

Real-time packet reception with software defined radio

Maria Raftopoulou 4620100 M.Raftopoulou@student.tudelft.nl

Salman 4593413 salman@student.tudelft.nl

Omer Zareen 4625919 M.O.Zareen@student.tudelft.nl

Wireless Networking, Quarter 3, 2017

1 Steps

The first step that we had to do, was to become sure that the SDR was connected and configured properly. To do that, we installed the software SDRSharp and tried to capture FM signals as it shows in figure 1.

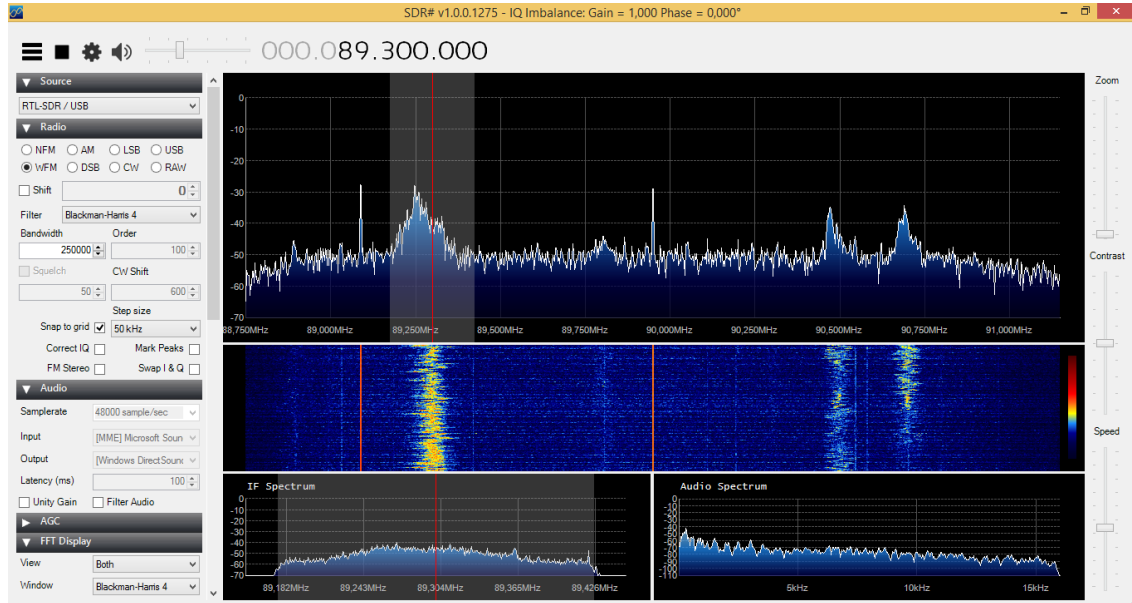


Figure 1: FM signal with SDRSharp

Knowing that the SDR is configured properly, we were able to tune to the pager frequency which is 169650kHz and capture a pager signal as it is shown in figure 2.

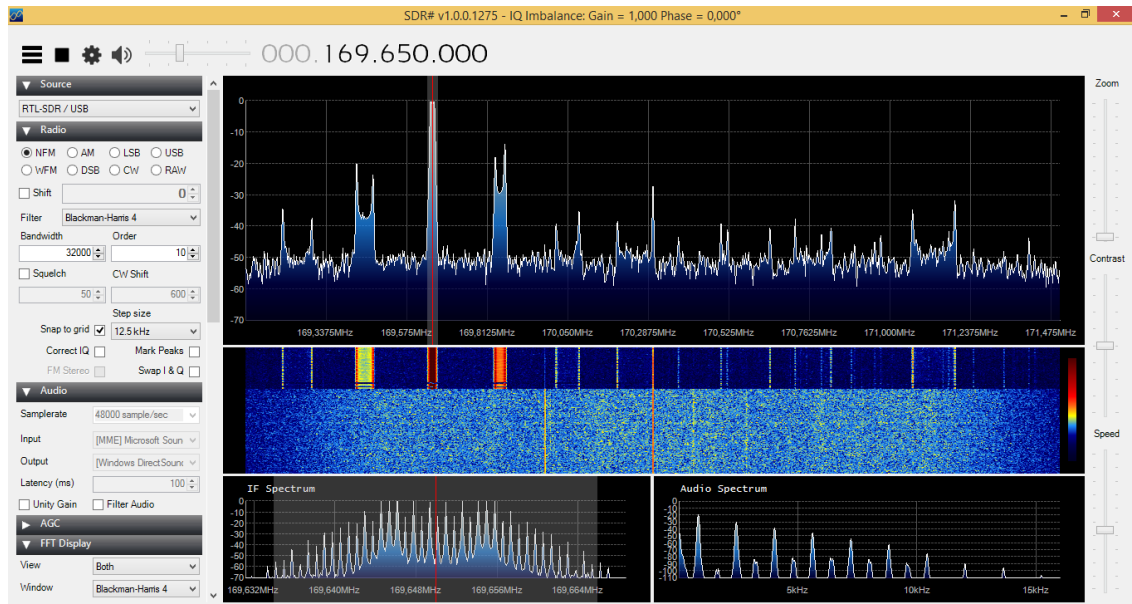


Figure 2: Pager signal with SDRSharp

The next step that we had to do, was to make sure that the signal that we were receiving was actually a pager signal. To verify that we installed the software PDW which decodes pager signals.

From the results we received (figure 3) we can see that the protocol used is FLEX-A with bit rate 1600bps.

Address	Time	Date	Mode	Type	Bitrate	Monitored Messages
1505600	11:38:21	08-03-17	FLEX-A	GROUP	1600	P 1 Geboubrand woning (middelbrand) (VK: 3) Akkerwinde 117 APW 9193 2232
2029568	11:38:21	08-03-17	FLEX-A	ALPHA	1600	P 1 Geboubrand woning (middelbrand) (VK: 3) Akkerwinde 117 APW 9193 2232
0726117	11:38:41	08-03-17	FLEX-A	ALPHA	1600	A2 Standplaats Maarsse Floraweg 17 3608BW 17 Maarsse
1423001	11:38:49	08-03-17	FLEX-A	GROUP	1600	B1 18175 Beatrix 3 Noord Banneweg 57 4204AA 57 Gorinchem RITHR 9067
1423375	11:38:49	08-03-17	FLEX-A	GROUP	1600	B1 18175 Beatrix 3 Noord Banneweg 57 4204AA 57 Gorinchem RITHR 9067
2029568	11:38:49	08-03-17	FLEX-A	ALPHA	1600	B1 18175 Beatrix 3 Noord Banneweg 57 4204AA 57 Gorinchem RITHR 9067
1520088	11:38:51	08-03-17	FLEX-A	ALPHA	1600	A2 Oostlaan 110 PAR 2641DV Directe inzet 15388 Ritnr: 39002
0920106	11:38:53	08-03-17	FLEX-A	ALPHA	1600	B2 Martinusweg 19 Zevenaar 6905AR 19 19530
1520432	11:39:13	08-03-17	FLEX-A	ALPHA	1600	B2 BMC-Me C08 Neuro / Lijnbaan 32 DRG 2612VA : 15432 Ritnr: 39004
0120160	11:39:17	08-03-17	FLEX-A	GROUP	1600	A1 13160 Rit 28396 Amsterdam Van Nijenrodeweg 1081EE 1094
0120399	11:39:17	08-03-17	FLEX-A	GROUP	1600	A1 13160 Rit 28396 Amsterdam Van Nijenrodeweg 1081EE 1094
2029568	11:39:17	08-03-17	FLEX-A	ALPHA	1600	A1 13160 Rit 28396 Amsterdam Van Nijenrodeweg 1081EE 1094
1523164	11:39:43	08-03-17	FLEX-A	ALPHA	1600	A1 Akkerwinde 117 APW 2403GR : (middelbrand) 16164 Ritnr: 39005
1533502	11:39:47	08-03-17	FLEX-A	SIL/TONE	1600	1
1520017	11:40:02	08-03-17	FLEX-A	ALPHA	1600	A1 Versetsheldenstraat 2 LSM 2264ME : 15117 Ritnr: 39006
1180000	11:40:08	08-03-17	FLEX-A	ALPHA	1600	TESTOPROEP HOOFDSYSTEM GMC BN (1)
1180000	11:40:08	08-03-17	FLEX-A	ALPHA	1600	TESTOPROEP BACK-UP SYSTEM GMC BN (2)
1520042	11:40:36	08-03-17	FLEX-A	ALPHA	1600	A2 DP2 Leidsechendam - Voorburg Via Donisetti 1 VBG 2272VK YMS 15142 Ritnr: 39007

Figure 3: Decoded pager signal from PDW

It is known that the FLEX-A model is using a 2 level FSK modulation and therefore the next step that we need to get done is to receive the signal and demodulate it. At first we decided to record the signal from SDRSharp and try to demodulate it in MATLAB. Due to the big size of the recorded data and to the high default sampling rate of SDRSharp that cannot be changed, MATLAB was not able to perform any calculations. At that point, we decided that we should record the signal from MATLAB as MATLAB enables us to specify the sampling rate and then try to demodulate that signal.

First of all, we installed the required packets of SDR to MATLAB and we plotted a small part of the signal in order to check that our code and configuration was working. The signal we received is shown in figure 4.

When it was the time to do the demodulation, we decided to first try to demodulate a known signal in order to check that our code is correct. Once, we will be sure that the demodulation code is working we will apply the code to the signal we receive from MATLAB. We know that our SDR dongle does the frequency shifting and filtering by itself and therefore we will be able to do the demodulation of the signal right after we receive it on MATLAB.

At this point we try to implement the MATLAB code for the demodulation. We were able to create a known signal with the known preamble 101010. This signal we generate, consists of the 6 preamble bits and 5 more random bits which represent the code. We use such a small signal size, as we want to see in detail what happens in every step of the demodulation and spot easily any errors. Also, we expect that the real signal will have 576 bits at the preamble. Before implementing the MATLAB code for the modulation and demodulation, we designed the FSK demodulation in simulink as shown in figure 5. The output of this shown in figure 6. After getting the results, we implemented this scheme in MATLAB via coding.

At the same time, we created a GNU file with all the steps that need to be done in order to achieve our final goal. The purpose of this file is to make us sure that if we follow all the steps that we included there, we will actually be able to get the data from the signal. The GNU radio is an interactive platform in Linux used for simulating communication systems using modular approach. It has a vast library containing various signal processing blocks. We first successfully interfaced RTL-SDR with GNUR radio and received and record the signal through it as shown in figure 7.

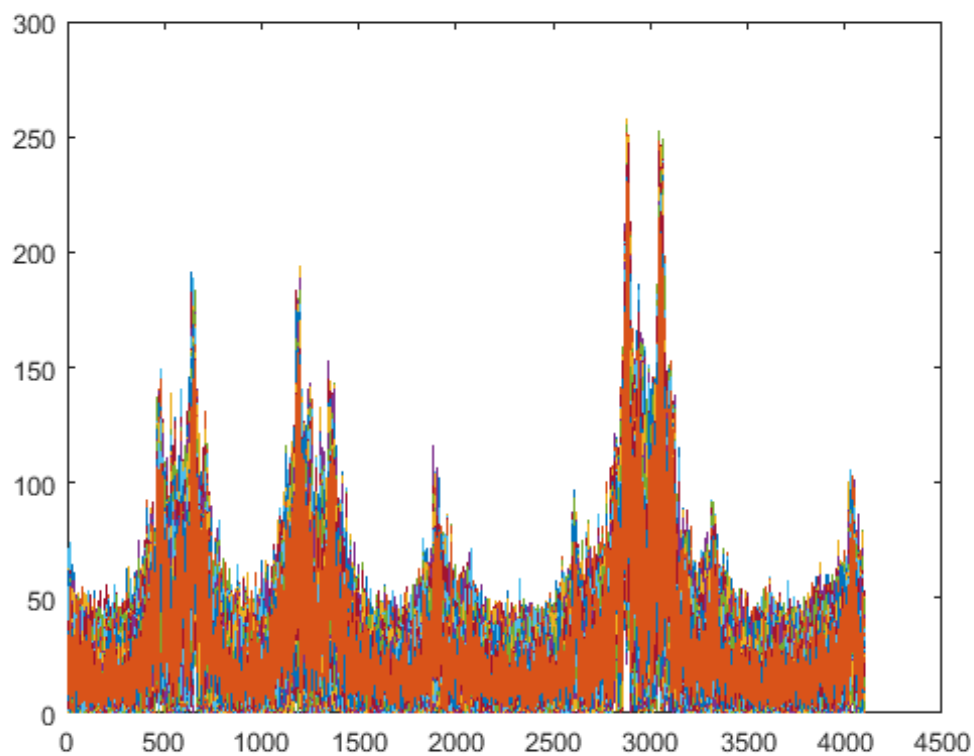


Figure 4: FFT of the received signal from MATLAB

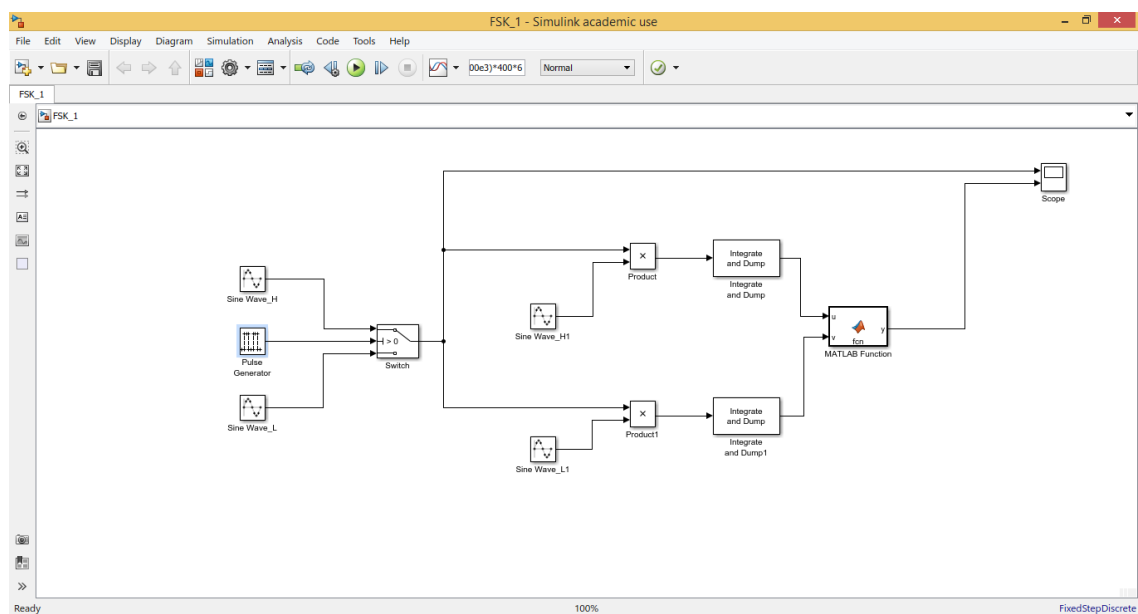


Figure 5: Diagram from Simulink

figure 8 represents the system used for demodulating the incoming signal. The recorded signal is first passed via low pass filter to select the desired frequency band. The filtered signal is

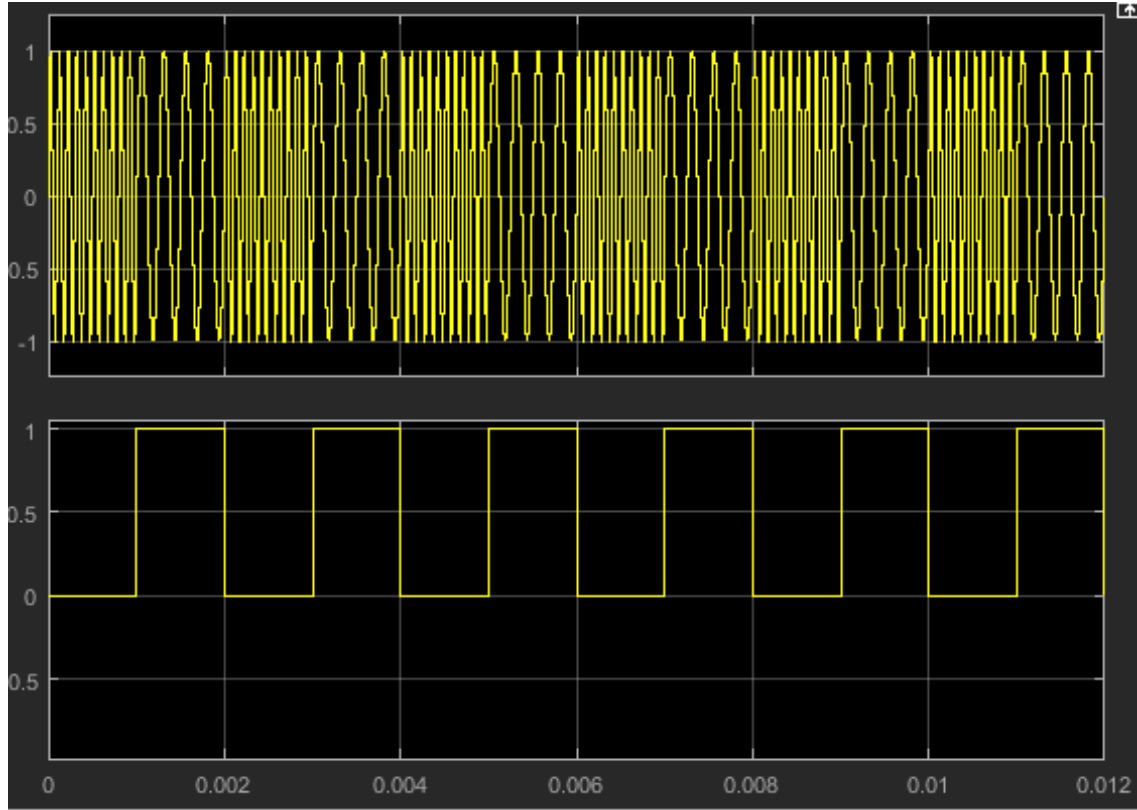


Figure 6: Output of FSK Modulator and De-modulator in Simulink

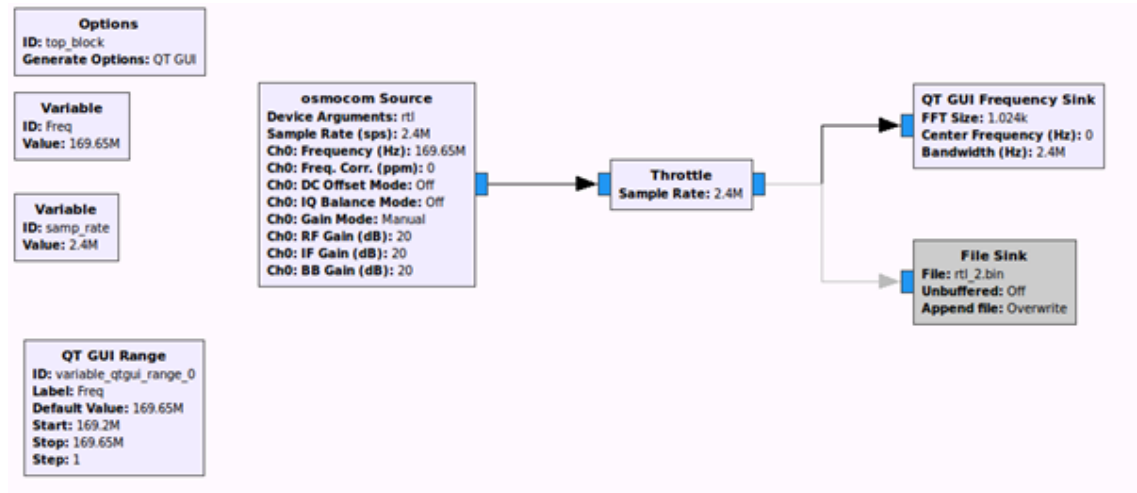


Figure 7: RTL-SDR Interface with GNURadio

then supplied to Quadrature demodulator block which is used to demodulate the FSK signal in GNUradio. The demodulated signal is then filtered to remove the noise from the signal. Then the clock recovery is performed using Clock recovery MM block. The output data is viewed via QT GUI Time sink block is shown in figure 9.

The bits are recovered but there are issues related to timing of the incoming bits. The time scale

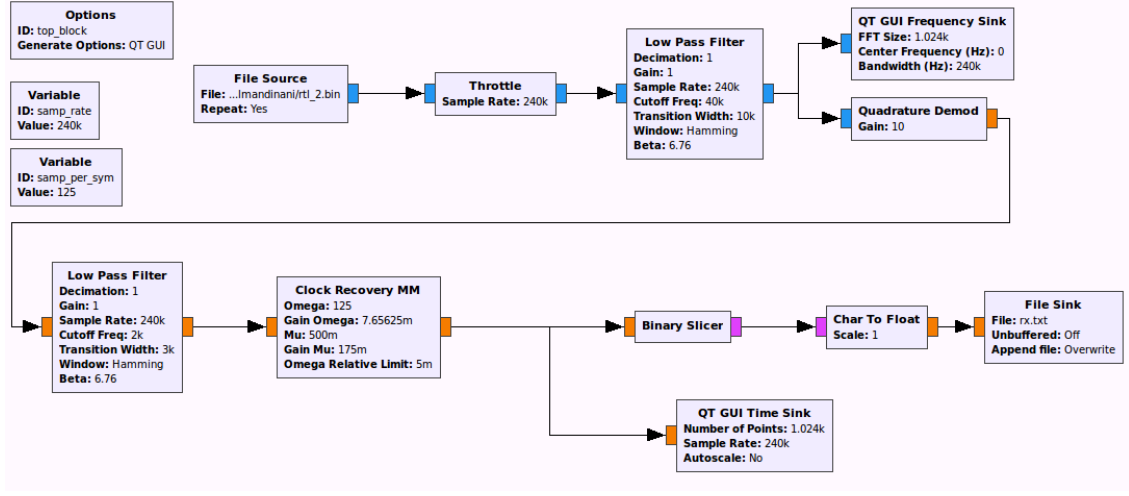


Figure 8: FSK demodulator for Pager signal

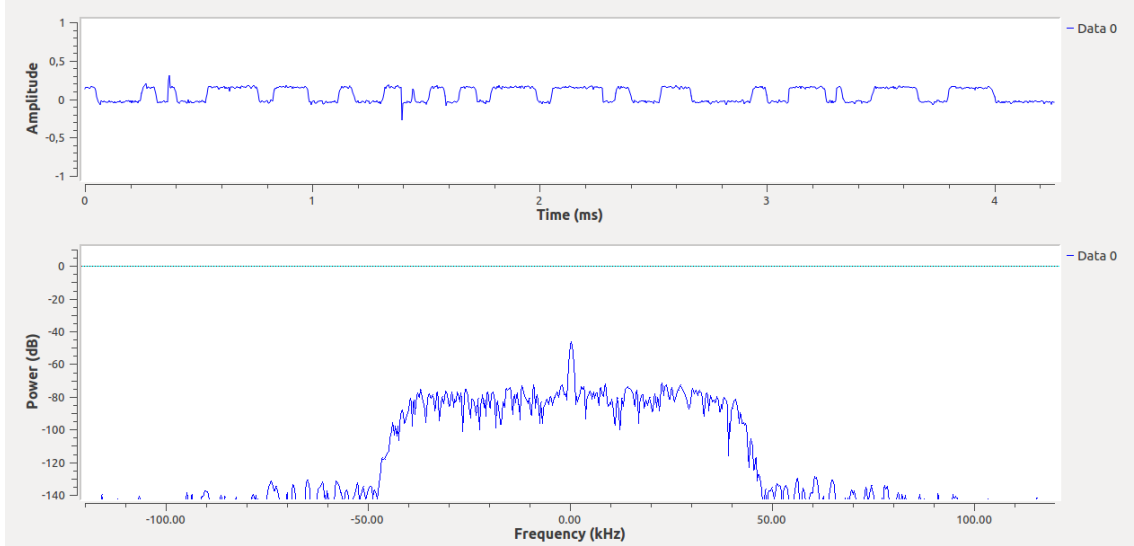


Figure 9: Output Data

in figure doesn't correspond to the data rate of FLEX signal which is 1.6 kbps. We are currently working on it to solve this issue.

Now that we are sure that the demodulation code we have is working, we should apply the code to the signal we receive from SDR to MATLAB. The difficulties that we expect to face during this step, is the noise from the signal and the unknown frequencies used for the modulation.

Once we demodulate that signal, we will then have to decode it. The challenging point to the decoding process will be understand which is the preamble.