ISSN (Online) 2321-2004
ISSN (Print) 2321-5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 3, Issue 5, May 2015

Design and Implementation of Effective QPSK Modulator Based on FPGA

ShruthiD[1], Chiranjeevi G N[2], DrSubhash Kulkarni[3]

Mtech Student, Department of ECE , PESIT South Campus, Bangalore, India [1]

Assistant Professor, Department of ECE, PESIT South Campus,Bangalore,India [2]

HOD, Department of ECE, PESIT South Campus, Bangalore, India[3]

Abstract:The objective of this project is to develop an implementable effective QPSK modulator that uses less power for operation. The design of QPSK modulator has be done using Matlab/Simulink and Xilinx System Generator. The modulator algorithm has been implemented on FPGA(Spartan3) using the Verilog Hardware Description Language on Xilinx ISE Design suite 13.2. Then the final designed and simulated results are applied for real time such as Satellite Communication, Video Conferencing etc.

Keywords: Matlab/Simulink, Xilinx, System generator, FPGA, Spartan3.

I. INTRODUCTION

Quadrature Phase Shift Keying is a modulation technique which is widely used in wireless communication system due to its ability to transmit twice the data rate for a given bandwidth. Even though the QPSK modulator consumes less power in a present devices but for a system such as satellite and mobile devices where their operations and power limited.Wireless communication systems require high data rate for efficient transmission of information. Modulation techniques have been introduced to increase the efficiency in data transmitting and receiving rate modulation method used in communication system is Quadrature Phase ShiftKeying (QPSK), which is one of the form of Phase Shift Keying (PSK) modulation scheme. The digital method was chosen due to its advantages of digital solutions are apparent. Some of the main advantages of the digital solution are repeatability, cost and the simpler reconfiguration when compared with analog solutions. In QPSK modulation, the carrier phase takes four discrete states that are used to indicate a group of two input data bits. Each group takes one of the QPSK states i.e. ±45º and ±135º.

Fig.1 below shows the constellation diagram of QPSK Signal**.**

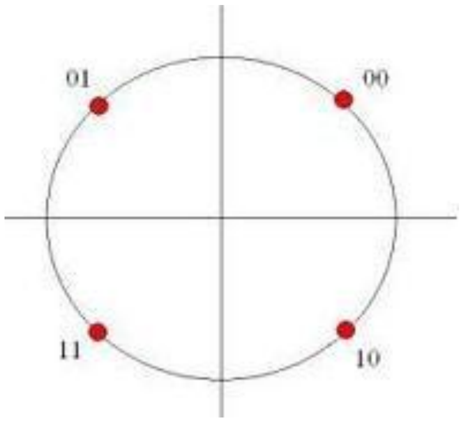


Fig.1.Qpsk signal Constellation

II. GENERATION OF A QPSK SIGNAL

Figure below shows the mathematical implementationof qpsk[1].

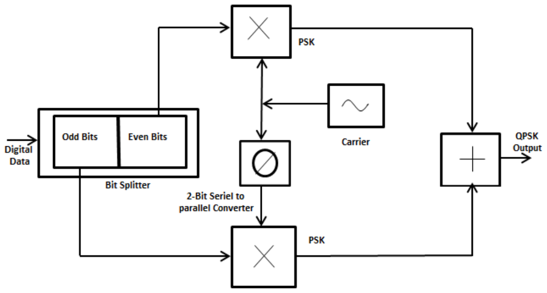


Fig.2.Qpsk Block Diagram

Where the first bit represents In-phase (I) and the second bit represent the Quadrature-phase(Q). QPSK modulation is a pair of binary PSK [BPSK], but the data transmission in QPSK is twice when compared to BPSK. The Bit Error Rate (BER) over Signal-to-Noise ratio (SNR) for both the modulation schemes is same. The two BPSK signals are added to produce the required QPSK signal. Since two bit information is transmitted in an interval T, the symbol period for QPSK is two times the bit period i.e. T=2Tb, while for BPSK the symbol period is same as bit period T=Tb. The QPSK signal utilizes half the bandwidth of BPSK signal which consumes low throughput and also has the complexity in hardware implementation [2][4].

The basic idea behind QPSK exploits the fact that cos (2πfct) and sin (2πfct) are orthogonal over the interval [0, Tb] when fc = k/Tb, k integer. Just as in analog modulation, this can be used to transmit two different messages over the same frequency band.[3].

At the input of the modulator, the Bernoulli binary generator is used as a data source which generates random data stream, then the data stream is applied to serial to parallel converter. digital data’s are split into even bits (i.e., bits 0,2,4 and so on) and odd bits (i.e.,bits 1,3,5 and

DOI 10.17148/IJIREEICE.2015.3510

so on) are stripped from the data stream by a "bit-splitter" and are multiplied with a carrier to generate a BPSK signal (called PSKI). At the same time, the data's odd bits (i.e., bits 1, 3, 5 and so on) are stripped from the data stream and are multiplied with the same carrier to generate a second BPSK signal (called PSKQ). However, the PSKQ signal's carrier is phase shifted by 90° before being modulated [1]. The qpsk modulator is designed andsimulated using Matlab/Simulink environment and System generator, modulator algorithm has implemented on Fpga(Spartan 3)using Verilog hardware description language Xilinx ISE design suite 13.2.[5]

III. OVERVIEW OF QPSK SYSTEM IN MATLAB\ SIMULINK

Simulink software models, simulates and analyses dynamic systems. With Simulink one can easily build models to meet the design needs. Simulink supports Linear and Nonlinear Systems, Modeled in Continuous time, Sampled time, or a Hybrid of the two. Systems can also be multirate-having different parts that are sampled at different rates. Simulink provides a graphical user interface (GUI) for building models asblock diagrams, allowing you to draw models as you would with pencil and paper. Simulink also includes a comprehensive block library of sinks, sources,linear and nonlinear components, and connectors. If these blocks do not meet design needs, however, you can also create your own blocks. The interactive graphical environment simplifies the modeling process, eliminating the need to formulate differential and difference equations in a language or program. Figure.3.Below shows the schematic of QPSK modulator in Matlab\Simulink has been designed using various blocks in Simulink library. Fig.4 shows the simulation result of qpsk signal.The Simulink Blockset contains:

Bernoulli Binary Generator Block: The Bernoulli Binary Generator block generates random binary numbers using a Bernoulli distribution. The Bernoulli distribution with parameter p produces zero with probability p and one with probability 1-p. The Bernoulli distribution has mean value 1-p and variance p(1-p). The Probability of a zero parameter specifies p, and can be any real number between zero and one.Serial to Parallel Converter Block:The Serial to Parallel block takes a series of inputs of any size and creates a single output of a specified multiple of that size. Unipolar to Bipolar Block: The Unipolar to Bipolar Converter block maps the unipolar input signal to a bipolar output signal. Carrier Block: It is used to generates the sine and cosine signal.
Sum Block: It is used to add inphase and quadrature phase signal which results in qpsk modulated signal.

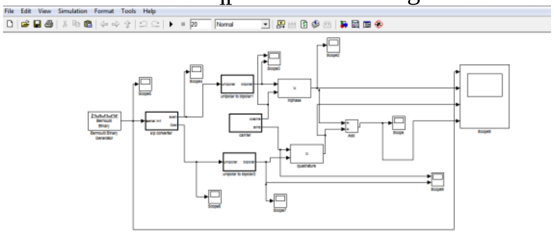


Fig.3. QPSK Schematic in Matlab\Simulink

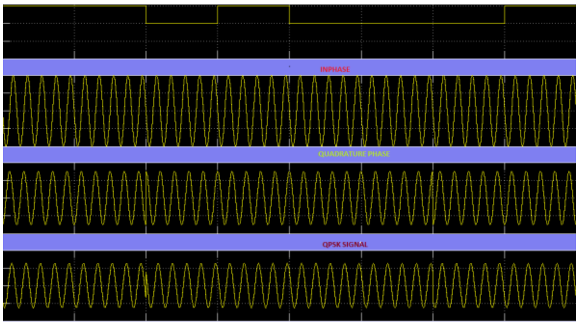


Fig.4.Simulation results of QPSK Signal in Matlab\Simulink

IV. IMPLEMENTATION OF QPSK SYSTEM IN SYSTEM GENERATOR

The Xilinx System Generator [7] for DSP is a system level modeling and design tool that facilitates FPGA design and has the ability to work at a higher level of abstraction. It enables the use of the MathWorks graphical model based Simulink design environment for FPGA design. The System Generator integrates itself with Simulink and FPGA designs are captured by using the Xilinx specific Blockset. Thus, designing a hardware model in Simulink is as simple as designing any other Simulink model with the only difference being the use of Xilinx Blockset instead of those found in Simulink.

The System Generator provides many DSP building blocks in the form the Xilinx DSP Blockset for the Simulink environment. The variety in this Blockset ranges from common DSP blocks such as adders, multipliers, registers etc to more complex blocks such as FFTs, filters, memories, forward error correction etc. Thus, previous experience with low level system design and HDLs is not required when using this tool.

The System Generator uses the Xilinx ISE software and IP core generators to convert designed model into the equivalent HDL code. The remaining FPGA implementation steps including synthesis, place and route, etc. are automatically performed to generate bit file that is downloaded on to the FPGA.

Fig.5 Illustrates the schematic of qpsk modulator in Xilinx \System generator and Fig.6 shows the output waveforms of QPSK signal in System generator.

The Xilinx\System generator Blockset contains:
The Gateway in blocks:The Xilinx Gateway In blocks are the inputs into the Xilinx portion of your Simulink design. These blocks convert Simulink integer, double and fixed-point data types into the System Generator fixed-point type. Each block defines a top-level input port in the HDL design generated by System Generator. The Gateway out blocks:Xilinx Gateway Out blocks are the outputs from the Xilinx portion of your Simulink design. This block converts the System Generator fixed-point data type into Simulink Double.

Logical Block:It performs bitwise logical operations on 2,

DOI 10.17148/IJIREEICE.2015.3510

3, or 4 fixed-point numbers. Operands are zero padded and sign extended as necessary to make binary point positions coincide; then the logical operation is performed and the result is delivered at the output port. Inverter Block:It calculates the bitwise logical complement of a fixed-point number. Delay Block:It implements the delay of L cycles.The delay value is displayed on the block in the form z^{-L}. Any data provided to the input of the block will appear at the output after L cycles.This block is used mainly for matching pipeline delays in other portions of the circuit. Mult Block: It computes the product of the data on its two input ports, producing the result on its output port. AddSub Block:The operation can be fixed (Addition or Subtraction) or changed dynamically under control of the sub mode signal.

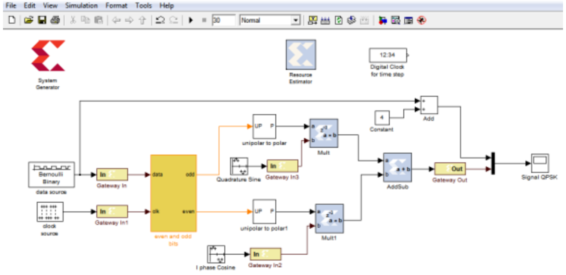


Fig.5. Schematic of QPSK In System Generator

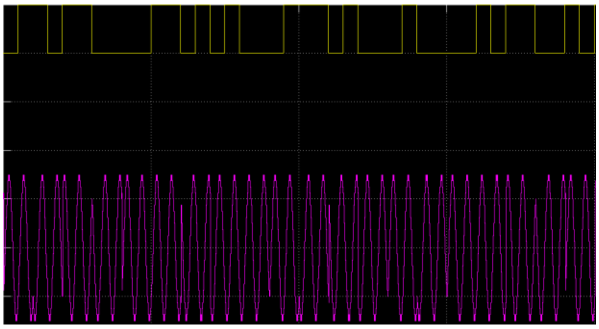


Fig.6. Simulation results of QPSK signal in System Generator.

V. RESULTS

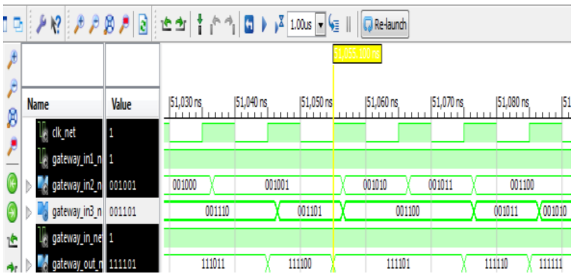


Fig.7.Shows the Behavioral Simulation of QPSK

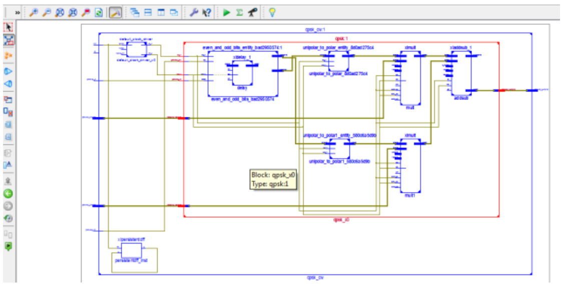
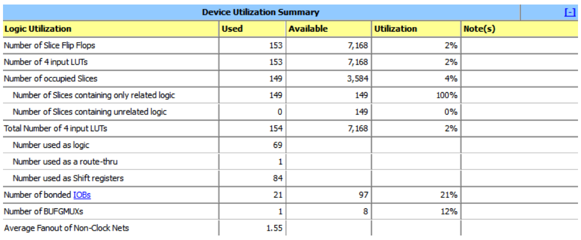
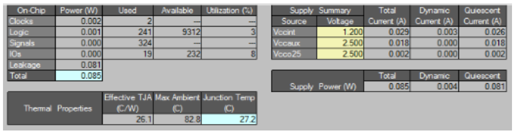


Fig.8.RTL Schematic of QPSK



Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	153	7,168	2%	
Number of 4 input LUTs	153	7,168	2%	
Number of occupied Slices	149	3,584	4%	
Number of Slices containing only related logic	149	149	100%	
Number of Slices containing unrelated logic	0	149	0%	
Total Number of 4 input LUTs	154	7,168	2%	
Number used as logic	69			
Number used as a route-thru	1			
Number used as Shift registers	84			
Number of bonded IOBs	21	97	21%	
Number of BUFGMUXs	1	8	12%	
Average Fanout of Non-Clock Nets	1.55			

Fig.9. Device Utilization for QPSK



On-Chip	Power (W)	Used	Available	Utilization (%)
Clocks	0.002	2	---	---
Logic	0.001	241	9312	3
Signals	0.000	324	---	---
IOs	0.000	19	232	8
Leakage	0.081			
Total	0.085			

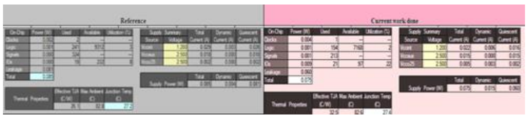
Supply Summary Source	Voltage	Total Current (A)	Dynamic Current (A)	Quiescent Current (A)
Vccint	1.200	0.029	0.003	0.026
Vccaux	2.500	0.018	0.000	0.018
Vcco25	2.500	0.002	0.000	0.002

	Total	Dynamic	Quiescent
Supply Power (W)	0.085	0.004	0.081

Thermal Properties	Effective TJA (C/W)	Max Ambient (C)	Junction Temp (C)
	26.1	82.8	27.2

Fig.10Power Utilization for QPSK

VI. COMPARISON OF RESULTS



Reference

On-Chip	Power (W)	Used	Available	Utilization (%)
Clocks	0.002	2	---	---
Logic	0.001	241	9312	3
Signals	0.000	324	---	---
IOs	0.000	19	232	8
Leakage	0.081			
Total	0.085			

Supply Summary Source	Voltage	Total Current (A)	Dynamic Current (A)	Quiescent Current (A)
Vccint	1.200	0.029	0.003	0.026
Vccaux	2.500	0.018	0.000	0.018
Vcco25	2.500	0.002	0.000	0.002

	Total	Dynamic	Quiescent
Supply Power (W)	0.085	0.004	0.081

Thermal Properties	Effective TJA (C/W)	Max Ambient (C)	Junction Temp (C)
	26.1	82.8	27.2

Current work done

On-Chip	Power (W)	Used	Available	Utilization (%)
Clocks	0.004	1	---	---
Logic	0.001	154	7168	2
Signals	0.001	213	---	---
IOs	0.009	21	97	22
Leakage	0.060			
Total	0.075			

Supply Summary Source	Voltage	Total Current (A)	Dynamic Current (A)	Quiescent Current (A)
Vccint	1.200	0.022	0.006	0.016
Vccaux	2.500	0.015	0.000	0.015
Vcco25	2.500	0.005	0.003	0.002

	Total	Dynamic	Quiescent
Supply Power (W)	0.075	0.015	0.060

Thermal Properties	Effective TJA (C/W)	Max Ambient (C)	Junction Temp (C)
	32.5	82.6	27.4

Table1.Comparison of results in two Platforms Spartan 3E and Spartan 3

VII. CONCLUSION

The QPSK System have been designed and implemented successfully in Matlab\Simulink and Xilinx\System Generator and the code generated from system generator has been simulated using ISE 13.2 and FPGA design as well as implemented on Spartan 3(starter kit board)using Verilog Hardware Description Language.The results are compared with reference[1].The obtained results can be analysed of real time applications such as satellite communication and video conferencing.

ACKNOWLEDGMENT

I acknowledge my gratitude to everyone who encouraged my project. I would like to express my deep sense of gratitude to **Mr.Chiranjeevi G N** Assistant Professor Dept. of ECE for his invaluable guidance, suggestions, and immense help throughout the course of the project. I would also like to thank **Dr.SubhashKulkarni** Head of the Department of ECE for his valuable suggestion in the development stage of this project.

REFERENCES

[1]. Kaliprasanna Swain and Manoj Kumar Sahoo " FPGA Implementation of QPSK Modulator Based on Matlab/ Xilinx System generator" International Journal of Analytical, Experiment and Finite Element Analysis (IJAEFEA) Issue.3,Vol.1, July 2014.

[2]. K.Anitha Umesharaddy, B.K.Sujatha "FPGA Implementation Of High Throughput QPSK Modulator using Verilog HDL" International Journal Of Advanced Computer Research Volume-4 Number-1 Issue-14 March-2014.

[3]. S.O. Popescu, A.S. Gonteanand, D.Ianchis,"QPSK Modulator on FPGA", SISY 2011, 2011 IEEE 9th International Symposium on Intelligent Systems and Informatics, September 8-10, 2011, Subotica, Serbia.

[4]. S.O. Popescu, A.S.Gontean and G.Budura,"BPSK System on Spartan 3EFPGA", SAMI 2012, 10th IEEE Jubilee International Symposium on Applied Machine Intelligence and Informatics, January 26-28, 2012, Herl'any, Slovakia

[5]. ISE 13.2, Quick Start Tutorial, Xilinx

[6]. Getting Started with Simulink user guide 2007

[7]. Getting Started with Xilinx\System Generator ISE 8.2.01,Matlab\Simulink R2006a.

DOI 10.17148/IJIREEICE.2015.3510