An Approach
For Designing and Implementing BER Testbed

Dingqing Lu, Ben Niu, and Yahia Tachwali

AGILENT TECHNOLOGIES INC., USA

***Abstract*— Rapid advancements in software and hardware technologies have enabled the development of highly complex and large scale communication systems. Current Electronic Design Automation (EDA) software facilitates the development of various signal processing techniques and processing algorithms through system modelling and simulation. In addition, the processing speed of current embedded systems has made complex, yet reliable algorithms, feasible to implement. Nevertheless, there is a critical gap in the development of such complex systems between the algorithmic development phase using simulation software and the implementation phase using the targeted hardware. This gap is filled by another phase known as the rapid prototyping phase through software-hardware cosimulation using Testbeds. A Testbed presents the opportunity to capture the impact of a real operation environment and ensure the functionality of the design in practice. In this paper, an approach to design and implement a Bit Error Rate (BER) Testbed is proposed. The BER is one of the most important performance metrics for communication systems. This paper presents the Core Software Package (CSP) that is used to design and implement all basic parts in a communication Testbed including a signal generator to support different signals, a channel emulator to emulate test environments, such as channel multipath fading and interferences and a receiver to support multi-system structures with time/frequency synchronization and phase error correction capability. The implementation results presented in this paper reflect the effectiveness of this test and verification approach for developing communication systems**

I. INTRODUCTION

System Testbed is very important for system performance evaluation. Many research works [1-2] on Testbeds for communication systems have been done. Each of them provides solution for a specific system test purpose. Many Testbeds are based on specific instruments with limited signal type and fixed system structure [4] and do not provide advanced receiver performance test measurements such as BER, FER, receiver sensitivity and selectivity [5].

A Testbed for complex communication systems, such as military communication systems, needs to support different types of signals, flexible receiver structures with real world environments and enough measurements for both transmitter (Tx) and receiver (Rx) performances. Another consideration is that Military system should be configurable easy enough for different test cases. To fit these requirements, the Testbed must have a flexible source and the receiver should be easy to customize. In Figure 1, a proposed Testbed structure is given.

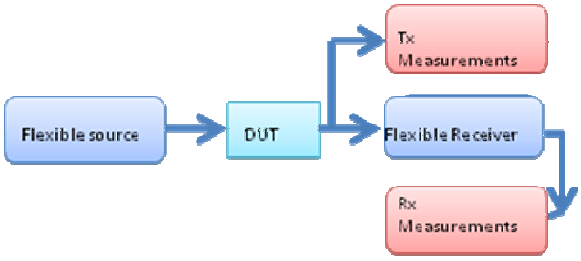

Figure 1. System Testbed basic structure

In this paper, a methodology to design and implement the Testbed is proposed. A CSP, in which communication models and system templates to support different test cases are collected together, is used to generate test signals in simulation, process received signals, measure processed data and then link to instruments for HW test purpose. After the evaluation completed, code generation is performed. Then, the CSP implements the design into FPGA for a real time Testbed.

The main contributions of this paper are:

- The flexibility of the developed communication Testbed which allows verifying communication receivers using wide range of standardized waveforms.
- The efficiency of the Testbed implementation which enables time-consuming measurements such as BER to be obtained in real-time fashion.

The remainder of this paper is organized as the follows. In Section II, a proposed Testbed based on software is constructed. In Section III, we proposed a way to make flexible test signal sources with test environments. Also a generic receiver structure is given to handle timing synchronization, the phase/frequency error correction. In Section IV, the real time receiver based on FPGA will be implemented. Test results will be presented to show the good performance. Section V gives the summary of this paper.

II. SYSTEM MODEL

To design the Testbed, first we need to create a system simulation design using CSP as seen from the Figure 2. As seen from Figure 2, the Information source is for generating baseband data in the simulation. For each signal frame, a preamble is inserted before the information data. And then the framed signal is mapped to certain formats such as M-QAM, M-PSK, etc,. As an example a QPSK signal format is used. Secondly, the framed baseband signal will be encoded and

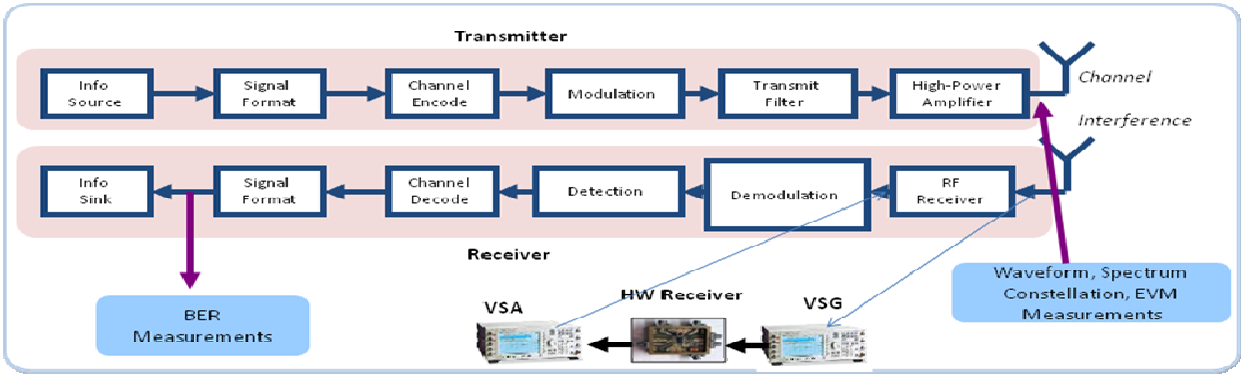

Figure 2. Simulation design for creating the Testbed

sent to RF modulator in which RF components, an Oscillator and a Mixer are used. The Oscillator with phase noise will affect the system performance. The RF signal will be sent through a Channel to the receiver in which both RF and DSP processing are considered.

The transmission signal can be described as

$$S(t) = A(t)\exp(\ j(2\pi f_c t)) \qquad (1)$$

At the receiver input the equivalent baseband received signal can be described as

$$S(t) = A(t)\exp(j(2\pi f_c t + \alpha + \varphi(t)) + n(t) \qquad (2)$$

In (1) and (2), *A(t)* is the amplitude of the signal, f_c is the carrier frequency and $\varphi(t)$ is the phase noise caused by the oscillator. As we know, oscillators are linear time-varying (LTV) systems [3]. So, $\varphi(t)$ can be handled as a stationary random process. In (2), *n(t)* is the channel AWGN noise with zero mean and No/2 power density. Without loss of generality, the initial phase α can be set to zero.

Note that the signal generation design in the Figure 3, can be easily customized based on the user's requirement. So, a flexible signal source is available to fit the Testbed requirement.

The system model will be used to study the system performance under the phase noise effect. For sampling time step of *Ts* in simulation, the received signal samples is given by

$$S_k = A_k \exp(j(2\pi f_c k T_s + \varphi_k) + n_k \qquad (3)$$

where

A_k is the k[th] sample of the amplitude of the signal,

φ_k is the phase noise sample caused by the oscillator, and

n_k is the AWGN channel noise sample.

In Figure 2, for each simulation model, there is a compliant simulation library in CSP to support many communication systems, including GSM, Edge, CDMA, WCDMA and OFDM either for commercial or military system. Before implement the Testbed, the simulation design is used for verification. Then, the Testbed will be created by using the simulation design.

III. TESTBED USING CSP

First let us construct a Testbed using the CSP. There are two configurations for the Testbed in this case.

1. Create Signal Source

To make a signal source for testing your DUT (design under test), let us refer the simulation design in Figure 2. For example, now the DUT is a RF Receiver. At the input of the RF receiver an instrument link is set up to download simulation data to a Vector Signal Generator (VSG). The test signal will be available for the DUT that is a hardware (HW) receiver in this test case.

In Figure 3, a simple example is given to explain how to create a custom waveform generator in details. First, bits sequence is generated using a bit pattern model, then format or frame it based on the waveform specification. An IF modulation is used followed by RF up conversion to transmit the signal at the communication band. To consider the communication channel distortion, a channel model is inserted. Multipath fading and channel noise plus interference can be specified by the user. A VSG link model is used to link the simulation data to the VSG instrument. Before downloading data to the instrument, the simulation waveform and spectrum can be measured. If the measured results are satisfactory, the VSG link can be activated to generate the test data to test the DUT.

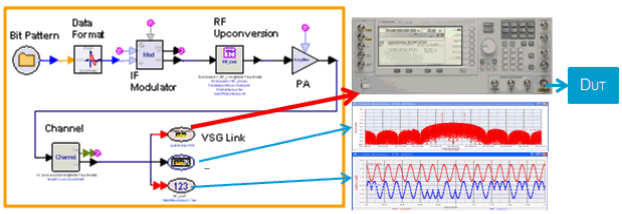

Figure 3. Simulation design for the Testbed

2. Software Receiver

To test the transmitter, we need a reference receiver. During the R&D development period, the receiver has not been built yet. So, a software receiver in CSP can be used for this purpose.

To make a reference software receiver, basic models for demodulation, decode, de-frame can be found directly from CSP library. However, some signal processing algorithms need to be created if the model is not in the current library. For example, let us consider a phase error correction model. The phase noise caused by the oscillator in the RF modulator $\varphi(t)$ is modelled as a correlated Gaussian noise with a non-zero mean value and a power density spectrum.

The signal is modulated using QPSK modulation technique. The constellations are measured at the receiver for the system with and without phase noise in Figure 4a and Figure 4b, respectively.

To analyse the measurement results, an estimated averaging signal states are defined as

$$\hat{S}^n = \frac{1}{N}\sum_{i=1}^{N} S_i^n, \quad S^n \in Q^n \text{ , n=1,2,3,4,…}$$

where N is the number of the signal complex random samples, S_i^n is located in the quadrant n, where n=1,2,3,4 and i = 1,2,…, M.

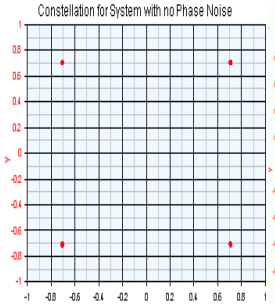

Figure 4a. Idea Constellation

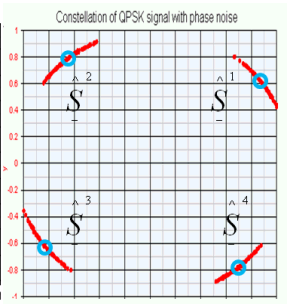

Figure 4b. Constellation with Phase Noise

In comparison with the Figure 4a, the four averaging signal states have a rotation with respect to the ideal states. In fact, a phase error is identified. Further observation shows that the phase rotation angle (phase error) depends on the mean value of the phase noise. When the mean value increases, the phase error increases. The phase error caused by the phase noise will affect the system performances dramatically, especially its BER performance. So, we have to find a way to correct the phase error.

An algorithm to correct the phase error is proposed for improving the system performances.

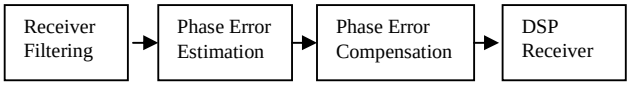

Figure 4. Phase Error Correction

Assume that a system with frequency error correction, we observe a block of M samples in the preamble portion. The k_{th} complex sample can be expressed as

$$\tilde{S}_k = a_k e^{j\varphi_k} + n_k \qquad (3)$$

where a_k, φ_k and n_k are the k[th] amplitude, phase and the amplitude noise samples of the preamble.

Once the preamble arrives the phase error estimator, the phase error estimator performs a correlation on the received preamble with original preamble, the phase error at this moment, $\hat{\varphi}$ will be estimated by

$$\hat{\varphi} = \tan^{-1}(Q/I) \qquad (4)$$

where $I = \frac{1}{M}\sum_{k=1}^{M} S_k^I, Q = \frac{1}{M}\sum_{k=1}^{M} S_k^Q$, S_k^I, S_k^Q

Then, the phase error compensator will insert a $-\hat{\varphi}$ to cancel the phase error.

A simulation design is created for the phase error correction algorithm in Figure 5. The model can be implemented using either C++ code or Math Lang code. Since the phase error correction algorithm is used, the Constellation results are improved as shown in Figures 6 and 7, in which the averaging signal states are perfectly align with the ideal QPSK constellation.

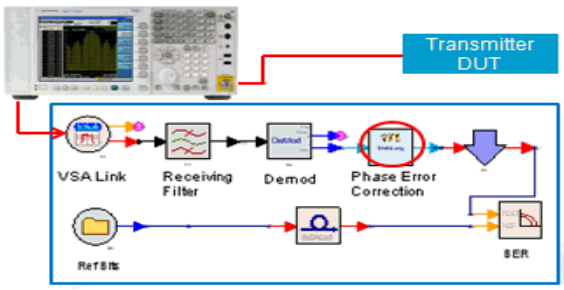

Figure 5. Block Diagram to create Software Receiver to test Transmitter

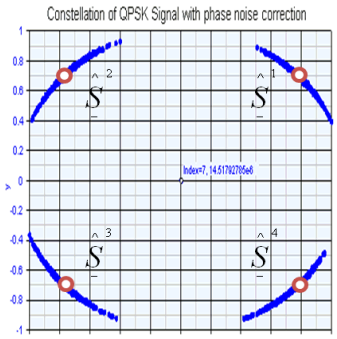

Figure 6. QPSK Constellation for System with Phase Noise using Phase Correction

The phase error correction also works for the system with both amplitude and phase noise.

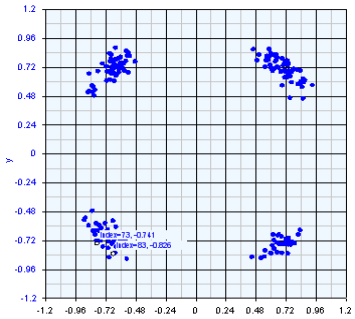

Figure 7. QPSK Constellation for System with Amplitude and Phase Noise using Phase Correction

To describe the system performance more accurately, the receiver BER is measured using the Testbed BER measurement. The first test case is that the phase correction (PC) algorithm is not used. As expected, the BER is very high (about 48%). Next, the phase correction algorithm is used and the BER value is close to the theoretical value.

For the system with both amplitude and phase noise, the BER is also measured. As we know, without the phase correction the BER is very high and with the phase correction the BER is reasonable.

Theoretical BER for system without noise	With Phase Noise and no PC	With Amplitude plus Phase Noise and no PC	With Phase Noise and PC	With Amplitude plus Phase Noise and PC
0	48%	50.3%	1e-6	2.1 e-5

Table 1. BER test results for 1e6 simulation samples

IV. TESTBED USING FPGA

In III, we have shown that the Testbed using CSP can be used for R&D testing, especially during the developing period. However, there is a disadvantage that is the BER test is time consuming. Long simulations are required to obtain statistically accurate BER measurements. To speed up the BER test, a Testbed using FPGA is designed and implemented for a real time test as depicted in Figure 8. The BER tester includes a digitizer to process the input RF/IF test signal, a FPGA card for the receiver and BER measurement functions and a control card to pass and display data. The test signal is generated by using CSP in Simulation and then is downloaded to a VSG. Also the clock signal is generated by another signal generator.

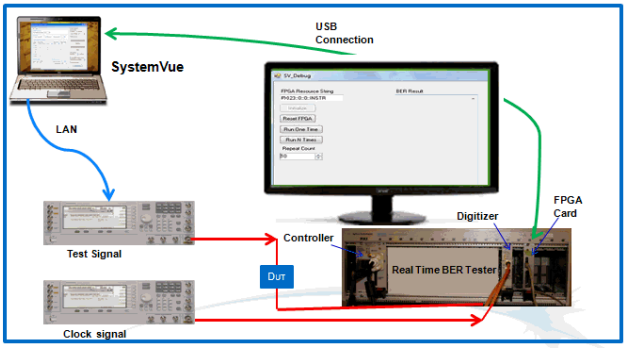

Figure 8. A BER Testbed based on FPGA

The CSP can be used as a design tool at this moment plays important role to design and implement the Receiver and BER measurement functions in the FPGA. In Figure 8, the procedure to design and implement the FPGA receiver and BER measurements are provided.

From Figure 9, three validations and tests are performed in CSP: Floating point, fixed point and bit-accurate cycle accurate (HW), respectively. Since the CSP can support all these requirements and integrates all parts together, the procedure can be down very smoothly and in a user-friendly fashion.

In the Testbed, receiver test signal can be generated by using VSG or recorded data in the field also can be used in the Testbed. The TX test results are shown in Figure 10, in which the QPSK transmission signal is measured. As a part of the BER test, a recorded PN data with injected errors were used and the cumulated Bit errors are measured as seen from Figure 11. In Figure 12, the original transmitted bits are compared to the bits recovered by the FPGA receiver. As can be seen they are aligned very well and the BER is close to zero for the High signal noise ratio of S/N=25 dB with limited test time.

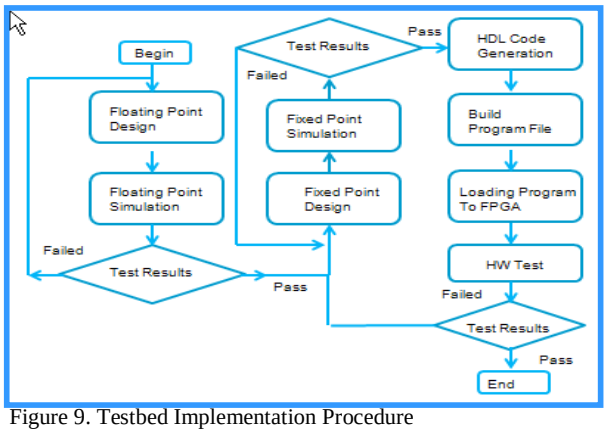

Figure 9. Testbed Implementation Procedure

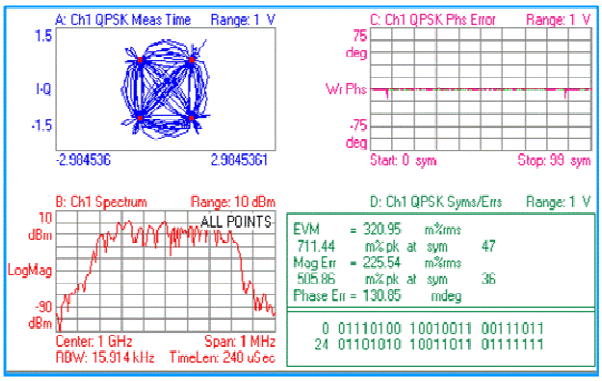

Figure 10. QPSK signal Constellation, EVM, and Spectrum

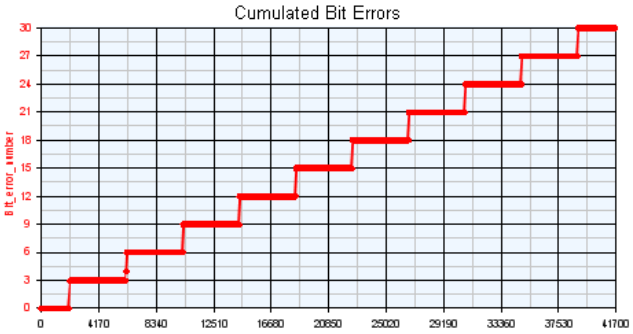

Figure 11. Cumulated Bit Errors

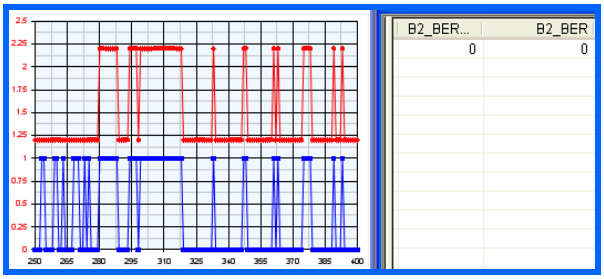

Figure 12. Original Vs Recovered Bits and Bit Error Rate

V. Conclusions

A new approach for designing and implementing BER Testbed has been presented. In the approach, the CSP gives flexibility to the signal generator, the environment setup and receiver structure. The CSP also simplifies the procedure of design as well as implementation. For R&D testing purpose, the CSP can directly construct a Testbed to do the transmission and receiving test as discussed in Section III. A real time Testbed also can be implemented using the CSP as described in Section IV.

Reference

[1] Suzuki, T., Iwai, H., Takeuchi, Y. , Watanabe, F. “A study on receiver performance of a wideband CDMA Testbed system for IMT-2000”, Vehicular Technology Conference, 1999 IEEE 49th, 1732 - 1736 vol.2.

[2] Hannikainen, M.; Vanhatupa, T.; Lemilainen, J.; Saarinen, J., “Windows NT Software Design and Implementation for a Wireless LAN Base Station” ACM International Workshop on Wireless Mobile Multimedia, 1999 (WoWMoM1999), pp. 2-9.

[3] Lee, T. H. and Hajimiri, A."Oscillator Phase Noise: A Tutorial,” IEEE J. Solid-State Circuits, VOL. 35, NO. 3, March 2000.

[4] Gustafsson, O.; Amiri, K.; Andersson, D.; Blad, A.; Bonnet, C.; Cavallaro, J.R.; Declerck, J.; Dejonghe, A.; Eliardsson, P.; Glasse, M.; Hayar, A.; Hollevoet, L.; Hunter, C.; Joshi, M.; Kaltenberger, F.; Knopp, R.; Le, K.; Miljanic, Z.; Murphy, P.; Naessens, F.; Nikaein, N.; Nussbaum, D.; Pacalet, R.; Raghavan, P.; Sabharwal, A.; Sarode, O.; Spasojevic, P.; Yang Sun; Tullberg, H.M.; Vander Aa, T.; Van der Perre, L.; Wetterwald, M.; Wu, M.; , "Architectures for cognitive radio testbeds and demonstrators — An overview," Cognitive Radio Oriented Wireless Networks & Communications (CROWNCOM), 2010 Proceedings of the Fifth International Conference on , vol., no., pp.1-6, 9-11 June 2010

[5] Pawelczak, P.; Nolan, K.; Doyle, L.; Ser Wah Oh; Cabric, D.; , "Cognitive radio: Ten years of experimentation and development," Communications Magazine, IEEE , vol.49, no.3, pp.90-100, March 2011