Removing that Center Frequency DC Spike in Gnuradio the Easy Way

By ghostop14

January, 2016

It's your first time with gnuradio and you love your hackrf. You've played with receiver software like SDRSharp and audio piping to decode your favorite signal of choice, and now you're ready to dig deeper and learn more about SDR. Everyone's talked about gnuradio, so you install it and fire up your first flowgraph. You drop in an osmocom source block and set the device to hackrf, set your sample rate, frequency, and gain then connect it to a frequency sink and hit the button to generate your flowgraph. The ease with which you just built a receiver and the excitement about the possibilities is overwhelming… you can't wait to hit play.

Then it happens. Right in the middle of your first flowgraph is this huge signal spike that you know is not the signal you want to receive, and as you change the frequency it follows you. What?!? So your first thought is you did something wrong. After all you're new to gnuradio and you're sure you're making a newbie mistake. First you make sure there really isn't a signal there. You go back to SDRSharp and there's no spike. Then you swap out your hackrf for your airspy and rtl-sdr dongle, feed that into gnuradio, and there's still no spike. What's going on? Why is my favorite SDR that I want to use doing this?

What you've stumbled on is an artifact of the way SDR radios do IQ sampling. Your first attempts at searching on the problem reveal that it's called a DC spike and it's going to appear in the raw IQ data and there's nothing you can do to stop it. So you go back to your favorite search engine because you can't be the first person to want to get rid of it and you find that folks say that you have 3 options: 1.) ignore it (yeah not happening. It's huge and right in the middle of my spectrum!) 2.) Offset tune away from it on your center frequency (which means every flowgraph I make or download I'm going to have to custom change to actually get a clean center frequency signal to make them work. There has to be a better way!), or 3.) filter it out.

Well, options 1 and 2 don't sound like great choices, how do I filter it out? Clearly software like SDRSharp and HDSDR are doing something to get rid of this center frequency spike. And if you look at the drivers/libraries for other radios, they do it too. The problem isn't unique to my hackrf. But looking around gnuradio, there isn't a block to do the same thing. Believe it or not, this turned me off from gnuradio for the longest time, since my hackrf was my favorite SDR. I mean for the frequency range, the ability to transmit, and the price point, you can't beat it. But that annoying IQ DC spike!

Now if you know gnuradio you may be thinking that there is a DC blocker module available in the standard installation, and there is an out-of-tree (OOT) gr-iqbal module you can download and install. While they work great for what they're designed to do, that's just not this problem (believe me, I tried them and they didn't do the trick and had to do some research to understand why).

If you want to jump straight to the answer, download and install https://github.com/ghostop14/gr-correctiq.git. It's the usual process for gnuradio blocks. Git clone it, mkdir build, cd build, cmake .., make, make install, ldconfig, then restart gnuradio. It's also been added as an option in the pybombs repo. Once installed, just drop the correctiq block in your flowgraph, feed it directly from your osmocom source before any other blocks, and you're off! It'll implement the same type of DC

correction algorithm that other solutions use uses and you won't have to use a multiply block or an XLATING FIR filter just to do an offset tune. The center frequency DC spike will just be gone leaving the original signal.

But if you're interested in the details, stay with me for a little more discussion to understand what the other options do and why they were the wrong choice for the problem.

In trying to deal with that spike, my journey started with thinking "well it's a DC spike. Why don't I just drop in the DC blocker and I should be good right?" No. It gets rid of the DC alright, but because of the DC on the center frequency, it basically zero's out any signal on the center frequency…. Permanently. You'll get a result that looks like the frequency plot below:
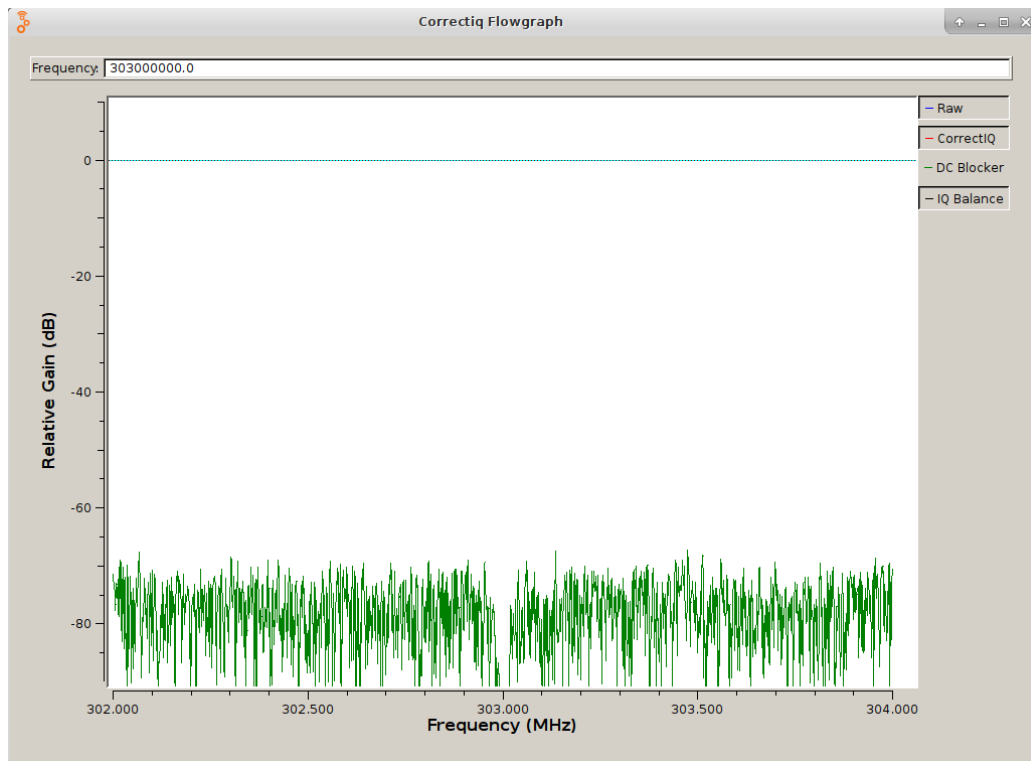
You can see the dip right at 303 MHz (the center frequency the radio is tuned to) and even when there's a signal the block is still there so you can't receive the unaltered signal. Because there's a continuous DC signal there, the DC blocker did exactly what it was supposed to do. It blocked it! Well if you want to tune your radio to the center frequency like you do with your other SDR's and have a true signal, this approach won't work. Time to look for something else.

The next approach that sounded promising was looking at the gr-iqbal module. So I downloaded it and installed it. Did some reading (yes I read the documentation on it) and it seemed straightforward enough. It does take 2 blocks for the optimization, which at first seemed unusual for what I was hoping to accomplish, but before I understood the math behind the problem, I thought it was worth a shot. So I dropped in an "IQ Bal Fix" block and fed its "Iqbal_corr" input with an "IQ Bal Optimize" block, and excitingly pressed play. And… the spike was still there. In fact it didn't look like it did anything when I overlaid the original raw data on top of it. The plot below showed the result:
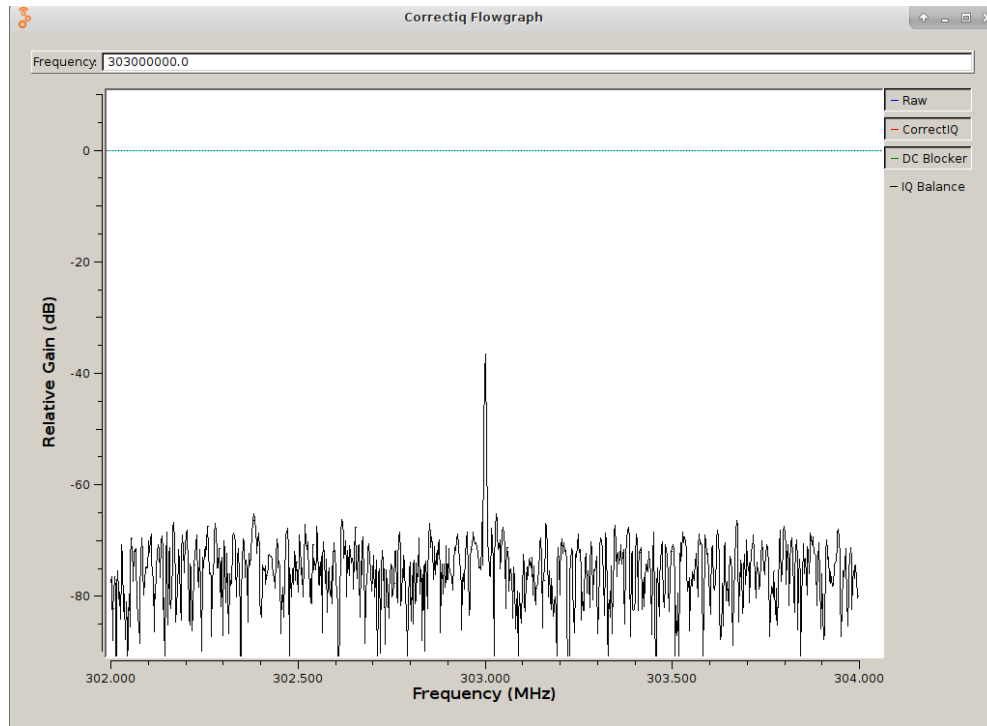
Wait… it's supposed to balance the IQ and correct it right? Why is it still there? The answer is IQ balancing is a different problem. IQ balancing has to do with the steady state DC voltage being different on the I and Q channels and causing problems in the data. IQ balancing simply makes the DC components match, not go away. So all I really accomplished was making the best quality DC spike I could. Not what I was hoping to accomplish.

So I struggled through this problem for a while and learned to offset tune with multiply blocks and XLating FIR filters just like everyone else suggested, or I switched to another SDR that didn't have the spike in the IQ data. But the problem was always nagging in the back of my mind. Why do the other SDR's and receiving software fix this issue but 1.) it's in my favorite hackrf, and 2.) there's no gnuradio block to take it out?

I finally had a few hours to look into the problem further and spent the time to search and understand what was happening, and the math behind fixing it. Then researched how others were doing the same thing in their code. Turns out the solution is simple. Since the data represents an alternating RF signal, over time the signal average in a clean signal should be zero (I know I'm oversimplifying it). When there's the IQ DC spike, that average isn't zero. So the solution is to calculate a weighted average over ongoing samples and simply subtract it from each future sample. It doesn't affect the overall quality of the filtered signal, but as long as the spike is on the center frequency, this approach very efficiently gets rid of it. And that was what I was hoping to accomplish.

So now that the problem and solution are clearly understood, the fix was straightforward. Fire up my favorite IDE, learn about building a gnuradio OOT module, and write some code. I have to admit I had already written a couple of other OOT modules, so that process I was already familiar with. But with that said, all-in-all it took less than 2 hours from starting a new project to testing a working module.

Here's how simple the fix is. With the gr-correctiq module installed, simply drop the module in right next to your osmocom source and the spike is gone!
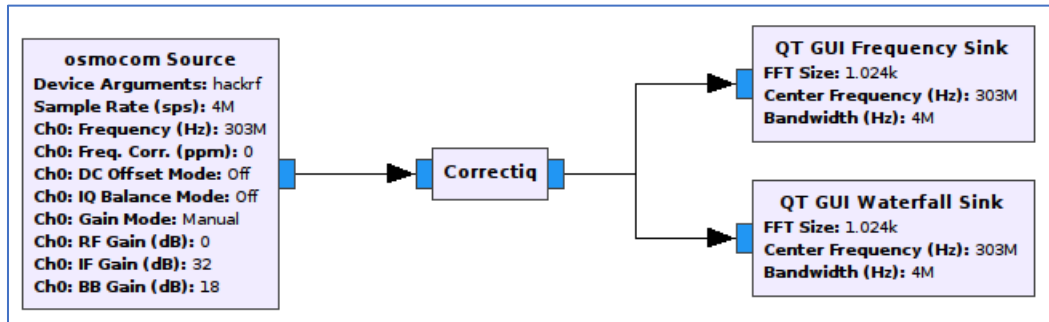
That's it! You're good to go. No more need for offset tuning to get rid of the spike. Now that we have a solution, if you have existing IQ recordings with the spike present, you can go back and filter it out too. Just feed the file through the block and write it back out to a new file. Just note that if you chose the "ignore it" option and the DC spike is on the center frequency and you were just working around the spike, it will be removed immediately. However [this is important] if you offset tuned before saving to your file, you will need to offset tune the spike back to center frequency for the removal to work.

For those of you that transmit as well, I'm sure you've noticed there's a similar IQ DC spike on the transmit side. Unfortunately correctiq can't fix that issue since that artifact happens in hardware after the transmitted data leaves the computer (if anyone has a fix for that too I'd be interested in hearing about it). But now you can at least receive clean signals on those flowgraphs you download without risking breaking them with incorrect offsetting or filter adjustments. And the good news is that if you do create flowgraphs with the correctiq block in it and change the osmocom source device to other SDR's it won't affect their signals (after all their average is already zero), so you can just leave it in.

Happy receiving!