

The Latest

HackRF One - 5 band beacon

Up

Posted by AG6QV Frank

Tags: [2m](#) | [GNU Radio](#) | [HackRF One](#)

Radio Beacons are used in HAM radio to test propagation. If you are able to receive a beacon on a given frequency you might also be able to make contacts with stations in the same area as the beacon, and on frequencies close by. Beacons can also be used to test antenna and receiver performances. Setting up test beacons for one or many HAM radio bands can be a large task. With HackRF One and Software Defined Radio (SDR) it can be a much easier task.

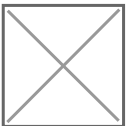
I have created a simple flowgraph that can select between 5 preconfigured frequencies on 6m, 2m, 70cm, 33cm, and 23cm. The flowgraph also includes a variable to define the beacon message. The message is a series of characters that will be translated into morse code when the program is launched.

Translation into morse code required the use of a custom block in GNU Radio Companion. Custom blocks can be written in Python and are relatively simple to get custom functionality into the flowgraph.

The image below shows the flowgraph with variables on the top and the logic at the bottom. I included a frequency sink to show where the signal is when the different frequencies are selected and to visualize the CW code.



The next image is a screenshot of the application running. The maximum output of HackRF One is 10-15dbm or 10-25mW so a small amplifier and an external antenna might help. In addition a valid HAM radio license is required to transmit any signal in the HAM bands.

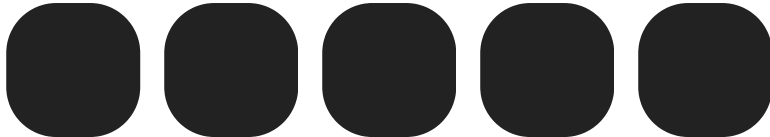


Download the [flowchart](#).

Update:

I made a new version of the [flowchart](#) for the multiband beacon. It now operates on 6 bands: 6m, 2m, 1.25m, 70cm, 33cm, 23cm. It can work on any number of bands/frequencies where each frequency is defined as an element in an array within the custom block that generates the CW code. It will automatically switch to the next band after completing a full transmission of the CW message.

Link to this Post



RackRF One Signal Generator

Up

Posted by AG6QV Frank

Tags: [GNU Radio](#) | [HAM](#) | [HackRF One](#)

Signal Generators are important tools to test HAM radio equipment. A very simple version that can generate 4 different carriers in some of the HAM radio bands is used to illustrate the possibilities of the HackRF One and other SDR systems.

First step is to set up a few variables within GNU Radio Companion. When a new project is created there is one variable provided called `samp_rate`. This is used to define the sample rates used for the DA/AD converters in the SDR. HackRF One operates with sample rates between 1 and 20 million in steps of 1 million.

In this example there are two additional variables added. The first one is to set the drive or output level. This is created as a variable called `drive` with a min and max value of 0 and 1 respectively. The UI element is a knob, but there are several other ways to control the values of any variable from the GUI. For the frequency I chose a radio button style with 4 available options, one for each band and coded the frequencies to be 50.1MHz (6m), 144.1MHz (2m), 432.100 (70cm) and 1296.1MHz (23cm). Having a simple input to enter the exact frequency is another option.



The two active component is the Constant Source used to generate the RF signal and the Soapy HackRF Sink to communicate with the hardware to generate the signal. In more advanced systems the source could be generated from audio files or a microphone input to allow transmission of a real signal.

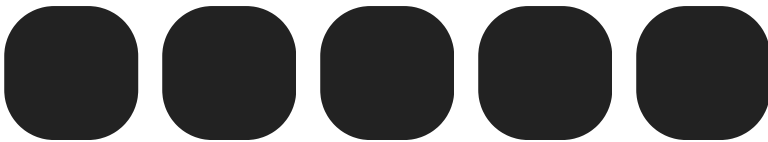
When the application is running it will generate an application as shown below:



A copy of the GNU Radio Companion project file can be downloaded [here](#).

To test the system I used my HT to listen to one of the 4 frequencies, turn up the drive until the signal was received. The maximum output power of the HackRF One device is in the range of 0-15dBm depending on the frequency selected.

Link to this Post



GNU Radio Revisited

Up

Posted by AG6QV Frank

Tags: [GNU Radio](#) | [HAM](#) | [HackRF One](#)

A long time ago I invested in a HackRF One device to play with Software Defined Radio (SDR). Over the years the available software has evolved and is evolving faster on a Linux platform compared to Windows. So when Microsoft introduced Windows Subsystem for Linux (WSL) I was exited to test it on my development system.

Unfortunately the joy was short lived as there was no support for USB devices. Today I discovered that there was a way to make it all work. This post is a description of the steps needed to install WSL, a Linux distribution, the Gnu Radio software and how to make USB devices available to the Linux environment.

In order to use the WSL system there are a couple of Windows features that must be turned on. Change these will require the computer to be rebooted. As shown in the two screenshots below the Hyper-V feature and Windows Subsystem for Windows must be enabled.



When everything is installed and the system rebooted it's time to install the Linux distribution. There are several available to choose from. The list of distributions can be viewed by opening a Windows Terminal and executing the command 'wsl --list --online'. If you already have an older version of WSL installed it will be a good idea to run 'wsl --update' to insure the latest version is installed.



Use the command 'wsl --install -d Ubuntu-24.04' to install the latest version of Ubuntu. It is possible to install multiple distributions, and even to run them side by side. Using the wsl command will start a Linux terminal using the default distribution. I like to use a program called [MobaXterm](#) to interact with the Linux distributions on my system or with remote servers. The program comes with an X server that allows executing graphical applications and it know about all the WSL distributions that exists on the system.

Microsoft has contributed the the open source project 'usbipd' and created a special version called usbipd-win. Installing or updating this toll will allow the installed USB devices to be shared across to a linux distribution. Execute the following comman in a Windows terminal to install the tool:

```
winget install --interactive --exact dorssel.usbipd-win
```

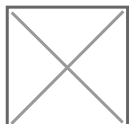
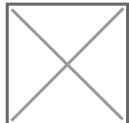
When the tool is installed use the usbipd command to list the available USB devises. Note the BUSID for the device you would like to make available to the Linux distribution.

```
usbipd list
```

On my system the Hack RF One device has BUSID 2-2 and the following command will make the device available in the Ubuntu environment.

```
usbipd attach --wsl --busid 2-2
```

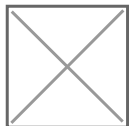
The two images below shows MobaXterm with the list of available distributions and the terminal opened.



In order to view the available USB devices in the Linux Terminal it is necessary to install some tools called 'usbutils'. This package is installed by the following command:

```
sudo apt install usbutils
```

After installation the lsusb command can be used to list the available devices. The image below shows the list of devices before and after attaching the the HACK RF One device as one of the available devices in the Linux environment.



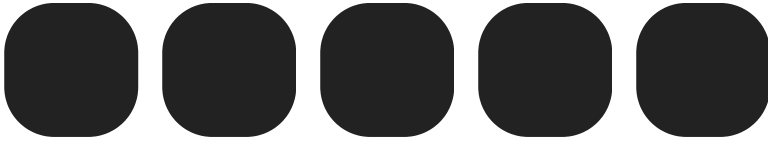
Installing the GNU Radio packages and the required GTK packages can be done with the command shown below.

```
sudo apt install gnuradio  
sudo apt-get install libgtk-4-1 libgtk-4-dev  
sudo apt install python3-gi gobject-introspection gir1.2-gtk-3.0
```

With everything installed the system is ready to use. I now have some work to do to update the projects I created when I first started to use GNU Radio. There has been many changes to the available objects. In the coming posts I'll share some of the projects I have created for signal generators, spectrum analyzers and other radio

related items.

Link to this Post



[Previous 1](#) [Get Next 1](#)

[Get RSS feed](#)

Get notified via email when new posts are published.

Sign Up

Recent Blog Posts

Blog Archives

[April 2025 {1}](#)

[March 2025 {1}](#)

[January 2025 {2}](#)

[October 2024 {5}](#)

[March 2024 {1}](#)

[August 2023 {1}](#)

[May 2023 {1}](#)

[April 2023 {1}](#)

[March 2023 {1}](#)

Tags

10 GHz {3}

2m {3}

GNU Radio {5}

HackRF One {4}

HAM {7}

HF {2}

PNW Microwave {2}

X-Band {1}

Calendar

May 2025						
Su	Mo	Tu	We	Th	Fr	Sa
				1	2	
		3				
4	5	6	7	8	9	
		10				
11	12	13	14	15	16	
		17				
18	19	20	21	22	23	
		24				
25	26	27	28	29	30	
		31				