Hindawi Publishing Corporation
Journal of Engineering
Volume 2013, Article ID 936495, 7 pages
http://dx.doi.org/10.1155/2013/936495

Research Article

Contribution to Synchronization and Tracking Modelisation in a CDMA Receiver

Aicha Alami Hassani, Mohcine Zouak, and Mostafa Mrabti

Signals Systems and Components Laboratory (LSSC), Electrical Engineering Department, Faculty of Sciences & Technology, P.O. Box 2202 FEZ, Morocco

Correspondence should be addressed to Aicha Alami Hassani; alamiaicha3@hotmail.com

Received 31 December 2012; Accepted 9 March 2013

Academic Editor: René Yamapi

We propose and analyze a noncoherent receiver with PN code tracking for direct sequence code division multiple access (DS-CDMA) communication systems. We employ the delay-lock loop (DLL) architectures for the tracking stage. The choice of DLL parameters is studied with special focus on DS-CDMA communication systems and orthogonality conditions. We described the modeling and simulation of the NCO using hardware description language VHDL. Details of the VHDL implementation are shown.

1. Introduction

Wireless access systems techniques are gaining worldwide interest in the academic and industrial community [1]. Among these systems direct sequence CDMA is the well-used one. The most important part of the baseband processing is the receiver, because it requires knowledge of one or several parameters, such as the users' code timings, powers, and carrier phases. Moreover, in general the code timing needs to be estimated with high accuracy, since timing errors have a large impact on the performance of many detectors [2]. Therefore, accurate code synchronization acquisition and tracking is an essential task to achieve good performance of a DS-CDMA system.

The design of synchronization techniques that are robust against external interference is of fundamental importance in many situations, such as in military or safety critical applications. The key issues in synchronization are code acquisition and tracking. Tracking stage is the "heart" of spread spectrum module. Indeed, this stage determines the performances of the global communication, that is to say the process gain and the acquisition time values. The code tracking is a continuous operation after the code phase has been acquired. The conventional approaches to timing acquisition and tracking are the correlator and the delay-locked loop (DLL).

From delay-locked loop (DLL) architectures, the most significant and promising tracking scheme to be studied is the early-late DLL. In order to achieve accurate and high-speed digital communication systems, the problem that must be resolved is how to get controllable, high precision, and high-frequency signal, more specifically, to produce ideal wave samples, which have variable frequency, numerical controlled oscillator is the essential unit of the digital communication system [3].

In this paper, we propose a simple and accurate VHDL description for tracking and synchronization parameters in DS-CDMA systems in the 802.11 standard context. The proposed architecture is implemented on FPGA.

First, start from the working principle, top-level structure diagram of controlled oscillator is given. Second, it writes a simulation program based on a detailed analysis of its functioning. Finally, the use of MODELSIM simulation is analyzed to verify the correctness of the design.

The remainder of this paper is organized as follows. Section 2 provides an overview of synchronization scheme, a review of tracking, and the system model, and the obtained results are discussed. Section 3 outlines some concluding remarks.

2. Synchronization VHDL Modeling

Usually the problem of timing acquisition is solved via a two-step approach as follows.

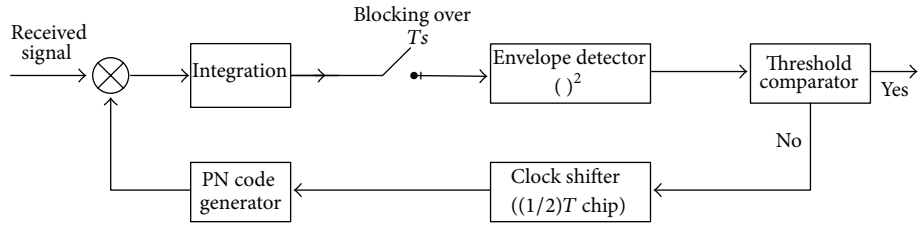

Figure 1: Serial acquisition scheme.

(i) Initial code acquisition (coarse acquisition or coarse synchronization) which synchronizes the transmitter and receiver to within an uncertainty of $\pm T_c$ (chip time).

(ii) Code tracking which performs and maintains fine synchronization between the transmitter and receiver.

Given the initial acquisition, code tracking is a relatively easy task and is usually accomplished by a DLL. The tracking loop keeps on operating during the whole communication period. If the channel changes abruptly, the DLL will lose track of the correct timing and initial acquisition will be reperformed.

In the next part, we present our synchronization model implemented in VHDL. We distinguish, in a first point, the acquisition part that is focused on correlation principle. And in a second point, we focus on the tracking part in which we describe the DLL and its various organs.

2.1. The Acquisition. As mentioned before, the objective of initial code acquisition is to achieve a coarse synchronization between the receiver and the transmitted signal. In a direct sequence spread spectrum (DS-SS) system, this is the same as matching the phase of the reference spreading signal in the despreader to the spreading sequence in the received signal. Several acquisition techniques exist such as serial search, parallel search, multidwell detection, and matched filter acquisition [4]. The acquisition strategy we choose is serial search. It is based on the following basic working principle depicted in Figure 1. In this method, the acquisition circuit attempts to cycle through and test all possible phases one by one (serially).

This acquisition principle is based on a correlation evaluation between the received signal and the locally generated PN code. For that, the autocorrelation function $R_c(t)$ is computed by a multiplier and an integrator/dump. The threshold comparator makes it possible to compare the computed value with a configurable threshold. Indeed, if the cross-correlation function is not maximal, a command circuit has to regenerate a time-shifted PN code until $R_c(t) = 1$.

The Integrator is an important organ of the system; it implements the function of a low-pass filter. This communication block is made up of a shift register that adds samples over a symbol time interval T_s, which depends on the code length (N) at the end of this interval resets. The symbol latch signal simultaneously resets the accumulator register

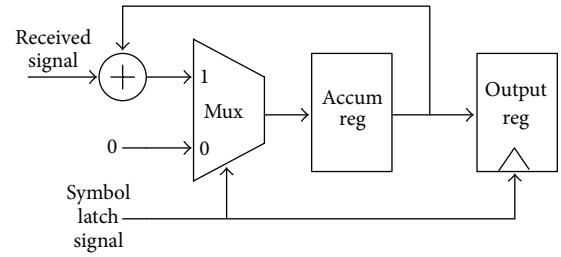

Figure 2: Integrate and Dump.

for the next symbol and latches the current accumulated symbol energy in the output register. The accumulator is run at system clock rate N times higher than the symbol rate. A symbol time is detected on the rising edge of the "dump" signal. This signal "dump" resets the integrator and starts the next phase of integration. The useful size for the accumulation function is the number of samples per symbol which is the number of samples (4 samples) over a chip time T_c multiplied by the length of the code N. It can be implemented as shown in the Figure 2.

The integrator becomes a binary matched filter in the event of the incoming signal being rectangular signal pulses. The following algorithm describes the function of the Integrator [1].

if count < N then
count ← count + 1
sum ← sum + data
else
output ← sum
count 0
end if

Figure 3 shows the simulation results of the integrator. It is noticed that the output signal of the dispreading function is generated on each rising edge of "dump", and it validates the operation principle of the integrator.

2.2. The Tracking. After the initial acquisition, which is a coarse search process, a fine synchronization known as tracking starts. The tracking maintains the local PN code in synchrony with the incoming signal to within half a chip time. There are two methods to implement code tracking in DS-SS

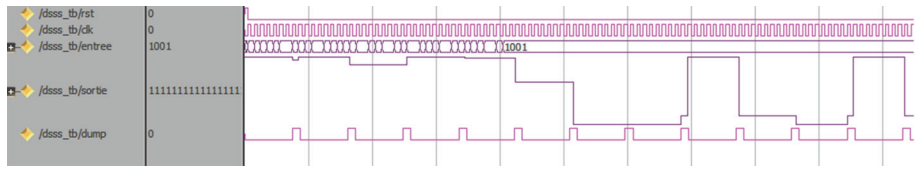

Figure 3: Simulation results for the integrator using MODELSIM.

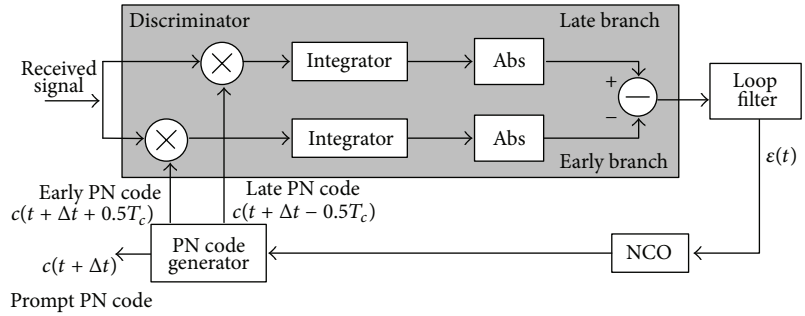

Figure 4: Delay Locked Loop.

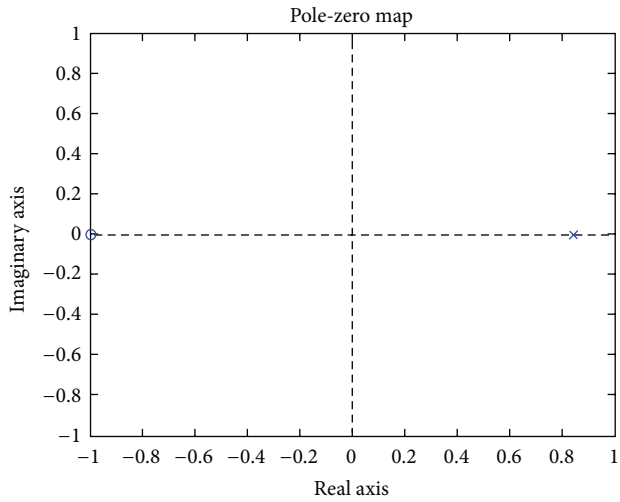

Figure 5: Stability diagram of the filter.

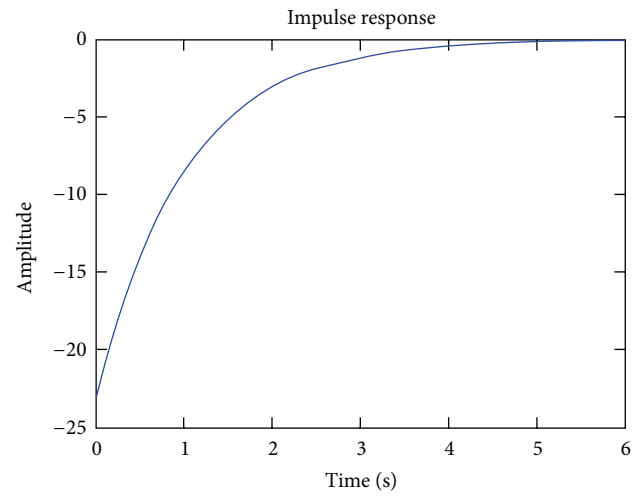

Figure 6: Impulse response $h(t)$.

systems: delay-locked loop (DLL) and tau-dither loop (TDL) [4]. The TDL uses a single correlator; however, the signal power is 3 dB smaller than in the DLL, and the tracking jitter is larger.

In the DLL, shown in Figure 4, the incoming signal is multiplied with two outputs of the local PN code generator which are delayed mutually by $T_c/2$. The cross-correlated values are filtered and envelope detected and subtracted.

This error signal is passed through a loop filter which drives a voltage controlled oscillator. VCO's output drives the PN generator. Thus, the error signal changes the phase of the local replica code so as to synchronize with the acquired signal. If the cross-correlator output from one correlator is greater than the other, then the VCO clock frequency is either advanced or retarded. This change tries to generate a PN code which produces a zero error signal as input to the VCO.

An early-late correlator is an example of a delay-locked loop which is used in tracking. The codes generated are c_E, c_P, and c_L referring to the early, prompt, and late codes, and they have a relationship as shown below [5]:

$$c_E(t) = c\left(t + \Delta t + 0.5T_c\right),$$
$$c_P(t) = c\left(t + \Delta t\right), \quad (1)$$
$$c_L(t) = c\left(t + \Delta t - 0.5T_c\right).$$

Functions achieving the tracking loop operations (Figure 4) are as follows.

(i) The discriminator, determining the tracking error.

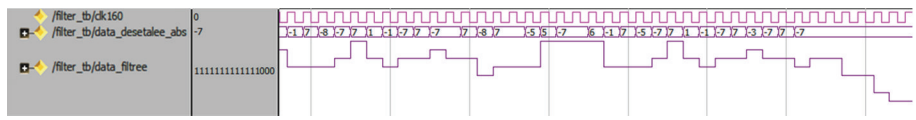

Figure 7: Digital RII simulation.

(ii) The loop filter, averaging the tracking error.

(iii) The numerical controlled oscillator (NCO), generating a 160-MHz (associated with $T_c/4 = 6.25$ ns: sampling time) clock modulated by the error value.

2.2.1. The Discriminator. The discriminator is based on an early/late correlator solution for the DLL design. The correlation digital functions associated with each branch are a multiplier, an integrator, and the computing absolute value. If the "early" branch correlation result is greater than the "late" one, a positive tracking error is created and will generate a Tearly = 6.25 ns + error influence period at the oscillator output (NCO). In the case of a "late" branch preponderance, the NCO period is equal to (6.25 ns-error influence).

2.2.2. The Loop Filter. Digital loop filter design is inhibiting noise and high frequency component same as to the analog loop filter. It also controls speed and precision of loop correction phase, which is a deciding factor to the dynamic performance and static performance of DLL. Besides filtering ripple wave, loop filter has another important effect; it decides transmission performance of phase lock loop control [3].

The digital filter is implemented with a first order low pass filter described by the following transfer function:

$$H(p) = \frac{1}{1+\tau p}. \tag{2}$$

To pass from Laplace domain towards the sampled domain, we use the bilinear transform:

$$\begin{aligned} H(z) &= H(p)_{|p=(2/T_e)\ ((1-z^{-1})/(1+z^{-1}))} \\ &= \frac{1}{1+(2/T_e)\times\left((1-z^{-1})/(1+z^{-1})\right)\times\tau} \\ &= \frac{\left(T_e/(T_e+2\tau)\right)+\left(T_e/(T_e+2\tau)\right)z^{-1}}{1+\left((T_e-2\tau)/(T_e+2\tau)\right)z^{-1}}, \end{aligned} \tag{3}$$

as

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0+b_1z^{-1}}{a_0+a_1z^{-1}}, \tag{4}$$

so

$$b_0 = b_1 = \frac{T_e}{T_e+2\tau} = 0.0805, \qquad a_0 = 1, \tag{5}$$

$$a_1 = \frac{T_e-2\tau}{T_e+2\tau} = -0.839. \tag{6}$$

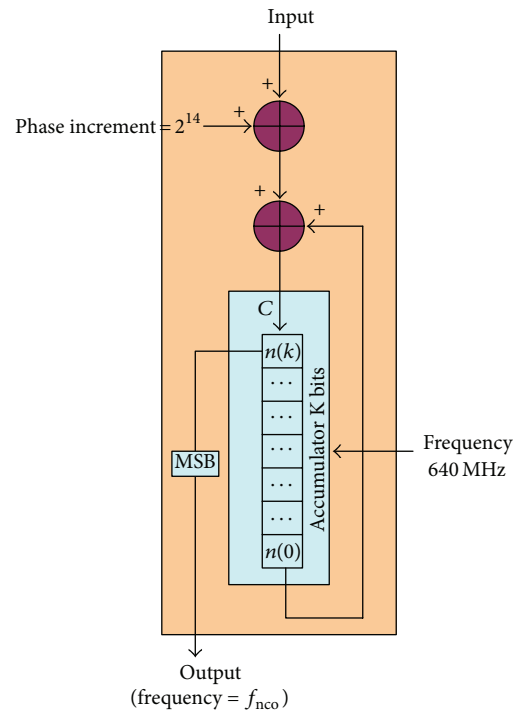

Figure 8: Numerical Controlled Oscillator.

For $\tau = 1\,\mu$s and $T_e = 25$ ns $\times\ 7 = 175$ ns filter sampling time for a code length 7, we obtain a digital infinite impulse response filter (RII); the equation is

$$y(n) = 0.0805x(n) + 0.0805x(n-1) + 0.839y(n-1). \tag{7}$$

Its stability is checked using MATLAB. The results confirm that the filter is a stable first-order filter since the poles are located inside the circle of unit radius $R = 1$ (Figures 5 and 6).

For the VHDL implementation, we do not use floating point values on FPGA; instead, we will multiply all the coefficients by 2^{13}, round to the nearest integer, and then shift the result to the right by 13 bits to effectively divide by 2^{13}. 13 is an arbitrary number of bits that gave a reasonable resolution to the floating point values calculated:

$$y(n) = \left(2^{-13}\right)\left[659x(n) + 659x(n-1) + 6873y(n-1)\right]. \tag{8}$$

The component implements the recursive equation (8). The design requires one previous value for the output and one previous value for the input. The (2^{-13}) multiplication in the

Figure 9: Output of the NCO whith and whitout command word.

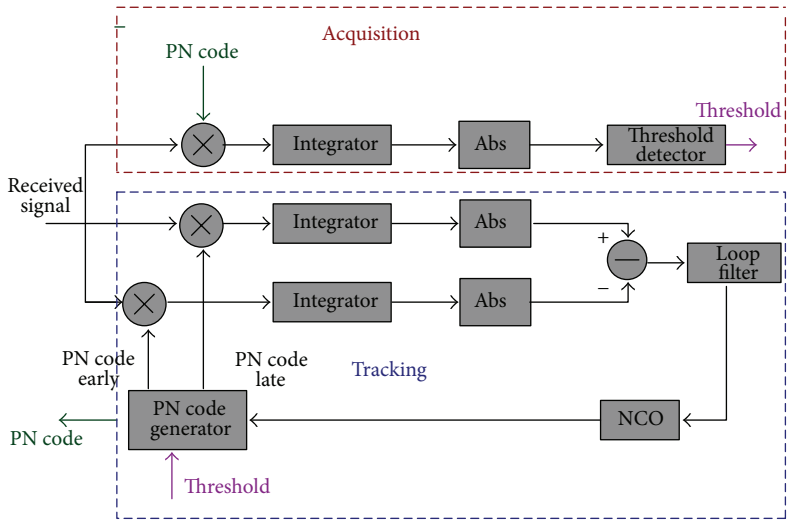

Figure 10: Synchronised code generator whit serial-search acquisition and tracking.

difference equation is accomplished with a shift to the right by 13 bits [6, 7].

The model is then simulated with MODELSIM using a sampling period of 6,25 ns. The simulation results are given on Figure 7. These results are in agreement with those obtained using MATLAB; this validates the filter model.

2.2.3. The NCO. The numerical controlled oscillator (NCO) forms an important part of digital receiver especially of DLL as it is used in the timing recovery. An NCO is the digital counterpart of an analog voltage controlled oscillator. Based on the error voltage, the output frequency is changed in an analog VCO. Similarly, in a digital NCO, based on the error input word, the output frequency is altered. An NCO consists of an accumulator, to which an incoming error signal is added (Figure 8). This error signal decides the output frequency of the NCO. If f_e is the clock frequency, ϕ the incoming error signal magnitude, and K is the number of bits of the accumulator, the free running output frequency of the NCO which is the most significant bit (MSB) [3] is given as:

$$f_{\text{clk\_nco}} = \left(\frac{f_e}{2^K}\right) \times [C], \tag{9}$$

$$C = \text{input word } (\phi) + \text{ phase increment.}$$

The NCO generates a square wave whose frequency is controlled by the error. As described above, it is used to change the frequency of the clock, thereby, the clock timing.

The frequency resolution is defined by

$$\Delta f = \left(\frac{f_e}{2^K}\right). \tag{10}$$

For the input frequency, we choose 640 MHz which is the multiple by 2^n of the central frequency 160 MHz (corresponding to $T_c = 25$ ns).

The chosen structure of the NCO for this configuration is

(i) the register length is 16;

(ii) the phase increment is $16384 = 2^{14}$;

(iii) the resolution is 9.8 KHz.

The simulation result of the NCO is shown in Figure 9. It can be seen that the output frequency variation of the NCO (clk_nco) depends on the input command word and without command word.

2.3. Acquisition and Tracking Association. The aim of this part is to realize a complete model of a CDMA receiver in VHDL. In Figure 10, the different elements of the model are shown, where we can see the block diagram of all parts receiver including synchronization.

2.3.1. PN Code Generator of the DLL. The spreading sequences used for the CDMA must be chosen to respect the orthogonality condition. Thanks to the auto correlation

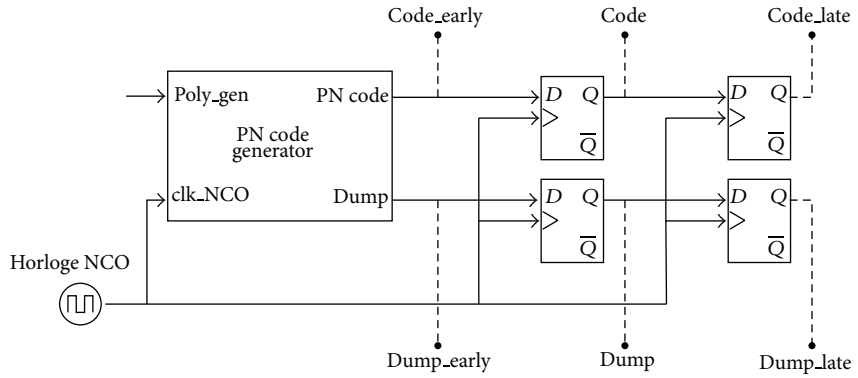

Figure 11: Structure of the receiver PN-Code generator.

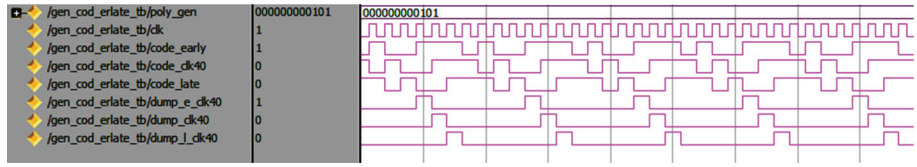

Figure 12: Simulation results of the PN-Code generator.

property, the correlation between the received signal and the local PN code allows the system synchronization. To achieve this condition, specifically we can use the maximal length sequence or the m-sequence because they have good auto-correlation properties and have been used in many applications including the IS-95 standard and UMTS... Also m-sequence can be easily generated by using a shift register therefore, they could be considered as a CDMA code set that can be generated with least hardware complexity for an implementation [7].

This generator is based on the same structure as that of the PN code generator already studied [8]. For synchronization with the codes in emission, the generator which is controlled by the numerical oscillator, output clock must provide the early and late PN codes for the branches of the discriminator (Figure 11).

The simulation results of the synchronized PN code generator are presented in Figure 12. We can see a late replica and an early replica of the local code based on shifting the prompt replica; this shift is controlled by the clock of the NCO via the command word at the input.

3. Conclusion

The work presented in this paper is a VHDL implementation of the synchronization stage of communication systems using the direct sequence spread spectrum. We proposed, developed, and optimized a prototype based on the VHDL language, allowing to describe the digital stages of the synchronization system. The description focused on acquisition and tracking organs. We choose to implement a serial acquisition and a delay-locked tracking loop. By various simulations, we validated first these structures, and the various parameters taken into account are also proposed (NCO parameters, loop filter coefficients). The architecture choice achieved; digital stages will be synthesized for an FPGA implementation.

This model can be considered as an IP and can be reused easily by many teams and different systems. This could lead to a reduction of the gap between a linear productivity and an exponential growth of circuit complexity which is of a great interest in the time to market concept.

References

[1] S. Bourdel, E. Campo, P. Melet, and L. Andrieux ", "From modelling of a CDMA transceiver in indoor environment to an ASIC circuit synthesis," *Journal of Telecommunications and Information Technology*, pp. 36–45, 2001.

[2] S. Parkvall, E. Ström, and B. Ottersten, "The impact of timing errors on the performance of linear DS-CDMA receivers," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 8, pp. 1660–1668, 1996.

[3] F. Lu, H. Samueli, J. Yuan, and C. Svensson, "700-MHz 24-b pipelined accumulator in 1.2-μm CMOS for application as a numerically controlled oscillator," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 8, pp. 878–886, 1993.

[4] H. Meyr, M. Moeneclaey, and S. A. Fechtel, *Digital Communication Receivers, Synchronization, Channel Estimation, and Signal Processing*, John Wiley & Sons, New York, NY, USA, 1998.

[5] M. K. Simon, "Noncoherent pseudonoise code tracking performance of spread spectrum receiver," *IEEE Transactions on Communications*, vol. 25, no. 3, pp. 327–345, 1977.

[6] S. Bourdel, "Modélisation et Simulations fonctionnelles d'un émetteur et d'unrécepteur radiofréquence à étalement de spectre pour une application domotique," INSA, Toulouse, France, 2000.

[7] E. Normark, L. Yang, C. Wakayama, P. V. Nikitin, and R. C. J. Shi, "VHDL-AMS modeling and simulation of /4 DQPSK

transceiver system," in *Proceedings of IEEE Behavioral Modeling and Simulation Conference (BMAS '04)*, pp. 119–124, 2004.

[8] A. Alami, M. Zouak, F. Abdi, and M. Mrabti, "VHDL-AMS modeling and simulation of adirect sequence spread spectrum (DS-SS)transmitter," *Méditerranéen Télécommunications Journal*, no. 2, 2011.