Journal of Engineering and Development, Vol. 18, No.3, May 2014, ISSN 1813- 7822

Optimal Design for Software Defined Radio Based Fpga

Asst. Prof. Dr. Mahmod Farhan
Department of Computer Tech. Eng.
Electrical and Electronic Tech. Coll.
drmahfa@yahoo.com

Lecturer. Dr. Majid S. Naghmash
Department of Elect. Power Eng.
Electrical and Electronic Tech. Coll.
msnengb99@yahoo.com

Lecturer. Dr. Faeza Abbas
Department of Electronic
Institute of Technology
dr.faezaa@yahoo.com

Abstract :

This paper presents, optimal design for the software defined radio (SDR) based Field Programmable Gate Array (FPGA) using modern software programs from Mathworks and Xilinx. The SDR mode have to be used which is extremely in meeting the increasing demands of the wireless communication and mobile industry. Nowadays, all digital communication systems need to easy adopted with more complicated coding and modulation techniques. The shortest and efficient paths to design an FPGA using MATLAB, Xilinx System Generator, ModelSim, synplify Pro and Integrated Software Environments (ISE) software tools is introduced in this research. The floating point design in MATLAB has been moved to fixed point values using the most attractive and friendly Xilinx Digital Signal Processing (DSP) system generator software a model based approach associated with assistance software from Mathworks and Synplicity. Result obtained shows an important utilization in Look Up Table (LUTs) and Slices in FPGA design.

Keywords- FPGA , SDR, MATLAB, System Generator, ISE

التصميم الامثل للنظام الراديوي المعرف برمجيا باعتماد مصفوفة البوابات المبرمجة

أ.م.د. محمود فرحان مصلح
قسم هندسة تقنيات الحاسبات
كلية التقنيات الكهربائية

م.د. ماجد صلال نغمشم.
قسم هندسة تقنيات القدرة
كلية التقنيات الكهربائية

م.د. فائزة عباس عبد
قسم التقنيات الالكترونية
معهد التكنولوجيا بغداد

الخلاصة :

هذه البحث يقدم التصميم الامثل للنظام الراديوي المعرف برمجيا(SDR) باعتماد مصفوفة البوابات المبرمجة باستخدام أحدث البرامجيات. تقنية (SDR) يجب ان تستعمل بشكل ملحوظ لتامين الطلبات المتزايدة في الاتصالات

الاسلكيه والصناعة النقالة للسماح لكل أنظمة الاتصال الرقمية لتبني تقنيات تحميل وتشفير أكثر تعقيدا ، .لذلك، في هذه البحث، يعتمد على الطريق الأقصر والكفوءه لتصميم مصفوفة البوابات المبرمجة من خلال استخدام برنامج مولد النظام و ModeSim و Synplify , MATLAB , pro ISE (بيئات البرامج المتكاملة) و أدوات برامج مقدمه . تصميم تمثيل قيم الإعداد في برنامج ماتلاب يحول إلى القيم الثابتة باستخدام البرنامج الأكثر شيوعا وحداثة المتمثل بقاعدة المعالج الرقمي للإشارة . إن حاملات الإشارة ورموزها يمكن اكتشافها بالاعتماد على دائرة قفل الطور .النتائج التي تم الحصول عليها تؤكد على إمكانية تحسين وتقليل عدد الجداول(LUT) والشرائح (Slices) في تصميم مصفوفة البوابات المبرمجة.

1. Introduction

The SDR offer a flexible wireless communication resolution for a extensive range of claims, as well as cellular phones, Global System of Mobile (GSM) and military communication radios [1]. Since all hardware is physically automatic using software turn out to be fairly easy and make the SDR extremely cost effective [2]. In the surrounded applications of communication systems, the Field Programmable Gate Array is extensively used due to its reconfigurability. The very important preference of SDR claimis FPGA because it's not required hardware change out and increase the device life by repair data stream file. The level of investment an entire system is FPGA developed on a single chip, while permit in platform testing and correcting of the system. Additionally, it present an opening of utilizing hardware-software co-design to extend a high performance system for unusual applications by excluding processors for definite software functions [3-6]. The digital design of FPGA can be realize by using Hardware Description Languages (HDL) which involve more information and skills in the tools such as VHDL and Verilog. The environment for model-based design in MATLAB/simulink is almost cover all industrial and scientific areas [7]. In 2008, the Implementation of a BPSK Transceiver on Hybrid Software Defined Radio Platforms is proposed by [8]. The number of slices and LUTs occupied by this research is quit high in which the power consumption will be high also. However, Multi-Standard Programmable Baseband Modulator For Next Generation Wireless Communication is presented by [9] which is used extremely low LUTs in the FPGA design. At 2012, the Dynamic Reconfiguration Technologies Based on FPGA in Software Defined Radio System is proposed by [10] with low slices and LUTs.

In this paper, the Xilinx digital signal processing (DSP) tool in system generator is investigated with graphical interface environments based MATLAB/ simulink and block set of DSP cores is used to model the DSP system. The model based SDR system is designed and implemented using FPGA. The conventional design flow go behind for this work is clarify during the block diagram as shown in **Figure(1)** [11]**.** Firstly, MATLAB/simulinkblock set has been used to build up a software model for the system followed by hardware physically in Very High Speed Description Language (VHDL) and validate its practical simulation using ModelSim. Subsequently, this VHDL model is ported into the MATLAB –Simulink model using System Generator block set talent. The functional of hardware is validate with the data generated from the model. The Xilinx integrated software environment (ISE) tools be used to implement the design on Xilinx FPGA vertix-4. In the early 1990s, the FPGA began to approach gate counts and speeds of Application Specific Integrated Circuit (ASIC), although ASIC remains superior for both parameters. However, when FPGA reached a semi-equivalent status, it became more attractive than ASIC for design and development work, due to its flexibility and reprogram ability [12]. With efficient massive parallel MAC (Multiply-Accumulate) structures, FPGA quickly became the method of choice for most of the Digital Signal Processing (DSP) algorithm implementations leading to high level of integration of various functionalities. However, the design flow for DSP algorithms and digitallogic design via FPGA.s did not develop in unison. The DSP design flow is much shorter. One of the most world-renowned software tools for DSP algorithm design is Matlab, produced by MathworksInc [13]. Many companies researching and developing DSP, controls, and communication algorithms currently realize all of their high level modeling in Matlab, including all of the necessary test harnesses. In some cases, especially for real time applications, these designers would utilize a subprogram of Matlab, called Simulink. Simulink is a block diagram approach (connect boxes together) to system modeling that allows for modeling of streaming data, multi rate systems, and other numerically application intensive areas [14].The issue that arose was that there was a gap between the Matlab/Simulink design flow, and FPGA design flow, as is illustrated by **Figure(1)** [15]**.** In conventional approaches, the digital designs on FPGA are implemented by using hardware description languages, which requires prior knowledge and experience in the tools such as VHDL and Verilog. In our preliminary investigation, it has been shown that the FPGA-based embedded system can be efficiently designed using Xilinx Vertix-4 kits and Verilog together. The Xilinx tools enable the designer to focus on the design in system level, instead of struggling in the details of programming. However there is still a long learning curve to be familiar with these new tools. MATLAB/Simulink is an environment for model-based design, which covers almost all industrial and scientific areas. It is worthwhile to bridge over MATLAB/Simulink and FPGA design [16].

Designers needed the parallel processing advantages offered by FPGA implementation. However, there was no software suite of tools that would allow a DSP algorithm designer to interface directly with an FPGA system architect. Furthermore, there was no streamlined process to verify equivalency of a hardware implemented DSP algorithm against the original test harnesses used in MATLAB. To solve this separation in design flow issue, a new electrical engineering work force skill began to be requested by companies around the world who specialized in both DSP algorithm development and FPGA design. This skill was the ability to execute brute force translation of MATLAB and C++ code into Verilog or HDL [17].

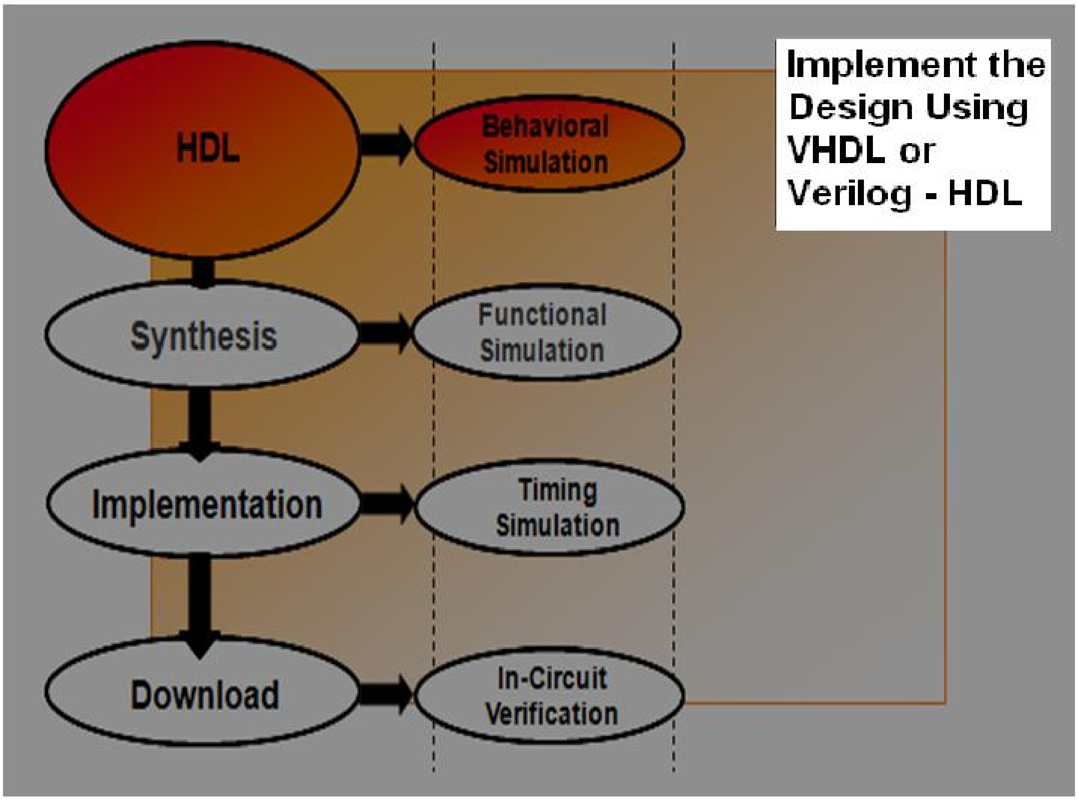


Fig .(1) HDL Verifications of DSP and FPGA Design Flows [11].

2. Proposed FPGA Design Flow

In this section, the design method and investigational setup of the FPGA design flow is approved as shown in **Figure (2)**. Software and hardware for SDR model was designed at register transfer level (RTL) manually in HDL. This is qualified by summarized the HDL code in the Xilinx block set from Xilinx system generator library. The hardware block is coupled to the simulink data during Xilinx gateway blocks for input and outputs. The functionality of hardware is confirmed by using HDL code. This methods permit to simulating hardware blocks with MATLAB simulink blocks design. The outcome of HDL simulation are changed from double precision values (floating point) to fixed point arithmetic values using output gateway blocks and passed through the block connected to the SDR model. The latency due to hardware is built-in in the model design by using delay block. The verification process between floating point and fixed point were done before connect together to become integrated design.

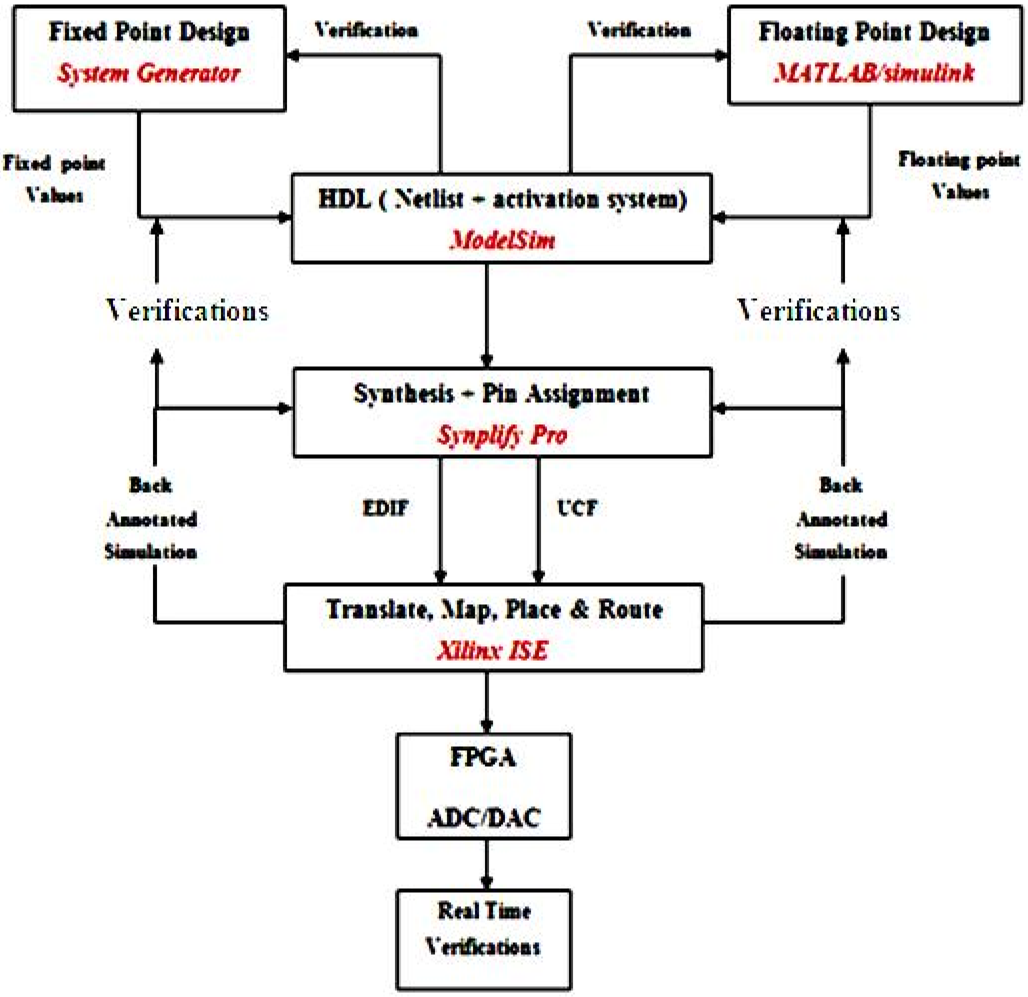


Fig .(2) : Proposed FPGA design flow

2.1 Simulink Model

To ensure and prove the proposed FPGA design path, an SDR model is designed and modeled using MATLAB and System Generator as illustrated in **Figure (3).** The model consist the transmitter, channel and receiver including all necessary part like modulation and filtering.

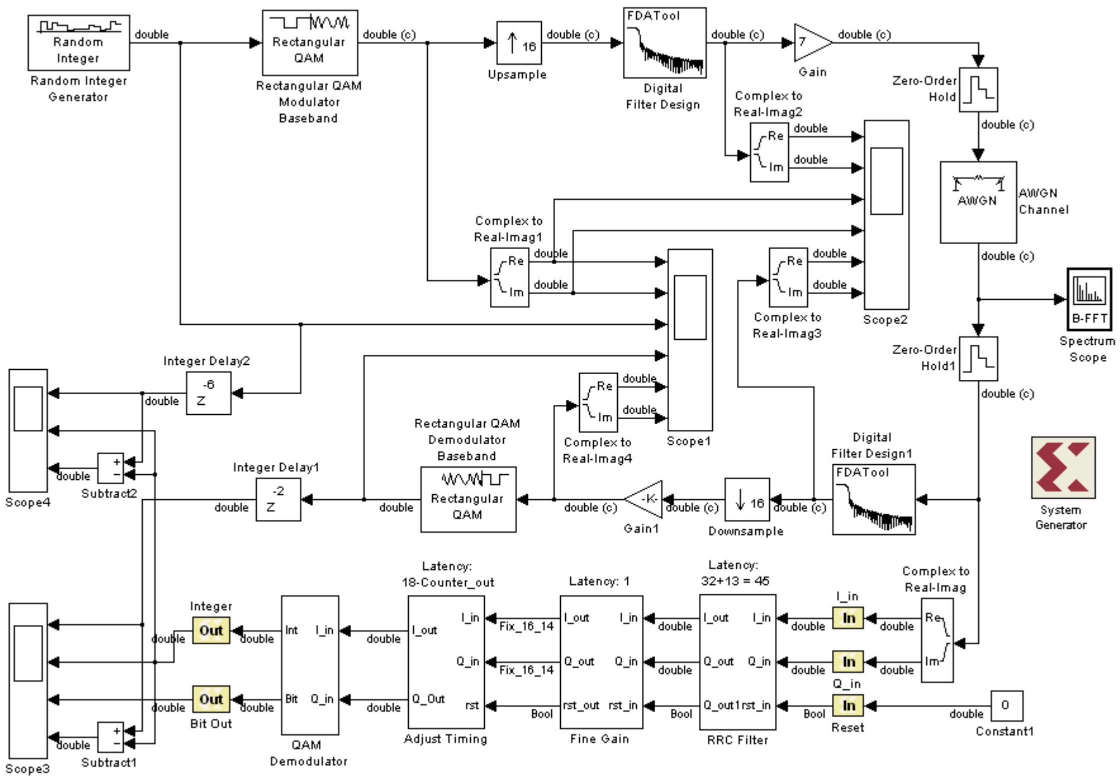


Fig .(3). SDR Model Using MATLAB and System Generator

2.2 Floating-to- Fixed Point Design

The majority realistic FPGA designs are incomplete to limited precision signal processing using fixed-point arithmetic because of the cost and complexity of floating point hardware. The mapping applications against FPGAs,a DSP algorithm designer, who frequently build up his applications in MATLAB, must find out the dynamicrange and preferred precision of input, intermediate and output signals in a design implementation to ensure that the algorithm reliability criterion are meet. The initial step in a flow to drawing MATLAB applications into hardware is the exchange of the floating point MATLAB algorithm into afixed point description using system generator. The floating point calculation in MATLAB can be transformed to a fixed point of exact accuracy for hardware design using system generator. The methods have been included in the Accel FPGA behavioral synthesis tool that reads inhigh-level descriptions of DSP applications written in MATLAB, and generates synthesizable RTL models in VHDL or Verilog by system generator and mapped onto the Xilinx Virtex-4 FPGAs. Tow phenomenon could happen through the floating point to fixed point transition, overflow and quantization error. To simulate the simulink model via system generator design, the signal have to convert before they reach any Xilinx blocks devoted for hardware. Therefore, they first converted from double precision floating point values to fixed point values represented in binary format with decimal for hardware accepted as shown in **Figure (4)**. In this case, the total number of bits is 12, and the binary point is 9, leaving 2 bits for integer plus the sign bit. Quantization shown in the right side of Figure 3, always occurs in the floating point to fixed point conversion process, since there must be some limit set to the number of decimal places that are kept for hardware representation. In this case, the whole number of bits is 12, and the binary point is 9, leaving 2 bits for the integer, plus the sign bit. Quantization, shown in the right side of Figure 4, always occurs in a floating point to fixed-point conversion, since there must be some limit set to the number of decimal seats that are reserved for hardware demonstration.

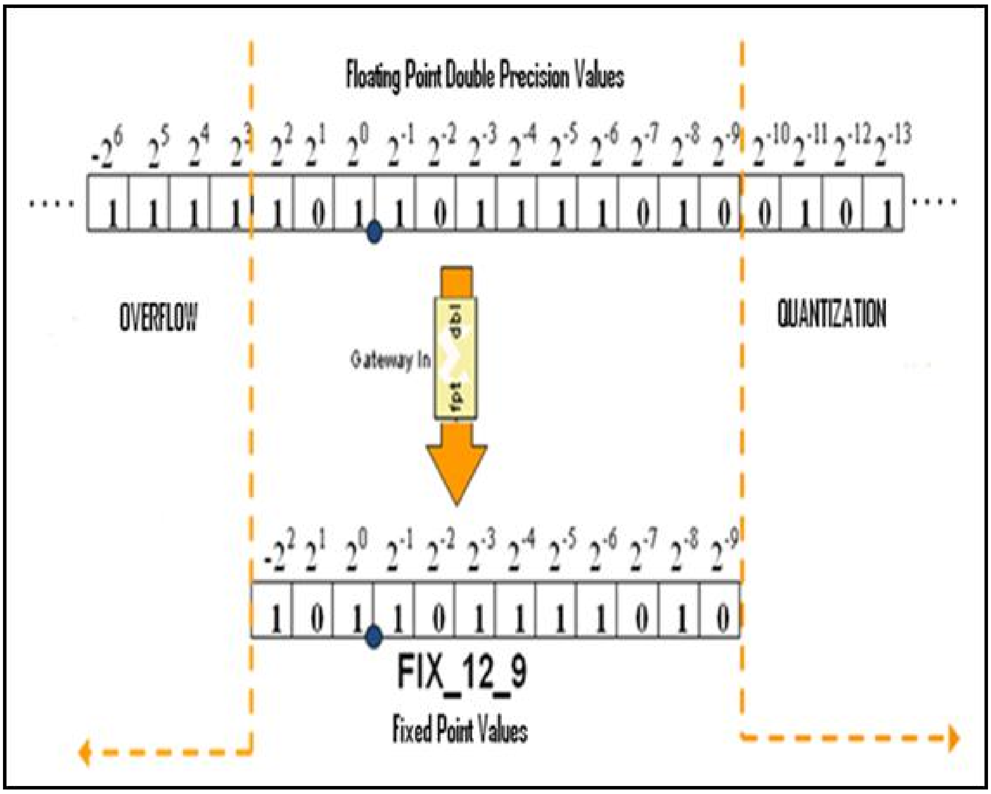


Fig .(4) : System Generator Gateway Double to Fixed-Point Conversion

2.3 HDL Design of Integrated system

The ModelSim software is used to design HDL module of activation system (plug-in Avnet Electronic Marketing P240 Analog Module)and Virtex-4 FPGA on-board ICS8442 Programmable low voltage differential signaling (LVDS) Clock Synthesizer with exact configurations. The HDL module of activation system should include all the specifications, characteristics, and features to ensure that ADC and DAC in P240 can function correctly [18-23]. The HDL module of activation system and HDLnetlist of SDR system in Verilog language are verified first before they are combined to become the HDL module of the integrated design. The simulated result of HDL netlist of the SDR system is similar to the simulation result of the system generator design. It should be noted that these simulation results are similar to the system generator design. Based on the real-time implementation of the integrated design using FPGA and P240 Analog Module, the ADC and DAC in P240 are activated and configured first before the FPGA design is run to avoid instability of ADC and DAC, which can produce undesired output to or from FPGA during the process of configuring ADC and DAC. Thus, the main clock enable (*CE*) of FPGA design is disabled during the transfer of serial programming interface (SPI) codes to ADC and DAC in P240. Controlling the main *CE* rather than the clock would be easier. This enable clear (*CE_CLR*), which requires additional logics to adjust the sampling phase of all the multi-sample data when de-asserted. The simulation results of the HDL module of integrated design is given in **Figures (5).**

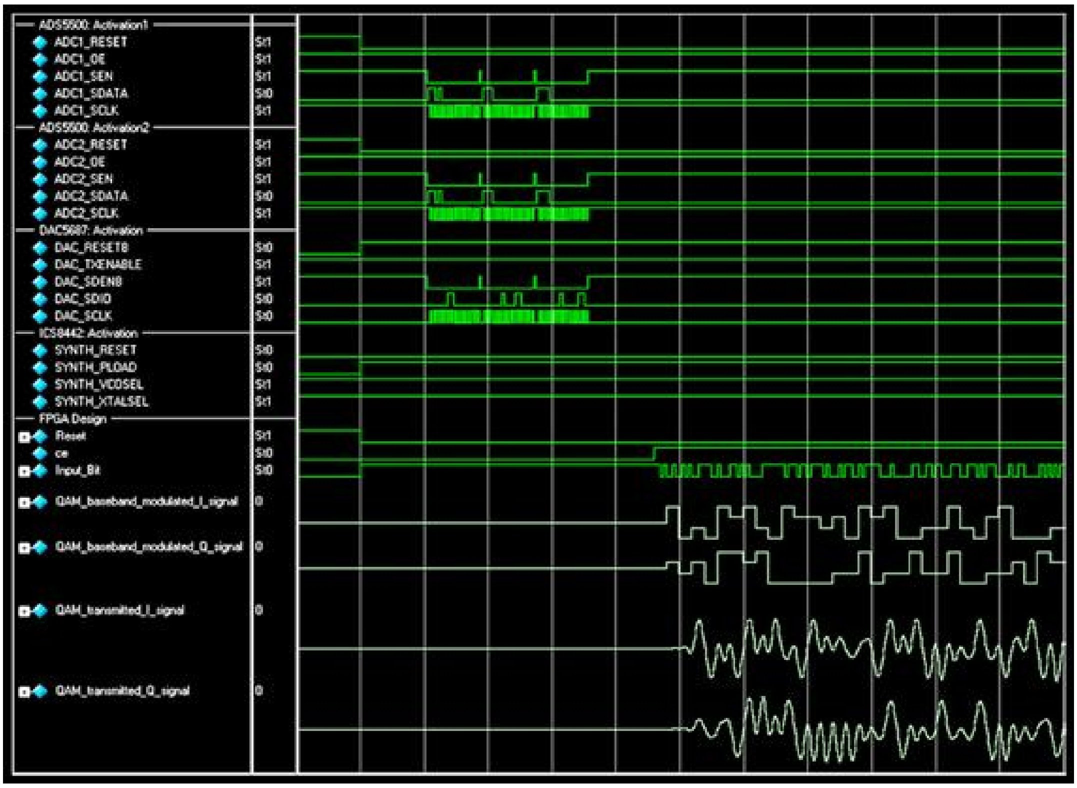
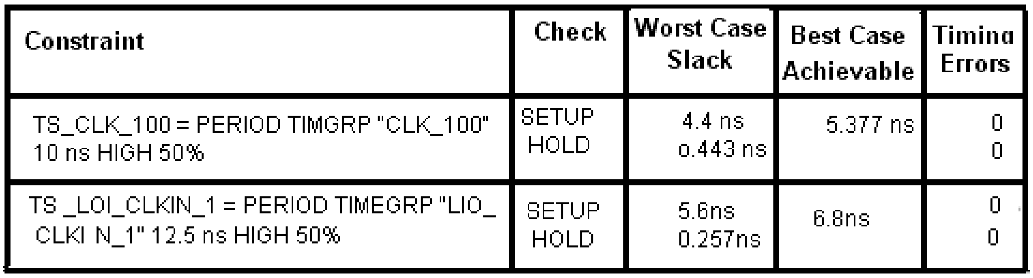


Fig .(5) Simulation Result of HDL Module of Integrated Design

2.4 HDL Synthesis

Although Xilinx ISE (integrated software environment) has its own synthesis tool called XST (Xilinx Synthesis Technology), it can only synthesize HDL netlist generated from the System Generator. Therefore, the Synplify Pro software is used to perform logic synthesis for the HDL module of integrated design in two stages, Logic compilation and optimization and Technology mapping. The Verilog HDL module of integrated design is compile to Xilinx FPGA structural elements and then optimize the integrated design to make it as small as possible to improve circuit performance. The Mapping step is to optimize the integrated design to Xilinx FPGA logic components using architectural-specific technique. Timing characteristics comprise another important factor that affects the performance of FPGA implementation. Thus, the estimated period (required path delay) for the FPGA element must not exceed the requested clock period. For this reason, the timing slack (requested period - estimated period) should have a positive value; otherwise, the integrated design has to be reworked. The clock frequencies used for ADC/DAC are set to 100 MHz for CLK_100 (ADC/DAC SPI process) and 80 MHz for LIO_CLKIN_1 (16-QAM SDR transmitter and receiver). The estimated timing report for the synthesized design meets the time constraints, as shown in **Tables(1)**.

Table (1) Estimated Timing Report

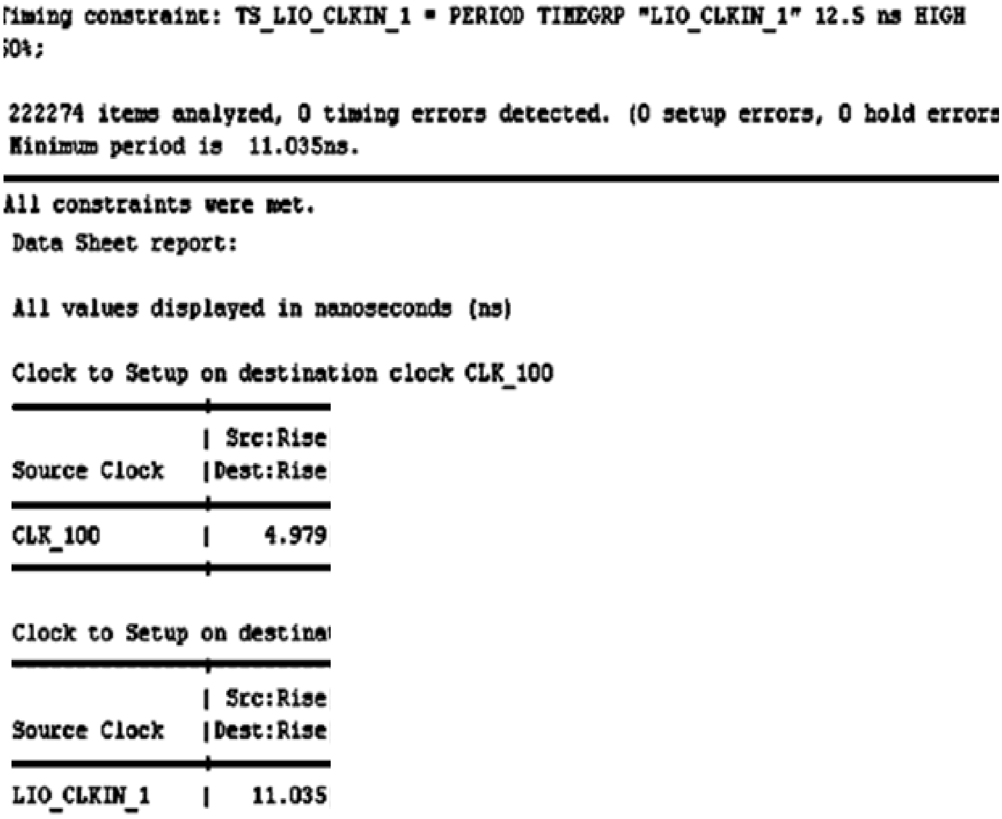


Constraint	Check	Worst Case Slack	Best Case Achievable	Timing Errors
TS_CLK_100 = PERIOD TIMGRP "CLK_100" 10 ns HIGH 50%	SETUP HOLD	4.4 ns o.443 ns	5.377 ns	0 0
TS _LOI_CLKIN_1 = PERIOD TIMEGRP "LIO_ CLKI N_1" 12.5 ns HIGH 50%	SETUP HOLD	5.6ns 0.257ns	6.8ns	0 0

2.5 FPGA Implementation

The final step in FPGA designing is the implementation of design model on Xilinx Virtex-4 using ISE software tools. The synthesis output files generated by the ISE software in electronic design interface file (EDIF) and user constraints file (UCF) forms represent the optimized netlist of integrated design, timing constraints, and FPGA pin assignment that have been implemented into the FPGA development board by three steps called Translate, Map and place &Route. The Translate step is convert the netlist file of integrated design in EDIF format to native generic database (NGD) file, which contains logic description of hierarchical components and Xilinx primitives for the integrated design using NGD build program. The Map process is to Perform logical DRC on the NGD file and map the design logic to slices and input-output (I/O) cells in FPGA to create native circuit description (NCD) file. The Place and Route is the LUTs design in mapped NCD file is place and route into FPGA based on timing constraints using timing analysis tools with no errors found. Finally, The Bit generation and program download is used to generate configuration bit-stream file in BIT format form, and subsequently downloaded into FPGA via JTAG cable using the iMPACT program.

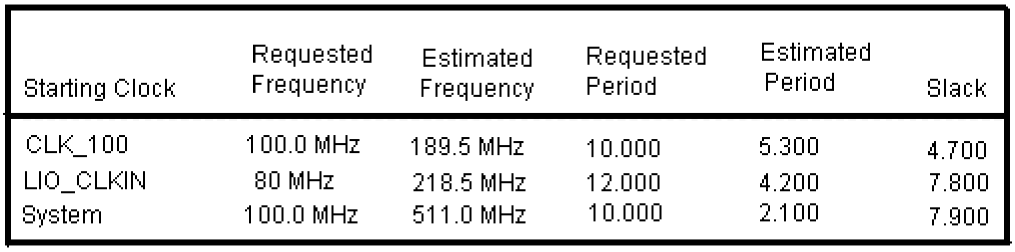
The timing requirement is satisfied as shown in the post-PAR static timing report as illustrated in **Figure (6)** and **Tables (2)**



```
Timing constraint: TS_LIO_CLKIN_1 = PERIOD TIMEGRP "LIO_CLKIN_1" 12.5 ns HIGH
50%;

222274 items analyzed, 0 timing errors detected. (0 setup errors, 0 hold errors
Minimum period is  11.035ns.

All constraints were met.
 Data Sheet report:

 All values displayed in nanoseconds (ns)

 Clock to Setup on destination clock CLK_100
 ---------------+---------
                | Src:Rise
 Source Clock   |Dest:Rise
 ---------------+---------
 CLK_100        |    4.979
 ---------------+---------

 Clock to Setup on destinat
 ---------------+---------
                | Src:Rise
 Source Clock   |Dest:Rise
 ---------------+---------
 LIO_CLKIN_1    |   11.035
```

Fig. (6) post-PAR (final) static timing report

Table (2) Post-PAR Static Timing Report



Starting Clock	Requested Frequency	Estimated Frequency	Requested Period	Estimated Period	Slack
CLK_100	100.0 MHz	189.5 MHz	10.000	5.300	4.700
LIO_CLKIN	80 MHz	218.5 MHz	12.000	4.200	7.800
System	100.0 MHz	511.0 MHz	10.000	2.100	7.900

The implementation steps shown in **Figure (7)** are read in the constraints file that consists of three major steps: translate, map, and place and route.

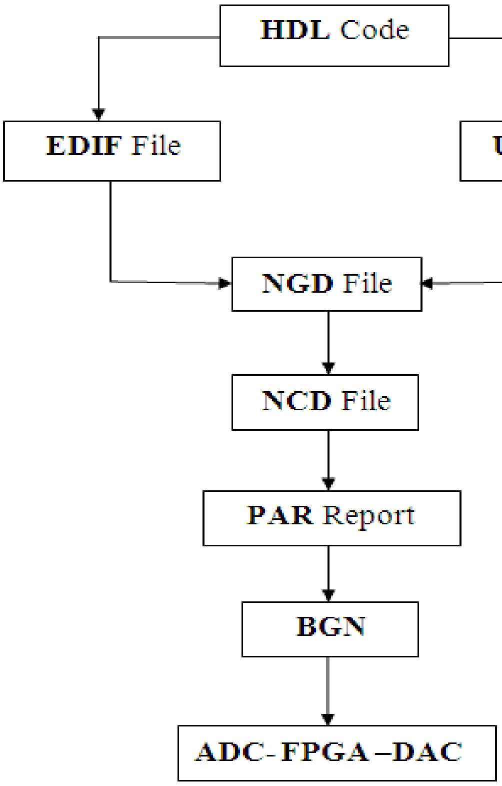


Fig .(7) FPGA Implementation ISE Steps

The translate step essentially flattens the output of the synthesis tool into a large single netlist. A netlist in general, is a large list of gates which is compressed at this stage to remove any hierarchy. The map step groups the logical symbols in the flattened netlist into physical components, specific to the target device. The place and route step places each of these physical components onto the FPGA chip and connects them through the switch matrix and dedicated routing lines.

3. Results

Figure (8) shows the input and output signals of the proposed SDR model. The difference between the original and the recovered message bits is zero, and the difference between the original and the recovered symbol integers is also zero. This indicates that the timing synchronization between the transmitter and the receiver is optimized and the symbol remapping removes the noise caused by AWGN using the tolerance range of threshold in decision-making technique. In this technique, the threshold level is limited by three values (0.667, 0.333, and 0.125) that represent the tolerance range in I and Q channels.

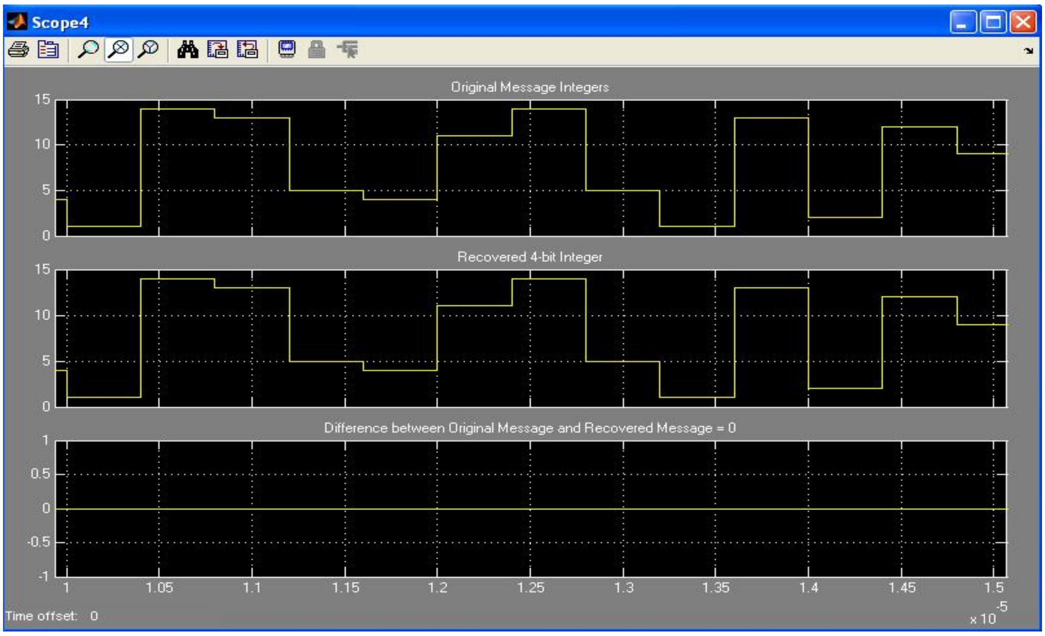
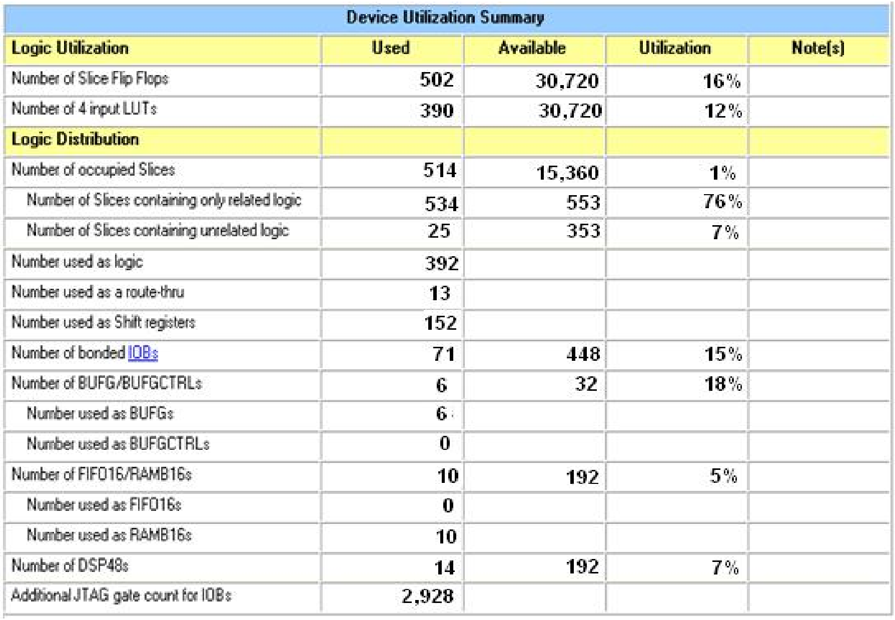


Fig .(8) Input and output signal of proposed software defined radio

The transmitter and receiver project status and device utilization summary are reported by the ISE program, as shown in **Table (3).** These tables provide the total number of slices and LUTs used in this design, which represent the total area used in FPGA. Depending on the number of devices used in FPGA, the total power consumption in the SDR model implementation can be seen, according to the number of slices and LUTs. The total LUP for proposed design is 390 with 514 Slices and illustrated in utilization summary.

Table (3) FPGA utilization summary

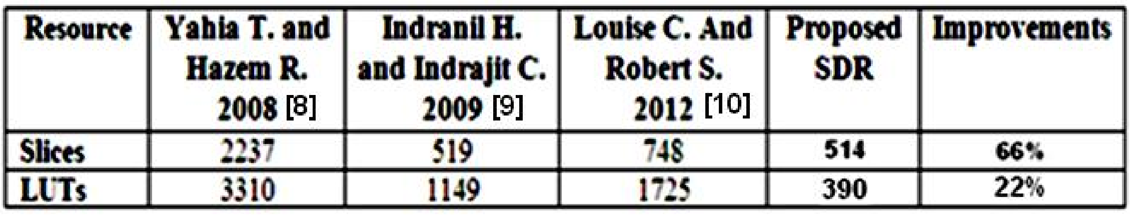


Device Utilization Summary

Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	502	30,720	16%	
Number of 4 input LUTs	390	30,720	12%	
Logic Distribution				
Number of occupied Slices	514	15,360	1%	
Number of Slices containing only related logic	534	553	76%	
Number of Slices containing unrelated logic	25	353	7%	
Number used as logic	392			
Number used as a route-thru	13			
Number used as Shift registers	152			
Number of bonded IOBs	71	448	15%	
Number of BUFG/BUFGCTRLs	6	32	18%	
Number used as BUFGs	6			
Number used as BUFGCTRLs	0			
Number of FIFO16/RAMB16s	10	192	5%	
Number used as FIFO16s	0			
Number used as RAMB16s	10			
Number of DSP48s	14	192	7%	
Additional JTAG gate count for IOBs	2,928			

The minimum number of slices and LUTs obtained by the conventional designs are 519 and 1149, respectively [9]. This means that the proposed SDR implementation is more efficient than the conventional design in terms of area and power consumption. A comparison between the proposed design and conventional design is shown in **Table (4).** Each researcher provides different resource of slices and LUTs.

Table (4) FPGA Slices and LUTs comparison



Resource	Yahia T. and Hazem R. 2008 [8]	Indranil H. and Indrajit C. 2009 [9]	Louise C. And Robert S. 2012 [10]	Proposed SDR	Improvements
Slices	2237	519	748	514	66%
LUTs	3310	1149	1725	390	22%

4. Conclusions

In this paper, a talented paths to design of Virtex-4 FPGAs has been developed and improved to work with 4G system and software defined radio. A novel technique has been used to reduce the LUT and slices in FPGA area therefore, decrease the power consumption and size in future implementation of FPGAs. The proposed technique paths is incorporate with MATLAB/ simulink and system generator. The Synplify pro synthesis and ISE software is also used to synthesis the HDL code and generate the bit stream to download it to FPGA board. Results obtained from the implementation of this design, has shown that the proposed algorithm reduces the FPGA logic utilization by nearly 47 % and 25% in terms of FPGA elements of slices and LUTs respectively, compared to conventional designs. In addition, a design and implementation of software defined radio model was done to verify the design performance under channel noise.

5. References

1. **V. B. Alluri, J. R. Heath, and M. Lhamon, " A new multichannnel ,coherent amplitude modulated, gate array technology implementation,"*IEEE Transaction on Signal Processing*, vol. 58, pp. 5369-5384,October 2010.**
2. **J. Mitola, " Software radios: survey, critical evaluation and futuredirections," *IEEE Aerospace and Electronics Systems Magazine*, vol. 8,pp. 25-36, April 1993.**
3. **M. Palkovic, P. Raghavan, M. Li, A. Dejonghe, L. Van der Perre, and F.Catthoor, " Future Software-Defined Radio Platforms and MappingFlows," *IEEE Signal Processing Magazine*, vol.27, no.2, pp.22-33,March 2010.**
4. **J. Kim, H. Seungheon, and S. Choi; , "Implementation of an SDRsystem using graphics processing unit," *IEEE CommunicationsMagazine* , vol.48, no.3, pp.156-162, March 2010.**

5. **Y.Lu, E. Oruklu, and J. Saniie, " Fast chirplet transform with FPGAbasedimplementation," *IEEE Signal Processing Letters*, vol. 15, pp.577-580, December 2008.**
6. **A. T. Moreo, P. N. Lorente. F. S. Valles, J. S. Muro, " Experiences ondeveloping computer vision hardware algorithm using Xilinx systemgenerator," Microprocessors and Microsystems: Special Issue onFPGAs; Case Studies in Computer Vision and Image Processing,, pp.411-419, November 2005**
7. **C. Chang, J. Wawrzynek and R. W. Brodersen, " BEE2: a high-endreconfigurable computing system," *IEEE Design & Test of Computers*,vol.22, no.2, pp. 114- 125, March-April 2005**
8. **YahiaTachwali and HazemRefai, "Implementation of a BPSK Transceiver on Hybrid Software Defined Radio Platforms", IEEE Explore, vol.2, PP 200-215, 2008**
9. **IndranilHataiand IndrajitChakrabarti, "Multi-Standard Programmable Baseband Modulator For Next Generation Wireless Communication", Department of Electronics and Electrical Communication Engineering, Indian Institute of Technology, West Bengal, India, 2009**
10. **Louise Crockett and Robert Stewart, "Dynamic Reconfiguration Technologies Based on FPGA in Software Defined Radio System", Journal of Signal Processing System, vol. 69 , PP 75–85 , 2012**
11. **Xilinx ISE 9.2i Software Manuals: Constraints Guide, and Development System Reference Guide, Xilinx, Inc., 2007.**
12. **Xilinx, " System Generator 12.1 User's Manual", Xilinx. Inc., 2010.**
13. **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.**
14. **The MathWorks, Inc. DSP – Digital Signal Processing& Communications. [Online]. July 10, 2009**
15. **Xilinx, Inc., *DSP Design Flows in FPGA Tutorial Slides*, 2003.**
16. **J. G. Prokis, *Digital Communications*, 4th ed., New York: McGraw- Hill, 2001.**
17. **V. K. V. R. Gari, "FPGA-based QPSK Transceiver Design", TechnicalReport, Department of Electrical and Computer Engineering, BradleyUniversity, November 2008.**
18. **Altera Corporation, " PLL and Clocking Glossary", Altera, 2010.**
19. **D. Benson, " The Design and Implementation of a GPS ReceiverChannel, An FPGA plus DSP Creats a Powerful Combination", *DSPMagazine*, October 2005.**
20. **Xilinx Corp., Virtex-4XtremeDSP Design Considerations UserGuide,2008**
21. **Xilinx Corp., DDS Compiler v2.1, 2008**
22. **Xilinx Corp., System Generator User Guide,2008**
23. **Xilinx Corp., Xilinx ISE 10.1 Software Manuals, 2008.**