Alma Mater Ground Station Transceiver

a Software Defined Radio for Satellite Communications

Marco Bosco, Paolo Tortora
Department of Industrial Engineering
University of Bologna
Forli, Italy
marco.bosco3@unibo.it, paolo.tortora@unibo.it

Davide Cinarelli
ALMASpace S.r.l.
Forli, Italy
davide.cinarelli@almaspace.com

***Abstract*— Software Defined Radio (SDR) technology offers a great opportunity for enabling spacecraft (S/C) engineers to create reconfigurable, lightweight communications systems that can be easily prototyped and, if needed, upgraded to add improved functionalities and support new protocols. In the framework of ESA's European Student Earth Orbiter (ESEO) project, the University of Bologna is responsible for the mission control centre, and is currently developing a SDR ground station (G/S) section, to enhance the current capabilities of the existing Alma Mater Ground Station (AMGS). The enhanced G/S will be used in both uplink and downlink operations, communicating with the ESEO S/C using amateur-radio frequencies at UHF-band. It is also capable of receiving data from several other small satellites transmitting at UHF-band and orbiting in low Earth orbit (LEO), offering a great hands-on experience to students.**

Keywords—Software Defined Radio; Ground Station; ESEO; GNURadio;

I. INTRODUCTION

The term 'Software Radio' was coined by Joseph Mitola III in [1] to mark the shift from hardware (HW) design dominated radio systems to systems where the largest part of the functionality is defined in software (SW). The Federal Communications Commission (FCC) has first approved the use of SDR in the United States in November 2004 [2]. The FCC defines SDR as a "radio that includes a transmitter in which the operating parameters of frequency range, modulation type or maximum output power (either radiated or conducted) can be altered by making a change in software without making any changes to hardware components that affect the radio frequency emissions" [3]. The Software Defined Radio (SDR) Forum and the IEEE have defined SDR as a "radio in which some or all of the physical layer functions are software defined" [4]. Traditionally, these physical layer functions are implemented as HW components (e.g., filters, amplifiers, modulators/demodulators, detectors, etc.). The HW redesign is expensive and time consuming, while SW code redesign requires less effort in terms of money and time. One of the most important advantages of SDR over the traditional radio communication systems is the flexibility. SDR offers the ability to transmit and receive different radio protocols or waveforms changing SW and without modifying the SDR platform (the combination of HW and operating environment where the application is running). Furthermore, data processing may be performed with any general-purpose computer rather than using specialized HW. There are several challenges and opportunities related to the use of SDR in different military and civilian fields, all well summarized in [5]. In the field of G/S technologies, this gives the possibility to easily modify the design to communicate with satellites using different modulation schemes and protocols. A G/S based on SDR HW is extremely flexible and is, for example, more suitable for worldwide distributed systems, as ESA's Global Educational Network for Satellite Operations (GENSO) system, where updates containing the SW for communicating using new waveforms could be automatically shared among different distant stations without the need for HW upgrades [6]. Moreover, SDR is very beneficial for education purposes: engineering students can simply develop SDR algorithms on a computer applying their knowledge in communication theory to practical applications minimizing the budget for supplementary expensive HW (e.g., oscilloscope, spectrum analyser, signal generator, etc.) as shown in [7]. In this paper, the use of a SDR as the transceiver of the Alma Mater Ground Station (AMGS) within ESA's European Student Earth Orbiter (ESEO) project is described in detail, providing information about the model of the SDR chosen to best fit our needs and the open source software to implement the digital signal processing (DSP). Afterwards, the ESEO project is introduced, focusing on the telecommunication system, and then the design of the AMGS is described. Finally, we draw some conclusions on the work carried out so far and the achieved results.

II. THE SOFTWARE DEFINED RADIO PLATFORM

Among the several available models of SDR, we selected the universal software radio peripheral (USRP) N210 (Fig. 1) [8], [9] equipped with an SBX daughterboard by Ettus Research Ltd.

The USRP is designed to allow general-purpose computers to function as high bandwidth SDR. In essence, it serves as a digital baseband and IF section of a radio communication system. All high-speed general-purpose operations like digital up and down conversion (DUC and DDC), decimation and interpolation are done on the field programmable gate array (FPGA). The powerful combination of flexible hardware, open-source software and a community of experienced users make it

978-1-4799-2069-3/14/$31.00 ©2014 IEEE

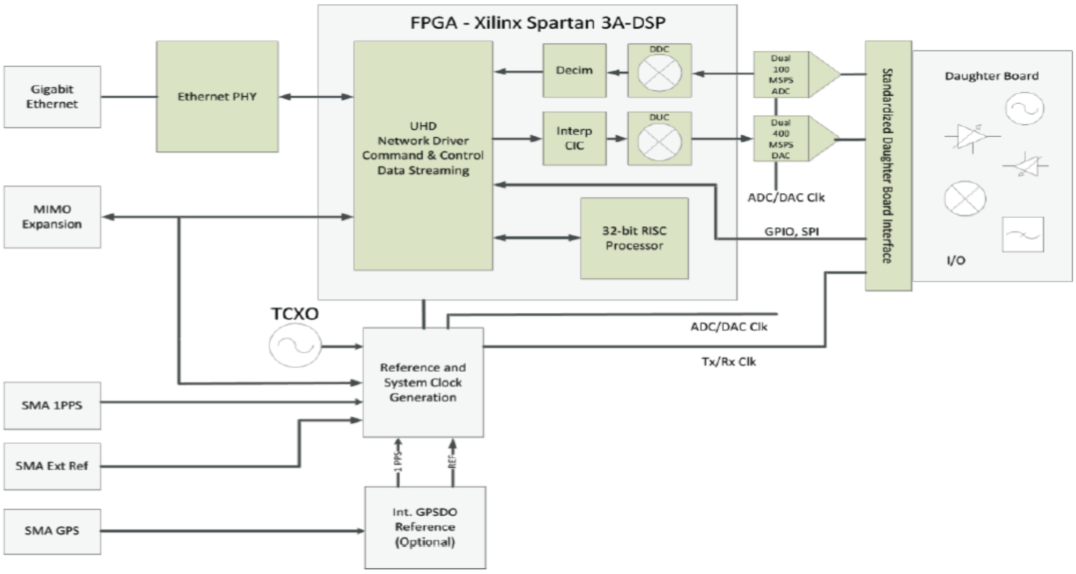

Fig. 1. USRP N210 architecture.

the ideal platform for a SDR development. The USRP N210 model provides high-bandwidth and high-dynamic range processing capability. It belongs to networked series (N-series), thus, data pass through the SDR and the PC using Ethernet frames.

The radio frequency (RF) front-end is performed by a RF daughterboard. For the AMGS, we selected an SBX daughterboard to process the signal at the original incoming RF. It is a homodyne and wide bandwidth transceiver with a 5 dB noise figure providing 40 MHz of bandwidth in the frequency range from 400 MHz to 4400 MHz (UHF and S-band) and working in full duplex mode on different transmit and receive frequencies, thanks to its independent receive and transmit local oscillators and synthesizers.

The Ettus Research has developed a free, open-source and multi-platform software compatible with all USRP devices named USRP Hardware Driver (UHD). It can be used standalone or with different third-party applications, for developing SDR platforms, such as LabVIEW, MATLAB/Simulink and GNU Radio (GR). The first two SWs are commercial while the latter is a free, open-source and multi-platform SW, even if it runs better in Linux since it is Linux native. The use of MATLAB/Simulink with SDR is described with several examples in [10] while an example of using SDR in LabVIEW is provided in [8]. GR uses a two-tier structure: the computationally intensive processing functions are implemented in C++ while application-defined control and coordination of blocks are developed in Python. A description of GR working mechanism can be found in [11]. GR comes with a graphical programming interface implemented in XML language named GR Companion (GRC) that allows generating flowgraphs by dragging and dropping the blocks from a list without writing the Python code. There are mainly four kinds of blocks: source, sink, operation and visualization. The benefits of using GR for satellite communications are shown in [12], the implementation in GR of an SDR transceiver compliant with the Consultative Committee for Space Data Systems (CCSDS) standards is reported in [13] and the design of an S-band G/S SDR transceiver using GR is described in [14]. A drawback of using GR is that documentation is spread through the GR website [15], forums, presentations, papers and thesis works, making the learning curve quite steep but the users' community is very active and increasing year by year.

III. THE EUROPEAN STUDENT EARTH ORBITER PROJECT

ESEO is a micro-satellite mission to low Earth orbit (LEO). It is being developed, integrated, and tested with the significant contribution of European university students as an ESA Education Office project. It is aimed at providing students with unparalleled hands-on experience to help prepare a well-qualified space-engineering workforce for Europe's future. In particular, ESEO satellite has the following mission objectives: taking pictures of the Earth and other celestial bodies from Earth orbit for educational outreach purposes, providing dosimetry and space plasma measurement in Earth orbit and its effects on satellite components and testing technologies for future education satellite missions [16]. The telecommunication system supports the functions of telemetry and telecommand (TMTC) for each phase of the mission. The spacecraft (S/C) telecommunication system consists of a redundant set of transceivers working at UHF-band for the uplink and the downlink. In addition, two payloads are included for telecommunication purposes, using L-band for uplink, and VHF and S-band for downlink: while the VHF/L-band system being developed by AMSAT-UK is used for educational purposes as voice transponder and low data-rate telemetry (TM) downlink, the on-board S-band high-speed transmitter payload, being developed in Poland by a Wroclaw University of Technology team, will be used to download scientific payload data. The simultaneous downlink or uplink at different bands is currently not considered (but not excluded) for the mission. Similarly, full duplex communication is not foreseen at UHF band. Half-duplex communication is identified as the standard TMTC communication. The ESEO G/S network consists of three stations:

- The main G/S, located in Forlì (Italy), is used in order to transmit telecommands (TC) and receive the whole amount of TM data at UHF-band.
- The one located at the University of Vigo (Spain) is used as backup for TMTC communication at UHF-band.
- The one located at the Technical University of Munich (Germany) is dedicated to download the whole amount of data generated by the different payloads at S-band.

The platform communication subsystem is allocated for a space amateur-radio link with a downlink frequency of 437.000 MHz and an uplink frequency of 435.200 MHz. The selection of the modulation scheme at UHF-band is compliant with the space amateur regulation: the assumed modulation scheme is Pulse Code Modulation (PCM), Non-Return-to-Zero-Level (NRZ-L), Gaussian Frequency Shift Keying (GFSK) with a BT factor of the Gaussian filter equal to 0.5. No further coding is used by default, as the link budget allows for the establishment of a reliable link also in the worst visibility case and for the higher bit rate. The default data rate will be 9.6 kb/s, and can be changed to 4.8 kb/s, by a proper TC. The protocol will always follow the space amateur AX.25 communications standard.

IV. ALMA MATER GROUND STATION DESIGN

The hardware of the AMGS is in part inherited from the previous G/S developed and tested for ALMASat-1 mission (TABLE I), with the introduction of significant updates.

TABLE I
AMGS OLD HARDWARE.

Component	Model
Antenna	UHF 2x19 elements Yagi
Low Noise Amplifier	Landwehr N GaAs 435 MAS
Radio	Icom IC-910H
Radio Level Converter	Icom CT-17
TNC	Kantronics KPC-9612+
Antenna Rotator	AlfaSpid RAS (AZ/EL)
Antenna Controller	Rot2Prog
DC Power Supply	Diamond Antenna GSV3000

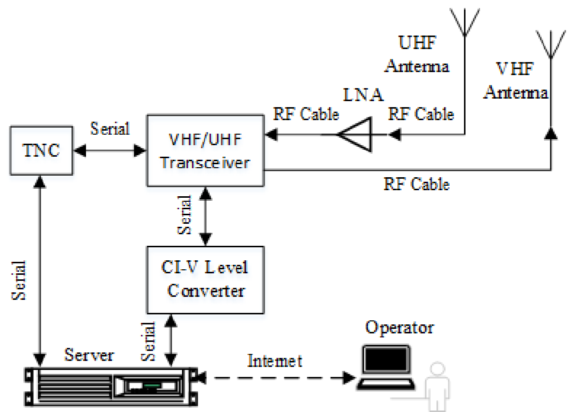

Fig. 2. AMGS old RF section scheme.

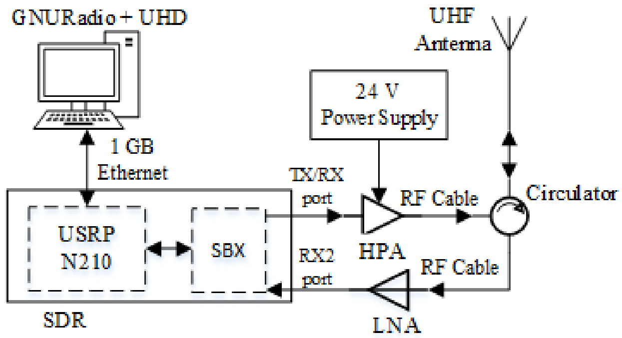

Fig. 3. AMGS new RF section scheme.

We have completely changed the RF section with respect to the previous HW. A complete SDR platform based on an USRP N210 with an SBX daughterboard (Fig. 3) has replaced the more classical architecture based on a terminal node controller (TNC) and an analog transceiver (Fig. 2). Using this new architecture, a high power amplifier (HPA) needs to be added between the USRP and the antenna since the maximum output power of SBX is 20 dBm (100 mW) with a 32 dB power control range. We selected the ZHL-50W-52 model by Mini-Circuits, which has a recommended maximum input power of 0 dBm and an output power of 50 W. This HPA is powered by a dedicated 24V power supply. Moreover, since the communication between ESEO and the AMGS will be in half-duplex mode, we will place a circulator between the UHF antenna, the TX and RX line. The SW development tool GR version 3.6.5 runs on Linux Ubuntu 12.04 Long Term Support (LTS) installed on a Dell Optiplex 360.

We designed a first version of the G/S transceiver SW in GRC (Fig. 4). The *UHD:USRP Sink* and *UHD:USRP Source* blocks are used to connect to the USRP device in transmission (TX) and reception (RX). The most important parameters of these blocks are the TX and RX frequencies, the address of the USRP, the antenna (*TX/RX* for TX and *RX2* for RX), and the gain in dB of the USRP which can also be changed at runtime. The *Frequency Xlating FIR Filter* block is used to translate the TX and RX frequency due to the S/C Doppler shift. This is computed by Orekit, a free and open-source low-level space-dynamics Java library widely adopted by space agencies [17]. The frequency Doppler shift is then passed to GR using an XML-Remote Procedure Call (RPC). The modulation/demodulation is implemented using the *GFSK Mod* and *GFSK Demod* blocks. The *GFSK Mod* block performs the NRZ line coding converting a stream of bits into ±1, the Gaussian filtering for bandwidth reduction and filter shaping using the specified value of BT, and the frequency modulation itself. The *sensitivity*, *s* parameter in the *GFSK Mod* block gives the phase change, ΔΦ per sample and is calculated by (1):

$$s = \Delta\Phi \,/\, sps = 2\pi \cdot \Delta f \cdot \Delta t = \pi \cdot k \,/\, sps \qquad (1)$$

Where *sps* is the number of samples per symbol and *k* is the modulation index given by (2):

$$k = 2\Delta f \,/\, R_s \qquad (2)$$

Where Δf is the frequency deviation and R_s is the symbol rate. Note that for Gaussian Minimum Shift Keying (GMSK) modulation, k assumes the minimum value of 0.5 and then the phase change between two consecutive symbols is equal to $\pi/2$. The *GFSK Demod* block takes as input the complex modulated signal at baseband and outputs a stream of bits packed one bit per byte. It performs the quadrature demodulation, the clock recovery using the Mueller and Muller (M&M) discrete error synchronizer [18] and hard decision decoding. Clock recovery technique requires to set some specific parameters as: the initial value, μ (chosen equal to 0.5) of the interpolator related to the adjustment of the sampling due to error signal computed by the M&M algorithm; the gain parameter, G_μ to adjust μ based on the timing difference between symbols (chosen equal to 0.175); the initial value, ω for the number of symbols between samples (chosen equal to *sps*); the gain parameter G_ω to adjust ω based on the error computed by the M&M algorithm, which is related to G_μ as in (3):

$$G_\omega = G_\mu^2 / 4 \tag{3}$$

Another parameter to be set is the omega relative limit, ω_{rel} that fixes the maximum and minimum value for ω and is equal to 0.005. As stated above, protocol encoder and decoder operations are completely flexible in SDR environment. GR provides some encoder/decoder blocks like the CCSDS basic convolutional code [20], the parallel concatenated convolutional code (PCCC) [21] and the serial concatenated convolutional code (SCCC) [22]. We added an out-of-tree module in GR to implement the AX.25 decoder blocks, which also gives the possibility to save the messages in a file. A very useful tutorial on how to create an out-of-tree module in GR can be found in [15]. We implemented the unnumbered information (UI) frame assembler/disassembler which are the parts of the AX.25 protocol used by the amateur satellite communications and some of the functions of the high-level datalink control (HDLC): NRZ to NRZI encoder, NRZI to NRZ decoder, bit stuffing/destuffing, frame delimiter and scrambler/descrambler. A description of all these functions can be found in [19]. We implemented all these functions in C++ language and tested them separately before creating the GR blocks. Another way, which is less complex and less efficient at the same time is implementing all these functions as Python scripts and import them as modules in the main Python script. The GR transceiver flowgraph is connected to the ESEO mission control system (MCS) through a transmission control protocol (TCP) connection using the *TCP Source/Sink* block to receive the TC data and send the TM data from/to the MCS. We also added some fast Fourier transform (FFT) plot blocks in the GR flowgraph to display the TX and RX signal in the frequency domain.

V. TEST CAMPAIGN

Before performing any tests, calibrations for the RF board are needed to minimize the TX/RX IQ imbalance and TX DC offset versus the LO frequency, improving the performance of our SDR. The UHD software comes with some calibration utilities for this scope. To improve our understanding on how the SDR works we performed functional testing of our USRP radios in both TX and RX. GR comes with two helpful tools for testing TX and RX named *uhd_fft* and *uhd_siggen_gui*, respectively. The block diagram for the receiving and transmitting test is illustrated in Fig. 5.

An MXG ATE analog signal generator (SG) N5161A by Agilent Technologies, capable of transmitting in the frequency band 100 kHz - 3 GHz, is connected to the TX/RX port of the

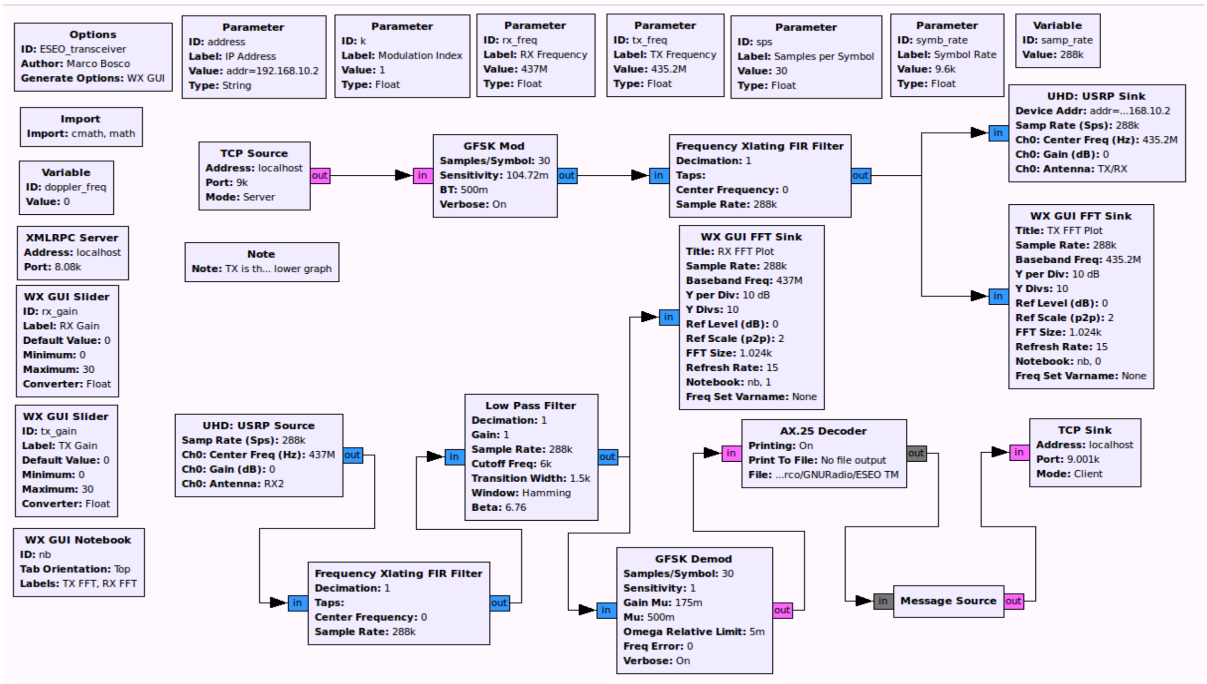

Fig. 4. AMGS transceiver flowgraph in GNU Radio companion (GRC).

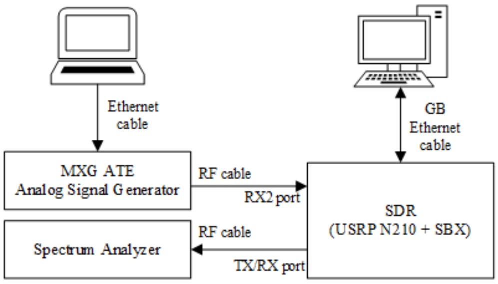

Fig. 5. SDR characterization test setup.

SDR through a RF cable. The SG is commanded by a laptop through a LAN extension for instrumentation (LXI) connection. Care must be exercised to guarantee that the maximum input specification of the SBX daughterboard is not exceeded. This level is -10 dBm but we suggest taking a large margin and fixing the limit to -20 dBm. The SDR is also connected at the RX2 port through an RF cable to a spectrum analyser (SA) N9320A by Agilent Technologies, capable of receiving in the frequency band 9 kHz – 3.0 GHz. The maximum input power for the SA is 30 dBm; however, this is not a limitation for this test since the maximum output power from the SDR is 20dBm. The SDR is then connected through a 1 GB Ethernet cable to a PC running GR on Linux Ubuntu. We first set the center frequency of the SDR at 437 MHz when receiving a signal generated at 437 MHz. In the FFT plot, three peaks are displayed. At 437 MHz there is the main peak as expected. However, we also have a peak at 436.996 MHz, which is due to the local oscillator (LO) frequency leakage and another one at 436.992 MHz which is the LO frequency minus the difference between the center frequency and the LO frequency named DSP frequency by GR. In fact, the PLL step size of the SBX daughterboard is 4 kHz, this means that the front-end will tune to the desired frequency as close as possible. The DDC will then compensate for the DSP frequency. When transmitting at other frequencies we also noticed peaks in the spectrum at multiples of the master clock rate, the sample rate of the ADC, equal to 100 MHz.
The receiving test consisted in transmitting a GFSK modulated signal using the SDR and displaying it in the SA as shown in Fig. 6. We checked the maximum output power to be between 17 and 20 dBm.

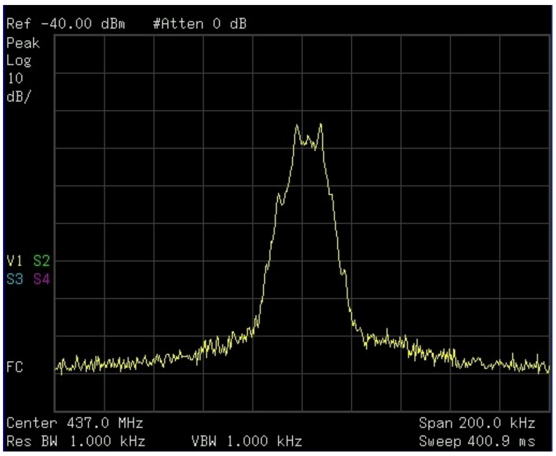

Fig. 6. GFSK modulated signal spectrum.

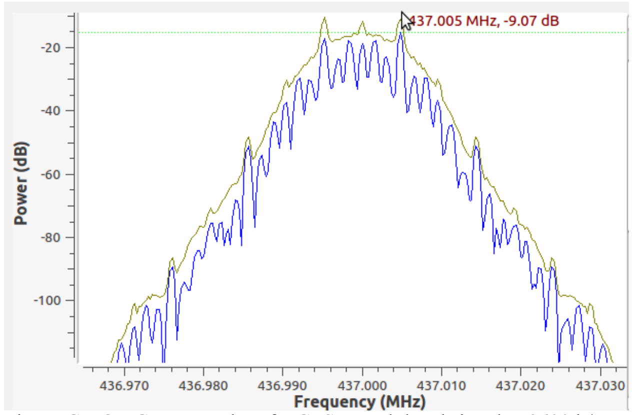

Fig. 7. GR QT GUI FFT plot of a GFSK modulated signal at 9600 b/s

The signal is also displayed using the GR *QT GUI Sink* block, which has four plots: FFT, waterfall, time domain and constellation. Fig. 7 depicts the FFT plot of the windowed signal with the max hold function enabled. The data cursor indicates that the two main peaks are at 437 MHz ± 4.8 kHz as expected.
Fig. 8 depicts the waterfall plot of the signal, the x-axis represents the frequency deviation in kHz from the center frequency of 437 MHz. Again, the data cursor indicates the two main peaks of the GFSK modulated signal.

Once the SDR was characterised, we tested a communication link between two identical SDRs, trying to reconstruct the link between ESEO and the G/S in terms of power, frequency, data-rate and modulation scheme referring to the ESEO link budget (TABLE II).
A directional bridge is used at the RX side to split the signal: the less attenuated path (1.5 dB loss) is connected to the SA to measure the signal level at the receiver because the GR FFT plots display the signal using a relative dB scale. The other path (16 dB loss) is connected to the SDR used as RX. The test setup is shown in Fig. 9.
We transmitted a stream of alternated bits (01010101), modulated as a GFSK signal at a data-rate of 9.6 kb/s. We regulated the transmitted signal power so that the isotropic

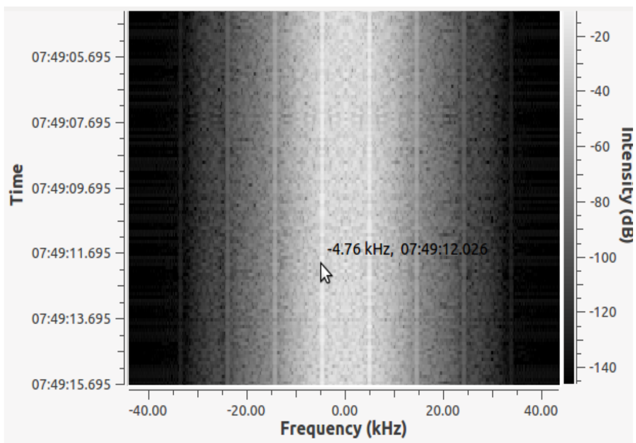

Fig. 8. GR QT GUI waterfall plot of a GFSK modulated signal at 9600 b/s.

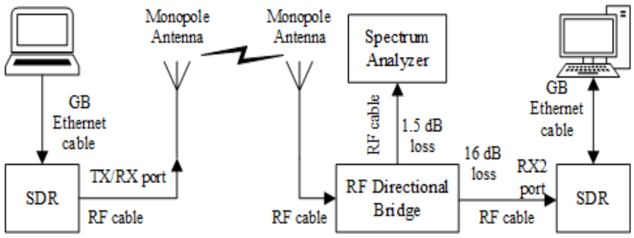

Fig. 9. SDRs TX/RX communication link setup.

signal level at G/S is the same as reported in the ESEO link budget.
The monopole antenna gain is reasonably assumed to be 0 dBi. The bits output by the demodulator block are stored into a binary data file using a proper block named *File Sink* for off-line analysis in MATLAB or GNU Octave. GR comes with a set of *.m* files that perform the conversion between binary files and readable data like *read_float_binary*. We could then compute the bit error rate (BER) and check that it is below the required value of 10^{-5}.

VI. Conclusion

We have introduced the SDR platform and described the re-design of the AMGS using the SDR as transceiver in the framework of ESEO project.

Preliminary tests have been completed on the SDR platform designed for the AMGS. We successfully transmitted data from one SDR platform to the other, considering the link budget between ESEO and the AMGS. In the next future, we will also test the SDR transceiver with the ESEO on-board transceiver to fully validate our G/S, in view of the approaching ESEO project critical design review (CDR).

TABLE II
Extract of ESEO UHF Link Budget

Parameter	Best case	Average case	Worst case	Units
Isotropic Signal Level at G/S	-118.11	-120.04	-124.63	dBm
G/S Figure of Merit (G/T)	-11.28	-12.63	-15.56	dB/K
Signal-to-Noise Power Density	69.19	65.87	58.31	dBHz
System Desired Data Rate	39.82	39.82	39.82	dBHz
TM System E_b/N_0	29.37	26.05	18.49	dB
E_b/N_0 Threshold	14.80	14.80	14.80	dB
System Link Margin	14.57	11.25	3.69	dB

References

[1] J. Mitola III, “Software radios - survey, critical evaluation and future directions”, in Telesyst. Conf., 1992. NTC-92., National, May 1992, pp. 13/15–13/23.

[2] Federal Communications Commission, “FCC approves first software defined radio”, November 2004.

[3] Federal Communications Commission, “In the matter of authorization and use of software defined radio”, September 2001, FCC 01-264, ET Docket No. 00-47.

[4] “SDRF cognitive radio definitions working document, SDRF-06-R-0011-V1.0.0”. [Online]. Available: http://groups.winnforum.org/d/do/1585.

[5] T. Ulversøy, “Software defined radio: challenges and opportunities”, IEEE Communications Surveys & Tutorials, Vol. 12, No. 4, Fourth Quarter 2010, pp. 531-550. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[6] R.L. Buffington, R.S. Erwin, J.F. Androlewicz, and J. Lyke, “GENSO, SPA, SDR, and GNU Radio: The pathway ahead for space dial tone”, Infotech@Aerospace Conference, Unleashing Unmanned Systems St. Louis, Missouri, 29-31 March 2011.

[7] A.L.G. Reis, A.F.B. Selva, K.G. Lenzi, S.E. Barbin, and L.G.P. Meloni, “Software Defined Radio on Digital Communications: a New Teaching Tool”, in Wireless and Microwave Technology Conference (WAMICON), 2012 IEEE 13th Annual, April 2012, pp. 1-8.

[8] S. Sherman, J. Kimery, “Software defined radio prototyping platforms enable a flexible approach to design”, IEEE Microwave Magazine, July 2012, pp. 76-80.

[9] “USRP N200/210 Datasheet”. [Online]. Available: https://www.ettus.com/content/files/07495_Ettus_N200-210_DS_Flyer_HR_1.pdf

[10] D. Pu, and A.M. Wyglinski, “Digital communication systems engineering with software-defined radio”, Artech House, 2013.

[11] F. Ge, C.J. Chiang, Y.M. Gottlieb, and R. Chadha, “GNU Radio-based digital communications: computational analysis of a GMSK transceiver” in Global Telecommunications Conference (GLOBECOM 2011), December 2011, pp. 1-6.

[12] J. Otomo, “GNU Radio for Satellite Communications”, 2012 GNU Radio Conference, September 2012.

[13] O. Sarwar, “Software Defined Radio (SDR) for deep space communication”, Master’s Thesis, September 2013, Luleå University of Technology, Rymdcampus, Kiruna, Sweden.

[14] A.M. Torío, “Software Defined S-Band Ground Station Transceiver for Satellite Communications”, Diploma Thesis, August 2011, Institute of Telecommunications, Department of Electrical Engineering, Vienna University of Technology.

[15] “GNU Radio website”. [Online]. Available: http://gnuradio.org/redmine/projects/gnuradio/wiki

[16] “ESEO Mission”. [Online]. Available: http://www.esa.int/Education/ESEO_mission

[17] “Orekit website”. [Online]. Available: http://www.orekit.org/

[18] H. Meyr. M. Moeneclaey, and S.A. Fechtel, “Digital communication receivers: synchronization, channel estimation and signal processing”, 1998 John Wiley & Sons, Inc., pp. 86-88.

[19] L.D. Halliday, “Communications Infrastructure for the MOST Microsatellite Project”, Master Thesis, 2000, Graduate Department of Aerospace Science and Engineering, University of Toronto.

[20] TM Synchronization and Channel Coding, Recommendation for Space Data System Standards, 131.0-B-2. Blue Book, Issue 2, Consultative Committee for Space Data Systems (CCSDS), August. 2011.

[21] S. Benedetto, “Design of parallel concatenated convolutional codes”, IEEE Transactions on Communications, Vol. 44, No. 55, May 1996, pp. 591-600.

[22] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, “Serial concatenation of interleaved codes: performance analysis, design, and iterative decoding”, IEEE Transactions on Information Theory, Vol. 44, Issue 3, May 1998, pp. 909-926.