International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 05, May-2015

Rapid Prototyping of Image Edge Detection and Display Sub System on FPGA using Soft Processor

Krishna Kishore Anumanchi
Student
Department of PG Studies
UTL Tech, VTU Extension Centre
Bangalore

K. V. Ramana Reddy
Assistant Professor
Department of PG Studies
UTL Tech, VTU Extension Centre
Bangalore

Dr. Siva Yellampalli
Prof & HOD
Department of PG Studies
UTL Tech, VTU Extension Centre
Bangalore

***Abstract*—— In this paper we introduce a faster system level prototyping approach for IP using Xilinx EDK flow approach. In addition to design under test this method uses set of pre-verified standard designs to validate IP of our interest. Proposed method utilizes Xilinx EDK to build system with microblaze processor on FPGA, and SDK to develop application on microblaze that controls the stimulus to IP.**

Keywords—Rapid FPGA Prototyping, Soft core processor, Embedded development kit

I. INTRODUCTION

FPGA prototyping is the traditional methodology to validate IP functionality on programmable silicon chip. As SoC's are becoming complex, the IP functionality also getting complex. To verify IP functionality for complex designs only software simulation is not sufficient as it is slow; to conquer this problem FPGA prototyping became solution [1]. For complex designs like image processing software simulation take more simulation cycles which can become costly. If we have setup where an IP can be easily integrated to a system which is a replica of actual system for which IP is designed for, the design confidence can be boosted quickly.

In this work we have developed a system on programmable chip to validate an algorithm for edge detection and integrated display controller to display edge detected output on VGA monitor.

Xilinx Platform Studio (XPS) is a fast system prototyping kit that can be used to architect complete verification platform on FPGA using a soft core processor as a controller of stimulus. Using XPS custom IP can be interfaced to processor through standard interface like PLB or AXI. Special function registers which controls the IP for different features can be exposed to user as software controllable registers. Once the IP is integrated and net list generated, it works at frequency that is actual frequency mentioned which is not possible with software simulation. This paper describes the idea of utilizing xilinx Embedded Development Kit (EDK) as fast prototyping as well as functional verification kit on FPGA.

II. CONVENTIONAL VALIDATION PLATFORM

Conventional FPGA validation platform contains a host PC which is the source for transaction to DUT through transactors connected to FPGA board. These transactors are either PCIe or AMBA compliant. The application code written on the host PC initiates transactions to IP through these transactors. In this approach extra glue logic or protocol conversion is required between IP and transactors. This kind of platform is time consuming as well as cost of the components also high. One more disadvantage of such implementation is integration of external board to host PC and debugging flexibility.

III. SOBEL EDGE DETECTION

The Sobel operator is widely used to detect edges of image, in the area of image and video processing, computer vision, robotics etc., the Sobel operator calculates the gradient of the image intensity at each point, giving the direction of the largest possible increase from light to dark and the rate of change in that direction. Mathematically, the gradient of a two-variable function (here the image intensity function) is at each image point a 2D vector with the components given by the derivatives in the horizontal vertical directions.

Sobel edge detection algorithm is implemented in hardware using Verilog. RTL is synthesized to the platform selected. To validate this algorithm it needs an image whose coefficients are stored in memory and controller attached which should co-ordinate the data transactions. This controller is platform dependent and may not be reusable.

IV. VGA CONTROLLER

The term VGA stands of Video Graphic Array. It is used to display images on the computer monitor. It generates timing and control for the monitor to display an image. Generally based on monitor's display resolution the controller adjusts itself to generate timing. In our implementation it is fixed to 640x480 @ 60 Hz, based on this information remaining signals generated.

V. PROPOSED METHOD AND IMPLEMENTATION

To operate an IP in real time it needs a master to initiate and control the stimulus. As described in conventional validation platform technique, IP needs a processor to generate input and a bus to transport that input.

Figure 5.1 describes detailed top level architecture of the method implemented. RTL for Sobel edge detector and VGA controller are coded independently. The system shown in Figure 5.1 is built using Xilinx Platform Studio (XPS). XPS provides a simple way to integrate user logic to microblaze through PLB bus.

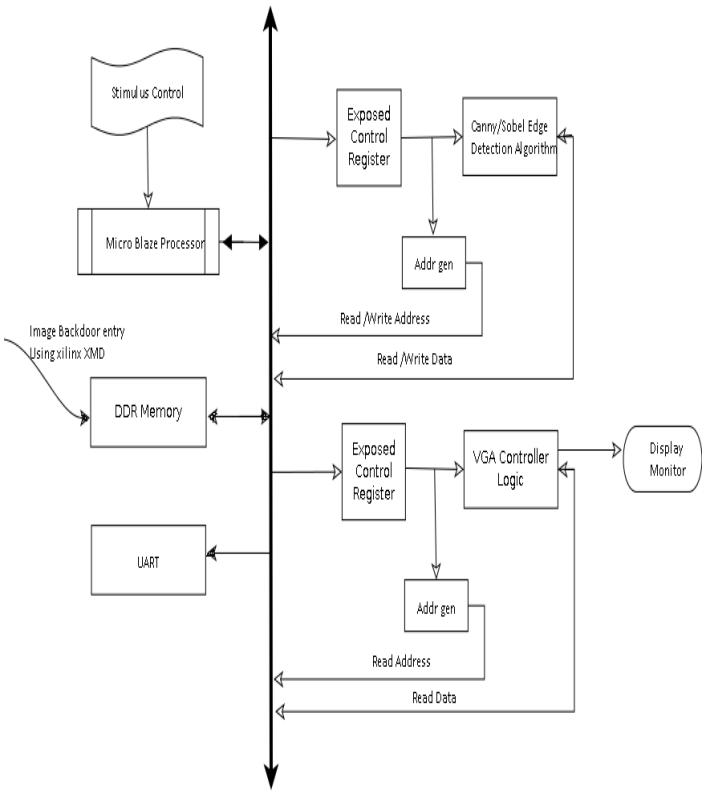

Fig 5.1: Proposed method

Edge detection and VGA controller IP's have their own control registers exposed to application through software accessible registers. The DDR memory connected to the processor through PLB bus is loaded with coefficient image data through back door entry using XMD debugger. Once the image is loaded the processing IP is enabled by application program written on Microblaze in C by writing control bit to 1. Once the control bit is set the read address generation logic generates read address and transactions to DDR memory resulting a data fetch to processing IP. After processing every row of data the output is written back to memory at different base address. After completing edge detection process on the coefficient data the process completion bit in control register is asserted. Once the process completion interrupt enabled, the memory is ready with processed image data. The control register of the VGA display controller IP is set to 1, which initiates the read transactions from memory and the read data is displayed on the monitor. The address generation logic for both write and read are attached to both the IP's.

The directory structure followed in this architecture is shown in fig 5.2. Directory structure makes the environment reusable as the individual cores not tightly coupled. Every user design connected is memory mapped to the processor bus, and every design can be allocated with user defined software accessible registers. These software accessible registers can be used for debugging purposes. As the entire system including processor is implemented on same FPGA, attaining observability through Chip-scope and controllability through XMD debugger is relatively comfortable.

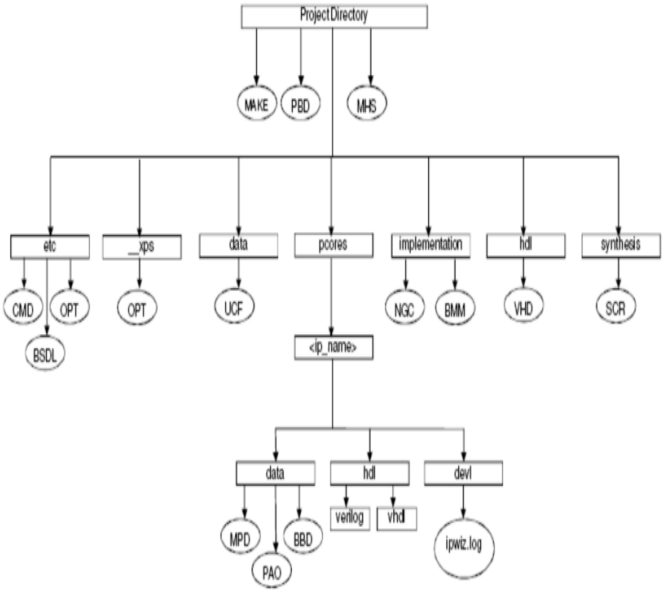

Fig 5.2: Directory structure

Directory structure plays a good role to make this method more flexible. The design of interest is placed in the hierarchy of "pcore/ip_name/hdl/". As the interface is standard, if the design logic changes, simply changing the intended file solves the problem.

VI. EXPERIMENTAL RESULTS

The custom IP is implemented in Verilog HDL, and complete system is implemented using Xilinx Platform Studio. Application development for stimulus driving is done through C- on eclipse IDE. Fig 6.1 shows the edge detected output of a gray scale input image. Input image coefficients are stored in the memory through backdoor process using XMD. The control registers for both Sobel and VGA are operated using application for back to back images. XMD is used for debugging.

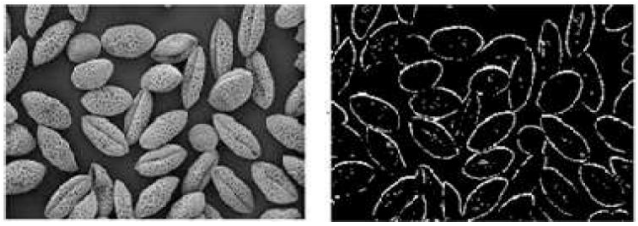

Fig 6.1 Edge detected output

VII. CONCLUSION

In this paper we have integrated a Sobel edge detection and VGA controller to the microblaze processor and synthesized entire system and invoked algorithm through application running on the processor. With the experiments conducted it is concluded the rapid prototyping of any IP can be done on FPGA using Xilinx EDK and can be extended to fast demo purposes or proof of concepts. The same setup can be extended with small glue logic to estimate the performance of the given design by capturing the active period of the design; this is to prove the re-usability of the proposed setup. Complete work is carried out on Spartan-3E FPGA starter kit.

VIII. REFERENCES

(1) M. Wageeh, A. Wahba, A. Salem, and M. Sheirah."FPGA Based Accelerator for Functional Simulation". Circuits and Systems, 2004. ISCAS '04. Proceedings of the 2004 International Symposium on, 5:V-317-V-320 Vol. 5, May 2004.
(2) M. Karimibiuki, K. Balston, A. Hu, and A. Ivanov, "Post-silicon code coverage evaluation with reduced area overhead for functional verification of SoC," in High Level Design Validation and Test Workshop (HLDVT), Nov 2011, pp. 92-97.
(3) Kenney, J. "Using a processor-driven testbench for functional verification of embedded SoCs". Embedded.com. 4
(4) Klein, R.; Piekarz, T., "Accelerating functional simulation for processor based designs". System-on-Chip for Real-Time Applications, 2005. Proceedings. Fifth International Workshop on Real Time Systems. 2005.
(5) C.Lavin, M. Padilla, P. Lundrigan, B. Nelson, and B. Hutchings. Rapid Prototyping Tools for FPGA Designs: RapidSmith. In Field-Programmable Technology. IEEE, Dec. 2010.
(6) K. Kepa, F. Morgan, K. Kósciuszkiewicz, L. Braun, M. Ḣubner, and J. Becker, "FPGA Analysis Tool: High-Level Flows for Low-Level Design Analysis in Reconfigurable Computing, " in ARC'09: Proceedings of the 5th International Workshop on Reconfigurable Computing: Architectures, Tools and Applications. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 62-73.
(7) R. Tessier, "Fast Placement Approaches for FPGAs, " ACM Trans. Des. Autom. Electron. Syst., vol. 7, no. 2, pp. 284-305, 2002.
(8) E. L. Horta and J. W. Lockwood, "Automated Method to Generate Bitstream Intellectual Property Cores for Virtex FPGAs, " in Proc. Field Programmable Logic.2004, 2004.
(9) C. Lavin, M. Padilla, S. Ghosh, B. Nelson, B. Hutchings, and M. Wirthlin, "Using Hard Macros to Reduce FPGA Compilation Time, " in Proceedings of the 20th International Workshop on Field-Programmable Logic and Applications, 2010, to appear.
(10) Eclipse.org. Available in: <www.eclipse.org> Accessed in: 18
(11) Yiannacouras, P.; Steffan, J.G.; Rose, J. "Exploration and Customization of FPGA-Based Soft Processors", Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on,On page(s): 266 - 277 Volume: 26, Issue: 2, Feb. 2007
(12) I. Kuon and J. Rose "Measuring the gap between FPGAs and ASICs", Proc. FPGA, pp.21 -30 2006
(13) P. Yiannacouras , J. G. Steffan and J. Rose "Application-specific customization of soft processor microarchitecture", Proc. Int. Symp. FPGA, pp.201 -210 2006
(14) R.D. Wittig, and P. Chow, "OneChip: An FPGA Processor with Reconfigurable Logic", IEEE Symposium on FPGAs for Custom Computing Machines, CS Press, Napa, CA, 1996, pp. 126-135.
(15) I. Yasri, N. H. Hamid, V. V. Yap, 2008. "Performance Analysis of FPGA Based Sobel Edge Detection Operator" IEEE.
(16) Input/Output Peripheral Devices Control Through Serial Through Serial Communication Using MicroBlaze Processor [IEEE International Conference on Devices Circuits and Systems, in Karunya University During 15-16th March 2012].