Design and Hardware Implementation of Digital Amplitude Modulation on FPGA

J.A. Galaviz-Aguilar[1], J.R. Cárdenas-Váldez[1], J.A. Reynoso-Hernández[2] and J.C. Núñez-Pérez[1]

[1] National Polytechnic Institute, Research and Development Center of Digital Technology (CITEDI-IPN).
Av. Instituto Politécnico Nacional 1310, Mesa de Otay, C.P. 22150, Tijuana, B. C., México.
[2]Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE),
Carretera Ensenada-Tijuana No. 3918, Zona Playitas, C.P. 22860, Ensenada, B. C., México.
{jgalaviz,jcardenas,nunez}@citedi.mx, apolinar@cicese.mx
Paper received on 12/11/13, Accepted on 01/19/14.

Abstract. This paper presents a novel design model of the basic principle of digital amplitude modulation implemented over a DSP-FPGA board. The design requirements are based in a sequential Top-Level methodology using VHDL. In the signal generation is used a Direct Digital Synthesis approach to control the accuracy of the carrier and modulated signal frequencies. The results are presented with simulations in Matlab and using a testbench in Modelsim to functional design verification. The experimental tests show the output modulated waveforms in order to evidence the correct implementation of the design.

Keywords: Digital Amplitude Modulation, Direct Digital Synthesis, FPGA, Look up table, Matlab/Simulink, VHDL.

1 Introduction

The modern mobile communication systems are based in digital schemes of modulation. Amplitude Modulation (AM) principle is the process where the information is carried via a fast frequency signal in the high frequency (HF) band. In AM the modulation signal controls the carrier amplitude, causing a linear change, but maintains the carrier frequency. Digital Amplitude Modulation (DAM) is a method commonly used in radio communication and presents advantages of accuracy and control of the signal compared with analog AM [1-2]. In this work the advantages of Field Programmable Gate Array (FPGA) based on flexibility for developing of digital hardware implementation is exploited, allowing a rapidly prototyping of the overall circuit.

Accordingly, in this work the DAM design is based in a direct digital synthesis (DDS) technique; which permits a digital controllability in the carrier frequency and modulating waves. This paper is organized as follows: Section 2 describes the whole

Mireya García-Vázquez, Grigori Sidorov, Sunil Kumar (Eds.).
Advances in computing science, control and communications.
Research in Computing Science 69, 2014, pp. 227- 234

model divided into designs blocks explaining the design and operation of the signal generator and digital amplitude theory. In Section 3 hardware simulation and verification of design are presented. In the Section 4 the testing results are depicted. Finally in Section 5 the conclusions are presented.

2 Model design and simulation

The implementation design is described in a high level model in the Matlab-Simulink environment. Furthermore, it is described with blocks that all belong to the DSP Builder Blockset Library. The particularity of this library is that Hardware Description Language (HDL) can be directly generated from the model description block and included in a Quartus II project. The project contains all the source code using VHDL Very High Speed Integrate Circuit Hardware Description Language (VHDL) and generated by the Signal Compiler tool. For this purpose is necessary to make some tasks in Quartus II software in order to fit the design project, pins assignments and timing constraints. Fig. 1 shows the overview model of the entire implementation using DSP Builder blocks.

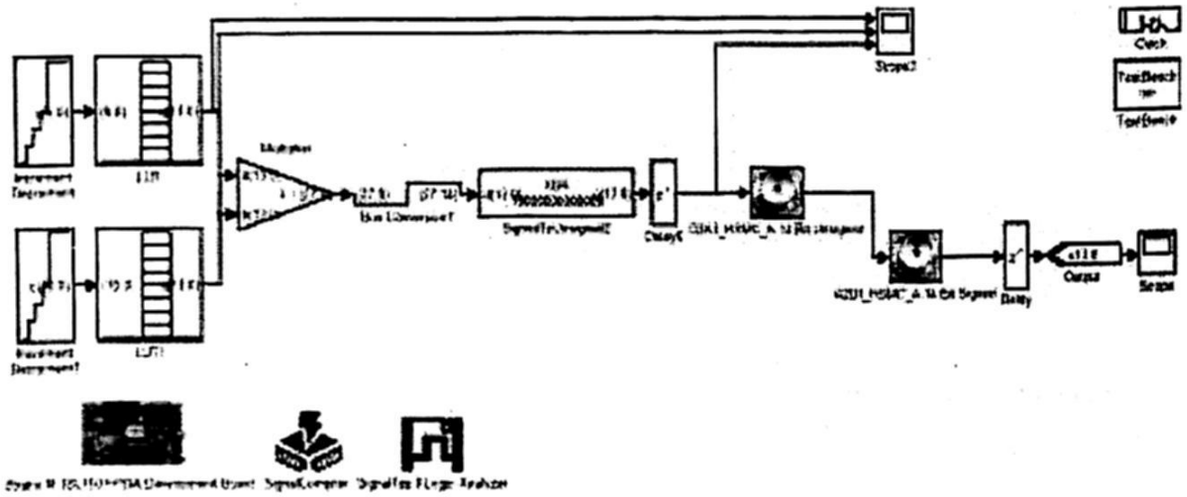


Fig. 1. Overall Design Model in Simulink.

2.1 Signal Generator using DDS Approach

The signal generator block is based on DDS methodology, which consists in generating a periodic and discrete-time waveform of known frequency F_o[3]. DDS is a total digital frequency synthesizer, this includes an increment accumulator, wave store using a look-up table (LUT) and digital to analog conversion (D/A). This principle is shown in Fig. 2, where the phase locked-loop (PLL) component represents the sampling clock time in the signal generator and the overall design logic into FPGA.

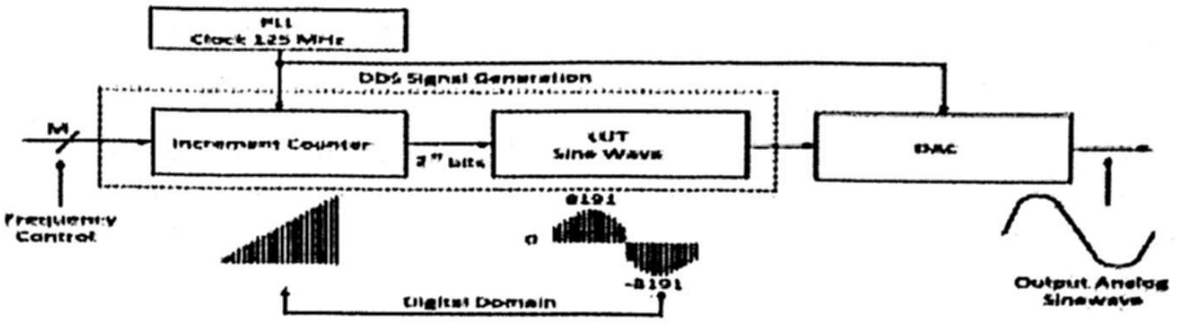


Fig. 2. DDS principle for digital signal generator.

The DDS signal generation shown in Fig. 2 is based on an increment block of 2^n-bits, which are addressed with a sine LUT. The LUT size has to correspond to the counter resolution. The output steps counter are addressed to the LUT, where is stored the sample data computed of sine function. The counter output value augments on each rising edge of the clock; the DDS block model is shown in Fig. 3.

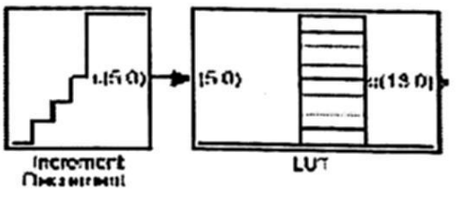


Fig. 3. DDS component blocks.

There are three steps to obtain a proper signal generation output. First, the number of samples in a signal period has to fit the size of the LUT [4]. Secondly, the LUT resolution is defined by the number of bits related to the counter depth. Finally, the maximum frequency of the signal has to be theoretically 2 times considering sampling Nyquist's criterion, but practically 10 times lower than the frequency of the internal clock established in 125 MHz. The following expression determinates the output sinewave frequency in the DDS component.

$$F_{out} = (M/2^n) f_{Clock} \tag{1}$$

Where M represents a digital control to determinate the output frequency [5-6]. The values of the LUT are coded with a word of n bits signed to compromise between precision and the bus size conversion at the input of a multiplier. It is necessary a Matlab array with a smaller length than 2^(addres width), which represents one cycle of a calculated sinewave data stored in the LUT, consider the

$$8191 * sin([0: 2 * pi/(2^5): (2 * pi)]) \tag{2}$$

2.2 Digital Amplitude Modulation Principle

The aim of this work is a fully DAM implementation method based in a digital control amplitude values of two LUT sinewaves. For this purpose a carrier discrete signal and load the signal modulation are considered [7], this process is realized using

a multiplier; which the product represents the amplitude modulate signal, in Fig. 4 are shown the input data of 14-bit that represents the frequencies to multiply. In the output is necessary a truncation of bits in order to send only the 14-bit most significant bit according with the DAC resolution.

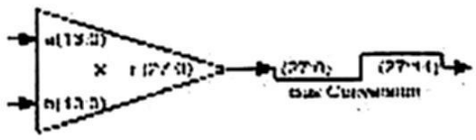
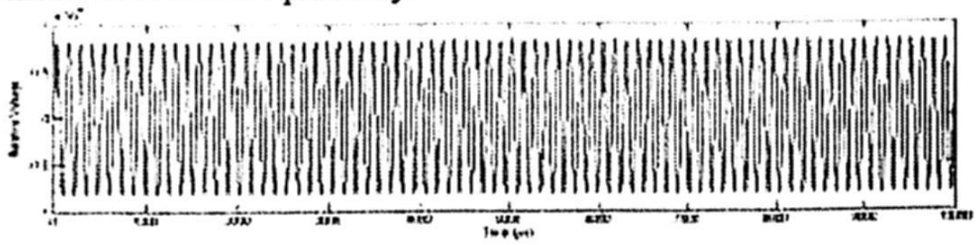


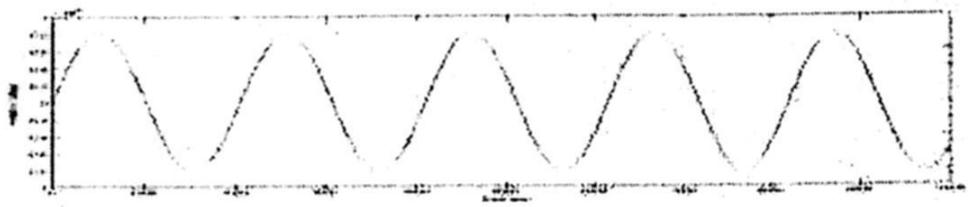
Fig. 4. Product multiplication block with 14-bit truncation.

2.3 Simulation of the Model in Simulink

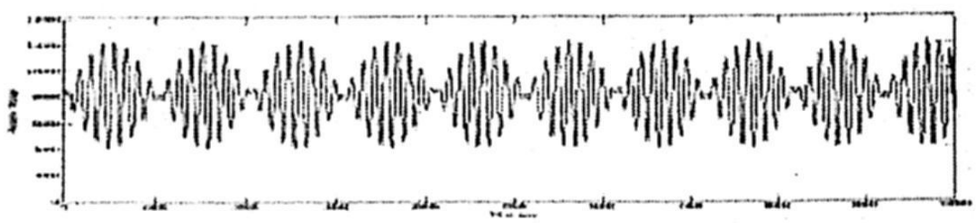
The simulation results in Simulink are able to verifications in generated signals into the DDS blocks; which permits improving the functionality before synthesis stage of the model into a VHDL. The simulation shown in Fig. 5 system verification to checks the carrier frequency and the modulated signals generated by the DDS component and the output modulated wave respectively.



(a)



(b)



(c)

Fig. 5. Simulation results into Matlab (a) RF carrier (b) Modulating wave (c) Modulated result.

2.4 Hardware Simulation and Verification of Design

After simulation, the model is verified using a test bench on Modelsim which allows debugging tasks of the design and checking the performance in FPGA device previous the synthesis stage and program. Fig. 6 represents the output analog modulated waveform in time using the sampling clock of 125 MHz, also can check the reset signal into the system.

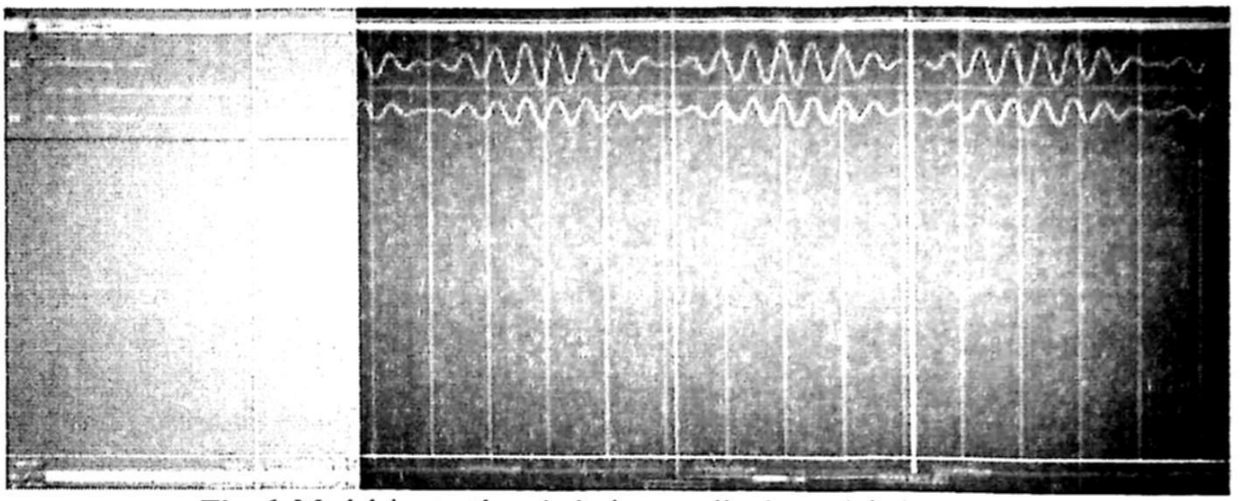


Fig. 6. Modelsim test bench design amplitude modulation output.

The design synchronization is based in a unique clock of 125 MHz which is shown in Fig. 7. According with this clock the internal data sampling is accomplished each cycle clock (rising edge) generating the sample data, the reference clock cycle has a period of 8 ns. The signal of the reset internal is running in synchronization with the clock.

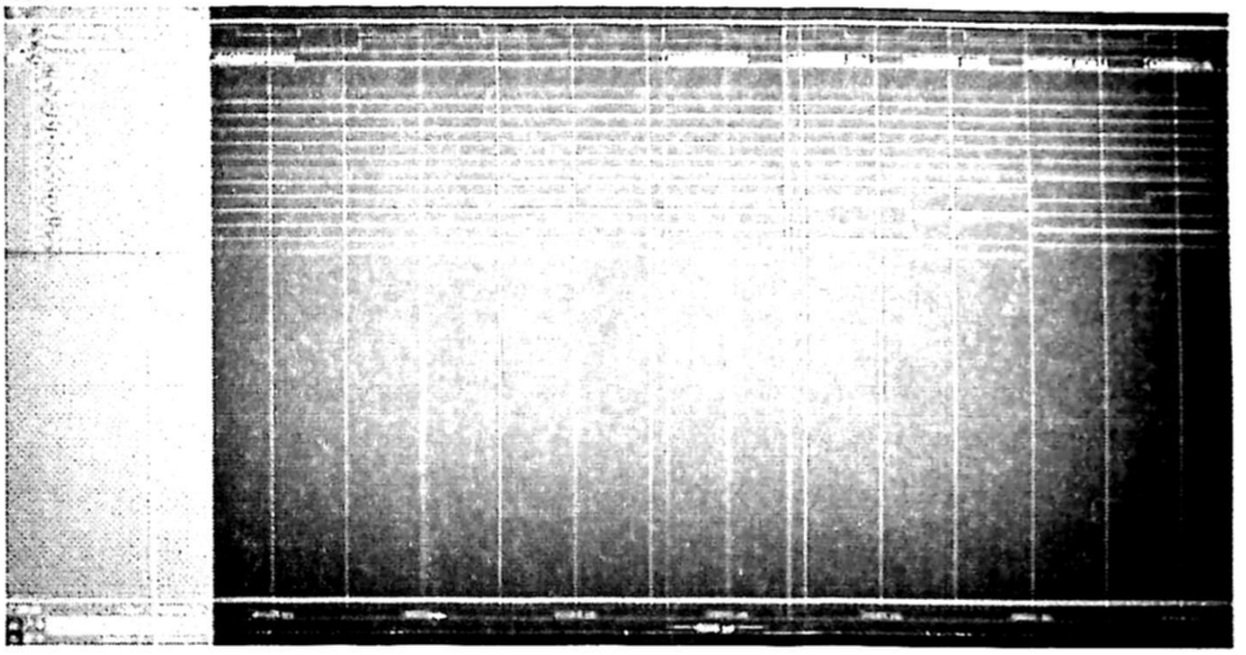


Fig. 7. Modelsim test bench design simulation.

3 Implementation design

For this design all the internal circuitry is controlled using a synchronous clock of 125 MHz, with an active low reset implemented through a PLL circuit; this clock also acts as the reference DAC sampling clock DAC. The design source code generated by the signal compiler tool which is processed, compiled and analyzed using the Quartus II tools several times to debugging and synthesize the overall circuit, before the implementation of the design in the FPGA device.

To check the VHDL of the design is necessary to open in a new project in order to complement and debug the code. Indeed, the DSP Builder does not allow the I/O pin

assignments, it has to be made manually using the pin planner tool of Quartus II Software. In addition, the interface between the FPGA and the DAC has to be created. The implementation structure of amplitude modulating design is integrated into the architecture of DSP Stratix III 3SL150 Development Board of Altera and a data acquisition card.

3.1 Conversion to Data Acquisition Card

The last step of the implementation consists in a conversion process that aims acquiring the data with the DAC. An overview of the conversion process can be seen in the Fig 8. For the design are used the channel A of the DAC where is necessary converts the representation signed to an unsigned type of 14-bit to send an analogic signal observable with an oscilloscope. The signal directly obtained at the output of bus type with the conversion XOR is used for the interpretation.

However, the real signal that would be used in case of including the implementation in a test bed is the one provided by the DAC.

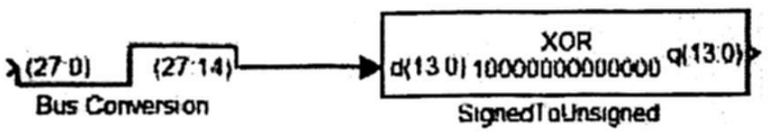


Fig. 8. Data type conversion chain.

3.2 Overall FPGA Resources Utilization

Total resources consumption by the implementation of model is shown in chip floor pan in the Fig. 9 where the significant usage is described by the utilization of embedded adaptive logic modules. In the design, implementing ROM resources of the FPGA for store the LUT sinewave sample data is needed, where is implemented over embedded memory block resources of 144 Kbits on chip.

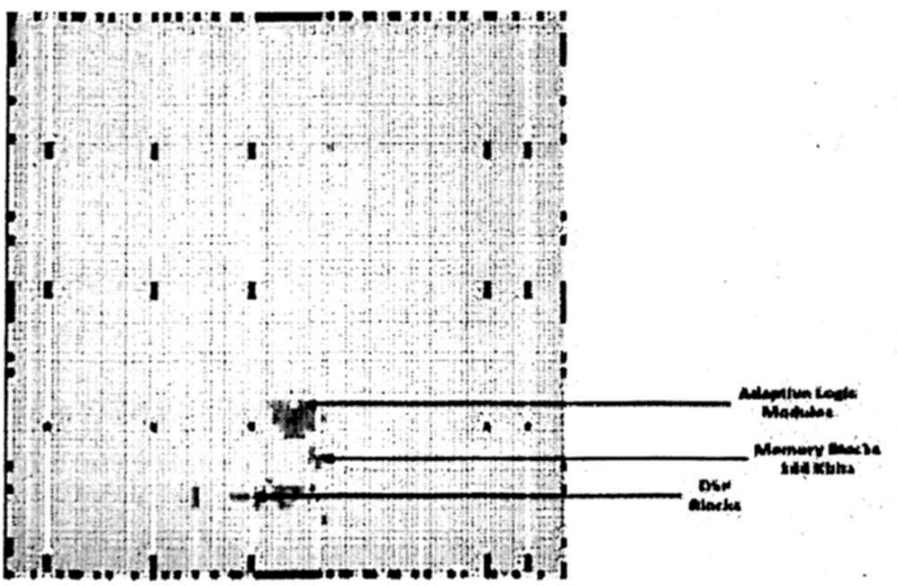


Fig. 9. Chip planner with overall resource usage.

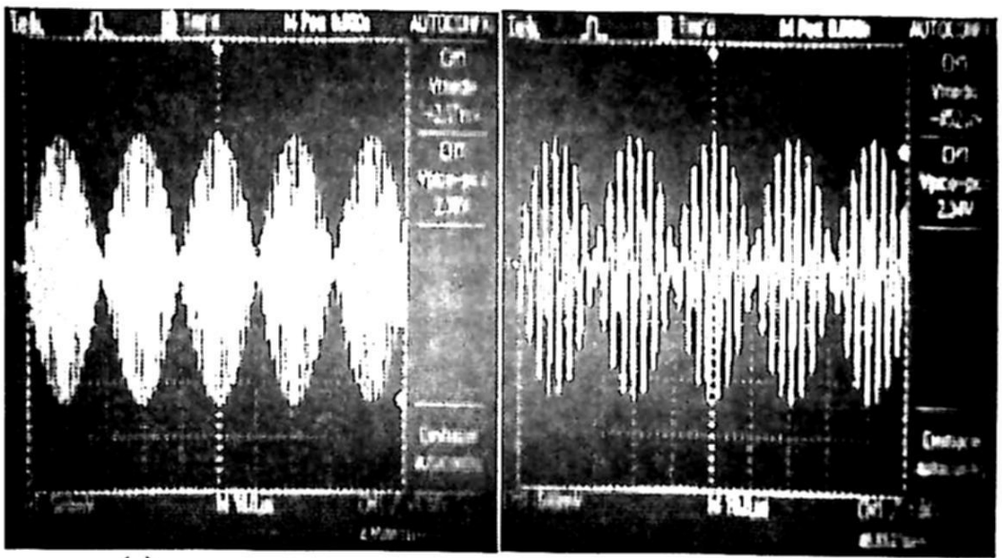
The overall resources used in the FPGA for the digital amplitude modulation design are summarized in Table 1; the consumed resources necessary by the whole design implemented are denoted in detail.

Table 1. Overall resources consumed in FPGA.

FPGA Resources	DSP Blocks (Multipliers)	Dedicated Logic Registers	Memory bits	Adaptive Logic Modules
Total used	2	738	43,008	330
FPGA Available	384	113,600	5,630,976	113,600
Percent used	<1 %	<1 %	<1 %	<1 %

Testing Results

The testing results are depicted in Fig. 10 generating two different modulated waveforms. The frequency of carrier shown in Fig. 10 (a) is established in 4 MHz and the modulated waveform in 125 KHz and (b) carrier is 2 MHz and 125 KHz modulated waveform.



(a) (b)

Fig. 10. Modulated transient waves in oscilloscope.

4 Conclusions

The design model presented in this paper basically demonstrates a short time frame implementation of a digital Amplitude Modulation system into the FPGA. Through the experiment of an implementation into a Stratix III FPGA device was presented hardware architecture to a digital amplitude modulation model. Given the complexity and time invest to design in VHDL; DSP Builder provides an efficient tool to design practical circuits applications based in digital amplitude modulation.

A general design method for an amplitude modulation model applicable in any other work has been described. Finally a precise description and justification of the components used in the hardware implementation had been given. Also in order to validate the improved performance of the design hardware simulations using a test bench in Modelsim were presented. In addition to that, the feasibility of the design theory was also compared with the design proposed method, based on DDS to provide accuracy in the output waveforms.

References

1. D. Xie, S. Tian and K. Liu "Design and Implementation of DDS based Digital Amplitude Modulation," in IEEE Circuits and Systems International Conference on Testing and Diagnosis, ICTD 2009, pp. 1-4, July 2009.
2. J. Gao, N. Liu, and X. Xu "Design of AM modulation signal generator based on Matlab/DSP Builder," in 2nd International Conference on Industrial and Information Systems (IIS). Vol. 1, pp. 527 - 530, 2010.
3. L. Cordesses, "Direct digital synthesis: a tool for periodic wave generation (part 1)," in IEEE Signal Processing Magazine, Vol. 21, No, 4, pp. 50 - 54, July 2004.
4. M. Butt and S. Masud. "FPGA based bandwidth adjustable all digital direct frequency synthesizer," in 9th International Symposium on Communications and Information Technology, ISCIT 2009. pp. 1399-1404, 2009.
5. G. Xiong, X. Zhou, and P. Ji "Implementation of the Quadrature Waveform Generator Based on DSP Builder," in International Symposium on Intelligent Information Technology Application Workshops, 2008. IITAW '08. pp. 773-776, 2008.
6. Z. Chail, and J. Shen "The Application of System Generator in Digital Quadrature Direct Up-Conversion," in Communications in Information Science and Management Engineering (CISME). Vol. 3, No, 4, pp. 192-197, ISSN: 2222-1859, 2013.
7. C. Erdoğan, I. Myderrizi, and S. Minaei "FPGA Implementation of BASK-BFSK-BPSK Digital Modulators," in IEEE Antennas and Propagation Magazine, Vol. 54, No. 2, pp. 262 -269, April 2012.