

# Step by Step Document on Transmitting and Receiving Modulated Waveform from HackRF One to RTL-2832

## Contents

Abstract:.....	2
Section 1:.....	2
1. Apparatus Used for This Report:.....	2
2. Installation: .....	2
2.1. Gnu Radio Companion. ....	3
2.2. HackRF One. ....	3
2.3. RTL-SDR Driver: .....	3
Section 2:.....	6
3. Theoretical Concepts .....	6
3.1. Amplitude Shift Keying:.....	7
3.1.1. ASK Modulator .....	7
3.1.2. ASK Demodulator .....	8
Section 3:.....	9
4. Implementation of ASK Using GRC: .....	9
4.1. ASK Transmitter Flow Graph:.....	10
4.2. Step 1: .....	10
4.3. Vector Source:.....	10
4.3.1. Parameters.....	10
4.4. Step 2: .....	11
4.5. Repeat: .....	11
4.5.1. Parameters.....	11
Example Flowgraph.....	11
4.6. Step 3: .....	11
4.7. Osmocon Sink: .....	12
4.8. ASK Receiver Flow Graph:.....	13

**Abstract:**

In this report, I represent the basic idea about the Software Defined Radio (SDR) and have demonstrated the fundamental digital modulation schemes like Amplitude Shift Keying (ASK) using GNU Radio Companion. Gnu Radio can also be used in real time communication with the help HackRF One or RTLSDR which are composed of FPGA. Combination of GNU Radio Companion and HackRF One can be used to replace the traditional hardware radios.

I will separate the report into three parts.

1. The first part will cover installation of the tools required with the given hardware.
2. The second part will cover theoretical aspects of the task-in-hand.
3. The last part will cover the Implementation of the Task in GNU Radio Companion.

**Section 1:**

Before that we need to see what we are using to get the correct software for them.

**1. Apparatus Used for This Report:**

1. HackRF One



2. NEW GEN. RTL2832 R828D & SDR



Now, that hardware is specified. Let's move on to the installation of driver for these devices.

**2. Installation:**

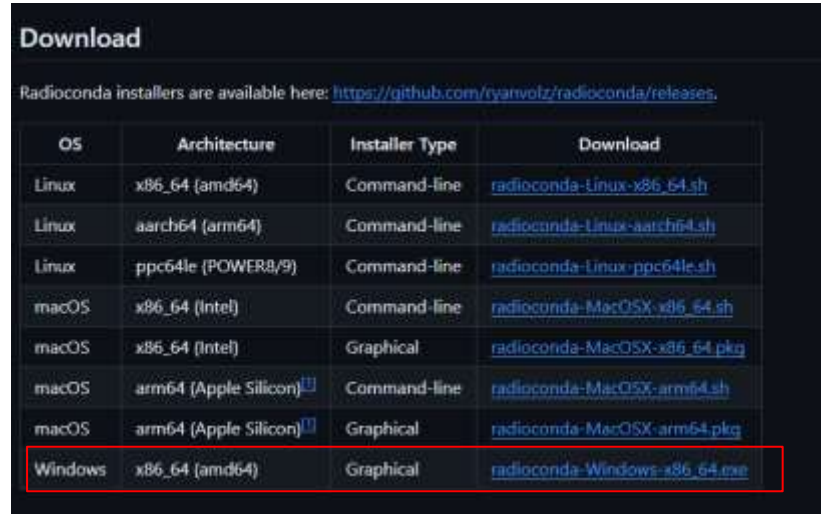
We need three things.

1. Installation of Gnu Radio Companion.
2. Drivers for HackRF One.
3. Drivers for RTL-SDR

## 2.1. Gnu Radio Companion.

To install gnu radio companion on windows 10 which is the OS I am using

- Go to the following link:→ <https://github.com/ryanvolz/radioconda>
- Scroll down until you see this



**Download**

Radioconda installers are available here: <https://github.com/ryanvolz/radioconda/releases>.

OS	Architecture	Installer Type	Download
Linux	x86_64 (amd64)	Command-line	<a href="#">radioconda-Linux-x86_64.sh</a>
Linux	aarch64 (arm64)	Command-line	<a href="#">radioconda-Linux-aarch64.sh</a>
Linux	ppc64le (POWER8/9)	Command-line	<a href="#">radioconda-Linux-ppc64le.sh</a>
macOS	x86_64 (Intel)	Command-line	<a href="#">radioconda-MacOSX-x86_64.sh</a>
macOS	x86_64 (Intel)	Graphical	<a href="#">radioconda-MacOSX-x86_64.pkg</a>
macOS	arm64 (Apple Silicon) <sup>[1]</sup>	Command-line	<a href="#">radioconda-MacOSX-arm64.sh</a>
macOS	arm64 (Apple Silicon) <sup>[1]</sup>	Graphical	<a href="#">radioconda-MacOSX-arm64.pkg</a>
Windows	x86_64 (amd64)	Graphical	<a href="#">radioconda-Windows-x86_64.exe</a>

• *Figure 1: Download options for gnu radio companion*

- Click the highlighted from Figure 1 to download it for windows.
- After it is downloaded click on the exe file to install the GRC.

## 2.2. HackRF One.

Now, there is no need to install any drivers for HackRF One. It automatically installs the driver for it by itself. It is plug and play system.

## 2.3. RTL-SDR Driver:

Last Thing, we need is the drivers for RTL2832. To Install drivers for this device following steps are required.

1. To install drivers on the device, we need Zadig.
2. Which can be downloaded by using the following link.
3. <https://zadig.akeo.ie/>.
4. Go to Link and navigate to Download and download the Latest Version by clicking onn the version as shown in the Figure Below.



Figure 2: Download for the Zadig.

- Now, get the Drivers using this link.
- <https://github.com/rtlsdrblog/rtl-sdr-blog/releases/latest/download/Release.zip>
- After getting the drivers. Follow these steps.

1. **Double click on install-rtlsdr.bat** from within the extracted folder. On some versions of Windows, you may get a SmartScreen warning. Click on More Info, then Run Anyway. This will start a command prompt that will download all the drivers required to make GRC work with RTL-SDR. Once completed, press any key to close the command prompt.

If the batch file ran successfully the files rtlsdr.dll and zadig.exe will be downloaded into the SDR# directory. If they were not downloaded then your PC or anti-virus solution may be misconfigured and may have trouble running batch files (Check that the folder is not read only, and not located in the Program Files directory).

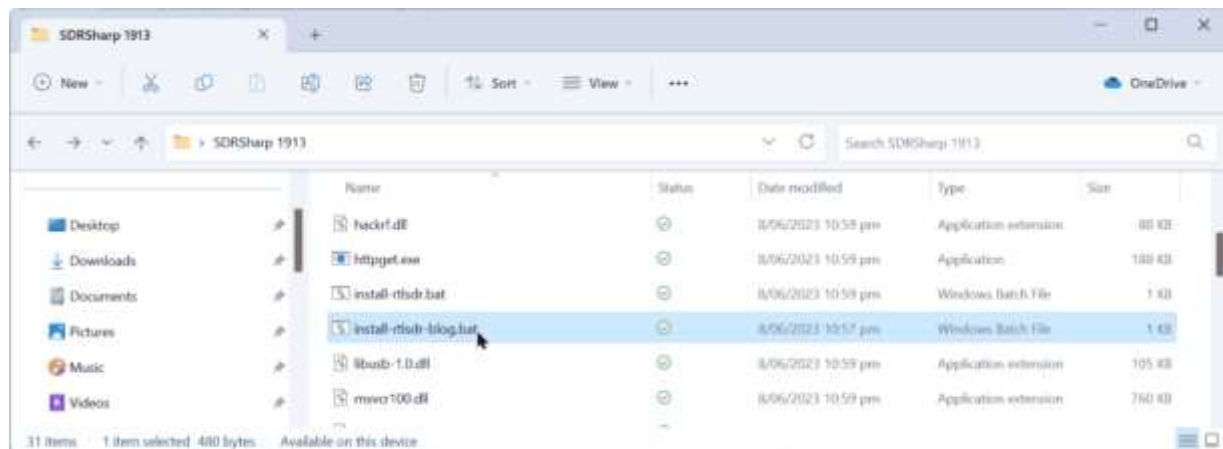


Figure 3: Extracted Driver Files

2. **Plug in your dongle.** Do not install any of the software that it came with (if any), and ensure that you wait a few seconds for plug and play to finish *attempting* to install the dongle (it will either fail or install Windows DVB-T TV drivers). If you've already installed the DVB-T drivers that came on the CD bundled with some dongles, uninstall them first.
3. Find the file called **zadig.exe**. Right click this file and select "Run as administrator".

4. In Zadig, go to "**Options->List All Devices**" and make sure this option is checked. If you are using Windows 10 or 11, in some cases you may need to also **uncheck "Ignore Hubs or Composite Parents"**.

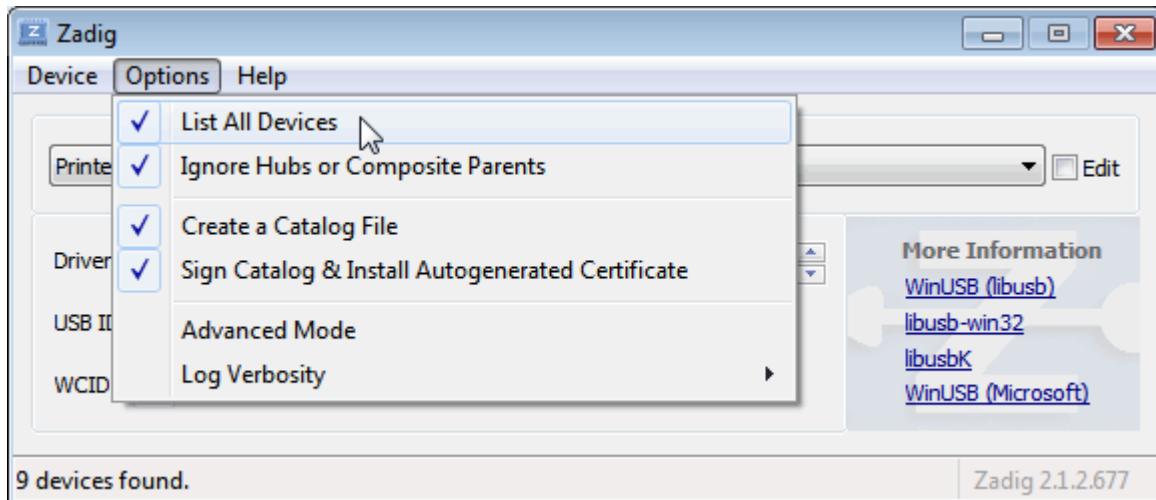


Figure 4: Zadig Options

5. Select "**Bulk-In, Interface (Interface 0)**" from the drop-down list. Make sure it is Interface 0 (ZERO), and not "1". Note on some PCs you may see something like **RTL2832UHIDIR** or **RTL2832U** or **Blog V4** instead of the bulk in interface. This is also a valid selection. Double check that USB ID shows "0BDA 2838 00" as this indicates that the dongle is selected.

**WARNING: DO NOT select *anything else* or you will overwrite that device's driver! DO NOT click around randomly in Zadig.** If you do you are likely to overwrite your mouse, keyboard, printer, soundcard etc drivers. Many bad reviews we get are due to people clicking around randomly in Zadig, so PLEASE check what you are doing first.

6. **Make sure the box to the right of the arrow shows WinUSB.** The box to the left of the green arrow is not important, and it may show **(NONE)** or **(RTL...)**. This left-hand box indicates the currently installed driver, and the box to the right the driver that will be installed after clicking Replace/Install Driver.

7.

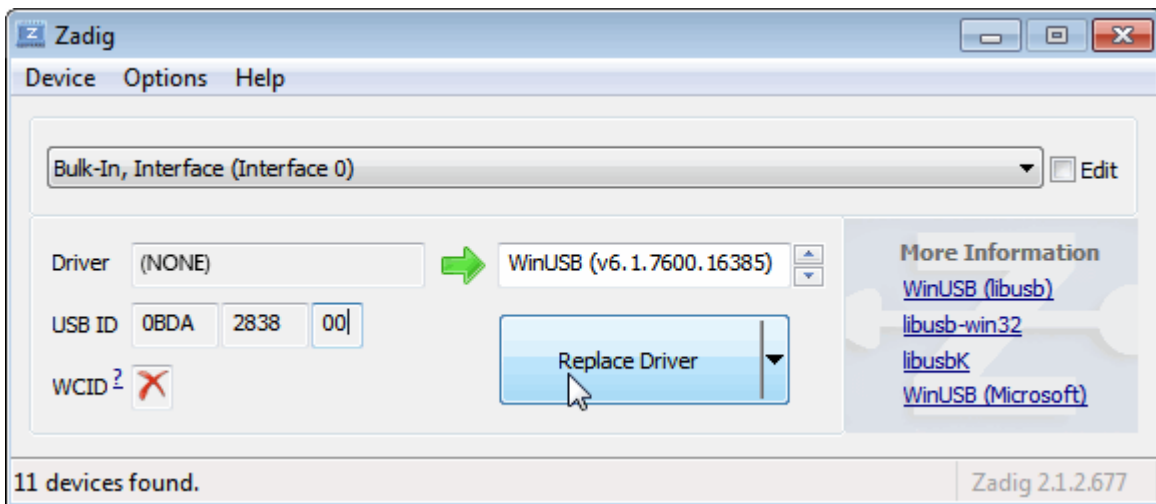


Figure 5: Zadig Choices

8. **Click Replace Driver.** On some PC's you might get a warning that the publisher cannot be verified, but just accept it by clicking on "Install this driver software anyway". This will install the drivers necessary to run the dongle as a software defined radio.

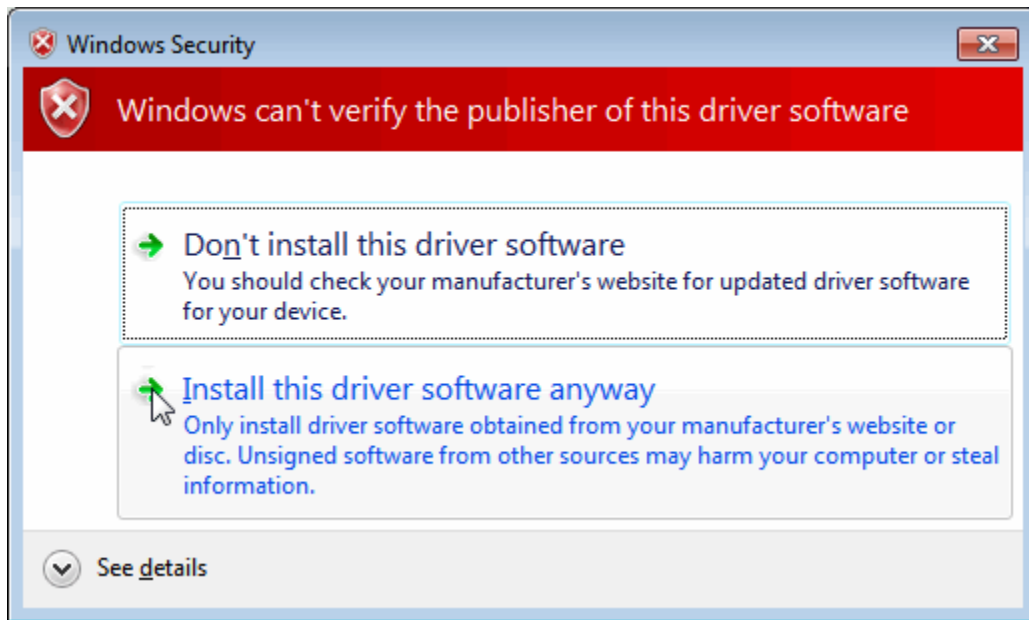


Figure 6: Bypass Security.

## Section 2:

### 3. Theoretical Concepts

The first thing is to set goal to achieve some waveform for our targets.

We want to do the following. Transmitting and Receiving:

- AM
- ASK
- FSK

Let's Understand these first:

### 3.1. Amplitude Shift Keying:

**Amplitude Shift Keying** *ASK* is a type of Amplitude Modulation which represents the binary data in the form of variations in the amplitude of a signal.

Any modulated signal has a high frequency carrier. The binary signal when ASK modulated, gives a **zero** value for **Low** input while it gives the **carrier output** for **High** input.

The following figure represents ASK modulated waveform along with its input.

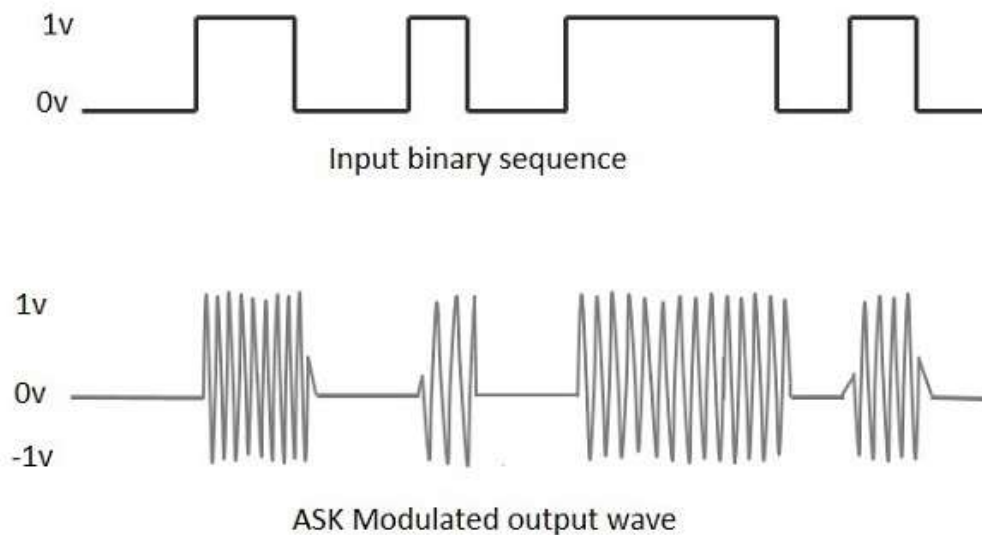


Figure 7: ASK Waveform

To find the process of obtaining this ASK modulated wave, let us learn about the working of the ASK modulator.

#### 3.1.1. ASK Modulator

The ASK modulator block diagram comprises of the carrier signal generator, the binary sequence from the message signal and the band-limited filter. Following is the block diagram of the ASK Modulator.

### ASK Generation method

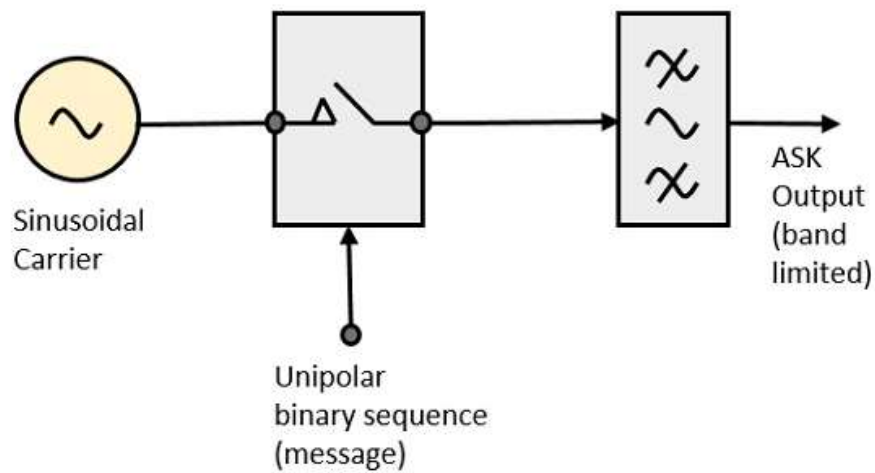


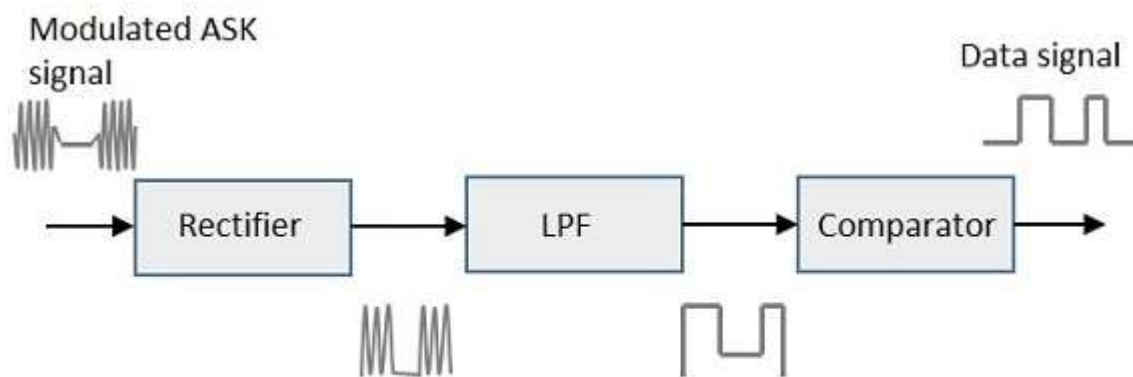
Figure 8: Architecture for Transmission

The carrier generator, sends a continuous high-frequency carrier. The binary sequence from the message signal makes the unipolar input to be either High or Low. The high signal closes the switch, allowing a carrier wave. Hence, the output will be the carrier signal at high input. When there is low input, the switch opens, allowing no voltage to appear. Hence, the output will be low.

The band-limiting filter, shapes the pulse depending upon the amplitude and phase characteristics of the band-limiting filter or the pulse-shaping filter.

#### 3.1.2. ASK Demodulator

The Asynchronous ASK detector consists of a half-wave rectifier, a low pass filter, and a comparator. Following is the block diagram for the same.



### Asynchronous ASK detector

Figure 9: Architecture for Receiver



The modulated ASK signal is given to the half-wave rectifier, which delivers a positive half output. The low pass filter suppresses the higher frequencies and gives an envelope detected output from which the comparator delivers a digital output.

### Section 3:

#### 4. Implementation of ASK Using GRC:

Let's move onto the GNU Radio Companion. Let's see the installation of gnu radio companion.

- After installing the grc. Turn on GRC.

The logo should look like this:



Click on this logo to turn on gnu radio companion.

The default screen will look like this:

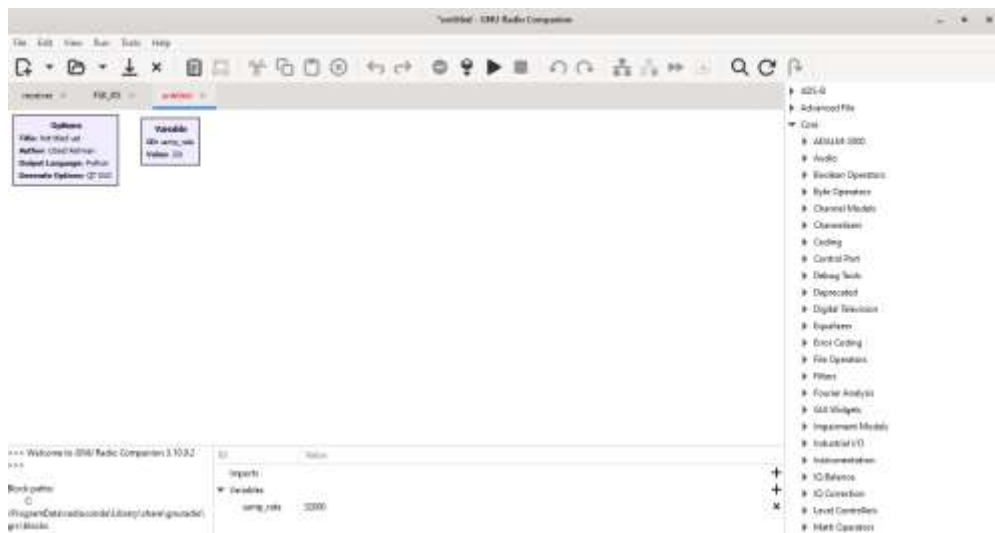


Figure 10: GNU Radio Companion Default Screen

Now to do our task for ask transmission and receiving for our purpose. Let's understand the block to help understand. This section is dedicated to understand blocks.

Now let's pick up the block for the transmission of the ask. We need to define the data we need to send through our transmitter of ask. The Flow Graph of the ASK Transmitter looks something like this

#### 4.1. ASK Transmitter Flow Graph:

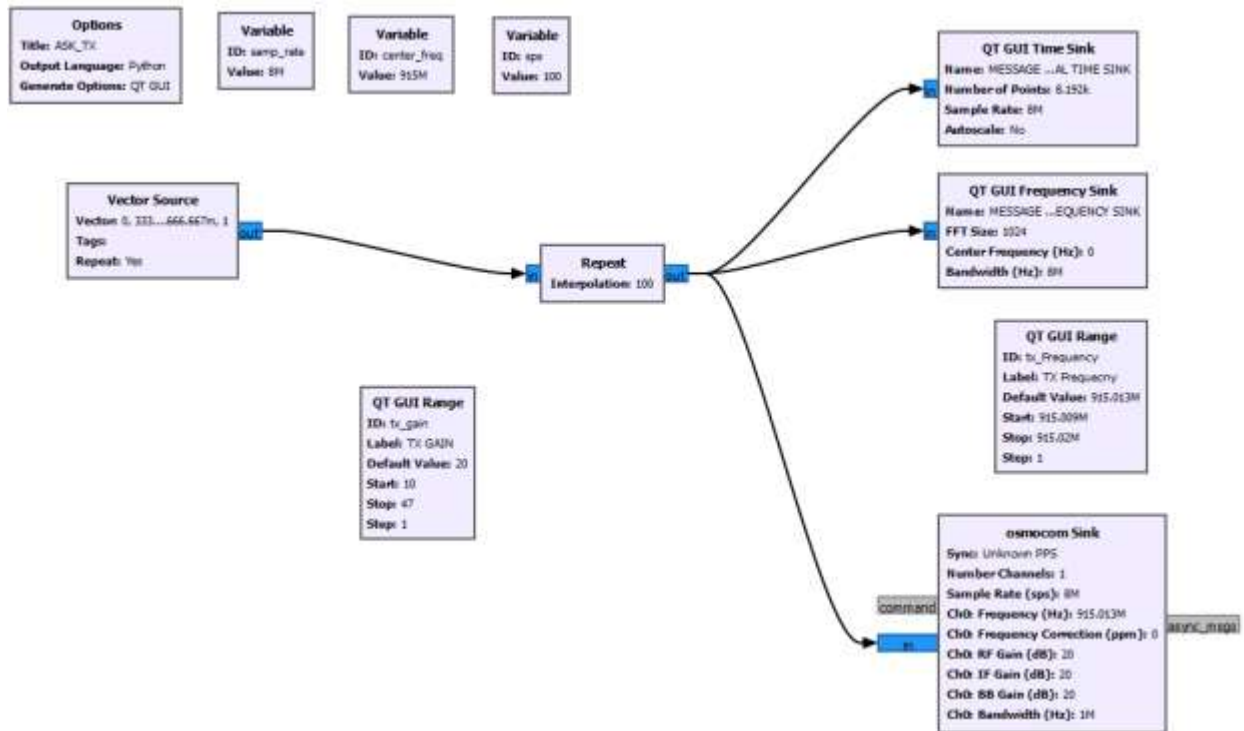


Figure 11: ASK Transmitter Flow Graph

#### 4.2. Step 1:

To do that we need the vector source input. We can get it by using Vector source block in GNU Radio Companion. It is under the category Core -> Misc -> Vector Source.

#### 4.3. Vector Source:



Figure 12: Vecctor Source Block

##### 4.3.1. Parameters

##### Output Type

Output data type of data, possible values are complex, float, int, short, byte

Default value = complex

##### Vector (*R*)

Vector to be generated

Default value = (0, 0, 0). Set it to (0, 1/3, 2/3, 1)

## Repeat

Whether or not to repeat the vector when it's done. Set it to yes for continuous signal.

## Vector Length

Length of the output vector, i.e. if set to 1 then it will output just a normal stream.

### 4.4. Step 2:

Now, we need to interpolate the signal to let it show up to receiver.

### 4.5. Repeat:

Repeat each input Interpolation times

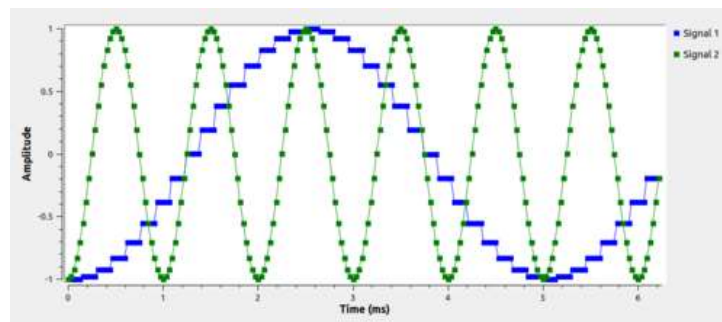
#### 4.5.1. Parameters

Interpolation ( $R$ )

Number of times to repeat the input, acting as the interpolation factor.

## Example Flowgraph

This flowgraph shows a sine wave with each value being repeated 5 times, compared to the non-repeated original.



Now, we can transmit data to the radio frequency spectrum using the osmocomb sink. We can observe the frequency sink and the time sink

We have now implemented the data to be transmitted. Now, we have the bits to be sent out in the form of packets. Each packet contains 4-bit to be transmitted.

### 4.6. Step 3:

Now, we need to actually transmit the data packets to the frequency spectrum.

We can do so by using the osmocon sink block. The osmocon block throws the data out to the radio frequency spectrum. Details of the osmocon blocks are as follows:

#### 4.7. Osmocon Sink:

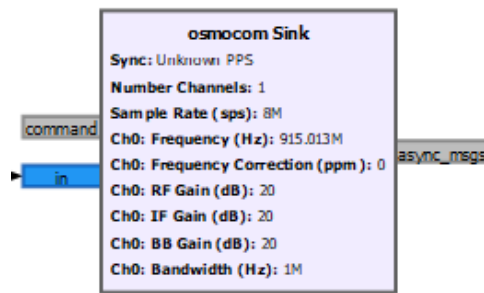


Figure 13: Osmocon sink block

##### Sample Rate:

The sample rate is the number of samples per second output by this block on each channel.

##### Frequency:

The center frequency is the frequency the RF chain is tuned to.

##### Freq. Corr.:

The frequency correction factor in parts per million (ppm). Set to 0 if unknown.

##### RF Gain:

Overall RF gain of the device.

##### IF Gain:

Overall intermediate frequency gain of the device.

This setting is available for RTL-SDR and OsmoSDR devices with E4000 tuners and HackRF in receive and transmit mode. Observations lead to a reasonable gain range from 15 to 30dB.

##### BB Gain:

Overall baseband gain of the device.

This setting is available for HackRF in receive mode. Observations lead to a reasonable gain range from 15 to 30dB.

##### Bandwidth:

Set the bandpass filter on the radio frontend. To use the default (automatic) bandwidth filter setting, this should be zero.

After Passing through these blocks. The Waveform looks something like this. The configuration is as follows:

- **Sample Rate** = 8Mps.
- **Center Frequency** = 915Mhz.
- **Repeat Interpolation** = 100.
- **TX Gain**: Slider From 10-47 with step of 1.
- **TX Frequency**: Slider From 915.009Mhz – 915.02Mhz

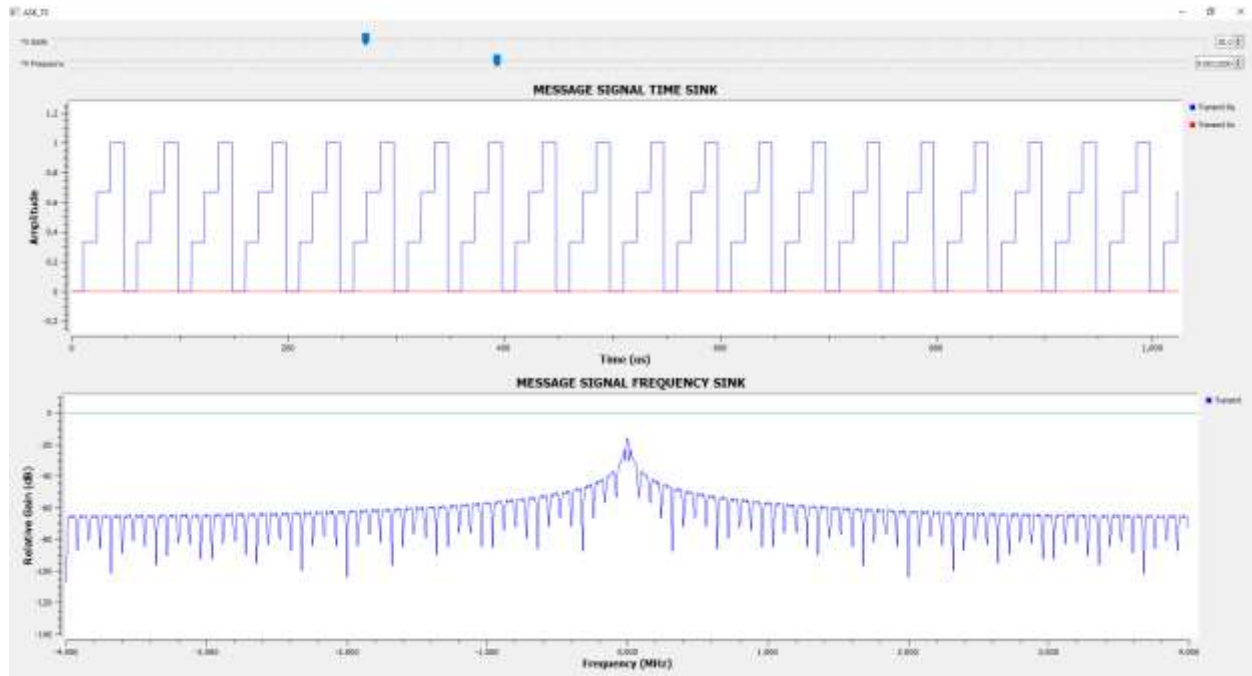


Figure 14: ASK Data Transmission

Now, that data is being transmitted from HackRF One. Its time to make a flowgraph for the demodulator receiver for RTL-SDR

#### 4.8. ASK Receiver Flow Graph:

Now, to receive the signal, I am going to use RTL-SDR, so I need to change some blocks to receive the data I want in RTL-SDR Device. Now, to receive the signal and to demodulate it. We need to know the center frequency and interpolation factor to demodulate it.

The Center Frequency is 915Mhz and interpolation factor is 100.

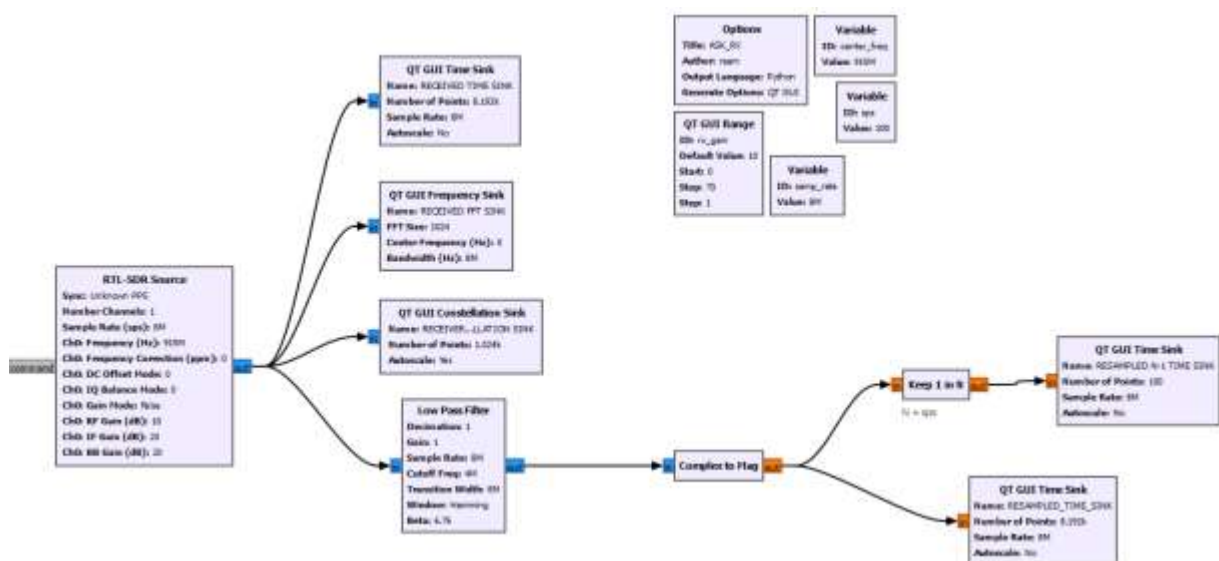


Figure 15: Flow Graph for ASK Receiver.

Now, here we are using RTL-SDR Source instead of osmocomb source because of our device. If we were using HackRF to receive we would have used osmocomb source instead. After that we are using low pass filter to take only the center frequency as the output and discarding everything else.

After that, we convert the complex output to float type. Blue Indicates Complex Data Type and Orange indicates float data type.

Keep 1 in N Blocks decimates the Interpolation Factor of 100. To receive the original data. Whereas the Downward data indicate the interpolated data.

The Output Waveform looks something like this.

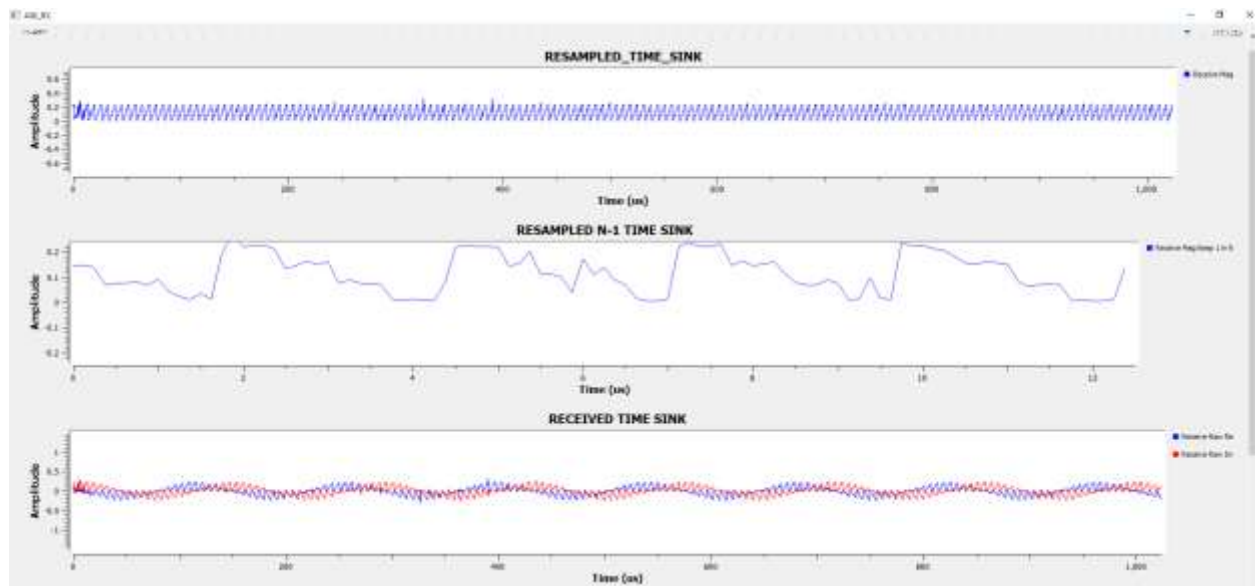


Figure 16: Demodulated Data

Here, we have completed ASK Transmission and Receiving using GRC with HackRF One and RTL-SDR.