Peidong Qi
04/17/2018
AMRUPT, Spring 2018

RTL SDR to Raspberry Pi Connection and Datalogging

Goals

Install GRC on Raspberry Pi. Install RTL SDR card on Raspberry Pi

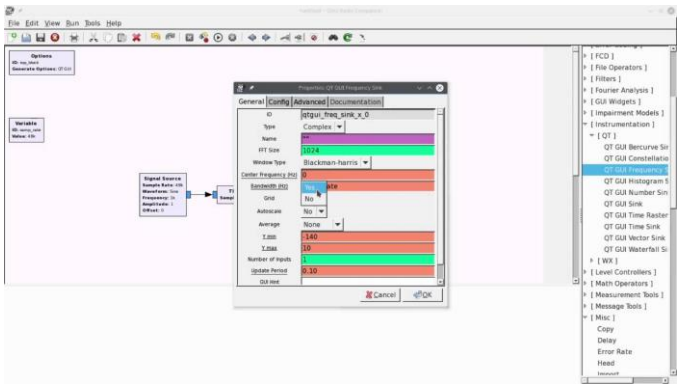
General approach

GNU Radio Companion (GRC) is a graphical tool for creating signal flow graphs and generating flow-graph source code. The requirements for GRC is python 2.5 or above. We have install python3 on Raspberry Pi already. GRC is bundled with gnuradio, so we already installed GRC when we install gun radio. However, GRC will not be installed if you were missing any of the required components. Install any missing components and re-run configure/install:

```
cmake ..
make
sudo make install
```

GRC should appear in the list of configured components.

To view the blocks' decimation from inside GRC, we need to install doxygen and configure gnuradio with doxygen support.



Here is the user interface of GRC, we can create a new flowgraph. Then we can add function blocks to the flowgraph.

After we install GRC on raspberry Pi, I try to run dirction-finding project from coherence-receiver website.

Code level

Pi2pi block

```
int pi2pi_ff_impl::work(int noutput_items, gr_vector_const_void_star & input_items, gr_vector_void_star & output_items) {
    const float * in = (const float * ) input_items[0];
    float * out = (float * ) output_items[0]; // Do <+signal processing+>
    for (int i = 0; i < noutput_items; ++i) {
        out[i] = fmod( in [i] + 20 * PI, 2 * PI);
        if (out[i] > PI) out[i] -= 2 * PI;
    } // Tell runtime system how many output items we produced.
    return noutput_items;
}
```

Phase2doa block

```
int phase2doa_ff_impl::work(int noutput_items, gr_vector_const_void_star & input_items, gr_vector_void_star & output_items) {
    const float * in = (const float * ) input_items[0];
    float * out = (float * ) output_items[0];
    float arg;
    for (int i = 0; i < noutput_items; i++) { // DOA = arccos(phase/(2*pi*alpha)) - pi/2
        arg = in [i] / (2 * M_PI * d_alpha); // clip at -1 and 1 so we don't get NaNs
        if (arg > 1) {
            arg = 1;
        } else if (arg < -1) {
            arg = -1;
        }
        out[i] = acos(arg) - M_PI_2;
    } // Tell runtime system how many output items we produced.
    return noutput_items;
}
```

Hold_ff block

```
int hold_ff_impl::work(int noutput_items, gr_vector_const_void_star & input_items, gr_vector_void_star & output_items) {
    const float * in = (const float * ) input_items[0];
    float * out = (float * ) output_items[0];
    if (!d_hold) { //pass every sample through, save the last one into d_value
        memcpy(out, in , sizeof(float) * noutput_items);
        d_value = in [noutput_items - 1];
    } else { //hold = true, so just return the last given value
        std::fill(out, out + noutput_items, d_value);
    } // Tell runtime system how many output items we produced.
    return noutput_items;
```

```
11. }
```

Planned Course of Action

Try to run a sample project with GRC. Keep study the dirction-finding project from coherence-receiver website.

Resources and relevant Forum Posts

[1] “RTL-SDR Blog silver dongle first impressions, compared to NooElec blue dongle”https://medium.com/@rxseger/rtl-sdr-blog-silver-dongle-first-impressions-compared-to-nooelec-blue-dongle-4053729ab8c7
[2] “VIDEO TUTORIAL: INSTALLING GQRX AND RTL-SDR ON A RASPBERRY PI”https://www.rtl-sdr.com/video-tutorial-installing-gqrx-and-rtl-sdr-on-a-raspberry-pi/
[3] “DIGITAL RADIO WITH A RASPBERRY PI” http://www.michaelcarden.net/?p=48
[4] “Guided Tutorial GNU Radio in Python ”https://wiki.gnuradio.org/index.php/Guided_Tutorial_GNU_Radio_in_Python
[5] “GRCon17 - Real-Time Direction Finding Using Two Antennas on an Android Phone - Sam Whiting” https://www.youtube.com/watch?v=jptYYiHth8U
[6]” Real-Time Direction Finding Using Two Antennas on an Android Phone - Sam Whiting” https://www.gnuradio.org/wp-content/uploads/2017/12/Todd-Moon-Gnuradio-DOA.pdf
[7]”gnuradio-doa” https://github.com/samwhiting/gnuradio-doa/tree/master/flowgraphs
[8]” GNURadioCompanion” https://wiki.gnuradio.org/index.php/GNURadioCompanion