50µs is used, but in the US it is 75µs.

Squelch

滤波器宽度

不同的传输模式使用不同的带宽。SDR++ 有默认的滤波器来匹配带宽（见上表），但你可以通过输入数值或点击‘-’和‘+’符号来缩小或扩大滤波器来更改它们。你也可以通过拖动 VFO 频段的边缘（显示在频谱图上）来进行调整。你可能这样做是为了使音频更悦耳、去除来自频率接近的信号的干扰，或确保接收到数据信号上的所有信息。

捕捉间隔

这用于设置接收器调谐时移动的频率步长（或“增量”）。

例如，对于调频广播电台，您希望将其设置为100 kHz。然后，每次滚动鼠标滚轮（当光标位于频率显示或频谱瀑布上时），频率会从例如93.1 MHz移动到93.2 MHz，再移动到93.3 MHz。

每种模式的快照间隔由 SDR++ 保留。因此，如果您将 USB 的快照间隔更改为 2800Hz，所有其他 USB 信号将使用此滤波宽度，直到您再次更改它。

有关调谐的更多信息，请参阅有关频率显示和调谐的部分。

减弱调制（仅限NFM和WFM）

在调频（FM）信号中，音频频率——特别是高频范围——在发射端会被增强（强调），然后在接收端被降低（去强调）。

您可以关闭去重音功能或调整应用的量。在欧洲使用50µs，而在美国则为75µs。

压制

The squelch cuts out the audio unless a signal exceeds the chosen level. It's useful for muting the background noise whilst allowing you to hear the wanted signal when it goes above the noise level.

You can set this by ticking the box to activate it, then dragging the marker to the right until the audio mutes. Check that the audio breaks through when a wanted transmission occurs. Adjust if necessary.

You will need to adjust the squelch level for different frequency ranges because noise levels will vary.

AGC Attack and Decay (not NFM or WFM)

This allows you to set the parameters for automatic gain control. This is useful for when the level of the signal - or the volume of the modulated signal - changes rapidly. You can set how quickly it reacts to this.

- 'Attack' governs how quickly the level is changed to compensate for increased signal strength
- 'Decay' governs how fast the level changes to a reduced signal strength. This is usually much slower than attack to prevent noise levels being boosted during short gaps, for instance when someone pauses between sentences.

The AGC levels you set are kept for each modulation type. For instance, if you set the levels for USB, then set them again for CW, the levels for USB will not be changed, and will be there when you return to using USB..

Carrier AGC

This applies the automatic gain control adjustments according to the level of the AM carrier rather than the demodulated audio level. This may produce quieter, more pleasant and consistent audio. It doesn't work as effectively where you have a signal that is fading in and out.

IF Noise Reduction (NFM and WFM only)

静噪功能会切断音频，除非信号超过所选的电平。它对于在背景噪音下静音非常有用，同时当所需信号超过噪音水平时可以听到它。

您可以通过勾选框来激活它，然后将标记拖到右侧直到音频静音来设置此项。确保在有需要的传输发生时音频能够传出。如有必要，请进行调整。

您需要根据不同的频率范围调整静噪电平，因为噪声水平会有所不同。

自动增益控制攻击和衰减（非 NFM 或 WFM）

这允许您设置自动增益控制的参数。当信号的电平或调制信号的音量快速变化时，这非常有用。您可以设置它对此反应的速度。

- “Attack” 控制为了补偿信号强度的增加，电平变化的速度
- “衰减”决定信号强度减少时电平变化的速度。通常它比“攻击”慢得多，以防在短暂间隙期间（例如有人在句子间停顿时）噪音水平被提高。

您设置的AGC（自动增益控制）水平会为每种调制类型保留。例如，如果您为USB设置了水平，然后再为CW设置一次，USB的水平不会改变，并且当您返回使用USB时仍然会保持原样。

载体AGC

这会根据调幅载波的电平而不是解调后的音频电平来应用自动增益控制调整。这可能会产生更安静、更悦耳且更稳定的音频效果。在信号时强时弱的情况下，它的效果不如理想。

中频噪声抑制（仅适用于窄频调制和宽频调制）

You can use IF noise reduction to make a received signal clearer and cut out background noise (‘QRN’) and some man-made interference (‘QRM’). This is available for wide (broadcast) and narrow frequency modulated signals only.

When you are in NFM mode, you have an additional setting for either

- NOAA APT weather satellite transmissions
- voice transmissions
- narrow band modes - also can be used on voice to reduce hiss.

‘IF’ noise reduction works at the ‘intermediate frequency’ stage of the SDR receiver.

Stereo (WFM only)

This toggles between stereo and mono where there is an FM stereo broadcast. Switching on stereo when you have a weak signal can add significant ‘hiss’.

Low Pass (NFM and WFM only)

This applies a filter to the audio signal to half the bandwidth of the VFO. In technical terms, it removes anything above a modulation index of 1. It can help remove high frequency hiss. You can keep this on, unless it interferes with the reception of a data signal where it’s not recommended.

RDS (WFM only)

SDR++ can decode digital ‘Radio Data System’ signals broadcast as part of an analog FM broadcast. This can include the station name and program information. ‘Alternate Frequency’ and ‘Traffic Program’ information is not supported.

The RDS data is displayed in the top left of the Spectrum Display (FFT), just under the frequency display.

您可以使用 IF 降噪功能使接收到的信号更清晰，并消除背景噪音（‘QRN’）以及一些人为干扰（‘QRM’）。此功能仅适用于宽带（广播）和窄带调频信号。

当你处于 NFM 模式时，你有一个额外的设置选项，用于

- NOAA APT 气象卫星传输
- 语音传输
- 窄带模式 - 也可用于语音以减少嘶嘶声。

“IF” 噪声抑制在 SDR 接收机的“中频”阶段工作。

立体声（仅限WFM）

在有调频立体声广播的情况下，这会在立体声和单声道之间切换。当信号较弱时开启立体声可能会产生明显的“噪声”。

低通（仅限 NFM 和 WFM）

这会对音频信号应用一个滤波器，将VFO的带宽减半。从技术上讲，它会去除调制指数大于1的所有内容。它可以帮助去除高频嘶嘶声。除非它干扰数据信号的接收，否则可以保持开启；在接收数据信号时不建议使用。

RDS（仅限WFM）

SDR++ 可以解码作为模拟 FM 广播一部分播放的数字“广播数据系统”信号。这可以包括电台名称和节目信息。不支持“备用频率”和“交通节目”信息。

RDS 数据显示在频谱显示（FFT）的左上方，就在频率显示的下方。

RDS is very sensitive to signal strength. You'll get better results with a strong signal, RDS will work in both mono and stereo modes.

Noise blanker (USB, LSB and DSB only)

The noise blanker works against 'impulse' type noise, such as a car ignition system or lightning.

Select this if you are suffering from this type of interference. Adjust the slider until the interference is reduced, but not so far that audio quality is degraded. This feature is still being worked on, so your results may vary.

Audio and Sinks

A 'sink' is a term for anything used to output sound from a demodulator or decoder. The raw data (or IQ signal) from the SDR receiver is taken by SDR++ and turned into sound that can be sent to your soundcard or over a network. You can then hear the signal through your speakers or it can be passed to another program (for instance. to decode a data transmission) via a virtual audio cable/loopback or to another computer by UDP/TCP over a network.

RDS 对信号强度非常敏感。信号强的时候效果会更好，RDS 在单声道和立体声模式下都能工作。

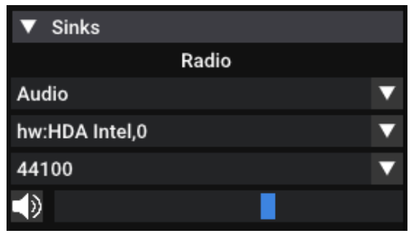
噪声抑制器（仅限USB、LSB和DSB）

噪声抑制器可以对抗“脉冲”类型的噪声，例如汽车点火系统或闪电产生的噪声。

如果您受到这种类型的干扰，请选择此项。调整滑块直到干扰减轻，但不要调得过头以至于音频质量下降。此功能仍在开发中，因此您的效果可能会有所不同。

音频和接收器

“接收端”是指用于从解调器或解码器输出声音的任何设备。来自SDR接收机的原始数据（或IQ信号）会被SDR++ 接收，并转换为可以发送到声卡或通过网络传输的声音。然后，您可以通过扬声器听到信号，或者通过虚拟音频线/回环传递给另一程序（例如，用于解码数据信号），或者通过网络通过UDP/TCP传送到另一台计算机。



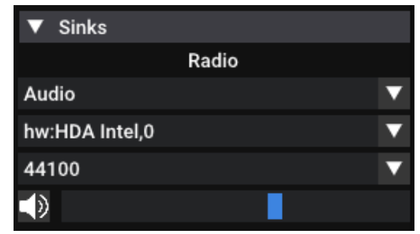
If you open the ‘Sinks’ module in the menu, you will see a ‘Radio’ sub-heading. This is the default name for the output from the VFO (or ‘tuner’) in SDR++. The Radio audio will generally be the only working ‘sink’ - or final output - for the SDR receiver when you first install it. You may later have more than one sink if you set up multiple VFOs, or if you add the M17 decoder module. You can adjust the sink to send the output to a network connection - and would select this from the drop down menu.

From here you can select the output for the audio. This would normally already be configured as your sound card (hw:HDA Intel,0 in the illustration above), but you may want to make a different choice if your soundcard has different outputs, for instance speakers or headphones.

Alternatively, if you have installed a virtual output such a VB-Audio Cable (in Windows), this will appear here. You can use a virtual audio cable to route your audio to a program running on your device (such as WSJT-X) to decode digital modes.

The bottom row of this module is the output level, and here duplicates the volume control on the top bar of SDR++.

You can create a sink for each additional VFO you use - see more about multiple VFOs



如果您在菜单中打开“接收器（Sinks）”模块，您会看到一个“收音机（Radio）”子标题。这是 SDR++ 中 VFO（或“调谐器”）输出的默认名称。收音机音频通常是 SDR 接收器首次安装时唯一可用的“接收器输出”或最终输出。如果您设置了多个 VFO，或者添加了 M17 解码模块，您以后可能会有多个接收器输出。您可以调整接收器输出以通过网络连接发送输出，并且可以从下拉菜单中进行选择。

在这里，您可以选择音频的输出。通常情况下，这已经配置为您的声卡（在上图中为 hw:HDA Intel,0），但如果您的声卡有不同的输出，例如扬声器或耳机，您可能想要选择其他选项。

或者，如果您安装了虚拟输出设备，例如 VB-Audio Cable（在 Windows 上），它将显示在此处。您可以使用虚拟音频电缆将音频传输到设备上运行的程序（例如 WSJT-X）以解码数字模式。

该模块的底部一行是输出音量，这里重复了 SDR++ 顶部栏的音量控制。

你可以创建为你使用的每个额外的VFO创建一个接收器 - 了解更多关于多个VFO的信息　s

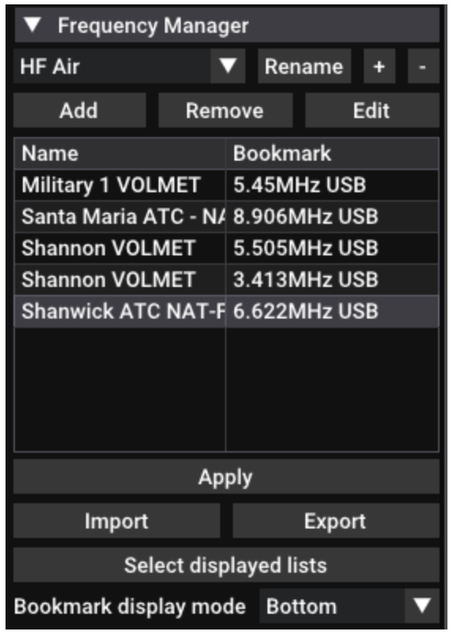
Frequency Manager

Frequency manager is a simple but versatile way of saving and displaying your favourite frequencies in ‘bookmarks’.

You can save them, together with the bandwidth and mode, and store them in different lists. Bookmarks cannot be saved with the same name, but adding an extra character or even one or more spaces to the bookmark name is enough to enable it to be saved.

Bookmarked frequencies can be shown in the spectrum display.

You can open Frequency Manager from the menu, then click on the arrow next to it’s name in the menu to reveal the full module.



Lists

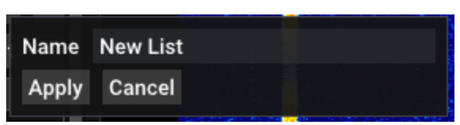
View and choose your lists

When you have created several lists just one of them will be displayed. Click on the down arrow next to that list’s name to display all your lists. Click on the one you wish to choose.

You can then click on any bookmark in that list to tune SDR++ to that frequency.

Create a list

SDR++ already has one ‘General List’ created. To create other lists, click on the plus (‘+’) symbol on the row next to the currently displayed list. This will open a box where you can type in the name of your list.



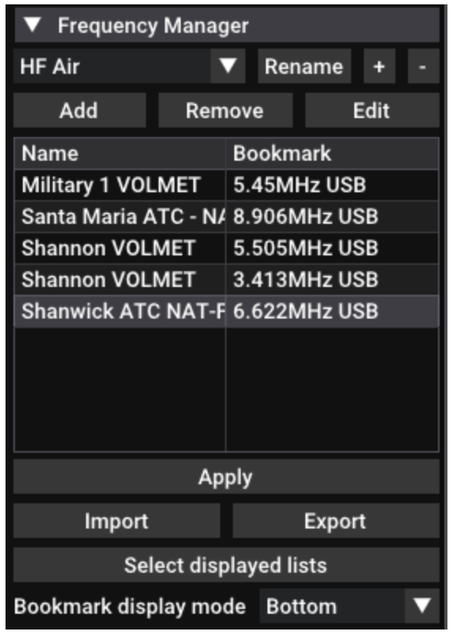
频率管理器

频率管理器是一种简单但多功能的方式，用于在“书签”中保存和显示您喜欢的频率。

你可以将它们与带宽和模式一起保存，并存储在不同的列表中。书签不能使用相同的名称保存，但在书签名称中添加一个额外的字符或一个或多个空格就足够使其能够保存。

书签频率可以在频谱显示中显示。

您可以从菜单中打开频率管理器，然后点击菜单中名称旁的箭头以显示完整模块。



列表

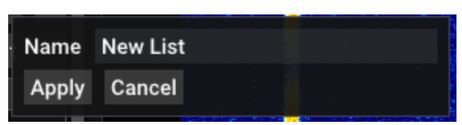
查看并选择您的列表

当你创建了多个列表时，只有其中一个会显示。点击该列表名称旁的下拉箭头以显示你所有的列表。然后点击你希望选择的列表。

然后你可以点击该列表中的任意书签，将SDR++ 调到该频率。

创建一个列表

SDR++ 已经创建了一个“通用列表”。要创建其他列表，请点击当前显示列表所在行旁边的加号（‘+’）符号。这将打开一个框，您可以在其中输入列表的名称。



You might want to create frequency lists for airband, marine, shortwave or medium wave broadcast, FT8, CB channels, local DMR, or for themes such as local mosques, rescue services and so on.

Rename a list

You can easily rename a list by clicking on the ‘Rename’ button.

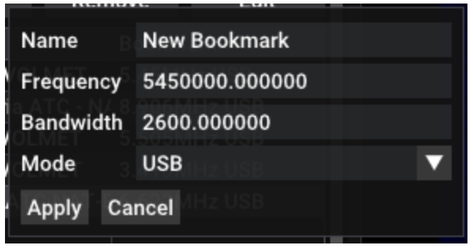
Remove a list

You can delete a list by clicking on the minus (‘-’) button. This will delete all the bookmarks in it.

Bookmarks

Create a bookmark

To save a frequency as a bookmark click on ‘Add’ This brings up a box where you can create a name for the bookmark. It will automatically pre-complete the bookmark with the frequency, bandwidth, and mode SDR++ is tuned to. However, you can edit these if you wish.



Click on ‘Apply’ to save.

The bookmarks will be shown in the list sorted in alphabetical order of the name. Bookmarks do not support unicode characters. Bookmarks should have a unique name, because duplicates are not permitted. If you do want to use the same name, then one or more spaces after the name will be enough for it not to be treated as a duplicate.

Tuning to the frequency of a bookmark

To do this you simply open the list to show the bookmark and double-click on it.

You can also click on a bookmark that is showing on the waterfall to tune to it.

您可能需要为空中波段、海事、短波或中波广播、FT8、CB 频道、本地 DMR，或者诸如本地清真寺、救援服务等主题创建频率列表。

重命名列表

您可以通过点击“重命名”按钮轻松重命名列表。

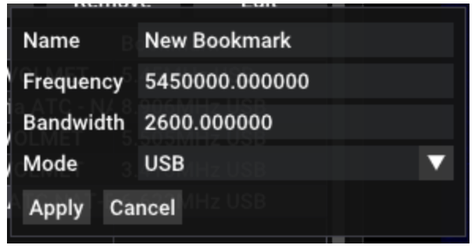
移除列表

您可以通过点击减号（‘-’）按钮来删除列表。这将删除其中的所有书签。

书签

创建书签

要将一个频率保存为书签，请点击“添加”。这会弹出一个框，您可以在其中为书签创建名称。系统会自动使用SDR++ 当前调谐的频率、带宽和模式来预填书签。不过，如果您愿意，也可以编辑这些信息。



点击“应用”以保存。

书签将在列表中按名称的字母顺序显示。书签不支持 Unicode 字符。书签应具有唯一名称，因为不允许重复。如果您确实想使用相同的名称，那么在名称后添加一个或多个空格就足以使其不被视为重复。

调至书签的频率

要执行此操作，只需打开列表以显示书签，然后双击它。

你也可以点击瀑布图上显示的书签来调到该频点。

Remove a bookmark

You can delete a bookmark by opening the list it is in, highlighting the bookmark and selecting the ‘Remove’ button.

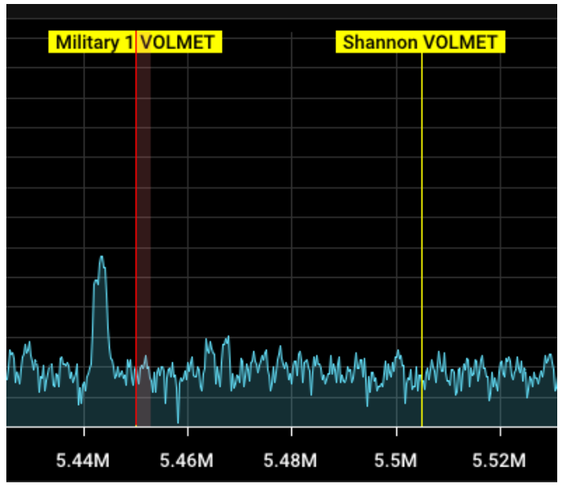
Edit a bookmark

You can edit a bookmark by opening the list it is in, highlighting the bookmark (with a single click) and clicking on the ‘Edit’ button. This brings up a similar box to when you created it, and the name, frequency, bandwidth and mode can all be edited.

Displaying bookmarks in the spectrum display

By default, the bookmarks in all your lists will be shown on the spectrum display. You can alter this by clicking on ‘ Select displayed lists’. This will show all your lists. Tick the ones you want to be displayed.

You can choose whether the names of the bookmarks are displayed at the bottom or the top of the Spectrum display. Here’s an example where they’re displayed at the top. SDR++ is tuned to the one on the left.



移除书签

您可以通过打开书签所在的列表，选中该书签，然后点击‘删除’按钮来删除书签。

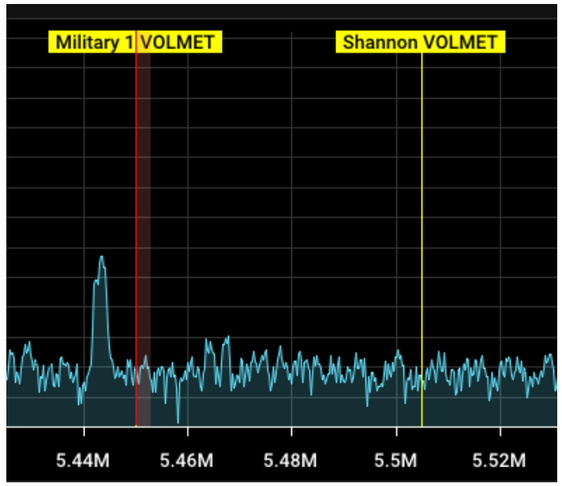
编辑书签

你可以通过打开书签所在的列表、单击选中书签，然后点击“编辑”按钮来编辑书签。这会弹出一个与创建书签时类似的对话框，你可以编辑书签的名称、频率、带宽和模式。

在频谱显示中显示书签

默认情况下，您所有列表中的书签将显示在光谱显示中。您可以通过点击“选择显示的列表”来更改此设置。这将显示您所有的列表。勾选您希望显示的列表。

您可以选择书签名称显示在 Spectrum 显示的底部或顶部。这里是一个显示在顶部的例子。SDR++ 调整到左侧的那个。



Importing and exporting bookmarks

You can import bookmarks into SDR++ that you have exported to a file from another instance of the program. For instance, if you are running SDR++ on another computer then you can copy over the saved bookmarks from there.

To export your bookmarks, you need to

- open the Frequency Manager module
- display the list that you want to export
- highlight the bookmarks you wish to export (click on each one, with the control key pressed down)
- press the export button
- Save the file with a name of your choice (for instance the 'List' name) and the file suffix `.json`

To import your saved bookmarks

- open the Frequency Manager module
- display or create the list(s) that you want to import the bookmarks into
- Select 'Import', locate the json file with the bookmarks and press OK

Bookmarks can be shared - so if you have a list of frequencies for your favourite stations you can export this and provide it to someone else with SDR++, who can then import and use them.

The full list of bookmarks, lists and settings are contained in the `frequency_manager_config.json` file in the directory where SDR++ is installed. If you are upgrading between versions of SDR++ you can either

- Create a copy of this file and move it into the root folder of your upgraded version
- Install SDR++ to the same folder without deleting files - and the upgraded version will use this file for frequency manager
- Export the bookmarks you want to save, so you can restore them afterwards. This will take longer than the other options if you want to keep all your bookmarks.

导入和导出书签

您可以将书签导入到 SDR++，这些书签是从程序的另一个实例导出到文件中的。例如，如果您在另一台计算机上运行 SDR++，则可以从那里复制保存的书签。

要导出您的书签，您需要

- 打开频率管理模块
- 显示您想要导出的列表
- 突出显示你希望导出的书签（按住控制键点击每一个书签）

- 点击导出按钮
- 使用你选择的名称保存文件（例如使用 'List' 作为名称），文件后缀为 .json

导入您保存的书签

- 打开频率管理模块
- 显示或创建要导入书签的列表
- 选择“导入”，找到包含书签的 JSON 文件，然后按确定

书签可以共享——所以如果你有最喜欢的电台频率列表，你可以导出它并提供给使用 SDR++ 的其他人，他们随后可以导入并使用这些频率。

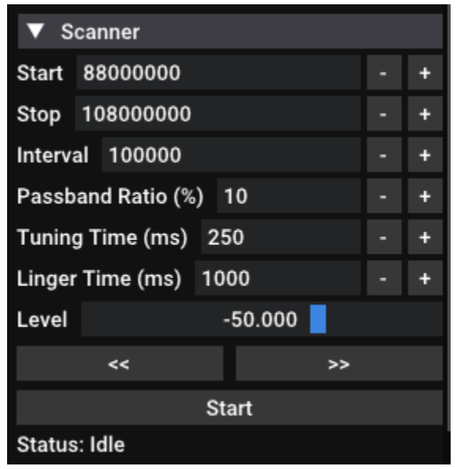
完整的书签、列表和设置都包含在 SDR++ 安装目录中的 frequency_manager_config.json 文件中。如果您在升级 SDR++ 的版本时，可以选择

- 创建此文件的副本，并将其移动到已升级版本的根文件夹中
- 将 SDR++ 安装到同一文件夹而不删除文件——升级后的版本将使用此文件作为频率管理器
- 导出您想要保存的书签，以便之后可以恢复。如果您想保留所有书签，这将比其他选项花费更多时间。

Scanner

This feature is available currently on nightly builds on SDR++ as a module you can add, but should be available for the version 1.1.x release.

The scanner searches across a specified range of the frequency spectrum. It monitors the whole range through accessing the FFT. This is very quick, especially if the frequency range is wholly within the bandwidth of your SDR receiver.



The scanner will move instantly to a signal it detects. It does not need to laboriously move from one frequency to another, like a conventional scanner radio.

It will move the VFO to the strongest active signal above a set level.

If you don't want to listen to the signal you can move to the next detected signal on another lower or higher frequency.

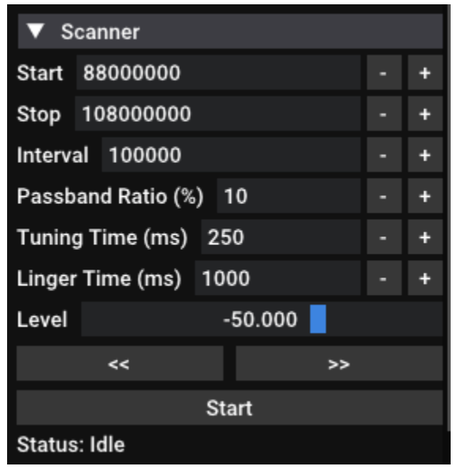
The scanner is very fast at detecting transmissions, but may also stop on strong interference ('QRM'). There is no facility to lock out a frequency in the scanned range.

At present the scanner does not scan memory banks of frequencies in Frequency Manager, nor can you save search ranges. The ranges you set are not currently persistent and will be lost when you close SDR++.

扫描仪

此功能目前在 SDR++ 的夜间构建版本中作为可添加的模块可用，但应在 1.1.x 版本发布时可用。

扫描器会在指定的频率范围内进行搜索。它通过访问快速傅里叶变换（FFT）来监控整个范围。这非常迅速，尤其是当频率范围完全在您的SDR接收器带宽内时。



扫描仪会立即移动到它检测到的信号。它不需要像传统扫描收音机那样费力地从一个频率移动到另一个频率。

它将把VFO移动到高于设定水平的最强活动信号处。

如果你不想听这个信号，你可以切换到另一个较低或较高频率上检测到的下一个信号。

扫描仪在检测传输方面非常快速，但也可能在强干扰（‘QRM’）时停止。在扫描范围内没有锁定频率的功能。

目前，扫描器无法扫描频率管理器中的频率存储库，也无法保存搜索范围。您设置的范围目前不会被保存，关闭 SDR++ 后将会丢失。

Scanner settings

Stop/Start: This sets the range of the frequency spectrum that is scanned. The setting is in hertz, so to scan the airband you might use 118000000 (118MHz) as the start frequency and 136000000 as the stop frequency.

> **Tip**: It can be better to scan smaller frequency ranges where you can see that there is no interference. You'll be surprised at how many new frequencies in use that you'll find with the SDR++ scanner as you'll not miss frequencies that a conventional scanner wouldn't reach before the transmission ended.

Interval: The scanner will be checking all the frequencies for activity but will set the channel spacing for these signals at a specific interval. So, if the frequency plan sets 12.5kHz channel spacing, you would set the interval to 12500.

When scanning, set the 'start' frequency at the first channel so the spacing for subsequent channels will be correct. The scanner will put the VFO at the centre frequency for each channel if it detects a signal within that channel's bandwidth.

You should ensure that the bandwidth in the radio module is set correctly for the type of signal for which you are searching. So, if you were scanning the broadcast FM bands, your interval would be 100kHz (100000), as stations are spaced at these intervals on the FM band, but your bandwidth for a wide FM transmission would be between 150 and 200kHz.

Passband ratio: This sets the amount of the bandwidth of a frequency (starting from the centre of the frequency) that will be checked for a modulated signal, and thus stop the scan so SDR++ can receive it. A narrow percentage will better reject adjacent channel interference

Tuning time: This sets the time for checking the FFT, once the level is exceeded, for a signal. The default is 250ms (a quarter of a second). This should be adequate for most purposes.

Linger Time: If the scanner has successfully detected and demodulated a signal, this sets the amount of time after the signal stops transmitting before the scanner abandons the frequency and resumes scanning. This delay enables there to be a pause between, for

扫描仪设置

起始/停止：此设置用于设定被扫描的频谱范围。该设置以赫兹为单位，因此如果扫描航空频段，你可以将起始频率设置为118000000（118MHz），停止频率设置为136000000。

> 提示：扫描较小的频率范围可能更好，这样你可以确认没有干扰。你会惊讶于使用 SDR ++ 扫描器时能发现多少新的频率使用情况，因为你不会错过传统扫描器在传输结束前无法捕捉到的频率。

间隔：扫描器将检查所有频率的活动，但会将这些信号的频道间隔设置为特定的间隔。
因此，如果频率计划设置为12.5kHz的频道间隔，则您应将间隔设置为12500。

扫描时，将“起始”频率设置为第一个频道，以便后续频道的间隔正确。如果扫描器在某个频道的带宽内检测到信号，它会将VFO设置在该频道的中心频率。

您应该确保无线电模块中的带宽设置与您正在搜索的信号类型相匹配。因此，如果您正在扫描广播调频（FM）频段，您的间隔应为100 kHz（100000），因为电台在FM频段上的间隔为这些值，而对于宽带FM传输，您的带宽应介于150到200 kHz之间。

通带比率：此设置决定从频率中心开始，将检查多少频率带宽以检测调制信号，从而停止扫描，使SDR++ 能够接收信号。较窄的百分比可以更好地抑制邻道干扰

调谐时间：这设置了检查 FFT 的时间，当信号超过设定水平时进行检测。默认值为 250 毫秒（四分之一秒）。对于大多数用途来说，这应当是足够的。

停留时间：如果扫描器已成功检测并解调信号，此设置定义了在信号停止传输后扫描器放弃该频率并恢复扫描之前的时间。这一延迟允许在信号之间有一个暂停，便于

instance, the base transmission and the reply from the mobile station, without the scanner leaving the frequency. You can thus hear the reply of the other station. The default is 1000ms (one second). You might increase this if stations are taking longer to reply to each other, or to give you longer to stop the scan so you can monitor that frequency.

Level: Conventional scanners use an audio squelch, so that they stop on a frequency where the level of signal is above the level at which the squelch is set. **SDR++ scanner does not use the audio squelch** to determine where it 'stops'. Instead it goes to signals above a set level on the FFT. Look at the noise level on the FFT and set it for something slightly above this. So, for example, if the noise is peaking at -70, start by setting it at -60.

Adjust the setting from there so the scanner stops on the signals you want to hear. You can drag the level to the right to make the scanner less sensitive by stopping only on stronger signals. You can drag it to the left to be more sensitive - but it will be more likely to stop on noise or interference.

> **Tip**: You might want to adjust the level with the squelch off, so you can hear everything that the scanner is stopping on. Adjust it by small increments until the VFO is skipping to the detected signals.

Resume scan (up or down >> or <<)

When the scanner stops on an unwanted signal or on interference, you can cause it to resume scanning and go to other signals though these controls. Obviously, >> sends it to the next identified signal higher in frequency, and the reverse with the other control.

Start/Stop: Commences or ends the scan. You will also need to have pressed the 'Play' button on the top bar to start SDR++ receiving. The scanner does not stop when the Play button is pressed again to stop SDR++ receiving.

Status indicator: This shows

- Idle - when it is not scanning
- Scanning
- Tuning - checking that a signal is present
- Receiving

例如，基本传输和来自移动台的回复，而扫描器无需离开该频率。这样你就可以听到另一台站的回复。默认值为1000毫秒（1秒）。如果各台站的互相回复时间较长，或者你希望有更长的时间来停止扫描以便监控该频率，你可以增加该值。

级别：传统扫描仪使用音频静噪功能，因此它们会在信号水平高于静噪设定值的频率上停止。SDR**++** 扫描仪不使用音频静噪来决定其‘停止’的位置。相反，它会关注FFT上高于设定水平的信号。观察FFT上的噪声水平，并将其设置为略高于噪声的值。例如，如果噪声峰值为-70，可以从设置-60开始。

从那里调整设置，使扫描仪在您想听到的信号上停下。您可以向右拖动级别，使扫描仪不那么灵敏，只在较强的信号上停下。您也可以向左拖动，使其更灵敏——但这样更可能会在噪音或干扰上停下。

> 提示：您可能需要在关闭静噪时调整音量，这样您就可以听到扫描器所阻止的所有信号。通过小幅度调整，直到VFO跳到检测到的信号为止。

恢复扫描（向上或向下 >> 或 <<）

当扫描仪停在不需要的信号或干扰上时，你可以通过这些控制使其继续扫描并转到其他信号。显然，>> 会将其发送到频率更高的下一个已识别信号，而另一个控制则相反。

开始/停止：开始或结束扫描。你还需要按下顶部栏的“播放”按钮来开始接收SDR++ 。当再次按下播放按钮停止接收SDR++ 时，扫描器不会停止。

状态指示器：显示此内容

- 空闲 - 当它未在扫描时
- 扫描
- 调谐 - 检查信号是否存在
- 接收

Scanner tips:

- Save the frequencies you identify in Frequency Manager, so when the VFO goes to them you can quickly see what the transmission is by checking the marker displayed above it on the Spectrum Display.
- The Spectrum Display may move about as the scanner moves the VFO to different frequencies. If you find this disorienting, you can either
    - Set the tuning mode selector to centre tuning mode so that the VFO is always centre screen, but this can slow down the scanner or
    - Adjust the scan range and sampled bandwidth of the SDR receiver so that the displayed spectrum is stable and the only the VFO 'cursor' moves around. This also has the advantage that the waterfall will be stable and can show a history of signals received on these frequencies within the sampled spectrum. It will, though, reduce the number of signals you detect, and will restrict you to a very small portion of a band if you have an SDR with a small sample width such as the HF+ Discovery (about 600kHz) or an RTL-SDR (about 2MHz).

扫描仪提示：

● 将您在频率管理器中识别的频率保存，这样当VFO切换到这些频率时，您可以通过检查频谱显示上方显示的标记快速了解传输内容。

● 随着扫描器将VFO移至不同的频率，频谱显示可能会移动。如果你觉得这样令人迷惑，你可以选择

○ 将调谐模式选择器设置为中心调谐模式，使VFO始终在屏幕中心，但这可能会减慢扫描速度或

○ 调整SDR接收机的扫描范围和采样带宽，使显示的频谱稳定，并且只有VFO“光标”在移动。这还有一个好处，即瀑布图将保持稳定，并可以显示在采样频谱内接收到的这些频率的信号历史。不过，这会减少你检测到的信号数量，如果你的SDR采样宽度较小（如HF+ Discovery约600kHz或RTL-SDR约2MHz），你将只能接收到频段的一小部分。

Recorder

You are able to record either the

- audio from the signal you are tuned to or
- the ‘baseband’.

The baseband is the whole of the raw IQ data from the bandwidth the SDR is processing. You can later select that file in the ‘Source’ module and play it back as if it was a live session using your SDR.

Some people record the baseband to explore their favourite band, for instance MW dxers will record the whole medium wave band through part of a night and play it over and over to find distant stations on all the different frequencies recorded.

By default the recorder will save the file in a ‘recordings’ sub-folder of the directory where SDR++ is installed. In Linux this is in ~/.config/sdrpp. Click on the 3 dots if you want to change the folder.

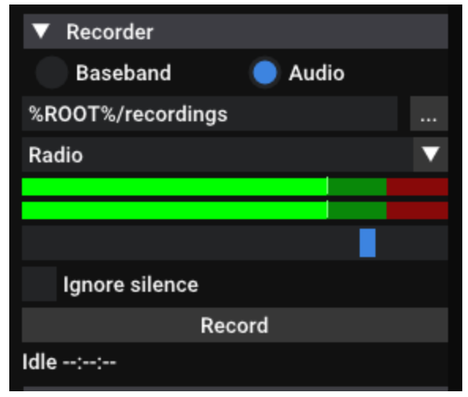
There is a signal strength meter to enable you to adjust the output so that only the strongest peaks go into the red.

录音机

您可以记录其中任何一个

- 来自您所调谐信号的音频或
- “基带”。

基带是指SDR处理带宽的全部原始IQ数据。之后，您可以在“源”模块中选择该文件，并像使用SDR进行实时会话一样播放它。



有些人会录制基带来探索他们喜欢的电台，例如中波远距接收爱好者会在一晚上的时间里录制整个中波段，然后反复播放，以在录制的各个频率上寻找遥远的电台。

默认情况下，录音机会将文件保存在 SDR++ 安装目录下的“recordings”子文件夹中。在 Linux 系统中，这个目录位于 ~/.config/sdrpp。如果您想更改文件夹，请点击三个点。

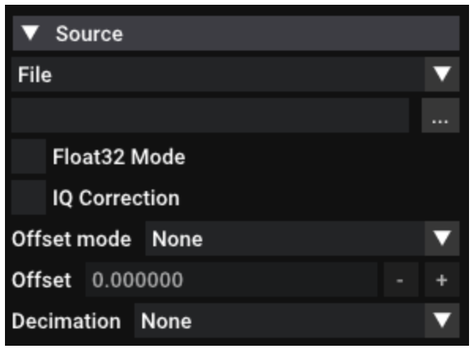
有一个信号强度计，可以让您调整输出，使只有最强的峰值进入红色区域。

You can use ‘Ignore silence’ to reduce the file size of your audio recordings - because it won’t record periods when there is no audio. This is useful when you are using the squelch to monitor and record intermittent transmissions using NFM.

Audio and baseband (IQ) recordings are in the .WAV format and have a maximum file size of 4GB. Baseband recordings are 16 bit signed samples.

When you are ready, press ‘Record’, which will change into a ‘Stop’ button whilst recording. You can use this, of course, to halt the recording.

To play back recordings of the raw IQ (baseband) data, use the ‘File’ option in Sources:



You can choose the recorded file by selecting the three dots then navigating to the saved file. As with using the SDR receiver as a source, you can apply an offset and decimation etc.

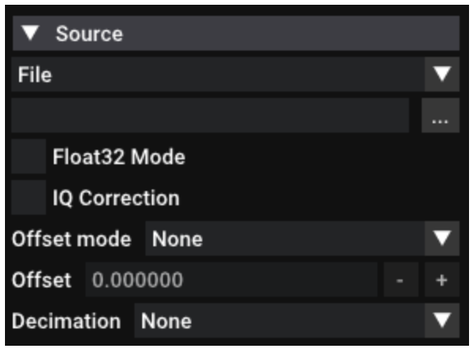
To play an audio recording, simply use the standard audio player for your operating system.

您可以使用“忽略静音”来减小音频录音的文件大小——因为它不会录制没有音频的时间段。当您使用消噪功能来监控和录制间歇性传输（NFM）时，这非常有用。

音频和基带（IQ）录音为 .WAV 格式，最大文件大小为 4GB。基带录音是 16 位有符号样本。

准备好后，按下“录音”按钮，录音时该按钮会变为“停止”。当然，你可以使用它来停止录音。

要回放原始 IQ（基带）数据的录音，请在“源”中使用“文件”选项：



您可以通过选择三个点然后导航到已保存的文件来选择录制的文件。与使用SDR接收器作为源一样，您可以应用偏移量、抽取等设置。

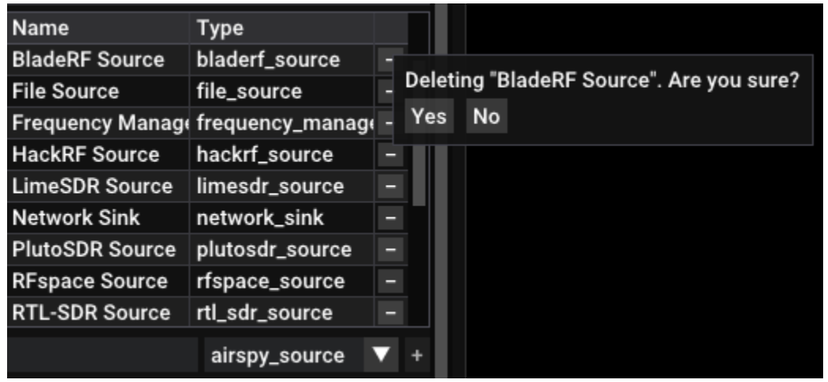
要播放音频录音，只需使用您操作系统的标准音频播放器。

Adding and removing modules

You can add - and remove - many modules for SDR++. Adding modules can increase its functionality and flexibility. Removing modules can help you slim down the SDR++ so it operates more efficiently without unnecessary functions. You use the Module Manager to do this.

Removing a module

To remove a module you simply need to use the ‘minus’ symbol next to its name in Module Manager. You’ll be asked to confirm this.



Adding a module

We are going to use the example of adding additional VFOs to show how to add a module.

Adding Multiple VFOs

SDR++ allows you to tune to more than one frequency at the same time using the same SDR receiver.

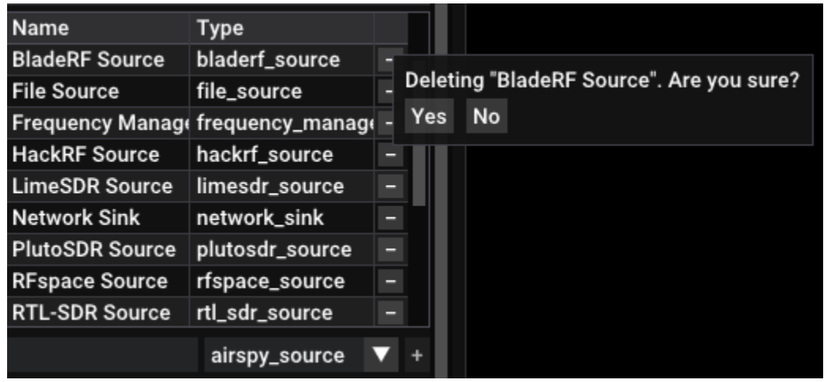
The frequencies you choose have to be within the ‘sample width’ of the SDR receiver. For example if you have an RTL SDR blog dongle, it only covers about 2MHz of RF bandwidth. If you zoom out you will see all of this sample width in the spectrum display and waterfall -

添加和移除模块

您可以为 SDR++ 添加和移除许多模块。添加模块可以增强其功能性和灵活性。移除模块可以帮助您精简 SDR++ ， 使其在没有不必要功能的情况下更高效地运行。您可以使用模块管理器来执行此操作。

移除模块

要删除一个模块，您只需在模块管理器中点击其名称旁的“减号”符号。系统会要求您进行确认。



添加模块

我们要去 使用添加额外VFO的示例来演示如何添加模块 .

添加多个VFO

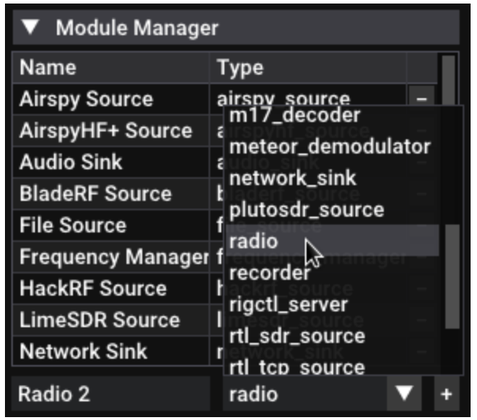
SDR++ 允许您使用同一个SDR接收器同时调谐到多个频率。

您选择的频率必须在SDR接收机的“采样宽度”范围内。例如，如果您有一个RTL SDR博客加密狗，它只能覆盖大约2MHz的射频带宽。如果您缩小视图，您将在频谱显示和瀑布图中看到整个采样宽度。

for instance it might cover 6MHz to 8MHz. The 2 frequencies that you tune to must be within this - so you could tune to both a China Radio International on 7220kHz AM in the 41m broadcast band, and to a radio amateur on 7120 kHz LSB on the 40 metre amateur band.

How to set up a second VFO

Go to Module Manager and create a new ‘radio Module’. To do this:



1. Type a name for the new module in the box on the bottom left - here ‘Radio 2’
2. Select ‘radio’ from the drop down ‘Type’ menu on the right
3. Click on the ‘+’ button to create.

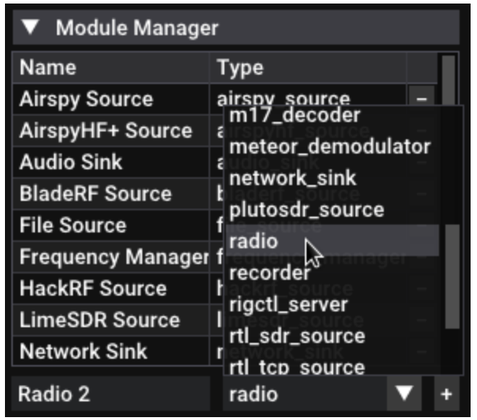
This will create the new module which will be placed at the bottom of the menu. You can drag it into a convenient position, for instance below the other VFO entry (‘Radio’). You will see a second VFO band appear in the Spectrum display in a different color which shows the frequency of the second VFO.

Before you can use your second VFO, you need to configure a ‘sink’ for its audio output to go into. This ‘sink’ is created automatically when you created the second VFO:

例如，它可能覆盖 6MHz 到 8MHz。你调谐的两个频率必须在此范围内——所以你可以调谐到中国国际广播在 41 米广播波段的 7220 kHz AM，同时也可以调谐到在 40 米业余波段的 7120 kHz LSB 的业余无线电频道。

如何设置第二个VFO

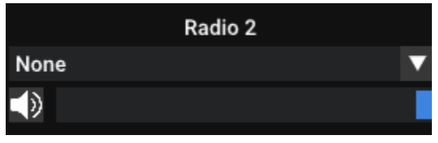
前往模块管理器并创建一个新的“无线电模块”。操作如下：



1. 在左下角的框中输入新模块的名称——这里输入“Radio 2” 2. 在右侧的“类型”下拉菜单中选择“广播” 3. 点击“+”按钮以创建。

这将创建一个新的模块，该模块将放置在菜单底部。你可以将它拖动到方便的位置，例如在另一个 VFO 条目（“无线电”）下方。你将在频谱显示中看到第二个 VFO 频段以不同的颜色出现，显示第二个 VFO 的频率。

在您使用第二个VFO之前，您需要为其音频输出配置一个“接收端”。当您创建第二个VFO时，这个“接收端”会自动创建：

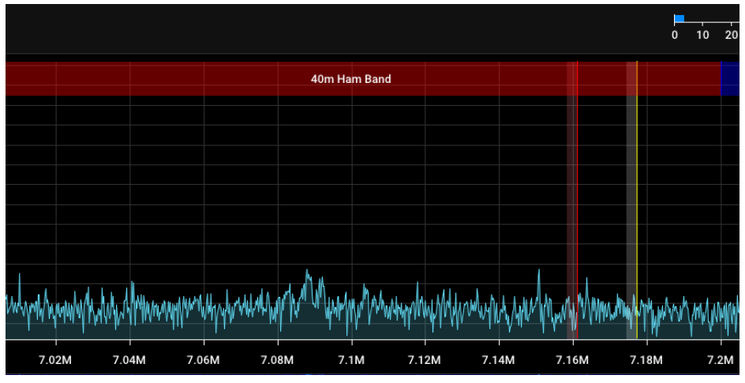


To listen to this second VFO

- click the drop down box and select ‘audio’ (you can also send it over a network)
- select the soundcard or speakers etc that you want to use
- drag the slider for audio level into the middle so the sound won’t be too loud.

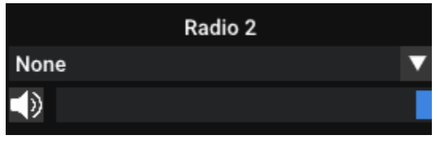
Finally, to use the second VFO:

- tick the box next to its name in the module you created
- start SDR++ receiving by clicking on the ‘start’ arrow
- tune the second VFO by dragging the VFO band to a new frequency. It will change color to show that it is the VFO being controlled. You can select the modulation mode (AM, USB etc.) independently of the mode of the first VFO.



In this picture there are 2 VFOs, with the one being controlled in red, and the other in yellow.

If you want to identify a VFO, hover your mouse over the VFO band in the spectrum display and it will show the VFO’s name. The frequency ‘line’ in the VFO band will be fixed depending on whether it is being controlled or not. However, you can alter the color of the rest of the band on each one by using the VFO color control in the menu.

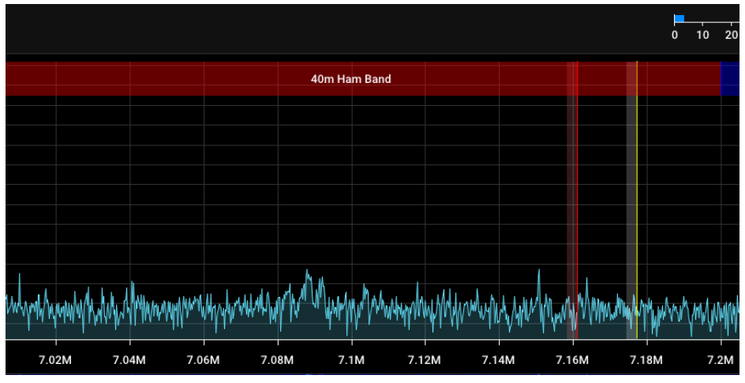


监听这个第二个VFO

- 点击下拉框并选择“音频”（你也可以通过网络发送）
- 选择您想使用的声卡或扬声器等
- 将音量滑块拖到中间位置，以免声音过大。

最后，使用第二个VFO：

- 在你创建的模块中勾选其名称旁的框
- 点击“开始”箭头以启动 SDR++ 接收
- 通过拖动VFO频段到新的频率来调谐第二个VFO。它会变色以显示正在被控制的VFO。您可以独立于第一个VFO的模式选择调制方式（AM、USB等）。



在这张图片中有两个*VFO*，一个被控制的显示为红色，另一个为黄色。

如果你想识别一个VFO，请将鼠标悬停在频谱显示中的VFO频段上，它将显示VFO的名称。VFO频段中的频率“线”将固定不变，这取决于它是否正在被控制。不过，你可以通过菜单中的VFO颜色控制更改每个VFO其他频段的颜色。

The frequency selector will show only the frequency of the VFO that is currently being controlled.

You can cycle between the VFOs by using the page up and page down keys.

Having multiple VFOs is very useful. You could use one VFO to send a USB audio signal to WSJT-X via a virtual audio cable to monitor FT8 transmissions, and a second to listen to radio amateurs talking on SSB, further up the same band (or on different bands if you have a large sample width). Alternatively, you might use multiple VFOs to monitor several signals on a satellite downlink, or to stand watch over several frequencies where signals are intermittent - like air traffic control.

频率选择器将仅显示当前正在控制的VFO的频率。

你可以使用Page Up和Page Down键在VFO之间切换。

拥有多个VFO非常有用。你可以使用一个VFO通过虚拟音频线向WSJT-X发送USB音频信号来监控FT8传输，并使用第二个VFO收听同一频段上更高频率（或者如果你有较宽的采样宽度，也可以在不同频段）的业余无线电通信。或者，你可以使用多个VFO来监控卫星下行的多个信号，或监控多个频率上间歇出现的信号——比如空中交通管制。

Advanced Features and applications

Connecting to remote SDRs

SDR++ can act as a client for remote servers over a network connection, including SDR++ server and Airspy Spyservers. SDR++ server is simple because it is built in, and we would recommend that you use this when setting up a remote receiver for your own use.

Please note: if SDR++ can't connect to a remote server, it may appear to do nothing for up to 20 seconds before timing out. This delay is because some SDR receivers can take time to be started by the server or because the remote server has crashed. Occasionally, this might cause SDR++ to crash.

SDR++ Server

SDR++ will operate as a server to provide the IQ stream over a network connection to another instance of SDR++ running on a different device elsewhere on a network - including over an internet connection. You can then use the client SDR++ to work with that signal in the same way as if it was an SDR receiver on your own device. The server can be run from the command line, for instance on a headless Raspberry Pi 3 or 4.

On the device acting as server you'll need to launch SDR++ in server mode, specifying the port to output the IQ stream over the network. Make a note of the IP address of the device. On your device you'll need to open the port and alter any firewall settings.

If connecting over the internet then you'll need to alter some settings on your router: you need to open the port and direct the input on that port towards the IP address of the device hosting the SDR++ server.

You can launch the server from the command line with

```
sdrpp --server --port [port no]
```

For instance to start it on port 12001

```
sdrpp --server --port 12001
```


高级功能和应用

连接到远程 SDR

SDR++ 可以作为客户端，通过网络连接连接远程服务器，包括SDR++服务器和Airspy Spyservers。SDR++ 服务器很简单，因为它是内置的，我们建议在设置自己的远程接收器时使用它。

请注意：如果 SDR++ 无法连接到远程服务器，它可能会在超时前的最长 20 秒内看似无任何反应。造成此延迟的原因可能是某些 SDR 接收器需要服务器的时间来启动，或者远程服务器已崩溃。偶尔，这可能会导致 SDR++ 崩溃。

SDR++ 服务器

SDR++ 将作为服务器运行，通过网络连接向另一台设备上运行的SDR++ 实例提供IQ流——包括通过互联网连接。然后，您可以使用客户端SDR++ 以与在您自己的设备上使用SDR接收器相同的方式处理该信号。服务器可以从命令行运行，例如在无头的Raspberry Pi 3或4上。

在作为服务器的设备上，您需要以服务器模式启动 SDR++ ，并指定用于通过网络输出 IQ 流的端口。记下该设备的 IP 地址。在您的设备上，您需要打开该端口并修改任何防火墙设置。

如果通过互联网连接，则需要更改路由器上的一些设置：您需要打开端口，并将该端口的输入指向托管SDR++ 服务器的设备的IP地址。

您可以从命令行启动服务器，使用

```
sdrpp --server --port [端口号]
```

例如，在端口12001上启动它

```
sdrpp --server --port 12001
```

You can omit the port setting. SDR++ Server will start by default on port 5259, and the client will be set at that port by default, also.

You can also add `--addr` to specify which IP address SDR++ listens on, in case the server has multiple network devices or connections.

On the client device you should

- select SDR++ Server as the source
- specify the IP address and port of the server (default is 5259)
- select the sample type - 'Int8' (8 bit) should work and will give th best performance over a network because it reduces the data across it

SDR++ on your client device now has full control over the source settings of the SDR receiver on the device running on the SDR++ server.

- enter the type of SDR receiver in 'Source [REMOTE]'
- select compression if you have a more powerful CPU on you client device - this may also reduce the network bandwidth used
- Select 'Connect'
- Press the start arrow

You'll now start receiving the IQ stream and, if your settings are correct, you can use this as if it were the same SDR receiver connected directly to your client device. You'll be able to alter source settings such as AGC and bandwidth.

To disconnect, stop receiving on SDR++, and press disconnect.

You can stop SDR++ server by closing the terminal or 'cmd' box, or by hitting ctrl + C keys together.

SDR++ server works best on a good network connection. The signal may be intermittent on a poor connection, particularly where a wifi connection is used for the server or client.

您可以省略端口设置。SDR++ Server 默认会在端口 5259 启动，客户端也将默认设置为该端口。

你也可以添加 --addr 来指定 SDR++ 所监听的 IP 地址，以防服务器有多个网络设备或连接。

在客户端设备上，您应该

- 选择 SDR++ Server 作为源
- 指定服务器的 IP 地址和端口（默认端口是 5259）
- 选择样本类型 - ‘Int8’（8位）应该可用，并且在网络上会提供最佳性能，因为它减少了通过网络的数据量

现在，您客户端设备上的 SDR++ 可以完全控制运行在 SDR++ 服务器上的设备的 SDR 接收器的源设置。

- 在“Source [REMOTE]”中输入SDR接收器的类型
- 如果您的客户端设备有更强大的 CPU，请选择压缩——这也可能减少使用的网络带宽
- 选择“连接”
- 按下开始箭头

您现在将开始接收 IQ 流，如果您的设置正确，您可以将其像直接连接到客户端设备的同一 SDR 接收器一样使用。您将能够更改源设置，例如 AGC 和带宽。

要断开连接，请停止在SDR++上接收，然后按断开。

你可以通过关闭终端或“cmd”窗口，或者同时按下 Ctrl 和 C 键来停止 SDR++ 服务器。

SDR++ 服务器在良好的网络连接下效果最佳。在网络连接不佳的情况下，信号可能会间歇性出现，尤其是在服务器或客户端使用 Wi-Fi 连接时。

SpyServer

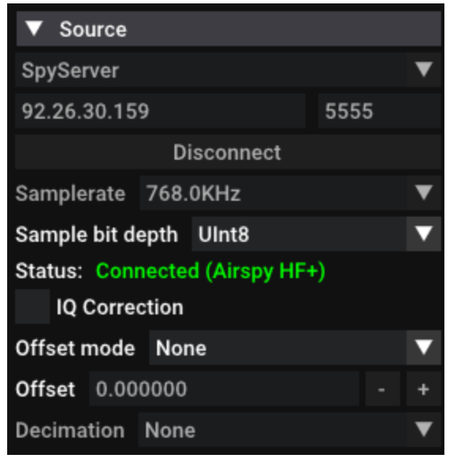
You can control SDR receivers that are not directly connected to your computer using SDR++. This includes SDR receivers using the Airspy Spyserver software.

SDR++ server may be a better choice: Spyserver only works with RTL-SDR, Airspy and Airspy HF+ receivers, whereas SDR++ server works with any SDR receiver or protocol supported by SDR++. SpyServer is not open source, and you have less control over the remote SDR receiver source settings.

You can set up the Spyserver on your own network, but this section deals with how to connect and listen rather than set this up.

You can find a tutorial on setting up a Spyserver on the RTL-SDR blog.

You need to set up a connection by selecting SpyServer in the Source menu:



To listen to an external SpyServer receiver:

- Go to the list of SDRs on the Airspy Spyserver network at https://airspy.com/directory/

间谍服务器

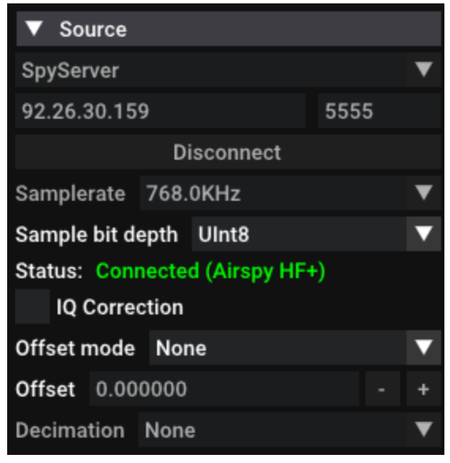
您可以使用 SDR++ 控制未直接连接到计算机的 SDR 接收器。这包括使用 Airspy Spyserver 软件的 SDR 接收器。

SDR++ 服务器可能是更好的选择：SpyServer 仅适用于 RTL-SDR、Airspy 和 Airspy HF+ 接收器，而 SDR++ 服务器则可与 SDR++ 支持的任何 SDR 接收器或协议配合使用。SpyServer 不是开源的，你对远程 SDR 接收器的源设置控制较少。

您可以在自己的网络上设置 Spyserver，但本节讲的是如何连接和监听，而不是如何设置它。

你可以在 RTL-SDR 博客上找到关于设置 Spyserver 的教程。

您需要通过在“源”菜单中选择 SpyServer 来建立连接：



要监听外部 SpyServer 接收器：

● 访问 Airspy Spyserver 网络上的 SDR 列表，网址：https://airspy.com/directory/

- Find a remote SDR that is showing a green wireless symbol to show that it has free slots for users to connect.  Click to open the details of the SDR
- Copy the web address for the SDR e.g. sdr://92.26.30.159:5555
- Open the Source menu on SDR++
- Put the IP address of the Spyserver SDR (i.e. 92.26.30.159 in the example above) into the left hand box.
- Put the port number - shown after the colon (:) and usually 5555 - but it can vary) into the right hand box
- In the Sample bit depth try 'UInt8' - this is the lowest quality but will pass the least data across your internet connection and give you the best chance of a listenable connection. You can raise it to a higher bit depth when you are sure the connection is a reliable one, or if you are having trouble with UInt8.
- Press 'connect'
- Press the start button in SDR++

Sometimes the audio will stutter and break up to start with. It should settle but experiment with other settings if it doesn't - for instance the sample bit depth or decimation. To disconnect, stop receiving on SDR++, and press disconnect.

If the SDR receiver that you connect to uses an up-converter, such as 'Ham it up' then you will need to adjust the offset (for more details see the description in the 'Source' section earlier in this guide).

For a receiver on your home network you would enter the IP address for that server.

If you are testing a Spyserver on the same computer as SDR++, you would use 'localhost' or 127.0.0.1.

● 找到显示绿色无线符号的远程 SDR，以表明它有空闲插槽供用户连接。点击以打开 SDR 的详细信息 

● 复制 SDR 的网页地址，例如 sdr://92.26.30.159:5555

● 在 SDR++ 上打开“源”菜单

● 将 Spyserver SDR 的 IP 地址（例如上例中的 92.26.30.159）输入左侧的框中。

● 将端口号（冒号 : 之后显示的数字，通常是 5555，但可能会有所不同）输入右侧的方框

● 在采样位深中尝试使用 'UInt8'——这是最低的质量，但通过您的互联网连接传输的数据最少，并且能给您提供最有可能可听的连接。当您确定连接可靠，或者在使用 UInt8 时遇到问题时，可以将其提高到更高的位深。

● 按“连接”

● 在 SDR++ 中按下开始按钮

有时音频一开始会出现卡顿和断断续续的情况。通常会逐渐稳定，如果没有，可以尝试调整其他设置，例如采样位深或抽取率。要断开连接，请在 SDR++ 上停止接收，然后按断开连接。

如果你连接的SDR接收器使用了上变频器，例如“Ham it up”，那么你需要调整偏移量（有关更多详情，请参见本指南前面“来源”部分的描述）。

对于家庭网络上的接收器，您需要输入该服务器的IP地址。

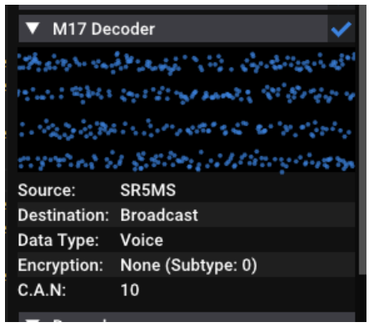
如果你在与 SDR++ 相同的计算机上测试 Spyserver，你可以使用 ‘localhost’ 或 127.0.0.1。

M17 decoder

M17 is a new open source digital audio mode that also uses open source hardware. SDR++ has a module to decode M17 that you can add from module manager. To do this, name the module M17 at the bottom of the Module Manager menu section. Select 'm17_decoder' from the drop down list then '+' to add the module to SDR++.

This adds

- a menu item for the decoder
- a dedicated M17 VFO, that will be shown on the spectrum display
- a new 'sink' for the audio output. You should ensure that the audio is directed to the sound card you want to use.



When you find an M17 transmission the M17 menu item will show the signal pattern (on which you can overlay reference lines), and the source name (usually the amateur radio operator's callsign).

Rig control server - allowing other programs to control SDR++

The rig control server will allow SDR++ to be controlled by another program, such as gpredict, to listen on a specific frequency, and record the signal. You will need to align the server settings in SDR++ with those in the program to enable this to work.

To use Rigctl, for instance, with WSJT-X on the same PC that you are using for SDR++, then:

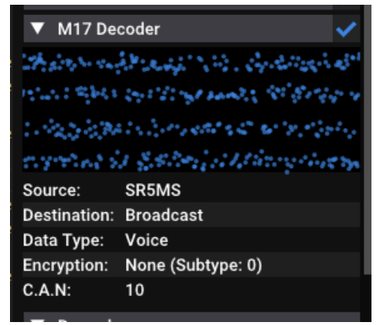
- In SDR++ set the rigctl server to 'localhost' and port '4532'
- If you have multiple VFOs select the one you want to be controlled by the server
- ensure 'Tuning' is ticked

M17 解码器

M17 是一种新的开源数字音频模式，同时也使用开源硬件。SDR++ 有一个可以解码 M17 的模块，你可以从模块管理器中添加。为此，在模块管理器菜单底部将模块命名为 M17。从下拉列表中选择“m17_decoder”，然后选择“+”将模块添加到 SDR++。

这增加了

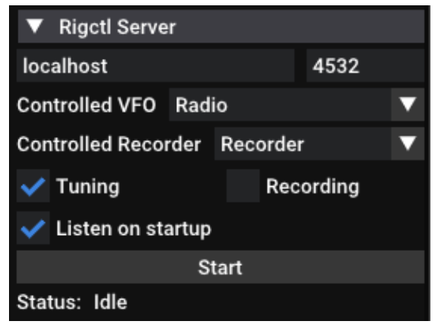
- 解码器的菜单项- 专用的 M17 VFO，将显示在频谱显示上- 音频输出的新“接收端”。你应确保音频被导向你想使用的声卡。



当你找到 M17 传输时，M17 菜单项将显示信号模式（你可以在其上叠加参考线），以及信号源名称（通常是业余无线电操作员的呼号）。

设备控制服务器 - 允许其他程序控制SDR++

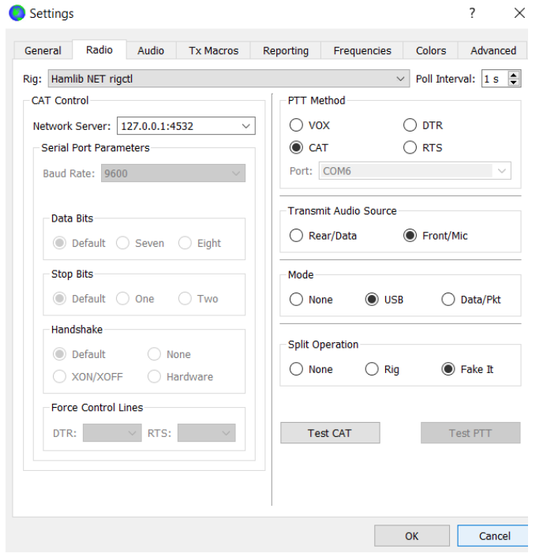
设备控制服务器将允许通过另一个程序（例如 gpredict）控制 SDR++，以监听特定频率并记录信号。您需要将 SDR++ 中的服务器设置与该程序中的设置对齐，以使其正常工作。



例如，要在同一台用于 SDR++ 的 PC 上将 Rigctl 与 WSJT-X 一起使用，则：

- 在 SDR++ 中，将 rigctl 服务器设置为“localhost”，端口设置为“4532”
- 如果您有多个VFO，请选择您希望由服务器控制的那个
- 确保勾选“调音”

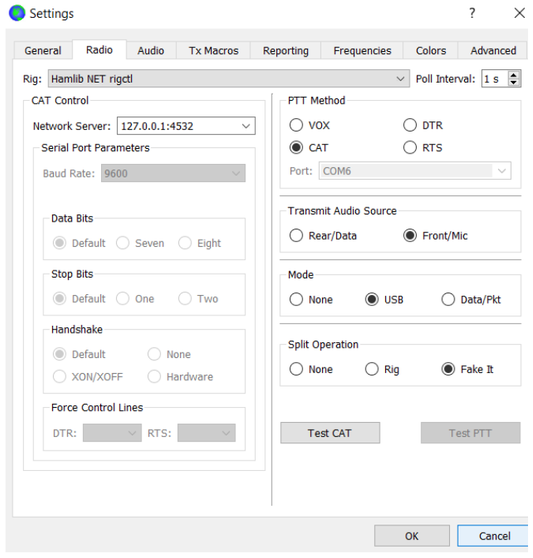
- start the rigctl server in SDR++ - it will move from 'Idle' to 'Listening'.
- In WSJT-X create a new configuration and call it SDR++
- In the configuration screen of WSJT-X:
  - Set Rig to 'Hamlib Net Rigctl'
  - Enter the network server as '127.0.0.1:4532'
  - Select PTT method as 'CAT'
  - Select Mode as 'USB'
  - Select split operation as 'Fake It'
- Press 'Test CAT' and check it goes green
- Check that SDR++ shows the status as 'Connected'
- Press the 'start' arrow in SDR++ to begin listening



You would also then need to have

- a virtual audio cable (or Linux equivalent) installed to route the audio from SDR++ to WSJT-X
- WSJT-X audio configuration set to receive audio from that source
- the 'sink' in SDR++ set to route the radio audio to the virtual audio cable.

- 在 SDR++ 中启动 rigctl 服务器——它将从“空闲”状态变为“监听”状态。
- 在 WSJT-X 中创建一个新配置，并将其命名为 SDR++
- 在 WSJT-X 的配置界面:
  - 将设备设置为 ‘Hamlib 网络 Rigctl’
  - 将网络服务器输入为 ‘127.0.0.1:4532’
  - 选择 PTT 方法为 ‘CAT’
  - 选择模式为“USB”
  - 选择拆分操作为‘伪造’
- 按下“测试 CAT”并检查是否变为绿色
- 检查 SDR++ 显示状态是否为“已连接”
- 在 SDR++ 中按下“开始”箭头以开始收听



你也需要有

- 安装了虚拟音频线（或 Linux 等效工具）以将 SDR++ 的音频传输到 WSJT-X
- WSJT-X 音频配置设置为从该来源接收音频
- 在 SDR++ 中将“接收端”设置为将无线电音频路由到虚拟音频线。

If you want the SDR++ to start the rigctl server when you open it, tick the box for 'Listen on startup'. Tick the 'Recording' box if you also want the output (usually the audio) to be recorded at the same time that SDR++ is being controlled by the external program (for instance for the duration of a satellite pass.

Meteor Demodulator

You can add this from the module manager. This extracts data from the Russian Meteor M2 weather satellite LRPT transmissions around 137MHz. The data needs a decoder such as 'SatDump' to produce a weather picture.

As of July 2022 there have been issues with the signal transmitted from the M2 Meteor satellite. SDR++ has been modified to address this. Select 'Broken modulation if Meteor is broken, or you'll get bad results. If Meteor is NOT broken, do NOT use it or you'll get bad results.

You can tell that Meteor is broken because:

1) There is a spike in the appearance of the Meteor signal in the FTT/ Spectrum display - it's supposed be smooth

2) The constellation is abnormal, as shown here.

如果您希望 SDR++ 在打开时启动 rigctl 服务器，请勾选“启动时监听”选项。如果您还希望在外部程序控制 SDR++ 的同时录制输出（通常是音频），例如在卫星经过期间录制，请勾选“录制”选项。

流星解调器

您可以从模块管理器中添加此功能。这可以从俄国Meteor M2气象卫星在约137MHz的LRPT传输中提取数据。数据需要像“SatDump”这样的解码器来生成气象图像。

截至2022年7月，从M2 Meteor卫星发射的信号出现了问题。SDR++ 已被修改以解决此问题。选择“如果Meteor损坏则使用破损调制”，否则你会得到错误的结果。如果Meteor没有损坏，请不要使用它，否则你会得到错误的结果。

你可以通过以下方式判断 Meteor 已损坏:

1) FTT/频谱显示中陨石信号的出现有一个尖峰——本应是平滑的

2) 星座异常，如图所示。

Using external Programs with a virtual audio cable

Windows

A virtual audio cable is a piece of software that can take an audio signal 'piped' from an SDR program - that would normally go to your speakers - and instead route it to programs that can use that audio. It doesn't use any external cables.

In that way the audio containing data signals can be decoded by a more specialist program that, for instance, can decode digital DMR and M17 voice signals, CW, radioteletype (RTTY), fax, FT8 and other amateur modes, digital broadcast modes such as DAB+ and DRM, satellite data, pagers, ADSB and ACARS messages from aircraft and so on.

You can download a free virtual audio cable from https://vb-audio.com/Cable/

Linux

Linux distros do not need a virtual audio cable but can accomplish the same thing in different ways.

ALSA is the native Linux sound system. You can add 'loopbacks' to do this[2]. You need to go to a terminal and launch your favourite text editor as root. So for gedit:

```
sudo gedit /etc/modules
```

In the text editor add as the last line: -

```
snd_aloop
```

Then save the file.

You'll then have at least 2 loopback outputs to use. You can select one of these e.g. 'hw:Loopback,0' in the 'Sinks' module as the radio audio output instead of your sound card. You can then choose this as the input in any Linux based program you are using. This will not work on the standard Raspberry Pi OS, so you should use a PulseAudio solution.

[2] (source) https://www.onetransistor.eu/2017/10/virtual-audio-cable-in-linux-ubuntu.html

使用虚拟音频线与外部程序

窗口

虚拟音频线是一种软件，它可以将从 SDR 程序‘管道’传出的音频信号——通常会发送到你的扬声器——改为路由到可以使用该音频的程序。它不使用任何外部电缆。

通过这种方式，包含数据信号的音频可以被更专业的程序解码，例如可以解码数字 DMR 和 M17 语音信号、CW、电传打字（RTTY）、传真、FT8 以及其他业余模式、数字广播模式如 DAB+ 和 DRM、卫星数据、寻呼机、飞机的 ADSB 和 ACARS 信息等。

You 可以从 https://vb-audio.com/Cabl 下载免费的虚拟音频线 e/

Linux

Linux 发行版不需要虚拟音频线，但可以通过不同的方式实现相同的功能。

ALSA 是 Linux 原生的声音系统。你可以添加“回环”来实现这一点。你需要 [2] 打开终端，并以 root 身份启动你喜欢的文本编辑器。因此，对于 gedit:

```
sudo gedit /etc/modules
```

在文本编辑器中作为最后一行添加：-

```
snd_循环
```

然后保存文件。

然后你将至少有两个环回输出可用。你可以在“Sinks”模块中选择其中一个，例如‘ hw:Loopback,0’，作为无线电音频输出，而不是你的声卡。之后，你可以在任何你使用的基于 Linux 的程序中将其选择为输入。在标准的 Raspberry Pi OS 上这将无法工作，所以你应该使用 PulseAudio 解决方案。

[2] (源) https://www.onetransistor.eu/2017/10/在Linux Ubuntu中使用虚拟音频线.html

With PulseAudio you can create a `module-null-sink` in `pavucontrol`, then pipe SDR++ into it through the sink. On the recording tab, select the virtual device, 'monitor', as the input for the decoder or other program you are using.

Pulseaudio works on top of ALSA. You can use Pulseaudio to achieve the same thing. See this tutorial, which was written for the Raspberry Pi but can work on other Linux devices.

You can also either:

- install and use rtaudio
- use a network sink, especially if the program you send it to can use UDP/TCP.

Programs

Here are some of the applications you can use with SDR++ and a virtual audio cable. There is a more extensive list at the RTL-SDR blog website.

DMR with DSD+

DMR is a digital voice mode used on VHF and UHF (over NFM signals) by business walkie-talkies and by radio amateurs. Business users can include schools and universities, shopwatch nets, building sites, tram and bus operators.

DSD+ can take an audio signal and decode the DMR data into voice outputs and can also detect the color code and channel used. The free version of DSD+ can also decode another digital voice mode - NXDN. The free version is functional but the paid version - DSD+ Fastlane - does more and can operate as a standalone SDR program without using SDR++.

Download DSD+ or subscribe to DSD+ Fastlane.

FT8 and other amateur digital modes with WSJT-X

WSJT-X enables you to decode a wide variety of data modes including FST4, FST4W, FT4, FT8, JT4, JT9, JT65, Q65, MSK144, WSPR and Echo (Earth-Moon-Earth signals). Most use upper sideband. All of these are weak signal modes and can travel very large distances with the right equipment.

使用 PulseAudio，你可以在 pavucontrol 中创建一个 module-null-sink，然后通过该 sink 将 SDR++ 输入其中。在录音选项卡中，选择虚拟设备“monitor”作为解码器或你正在使用的其他程序的输入。

Pulseaudio 是建立在 ALSA 之上的。你可以使用 Pulseaudio 来实现相同的功能。请参阅这个教程，虽然是为树莓派编写的，但也可以在其他 Linux 设备上使用。

你也可以选择以下方式之一：

- 安装并使用 rtaudio
- 使用网络接收器，尤其是在你发送的程序可以使用 UDP/TCP 时。

程序

以下是一些可以与SDR++ 和虚拟音频线一起使用的应用程序。在RTL-SDR博客网站上还有更详细的列表。

带有 DSD+ 的 DMR

DMR是一种数字语音模式，在VHF和UHF（通过NFM信号）上被商业对讲机和无线电爱好者使用。商业用户可以包括学校和大学、店铺巡逻网络、建筑工地、电车和公交运营商。

DSD+ 可以接收音频信号并将DMR数据解码为语音输出，还可以检测使用的颜色码和频道。免费版本的DSD+ 还可以解码另一种数字语音模式——NXDN。免费版本功能完整，而付费版本 DSD+ Fastlane功能更多，可以作为独立的SDR程序运行，无需使用SDR++。

下载 DSD+ 或订阅 DSD+ Fastlane。

FT8 及其他使用 WSJT-X 的业余数字模式

WSJT-X 使您能够解码多种数据模式，包括 FST4、FST4W、FT4、FT8、JT4、JT9、JT65、Q65、MSK144、WSPR 以及 Echo（地-月-地信号）。大多数使用上边带。所有这些都是弱信号模式，并且在正确的设备情况下可以传输很长的距离。

See the Rig Control section above for details on how to control the frequency SDR++ receives with WSJTX. For more help there are user guides on the download link below, and YouTube has videos on configuring and using WSJT-X.

Download WSJT-X

Other digital modes

There are a number of applications that can decode a wide variety of text based data modes from the audio signal from SDR++: including RTTY (radioteletype), CW (morse code), fax and SSTV (slow scan 'TV' - actually color pictures).

Fldigi is an open source program and has a smaller but equally impressive list of modes that it can decode.

Download Fldigi

MultiPSK - there's hardly a mode that this doesn't decode - all of the above plus DStar, DMR, ACARS, SITOR, Navtex, Hellschreiber, POCSAG, ADS-B, ACARS and more. It's quite a complex application for beginners, and is designed to allow radio amateurs to transmit as well as receive in some of those modes.

Download Multipsk

Decode pagers

Multimon-ng - is a command line tool to decode free, open source application to monitor POCSAG and FLEX - used by pagers on VHF and UHF. You can often find a version in the repositories for your Linux distro, or you can build Multimon-ng from source. It's light enough to work on a Raspberry Pi. It's also available for Windows. You can find tutorials on using it online.

Download Multimon-ng for windows

有关如何使用 WSJTX 控制 SDR++ 接收的频率的详细信息，请参见上面的“设备控制”部分。如需更多帮助，可以查看下面下载链接中的用户指南，YouTube 上也有关于配置和使用 WSJT-X 的视频。

下载 WSJT-X

其他数字模式

有许多应用程序可以从SDR的音频信号中解码各种基于文本的数据模式，包括RTTY（无线电电传打字）、CW（摩尔斯电码）、传真和SSTV（慢扫描“电视”——实际上是彩色图片）。

Fldigi 是一个开源程序，支持解码的模式数量虽然较少，但同样令人印象深刻。

下载 Fldigi

MultiPSK——几乎没有它不能解码的模式——以上所有模式加上DStar、DMR、ACARS、SITOR、Navtex、Hellschreiber、POCSAG、ADS-B、ACARS等。对于初学者来说，这是一个相当复杂的应用程序，并且旨在让业余无线电爱好者在其中一些模式下不仅接收信息，还能发送信息。

下载 Multipsk

解码寻呼机

Multimon-ng 是一个命令行工具，用于解码 POCSAG 和 FLEX 的免费开源应用程序，这些协议用于 VHF 和 UHF 上的寻呼机。你通常可以在你的 Linux 发行版的仓库中找到一个版本，或者你可以从源代码构建 Multimon-ng。它足够轻量，可以在 Raspberry Pi 上运行。它也可用于 Windows。你可以在网上找到使用它的教程。

下载适用于 Windows 的 Multimon-ng

PDW is also an open source monitor for POCSAG and FLEX as well as ACARS: the data transmissions from aircraft on the AM VHF airband. It uses a graphical interface and is easy to use.

Download PDW

Optional built modules

There are unofficial external modules for SDR++ that are not included in the installation package but can be built by advanced users.

Not shown here are modules for

- Demodulating unencrypted Tetra transmissions
- Controlling SDR++ using an Arduino
- Inmarsat C demodulator

Radiosonde decoder plugin

You can include this module when building SDR++ from source in Linux. Like other community plugins, its development is not currently supported by the SDR++ team. The plugin takes sensor data broadcast from radiosondes such as weather balloons and plots the height and location.

You can obtain the instructions and source code from:
https://github.com/dbdexter-dev/sdrpp_radiosonde (*Photo courtesy of Davide Belloli)*

PDW 也是一款开源监控工具，可用于 POCSAG、FLEX 以及 ACARS（飞机在 AM VHF 空中频段上的数据传输）。它使用图形界面，操作简便。

下载 PDW

可选的内置模块

SDR++ 有一些非官方的外部模块，这些模块不包含在安装包中，但高级用户可以自行构建。

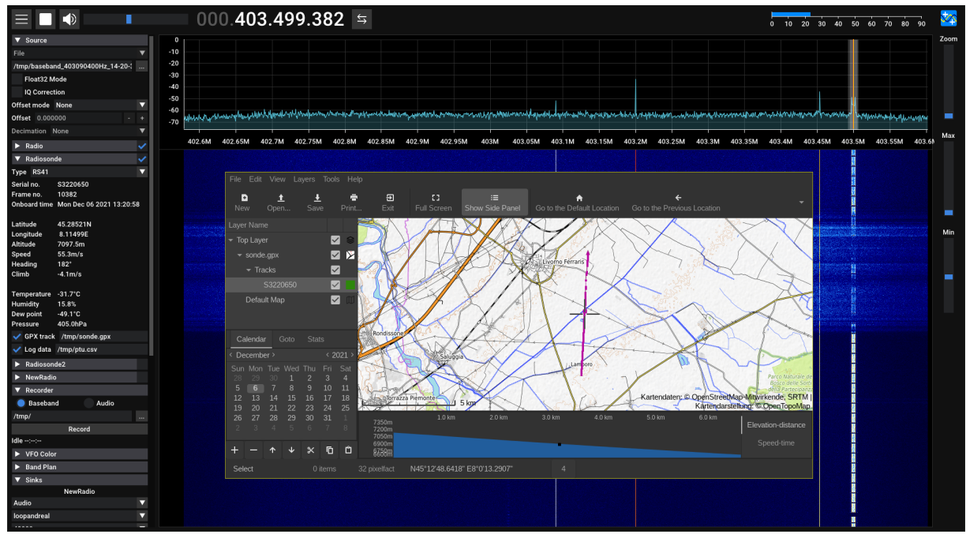
此处未显示的模块包括

- 解调未加密的Tetra传输
- 使用 Arduino 控制 SDR++
- 英迈萨特 C 解调器

无线电探空仪解码器插件

在 Linux 中从源代码构建 SDR++ 时，您可以包含此模块。像其他社区插件一样，它的开发目前不受 SDR++ 团队支持。该插件接收无线探空仪（如气象气球）广播的传感器数据，并绘制高度和位置。

您可以从以下网址获取说明和源代码：
https://github.com/dbdexter-dev/sdrpp_radiosonde （照片由Davide Belloli提供）



Troubleshooting

SDR++ works without problems for the vast majority of users. Reading this manual can help you avoid getting a problem. But software defined radio can be complicated, especially for people new to it, and there are a lot of settings to get wrong! This is a collection of questions and answers about the problems some users experience, that we hope you find helpful.

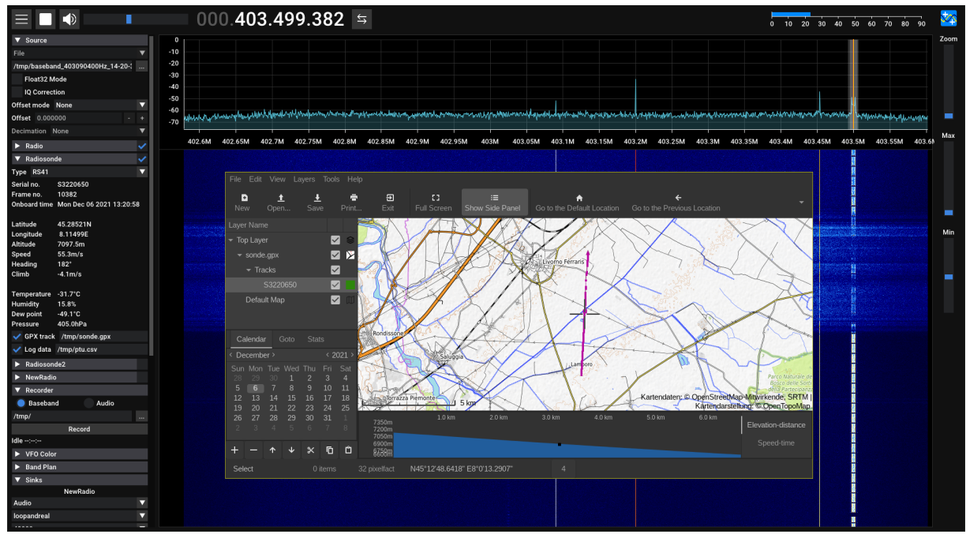
You can find further troubleshooting information in the SDR++ readme file and get help from other users on the SDR++ Discord Server help channels.

Audio issues

- I have no audio
- The sound from SDR++ 'stutters' and breaks up when I'm listening to a station
- I'm having problems with PipeWire in Linux

Look and feel

- The controls are too big/small on my device



故障排除

对于绝大多数用户来说，SDR++ 可以正常工作。阅读本手册可以帮助您避免遇到问题。但是，对于新手来说，软件定义无线电可能比较复杂，并且有很多设置可能会出错！这是关于一些用户遇到问题的问答集合，希望对您有所帮助。

您可以在 SDR++ 说明文件中找到更多故障排除信息，并在 SDR++ Discord 服务器的帮助频道获得其他用户的帮助。

音频问题

- 我没有音频
- 当我收听某个电台时，SDR++ 的声音会“断断续续”并出现中断
- 我在 Linux 上使用 PipeWire 时遇到问题

外观与感觉

- 我的设备上的控件太大/太小

Receiving

- I have audio but no signal where there should be one?
- There are signals shown in the FFT or waterfall but my VFO doesn't move to the peak when I tap/click to tune it?
- I have signals (above 14MHz) that shouldn't be there

Hardware and configuration

- SDR++ doesn't list my SDRPlay receiver as a source
- I'm having problems with SDR++ in Windows
- SDR++ won't run, and I only see a window flash for an instant
- How do I reset the SDR++ configuration back to when it was installed?
- My SDR++ has gone back to default configuration and lost its settings and/or bookmarks
- I'm worried that I will lose my stored frequency bookmarks when I update SDR++
- When I try to run SDR++ after installing it in Windows, it says vcruntime140.dll and msvcp140.dll are missing
- SDR++ crashes when starting a HackRF

I have no audio

Check:

- Is your SDR receiver connected and working?
- Have you pressed the 'Start' button?
- Is the volume slider turned up enough
- Is the AGC or gain set appropriately?
- Is the box ticked to enable your 'Radio' module or whichever VFO you are using?
- Do you have a 'sink' for the Radio/VFO you are using? Is this set to 'audio' and to the correct output e.g. the soundcard and not to a virtual audio cable?
- Is your computer audio control muted or set to the correct volume level?

The sound from SDR++ 'stutters' and breaks up when I'm listening to a station

This could be caused by a number of things:

接收

- 我有音频，但在应该有信号的地方没有信号?
- 在 FFT 或瀑布图中可以看到信号，但当我轻触/点击调谐时，我的 VFO 并不会移动到峰值位置?
- 我有不应该存在的信号（高于14MHz）

硬件和配置

- SDR++ 没有将我的SDRPlay接收器列为源
- 我在 Windows 上使用 SDR++ 时遇到问题
- SDR++ 无法运行，我只看到窗口闪了一下
- 我如何将SDR++ 配置重置回安装时的状态?
- 我的SDR++ 已恢复为默认配置，并丢失了其设置和/或书签
- 我担心在更新 SDR++ 时会丢失已存储的频率书签
- 当我在 Windows 上安装 SDR++ 后尝试运行它时，系统提示缺少 vcruntime140.dll 和 msvcp140.dll
- 启动 HackRF 时 SDR++ 崩溃

我没有音频

检查：

- 您的SDR接收器是否已连接并正常工作?
- 你按下“开始”按钮了吗?
- 音量滑块是否已调高
- 自动增益控制（AGC）或增益设置是否适当?
- 是否勾选了启用您的“收音机”模块或您正在使用的VFO?
- 你有为你正在使用的收音机/VFO 设置“接收端”吗？它是否设置为“音频”并指向正确的输出，例如声卡，而不是虚拟音频线?
- 你的电脑音频是否静音或设置到正确的音量?

所以 当我听一个电台时，来自 SDR++ 的“断断续续”会卡顿和中断 n

这可能是由多种原因引起的：

- The resources on your device - like CPU usage - could be close to maximum. Try monitoring what the CPU usage is with Windows Task Manager - use CTRL+ALT+delete to start it. There are equivalents in other operating systems. Close down other programs or apps.
- You might be able to cure it by altering the bit rate in the 'Sink' for the VFO you are using. Try altering it to a different value like 48000 or 44100.
- Check you haven't got the squelch on.
- Check whether it does this for other transmissions - it might be the particular transmission and not SDR++.

In Debian based linux distros you'll usually need to apply this fix.

I'm having problems with PipeWire in Linux

Some people experience their audio freezing or artifacting when using PipeWire for audio on Linux distributions. If you run

```
export PIPEWIRE_NOJACK=1
```

before opening SDR++, then this may stop this. Alternatively you can disable the Jack client altogether and use the Pulse audio layer only.

The controls are too big/small on my device

Use the High-DPI scaling control to adjust the settings so that these are a convenient size for you. You will need to restart SDR++ after adjusting this. On Android devices, if you choose a scale that makes the display too small, so it is impossible to navigate to the DPI scaling setting, then attach a mouse through an OTG cable attached to your device.

I have audio on my device/computer but no signal where there should be one?

Not all stations broadcast continuously - tune to a known continuous broadcast, for instance a local broadcast station or a 24 hour VOLMET station. If there is still no signal check the following:

- Is your antenna connected?
- Is the AGC or gain set appropriately?

● 你设备上的资源——例如 CPU 使用率——可能接近最大值。尝试使用 Windows 任务管理器监控 CPU 使用率——按下 CTRL+ALT+DELETE 启动它。在其他操作系统中也有类似的工具。关闭其他程序或应用程序。

● 你可以尝试通过更改你正在使用的 VFO 的“Sink”中的比特率来解决此问题。尝试将其更改为不同的值，例如 48000 或 44100。

● 检查你是否没有打开静噪功能。

● 检查它是否对其他传输也会这样——可能是特定的传输问题，而不是 SDR++ 的问题。

在基于 Debian 的 Linux 发行版中，你通常需要应用此修复。

我在 Linux 中使用 PipeWire 时遇到了问题

在使用 PipeWire 在 Linux 发行版上播放音频时，有些人会遇到音频冻结或出现瑕疵。如果你运行

导出 PIPEWIRE_NOJACK=1

在打开 SDR++ 之前，这可能会阻止此问题。或者，你可以完全禁用 Jack 客户端，仅使用 Pulse 音频层。

我的设备上的控件太大/太小

使用高DPI缩放控制来调整设置，使其尺寸适合您。调整后，您需要重新启动SDR++ 。在安卓设备上，如果您选择的缩放比例使显示过小，从而无法导航到DPI缩放设置，请通过OTG线将鼠标连接到您的设备上。

我的设备/电脑上有音频，但在本应有信号的地方没有信号?

并非所有电台都持续广播——请调至已知的持续广播电台，例如本地广播电台或全天候的VOLMET电台。如果仍然没有信号，请检查以下内容：

● 你的天线连接好了吗?

● 自动增益控制（AGC）或增益设置是否适当?

- If you have an RTL-SDR dongle is direct sampling set to Q-branch (for stations below 28 MHz) or disabled (for VHF and UHF transmissions)?
- Have you selected the appropriate modulation e.g. AM or FM or lower/upper sideband etc.?
- Is your audio sink set to output to speakers?

There are signals shown in the FFT or waterfall but my VFO doesn't move to the peak when I tap/click to tune it?

Change the snap interval to a smaller one so you can use smaller tuning steps.

I have signals above 14MHz that shouldn't be there?

This is a known issue with RTL-SDR dongles operating in direct sampling mode. They repeat or 'mirror' strong stations from below 14MHz so they are still received above that frequency. This can also reverse the sideband on SSB signals, so they are received on lower sideband in the 'image', when the original on the correct frequency is on upper sideband.

You could fit a high pass filter to cut out signals below 14 MHz, upgrade to an SDR receiver with better filtering, or buy an upconverter like the 'Ham it up'..

Alternatively, your receiver might be overloaded by strong signals from local stations, particularly from FM and AM broadcast stations. You can buy special filters to reduce these. SDRPlay receivers have these filters built in and they can be selected in SDR++.

SDR++ doesn't list my SDRPlay receiver as a source

The SDRPlay service has to be installed and running for SDRPlay devices to be picked up when SDR++ loads. If this happens:

In Windows

1. Ensure that you have SDRPlay selected in the Source menu.
2. Wait a few moments after SDR++ starts, then press 'refresh' in the Source menu to see if the receiver serial number appears. If that doesn't work...
3. Close SDR++

- 如果您有 RTL-SDR 加密狗，直接采样是设置为 Q 支路（针对 28 MHz 以下的台站）还是禁用（针对 VHF 和 UHF 传输）？
- 您是否选择了适当的调制方式，例如 AM、FM 或下/上边带等?
- 你的音频输出设备是否设置为扬声器?

FFT 或瀑布图上显示有信号，但当我点击/轻触调谐时，我的 VFO 并不会移动到峰值位置?

将快照间隔改为更小的，这样你就可以使用更小的调节步骤。

我有高于14MHz的信号，本不应该出现这些信号吗?

这是使用直接采样模式操作的RTL-SDR加密狗已知的问题。它们会重复或“镜像”14MHz以下的强信号，因此这些信号在该频率以上仍然能够接收到。这也可能会反转SSB信号的边带，因此它们会在“镜像”中以下边带接收，而在正确频率上的原始信号则为上边带。

你可以安装一个高通滤波器来切除 14 MHz 以下的信号，升级到具有更好滤波功能的 SDR 接收机，或者购买像 'Ham it up' 这样的上变频器。

或者，你的接收器可能会因为来自本地台的强信号而过载，尤其是来自调频（FM）和调幅（AM）广播电台的信号。你可以购买专用滤波器来减轻这些干扰。SDRPlay 接收器内置了这些滤波器，并且可以在 SDR++ 中选择使用。

SDR++ 没有将我的SDRPlay接收器列为源

必须安装并运行 SDRPlay 服务，才能在 SDR++ 加载时检测到 SDRPlay 设备。如果发生这种情况：

在 Windows 中

1. 确保在“源”菜单中选择了 SDRPlay。 2. 在 SDR++ 启动后稍等片刻，然后在“源”菜单中按“刷新”以查看接收机序列号是否出现。如果仍然无效... 3. 关闭 SDR++

4. Ensure you have the SDRPlay drivers installed (see Appendix 1 for links to download these) then either
5. Start SDRUno, then close it down - this will start the SDRPlay service

OR

6. In the Windows search box type ‘Services’ and open the App.
7. Find ‘SDRPlay API Service’ in the list. Right click on this and select ‘Properties’.
8. In the Properties box
    a. change ‘Startup type’ to automatic and
    b. if ‘service status’ says it is not running, select ‘Start’.

Your SDRPlay receiver should now be found when you restart SDR++.

In Linux, follow the instructions and videos for installation on the SDRPlay website. This involves both the installation of both the API and Soapy SDR .

I’m having problems with SDR++ in Windows

You can begin to analyse problems with less obvious solutions by starting SDR++ from the command line and examining the messages when SDR++ starts for the problem.

- In Windows File Explorer, navigate to the folder where you have installed SDR++
- Right click on the folder icon and select ‘Open in Terminal’
- start SDR++ in logging mode `sdrpp.exe -c`

In Linux you can simply open a terminal and type the `sdrpp` command and the messages will be shown in the terminal window.

Look through the messages on screen for any obvious issues, such as modules that fail to load

SDR++ won't run and I only see a window flash for an instant

SDR++ uses Open-GL for it’s display. All modern graphics drivers should support this. Update your graphics card drivers to the most up to date version.

How do I reset the SDR++ configuration back to when it was installed?

4. 确保您已安装 SDRPlay 驱动程序（有关下载链接，请参见附录 1），然后选择以下任一操作

5. 启动SDRUno，然后关闭它——这将启动SDRPlay服务

OR

6. 在 Windows 搜索框中输入“服务”，然后打开该应用程序。 7. 在列表中找到“SDRPlay API 服务”。右键点击它并选择“属性”。 8. 在属性框中： a. 将“启动类型”更改为自动； b. 如果“服务状态”显示未运行，请选择“启动”。

当您重新启动 SDR++ 时，您的 SDRPlay 接收器现在应该可以被发现。

在 Linux 上，请按照 SDRPlay 网站上的说明和视频进行安装。这包括 API 和 Soapy SDR 的安装。

我在 Windows 上使用 SDR++ 时遇到问题

您可以通过从命令行启动 SDR++ 并在 SDR++ 为问题启动时检查消息，开始分析那些解决方案不那么明显的问题。

- 在 Windows 文件资源管理器中，导航到你已安装 SDR++ 的文件夹
- 右键点击文件夹图标，然后选择“在终端中打开”
- 以日志模式启动 SDR++ sdrpp.exe -c

在 Linux 中，你只需打开终端并输入 sdrpp 命令，消息就会显示在终端窗口中。

查看屏幕上的消息是否有任何明显问题，例如无法加载的模块

SDR++ 无法运行，我只看到窗口闪了一下

SDR++ 使用 Open-GL 进行显示。所有现代显卡驱动都应该支持此功能。请将显卡驱动更新至最新版本。

我如何将SDR++ 配置重置回安装时的状态?

With SDR++ closed, delete any files with `config.json` in the name, that are in the directory where SDR++ is installed.

You should back up `frequency_manager_config.json` if you want to save your bookmarked frequencies.

When SDR++ restarts it will rebuild these files.

My SDR++ has gone back to default configuration and lost its settings and/or bookmarks

This can be caused if your PC or device, or the running instance of SDR++, crashes at the point that it is saving the configuration file or the frequency manager file. In those circumstances, these files can be corrupted. This is usually very rare. When SDR re-starts it sees that those files are corrupt and will overwrite them with a new default configuration.

You can either back-up those files regularly, or create regular restore points for your PC or device, to prevent loss of data.

I'm worried that I will lose my stored frequency bookmarks when I update SDR++

If you install a more up-to-date version of SDR++ in the same folder it overwrites some files from your currently installed version, but does **not** disturb the frequency manager configuration file. If you are worried you can either:

- export your frequency lists from within SDR++ to back them up, or
- Back up the `frequency_manager_config.json` file - which is in the folder in which you installed SDR++. For instance creating a copy and renaming it as something different such as `frequency_manager_old.json`

When I try to run SDR++ after installing it in Windows, it says vcruntime140.dll and msvcp140.dll are missing

These files are part of the Visual C++ Redistributable for Visual Studio and may not be present on your system. To solve this, you'll find the latest version for your system at Microsoft Visual C++ Redistributable latest supported downloads

在关闭 SDR++ 后，删除安装 SDR++ 的目录中名称包含 config.json 的任何文件。

如果你想保存已收藏的频率，你应该备份 frequency_manager_config.json。

当 SDR++ 重启时，它将重建这些文件。

我的SDR++ 已恢复到默认配置，并丢失了其设置和/或书签

如果您的电脑或设备，或者正在运行的 SDR++ 实例在保存配置文件或频率管理器文件时崩溃，就可能导致这种情况。在这些情况下，这些文件可能会损坏。这通常非常罕见。当 SDR 重新启动时，它会发现这些文件已损坏，并会用新的默认配置覆盖它们。

您可以定期备份这些文件，或者为您的电脑或设备创建定期的还原点，以防止数据丢失。

我担心在更新 SDR++ 时会丢失我存储的频率书签

如果你在相同的文件夹中安装更新版本的 SDR++，它会覆盖你当前已安装版本的一些文件，但不会影响频率管理器的配置文件。如果你担心的话，你可以选择以下方式之一：

- 从 SDR++ 内导出您的频率列表以进行备份，或者
- 备份 frequency_manager_config.json 文件——该文件位于您安装 SDR++ 的文件夹中。例如，可创建一个副本并将其重命名为 frequency_manager_old.json 等不同名称。

当我在 Windows 安装 SDR++ 后尝试运行时，它提示缺少 vcruntime140.dll 和 msvcp140.dll

这些文件是 Visual Studio 的 Visual C++ 可再发行组件的一部分，可能未安装在您的系统中。为了解决此问题，您可以在 Microsoft Visual C++ 可再发行组件最新支持下载页面找到适用于您系统的最新版本。

SDR++ crashes when starting a HackRF

If you also have the SoapySDR module loaded (not necessarily enabled), this is a bug in libhackrf. It's caused by libhackrf not checking if it's already initialised. Remove the soapy source by using module manager.

启动 HackRF 时，SDR++ 崩溃

如果你也加载了 SoapySDR 模块（不一定启用），这是 libhackrf 的一个漏洞。原因是 libhackrf 没有检查自己是否已经初始化。可以通过模块管理器删除 Soapy 源。

Appendix 1 - Links to driver downloads for SDR receivers

Here are the links for drivers for supported SDR receivers:

- RTL-SDR dongles - https://zadig.akeo.ie/# - not needed because they are built into SDR++
- SDRPlay receivers such as the RSP1, RSP1A, RSPdx - https://www.sdrplay.com/softwarehome/
- Other receivers supported by SoapySDR
- Airspy SDR receivers including the Airspy HF+ Discovery - https://airspy.com/download/
- Adalm Pluto SDR -. https://wiki.analog.com/university/tools/pluto/drivers/windows
- RFSpace devices - http://www.rfspace.com/RFSPACE/Support.html
- BladeRF - https://www.nuand.com/win_installers/
- Lime SDRs - https://www.sdr-radio.com/limesdr There's also a guide to installing the drivers at https://wiki.myriadrf.org/LimeSDR-USB_driver_installation
- HackRF - The guides on developing and use the HackRF are at https://greatscottgadgets.com/sdr/1/
- Aaronia SPECTRAN SDRs

附录 1 - SDR 接收器驱动程序下载链接

以下是支持的 SDR 接收器的驱动程序链接：

- RTL-SDR 加密狗 - https://zadig.akeo.ie/# - 不需要，因为它们已内置于 SDR++
- SDRPlay 接收器，如 RSP1、RSP1A、RSPdx - https://www.sdrplay.com/softwarehome/
- SoapySDR 支持的其他接收器
- Airspy SDR 接收器，包括 Airspy HF+ Discovery - https://airspy.com/download/
- Adalm Pluto SDR -. https://wiki.analog.com/university/tools/pluto/drivers/windows
- RFSpace 设备 - http://www.rfspace.com/RFSPACE/Support.html
- BladeRF - https://www.nuand.com/win_installers/
- Lime SDRs - https://www.sdr-radio.com/limesdr 这里还有一份安装驱动的指南：https://wiki.myriadrf.org/LimeSDR-USB_driver_installation
- HackRF - 关于开发和使用 HackRF 的指南请访问 https://greatscottgadgets.com/sdr/1/
- Aaronia SPECTRAN 软件定义无线电

Credits and SDR++ online links

This user manual is written by John Donkersley G0OXO with contributions from Alexandre Rouma ON5RYZ. SDR++ is created by Alexandre Rouma.

SDR++ Online

https://sdrpp.org

SDR++ Github page

Contribute to Ryzerth on Patreon - get early access to modules, Android versions and information.

Follow SDR++ on social media:

- Whatsthegeek on Twitter
- Ryzerth YouTube channel

SDR++ also has a Discord server you can join to discuss SDR++ and all things radio

积分和特别提款权（SDR）++ 在线链接

本用户手册由 John Donkersley G0OXO 编写，Alexandre Rouma ON5RYZ 提供了贡献。SDR++ 由 Alexandre Rouma 创建。

SDR++ 在线

https://sdrpp.org

SDR++ GitHub 页面

在 Patreon 上支持 Ryzerth - 提前获取模块、安卓版本和相关信息。

在社交媒体上关注 SDR++：

- Twitter 上的 Whatsthegeek
- Ryzerth YouTube频道

SDR++ 也有一个 Discord 服务器，你可以加入讨论 SDR++ 以及所有关于无线电的内容