EPFL SPACECRAFT TEAM

This project was carried out by a student from École Polytechnique Fédérale de Lausanne, member of the EPFL Spacecraft Team whose first space mission named CHESS is supported by the EPFL Space Center.

Conception and testing of an acquisition pipeline for a UHF Antenna

Author:
Adrien Dupont

Supervisor:
Hannes Bartle

Professor:
Jean-Paul Kneib

July 22, 2024

eSpace
EPFL Space
Center

Contents

1 Introduction

1.1 CHESS

The Constellation of High-Performance Exospheric Science Satellites is the main mission of the student-led EPFL Spacecraft Team. This mission comprises two 3U CubeSats equipped with scientific instruments developed by the universities of Bern and Zurich, aiming to measure the chemical composition of the upper atmosphere of our planet. These two satellites will be placed in two different orbits, one circular and the other elliptical, to ensure complementary coverage of the measurements. This project focuses on the first one, Pathfinder 1, to be launched in 2027 into a sun-synchronous circular orbit.

Telecommunication for both satellites will be divided into two independent modules: UHF and X-band. The first one will be used to downlink telemetry and uplink telecommands, while the second will be used to downlink the heavy scientific data at a higher data rate. With only a few minutes of communication windows each day, a reliable UHF Ground Station is essential to operate the satellites.

1.2 Project scope

This project follows the semester projects of Juliette Challot [8] and Bastien Labat [13], which were focused on designing the GS and its operation. Following the UHF Ground Station's deployment in 2023 on the ELB roof, the focus is now oriented on the acquisition pipeline. This implies receiving signals, demodulating them accordingly, and processing the information. In this scope, the objectives of this project are:

- Understanding how information is translated into a signal, from modulation to structuring
- Using the GNU Radio software, simulate a modulation and demodulation script using a basic modulation scheme and test it on SatNOGS recordings
- Adapting this script to demodulate real-time satellite communications through the UHF antenna
- Testing the acquisition pipeline and evaluating its properties

1.3 Abbreviated terms

ALT	Altitude
AM	Amplitude Modulation
AZ	Azimuth
BER	Bit Error Rate
CHESS	Constellation of High-Performance Exospheric Science Satellites
COTS	Commercial off-the-shelf
CSP	CubeSat Space Protocol
EIRP	Effective Isotropic Radiated Power
FCS	Frame Check Sequence
FM	Frequency Modulation
FSK	Frequency Shift Keying
GFSK	Gaussian Frequency Shift Keying
GMSK	Gaussian Minimum Shift Keying
GS	Ground Station
HDLC	High-Level Data Link Control
IQ	In-phase Quadrature
LEO	Low Earth Orbit
LHCP	Left-Handed Circularly Polarised
NRZI	Non Return to Zero Inverted
OOK	On Off Keying
PDU	Protocol Data Unit
PSK	Phase Shift Keying
PSD	Power Spectral Density
QPSK	Quadrature Phase Shift Keying
RHCP	Right-Handed Circularly Polarised
SDR	Software Defined Radio
SNR	Signal to Noise Ratio
TLE	Two-Line Elements
UHF	Ultra-High Frequency

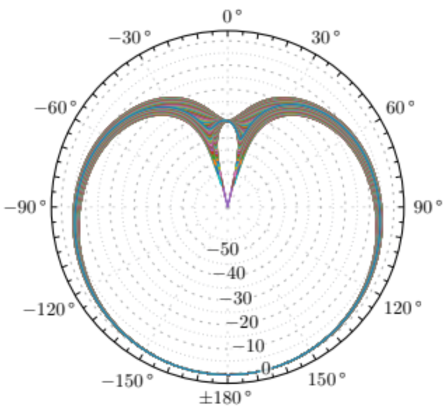
2 UHF Modules

2.1 UHF - Satellite module

For UHF telecommunication, both the transceiver and the antenna are commercial off-the-shelf (COTS) components.
For the transceiver, the **Endurosat UHF Transceiver II** [10] will allow half-duplex communication from 430 to 440 MHz, amateur frequencies suitable for our mission. The supported modulation schemes are 2GFSK (default scheme), OOK, GMSK and FSK (see section 3.1). On the transmission side, the Tx power is set to a default value of 1W but can be raised to 2W. Two different communication protocols are supported:

- EnduroSat Protocol Stack I (ESPS 1) protocol, designed to read telemetry and run commands in the UHF module (note: the ESTTC protocol is now deprecated)
- An AX.25 configurable telemetry beacon broadcast, on which we will be focusing in this project (see section 3.4). The popularity of this protocol in amateur satellite telecommunication facilitates the testing of our acquisition pipeline on active satellites. This protocol uses bit stuffing, scrambling, and encoding, detailed in section 3.4.

A key feature of the **Endurosat UHF Antenna Type III** [9] is its omnidirectional radiation pattern, as seen in Figure 2.1. This property facilitates the implementation of the AX.25 beacon, as packets will be received regardless of the CubeSat's orientation. Since the data sent in this beacon is critical, having constant access to the satellite's status is essential for this mission. The configurable beacon can transport up to 616 bits of payload data (see Figure 3.13), necessitating the selection of the most critical bits from the 19,216 bits of housekeeping data listed in the Mission Operations Plan [14].



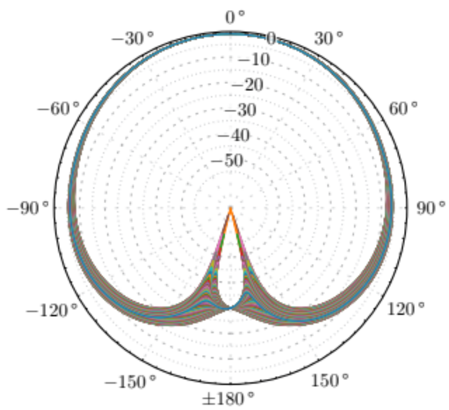


Figure 2.1: LHCP (left) and RHCP (right) Radiation Pattern of the Endurosat UHF Antenna Type III when mounted on a 3U Structure

2.2 UHF - Ground Station

2.2.1 Acquisition pipeline

On the hardware side, the acquisition pipeline is designed to be composed of an antenna with its phase line, a pre-amplifier, and a Software Defined Radio (SDR).

Antenna: Yagi Uda 2x16

Two Yagi Uda antennas are deployed in this ground station (GS) to cover both downlink and uplink operations. These antennas measure 3.18 meters in length, support a power output of 150W, and provide a gain of 16.5dBi. Both antennas can be configured for either right-hand circular polarization (RHCP) or left-hand circular polarization (LHCP). The satellite tracking is controlled by a Yaesu 5500DC Alt-Az rotator, interfaced with tracking software (e.g., Gpredict).

435MHz Cubesat Filtered Preamp

The signal received through the Yagi Uda antennas is very weak and requires immediate amplification to prevent additional signal loss due to noise. For this purpose, a pre-amplifier designed for UHF amateur frequencies is housed in a waterproof box and directly connected to the phase line. It provides a gain of 19dB at 435 MHz and has a noise figure of 0.75dB.

ADALM PlutoSDR

A Software Defined Radio (SDR) corresponds to a radio communication system where traditional hardware signal processing components are replaced by software processing, offering more flexibility and potentially reducing costs. The PlutoSDR is an open-source receiver and transmitter that operates at frequencies from 325 MHz to 3.8 GHz. It supports both half-duplex and full-duplex mode; however, adding a second PlutoSDR could facilitate communication, alternating rapidly between uplink and downlink sequences. While primarily used for educational purposes, the PlutoSDR is capable of functioning in mission-critical applications. A GPSDO clock has been installed to improve synchronization, although it currently does not override the default clock.

2.2.2 Operations

Figure 2.2 displays a simplified functional diagram of the UHF ground station's operation. A control PC has been installed in ELB 216, acting as our control room. This computer has remote access to the electrical cabinet's Raspi, and is used for heavy signal processing, mission scheduling, and general centralized operation. To transfer these signal processing tasks from the Raspi to the control PC, a TCP transmission of the raw received signal is set up through the EPFL network (see section 4).

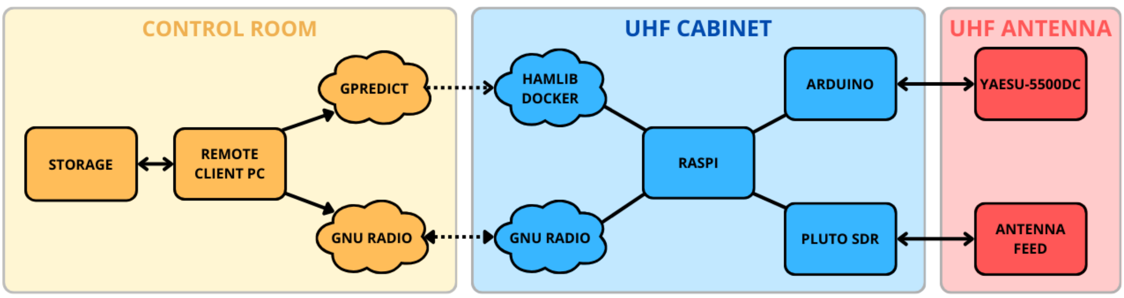


Figure 2.2: Functional Diagram of the UHF Ground Station

2.3 Softwares

2.3.1 GNU Radio

GNU Radio is an open software development toolkit very useful in signal processing. It provides software blocks to facilitate the design of complex signal processing applications, both on the modulation or demodulation side. To be able to interface GNU Radio with our PlutoSDR, the SoapySDR library [1] is used, providing direct Sink and Source blocks. Lastly, a very useful out-of-tree module is the gr-satellites library [11] which implements diverse signal operations used in satellite communications.

2.3.2 SatNOGS

SatNOGS (Satellite Networked Open Ground Station) [2] is an open-source project providing a collaborative network of satellite ground stations, which can be used to receive data from satellite communications. This network was very useful during this project to test demodulation scripts on communication recordings before using them on our actual antenna. Once the UHF Ground Station is fully operational, making it accessible to the world through this network would be a great opportunity.

2.3.3 GPredict

Gpredict is an open-source software application designed for satellite tracking and orbit prediction. Based on the software Predict, it offers a graphical interface, facilitating operations (see Figure 2.3). Orbital information on the tracked satellite (TLE) is either updated through the network (namely Celestrak [3]) or manually, making it easy to track any satellite. On one side, it can be directly interfaced with the rotor's controller for the tracking, and on the other, interfaced with GNU Radio for real-time Doppler correction.

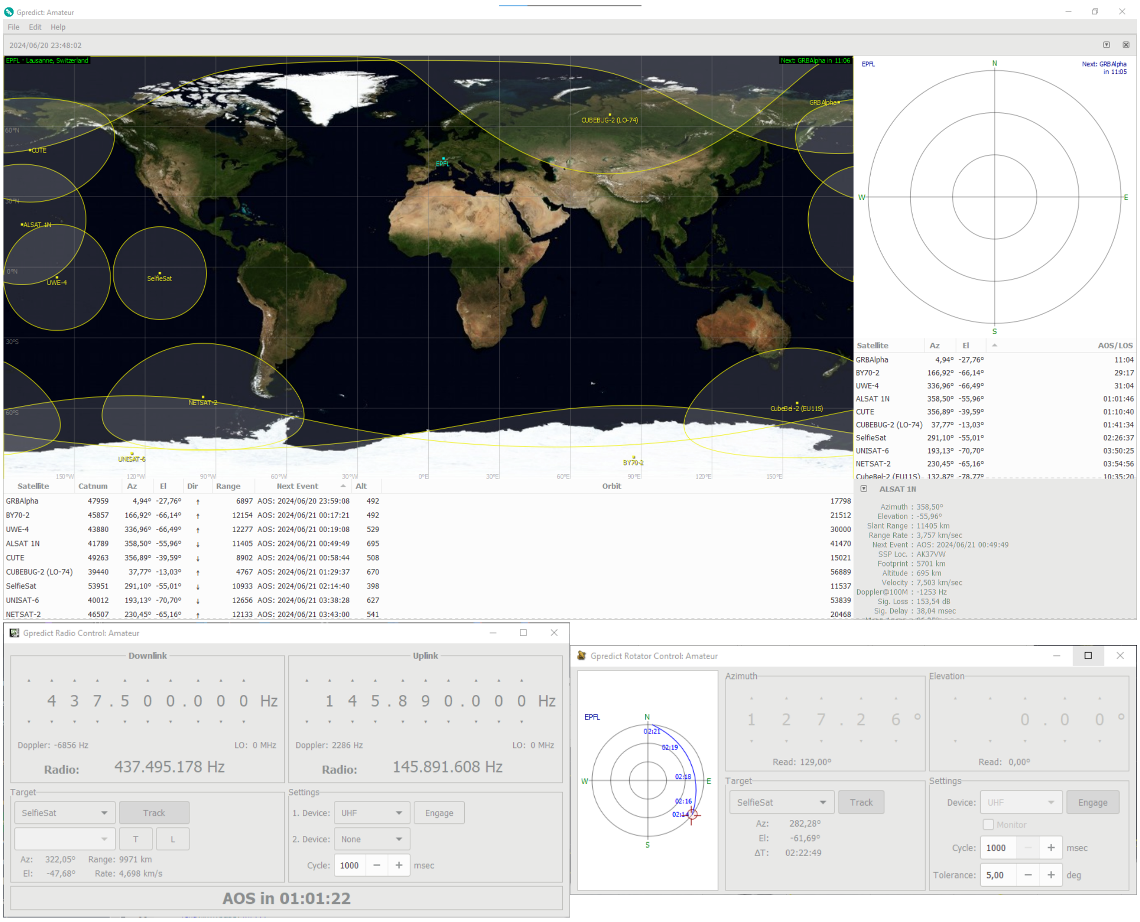


Figure 2.3: Gpredict Graphical User Interface (on top orbital information of all chosen satellites, on the bottom left the Frequency control and Doppler correction, on the bottom right the rotor tracking control)

3 Signal Architecture

3.1 Modulation schemes

3.1.1 Introduction to modulation

Modulation corresponds to the process of transmitting information on a carrier wave by modifying through time certain parameters. The carrier wave corresponds typically to a high-frequency wave that can travel better through air or any other medium. On the other side, the baseband signal corresponds to the unmodulated signal that directly displays the information to transmit. Generally speaking, a signal is characterised by three parameters : its frequency, amplitude and phase. Every modulation scheme will be based on the variation of at least one of these parameters [16].

Amplitude Modulation

This modulation scheme involves modifying the intensity of the carrier to transmit information. The two main resulting schemes are On Off Keying (figure 3.1), where the second state is an intensity of 0, and Amplitude modulation (figure 3.2). OOK corresponds to the simplest modulation, as a 1 corresponds to an active signal and a 0 no signal.

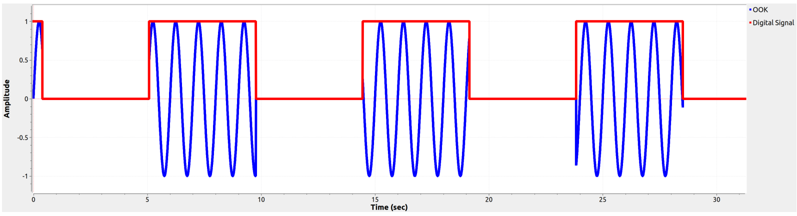


Figure 3.1: On Off Keying (OOK) modulation

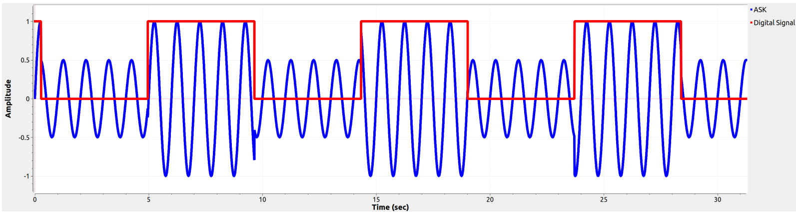


Figure 3.2: Amplitude Modulation (AM)

Frequency Modulation

FM modulation probably is the most known scheme, as it is widely used in commercial radio broadcasting. In these applications, the carrier's frequency varies according to the base band signal's frequency variations. In signal processing terms, it corresponds to simply multiplying the high frequency carrier signal with the base band signal. It can also be very easily demodulated, as a simple frequency shift gets rid of the carrier signal and brings us back to base band with the original signal. This modulation scheme is thus preferred for voice transmissions due to its very simple implementation.

From a CubeSat telecommunication point of view, the information to transmit corresponds to bit sequences. For this purpose, the modulation of the frequency alternates between discrete frequency values, each representing a value (figure 3.3). This modulation scheme basis is very common in UHF amateur frequencies due to its robustness and simple implementation.

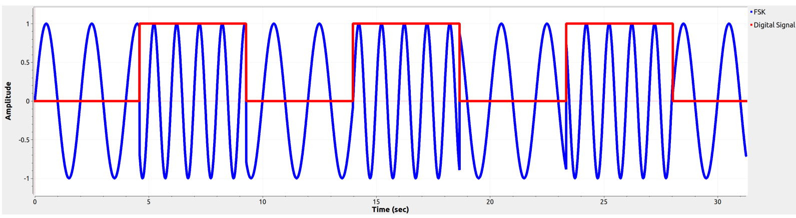


Figure 3.3: Frequency Shift Keying (FSK) modulation

Phase Modulation

The last parameter that can be varied is the signal phase. Phase Shift Keying modulates a signal by shifting by a discrete number the phase of a signal to display a change of bit value (see figure 3.4). This modulation scheme requires a more complex implementation due to synchronization requirements, although it presents overall better performances than the other modulation schemes. This advantage makes it often a better choice at even higher frequencies (L-band, X-band, etc) where the objective is to maximize the data rate and minimize error.

The Quadrature Phase Shift Keying (QPSK) uses the fact that a complex signal is composed of an in-phase (I) signal and a quadrature (Q) signal. In this scheme both IQ signals are modulated, increasing the information transmitted per second and thus the data rate. This modulation scheme will for example be used for X-band communication in the Chess mission to downlink scientific payload at a very high data rate.

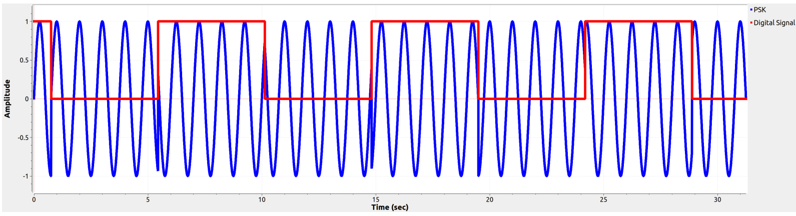


Figure 3.4: Phase Shift Keying (PSK) modulation

In these modulation examples, only two levels were used : two amplitudes, two discrete frequencies, two phase shifts. The number of levels used to modulate a signal is generally placed before the scheme name (ex 2FSK, 4PSK, etc...). Using multiple levels offers a higher data rate as more information can be transmitted with less parameter variations. While also providing a better spectral efficiency, its is also less robust to noise and more complicated to implement.

Lastly, the Baudrate corresponds to the number of symbols transmitted per second. In binary modulation schemes, this corresponds to the data rate as one variation corresponds to one bit transmitted. However, in schemes with multiple levels, the data rate corresponds to the number of symbols multiplied by the number of parameter levels.

3.1.2 Frequency Shift Keying

As described in section 2.1, the transceiver only supports Frequency Shift Keying based modulation schemes. For this purpose, a basic full FSK modulation and demodulation GNU radio has been developed to understand every step of the signal processing. The modulation part of the GNU radio script is displayed in figure 3.5.

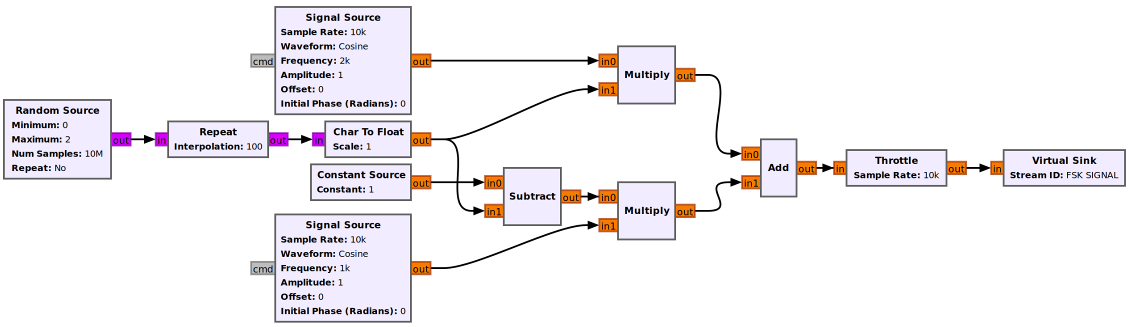


Figure 3.5: GNU Radio script of FSK modulation

In this sequence, a random source continuously starts by sending bits to be modulated. Two signal sources then create a signal of respectively 1kHz and 2kHZ. If the random bit source transmits a 1, the 2kHz signal will be the one transmitted. On the other side, a 0 will transmit the 1kHz signal (the subtract module transforms the 0 into a 1). Both signals are then combined to form a FSK modulated signal. The throttle block is used in order to set an upper limit of sampling rate, especially in scripts containing purely simulation blocks (and no SDR input). The sample rate is thus set at 10kHz, five times the highest frequency to reconstruct. This modulation script produces the signal displayed in figure 3.6 where we can clearly observe both alternating discrete frequencies.

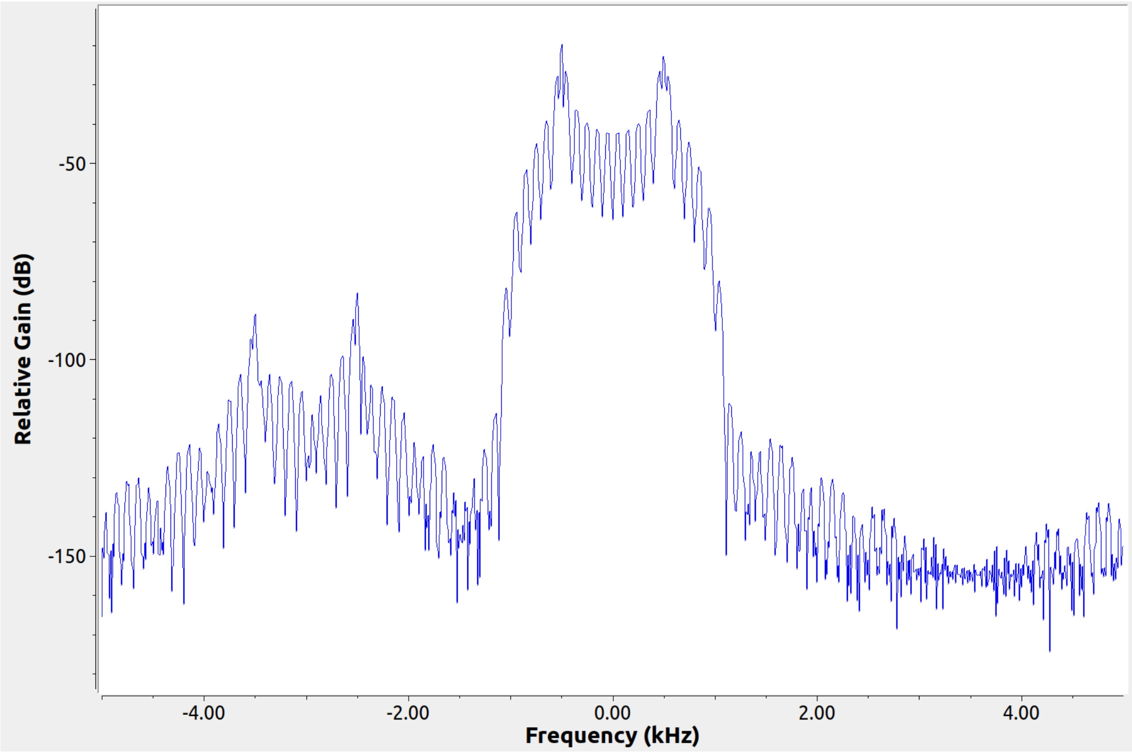


Figure 3.6: Power Spectral Density at base band of a FSK modulated signal

Figure 3.7 represents the FSK demodulation script taking as input the modulated signal from script 3.5.

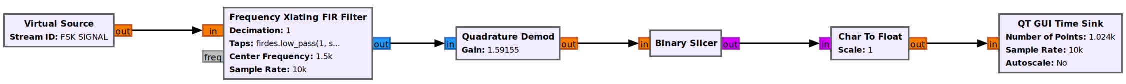


Figure 3.7: GNU Radio script of FSK demodulation

The following blocks are implemented :

Frequency Xlating FIR Filter
This block starts by performing a frequency shift to reach the centre of the signal, in the middle of both discrete frequency levels (here 1.5kHz). The second feature is a Finite Impulse Response filter who removes unwanted noise and interferences.

Quadrature Demod
This demodulation block can be used to demodulate any FSK based modulation scheme. The input is a complex base band signal (now that the frequency has been shifted to the center of the signal we are back at base band). The output corresponds to the frequency where the intensity is the highest, multiplied by a gain. The gain corresponds to [4]:

$$\text{gain} = \frac{\text{sample\_rate}}{2\pi \times FSK\_deviation}$$

This gain will take the amplitude of the binary output to +0.5 and -0.5 (see figure 3.8).

Binary Slicer

The output of the Quadrature Demod block corresponds to a float output and not a binary output. To transform this binary looking signal into a true binary signal, we use a Binary Slicer block (fig 3.8).

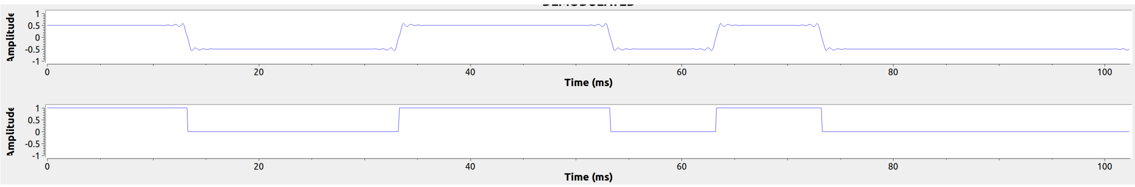


Figure 3.8: GNU Radio script of FSK demodulation

3.1.3 Gaussian Frequency Shift Keying

In FSK, the abrupt frequency transitions leads to a spectral splatter, causing the signal to occupy a larger bandwidth. Indeed, these changes can be seen as Dirac delta impulses in the frequency domain. Gaussian Frequency Shift Keying uses a Gaussian filter to smoothen these frequency transitions, resulting in a better spectral occupation (see section 3.2.2). Additionally, this scheme ensures a continuous phase throughout the signal, avoiding phase shits and consequently reducing spectral occupation

3.1.4 Gaussian Minimum Shift Keying

Minimum Shift Keying is a modulation scheme also based on FSK. However, here the frequency deviation is lowered to the smallest possible value while allowing the signal to remain perceivable. This optimal deviation corresponds to exactly half the bit rate, hence the "minimum" shift keying. This scheme also ensures a continuous phase throughout the signal. Similarly to GFSK, this scheme ensures a continuous phase throughout the signal and implements a Gaussian filter to avoids abrupt frequency transitions.

3.2 Comparison

3.2.1 BER

The Bit Error Rate of modulation schemes is essential in the establishment of a link budget, as a target BER corresponds to a specific energy per bit to noise power spectral density ratio ($\frac{E_b}{N_0}$). In this section we compare the BER of non-coherent FSK, GFSK and GMSK.
Coherent FSK corresponds to a form of FSK where in addition to the magnitude, the phase is also used to transmit information. This modulation scheme, although it presents better overall BER performance, will not be explored here due to its implementation needs. Additionally, the Doppler frequency shift present in LEO orbits complicates phase coherence.

The BER for FSK is can be theoretically described by the following formula [12] :

$$\mathrm{BER}_{\text{Non-Coherent FSK}} = \frac{1}{2}e^{-\frac{E_b}{2N_0}}$$

$$\text{BER}_{\text{Coherent FSK}} = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

where $Q(x)$ is the Q-function defined as:

$$Q(x) = \frac{1}{2}\text{erfc}\left(\frac{x}{\sqrt{2}}\right)$$

In the case of a GFSK or GMSK modulation script, the BER should be tested and evaluated through a comparison of input and output streams. For this purpose, a BER block exists in GNU Radio. However, the values displayed by this block seemed incoherent with theoretical values. While simulating the BER for the GFSK modulator fetched from the gr-satellites, I decided to rather use the `berr` function present in Matlab. The first step was to plot the correlation between the input and output stream of bits to find the delay resulting from the modulation and demodulation processing. The results of this simulation are presented in figure 3.9. The estimated curve has a higher BER than other theoretical approximations. This excess of errors could come from a bad parameter design for the GFSK modulator and demodulator or directly from the built in modulation GFSK blocks. Moreover, non coherent modulation schemes are very sensible to delays, which can be hard to estimate precisely .

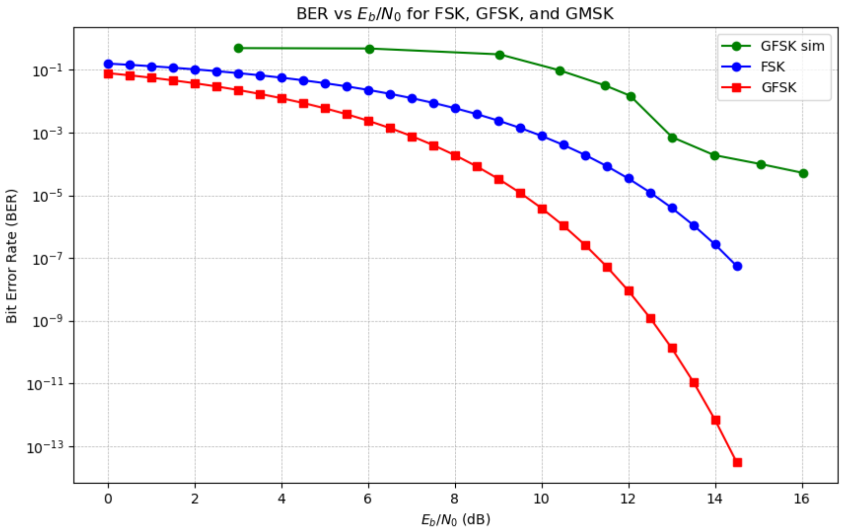


Figure 3.9: Spectral occupation of FSK, GFSK and Simulated GFSK modulation schemes

For GFSK and GMSK, the BER can only be approximated as it is highly dependent on the modulation parameters [18]. GMSK being a special case of GFSK where the modulation index is equal to 0.5, both BERs are generally very similar [17] :

$$\text{BER}_{\text{GFSK}} \approx \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_b}{2N_0}}\right)$$

In general, GFSK and GMSK have better BER performances than non coherent FSK due to their continuous phase nature and Gaussian filtering. On the other side, coherent FSK offers a better performance than other FSK variations, due to the use of both amplitude and phase to transmit information.

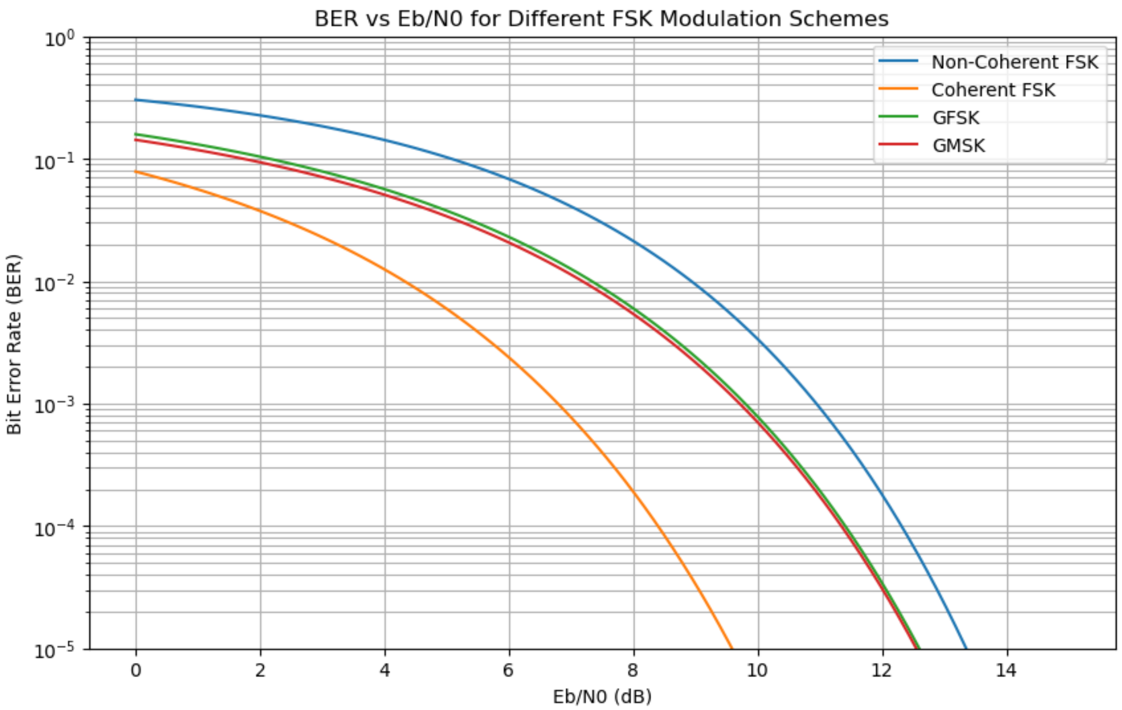


Figure 3.10: Spectral occupation of FSK, GFSK and GMSK modulation schemes

3.2.2 Spectrum occupation

The spectral efficiency of a modulation scheme needs to be taken into account, as lower efficiency requires a larger bandwidth allocated to the satellite for communication. Depending on the frequency domain and application, this spectral occupation can be more or less critical, significantly impacting the choice of modulation scheme.

Figure 3.11 compares the spectral efficiencies of FSK, GFSK, and GMSK modulations at similar frequencies. The simulation utilized the FSK modulator mentioned in section 3.1.2, combined with GFSK and GMSK modulators from the gr-satellites library.

FSK exhibits the lowest efficiency due to the spread of Dirac impulses on each side of the modulated signal. GFSK maintains the same modulation bandwidth as FSK but mitigates lateral noise propagation through Gaussian filtering. In contrast, GMSK achieves a noticeably smaller bandwidth compared to the others and demonstrates the best possible spectral efficiency for the given bit rate.

3.2.3 Implementation

The last aspect taken into account in this modulation scheme comparison is the complexity of implementation, which should not be underestimated especially in a student-led mission. In this project, we will focus on non coherent FSK, as is it lighter to implement and easier to update to GFSK and GMSK. Moreover, it is possible to demodulate GFSK signals with FSK demodulation scripts, although not optimally. GMSK, due to its optimal nature, requires additional steps of implementation and more precise demodulation.

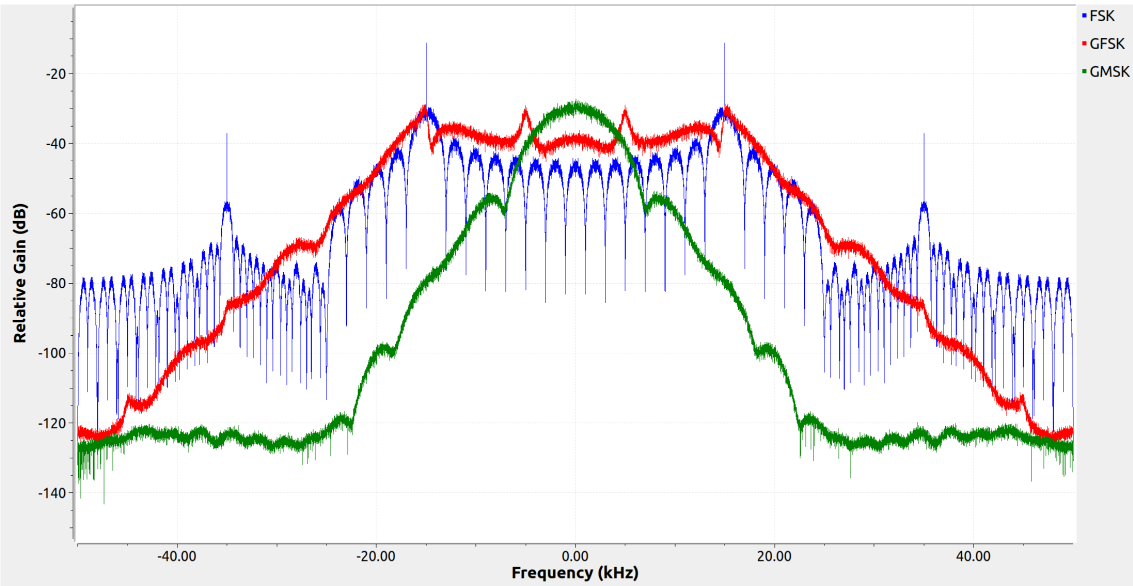


Figure 3.11: Spectral occupation of FSK, GFSK and GMSK modulation schemes

3.3 Detection and Framing

Determining whether or not the output of 1s and 0s of a demodulator actually transmit any information at all is crucial, especially with very small transmissions such as beacons. The process of data framing aims at transmitting sequentially information in packets rather than transmitting a continuous stream of data, resulting in a better transmission performance. These packets are initiated with a Header and terminated with a Trailer, both composed of a flag (a specific sequence of bits).

A very common example of framing protocols is the High-Level Data Link Control (HDLC) [5].

HEADER FLAG	ADRESS	CONTROL COMMAND	PACKET	FRAME CHECK SEQUENCE	TRAILER FLAG
01111110 (7E)	8 BITS	8 BITS	...	16 - 32 BITS	01111110 (7E)

Figure 3.12: HDLC Framing Structure

The Flag used in both the HDLC Header and Trailer is 0111110 or 7E in hexadecimal. The gr-satellite library contains blocks used to frame and deframe packets using this HDLC protocol (see section 4.1). The deframing block outputs only the packet structure without the reserved HDLC bytes displayed in figure 3.12.

The detailed packet structure is determined by other data link layer protocols such as the well known Ethernet or AX.25 (see section 3.13).

The Frame Check Sequence situated at the end of the packet corresponds to a mechanism of error detection. If after receiving a packet the FCS labels the packet as corrupt, the information is discarded and the specific packet can be transmitted a second time. This error protection is useful in transmitting through channels of great variability (bad weather conditions or unexpected interferences for example).

Another example of a very common Framing protocol, the CubeSat Space Protocol, is detailed in section 4.2.

3.4 Packet structure

Once the received packet is deframed, a data link layer protocol is needed to structure in detail the actual payload information.

Figure 3.13 displays the AX.25 protocol used for our transceiver's configurable telemetry beacon.

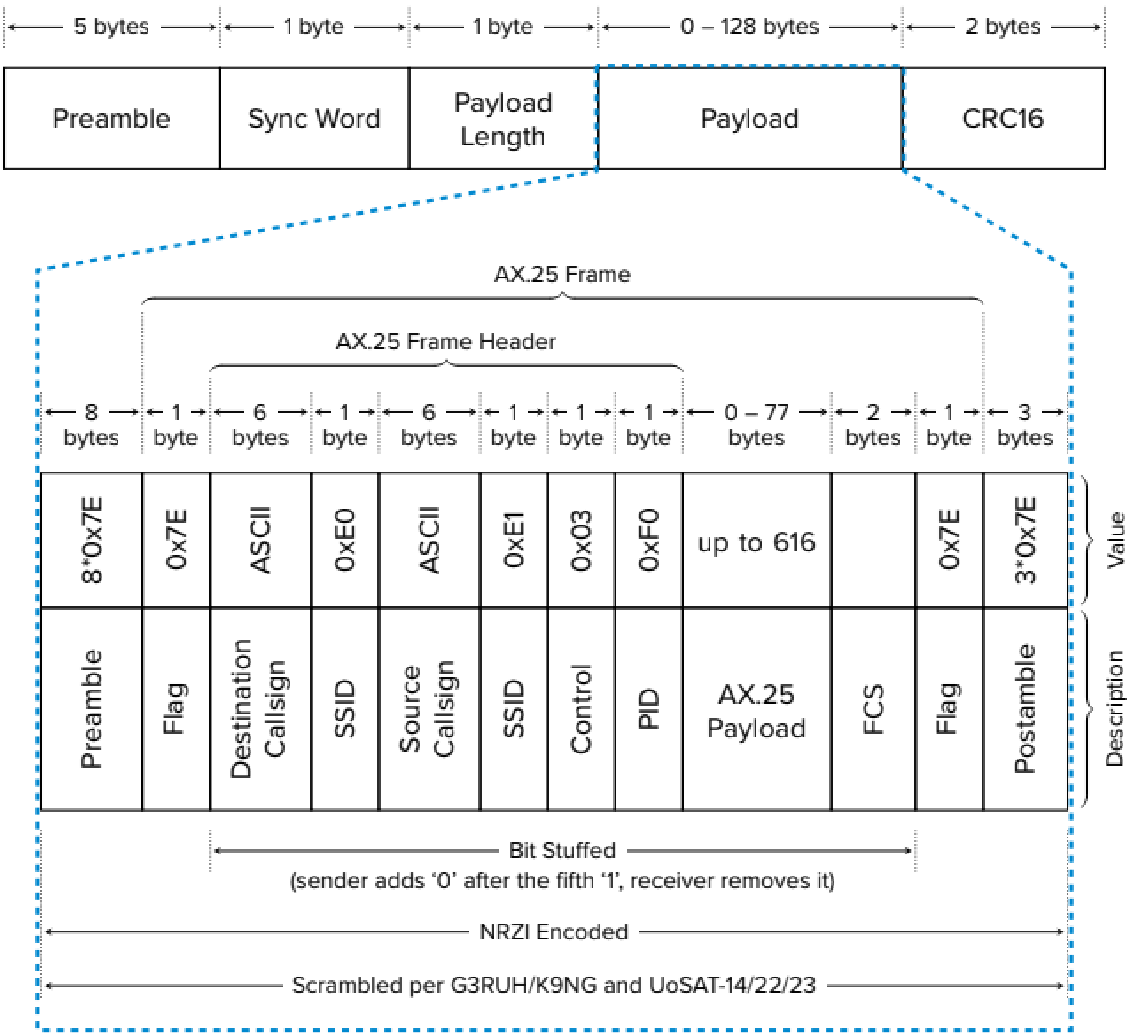


Figure 3.13: AX.25 Protocol UI Frame Structure

This AX.25 protocol uses three bit operations to optimise transmission performance :

Bit Stuffing

Throughout a packet, the header and trailer flag 01111110 could appear unintentionally in payload data. The role of bit stuffing is avoiding this ambiguous situation where the receiver could read this end of packet flag and lose the rest of the transmitted packet. For this, A 0 is added after a sequence of five 1s. From the receiver side, if a sequence of 5 consecutive 1s is encountered and followed by a 0, this 0 is removed in order to return to the original data.

NRZI Encoding

Non-Return-to-Zero Inverted encoding is a binary signal mappingn technique aiming at reducing errors due to long sequences of identical bits. For this, a 1 will represent a transition between the precedent and the next bit of the original signal, whereas a 0 represents a signal remaining at the same level (for example, 1001101 would become 10111011).

G3RUH Scrambling

Similarly to NRZI encoding, scrambling aims at avoiding long sequences of identical bits that often cause synchronization problems. The data is modified by a predefined random sequence based on the following polynomial : $1 + x^{12} + x^{17}$. More precisely the the transmitted bit is the EXOR of the current data bit, the bit transmitted 12 bits prior and the bit transmitted 17 bits prior as well. The polynomial known on both the transmitter and receiver side, descrambling is made this time with a XOR. This specific scrambling method is very common in amateur satellite communications.

4 Signal Acquisition

4.1 Acquisition Scripts

The following script combines interfacing with the PlutoSDR, FSK demodulation, and real-time frequency tracking.

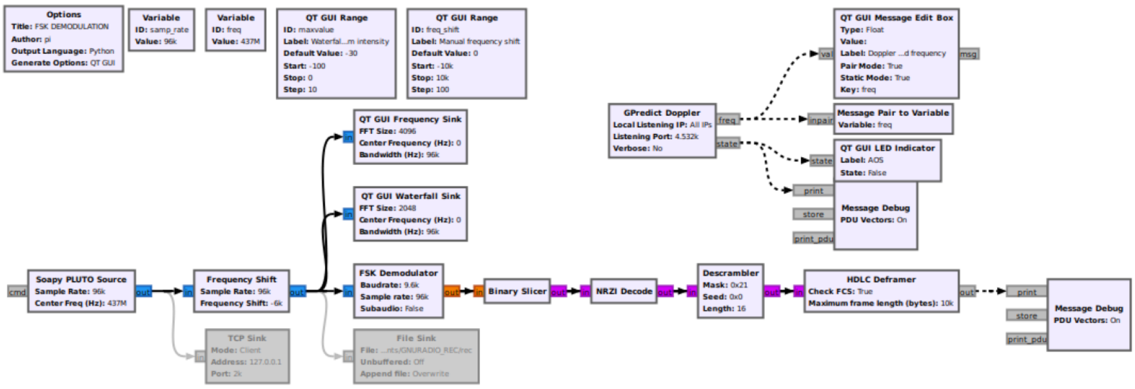


Figure 4.1: FSK acquisition script interfaced with the PlutoSDR and Gpredict

The following blocks are used in this acquisition script:

Soapy PLUTO Source
: This block interfaces the script with the PlutoSDR, providing input parameters such as the sample rate and the center frequency for the local oscillator. The output is an IQ signal from the measurements of the SDR. For this FSK demodulation, the sample rate is set to 96 kHz, which is ten times the expected symbol rate. Although most FSK beacons have a baud rate of 9600, the sample rate should be adapted for each application. According to the Shannon-Nyquist theorem, this sample rate should be at least twice the desired signal frequency. However, it is generally set at five to ten times the symbol rate. The bandwidth parameter is set to the sample rate to ensure efficient use of computational resources. The desired center frequency for signal acquisition is transmitted to this block through a variable updated by the Gpredict Doppler block. If another SDR is used in the future, the SoapySDR library contains other SDR interfacing blocks.

Frequency Shift
: After testing this script on multiple satellite passes, a frequency offset appeared in certain measurements, which can be explained by inaccuracies in the TLE information provided by the network. Manual correction can be applied to center the signal. The default frequency shift is zero.

FSK Demodulator
: This demodulation block, fetched from the gr-satellites library [11], replaces the Frequency Xlating FIR Filter and the Quadrature Demod blocks from Section 3.1.2. FSK signals are detected and converted into floating point values. The input is the baud rate, set at 9600. This demodulator can work with both float and complex inputs.

Binary Slicer
: The floating point values are converted into binary data (0 or 1).

NRZI Decode

This block performs NRZI decoding as described in Section 3.4.

Descrambler

This block descrambles the bits to retrieve the original signal. The descrambler uses the G3RUH descrambling algorithm, which is represented as the mask `0x21` and a length of 16.

HDLC Deframer

This deframer takes the input bit stream and converts it into packets. These packets are sent out as Protocol Data Units (PDU), which are now decoded data streams.

Message Debug

This block prints messages to the GNU Radio terminal for easy debugging. This block is temporary, as currently we do not process the packets and only display them in a terminal, as seen in Figure 4.2.

```
***** VERBOSE PDU DEBUG PRINT ******
pdu length =         22 bytes
pdu vector contents =
0000: 05 00 85 01 00 10 00 00 00 00 03 d2 20 6a 00 ff
0010: 5a 57 41 81 d4 72

***** VERBOSE PDU DEBUG PRINT ******
pdu length =         22 bytes
pdu vector contents =
0000: 05 00 85 01 00 10 00 00 00 00 03 d2 20 73 00 00
0010: 5a 57 a2 d0 fa 34

***** VERBOSE PDU DEBUG PRINT ******
pdu length =         22 bytes
pdu vector contents =
0000: 05 00 85 01 00 10 00 00 00 00 03 d2 20 73 00 01
0010: 5a 57 07 91 68 4a
```

Figure 4.2: Example of Protocol Data Unit (PDU) reception

Gpredict Doppler

Real-time Doppler correction is essential in LEO satellite communication as the frequency offset reaches 10 kHz for most passes. Compared to the frequency deviations and bandwidths used in FSK modulation, this offset is non-negligible. This block is fetched from the gr-gpredict-doppler library [6] and offers the possibility to modify a variable through a Message Pair to Variable block. The frequency input is sent from the Gpredict Radio Control running on the remote client PC.

TCP Sink

This block establishes the TCP transmission with the other instance of GNU Radio running on the remote client PC. Here the block is disabled for demonstration purposes.

File Sink

This block records the IQ samples of the signal throughout the acquisition. Logging and storing this information is crucial in a mission, as it offers the opportunity to reprocess the signal later in case of poor demodulation. However, depending on the sample rate, these files can become very large. A storage server accessible through the EPFL network should be set up to facilitate the logging (at least temporarily) of these passes.

The gr-satellites library used for the FSK demodulator block offers other types of amateur satellite communication blocks. To adapt this acquisition script to other modulation schemes such as GFSK and GMSK, modulation and demodulation blocks are available. The inputs for the modulator blocks are the number of samples per symbol and the Gaussian filter bandwidth.

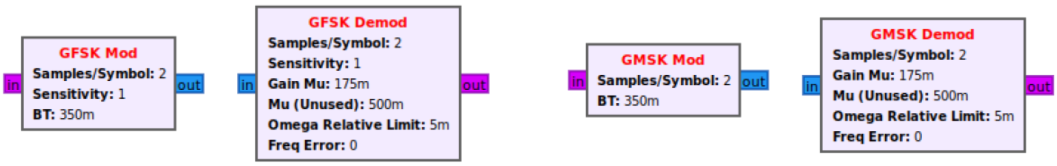


Figure 4.3: GFSK and GMSK modulation and demodulation blocks fetched from gr-satellites

In addition to the signal processing, this script provides a graphical interface to enable supervision of the signal acquisition. The QT library is used to generate the GUI. The following features are displayed:

Waterfall
A Waterfall displays the intensity of a signal in the frequency domain over time. This representation is useful to manually center the signal if a frequency offset is visible.

Frequency Sink
The frequency sink helps to better visualize the gain of the signal compared to the noise.

Waterfall Maximum Intensity Slider
This slider is used to adjust the maximum gain value represented on the waterfall to better distinguish the signal from the noise.

Frequency Shift Slider
This slider is used to manually correct possible frequency offsets.

Doppler Corrected Frequency
This block displays the frequency input received by the Gpredict instance running on the remote client PC. If no input is received, the box remains blank.

Acquisition of Signal Indicator
This indicator displays green if the current tracked satellite has a positive elevation from the ground station's point of view, indicating a current acquisition of signal. The false state is displayed in red.

4.2 Telemetry Downlink

To test the acquisition pipeline and simulate periodic telemetry downlink, the amateur satellite SelfieSat [7] was used as an example. This CubeSat, led by students of the Norwegian University of Science and Technology, displays similar telecommunication parameters

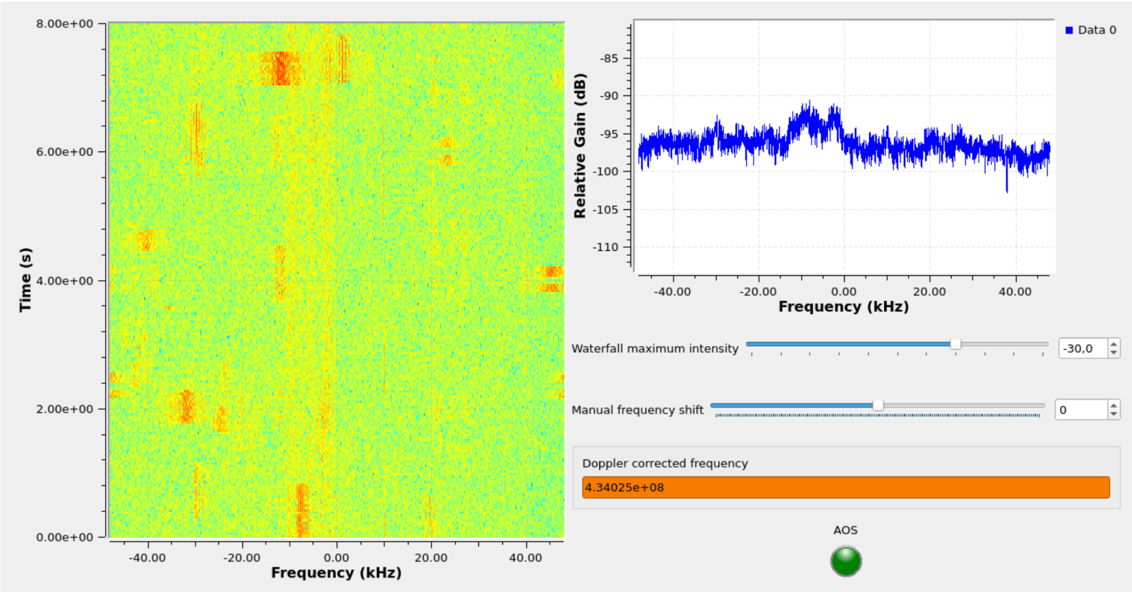


Figure 4.4: Graphical Interface generated by the FSK acquisition script

as Pathfinder I. It operates in the same UHF amateur frequencies by sending telemetry beacons at 437.5 MHz. The modulation scheme used is FSK with a baud rate of 9600, a very common choice in these amateur frequencies due to its simplicity of implementation. The 22-byte beacons follow the AX.25 protocol with G3RUH scrambling, as seen in Figure 4.5.

Bytes	1	2	3	4	5	6	7	8	9	10	11	12
Description	CSP HEADER				DATA LENGTH		ALARM MASK				EPS COUNTER	
Beacon example	05	00	85	01	00	10	00	00	00	00	03	d2

Bytes	13	14	15	16	17	18	19	20	21	22
Description	EPS VBATT		OUTMASK	ID	STATES	Unknown	CHECKSUM			
Beacon example	20	73	00	01	5A	57	07	91	68	4a

Figure 4.5: SelfieSat beacon structure

CSP corresponds to the CubeSat Space Protocol, a simple lightweight protocol common in amateur satellite missions.

The CSP-wrapped beacon has the following structure:

CSP Header

The header contains the following information: Priority of the packet, Source ID, Destination ID, Destination Port, Source Port, and other technical bits.

Data Length

These two bytes are reserved to announce the size of the transmitted packet. Consequently, the maximum packet size is 2^{16} or 65536 bytes of information.

Alarm Mask

The next reserved bytes correspond to the most important indicators of good onboard health. They are thus placed first in the payload data order. The different alarms correspond to:

- Temperature sensors

- Current and voltage sensor
- Angular velocity superior to the upper threshold
- Angular velocity inferior to the lower threshold
- Electronic Power System
- Temperature sensors of different subsystems

EPS Counter
EPS boot count.

EPS VBATT
Voltage of EPS battery (millivolts).

OUTMASK
Bitmask of state of EPS channels.

ID Identifier for the beacon. Increments once every beacon.

States The OBC main's internal FSM states (OBC, ADCS, payload).

Checksum
These last four bytes are reserved for verifying that the received data is identical to the transmitted data. This algorithm can detect single-bit errors and burst errors.

As a test campaign, SelfieSat beacons were downlinked and processed once per day for ten days. During this operation, the passes were logged, and the EPS VBATT and EPS COUNTER were stored and plotted in Figures 4.6 and 4.7.

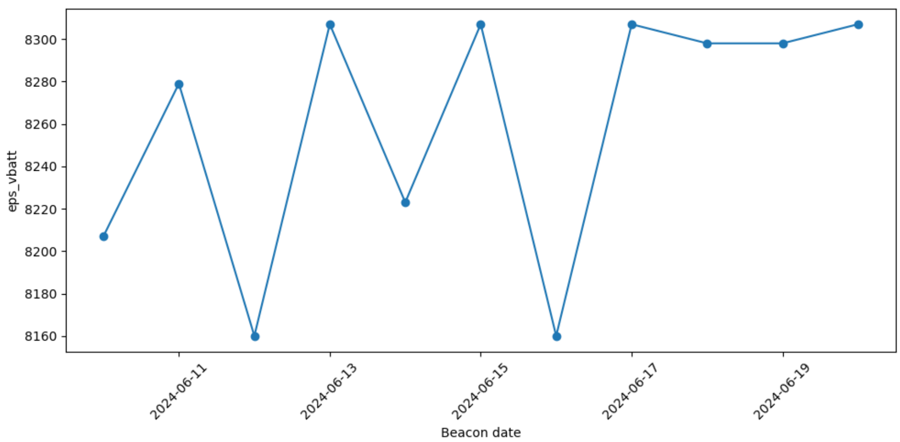


Figure 4.6: SelfieSat's battery voltage (in mV) over time

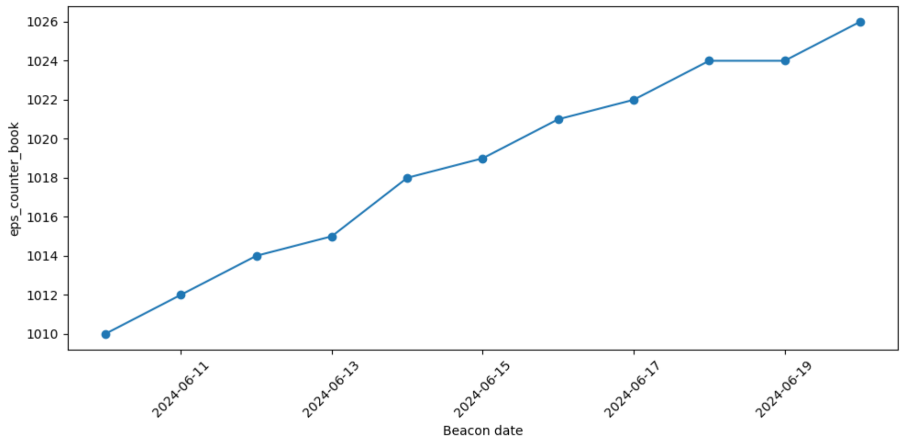


Figure 4.7: SelfieSat's EPS boot counter over time

5 Ground Station testing

Following the success of multiple satellite communication downlinks over time, the next step is characterising this acquisition pipeline.

5.1 SNR estimation

The following SNR estimations were made during a Selfiesat pass that reached a maximal elevation of 85° of elevation, an ideal situation to cover the total elevation range. The satellites trajectory was approximately oriented from North to South. Figure 5.1 display the carrier-to-noise ratio throughout this specific pass.

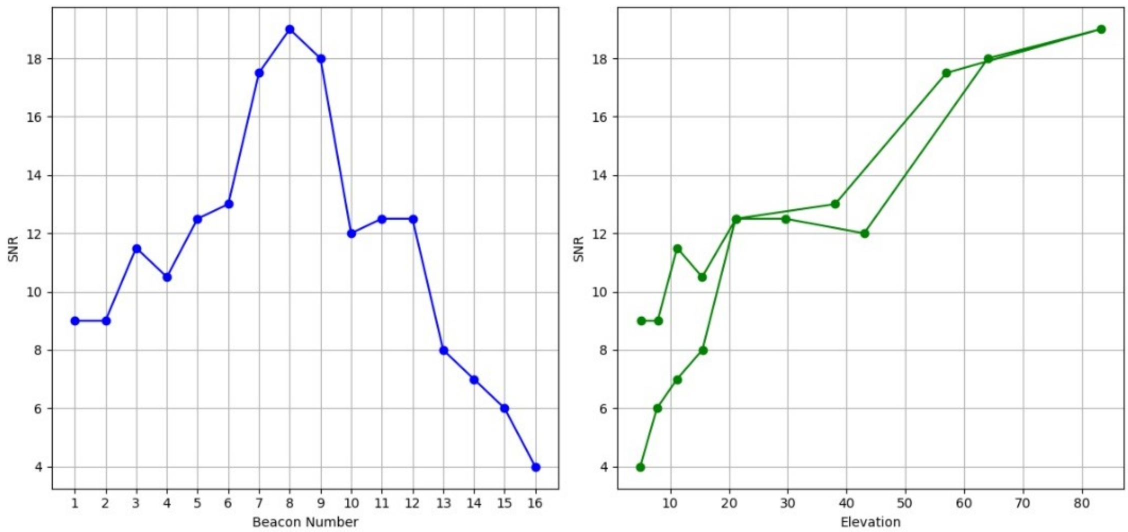


Figure 5.1: SNR Measurements of a SelfieSat pass

We can observe that the left graph is quite asymmetrical. This can be explained by the fact that depending on where the antenna is pointing, more or less noise is received due to sources of interferences. From a geographical point of view, we can clearly tell that the antenna receives more noise pointing north than south towards the Lac Léman. We also notice that beacons are correctly demodulated under the threshold of 15° used in the general mission planning. This threshold depends on geographical landmarks (mountains) and general noise. A precise 360° mask estimation of the field of view from the ELB roof would be useful from a mission planning point of view, determining more realistically which pass can be used to communicate.

5.2 Link Budget

The next step of the testing and characterisation of the UHF GS is estimating its System Noise Temperature Ts. For this, lets take the Link Budget calculation formula [15]:

$$\frac{E_b}{N_0} = EIRP + L_s + L_a + G_r - k - T_s - R$$

where:

$\frac{E_b}{N_0}$ is the energy per bit to noise power spectral density ratio (dB)

$EIRP$ is the Effective Isotropic Radiated Power (dB)

L_s is the space loss (dB)

L_a is the atmospheric loss (dB)

G_r is the gain of the receiving antenna (dB)

k is the Boltzmann constant (dB)

T_s is the system noise temperature (dB)

R is the data rate (dB)

SelfieSat's Transmission properties are the following :

Transmitter: Spacecraft	**Unit**	**Value**	**Source**
Power	[dBm]	33,0103	UHF_Radio_SAT2RF1-1B_NA-UHF-G0-R10.pdf
Power	[dBW]	3,0103	Calculated from previous
Antenna gain	[dBi]	-4,3311	EnduroSat_UHF_Antenna_III_module-R1.pdf
Cable loss	[dB]	0,3	Egil's estimate (HYPSO)
EIRP	[dBW]	-1,6208	Power+TX antenna gain

Figure 5.2: Power transmission of SelfieSat

With these values, we are now able to isolate Ts by taking measurements of Eb/N0 during SelfieSat passes. This next step will reinforce the results computed in the UHF Link Budget, which will help the general mission planning.

6 Conclusion

In this report, we have explored each step of information transmission through signals. We have developed an acquisition script that enables us to remotely downlink any FSK and GFSK signal passing over Lausanne, marking a significant milestone in the development of the Ground Station. It is now crucial to test and characterize the Ground Station to ensure we can accurately plan the mission and communication windows.

The UHF infrastructure will be temporarily relocated later this year due to renovations of the ELB roof. We can take this opportunity to enhance the mounting and overall stabilization of the antenna (addressing the backlash in the motors).

The development of the Ground Station, particularly the UHF antenna, is progressing ahead of the rest of the Spacecraft Team, as the launch is not scheduled until 2027 or 2028. By this time, it is essential to ensure the GS is reliable and fully functional, and explore potential applications in the interim. This could involve integrating it into SatNOGS for broader accessibility, utilizing it for academic purposes, and collaborating with other organizations through proposed operations or redundancy measures.

Acknowledgements

Firstly, I would like to personally thank Hannes Bartle for his amazing supervision throughout the semester, and more generally for his dedication in helping the Spacecraft Team.

I thank Dimitri Hollosi, Quentin Cabrol, Juliette Challot, and Bastien Labat for their work on the UHF Ground Station that I am honoured to continue.

I thank the Ground Segment Pole and especially Cyprien Requin for their help in deploying the UHF antenna this year.

Lastly, I thank Léonard Lebrun, Tom Schlatter, and Emeric Hartmann for their support throughout this amazing year in our ELB 216.

I really enjoyed realizing this project and hope the UHF Ground Station will continue thriving with new projects and applications.

References

[1] URL: https://github.com/pothosware/SoapySDR.

[2] URL: https://satnogs.org/.

[3] URL: https://celestrak.org/.

[4] URL: https://wiki.gnuradio.org/index.php/Quadrature_Demod.

[5] URL: https://en.wikipedia.org/wiki/High-Level_Data_Link_Control.

[6] URL: https://github.com/wnagele/gr-gpredict-doppler.

[7] URL: https://orbitntnu.com/selfiesat/.

[8] Juliette Challot. *Semester Project Report - Telecommunication : The UHF Module.* 2022.

[9] Endurosat. *UHF Antenna Type III - User Manual.*

[10] Endurosat. *UHF Transceiver Type II - User Manual.*

[11] Dani Estevez. *Gr-satellites.* URL: https://github.com/daniestevez/gr-satellites.

[12] Robert G. Gallager. *Principles of Digital Communication.* Cambridge University Press, 2008.

[13] Bastien Labat. *Semester Project Report - UHF Ground Station Analysis at System Level.* 2022.

[14] Adrien Dupont Léonard Lebrun. *CHESS Pathfinder 1 Mission - Mission Operation Plan.* 2024.

[15] Nada Eddahir Léonard Lebrun. *CHESS Pathfinder 1 Mission - Link Budget : UHF.* 2024.

[16] Dr. Marc Lichtman. *PySDR: A Guide to SDR and DSP using Python.* URL: https://pysdr.org/.

[17] J.G. Proakis. *Digital Communications.* Electrical engineering series. McGraw-Hill, 2001. ISBN: 9780072321111. URL: https://books.google.ch/books?id=sbr8QwAACAAJ.

[18] Lajos Hanzo Raymond Steele. *Mobile Radio Communications.* IEEE Xplore, 1999.