



Software Defined Radio Merapi Telemetry Decoder v1.0

INSTRUKSI MANUAL - *QUICK INSTALL*

Handiko Gesang Anugrah Sejati, S.T.
Unit Rumah Produksi PT. Datto Asia Teknologi
2018

I. KELENGKAPAN

Sistem penerima telemetri Merapi berbasis SDR (Software Defined Radio) ini membutuhkan subsistem yang terdiri dari:

1. Sistem Antena
2. Perangkat Keras SDR
3. Platform GNU Radio
4. Platform IoT

Keempat subsistem tersebut dibutuhkan agar data telemetri dari Merapi dapat diunggah menuju server IoT (Internet of Things) yang dalam contoh aplikasi ini menggunakan ThingSpeak.

1. Sistem Antena

Oleh karena sinyal telemetri Merapi yang cukup lemah, (hanya dipancarkan dengan daya sebesar $\pm 0,5$ Watt pada frekuensi 433,25 MHz) sistem antena yang baik dibutuhkan pada stasiun penerima. Untuk memaksimalkan SNR (Signal-to-Noise Ratio), dianjurkan untuk menggunakan antena directional, semisal 4-element Yagi atau yang lebih baik. Pengarahan antena directional juga merupakan hal yang penting mengingat sempitnya beamwidth dari antena yang digunakan. Stasiun telemetri Merapi (Ds. Balerante) terletak pada koordinat $7^{\circ}35.52'S$ $110^{\circ}27.48'E$ atau sekitar arah **Utara-Timur Laut (NNE) relatif dari pusat Kota Yogyakarta.**

2. Perangkat Keras SDR

Perangkat keras SDR yang digunakan pada aplikasi ini adalah RTL-SDR. Mengingat perangkat ini dapat menerima pada rentang frekuensi antara 24 MHz – 1700+ MHz, terdapat dua skenario yang dapat digunakan untuk menerima sinyal telemetri Merapi. Kedua skenario

tersebut adalah : penerimaan direct (UHF) dan penerimaan via cross-band repeater (**UHF-to-VHF**). Demi alasan portabilitas dan kehandalan, dianjurkan untuk melakukan penerimaan via cross-band repeater, yaitu sinyal telemetry Merapi pada frekuensi **433,25 MHz** dipancarkan ulang pada frekuensi **144,12 MHz** (atau frekuensi lain yang disepakati) dengan daya pancar yang relatif kuat sehingga kebutuhan akan sistem antena yang memiliki sifat pengarahan yang tinggi dapat ditekan. Dengan demikian, hanya memanfaatkan antena portable bawaan RTL-SDR, penerimaan data telemetry Merapi tetap dapat dilakukan. Skenario ini hanya dapat dilakukan pada cakupan repeater cross-band yang digunakan.

3. Platform GNU Radio

Untuk mengolah sinyal yang diterima oleh RTL-SDR, dibutuhkan sebuah platform DSP (Digital Signal Processing) dalam computer bersistem operasi Linux, yaitu GNU Radio. GNU Radio pada dasarnya adalah perangkat lunak DSP yang akan menerima sinyal telemetry yang ditangkap oleh RTL-SDR, mengolahnya, mendemodulasikan, mendecode, dan melakukan data-logging kedalam bentuk file berformat .csv atau diunggah ke server IoT. GNU Radio yang telah diinstal pada sistem operasi Linux digunakan untuk menjalankan aplikasi yang telah ditulis dalam bentuk aplikasi GRC (GNU Radio Companion). **Aplikasi GRC ini didistribusikan dalam bentuk source-code dan harus dikompilasi (compile) terlebih dahulu pada komputer yang akan menjalankan** agar dapat dieksekusi oleh GNU Radio.

4. Platform IoT

Untuk menampilkan data telemetri yang telah diterima dalam bentuk aplikasi online, digunakan platform IoT (Internet-of-Things) yang dalam hal ini adalah ThingSpeak (atau platform IoT lainnya sesuai kebutuhan). Yang perlu diperhatikan dalam penyediaan platform IoT ini adalah dibutuhkannya sambungan internet yang handal, **bersifat auto-connect**, dan tersedia secara kontinu. Tidak lupa pula pengaturan metoda pengunggahan ke server IoT harus tercatat/terdokumentasi dengan baik.

II. INSTALASI PLATFORM GNU RADIO

Platform GNU Radio yang digunakan adalah versi distribusi Linux. Sebuah komputer personal yang telah terinstal sistem operasi Linux (dianjurkan menggunakan versi Ubuntu 17.10) perlu dilakukan updating dan instalasi dependencies yang dibutuhkan oleh GNU Radio. Seluruh proses instalasi dilakukan dalam Terminal dengan mengetikkan baris perintah (command line). Adapun command line yang dibutuhkan adalah sebagai berikut:

1. Updating

```
sudo apt-get update && sudo apt-get upgrade -y
```

2. Instalasi Dependencies:

```
sudo apt-get install libboost-all-dev swig cmake git gcc -y
```

3. Instalasi GNU Radio dan RTL-SDR

```
sudo apt-get install gnuradio gr-osmosdr rtl-sdr -y
```

Setelah selesai, jalankan perintah:

```
cd /etc/modprobe.d  
sudo nano blacklist.conf
```

Pada baris yang kosong (sekitar 4 – 5 baris dari atas, atau baris manapun) masukkan teks berikut:

```
blacklist dvb_usb_rtl28xxu
```

(perhatikan, huruf sebelum 28xxu adalah **huruf-l**)

Tekan **CTRL+O** (save) kemudian **CTRL+X** (exit). Langkah terakhir adalah dengan melakukan restart komputer. Setelah restart, colokkan RTL-SDR kedalam port USB dan jalankan perintah berikut:

```
rtl_test
```

Jika rangkaian instalasi di atas berhasil maka akan muncul tanggapan yang sama atau mirip seperti gambar berikut:



```
handiko@Lenovo-IP110: ~  
File Edit Tabs Help  
handiko@Lenovo-IP110:~$ rtl_test  
Found 1 device(s):  
 0: Realtek, RTL2838UHIDIR, SN: 00000002  
  
Using device 0: Generic RTL2832U OEM  
Found Rafael Micro R820T tuner  
Supported gain values (29): 0.0 0.9 1.4 2.7 3.7 7.7 8.7 12.5 14.4 15.7 16.6 19.7 20.7 22.9 25.4 28.0 29.7 32.8  
33.8 36.4 37.2 38.6 40.2 42.1 43.4 43.9 44.5 48.0 49.6  
[R82XX] PLL not locked!  
Sampling at 2048000 S/s.  
  
Info: This tool will continuously read from the device, and report if  
samples get lost. If you observe no further output, everything is fine.  
  
Reading samples in async mode...  
lost at least 64 bytes  
█
```

Untuk menguji keberhasilan instalasi GNU Radio, jalankan perintah berikut:

```
gnuradio-config-info --prefix
```

dan akan muncul tanggapan berupa

```
/usr
```

atau

```
/usr/local
```

Catat tanggapan ini. Tanggapan di atas menunjukkan lokasi direktori instalasi GNU Radio. Jika tanggapan di atas adalah `/usr`, maka pada langkah selanjutnya, aplikasi penerima SDR Merapi akan diinstal pada direktori tersebut. Sama halnya pula jika lokasi direktori instalasi adalah `/usr/local`.

Untuk menguji keberhasilan instalasi dependencies yang dibutuhkan untuk menginstal aplikasi SDR Merapi, lakukan langkah sebagai berikut:

Salin folder distribusi gr-Merapi ke /home directory. Kemudian jalankan perintah:

```
cd gr-Merapi
```

```
mkdir build
```

```
cd build
```

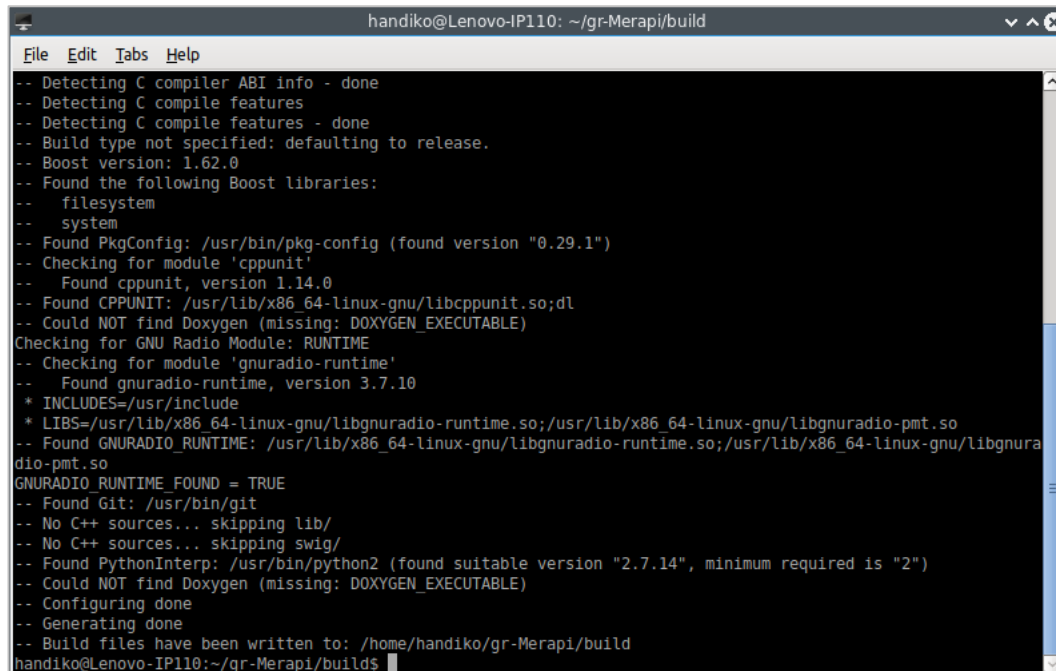
```
cmake ../ -DCMAKE_INSTALL_PREFIX=/usr
```

atau

```
cmake ../ -DCMAKE_INSTALL_PREFIX=/usr/local
```

Parameter `/usr` ataupun `/usr/local` bergantung pada tanggapan perintah `gnuradio-config-info --prefix` yang telah dijalankan dan dicatat sebelumnya. Jika seluruh proses instalasi berhasil, maka tanggapan

atas perintah `cmake ../ -DCMAKE_INSTALL_PREFIX=/usr` akan memuat kalimat sebagai berikut:



```
handiko@Lenovo-IP110: ~/gr-Merapi/build
File Edit Tabs Help
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Build type not specified: defaulting to release.
-- Boost version: 1.62.0
-- Found the following Boost libraries:
--   filesystem
--   system
-- Found PkgConfig: /usr/bin/pkg-config (found version "0.29.1")
-- Checking for module 'cppunit'
--   Found cppunit, version 1.14.0
-- Found CPPUNIT: /usr/lib/x86_64-linux-gnu/libcppunit.so;dl
-- Could NOT find Doxygen (missing: DOXYGEN_EXECUTABLE)
Checking for GNU Radio Module: RUNTIME
-- Checking for module 'gnuradio-runtime'
--   Found gnuradio-runtime, version 3.7.10
* INCLUDES=/usr/include
* LIBS=/usr/lib/x86_64-linux-gnu/libgnuradio-runtime.so;/usr/lib/x86_64-linux-gnu/libgnuradio-pmt.so
-- Found GNURADIO_RUNTIME: /usr/lib/x86_64-linux-gnu/libgnuradio-runtime.so;/usr/lib/x86_64-linux-gnu/libgnuradio-pmt.so
GNURADIO_RUNTIME_FOUND = TRUE
-- Found Git: /usr/bin/git
-- No C++ sources... skipping lib/
-- No C++ sources... skipping swig/
-- Found PythonInterp: /usr/bin/python2 (found suitable version "2.7.14", minimum required is "2")
-- Could NOT find Doxygen (missing: DOXYGEN_EXECUTABLE)
-- Configuring done
-- Generating done
-- Build files have been written to: /home/handiko/gr-Merapi/build
handiko@Lenovo-IP110:~/gr-Merapi/builds
```

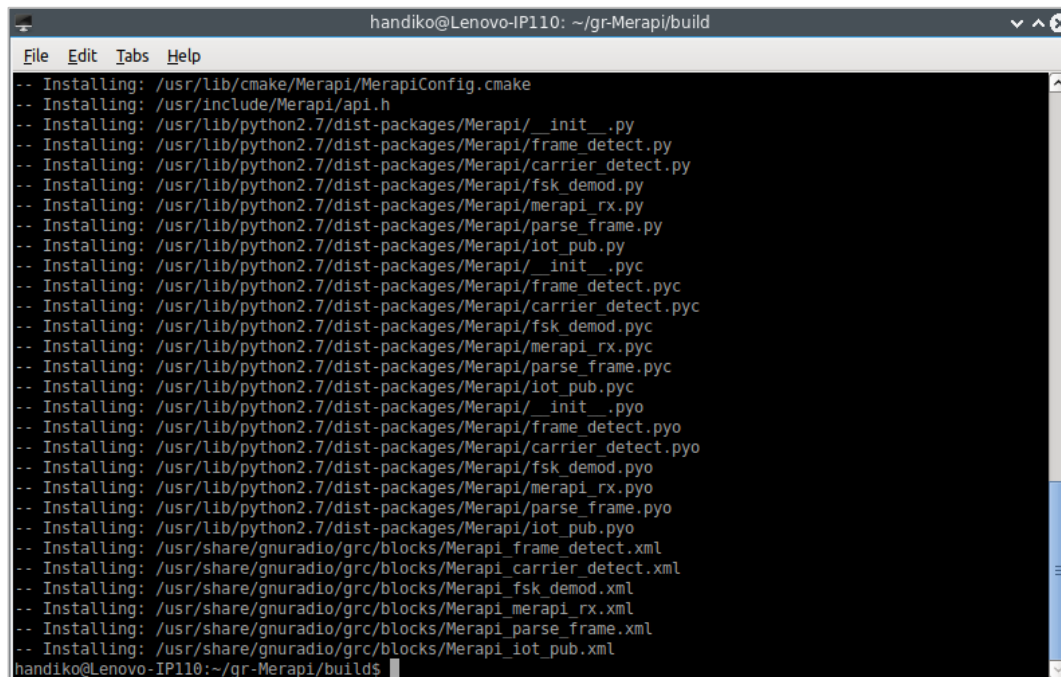
III. INSTALASI SDR MERAPI

Aplikasi SDR Merapi yang didistribusikan dalam bentuk source-code perlu dikompilasi terlebih dahulu dan diinstal ke dalam GNU Radio. Proses kompilasi dan instalasi ini cukup dilakukan **hanya satu kali** pada komputer Linux yang belum pernah diinstal aplikasi tersebut.

```
cd
cd gr-Merapi
mkdir build
cd build
cmake ../ -DCMAKE_INSTALL_PREFIX=/usr
atau
cmake ../ -DCMAKE_INSTALL_PREFIX=/usr/local
```

```
sudo make  
sudo make install
```

Jika hingga tahap ini instalasi berhasil, maka pada terminal akan terlihat output sebagai berikut;



```
handiko@Lenovo-IP110: ~/gr-Merapi/build  
File Edit Tabs Help  
-- Installing: /usr/lib/cmake/Merapi/MerapiConfig.cmake  
-- Installing: /usr/include/Merapi/api.h  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/_init_.py  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/frame_detect.py  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/carrier_detect.py  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/fsk_demod.py  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/merapi_rx.py  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/parse_frame.py  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/iot_pub.py  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/_init_.pyc  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/frame_detect.pyc  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/carrier_detect.pyc  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/fsk_demod.pyc  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/merapi_rx.pyc  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/parse_frame.pyc  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/iot_pub.pyc  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/_init_.pyo  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/frame_detect.pyo  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/carrier_detect.pyo  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/fsk_demod.pyo  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/merapi_rx.pyo  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/parse_frame.pyo  
-- Installing: /usr/lib/python2.7/dist-packages/Merapi/iot_pub.pyo  
-- Installing: /usr/share/gnuradio/grc/blocks/Merapi_frame_detect.xml  
-- Installing: /usr/share/gnuradio/grc/blocks/Merapi_carrier_detect.xml  
-- Installing: /usr/share/gnuradio/grc/blocks/Merapi_fsk_demod.xml  
-- Installing: /usr/share/gnuradio/grc/blocks/Merapi_merapi_rx.xml  
-- Installing: /usr/share/gnuradio/grc/blocks/Merapi_parse_frame.xml  
-- Installing: /usr/share/gnuradio/grc/blocks/Merapi_iot_pub.xml  
handiko@Lenovo-IP110:~/gr-Merapi/build$
```

Kemudian jalankan perintah :

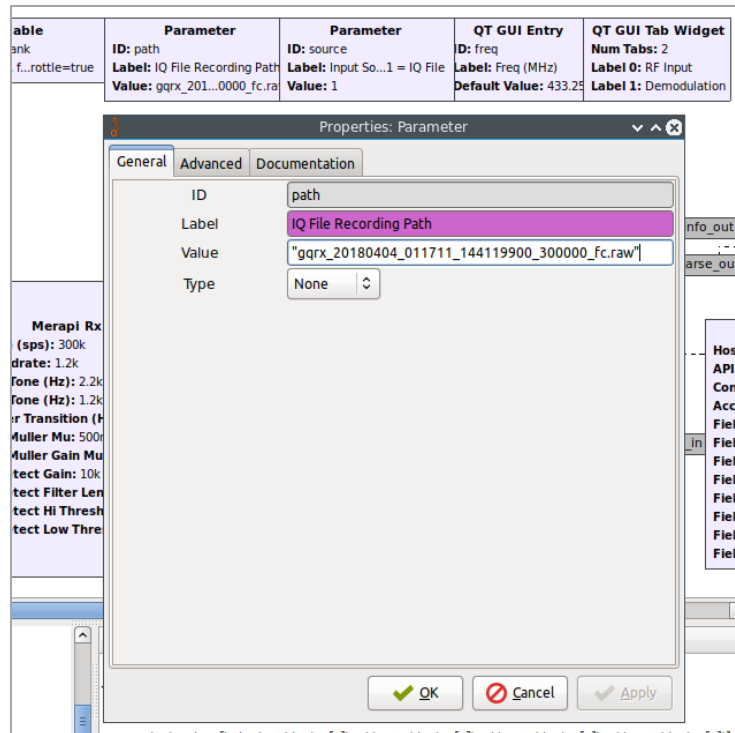
```
sudo ldconfig  
cd
```

IV. KONFIGURASI SDR

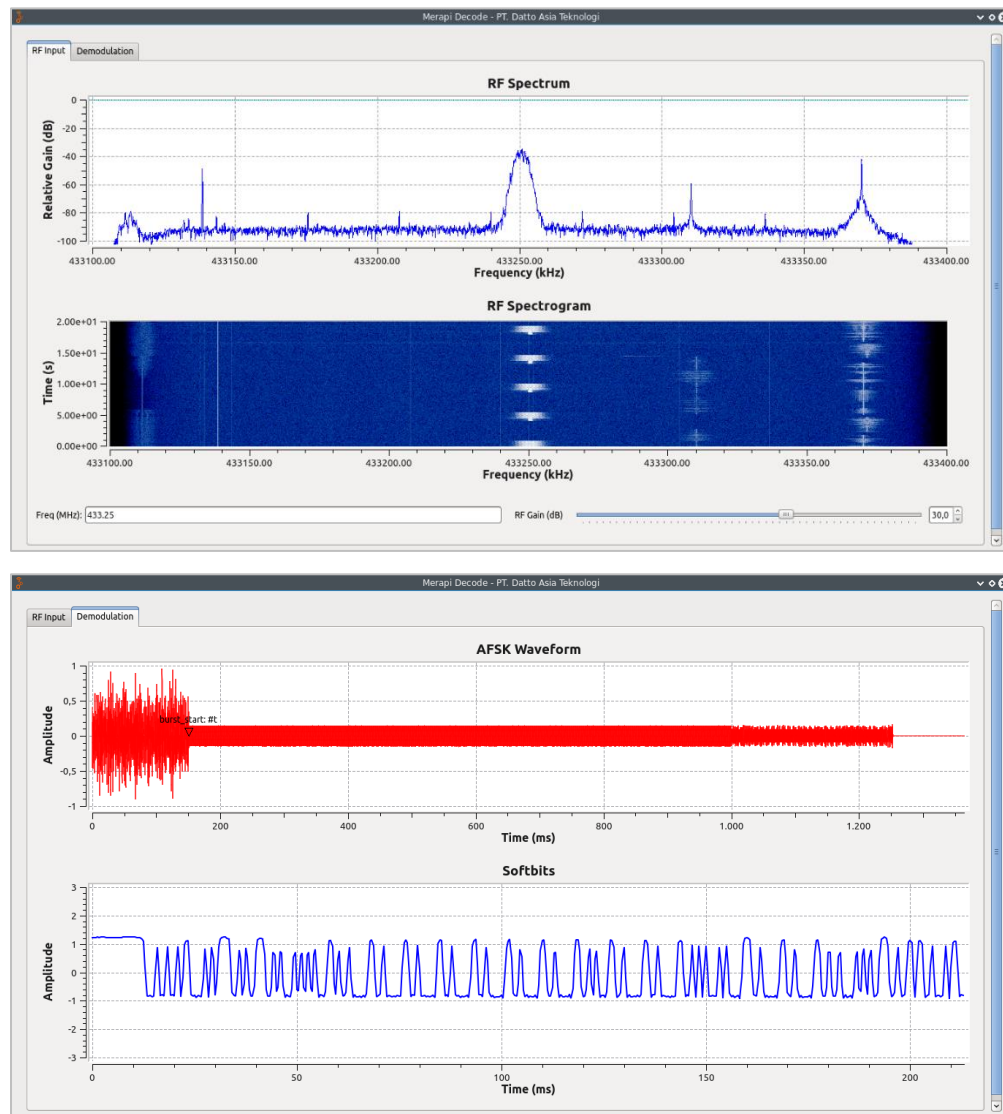
Bagian pertama dari langkah ini adalah uji coba hasil instalasi SDR Merapi Decoder dengan menjalankan file uji.

1. Buka perangkat lunak GNU Radio dengan mengetikkan perintah `gnuradio-companion` dan akan terbuka jendela perangkat lunak sebagai berikut:

5. Pada flowgraph merapi_decode.grc, cari block Parameter dengan ID **path**, dan masukkan **"gqrx_20180404_011711_144119900_300000_fc.raw"** sebagai Value kemudian klik OK.



6. Jalankan flowgraph dengan file uji. Jika instalasi berhasil, akan tampil window sebagai berikut:



7. Buka Terminal new tab dan ketikkan perintah `nc localhost 52001` lalu ENTER. Pada terminal akan muncul data merapi hasil decode realtime sebagai berikut:

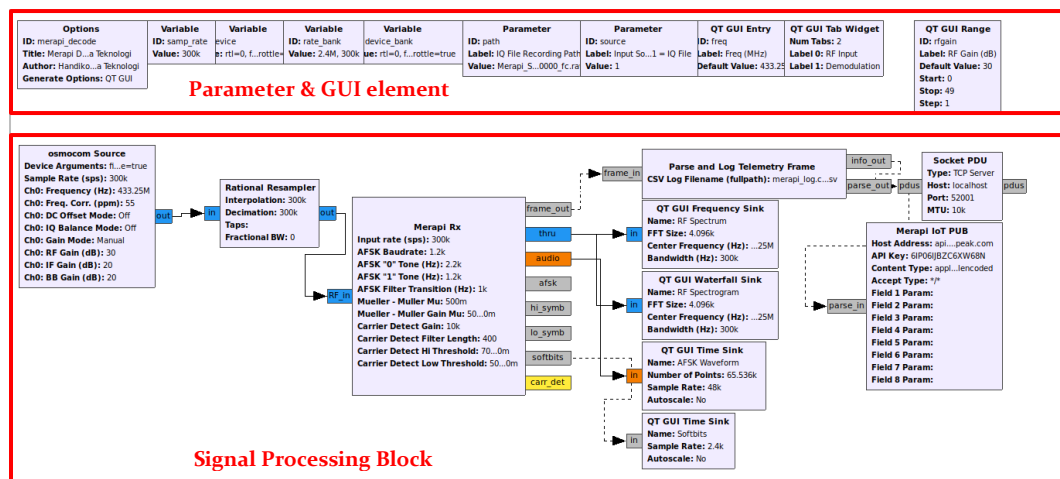
```

handiko@Lenov... x handiko@Lenov... x
handiko@Lenov:~$ nc localhost 52001
[Wed May 2 10:20:47 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -840.0 % RAIN = 25.5 mm
[Wed May 2 10:20:52 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -841.0 % RAIN = 25.5 mm
[Wed May 2 10:20:57 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -845.9 % RAIN = 25.5 mm
[Wed May 2 10:21:02 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -840.95 % RAIN = 25.5 mm
[Wed May 2 10:21:07 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -840.0 % RAIN = 25.5 mm
[Wed May 2 10:21:12 2018] HOR = 0.0 v/v VERT = 0.1 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -841.0 % RAIN = 25.5 mm
[Wed May 2 10:21:17 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -845.85 % RAIN = 25.5 mm
[Wed May 2 10:21:22 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.5 deg.C Gas Cons. = 14.85 % HUM = -840.9 % RAIN = 25.5 mm
[Wed May 2 10:21:27 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -845.75 % RAIN = 25.5 mm
[Wed May 2 10:21:32 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.5 deg.C Gas Cons. = 14.85 % HUM = -840.9 % RAIN = 25.5 mm
[Wed May 2 10:21:37 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -845.8 % RAIN = 25.5 mm
[Wed May 2 10:21:42 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.5 deg.C Gas Cons. = 14.85 % HUM = -840.9 % RAIN = 25.5 mm
[Wed May 2 10:21:47 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.5 deg.C Gas Cons. = 14.85 % HUM = -845.8 % RAIN = 25.5 mm
[Wed May 2 10:21:52 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.5 deg.C Gas Cons. = 14.85 % HUM = -840.85 % RAIN = 25.5 mm
[Wed May 2 10:21:57 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.5 deg.C Gas Cons. = 14.85 % HUM = -845.8 % RAIN = 25.5 mm
[Wed May 2 10:22:02 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.5 deg.C Gas Cons. = 14.85 % HUM = -840.9 % RAIN = 25.5 mm
[Wed May 2 10:22:07 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -845.7 % RAIN = 25.5 mm
[Wed May 2 10:22:12 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.5 deg.C Gas Cons. = 14.85 % HUM = -840.85 % RAIN = 25.5 mm
[Wed May 2 10:22:17 2018] HOR = 0.0 v/v VERT = 0.05 v/v TEMP = 34.4 deg.C Gas Cons. = 14.85 % HUM = -845.7 % RAIN = 25.5 mm

```

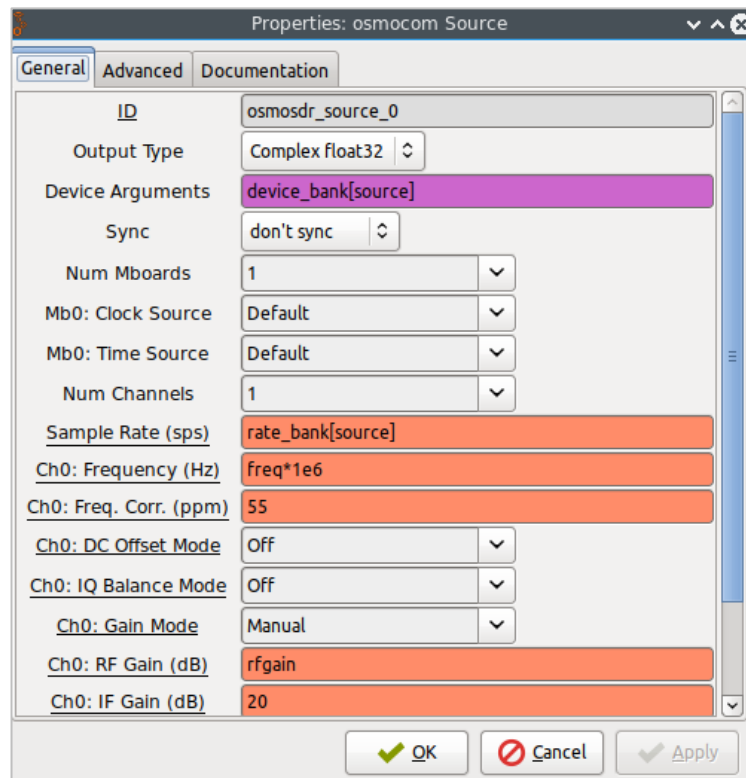
Bagian kedua dari langkah ini adalah pengaturan parameter dari tiap blok pengolah sinyal dalam flowgraph **merapi_decode.grc**. Untuk keperluan backup, pastikan file **merapi_decode.grc** yang dibuka/digunakan telah berada pada /home. Jika GNU Radio masih dalam keadaan membuka **merapi_decode.grc** dari folder /gr-Merapi/example, **Save As** file tersebut ke dalam /home. File yang terdapat dalam gr-Merapi sebaiknya tidak dimodifikasi dalam bentuk apapun sehingga dapat digunakan sebagai backup.

Mari kita lihat flowgraph lengkap dari merapi_decode.grc berikut. Isi dari flowgraph tersebut terbagi menjadi dua bagian yaitu **parameter dan GUI element (PGE)** serta **signal processing block (SPB)**.



1. osmocom Source (SPB)

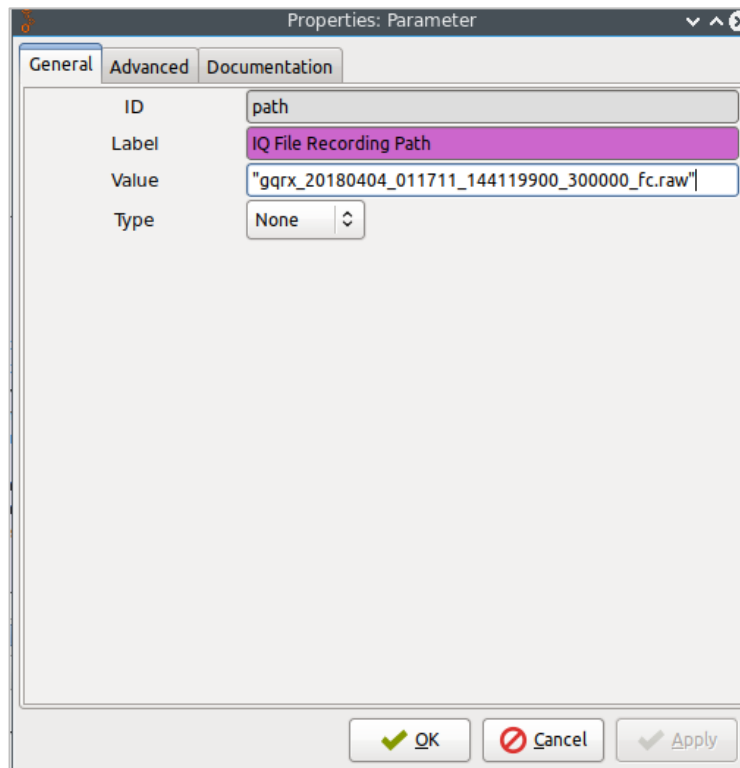
Blok ini berfungsi sebagai interface antara perangkat keras SDR (dalam hal ini RTL-SDR) dengan platform GNU Radio. Pada aplikasi ini, **osmocom Source** telah diatur secara default sebagai berikut:



2. Parameter path (PGE)

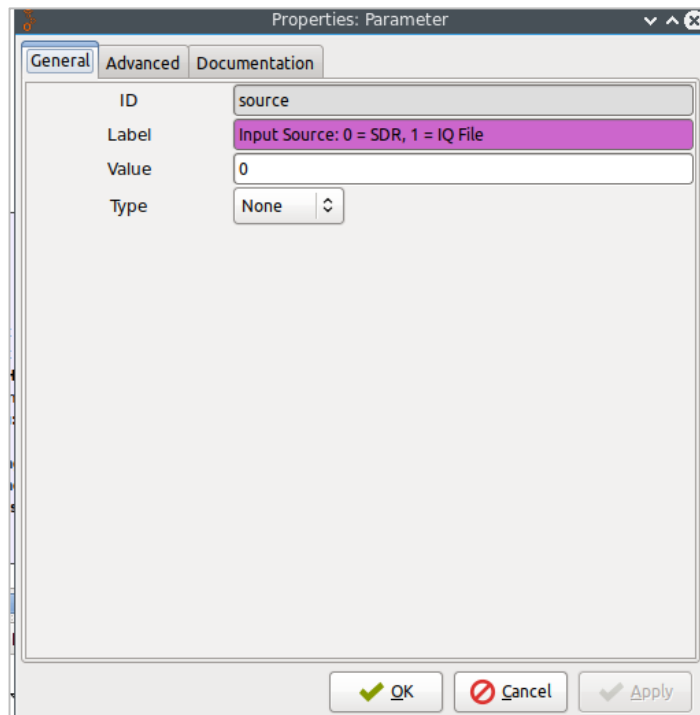
Dalam moda uji coba, parameter path digunakan oleh **osmocom Source** untuk membaca file uji (rekaman baseband sinyal telemetri merapi) sebagai sinyal uji. Oleh karena itu pada **Value**, masukkan nama file uji dan lokasi direktorinya. Dengan mengacu pada bagian uji coba di atas, file uji yang dimaksud adalah

ggrx_20180404_011711_144119900_300000_fc.raw dan lokasi direktori adalah /home. Jika file uji tersebut telah diletakkan dalam /home, maka **Value** cukup diinputkan seperti gambar berikut:.



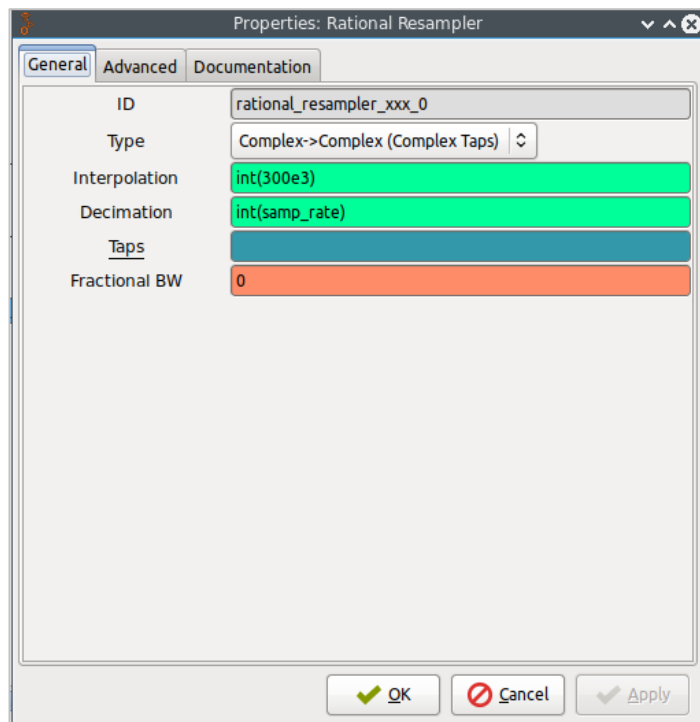
3. Parameter source (PGE)

Blok ini untuk menentukan apakah flowgraph dijalankan dalam moda uji (membaca rekaman file uji) atau moda operasional (menerima sinyal telemetry Merapi secara realtime menggunakan perangkat SDR). Oleh karena itu pengaturan parameter ini **PENTING** adanya dan harus dilakukan dengan benar. Untuk **MODA UJI**, Value dari parameter ini **adalah 1** dan untuk **MODA OPERASIONAL** Value yang diinginkan **adalah 0**.



4. Rational Resampler (SPB)

Biarkan blok ini dalam kondisi default.



5. Merapi Rx (SPB)

Blok ini adalah jantung dari proses pengolahan sinyal telemetry merapi yang diterima menjadi frame data/paket data yang kemudian akan dipisahkan menurut parameter/data field yang bersesuaian. Blok ini berisi proses-proses diantaranya adalah: Carrier detection, FM demodulation, AFSK demodulation, Mueller-Mueller based Clock Recovery, dan Binary Slicing. Parameter blok ini telah diatur secara optimal dengan kondisi default berikut:

Parameter	Value
ID	Merapi_merapi_rx_0
Input rate (sps)	300e3
AFSK Baudrate	1200
AFSK "0" Tone (Hz)	2200.0
AFSK "1" Tone (Hz)	1200.0
AFSK Filter Transition (Hz)	1e3
Mueller - Muller Mu	0.5
Mueller - Muller Gain Mu	0.5
Carrier Detect Gain	10e3
Carrier Detect Filter Leng	400
Carrier Detect Hi Thresh	0.7
Carrier Detect Low Thresh	0.5

Value yang perlu diatur secara manual untuk sistem telemetry berbasis AFSK:

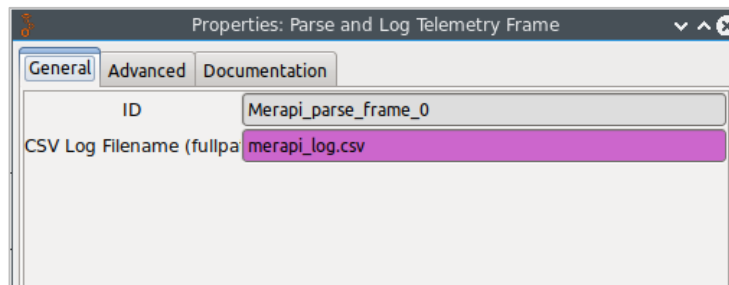
- **AFSK Baudrate**, nilai yang umum: 300, 600, 1200, 4800, 9600. Harus merujuk pada parameter sistem telemetry yang digunakan.
- **AFSK "0" Tone**, untuk sistem pengirim telemetry berbasis TCM3105, isikan frekuensi yang membawa bit-0 ("space"). Untuk

sistem 1200 baud, berbasis TCM3105, dan standard Bell 202, frekuensi “space” adalah 2200 Hz.

- **AFSK “1” Tone**, sebagaimana parameter **AFSK “0” Tone**, parameter ini perlu merujuk pada frekuensi “mark” yang digunakan. Dalam aplikasi ini digunakan sesuai standard yaitu 1200 Hz.
- **Carrier Detect Hi dan Low Threshold**, masing-masing harus disesuaikan dengan rerata level kuat sinyal yang diterima.
- **Mueller - Mueller Mu dan Gain Mu**, parameter ini adalah parameter yang paling sulit untuk ditentukan dan **harus diatur spesifik terhadap waveform sinyal telemetry yang dikerjakan**. Untuk keperluan aplikasi ini, nilai yang tercantum telah merupakan nilai optimal.

6. Parse and Log Telemetry Frame (SPB)

Blok ini memisahkan data field (mem-parsing) data dari frame/paket yang terdeteksi oleh blok Merapi Rx. Untuk tampilan dalam format string, data merapi dapat diambil pada output port label “info_out” sedangkan data terformat untuk dipublish menuju platform IoT dapat diambil pada output port berlabel “parse_out”.



Blok ini memiliki parameter berupa nama file berformat .csv yang digunakan sebagai data logger. Seluruh data yang berhasil didecode akan terekam dalam file ini.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	[Mon Aug 13 10:36:10 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.2	RAIN =	25.5					
2	[Mon Aug 13 10:36:15 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-840.05	RAIN =	25.5					
3	[Mon Aug 13 10:36:20 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.15	RAIN =	25.5					
4	[Mon Aug 13 10:36:25 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-845.9	RAIN =	25.5					
5	[Mon Aug 13 10:36:30 2018]	HOR =	0.0	VERT =	0.1	TEMP =	34.4	GAS =	14.85	HUM =	-841.2	RAIN =	25.5					
6	[Mon Aug 13 10:36:35 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-840.2	RAIN =	25.5					
7	[Mon Aug 13 10:36:40 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.65	RAIN =	25.5					
8	[Mon Aug 13 10:36:45 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-840.1	RAIN =	25.5					
9	[Mon Aug 13 10:36:50 2018]	HOR =	0.0	VERT =	0.1	TEMP =	34.4	GAS =	14.85	HUM =	-841.25	RAIN =	25.5					
10	[Mon Aug 13 10:36:55 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-840.15	RAIN =	25.5					
11	[Mon Aug 13 10:37:00 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.65	RAIN =	25.5					
12	[Mon Aug 13 10:37:05 2018]	HOR =	0.0	VERT =	0.1	TEMP =	34.4	GAS =	14.85	HUM =	-840.1	RAIN =	25.5					
13	[Mon Aug 13 10:37:36 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.2	RAIN =	25.5					
14	[Mon Aug 13 10:37:41 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-840.05	RAIN =	25.5					
15	[Mon Aug 13 10:37:46 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.15	RAIN =	25.5					
16	[Mon Aug 13 10:37:51 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-845.9	RAIN =	25.5					
17	[Mon Aug 13 10:37:56 2018]	HOR =	0.0	VERT =	0.1	TEMP =	34.4	GAS =	14.85	HUM =	-841.2	RAIN =	25.5					
18	[Mon Aug 13 10:38:01 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-840.2	RAIN =	25.5					
19	[Mon Aug 13 10:41:31 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-840.05	RAIN =	25.5					
20	[Mon Aug 13 10:44:49 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.2	RAIN =	25.5					
21	[Mon Aug 13 10:44:54 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-840.05	RAIN =	25.5					
22	[Mon Aug 13 10:44:59 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.15	RAIN =	25.5					
23	[Mon Aug 13 10:45:04 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-845.9	RAIN =	25.5					
24	[Mon Aug 13 11:38:12 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.2	RAIN =	25.5					
25	[Mon Aug 13 11:38:25 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.2	RAIN =	25.5					
26	[Mon Aug 13 11:38:30 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-840.05	RAIN =	25.5					
27	[Mon Aug 13 11:38:35 2018]	HOR =	0.0	VERT =	0.05	TEMP =	34.4	GAS =	14.85	HUM =	-841.15	RAIN =	25.5					

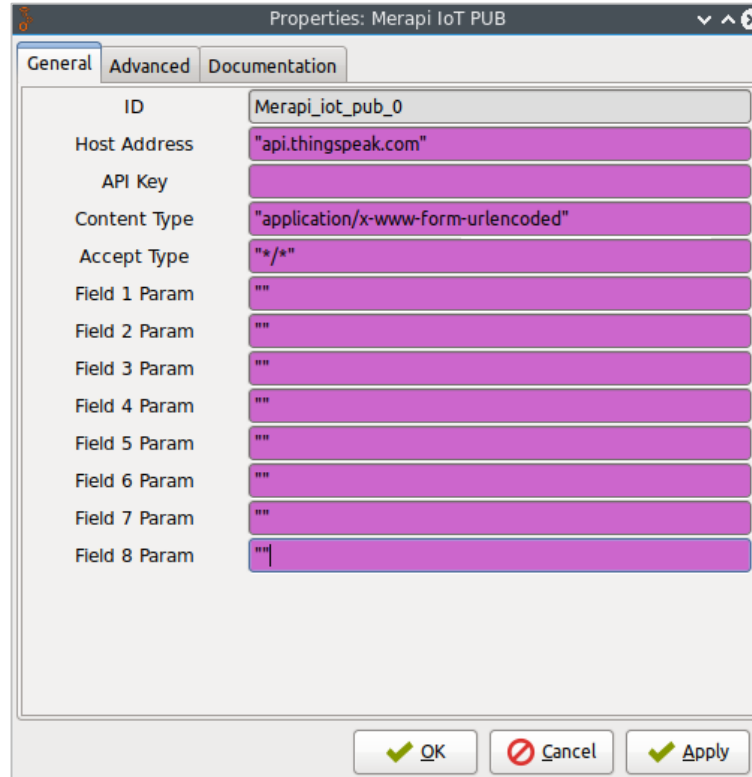
7. Socket PDU (SPB)

Blok ini berguna untuk menampilkan data merapi dalam bentuk string di Terminal. Secara default, blok ini memiliki parameter sebagai berikut:

Properties: Socket PDU	
General	Advanced Documentation
ID	blocks_socket_pdu_0
Type	TCP Server
Host	localhost
Port	52001
MTU	10000
TCP No Delay	Disabled
<input type="button" value="OK"/> <input type="button" value="Cancel"/> <input type="button" value="Apply"/>	

8. Merapi IoT PUB (SPB)

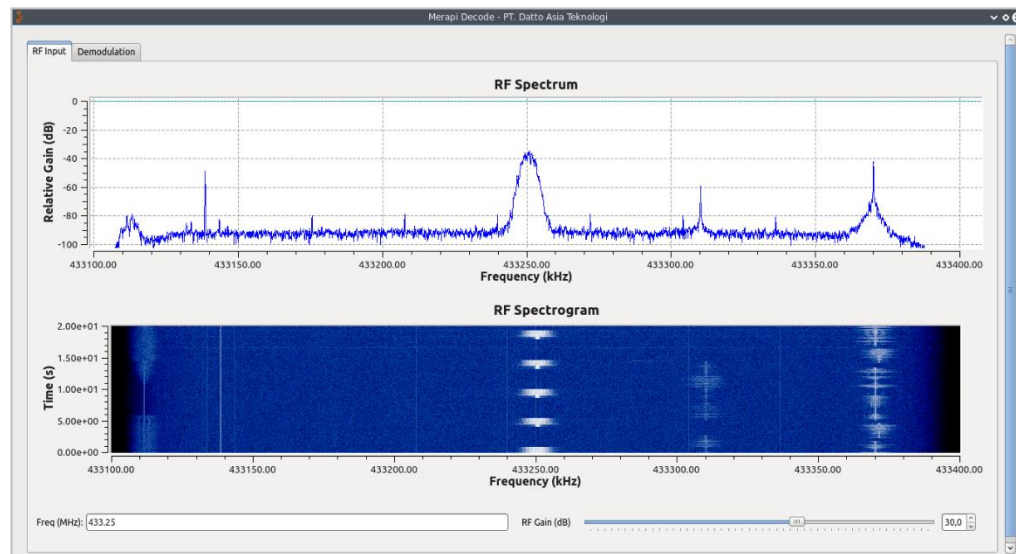
Blok ini berguna untuk mempublish data merapi dalam format yang kompatibel dengan platform IoT (sebagai contoh ThingSpeak). Parameter yang perlu diatur adalah Host Address dan API Key (diisikan dengan diapit tanda petik ganda "...").



Properties: Merapi IoT PUB	
General Advanced Documentation	
ID	Merapi_iot_pub_0
Host Address	api.thingspeak.com
API Key	
Content Type	application/x-www-form-urlencoded
Accept Type	*/*
Field 1 Param	
Field 2 Param	
Field 3 Param	
Field 4 Param	
Field 5 Param	
Field 6 Param	
Field 7 Param	
Field 8 Param	
OK Cancel Apply	

9. Pengaturan Selama RUNTIME

Pengaturan ini sangat **PENTING** dilakukan dengan guna untuk **mengoptimalkan parameter pengolahan sinyal terutama terhadap level sinyal yang diterima** (bergantung pada instalasi antenna dan lokasi penerimaan) **serta frekuensi kerja** (tergantung pada penggunaan frekuensi yang telah diatur/disepakati, dan nilai offset dari perangkat RTL-SDR yang digunakan). Pengaturan ini adalah pengaturan nilai **Freq** dan **RF Gain** dalam window (GUI) aplikasi saat dijalankan.



Atur nilai **Freq (MHz)** sesuai frekuensi kerja yang telah ditentukan (dalam contoh ini adalah 433,250 MHz) hingga sinyal yang dimaksud berada di tengah-tengah tampilan **RF Spectrum** dan **RF Spectrogram**. Untuk nilai **RF Gain (dB)** variasikan nilai (gunakan slider) dari yang terendah, secara perlahan-lahan ditingkatkan menuju nilai optimal. Nilai optimal dari **RF Gain** adalah nilai tertinggi yang tidak menyebabkan receiver overload. Nilai ini bergantung pada kuat sinyal yang diterima yang bergantung pula terhadap kondisi instalasi antenna dan jarak penerimaan dari stasiun telemetri merapi/repeater yang digunakan. Beberapa panduan nilai **RF Gain** yang optimal adalah sebagai berikut:

Konndisi Penerimaan	RF Gain (dB)
Jarak stasiun telemetri/repeater sangat dekat. RTL-SDR menggunakan antenna helical/stick outdoor.	5 - 10 dB
Jarak stasiun telemetri/repeater sangat dekat. RTL-SDR menggunakan antenna helical/stick indoor.	5 - 20 dB

Jarak stasiun telemetri/repeater sangat dekat. RTL-SDR menggunakan antenna luar (vertical/yagi).	0 – 10 dB
Jarak stasiun telemetri/repeater cukup jauh. RTL-SDR menggunakan antenna helical/stick outdoor.	20 – 30 dB
Jarak stasiun telemetri/repeater cukup jauh. RTL-SDR menggunakan antenna helical/stick indoor.	30 – 45 dB
Jarak stasiun telemetri/repeater cukup jauh. RTL-SDR menggunakan antenna vertikal (gain rendah).	10 – 30 dB
Jarak stasiun telemetri/repeater cukup jauh. RTL-SDR menggunakan antenna yagi (gain tinggi).	10 – 20 dB

Nilai **RF Gain** optimal di atas adalah hanya sebagai rujukan awal dan dapat bervariasi tergantung dari kondisi penerimaan/instalasi. Dalam moda uji (membaca file uji) pengaturan **RF Gain** dan **Freq** tidak perlu dilakukan.