

解析の手順書(GNU Radio)

下記4つの解析ソフトが「Sample_code_GNURadio.zip」に含まれている。

1. $\pi/4$ shift QPSK 復調ソフト(DQPSK_decoder_NEXUS_DS.grc)
2. CCSDS デコーダ(CCSDS_DECODER_binout.grc)
3. パケットごとに切り分けるソフト(Frame_Synch.slx)
4. AX.25 デコーダ(AX25_Decoder.m)

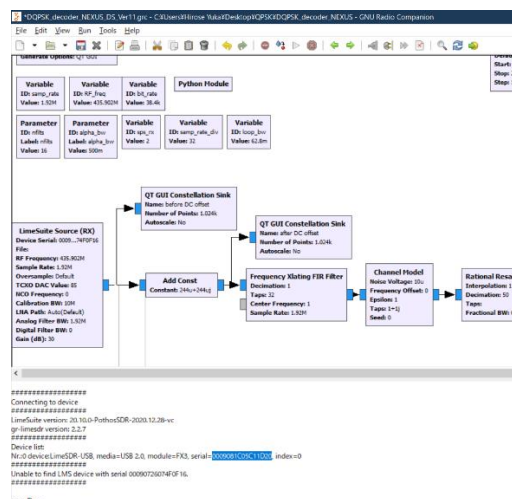
この4ソフトをダウンロードし、解析を行う。

環境構築については「Built_environment_GNURadio.pdf」に書かれているので、こちらもダウンロードする。

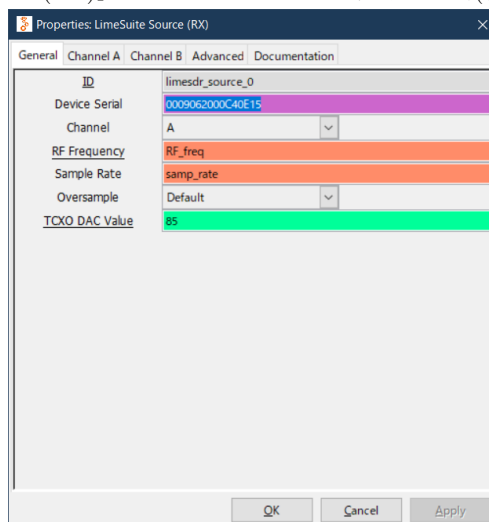
1. DQPSK_decoder_NEXUS_DS.grc (Windows 環境下 GNU Radio)

ここでの環境構築に関しては、環境構築手順書(GNU Radio)参照すること。

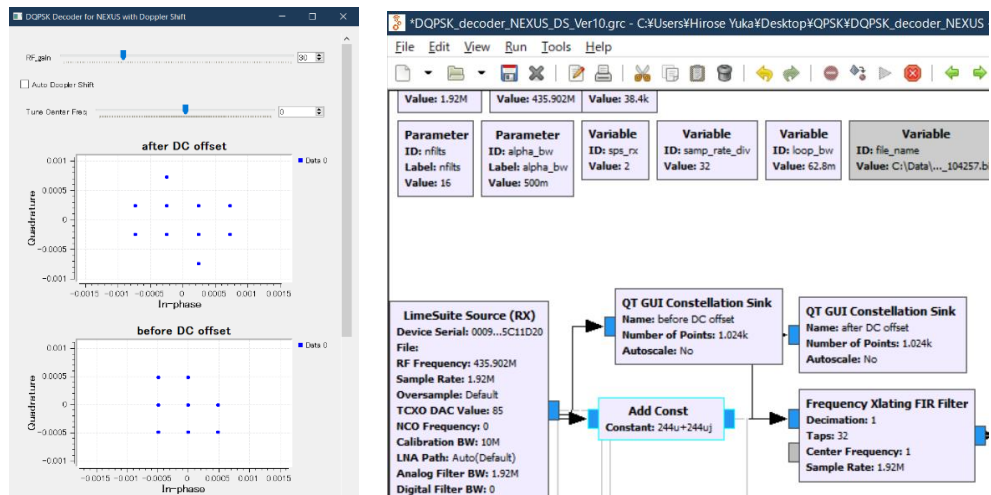
- ① LimeSDR シリアル番号取得のために、まずデフォルトのまま実行する。表示される LimeSDR のシリアル番号をコピーする。



- ② ブロック「LimeSuite Source(RX)」で LimeSDR シリアル番号を更新(初回のみ)。



- ③ 次に、DC オフセットの補正を行う。ソフトを実行し、before DC offset(下)のコンスタレーションの中心が座標中心に来るよう、ブロック「Add Const」の値を適宜変更[藤井1]する。[藤井2]なお、実数部分が I 軸、虚数部分が Q 軸を表す。

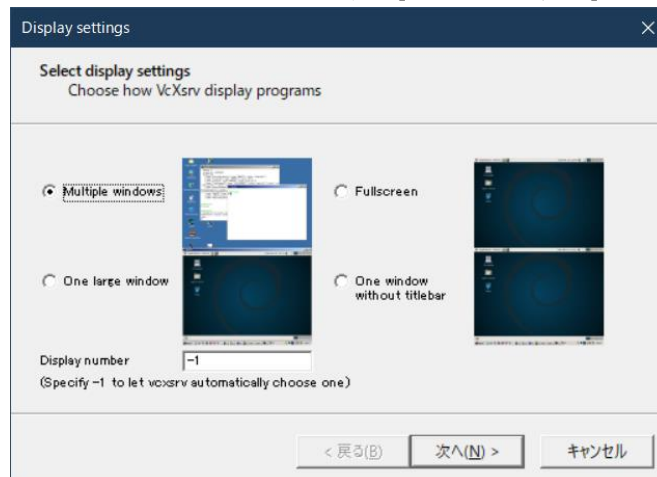


- ④ 上記の準備を整えた後、ソフトを実行し、出力 window の「Auto Doppler shift」にチェックする。
出力ファイル：「DQPSK_decode_data」

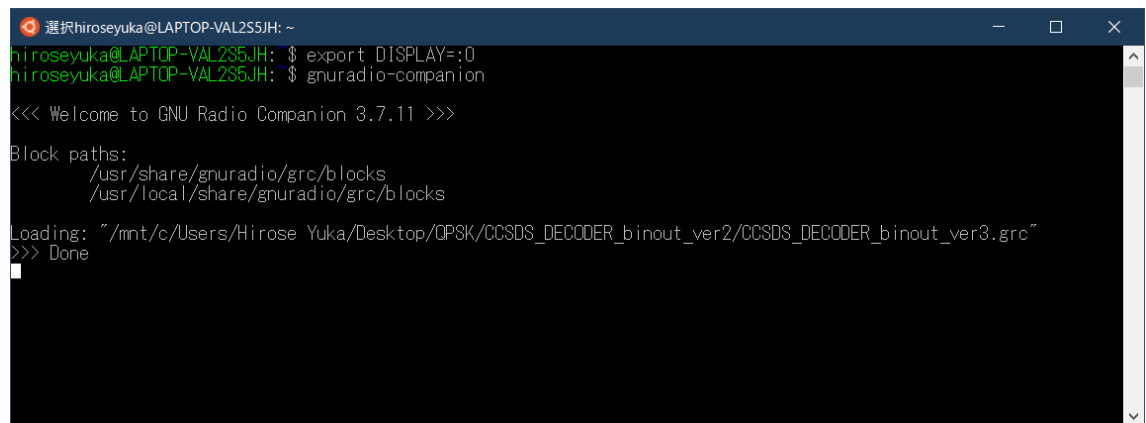
2. CCSDS_DECODER_binout.grc(Linux 環境下 GNU Radio)

Ubuntu 上の GNURadio で、CCSDS_DECODER を実行する。ここでは、Windows Subsystem for Linux(WSL)を利用して Ubuntu を立ち上げ、その上でソフトを実行する方法について述べる。このときの環境構築に関しては、環境構築手順書(GNU Radio)参照。[藤井3]環境構築手順書(GNU Radio)参照。

- ① Xlaunch を立ち上げる。全てデフォルトのまま「次へ」を選択し「完了」。



- ② Ubuntu を立ち上げる。
③ Ubuntu 上でコマンド「export DISPLAY=:0」を実行。
④ Ubuntu 上でコマンド「gnuradio-companion」を実行することで、Linux 版 GNU Radio が立ちあがる。



```
選択hiroseyuka@LAPTOP-VAL2S5JH: ~
hiroseyuka@LAPTOP-VAL2S5JH: ~$ export DISPLAY=:0
hiroseyuka@LAPTOP-VAL2S5JH: ~$ gnuradio-companion

<<< Welcome to GNU Radio Companion 3.7.11 >>>

Block paths:
  /usr/share/gnuradio/grc/blocks
  /usr/local/share/gnuradio/grc/blocks

Loading: "/mnt/c/Users/Hirose Yuka/Desktop/QPSK/CCSDS_DECODER_binout_ver2/CCSDS_DECODER_binout_ver3.grc"
>>> Done
```

⑤ File Source, File Sink のディレクトリを作業環境に合わせて指定し直し, 実行する.

出力ファイル: 「CCSDS_decode_data」

3. Frame Synch.slx を実行する. (Simulink)

デフォルトのままソフトを実行する.

出力ファイル: 「RS_decodedata.mat」

4. AX25_Decoder.m(MATLAB)

ディレクトリを作業環境に合わせて指定し直し, MATLAB で AX25_Decoder.m を実行する.

出力ファイル: 「AX25_decode_data..txt」