Python Radio 28: It’s Easier than You Think

Understanding how the hardware works.

Simon Quellen Field

Simon Quellen Field

Follow

13 min read

·

Sep 25, 2024

Listen

Share

More

Press enter or click to view image in full size

Abstract art

MidJourney

Programmers are smart people. But sometimes they lack confidence when it comes to hardware issues. Here we will explain how the hardware works and why it works, in a language tailored to programmers. But first, we will build the hardware.

In the previous article, we built a repeater to get more range from our HC-12 transceivers. But we still used the inefficient little spring antennas shipped with the radios.

With those antennas, we can get about 300 meters of range instead of the 1000 meters claimed for the radios. The quarter wave ground plane antenna we built earlier would get the full kilometer (and a bit more).

We chose the ground plane antenna over a dipole because it is easy to adjust the angle of the ground plane wires to 42 degrees to get the 50-ohm impedance match we wanted. But today we are going to build a dipole and match the impedance using a gadget called a gamma match.

For reasons I will explain shortly, we will build our antenna on a sheet of foam-core board 48 inches by 36 inches.

In garden stores, they sell gummed copper tape for keeping snails and slugs away from flower beds. This is very convenient stuff to use when building an antenna on foam core. You can solder to it easily. You can tune the antenna easily by folding back a bit of the tape at the ends. This is better than cutting it because you can undo the folding if you have gone too far.

Below is the dipole, after tuning it with the NanoVNA (vector network analyzer, a very convenient tool when building antennas, and available at Amazon.com for fifty dollars).

Press enter or click to view image in full size

Half-wave dipole.

Half-wave dipole (photo by author)

NanoVNA

NanoVNA (photo by author)

The dip in the yellow curve shows the antenna is tuned to 433.4 MHz. The shallowness of the curve shows that the antenna is wide-band, and does not have a steep resonance at 433.4. This will allow it to work well at the entire frequency range of the HC-12.

The green curve shows that we have matched the impedance to 50 ohms.

We did the impedance match using the long center conductor of the coax. By connecting it far away from the center of the dipole, we can lower the impedance from the dipole’s 72 ohms down to 30 ohms or less (should we want to go that low).

In this case, the impedance match was only a few millimeters from the center of the dipole. I soldered it in place, and the impedance jumped to 55 ohms, which is still just fine.

If you want to build this antenna and don’t have a NanoVNA (yet) the dipole ended up 26 centimeters long, and the gamma match was 3 millimeters long.

Press enter or click to view image in full size

Dipole connected to HC-12 and RP2040.

Dipole connected to HC-12 and RP2040

To connect the dipole to the HC-12 we use an SMA female to IPEX cable pigtail. These are available on AliExpress.com for 28 cents. You could just solder the antenna hole on the HC-12 to one leg of the dipole and ground to the other, but I wanted to use the NanoVNA to tune the antenna and it has an SMA input. The HC-12 has a nice tiny IPEX connector on the board.

We will use the following main.py module to send text to the repeater we built in a previous project.

From machine import Pin

From hc12 import HC12

From time import sleep

LED = Pin(25, Pin.OUT)

Def main():

Radio = HC12(1, 4, 5, 3, 1200)

Radio.long\_distance()

Radio.command(“C001”)

Radio.status()

Count = 0

While True:

Send = “Sending: “ + str(count)

Print(send)

LED(1)

Radio.write(send)

LED(0)

Sleep(5)

Count += 1

Main()

I promised earlier that I would explain why we needed so much foam-core. We are going to use it now.

Imagine you had a light bulb and you wanted to double the amount of light sent in one direction. You would put a mirror behind the light bulb.

In radio, it goes by the synonym: a reflector.

We could just put it anywhere behind the dipole, but if we put it a quarter wave away, the reflection adds to the dipole’s energy in phase, so we get the full effect.

Press enter or click to view image in full size

2-element Yagi-Uda antenna.

2-element Yagi-Uda antenna (photo by author)

The reflector should be a bit longer than the dipole. About 5% longer works well (I went crazy in the photo above). From the center of the dipole tape to the center of the reflector tape is a quarter wave. Don’t worry a lot about millimeters here, the spacing is fairly forgiving. The full sine wave is 70 centimeters, and even if the reflected wave doesn’t match up perfectly with the transmitted wave, it will still add a lot. Besides, this antenna works for almost the entire 70-centimeter band, from 424 MHz to 442 MHz, so it will only be absolutely perfect at one tiny spot, but work just fine everywhere else.

The gain of our antenna just went from about 2 to over 4. And when we receive signals, we get far less noise from behind us, so our signal-to-noise ratio almost doubled too.

This design is a 2-element Yagi-Uda antenna, named after the Japanese engineers who invented it.

Four times better than an isotropic antenna (one that transmits equally in all directions, like our light bulb) is pretty good. But we can do better.

If we put some slightly shorter conductors in front of the dipole, they act as “parasitic elements”. They absorb some of the radio energy and re-radiate it. If they are carefully spaced, they can constructively and destructively interfere with the radio waves, and act like a lens in front of the dipole.

There are many Yagi calculators on the web that will show you the proper spacing. I used one to calculate a 7-element Yagi-Uda antenna, and came up with this:

Press enter or click to view image in full size

7-element Yagi-Uda.

7-element Yagi-Uda (photo by author)

The gain is almost 10 times better than the dipole (9.88 dBd, or about 12 dBi). It performs so well, I have named it Carlos Antenna (Oye como va?).

How the hardware works

I have used several hardware words and concepts without explanation, and here is where I will catch you all up.

Impedance

Most “simple machines” you learn about in elementary physics, such as the lever, the pulley, the inclined plane, and the screw, are impedance-matching devices.

Press enter or click to view image in full size

Simple Machines

Flicker Creative Commons

An impedance-matching device trades off one thing for another. A lever lets you lift a heavy weight with little force, but the tradeoff is the distance your smaller force moves. Likewise with an inclined plane or ramp. You can carry a heavy weight up to a higher place, but you walk farther to do it.

A megaphone (the un-powered cone type) is an impedance matcher. It exchanges pressure for volume. This is most apparent when you use it backward, as in an ear trumpet, or in the toy I used to use in the swimming pool. We taped plastic wrap over the small end of a traffic cone and spoke into the large end. Swimmers underwater could hear us clearly.

When you shout at the water’s surface, most of the sound energy bounces off. There is an impedance mismatch between the sound pressure needed in air and that needed in water. Matching the impedance allows more energy to transfer to the water.

This reflection effect happens to radio waves too. If the impedance is not matched, the energy is reflected to the transmitter, sometimes so much that the output transistor overheats and dies.

A transformer is an impedance matcher. It trades off voltage and current. We measure electrical impedance in terms of ohms. Ohms describe the ratio between voltage and current. Ohm’s Law is simply that ratio. Voltage divided by current.

In direct current circuits, ohms are that simple. In radio frequencies, there are inductances (think coils of wire) and capacitors, both of which store energy, so that the voltage in a signal may lead or lag the current. Because this changes the ratio of voltage to current at any instant, it is expressed in ohms.

Voltage Standing Wave Ratio (VSWR)

Voltage standing waves.

Wikimedia Commons

When impedances are not matched, we get reflected energy. We can measure the voltage going in one direction and the voltage going in the reverse direction with a clever use of diodes.

If you tie a rope to a tree and send a pulse wave down the rope by jerking it up and down, the wave will travel to the tree and then bounce back to you. If you keep jerking the rope at a certain frequency, you will get a “standing wave” in the rope. This happens in radio a lot. Measuring the voltage in both directions gives us the voltage standing wave ratio (VSWR), a measure of how well we have matched the impedances.

If the tree was small and absorbed all of the energy in the rope, it would wave in the air and there would be no reflected wave back to us. In a radio circuit, we would see the voltage, but the current in the reverse direction would be negligible. The VSWR would be close to 1, and there would be no reflected energy.

Bandwidth

Impedance matching is a function of wavelength. This is because inductors and capacitors behave differently at different frequencies. Inductors resist changes in current, so the faster the changes happen, the more they resist. Capacitors are open circuits at low frequencies and work better at higher frequencies (the opposite of inductors).

If we plot the VSWR as we increase the frequency, there will be a dip around the resonant frequency of the antenna. We can measure the part of the dip that is below, say a VSWR of 2, and say the width of that area is the band of frequencies that we can effectively transmit using that antenna. This is bandwidth, and how it got its name.

The faster we send information, the more often the signal has to change. This is reflected in the bandwidth. It takes more radio spectrum to send high-speed data.

Antenna Tuning

There are several things we might want to optimize when building an antenna.

If we design the antenna to match the impedance of our transmitter, we can avoid using a transformer to do the impedance match for us. We saw how to do this with the ground plane antenna.

But we can also tune the antenna to the frequency we want to transmit on. A highly resonant antenna rejects frequencies outside of its narrow bandwidth. This can increase the signal-to-noise ratio, allowing for longer-range communication.

On the other hand, we might want to be able to change frequencies a lot, so we can talk or listen to different transmitters or stations. We would tune the antenna to have a wider bandwidth. The log-periodic antennas that used to be the way we all received television signals are an example. They are directional, but extremely wide band, since television channel 2 starts at 54 megahertz, while channel 13 ends up at 216 megahertz.

Radiation pattern

An isotropic antenna sends and receives from all directions, like our light bulb. Most antennas are not isotropic, since sending signals into the ground or into space is not an effective way to communicate. A dipole antenna doesn’t send or listen in the direction of the ends of the wire. If we hold it up vertically, little energy is wasted sending it into the ground or space.

Our Yagi-Uda antenna sends and listens much more in one direction than others. A three-dimensional plot of its radiation pattern looks like this:

Press enter or click to view image in full size

Yagi-Uda radiation pattern.

Yagi-Uda radiation pattern (Wikimedia Commons)

It has side lobes caused by diffraction, but most of the energy is in the big lobe in the X direction. Almost no noise comes in from behind the reflector.

Effective Area

There is an important concept in antenna theory called the “effective area”. It is the area blocked by a hypothetical sphere around an isotropic antenna.

Press enter or click to view image in full size

Antenna effective area.

Wikimedia Commons

The higher the frequency, the smaller this sphere becomes. It is an imaginary sphere, just a concept, but we know its area, and can thus know its radius (divide the area by pi). The reason this hypothetical sphere gets smaller is because the radiation resistance of an antenna is a function of frequency. The higher the frequency, the more the antenna resists putting out radio energy.

Even though we don’t have an isotropic antenna, this concept allows us to calculate the range of our communications.

Once we know the effective area, if we have the transmitter power, the antenna gain, and the distance we want to communicate, we can get the strength of the signal at the receiver.

The power of the HC-12 is 100 milliwatts.

The power falls as the square of the distance. This is the “inverse square law”, and is pretty simple to understand once you remember that the formula for the area of a sphere is 4 times pi times the radius squared. The radius in this case is the distance between the transmitter and the receiver. The power is a function of the part of the area of a sphere with that radius that is captured by the receiver’s antenna: its effective aperture.

Since we have Python at our disposal, let’s encapsulate all of that in a little program.

From math import sqrt, pi, log10

Min\_receive\_dBm = -117

Max\_transmit\_dBm = 20

Realistic\_receive\_dBm = -80

Freq = 433400000

Tx\_gain = 12

Rx\_gain = 12

Def dBm\_to\_watts(dBm):

Return 10 \*\* ((dBm – 30) / 10)

Def speed\_of\_light():

Return 299792458

Def wavelength(frequency):

Return speed\_of\_light() / frequency

Def path\_loss\_dB(distance, frequency):

Return 20 \* log10(distance) + 20 \* log10(frequency) + 20 \* log10((4 \* pi) / speed\_of\_light())

Def link\_budget(max, min):

Return max – min

Def effective\_aperture(frequency):

Return wavelength(frequency) \*\* 2 / (4 \* pi)

Def received\_power(frequency, tx\_power, tx\_gain, distance):

Return tx\_power / (4 \* pi \* distance \*\* 2) \* tx\_gain \* effective\_aperture(frequency)

Def dist(frequency, watts, tx\_gain, rx\_gain):

Return (((4 \* pi) \*\* (-1/speed\_of\_light())) / frequency) \* watts \* tx\_gain \* rx\_gain

Def nanowatts(watts):

Return watts \* 1000000000

Def picowatts(watts):

Return watts \* 1000000000000

Def femtowatts(watts):

Return watts \* 1000000000000000

Print()

Print(“Frequency:”.ljust(40, ‘ ‘), freq / 1000000, “MHz”)

Print(“Minimum receive power:”.ljust(40, ‘ ‘), picowatts(dBm\_to\_watts(realistic\_receive\_dBm)), “picowatts”)

Print(“Maximum transmit power:”.ljust(40, ‘ ‘), 1000 \* dBm\_to\_watts(max\_transmit\_dBm), “milliwatts”)

Print(“Wavelength:”.ljust(40, ‘ ‘), round(wavelength(freq) \* 100, 2), “centimeters”)

Print(“Effective aperture:”.ljust(40, ‘ ‘), round(effective\_aperture(freq) \* 10000, 2), “square centimeters”)

Print(“Received power at 1 km:”.ljust(40, ‘ ‘), round(picowatts(received\_power(freq, .1, 1, 1000)), 2), “picowatts”)

Print()

Print(“Link budget:”.ljust(40, ‘ ‘), link\_budget(max\_transmit\_dBm, realistic\_receive\_dBm), “dB”)

Z = 0

X = 0

Budget = link\_budget(max\_transmit\_dBm, realistic\_receive\_dBm)

While z < budget:

Y = 2 \*\* x

X += 1

Z = path\_loss\_dB(y, freq)

If z < budget:

Print((“Path loss at “ + str(y) + “ meters:”).ljust(40, ‘ ‘), round(z), “dB”)

Print()

Print(“With 12 dBi antennas at each end:”)

Z = 0

X = 0

Budget = link\_budget(max\_transmit\_dBm, realistic\_receive\_dBm)

While z < budget:

Y = 2 \*\* x

X += 1

Z = path\_loss\_dB(y, freq) – tx\_gain – rx\_gain

If z < budget:

Print((“Path loss at “ + str(y) + “ meters:”).ljust(40, ‘ ‘), round(z), “dB”)

The output for our HC-12 looks like this:

Frequency: 433.4 MHz

Minimum receive power: 10.0 picowatts

Maximum transmit power: 100.0 milliwatts

Wavelength: 69.17 centimeters

Effective aperture: 380.76 square centimeters

Received power at 1 km: 303.0 picowatts

Link budget: 100 dB

Path loss at 1 meters: 25 dB

Path loss at 2 meters: 31 dB

Path loss at 4 meters: 37 dB

Path loss at 8 meters: 43 dB

Path loss at 16 meters: 49 dB

Path loss at 32 meters: 55 dB

Path loss at 64 meters: 61 dB

Path loss at 128 meters: 67 dB

Path loss at 256 meters: 73 dB

Path loss at 512 meters: 79 dB

Path loss at 1024 meters: 85 dB

Path loss at 2048 meters: 91 dB

Path loss at 4096 meters: 97 dB

With 12 dBi antennas at each end:

Path loss at 1 meters: 1 dB

Path loss at 2 meters: 7 dB

Path loss at 4 meters: 13 dB

Path loss at 8 meters: 19 dB

Path loss at 16 meters: 25 dB

Path loss at 32 meters: 31 dB

Path loss at 64 meters: 37 dB

Path loss at 128 meters: 43 dB

Path loss at 256 meters: 49 dB

Path loss at 512 meters: 55 dB

Path loss at 1024 meters: 61 dB

Path loss at 2048 meters: 67 dB

Path loss at 4096 meters: 73 dB

Path loss at 8192 meters: 79 dB

Path loss at 16384 meters: 85 dB

Path loss at 32768 meters: 91 dB

Path loss at 65536 meters: 98 dB

Each 3 dB of gain is a doubling of signal strength. Our 12 dBi antenna is like a 16x telescope (doubling 4 times is 16).

To the transmitter, this looks like multiplying the output power.

To the receiver, it looks like multiplying the received signal strength, but at the same time reducing the noise level, since just like a telescope, the receiver now sees less noise-making landscape.

Our range just went from 4 kilometers to 65 kilometers.

All from a bit of foam core and some copper tape.

Radio