International Conference on Engineering Education and Research "Progress Through Partnership"
© 2004 VSB-TUO, Ostrava, ISSN 1562-3580

# Telemeasured Performances of a DSP based CDMA Software Defined Radio

Oreste ANDRISANO
IEIIT-BO/CNR, CNIT, DEIS at the University of Bologna, 2 Risorgimento street, 40136 Bologna, ITALY, oandrisano@deis.unibo.it, http://www-csite.deis.unibo.it/Staff/andrisano.html.

Andrea CONTI
IEIIT-BO/CNR, CNIT, DEIS at the University of Bologna, 2 Risorgimento street, 40136 Bologna, ITALY, aconti@deis.unibo.it.

Davide DARDARI
IEIIT-BO/CNR, CNIT, DEIS at the University of Bologna, 2 Risorgimento street, 40136 Bologna, ITALY, ddardari@deis.unibo.it.

Alberto ROVERSI
CNIT at the University of Bologna, 2 Risorgimento street, 40136 Bologna, ITALY, alberto.roversi@cnit.it.

KEYWORDS: *telemeasurement, remote control, software radio, code division multiple access, on-line laboratory*

ABSTRACT: *This paper addresses the characterization of systems and devices via telemeasurement through heterogeneous networks.*

*The methodology is generally valid and here applied to communication systems. Both dedicated circuits and DSP-based realizations are taken into account. In the last case the remote definition of the system is also considered. The software radio technology offers these advantages by means of digital signal processing techniques for which many functionalities are implemented via software with reduced cost with respect to hardware realizations. The example proposed in this paper uses a solution of remote downloading of small software files plus commands for transparent parameter setting. Possible system architecture for telemeasurement, developed in our laboratories, is proposed and described. It is useful for two typical situations: the case in which few valuable instruments and devices are available only in some particular laboratories, and the educational case where the number of instruments available is not sufficient with respect to the potential number of users. An integrated WEB server to manage both instruments and DSP platforms has been also developed. The measurement benchmark has been completely remote using a signal matrix; hence the presence of a human operator is no more needed. A complete remote measurement test-bed experience on a DSP-based CDMA satellite modem for satellite telecommunication systems is reported. Firstly a brief description of the hardware platform, the transmission chain and the code acquisition, tracking, offset recovery algorithms are presented. Secondly, the software architecture that performs the web-client access to control the instruments on the measurement benchmark and the reconfiguration by remote of the DSP-based modem are described. Finally, some numerical results in term of telemeasured Bit Error Rate with AWGN channel and telemeasured performances of Frequency Offset recovery algorithm in term of stability and convergence speed are given.*

## 1 INTRODUCTION

The software radio concept [1], [2] allows a large degree of flexibility of a telecommunication system with respect to traffic, QoS, and resource allocation efficiency requirements; it enables a multi-mode radio implementation to make available different standards on the same hardware.

This is especially useful in integrated terrestrial-satellite networks, where several standards have to coexist on the same devices: software reconfiguration should lead to the future implementation of multi-standard terminals, able at working worldwide. The software radio technology offers these advantages by means of digital signal processing techniques for which many functionalities are implemented via software with costs less than hardware realization.

The software reconfiguration should be done in two main ways: the reconfiguration by remote firmware downloading and the reconfiguration by parameter setting. The reconfiguration by remote firmware downloading has the advantage that can be performed in fully transparent way for the user by download software into hardware platform: it is useful to have this possibility for example when the transmitter and the receiver sides are divided by very long distances like in a radio digital communication bridge or a in digital satellite communication. The reconfiguration by parameter setting considers pre-existent loaded software with the possibility to modify system parameters by the user. .

In this paper a DSP-based remote configurable satellite modem is described. It uses a hybrid solution: remote downloading of small software files plus commands for transparent parameter setting. The modem architecture under consideration is designed in order to support an asynchronous Direct Sequence Spread Spectrum (DS/SS) CDMA with DQPSK digital modulation. It is applicable in those cases where the satellite acts as a transparent transponder and the same bandwidth resource is shared by several earth terminals using the CDMA technique. In those cases, all the processing is performed at the earth station.

In a DSP-based architecture a great effort has to be carried out to find efficient algorithms in term of computational complexity, due to digital signal processing limitations in computational performance. For this reasons canonical synchronization algorithms based on the use of matched filter are not suitable for DSP realization so the receiver algorithms for code acquisition is based on the sequential search of the synchronization symbols that are transmitted every $N_{sync}$ data symbols. The same methodology is used in tracking algorithm to recovery time deviation due to different sample rates between transmitter and receiver sides. Besides a carrier frequency offset recovery scheme implemented by a software algorithm in the receiver DSP, permits to track frequency deviation in the range $[-B_S/2, B_S/2]$, where $B_S$ is the symbol-rate.

A brief description of the hardware and software realization is reported; the possibility to change the transmission parameters loading and running via internet on the DSP platform, different executable programs is described. The possibility, that has a remote user, to obtain the measured values by either the DSP receiver (like Bit Error Rate, frequency recovered offset between the modulation and demodulation carriers) or by the measurement instruments involved in the transmission chain is shown. Finally the telemeasured performances are reported.

## 2 REMOTE UPLOADING

We give a short description of the implemented software architecture that realizes the remote experience. The software radio remote transmission system, illustrated in Fig. 1, it is composed by six elements: two target programs running on DSP boards, two server tasks running on dedicated personal computers (TXhost, RXhost), a WEB server (Apache [3]) to service the remote web clients requests and a Lab-view [4] WEB server that offers the Virtual Instrument Panels to control the measure instrumentation involved in the transmission chain.

As described at the beginning of this section the remote WEB user request are served from the WEB server by the use of more CGIs (Common Gateway Interface). The WEB site is powered by an Apache HTTP server, in the Apache WEB server are kept different CGIs with different functions: the program, called CGImodem, creates a TCP socket identified by the IP and the port addresses of the RX_Host and sends to it the transmission parameters selected by remote user:

sample rate, spreading factor, name of the data file to transmit and an identifier of the type of required measure, which identifies the target program to load into DSP memory.

The RXhost receives the parameters, sets the transmission parameters according to the request of client and loads them and the target program into the memory of the DSP; finally the DSP receiver algorithm is started. At this point RXhost sends a confirmation to the Apache server. When the CGImodem receives the confirmation that the DSP receiver program is running, communicates to TXhost program the transmission parameters using the same procedure; when the CGImodem receives confirmation that the DSP transmitter program is running, returns a HTML page containing the description of the real used transmission parameters, and a submit button: when the client user presses this button, the Apache server calls a second script called CGIber, which establishes a new connection with RXhost to obtain, at the end of the measure process, the bit error rate (BER) value and eventually the performances of the algorithm that recovers the frequency offset between carriers.

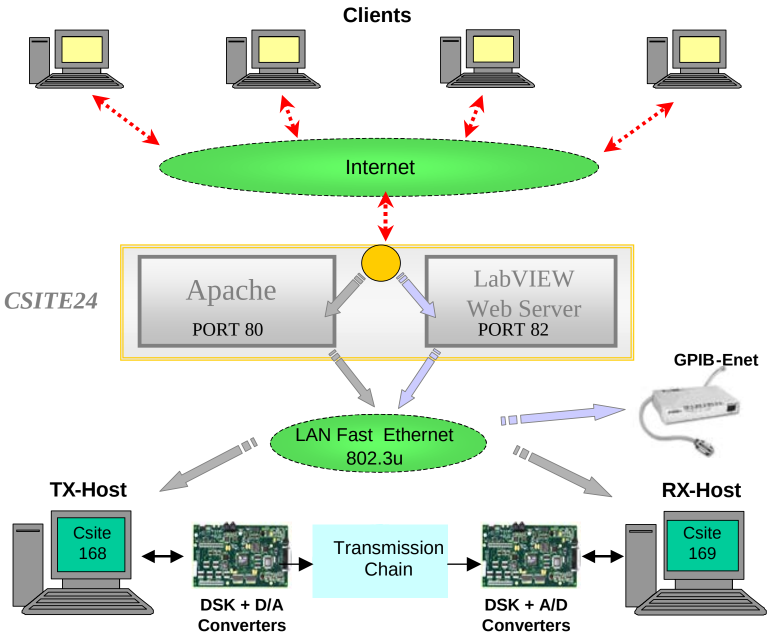


Figure 1 Remote Programmable Software Radio System

## 3 DESCRIPTION OF IMPLEMENTED ALGORITHMS

We give a short description of the chosen data frame format, the transmission scheme, the acquisition and tracking algorithms and the recovery frequency offset algorithm that implement the CDMA system. The chosen data frame format is composed by one synchronization symbol every $N_{SYNC}$ data symbols; the synchronization symbol is characterized by a different spreading code and a power level $h$ times greater than the power level dedicated to data symbols, as described in [5] where $T_S = 1/B_S$ is the symbol time. A detailed description of the detection probability and false alarm probability dependence from the ratio $h$ is reported in [6]. Different spreading codes for the synchronization and data symbols to limit false synchronization events during the transmission have been chosen. The transmission of data is preceded by the transmission of a preamble enabling the tracking process before the start of the transmission of data symbols. The modulation scheme realized is a DS/SS - DQPSK. The transmission algorithm consists of four basic parts: parameters setting, spread spectrum coding, DQPSK modulation and data transmission. The definition of the transmitter parameters like spreading factor $N$, the peak-to-peak voltage amplitude of the synchronism symbol $Vpp_{synch}$, the synchronism-to-data symbol power ratio $h$, the over-sampling factor $F_S$ and the sample rate $f_S$ is given by the host PC program. The spreading and over-sampling of the four complex symbols is initially performed and the results are stored in four buffers located into the internal memory of the transmitter DSP; in this way to transmit the symbols sequence we have only to match the complex symbols derived from the DQPSK modulation with the associated memory buffers as shown in Fig. 2. Eight buffers of coded symbols are produced and loaded into DSP internal memory by the host PC program, four data symbols and four synchronization symbols.

The spread over-sampled symbols are sent from the internal memory of the DSP processor to the D/A converters using the DMAC (Dynamic Access Memory Controller) that is present on the DSP board.

The reception algorithm consists of five parts: parameters setting, acquisition, tracking, decoding and frequency offset recovery. The definition of the receiver parameters like the spreading factor $N$, the over-sampling factor $F_S$, the sample rate $f_S$ and two threshold, $E_{\min}$ and $E_M/E_m$, used in the acquisition algorithm is given by the host PC program. Now we give a general description of the receiver algorithms implemented by DSP target program. The acquisition algorithm scheme is described in [5], the same solution used in the transmitter program was adopted in the receiver, that takes benefit from the use of DMA transfers, and all data transfers can be processed in background without the CPU intervention, hence saving elaboration power. As reported in the block-scheme of Fig. 3, we use a double buffer technique: meanwhile samples related to a received complex symbol are stored to a buffer via DMA, the CPU elaborates the complex symbol previously stored on the other buffer and programmes the parameter set of the DMA for the next transfer. At the end of a transfer, the DMA automatically restarts a new transfer to the initial address of the other buffer and generates a transfer-completion interrupt to the CPU, which start a new routine: whit this methodology, an interrupt is generated only each $N \cdot F_S$ samples (at the symbol time).

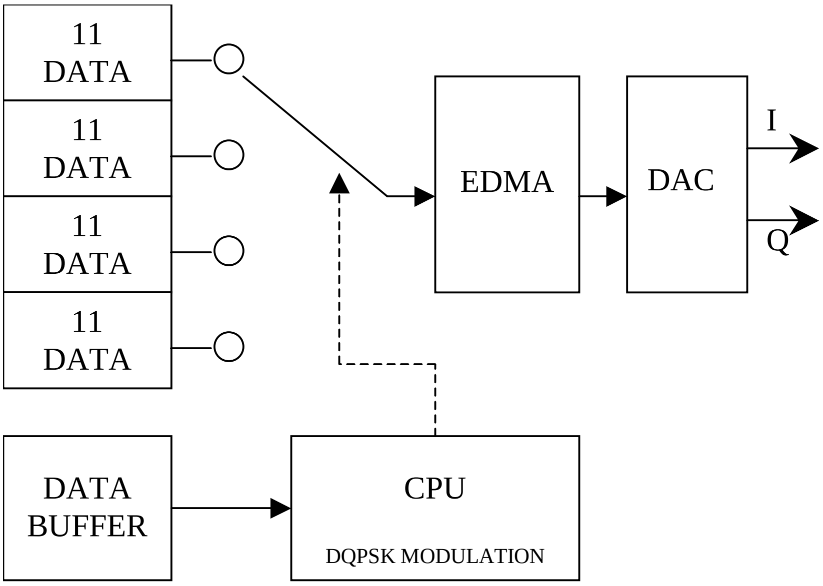


Figure 2 Use of the DMAC during Transmission

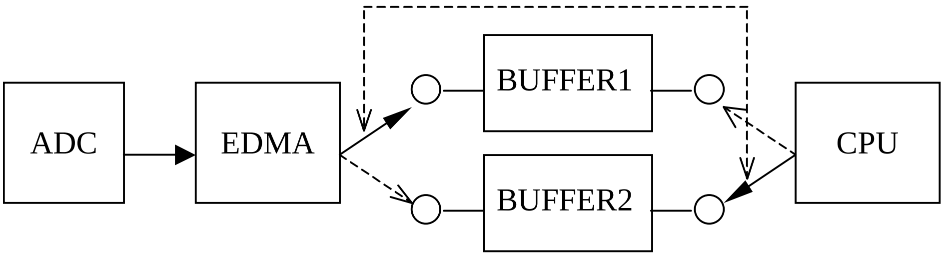


Figure 3 Double Buffer Technique

In the acquisition phase the receiver tries to acquire the synchronism time reference through the detection of the synchronization symbols that are transmitted every $N_{SYNC}$ data symbols.

The time sampling of the received signal is $t_S = T_S / (N \cdot F_S)$. Every $T_{synck} = N_{synck} \cdot T_S$ (the time that separates two synchronization symbols), $N \cdot F_S + N_{corr}$ received samples are put into a memory buffer, where, to simplify the code of the receiver program, $N_{corr}$ is multiple of $N \cdot F_S$, therefore the memory buffer is multiple of one symbol size.

During the $T_{synck} - t_S \cdot N_{corr}$ remaining time, $N_{corr}$ correlations are performed, using different sets of $N \cdot F_S$ samples from the memory buffer, each one shifted of one sample with respect to the previous set. The complexity of this algorithm depends on the number $N_{corr}$ of correlations performed during the $T_{synck} - t_S \cdot N_{corr}$ interval; hence increasing $N_{corr}$ the acquisition time reduces but the CPU utilization increases. The choice on $N_{corr}$ is a trade-off between computational complexity and acquisition time. From every complex correlation output, we calculate the corresponding energy value $E_k = I_k^2 + Q_k^2$, where $I_k$ and $Q_k$ are respectively the k-th correlation values for the in-phase channel and for the quadrature channel. The maximum value of $E_k$ called $E_{\max}$ is compared whit the threshold $E_{\min}$: if it exceeds this threshold we control if $E_{\max} > (E_M / E_m) \cdot E_{med}$, where $E_{med} = \sum_{k=1}^{N_{corr}} E_k / N_{corr}$ is the mean value of the correlation energies $E_k$. If $E_{\max}$ exceeds both the thresholds we enter in tracking mode and the starting position of the correlation related to $E_{\max}$ is taken as synchronism time reference and we pass to the tracking modality.

The choice of the two thresholds is a trade-off between detection probability and false alarm probability as described in [6].

The tracking algorithm scheme is deeply described in [5]. When the receiver enters in tracking modality the synchronism time reference, expressed in terms of sample units, is shifted in the maximum range $\lfloor -N_{shift}/2, N_{shift}/2 \rfloor$ starting from the expected time position of the following synchronization symbol. Considering the computational capacity of the DSP we chose a value of $N_{shift} = 30$. In this way the tracking algorithm allows the energy peak, a maximum time offset of $\pm N_{shift}/2$ samples can be recovered without decoding errors.

As far as the frequency recovery algorithm is concerned, we refer to the reported scheme in Fig. 4. The software implementation of frequency-offset recovery avoids the necessity of a hardware voltage controlled oscillator for each receiver station.

The implemented algorithm is a modified version of the algorithm proposed in [7] here working only during the synchronization symbol and once the code acquisition phase has been completed. It is based on a base-band digital closed loop (AFC) where the error is obtained as the difference between the de-spreading of two version of the base-band signal: the first is shifted in frequency by $+\Delta$ and the second by $-\Delta$ with respect to the actual estimated frequency offset, $\Delta$.

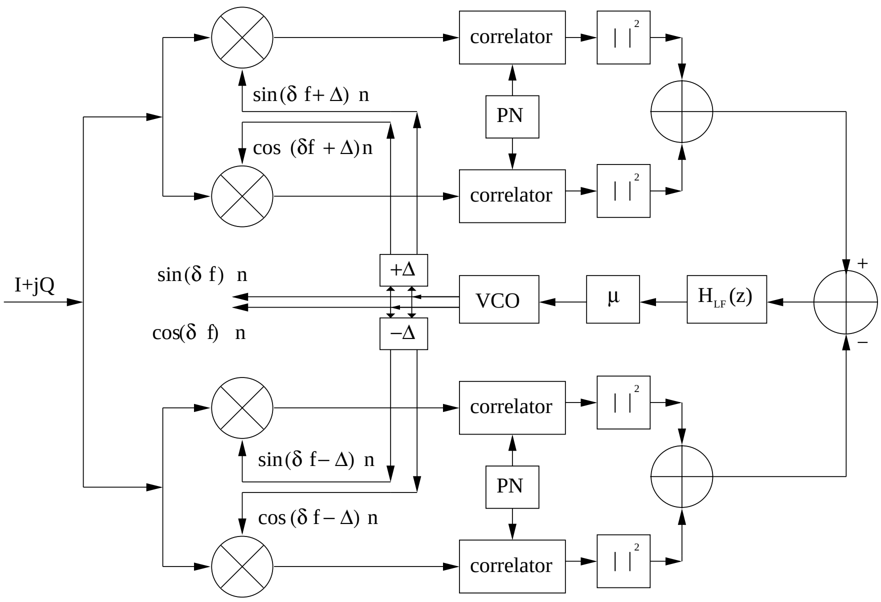


Figure 4 Scheme of the Frequency Recovery Offset Algorithm

The frequency-offset parameter $\Delta$ is a fraction of $B_S$. A low-pass filter filters the signal error; the output of the filter is amplified by a factor $\mu$ to obtain the digital VCO control signal. According to the value of $\mu$ there are two slope behaviours: at high values of $\mu$ the convergence response time is lower but it is more sensible to the signal error and the convergence graphic presents a residual ripple, whereas at low values of $\mu$ the effect of quantization becomes more evident and the response time is higher and convergence is more stable. In all cases the convergence is reached.

The samples of the sinusoids at the frequency $\delta f \pm \Delta$ are calculated at run time by interpolation of reference sinusoid values stored in a lookup table loaded in the internal memory of the DSP during the setting phase.

The receiver algorithm is able to work with frequency deviations in the range of $[-B_S/2, B_S/2]$.

At the reception side host-target interaction is necessary to execute BER measure: significant estimations need a very large number of bit to be transmitted. So the comparison between original transmitted file and received file, requires a lot of memory space: for this reason the building of received file and the host program does the calculation of BER. Therefore it is necessary to transfer received data from internal DSP memory to host PC memory during the reception phase. The same communication procedure is used in transmission side, so it is possible to transfer generic files of any length. When the host PC programs (RXhost and TXhost) receive a service request they set the transmission parameters and begin the interaction with the related DSP target program.

## 4 TRANSMISSION CHAIN AND TELEMEASURED PERFORMANCES

The hardware platform used for the development of the CDMA modem consists of three commercial boards of Texas Instruments: C6711 DSK [8], used for both transmitter and receiver sides; EVM TLV56919-5639 [9] D/A converter and EVM THS1206-1208 [10] A/D converter. Each DSK board is interfaced to a dedicated host computer called TX_Host and RX_Host. The transmission chain is represented in Fig.5 and is principally composed by:

- A DSK C6711 board with the transmitter DSP and a D/A converter;
- A 66-73 MHz Quadrature Passive Modulator (MiniCircuits ZFMIQ-70ML [11]) for the quadrature up-conversion to a frequency of 70 MHz;
- A function Generator Agilent 33250A [12] providing the frequency carrier (local oscillator) for the phase modulator;

- A Noise and Interference Test Set (N.I.T.S.) HP 3708A [13]. It allows the simulation of Additive White Gaussian Noise (AWGN) channels. We use the carrier to noise operating mode to add white gaussian noise on the IF modulated signal;
- A 66-73 MHz Quadrature Passive Demodulator (MiniCircuits ZFMIQ-70D [14]) for the quadrature down-conversion;
- A function Generator (Agilent 33250A) providing a second frequency carrier (local oscillator) for the phase demodulator;
- The A/D converter and the DSK C6711 board with the receiver DSP.

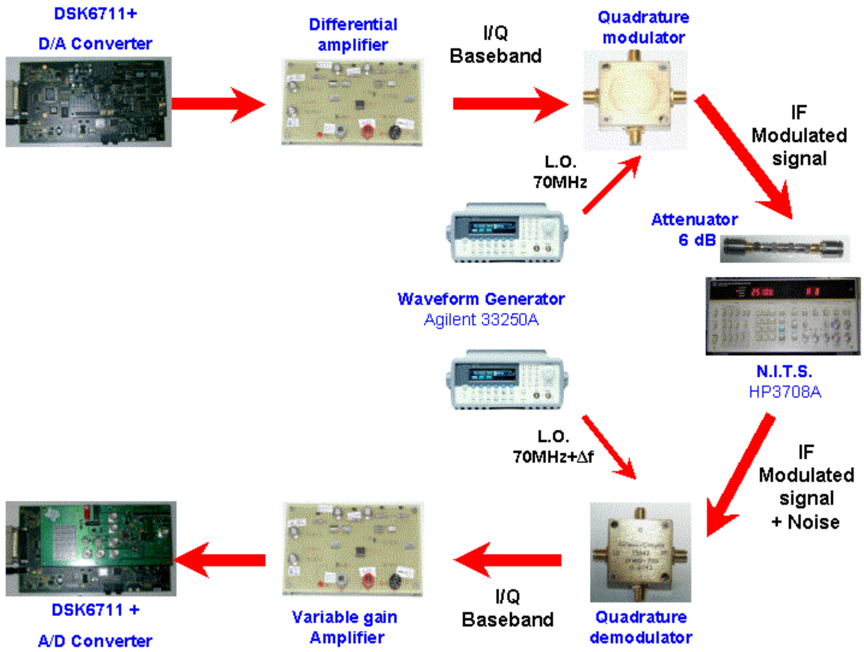


Figure 5 Transmission chain scheme

The analogue signals are switched to the different instruments through a switching matrix [15], the remote control of the matrix is possible, hence not any human operator is needed on the measure benchmark. Fig. 6 shows the system architecture that allows the remote users to obtain measured parameters and control the measurement instruments involved in the transmission chain. The signal matrix provides a convenient way to connect at run time different devices to various instruments without human direct intervention.

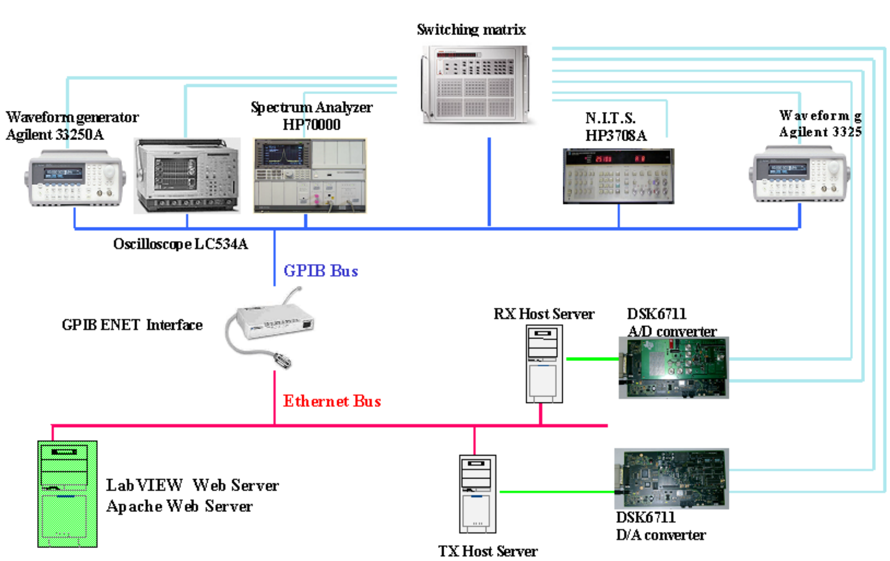


Figure 6 Telemeasure system architecture

When a remote client requires the control of an instrument or the signal matrix, the LabVIEW Web Server returns a WEB page that reproduces the Virtual Instrument front panel. Remote users can perform the signals measurements driving the matrix and the instruments along the transmission chain simple operating on a HTML page. As illustrated in Fig. 6, the instruments and the matrix are connected on the GPIB bus. The GPIB-Enet [16] provides a "bridge" between the IEEE 488.2 and TCP-IP protocols. Fig. 7 shows, for example, the telemeasured spectrum of the modulated signal with a sample rate $f_S = 780 Ksample/\sec$, with an over-sampling factor $F_S = 4$ and a spreading factor $N = 63$.

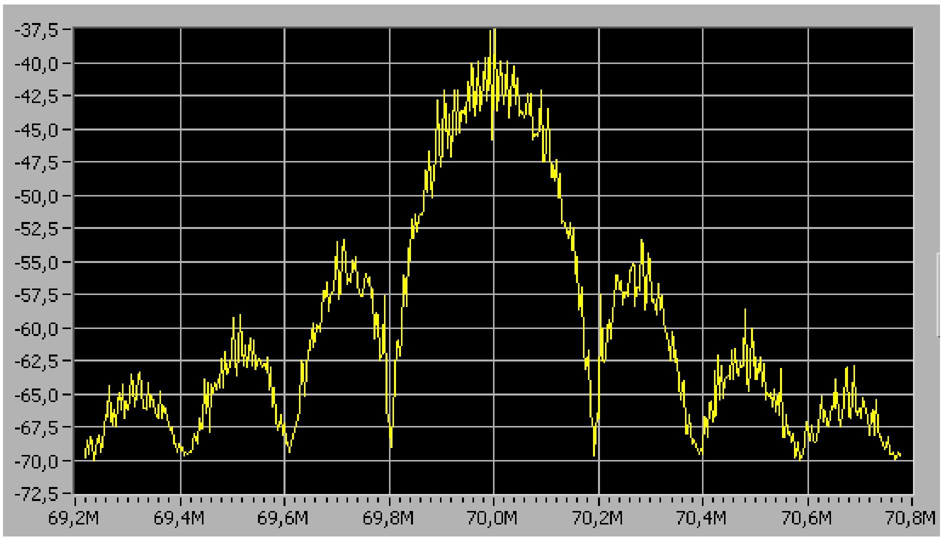

Figure 7 Telemeasured spectrum of the modulated signal

Fig. 8 shows the measured bit error rate (BER) referred to AWGN channel and related to different $E_b/N_0$ set on the N.I.T.S. instrument front panel. The blue curve presents the measured points in absence of frequency offset between carriers, the red one is referred to a normalized frequency offset $\delta f/B_S = 0{,}25$. Both the curves are related to a bit rate $B_r = 29{,}1 Kbit/\sec$ and a spreading factor $N = 7$. We notice the presence of an irreducible error that gives a flat of BER performances which value is approximately 8.E –04 when operating with frequency offset. Fig. 8 shows three estimated frequency offset transients (corresponding to the acquisition phase) against the synchronization symbols number (time), reported for three different values of the parameter $\mu$ of the frequency offset recovery scheme, that fix up stability and convergence speed. The curves are related to a bit rate $B_r = 29{,}1 Kbit/\sec$ and a measured frequency offset between local oscillator $\delta f = -7200 Hz$ corresponding to the maximum recoverable frequency offset at that value of $B_r$ ($\delta f/B_S = 0{,}7$).

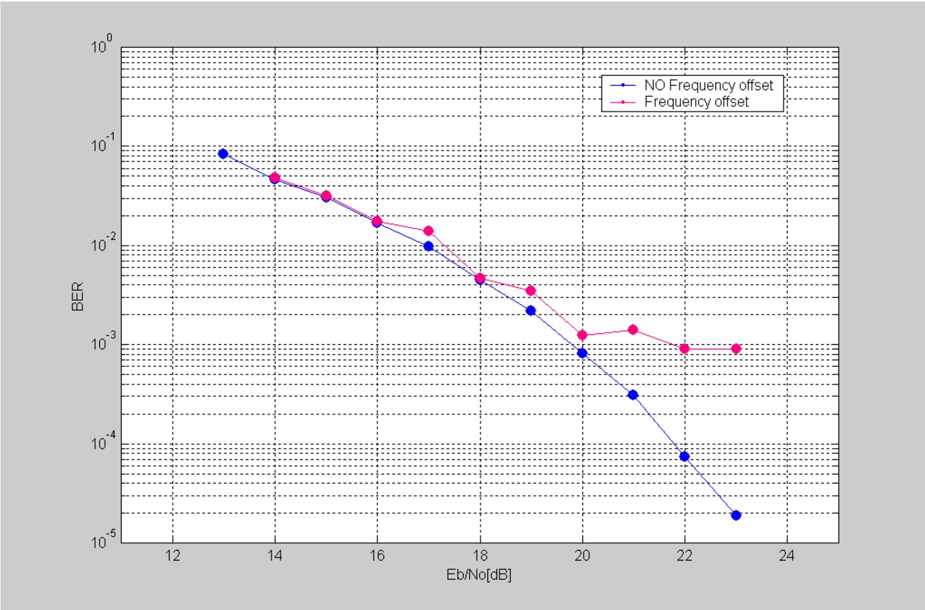


Figure 8 Telemeasured bit error rate vs Eb/No (set on the noise generator).

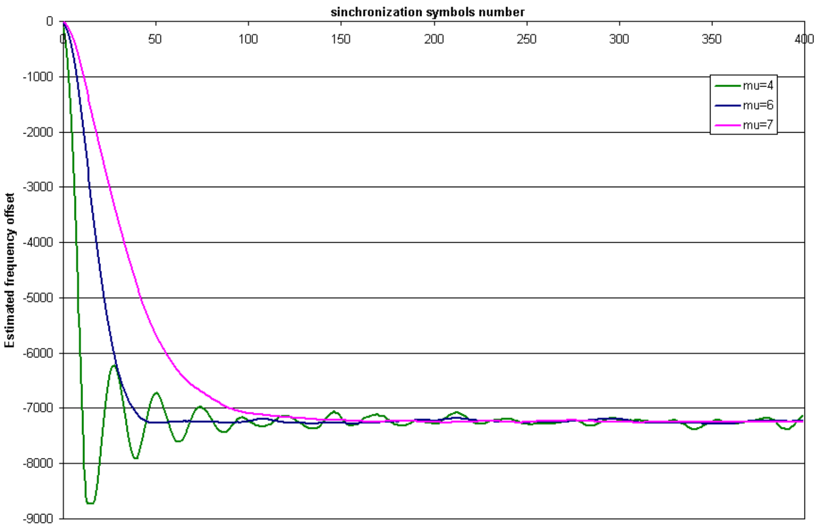


Figure 9 performances of the frequency recovery algorithm

## 5 CONCLUSIONS

In this paper we presented an experience of telemeasurement developed within the software radio technologies, hence not only the instruments but also the devices are remotely configured and controlled; moreover the use of a signal matrix switching allows to characterize the signals in the different sections of the measure chain, hence no human presence is even needed to change the cable connections on the measurement benchmark.

## ACKNOWLEDGEMENTS

The results described in this work were obtained by the support of

- Italian Space Agency and National Consortium for Telecommunications within the project "*Study, Design and Realization of Re-configurable Satellite Network for Multimedia Applications with Guaranteed Quality of Service*";
- FIRB project VICOM, "*Virtual Immersive Communication Project*", Italy

**REFERENCES**

[1] J. Mitola III, “Software Radio Architecture: A Mathematical Perspective”, *IEEE Journal on Selected Areas in Comm.,* Vol. 17, No. 4, April 1999, pp.514-538.
[2] Kenington, P.B.; Bennett, D.W.; Wilkinson, R.J.; McGeehan, J.P.; Sharp, B.; *The application of software radios to integrated satellite/terrestrial communications,* Integration of Satellite and Terrestrial OCS (Digest No: 1997/152), IEE Colloquium on, 7 April 1997
[3] Apache Software Foundation. Available from web: <URL: http://www.apache.org. >
[4] National Instruments. *LabVIEW 7 Express User Manual*.
[5] Integration of Multimedia Services on Heterogeneous Satellite Network an Italian Spatial Agency-National Inter-University Consortium for Telecommunications Project, *Final report,* June 2001.
[6] A. Conti, D.Dardari, *DSP-Based Satellite CDMA Modem: a Low Complexity Implementation,* Wireless Personal Communications 24: pp. 123.139, 2003.
[7] A.Q. Hu, P.C.K. Kwok and T.S. Ng, *MPSK DS/CDMA Carrier Recovery and Tracking Based on Correlation Technique*, IEE Electronics Letter, Feb. 1999.
[8] Texas Instruments, *DSKC6711 Starter Kit Texas Instruments documentation*. Available from web: <URL: http://www.ti.com >
[9] Texas Instruments, *TLV5619-5639 12-Bit Parallel DAC Evaluation Module* (literature number SLAU066). Available from web: <URL: http://www.ti.com >
[10] Texas Instruments, *THS1206, THS12082, THS10064, THS10082 Evaluation Module User's Guide* (literature number SLAU042B). Available from web: <URL: http://www.ti.com >
[11] MiniCircuits, *ZFMIQ-70ML phase modulator data sheet*. Available from web: <URL: http://www.minicircuits.com/dg03-210.pdf>
[12] Agilent Technologies, *Agilent 33250A 80 MHz Function/Arbitrary Waveform Generator User's Guide*.
[13] Hewlett Packard, *HP 37078A Noise and Interference Test Set Operating Manual*.
[14] Minicircuits, *ZFMIQ - 70D phase demodulator data sheet.* Available from web: <URL: http://www.minicircuits.com/dg03-212.pdf>
[15] Keithley Instruments, *Model 707A Switching Matrix Instruction Manual Rev.A.*
[16] J. Humphrey, V. Malhotra, V. Trujillo, Developing Distributed GPIB Test Systems Using GPIB-ENET/100 and Existing Ethernet Networks, National Instruments, Application Note 103. Available from web: <URL: http://www.ni.com >