ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

System Generator: FPGA Design tool for DSP Applications

Sharad B. Gholap[1], Dr. R. V. Kshirsagar[2], Amrut Manohar Nadgouda[3]

Research Scholar, PCE, Nagpur, RTMU, Nagpur, India[1]

Principal, PIGCE, Nagpur, RTMU, Nagpur, India[2]

PG Student, Sinhgad Institute of Technology, Lonavala, SPPU Pune, India[3]

ABSTRACT: Advanced sign preparing capacities have generally been executed on programmable stage of DSPs. Be that as it may, as the necessities of numerous computationally escalated applications are exceeding the preparing abilities of DSPs, the utilization of FPGAs has turned out to be particularly common. Additionally, the exploration is continuous to create and utilize abnormal state outline devices, which abbreviate the advancement time required for actualizing signal preparing arrangements utilizing FPGAs. In this paper, we display our approach of outlining the model for a helpful correspondence method, to be specific MIMO, utilizing Xilinx System Generator (XSG) and AccelDSP (for supporting XSG). Our methodology demonstrates the unmistakable points of interest, for example, reusability, shorter outline times because of less bugs, which such apparatuses offer. We additionally audit the relative value of FPGA over DSPs as stages for DSP execution. We overview condition of-craftsmanship utilizations of FPGAs for DSP applications, for example, MIMO and computerized hardware. As a configuration instrument for FPGA, extraordinary accentuation is on the points of interest and constraints of Xilinx System Generator as decision of productive outline apparatus is extremely vital choice. An effectively vast client base of Matlab and Simulink alongside System Generator's own predominant execution settles on it an inexorably favored decision of numerous DSP designers of today.

I. INTRODUCTION

As of late, Field Programmable Gate Array (FPGA) innovation has turned into a feasible focus for the usage of calculations suited to Digital Signal Processing applications. Field-programmable entryway exhibits (FPGAs) are nonconventional processors constructed basically out of rationale pieces associated by programmable wires. Every rationale square has one or more lookup tables (LUTs) and a few bits of memory. Accordingly, rationale squares can actualize self-assertive rationale capacities (up to a couple of bits). In this way FPGAs, overall can actualize circuit graphs, by mapping the doors and registers onto rationale pieces.. Of course, entropy based revelation is showed up as a healthy technique for ID with less recognition time. This technique relies on upon the thought that for an adjusted sign the entropy or precariousness is more diminutive than for a sign without any information. In this paper, we examine entropy estimation using cyclic parts of getting signs. This relies on upon the thought that the entropy of autocorrelation at different cyclic frequencies in the repeat space has a significant component territory. This, along these lines, is a direct result of the way that the Fourier change of the cyclic autocorrelation limit gives the power range at different cyclic frequencies and its quality will vacillate extensively for sign stood out from discretionary fuss. Before long, distinguishing execution [probability of area (Pd) and false alert probability (Pfa)] of a lone CR is routinely com ensured by multi way obscuring and shadowing issues in the medium. To direct the impact of these issues, pleasing identifying has been gave off an impression of being an effective method to redesign the revelation execution by abusing spatial contrasting qualities. The objective of this paper is to enhance the area probability by proposing another extent recognizing technique considering entropy estimation using cyclostationary segments of the got signals. This strategy is extended to united Multinode range recognizing using sensitive decision mix reason to improve the execution of the system. Finally, the execution of the proposed entropy area using cyclic segments is differentiated and imperativeness acknowledgment using Neyman–Pearson (NP) establishment, essentialness revelation using Bayesian guideline, cyclostationary highlight (SCF) recognizable proof and entropy area in Frequency territory. Automated Video Broadcasting-Terrestrial (DVB-T) signal (as showed up in Fig. 1) is considered as a PU signal under Additive White Gaussian Noise (AWGN), Rayleigh obscuring and shadowing environment. For completing the sign

DOI:10.15662/IJAREEIE.2015.0503140

get ready estimations into gear, Field Programmable Gate Arrays (FPGAs) are typically used. The purposes of hobby are rate of operation as a result of parallelism, low power use and reconfigurable segments. Since extent identifying will be finished in each CR unit, hardware execution of this is essential and generally as basic. In the later past, FPGAs are when in doubt broadly used as a piece of sign get ready applications. Along these lines, the proposed range identifying figuring in perspective of entropy estimation is realized in Xilinx Virtex-4 (XC4VSX35-FFG668-10) FPGA. A conclusive objective of the subjective radio is to secure the best open reach through scholarly capacity and configurability as depicted some time as of late. Since a substantial segment of the extent is starting now doled out, the most fundamental test is to give the approved reach without interfering to the transmission of other approved customers as spoke to in Figure. The scholarly radio engages the usage of briefly unused reach, which is insinuated as extent hole or white space. If this band is further used by an approved customer, the scholarly radio moves to another extent opening or stays in the same band, changing its transmission power level or regulation arrangement to keep up a key separation from impedance as showed up in Figure. The mental radio thought was exhibited , where the essential focus was on the radio data representation lingo (RKRL) and how the subjective radio can enhance the flexibility of individual remote organizations.

II. FPGA CONCEPT

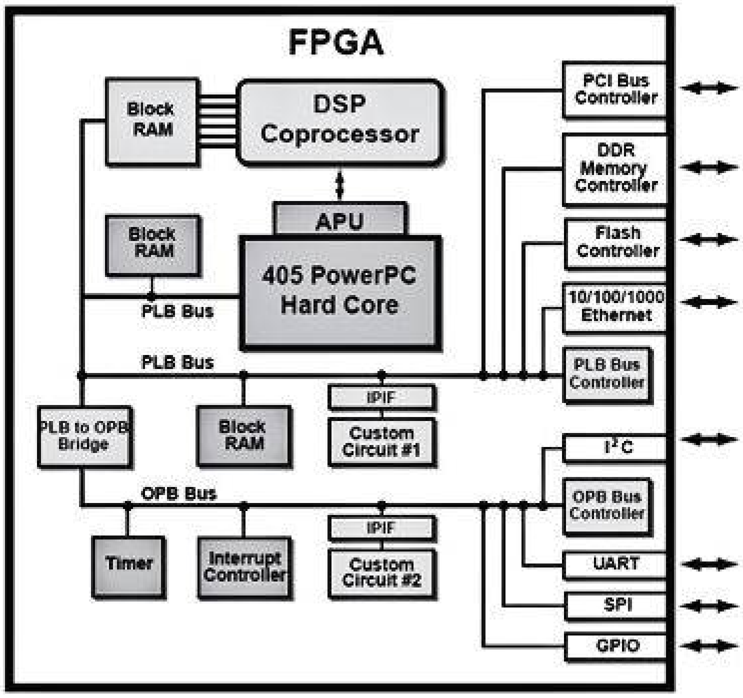


Fig No 01 Processor concept example

Every FPGA chip, or FPGA, is composed of a finite number of predefined resources with programmable interconnects. These interconnects implement the digital circuit you design with the LabVIEW FPGA Module. When you create an FPGA VI, you design a circuit schematic that describes how logic blocks are wired together on the FPGA.For the purpose of this example, the result of the architecture and processor evaluation is Xilinx’s XC4VFX20 component. This FPGA includes a 405 PowerPC processor, tri-mode Ethernet block, embedded memory and DSP slices. Our FPGA-based projected system requirements include a PCI bus interface, a 10/100 Ethernet connection, an external DDR memory controller for access to processor memory and an external Flash memory controller for access to stored program memory. Additionally, the system will support an I2C interface, an SPI interface, an RS-232 UART implementation and access to external switches and LEDs via GPIO signals. The system will also support a DSP function, and custom circuits. Figure 01illustrates the proposed system architecture. Xilinx’s system tool for implementing the embedded processor within the FPGA is the embedded development kit (EDK). EDK integrates the

DOI:10.15662/IJAREEIE.2015.0503140

system, hardware and software tools together into one package. By following the automated flow, an evaluation board may be used as a starting point for the project. The evaluation board chosen should include as many equivalent features as possible in common with the final target application. Availability of the right evaluation board can help reduce design schedule and risk. While it may not be possible to obtain an evaluation board with exactly the mix of peripherals and exact FPGA component desired, it should be possible to find a board with a similar part from the targeted FPGA device family. For this example, we will obtain a board with a XC4VFX12 component. Most evaluation boards include DDR memory, the 10/100 Ethernet PHY, dip-switches, LEDs and an RS-232 interface. The evaluation board should also support cable configuration and processor debug via a JTAG header. Once the evaluation board has been obtained, the EDK should be used to configure the evaluation board. This process involves stepping through the automated flow. An example automated project configuration flow follows:

1) Select a new project using the automated flow
2) Select the evaluation board that was obtained
3) Select the processor (for this example, the PowerPC processor will be selected)
4) Enable the processor core features (for this example, the processor core frequency will be 200 MHz, the bus frequency will be 100 MHz, cache-enabled, and a JTAG interface selected for debugging)
5) Select the device to be used. Big endian format is preferred for TCP/IP implementations
6) Device peripherals, addresses and modes of operation (for this example, DDR memory, Flash, Ethernet, and RS-232 are selected and configured)

III. DSP CONCEPT

1. What is DSP - Digital Signal Processing Design

DSP or Digital Signal Processing is the science of using computers to interpret digital patterns that exist everywhere in technology today. DSP applications have shaped the development of technology since the 1960s essential to the development of radar, sonar and space exploration. DSP applications analyze and interpret digital patterns or signals and mathematically manipulate them. Today, DSP is involved in both the implementation and growth throughout countless market applications including: oil exploration, medical imaging, audio and speech processing, digital image and video processing, mobile and telecommunications and much more. For scientists and engineers, DSP is a basic skill in many fields and almost every field that utilizes it has developed deep DSP technology with their own application specialized techniques, specific algorithms and mathematics.

2. DSP and FPGAs

Digital signal processing is one of the primary design applications for FPGAs in the market today. While FPGAs have been used in DSP applications for years, they have been gaining considerable traction in high performance applications because of their versatility and speed over traditional processors like microcontrollers. FPGAs provide tremendous computational throughput by using highly parallel architectures, and are hardware reconfigurable; allowing the designer to develop customized architectures for ideal implementation of their algorithms. Additionally, high performance DSP applications have required higher and higher levels of precision that FPGAs are capable of bringing to the table. Read more about Xilinx DSP Platforms

3. Xilinx DSP Solutions

Xilinx features DSP boards and kits for a variety of design and pricing needs. Within each product family Xilinx offers DSP-optimized sub-families that deliver the right balance of high-performance logic, serial connectivity and signal processing capabilities as part of the XtremeDSP™ DSP platform portfolio. The Xilinx DSP platform initiative helps you develop tailored high performance DSP solutions for aerospace and defense, medical, automotive, digital communications, multimedia, video, and imaging industries with new Spartan-6 FPGA DSP kit and the Virtex-6 FPGA DSP kit developed with Avnet, MathWorks, Texas Instruments and 4DSP. Xilinx DSP design products boast “Design Right out of the Box” including everything you need to speed up your design and development processes.

IV. KEY FEATURES OF SYSTEM GENERATOR

1. **DSP modelling-**
   Build and debug high-performance DSP systems in Simulink using the Xilinx Blockset that contains functions for signal processing (e.g., FIR filters, FFTs), error correction (e.g., Viterbi decoder, Reed-Solomon encoder/decoder), arithmetic, memories (e.g., FIFO, RAM, ROM), and digital logic.
2. **Bit and cycle accurate floating and fixed-point implementation**
   System Generator supports bit and cycle accurate fixed-point and bit and cycle accurate single, double and custom precision floating-point.
3. **Automatic code generation of VHDL or Verilog from Simulink**
   Implement behavioral (RTL) generation and target specific Xilinx IP cores from the Xilinx Blockset. Generate IP for quick importation into the Vivado IP Catalog for easy design reuse and model sharing.
4. **Hardware co-simulation**
   A code generation option that allows you to validate working hardware and accelerate simulations in Simulink and MATLAB. System Generator supports Ethernet (10/100/Gigabit) and JTAG communication between a hardware platform and Simulink.

V. OBJECTIVES

1. Providing example of advantages of high level design tool such as System Generator for FPGA design.
2. To study applications of XSG.

VI. RECENT TRENDS IN FPGA ARCHITECTURES AND APPLICATIONS

Today's FPGAs are entire programmable systems on a chip (SoC) which are able to cover an extremely wide range of applications. The main features of modern FPGAs are First described and compared with those of the past. In particular, the Altera Stratix III and Xilinx Virtex-5 families of devices, both using a 65 nm copper interconnect process,will be used as examples of contemporary FPGAs. FPGAs are composed from clusters of logic cells (LCs), interconnected via programmable routing resources [1]. The conjuration of the FPGA is stored on embedded static RAM within the chip, this controlling the contents of the LCs and multiplexers that perform routing. This basic architecture has not changed dramatically since their introduction in the 1980s. Early FPGAs used a logic cell consisting of a 4-input lookup table (LUT) and register. Present devices employ larger numbers of inputs (6-input for Virtex-5 and 7-input for Stratix III) and have other associated circuitry. Since area increases with the number of inputs but logic depth decreases, the trend for larger LUTs re_ect the increased interconnect to logic delay in modern integrated circuit (IC) technology. Clusters interconnect multiple LCs and serve to provide local routing within the cluster. The larger number of metal layers available in IC processes (12 layers in the case of Virtex-5) is rejected in better programmable interconnect density. Whereas early devices had issues with reusability, this is rarely the case in contemporary
devices. Embedded blocks are extensively used in FPGAs, serving to improve delay, power and area if utilized by the application, but waste area and power if unused. Early embedded blocks included fast carry chains, memories, phase locked loops, delay locked loops, boundary scan testing and multipliers. More recently, multipliers have been replaced by digital signal processing (DSP) blocks which add support for logical operations, shifting, addition, multiply-add, complex multiplication etc. Their generality has been improved through features such as multiple word length support and cascadability. As a result, they can implement primitives such as Filters, transforms and Floating point more efficiently. Direct support for most standard modern highspeedexternal memory standards, networking, communications buses, and IO are also included. Both embedded hard (e.g. PowerPC) and soft (e.g. NIOS II) processors for FPGAs
are available; hard processors appear to be on the decline, possibly due to their increased cost when not needed.

1. Platforms and Tools

Early research using FPGAs for large-scale computing applications used boards containing large arrays of FPGA devices [2]. A recent development is to use the multigigabit transceivers (MGT) available on FPGAs which allow boards to be interconnected via commodity network switches. An example is the Berkeley Emulation Engine 2 (BEE2) [3] illustrated in Figure 1. It is built around compute module boards which contain 5 Xilinx XC2VP70 FPGAs connected to 4 double data rate 2 (DDR2) dual inline memory modules (DIMMs), providing a maximum capacity of 4GB per FPGA. Each FPGA has two PowerPC 405 processor cores and a local mesh connects four of the FPGAsin a 2 dimensional compute grid via low-voltage CMOS(LVCMOS) parallel signaling. The remaining FPGA is used for control. Off-module communications are via 18 (2 from the control FPGA and 4 from each of the compute FPGAs) In_niband 4X channel bonded full duplex 2.5 Gbpslinks providing 180 Gbps off-module bandwidth. Modules can be interconnected in different topologies including tree, 3D mesh or crossbar and standard interfaces are used so standard In_niband and 10-Gigabit Ethernet switches can be used.
Finally, a 100 base-T Ethernet connection to the control FPGA is present for out-of-band communications, monitoring and control. Commercial machines with FPGA accelerators are also available. These include the Cray XD1, SRC SRC-7 and Silicon Graphics RASC blade. These all involve parallel organisations of high performance microprocessors tightly coupled to FPGA devices. In a manner similar to BEE2, nodes are interconnected via high speed switches and switching topologies can be altered via their con_guration. One obstacle to computing applications has been the problem of providing a means to supply an FPGA-based coprocessor with data. This has been addressed by companies such as XtremeData and Nallatech by providing front side bus interfaces which have greatly improved latency and bandwidth over standard peripheral buses such as PCI express. The new strategy involves replacing a CPU board on a multiprocessor motherboard with a pin-compatible FPGA board. As design complexity continues to increase, methods other than the traditional VHDL/Verilog register transfer level (RTL) approaches have become necessary. Tools such as Mentor Graphic's Catapult allow direct synthesis from C++ algorithms, greatly improving designer productivity. For DSP applications, tools to convert Simulink to synthesisableRTL have become mature, this path being particularly suitable for non-expert designers to build complex FPGA systems. Other new approaches include synthesis from SystemVerilog and SystemC.

2. Applications

FPGAs are suitable for an extremely diverse number of applications including: sorting and searching; signal processing; audio, video and image manipulation; cryptography; packet processing; random number generation and logic emulation. Important new markets are likely to include oil and gas exploration, financial engineering, bioinformatics, high definition video, software defined radio; automotive and mobile base stations. In this section, several examples of advanced applications of FPGA technology are presented. These examples are similar in that they are all large-scale applications requiring processing power surpassing processor and DSP-based solutions; production volume is limited; and their functionality is subject to change. These characteristics combine to make ASIC solutions less attractive.

VII. DESIGN METHODOLOGY OF SYSTEM GENERATOR

A transceiver can be designed using discrete electronic components. In general, the overall design is not flexible and highly dependent on technology and available devices, has long design time, occupies large area, has high power consumption and high delays and low maximum operating frequency. In general, the trend is to integrate the design in a digital integrated circuit and place around the necessary external components; this eliminates the previous inconveniences. It must be emphasized that these designs can be easily portable between devices, even from different manufacturers. This portability is possible because the design can be described with a standard hardware description language (HDL).

In digital systems, when floating point arithmetic is used, the range and precision can be adjusted with the number of bits of exponent and mantissa, it is then possible to obtain a wide range and high precision in this type of representation. However, floating point operations require many hardware resources and longtime execution (Hauck &DeHon, 2008). On the other hand, the fixed point arithmetic requires fewer hardware resources, but the range and precision can be improved only by increasing the number of bits. If the number of bits is constant, to increase the range causes a decrease in the precision. It is possible to use fixed point arithmetic in most applications when the range of signals is known or can be determined by statistical methods. In fixed point arithmetic the 2's complement representation is used because its arithmetic rules are simpler than the 1's complement representation. Ordinarily the systems can be designed using a standard hardware description language: VHDL (Very High Speed Integrated Circuit Hardware Description Language) (Pedroni, 2004) or Verilog (Palnitkar, 2003). Manual coded of complex systems using one of these languages is little flexible and has a great design time. To solve these problems several design programs have been developed. One of them is the System Generator from Xilinx, which is installed in Simulink. When System Generator is installed some Blocksets are included in Simulink of Matlab. Each block is configured after opening its dialog window, this permits fast and flexible designs. Basically, System Generator allows minimizing the time spent by the designer for the description and simulation of the circuit. On the other hand, the design is flexible; it is possible to change the design parameters and check quickly the effect on the performances and the architecture of the system. The functional simulation is possible even before the compilation of the model designed. The compilation generates the files of the structural description of the system in a standard hardware description language for the Integrated System Environment (ISE) for Xilinx FPGAs.

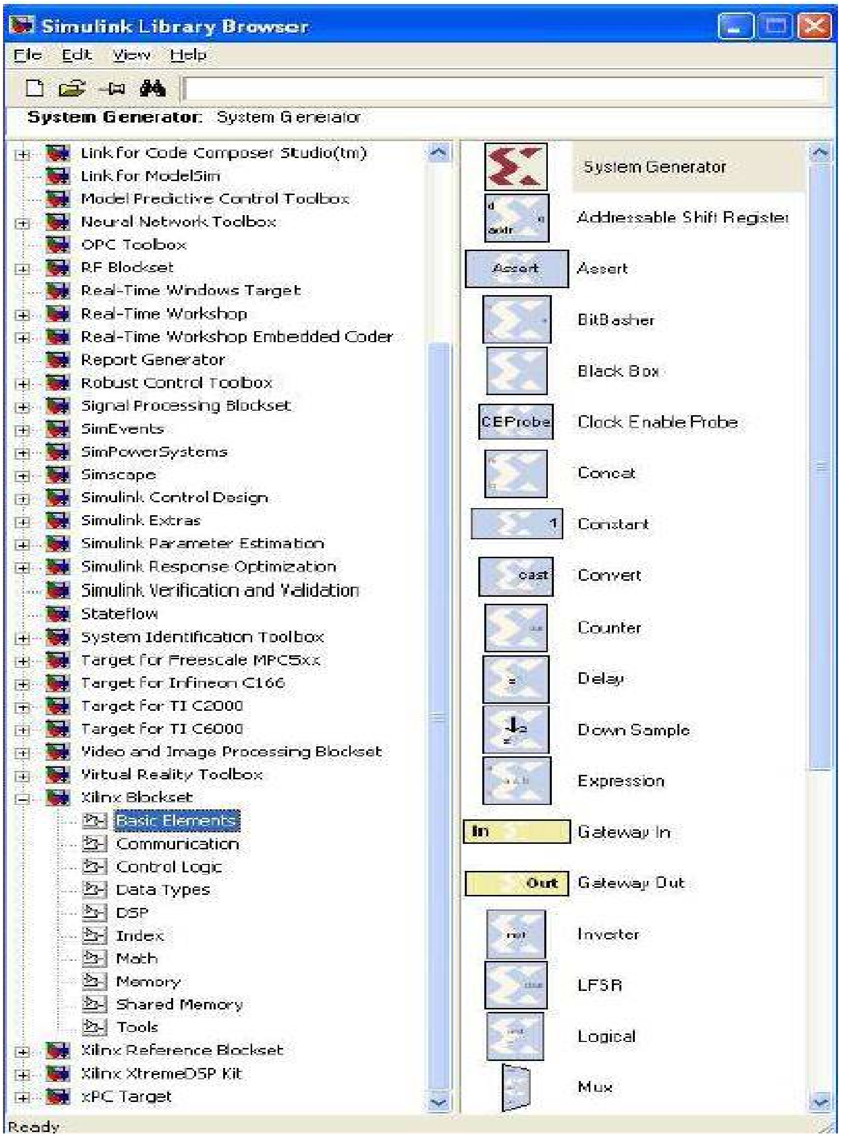


Fig. 2 . System Generator Blocksets in Simulink

The FPGA boundary in the Simulink model is defined by Gateway In and Gateway Out blocks. The Gateway In block converts the Simulink floating point input to a fixed point format, saturation and rounding modes can be defined by the designer. The Gateway Out block converts the FPGA fixed point format to Simulink double numerical precision floatingpoint format.In the System Generator the designer does not perceive the signals as bits; instead, the bits are grouped in signed or unsigned fixed point format. The operators force signals to change automatically to the appropriate format in the outputs. A block is not a hardware circuit necessarily; it relates with others blocks to generate the appropriate hardware. The designer can include blocks described in a hardware description language, finite state machine flow diagram, Matlab files, etc. The System Generator simulations are bit and cycle accurate, this means results seen in a simulation exactly match the results that are seen in hardware. The Simulink signals are shown as floating point values, which makes easier to interpret them. The System Generator simulations are faster than traditional hardware description language simulators, and the results are easier for analyzing. Otherwise, the VHDL and Verilog code are not portable to other FPGA manufacturers.

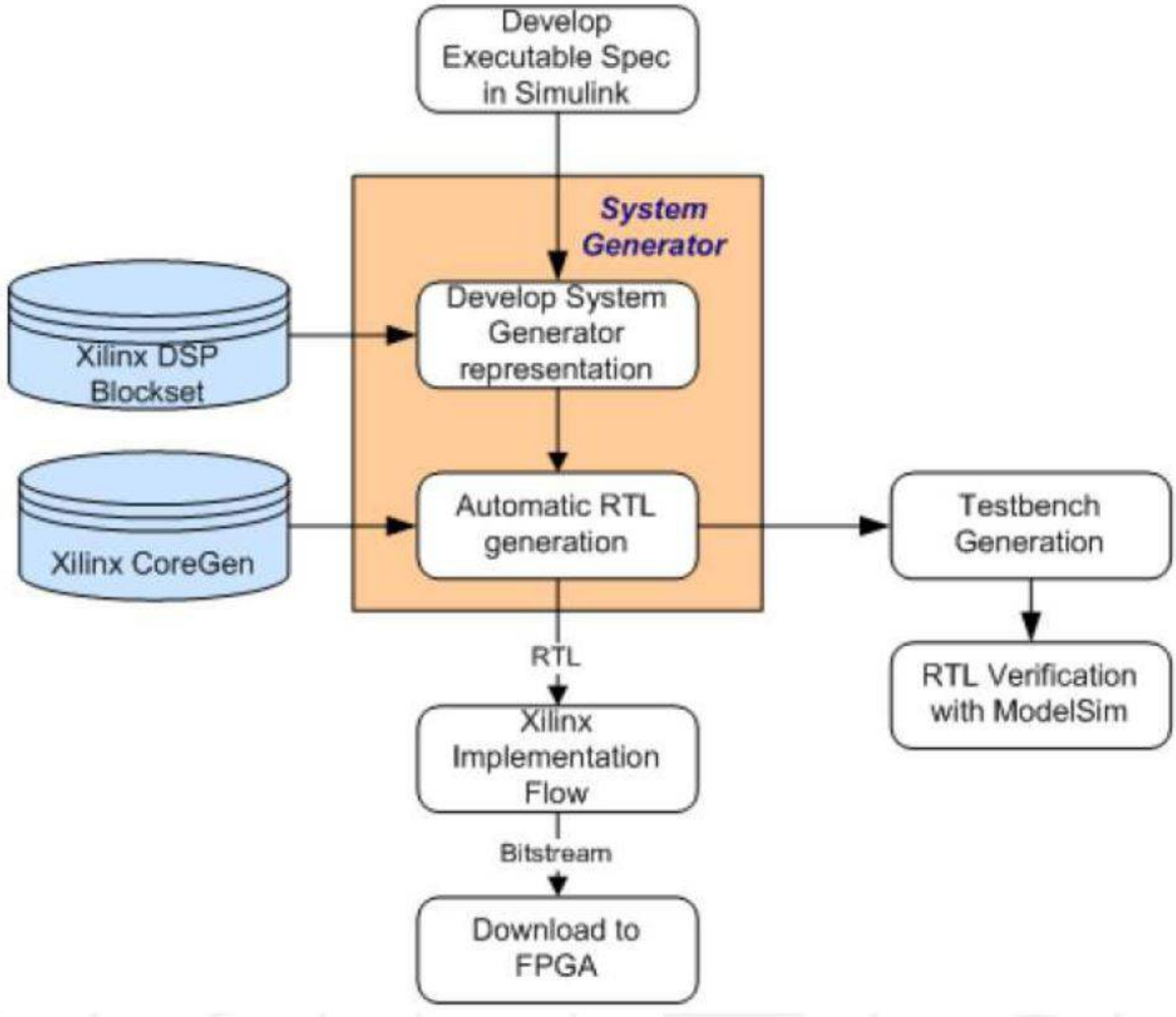

Develop Executable Spec in Simulink
System Generator
Xilinx DSP Blockset
Develop System Generator representation
Xilinx CoreGen
Automatic RTL generation
Testbench Generation
RTL Verification with ModelSim
RTL
Xilinx Implementation Flow
Bitstream
Download to FPGA

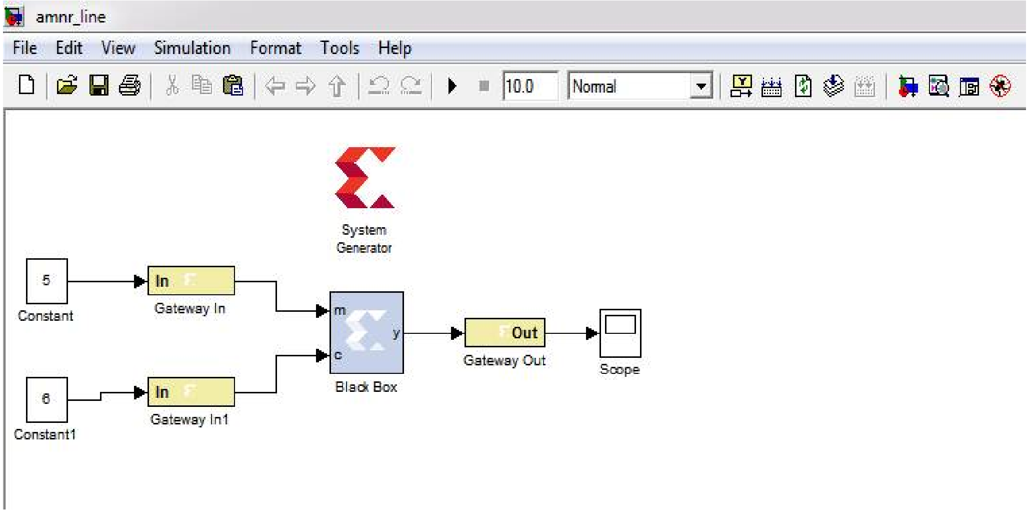
Fig. 3 . System Generator design flow

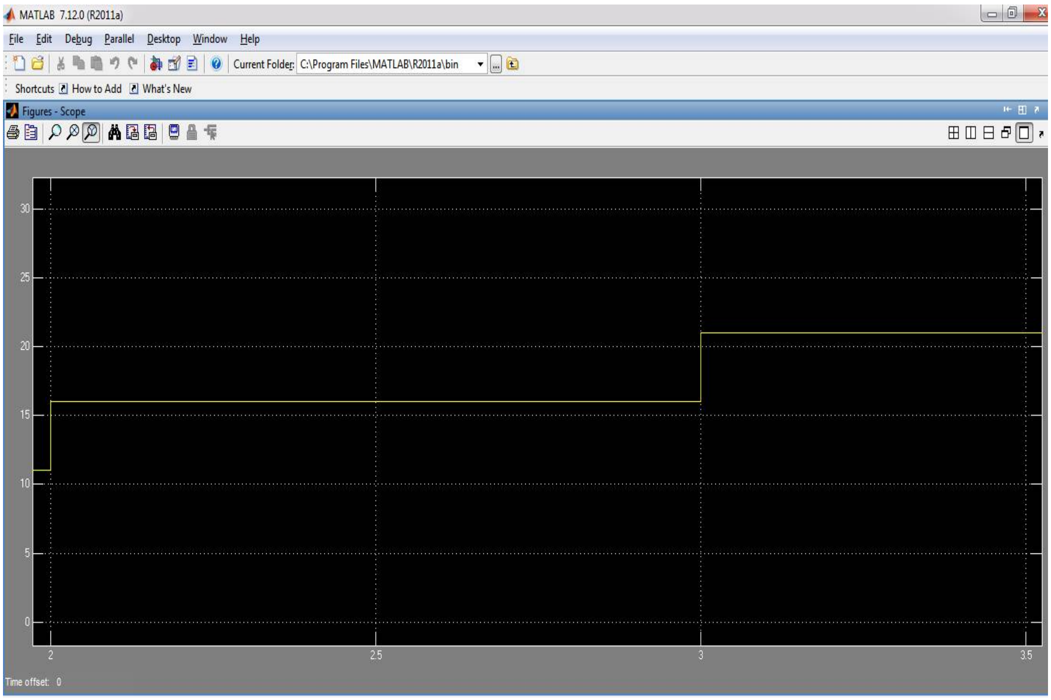
Every system designed with System Generator must contain a System Generator block this block specifies how simulation and code generator can be used. Firstly, the type of compilation in the System Generator block can be specified to obtain: HDL netlist, Bitstreamfor programming, etc. Secondly, the FPGA type can be chosen. The target directory defines where the compilation writes the files of Integrated System Environment project. The synthesis tool specifies which tool is chosen for synthesizing the circuit: Synplify, SynplifyPro or Xilinx Synthesis Tool (XST). In the hardware description language the designer can choose between VHDL and Verilog. Finally, clock options defines the period of the clock, its input pin location, the mode of multirate implementation and the Simulink system period, which is the greatest common divisor of the sample periods that appear in the system. In the block icon display, the type of information to be displayed is specified.

VIII. STATE OF ART

TITLE OF PAPER	ISSUE DISCUSSED	OUTCOMES	METHODOLOGY
FPGA implementation and performance study of spectrum sensing based on entropy estimation using cyclic features	Spectrum Sensing Technique	Detection of Signals	Implementing the Matlab, Simulink and Xilinx
Cyclostationary Signatures in Practical Cognitive Radio Applications (Paul D. Sutton, Member, IEEE, Keith E. Nolan, Member, IEEE, and Linda E. Doyle, Member, IEEE)	Cyclostationary Signal Analysis	The implementation of a full OFDM-based transceiver	Mathematical Analysis of Cyclostationary Detection Method
Practical Implementation of a Cognitive Radio System for Dynamic Spectrum Access	Implementation of Cognitive Transmitter and Receiver	Detection of A Signal	OFDM is used and GUI is developed
Simulink based spectrum Sensing (Avila.J,Thenmozhi. K)	Simulink Model Of Cyclostationary Detection Method	Simulink Models	Simulink Model Study
System Generator: The State-of-art FPGA Design tool for DSP Applications (Department of Electronics and Computer IIT, Rourkee)	Applications of FPGAs for DSP	The values of the percentage of area used and the placement speed are shown and compared	Methodology of designing the model for a useful communication technique using Xilinx System Generator (XSG)

IX. SIMULATION RESULTS




MATLAB 7.12.0 (R2011a)
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Program Files\MATLAB\R2011a\bin
Shortcuts How to Add What's New
Figures - Scope
30
25
20
15
10
5
0
2
2.5
3
3.5
Time offset: 0

X. CONCLUSION

Numerous analyses have been performed which substantiate the thought that FPGAs can beat DSPs and installed processors in sign preparing, while as yet being very vitality proficient. Xilinx System Generator is a framework level displaying device that encourages FPGA equipment outline by expanding Simulink/Matlab from multiple points of view keeping in mind the end goal to give an effective demonstrating environment.

REFERENCES

[1] System Generator: The State-of-art FPGA Design tool for DSPApplications (Sparsh Mittal, Saket Gupta, and S. Dasgupta- Department of Electronics and Computer Engineering, Indian Institute of Technology RoorkeeRoorkee, India)

[2] M. Cummings and S. Huruyama, “FPGA in the Software Radio,” IEEE Comm Magazine, volume: 37, no. 2, pp. 108-112, February1999.

[3] Xilinx Inc., Virtex-6 SXT for DSP and memory-intensive applications with low-power serial connectivity, http://www.xilinx.com/products/v6s6.htm. Last visited: April. 2011.

[4] A.J. Viterbi, “Convolutional codes and their performance in Communication systems,” IEEE Trans. Commun., vol. COM-19, pp 751-772, Oct., 1971.

[5] B. Sklar, “Digital Communication Fundamentals and Applications,” Prentice Hall, Englenwood Cliffs, New Jersey, 2000, Part 2 chapter 7.

[6] GSM 05:03: “Channel coding”, Version 8.9.0 Release 1999.

[7] Francois Chan, David Haccoun, “Adaptive Viterbi Decoding of Convolutional Codes over Memory less Channels” in IEEE TRANSACTIONS ON COMMUNICATIONS, VOL. 45, NO. 11, NOVEMBER 1997.

[8] SherifWelsen Shaker, SalwaHussienElramly, Khaled Ali Shehata, “Design and Implementation of Low-Power Viterbi Decoder for Software-Defined WiMAX Receiver” 17th Telecommunications forum TELFOR 2009.

[9] Russell Tessier, SriramSwaminathan, RamaswamyRamaswamy, Dennis Goeckel, and Wayne Burleson,” A Reconfigurable, Power-Efficient Adaptive Viterbi Decoder” IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 13, NO. 4, APRIL 2005.

[10] Juan Mauel Campos, Rene Cumplido, “a Run time reconfigurable architecture for Viterbi decoder” 3rd International Conference on Electrical and Electronics Engineering (ICEEE 2006).