

Design and Implementation of Programmable Transceivers Based Software Defined Radio Using FPGA

Saleim Hachem Farhan

Abstract: In this paper, an efficient programmable transceivers based software defined radio (SDR) technology is proposed. In the most wireless and mobile system and even digital communication, the modem scheme and pulse shaping filter play vital role to optimize the information size and security. Therefore, the digital modem is used to provide these issues in SDR technology. To implement this type of modem in FPGA device, many software should be used like MATLAB, system generator, ModelSim, and ISE produced by Xilinx company to generate the HDL code and bit stream. The HDL code of activation system (ADC/DAC/FPGA Synthesizer) and Netlist design has been integrated together to become the HDL code of integrated design. The implementation results shows that the real time results is quite close to the simulation results due to accurate adjustments of design process and timing constraint were low FPGA area in term of Slices and LUTs is produced.

Index Terms: Programmable Transceiver, SDR, FPGA

I. INTRODUCTION

Software radio or software defined radio (SDR) becomes more popular due to its highly configurable hardware and software platforms, as contrast to the complex hardware platform [1]. Intermediate frequency (IF) and baseband signal processing functions could be performed by SDR, Digital Signal Processing (DSP) and algorithms [2]. In SDR implementation the most silicon keys are field programmable gate arrays (FPGAs), digital signal processors (DSPs) and application-specific integrated circuits (ASICs). For intermediate frequency (IF) stage and wideband modem processing, the better approach could be found in FPGA since it offers high speed, high level of integration, high flexibility, and low development cost; while it might contain high power consumption because the incompetent use of the FPGA logic elements [3]. The modulation and demodulation schemes in SDR technology is uncomplicated one-dimensional modulation scheme that the phase of carrier sinusoidal signal changes abruptly by 1800 or π radian for each transition of modulating binary progression [4]. More than a few investigations on digital design of SDR-based PSK have been completed as in [5-8], however there still stay some dangerous issues of interfacing analog-to-digital converter (ADC) and digital-to-analog converter (DAC) with FPGA. In addition to the conventional design, the Hardware Description Language (HDL) module configurations of expansion P240 Analog module and clock

synthesizer that controls FPGA system clock frequency is presented in this paper. Through using SDR technology the family of radio harvest used in wireless communication to be building in most platform design and many researches going on in this field permit to new generation to come faster into market. Because the SDR technology allows to be reused through radio products, it will be more advanced in near future and the design cost will be reduced considerably. This virgin of wireless communication permits the users to communicate with whomever they need to communicate and in whatever way according to their aspiration. As an example for the SDR application, in the video call currently used in third generation. Similarly any type of application could be include using SDR in a single chip. The SDR engineers will offers a single radio transceiver capable of playing the vital roles of cordless phone, cell phone like GSM and WCDMA. Global position system and other systems is still in the realm of science fiction, problems from any location on the surfaces of earth and maybe in space as well. The universal structure of reconfigurable computing processor in SDR is illustrated in Figure 1. The SDR is a software radio in which all the physical layers are software defined and effectively uses the area in FPGA [9-14].

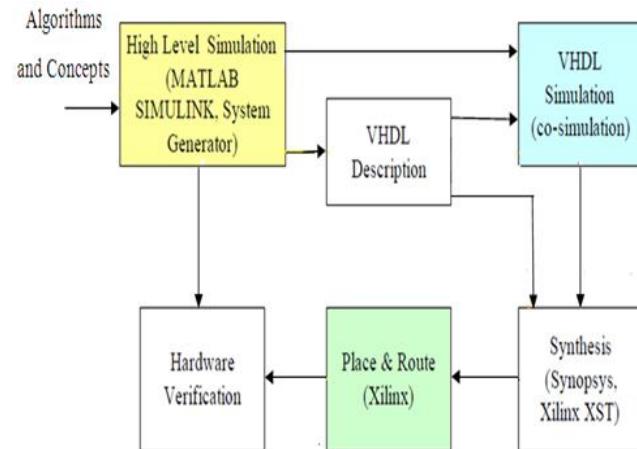


Figure 1: Universal structure of reconfigurable computing processor in SDR technology

II. TRANSCEIVER SIMULATION

A. Matlab/Simulink Design.

The design of SDR transceiver with QAM modulator and demodulator is shown in Figure 2. The output waveform of the transmit/receive path is shown in Figure 3. The SIMULINK blocks set in MATLAB is used to simulate the proposed SDR transceiver with floating point values of each parameter.

Revised Manuscript Received on 30 January 2014.

* Correspondence Author

Saleim Hachem Farhan*, Electronic Technique Dept. Institute of Technology, Baghdad, Iraq.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](#) article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>

The eye and scatter plot of transmit and receive signals is used to evaluate and analysis the performance of the model

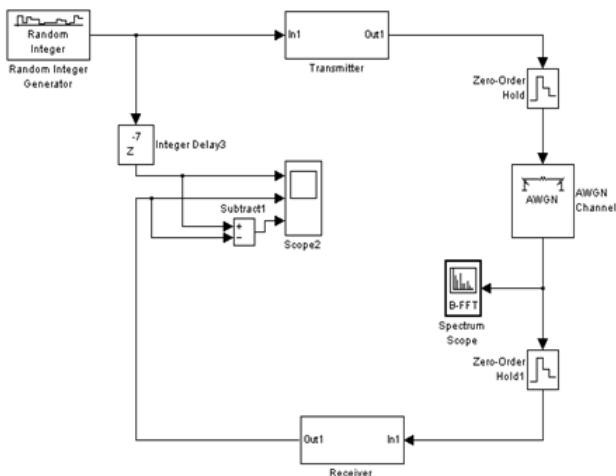


Figure 2. SDR Transceiver model

From the scope 2, one can see that original message input is recovered by the receiver without different due to accurate processing and timing adjustment. Therefore, the different between the transmit message and recovered bit integer is zero as clearly illustrated in figure 3.

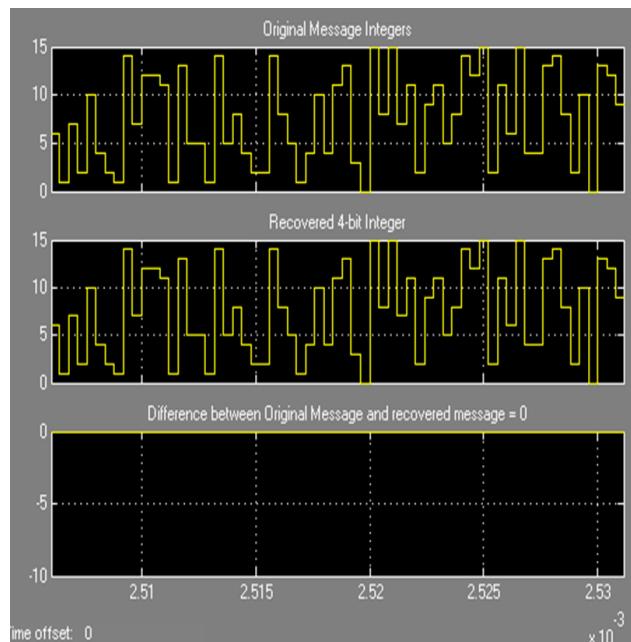


Figure 3: scope 2 waveform

B. Design and Simulation Using DSP Design Tools

Based on the available resource [15-19], the design of QAM modulation and demodulation is developed in DSP Design Tools. The model has been designed in Xilinx System Generator with fixed point values for each parameter. Then the fixed point values are verified with Matlab/Simulink design in floating point values to show the accuracy and flitting of each value. The Graphical User Interface (GUI) of the design is model-based structure using Simulink and Xilinx specific block sets as shown in Figure 4 and Figure 5 for transmitter and receiver respectively.

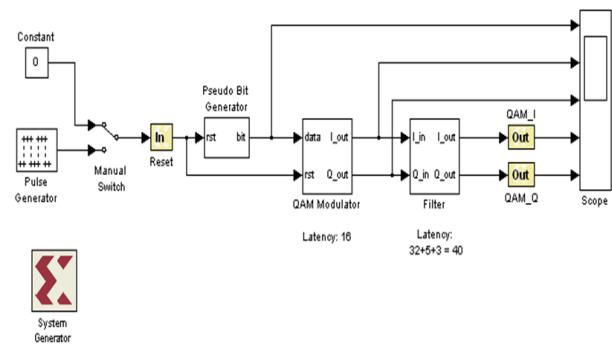


Figure 4: transmitter design using System Generator

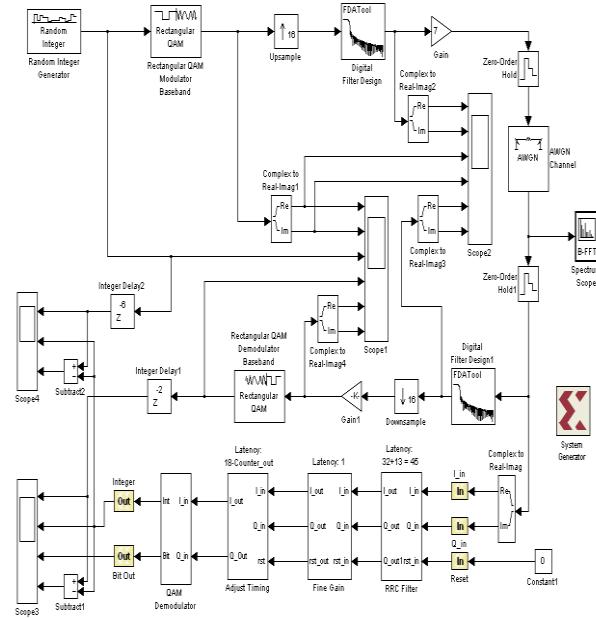
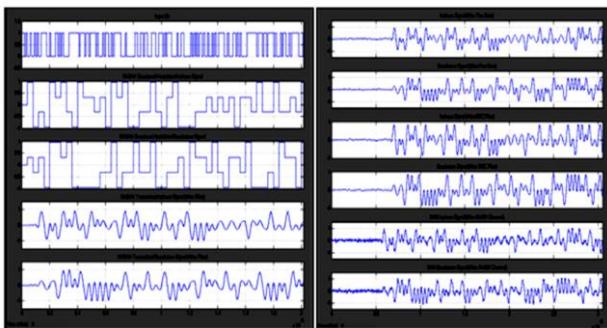


Figure 5: receiver design using system generator

The System Generator Blocks token is used to set Simulink system period, and FPGA clock period: 10 ns (100 Msps sampling rate). It is used to generate Verilog HDL netlist of QAM transmitter with its test-bench file. Reset Block is used as Xilinx Gateway In port that acts as input port of FPGA. It samples incoming signal from Constant block at 1 unit sample time, and yield Boolean output to input reset of the PR Bit Generator, Polar Conversion, and Sin Generator subsystems. The PR Bit Generator Subsystem is used as pseudo-random (PR) concept to generate random input bit source to QAM transmitter system. The QAM Signal Block is used as Xilinx Gateway Out port that acts as output port of FPGA. It also translates the incoming fixed point signal (from Multiplier) into double precision value that will be displayed in Scope block. Simulink sink block is used to displays the simulation results of transmitter and receiver output waveforms as shown in Figure 6. Hence, the verification between the floating point design in MATLAB and the fixed point design in system generator is achieved without mistaken due to zeroing different between two designs. Now the model is ready to implement in FPGA following many steps in design flow as shown in Figure 7.



(a) Transmitter waveform (b) Receiver waveform
Figure 6: Transmit receive waveforms of QAM modem

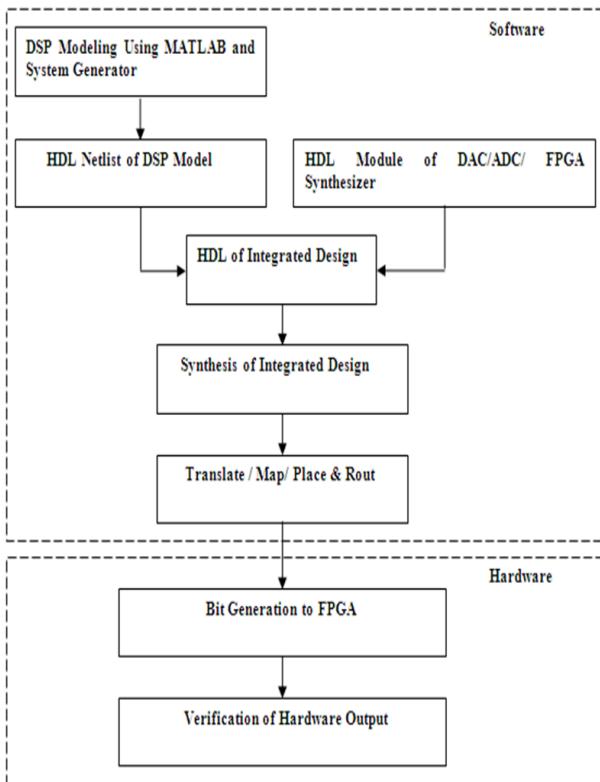


Figure 7: FPGA design flow for Transceivers

C. HDL Design of ADC,DAC, and FPGA Synthesizer

The Avnet P240 Analog Module has 2 analog input and 2 analog output channels as given in company product user guide. The analog input channel contain 2 sets of THS4509 WB differential amplifier, low-pass 7th order RLC anti-alias filter , and Texas Instrument (TI) ADS5500 (14-bit, 125 MHz) ADC; while the analog output channel consists of one TI DAC5687 (16-bit, 500 MHz, interpolation) dual-channel DAC, 2 low pass 5th order RLC reconstruction filters, and two 50 ohm coupled transformers. The CDCP1803 clock buffer on P240 supplies one quarter (1/4) of source clock to each of two ADS5500 ADCs and 1× source clock to DAC5687 CLK2 . The configurations of ADC and DAC are set through Serial Programming Interface (SPI). The HDL simulation result of setup configuration (ADC/DAC/FPGA synthesizer) is illustrated in Figure 8 and the HDL simulation result of QAM is illustrated in Figure 9.

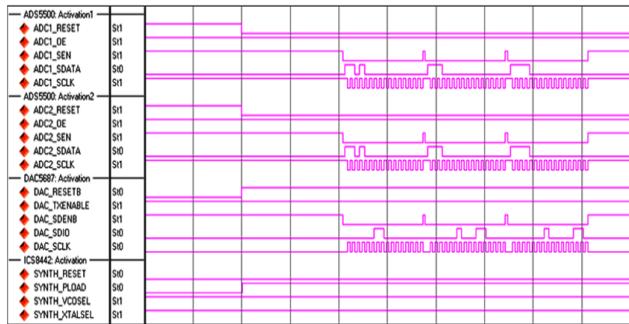


Figure 8 HDL simulation result of setup configuration

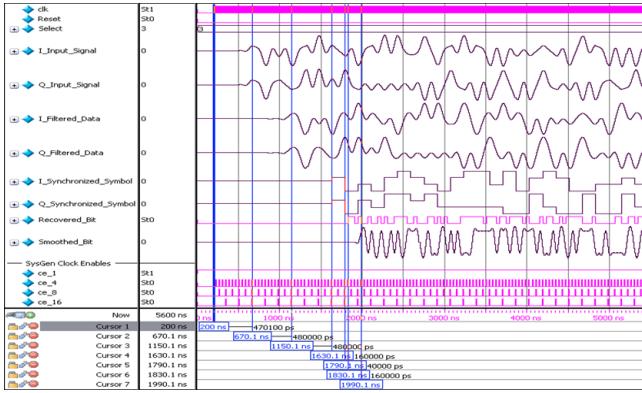


Figure 9: HDL Simulation for QAM

D. HDL of Integrated Design

The Verilog HDL module of setup configuration and Verilog HDL netlist of QAM transceiver are verified firstly before combining both to become HDL module of integrated design . The simulation results in ModelSim environment are shown in Figure 9 . The HDL simulation result of QAM Netlist is show in Figure 10. Considering the real-time implementation of integrated design of QAM and setup configuration using FPGA and P240 Analog Module , the ADC and DAC in P240 should be configured first prior to the running of DSP design , in order to avoid instability of ADC and DAC that can produce undesired outputs to or from FPGA during process of configuring ADC and DAC. Thus, the clock enable (ce) of FPGA design is disabled during the transfer of SPI codes to ADC and DAC in P240.



Figure 10: HDL Simulation of Integrated Module for QAM

Controlling the main *ce* would be easier rather than clock enable clear (*ce_clr*) which requires additional logics to adjust sampling phase of all the multi sample data when it is de-asserted.

E. Synthesis of Integrated Design

Although Xilinx ISE suite has its own synthesis tool Xilinx Synthesis Technology, but it can only synthesize HDL netlist generated from System Generator. Therefore, Synplify Pro software is used to perform logic synthesis for the HDL module of integrated design in 2 stages of logic compilation and optimization, and technology mapping. Before doing the final stage of synthesis, FPGA pins (pad locations) have to be assigned accordingly to user guide of. Notice that the final synthesis for the integrated design would ignore the pin assignment of ADC_IN and DAC_DB since no analog input is involved in QAM transmitter, and only one DAC analog output is used for QAM transmitted signal.

Timing characteristics is an important issue that will affect the performance of FPGA implementation. The required path delay (estimated period) for Xilinx FPGA element should be less than the requested (constrained) clock period. So, timing slack (requested period – estimated period) should be positive value; otherwise the integrated design has to be reworked. The clock frequencies used for the ADC/DAC SPI process and transmitter i.e. CLK_100 and LIO_CLKIN_1 (or ADC_CLKOUT) are set to 100 MHz for both. The positive slack values in timing report of design synthesis in Table 1 meet the timing requirement.

Table 1: Estimated Timing Report for QAM Modem

Constraint	Check	Worst Case	Best Case	Timing
		Slack	Achievable	Errors
TS_CLK_100 = PERIOD TIMEGRP "CLK_100" 10 ns HIGH 50%	SETUP HOLD	3.412ns 0.542ns	4.588ns	0 0
TS_LIO_CLKIN_1 = PERIOD TIMEGRP "LIO_CLKI" 12.5 ns HIGH 50%	SETUP HOLD	4.607ns 0.257ns	5.893ns	0 0

III. IMPLEMENTATION OF FPGA

A. Software Achievement.

The synthesis output files required by Xilinx ISE software are in EDIF (Electronic Design Interface File) and UCF (User Constraints File) formats, which represent the optimally synthesized netlist of integrated design, and timing constraints and FPGA pin assignment respectively. To implement the synthesized design into Virtex-4 FPGA development board, Xilinx ISE performs the following steps as illustrated in Figure 11.

Convert netlist of integrated design in EDIF format to NGD (Native Generic Database) file that contains logical description of hierarchical components and Xilinx primitives by using NGDBuild program. The logical DRC (Design Rule Check) on the NGD file, and then map the design logic to slices and I/O cells in Virtex-4 FPGA to create Native Circuit Description (NCD) file. The area constraints can be sized properly using PACE (Pin-out Area Constraints Editor); then re-run Map. Place and route the design in mapped NCD file into Virtex-4 FPGA based on timing constraints using Timing Analysis tools. The output is fully routed NCD file. Translate the fully routed NCD file to configuration bit-stream (.BIT) file using BitGen program. Finally,

download the bit-stream file into Virtex-4 FPGA via JTAG cable using iMPACT program.

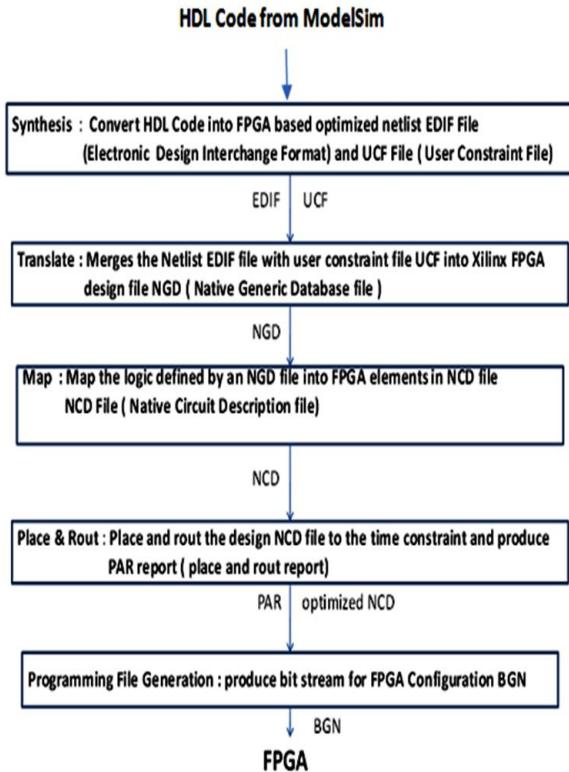


Figure 11: FPGA Implementation steps

The device utilization summary for Virtex-4 FPGA is shown in Figure 12 and Figure 13. As shown in the post PAR static timing report in Table 2, the positive worst case slacks (constrained period – best case required period) fulfill the timing requirement. The maximum allowable sampling rate for QAM modem used in Virtex-4 FPGA is 206.6543 MHz ($= 1/4.839$ ns). However, the maximum sampling rate of LIO_CLKIN_1 from ADS5500 is 125 MHz; thus the maximum sampling rate of QAM modem that can be used in Virtex-4 FPGA MB development board with P240 Analog Module is 125 MHz.

Table 2: Estimated Timing Report for QAM Modem

Constraint	Check	Worst Case	Best Case	Timing
		Slack	Achievable	Errors
TS_LIO_CLKIN_1 = PERIOD TIMEGRP "LIO_CLKI" 12.5 ns HIGH 50%	SETUP HOLD	0.465ns 0.358ns	10.035ns	0 0
TS_CLK_100 = PERIOD TIMEGRP "CLK_100" 10 ns HIGH 50%	SETUP HOLD	4.021ns 0.512ns	3.979ns	0 0



Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	392	30,720	1%	
Number of 4 input LUTs	280	30,720	0.9%	
Logic Distribution				
Number of occupied Slices	370	15,360	2%	
Number of Slices containing only related logic	300	353	84%	
Number of Slices containing unrelated logic	22	353	6%	
Total Number of 4 input LUTs	450	30,720	1%	
Number used as logic	302			
Number used as a route-thru	9			
Number used as Shift registers	145			
Number of bonded IOBs	63	448	14%	
Number of BUFG/BUFGCTRLs	6	32	2%	
Number used as BUFGs	6			
Number used as BUFGCTRLs	0			
Number of FIFO16/RAM16s	8	192	4%	
Number used as FIFO16s	0			
Number used as RAM16s	8			
Number of DSP48s	14	192	7%	
Total equivalent gate count for design	549,154			
Additional JTAG gate count for IOBs	2,892			

Figure 12: Device utilization for Virtex-4 FPGA Transmitter

Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	10,420	30,720	33%	
Number of 4 input LUTs	6,890	30,720	22%	
Logic Distribution				
Number of occupied Slices	6,290	15,360	40%	
Number of Slices containing only related logic	6,185%	6,187	100%	
Number of Slices containing unrelated logic	0	6,187	0%	
Total Number of 4 input LUTs	7,930	30,720	25%	
Number used as logic	7,900			
Number used as a route-thru	189			
Number used for Dual Port RAMs	31			
Number used as Shift registers	166			
Number of bonded IOBs	87	448	19%	
Number of BUFG/BUFGCTRLs	6	32	18%	
Number used as BUFGs	6			
Number used as BUFGCTRLs	0			
Number of DSP48s	6	192	3%	
Number of RPM macros	1			
Total equivalent gate count for design	189,236			
Additional JTAG gate count for IOBs	4,290			

Figure 13: Device utilization for Virtex-4 FPGA Receiver

Figure 12 and Figure 13 lists the total number of slices and look-up tables (LUTs) used in this design. In each slice they are two LUTs and two FFs, during the PAR, the ISE software put all necessary LUTs close to each other for minimum propagation of data, for that some LUT inside slice not used and in some slice only FFs is used without LUT so, the number of LUTs cannot be calculated manually, therefore the LUTs could be less or more than Slices depend on software

optimization. The device utilization summary generated by ISE software represents the available logic elements in FPGA and logic elements used by the in hand project been designed. The percentage of used logic elements to the available logic elements is calculated as follow:

$$\frac{\text{used logic elements}}{\text{available logic elements}} \quad (1)$$

For example:

$$\text{Utilized number of Slice Flip Flop} = (392/30720) \times 100 = 1\% \quad (2)$$

$$\text{Utilized number 4-input LUTs} = (280/30720) \times 100 = 0.9\% \quad (3)$$

$$\text{Utilized number of occupied Slices} = (370/15360) \times 100 = 2\% \quad (4)$$

$$\text{Utilized number of bonded IOBs} = (300/353) \times 100 = 84\% \quad (5)$$

B. Hardware achievement Result .

The DAC Channel A output from P240 Analog Module is connected to oscilloscope in order to show real-time result in analog domain. By observing the real-time result of QAM transmitted and received waveforms as shown in Figure 14 and Figure 15 respectively. The measured time difference between 2 adjacent positive peaks (Δt) is 200 ns. Thus, the exact carrier frequency of real-time result is 5 MHz ($= 1/200$ ns) which is equal to the estimated values. The minor difference of amplitudes ($\Delta V = 0.04V$) for the first peak (1 V) and second peak (0.96 V) is due to limitation of the 50-ohm transmission coupled transformer after DAC in P240 Analog Module. Though, contrast of the empirical real-time result and simulated results for BPSK transmitted signal has shown equivalence in time domain

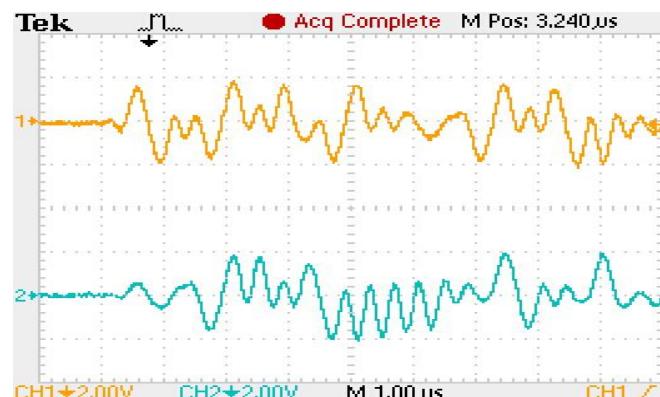


Figure 14:Real-time waveforms of QAM Transmitter output

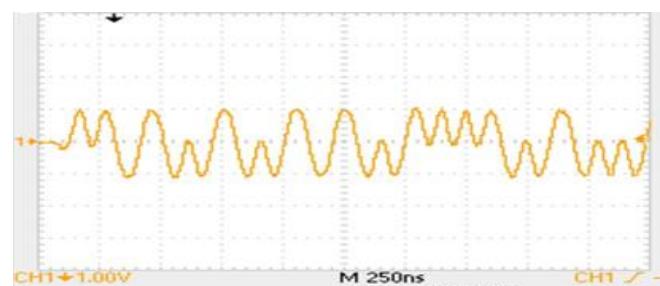


Figure 15:Real-time waveforms of QAM receiver output

IV. CONCLUSIONS

This paper has offered significant information to design and implement the programmable transceivers in Virtex-4 FPGA MB development board with DAC in P240 Analog Module. A lot of software and design tools have been used to verify the design output in terms of behavior, functionality, synthesis, timing, and area constraints. The comparison of empirical real-time and simulated results shows the success of FPGA and DAC implementations that would be further processed by external analog RF devices for complete wireless SDR system. The timing issues such as sample rate, constraints, and matching should be concerned in-depth if the input bit is coming from external source which is different from the presented design. Inserting pulse-shaping filtering between polar conversion and mixer can further reduce inter-symbol interference (ISI) to enhance the receiver performance. This modem could be more tuning to get better results in future with different noise channel like fading and ISI distortion.

REFERENCES

1. Mitola, "The software radio architecture", IEEE Communications Magazine, 33/5, PP.26-38, 1995
2. A. Pal, "FPGA-based (Xilinx) Embedded System Design," Workshop on Microcontroller and FPGA-based Embedded System Design, Department of Computer Science & Engineering, IIT Kharagpur, India, July 2007.
3. The MathWorks, Inc. Digital Signal Processing & Communications.[Online].Available http://www.mathworks.com/applications/dsp_comm 2009
4. Xilinx, Inc., DSP Design Flows in FPGA Tutorial Slides, 2003.
5. J. G. Proakis, Digital Communications, 4th ed., New York: McGraw-Hill, 2001.
6. D. Haessig, J. Hwang, S. Gallagher, and M. Uhm, "Case-Study of a Xilinx System Generator Design Flow for Rapid Development of SDR Waveforms," in Proc. SDR 05 Forum Tech. Conf. and Product Exposition, Orange County, California, 2005.
7. C. Dick, F. Harris, and M. Rice, "Synchronization in Software Radios Carrier and Timing Recovery Using FPGAs," IEEE Symp. Field-Programmable Custom Computing Machine, pp. 195- 204. 2000
8. System Generator for DSP User Guide, Release 9.2.01, Xilinx, Inc., 2007.
9. Xilinx ISE 9.2i Software Manuals: Constraints Guide, and Development System Reference Guide, Xilinx, Inc., 2007.
10. P240 Analog Module User Guide, Rev 1.0, Avnet, Inc.. [Online]. Available: http://www.files.em.avnet.com/files/177/p240_analog-ug.pdf, May 2009
11. Virtex-4 MB Development Board User's Guide, Ver. 3.0, Avnet Memec., [Online].Available: http://www.files.em.avnet.com/files/177/v4mb_user_guide_3_0.pdf, 2008
12. "ADS5500: 14-bit, 125 Msps, Analog-to-Digital Converter Data Sheet," Texas Instrument, Inc., Feb. 2007. [Online] Available: <http://focus.ti.com/lit/ds/symlink/ads5500.pdf>
13. "DAC5687: 16-bit, 500 Msps, 2x-8x, Interpolation Dual-Channel Digital-to-Analog Converter (DAC) Data Sheet," Texas Instrument, Inc., June 2005. [Online]. Available: <http://focus.ti.com/lit/ds/symlink/dac5687.pdf>
14. Synplicity FPGA Synthesis Reference Manual, Synplicity, Inc., 2007.
15. Chan, et al, "Design and Complexity Optimization of a New Digital IF for Software Radio Receivers With Prescribed Output Accuracy" IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS, VOL. 54, NO. 2, pp 351-366, 2007.
16. Durke, et al, "Field Programmable Gate Array Based SDR Design" phd. Thesis, Naval postgraduate school, Department of Electrical and Computer Engineering, California: http, 2009.
17. Giannini, "Baseband Analog Circuits For Software Defined Radio", Springer, www.springer.com, ISBN 978-1-4020-6538-5 (e-book), PP. xi, 2008.
18. Pedro and Nuno, "Multi-Mode Receiver for Software Defined Radio", Institute of Telecommunication – University of Aveiro – Portugal, 2008.
19. Proakis, and Salehi, "Digital Communications", 5th ed., New York, NY: McGraw-Hill, Ch. 3, pp. 95-148, 2008.

AUTHOR PROFILE



Assist. Lecturer Eng. Saleim Hachem Farhan
Received his B.Sc. Academy of Engineering - Sarajevo - Bosnia's Republic in 1982, and received the M.Sc. degree in Electrical Engineering - University of Belgrade in 1984. Currently he is Assist. Lecturer, Researcher and training supervisor, Dep. of Electronic in Institute of Technology Baghdad.