

DESIGN AND IMPLEMENTATION OF SCAN REGISTER INSERTION USING FPGA FOR HIGH EFFICIENCY

* N VANI, **G SAMATHA

*MTEch student, Dept of ECE, JBIET, Hyderabad, TS, India.

** Associate Professor & HOD, Dept of ECE, JBIET, Hyderabad, TS, India.

Abstract - Recent increase in the complexity of the circuits has brought high-level synthesis tools as a must in the digital circuit design. Scan flip-flop insertion for aiding design for testability invites additional hardware overhead, thereby deteriorating the performance of the circuit. In this paper, we shall demonstrate a novel FPGA based implementation of inserting scan registers in commonly used Finite State Machines and pipelined data path circuits with no hardware overhead or compromise in performance. All our proposed designs have been realized using a relatively low-level design methodology involving target FPGA family based primitive instantiation, coupled with their constrained placement on the Xilinx FPGA fabric. Implementation results clearly reveal the superiority of our proposed architectures in comparison to equivalent circuits derived through behavioural modeling with respect to area and speed. Additionally, our proposed scan register inserted circuits compare favourably with circuits designed without the scan flip-flops. Coupled with this, lies the ease of an automated generation of the corresponding Hardware Description Language (HDL) and placement constraints and their portability among other advanced FPGA families from Xilinx

Keywords: FPGA, HDL, Low level design, Flip flop, Xilinx, pipelined data.

I. INTRODUCTION

Application development tends to pack more features per product. In order to cope with competition, added features usually employ complex algorithms, making full use of existing processing power. When application performance is poor, one may envision accelerating the whole application or a computationally demanding kernel using the following solutions: multi-core microprocessors: may not accelerate non-standard computations (exponential, logarithm, square-root) and performance suffers when implementing low-grain parallelism due to inter process communication application-specific integrated circuits (ASICs): the price tag is often too big, Field Programmable Gate Arrays (FPGAs): provide a trade-off between the performances of ASICs and the costs of microprocessors. A Typical reconfigurable computing application is made up of the hardware mapped on the FPGA device(s) present on a coprocessor board and the software, which runs on the general purpose processor. Debugging of these applications involves debugging of both

the hardware and software components. Hardware simulation is one of the most widely used Techniques for hardware debugging and validation. It allows the designer to examine the circuit in detail, but can be prohibitively slow. Large designs can take anywhere between a few hours to a few days for the complete simulation. Design for testability (DFT) is an essential prerequisite for enhancing the controllability and observability of a circuit. This often involves replacing the normal flip-flops (FFs) by scan FFs, which includes addition of a multiplexer at the input of the normal FFs and the select line decides between the normal mode and the test or scan mode of operation. In the normal mode, the circuit operates in accordance to the specified functionality, while in the test or scan mode, the series of FFs are converted into a shift register, through which any desired input bit sequence can be shifted-in serially through a dedicated scan in pin, or the entire state of the circuit can be read out by shifting out the FF contents through a dedicated scan out pin. Such circuit modifications invite additional hardware and increase the critical path delay owing to introduction of multiplexers. Scan chain insertion on FPGAs, which resulted in resource overhead.

II. RELATED STUDY

However, for certain class of circuits, the limitations imposed through the additional circuitry can be mitigated when such scan FF inserted circuits are deployed on modern FPGA families, which support optimum logic resources. In order to reap the benefits of the advanced FPGA architectures, it is not sufficient to enter a behavioural or Register Transfer Level (RTL) mode of Hardware Description Language (HDL). At the design entry stage of the FPGA design flow. This is because the logic synthesis heuristics used by the CAD tools, often explore a narrow design space close to the architectural description given as input at the design entry level, and are often unable to perform the requisite algebraic factoring, sub-expression sharing or apply the appropriate logic identities to realize an efficient technology mapped circuit. A possible reason might be that the CAD tools cater to broader domainof applications, and individual attention to the fine nuances of a well-crafted design may not always be feasible. For an optimized FPGA based implementation, it is desirable that the designer possesses the requisite knowledge of the FPGA slice architectures of the target family, and reconstructs the

Boolean logic of the user design into suitable expressions, such that the technology mapping phase generates an efficient, optimized net list. Target FPGA specific primitive instantiation is a recommended approach for design optimization as it directly configures the slice logic of the FPGA.

Iterative and modular circuits realized using the bit-sliced design paradigm are often the ideal candidates for such a design approach. The portability of such designs across related FPGA families from the same vendor supporting the same design elements is often feasible, with almost zero or minor tweaks at the HDL level. Modern day Xilinx FPGAs support six input Look-Up Tables (LUTs), with dual outputs. Often for a specific design, a LUT based implementation results in underutilization of the configured LUTs, where all the six inputs are not used. Additionally, the dual output functionality may also not get inferred at places where such realization was feasible, thereby resulting in more hardware overhead and delay. In this paper, we have targeted such configured, yet underutilized LUTs, to add extra functionality into the design without disturbing the original architecture. The added feature is the Design for Testability (DFT), where the multiplexing arrangement necessary to realize scan FFs, has been accomplished by increasing the utilization ratio of the LUTs and carry chains that realized the original design.

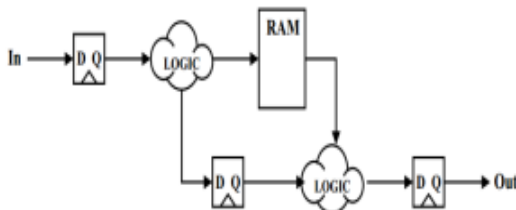


Fig.2.1. Basic model.

III. AN OVERVIEW OF PROPOSED SYSTEM

Binary counter, which serves as a fundamental component of many control path implementations, should possess the following desirable features of respectability, load ability; bidirectionality, count-enability, and terminal count detect ability. Another important advantage of fine data dependency analysis is that one can detect and parallelize codes that standard techniques (like the ones used in most HSL tools) cannot. Detecting the parallelism is mandatory but not sufficient to improve performances. One should take into consideration the deployment platform on which the algorithm will run. In our case we use FPGAs that have an advantage compared to most multi core processing systems that one can use fast dedicated lines to communicate between processing elements. In order to ensure a correct computation, we use fine data dependency analysis techniques together with advanced code transformation techniques.

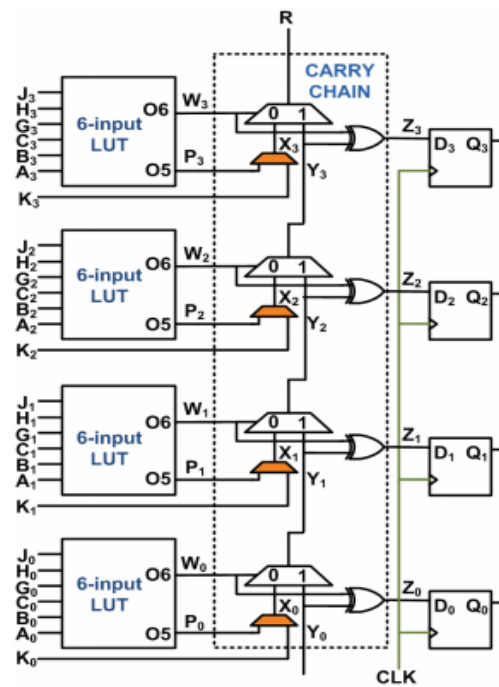


Fig.3.1. A simplified Virtex-7 slice architecture.

The shaded multiplexers inside the convey chain demonstrate that they can't be physically designed or instantiated, just the CAD device can arrange them as per the HDL. The last yield of the convey chain R of a solitary cut can be communicated as a course of four 2:1 multiplexer as pursues:

$$R = \overline{W}_3X_3 + W_3[\overline{W}_2X_2 + W_2(\overline{W}_1Y_1 + W_1\{\overline{W}_0X_0 + W_0Y_0\})] \tag{1}$$

In this architecture, the counter operates in two different modes: the normal mode and the scan mode. In the normal mode of operation, the counter operates as per the specified functionality. In the test mode, the serial data is fed into the carry chain through a multiplexing arrangement as shown in Fig, and the carry chain fabric is configured to connect all the FFs in a serial-input serial-output (SISO) mode via the LUTs, without disturbing the parallel read-out capability of the counter.

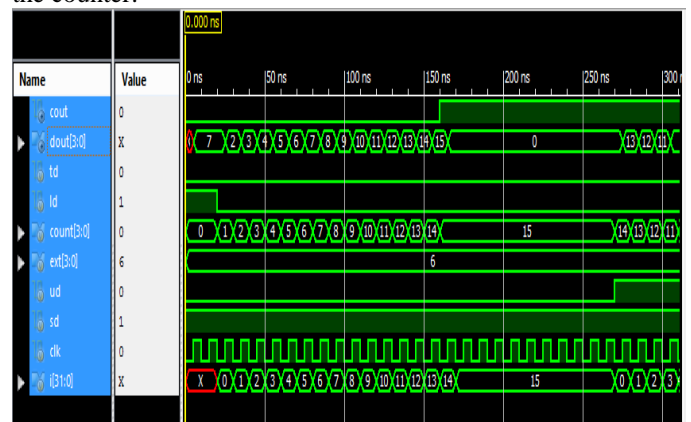


Fig.3.2. Bidirectional converter.

They are set at the contributions, after the pre-snake and after the increase. A blessed symptom is this can spare registers in a plan, utilizing the DSP's bock inward registers rather than those in the rationale texture. Intel gadgets indicate their DSP hinders as Variable Precision DSP Blocks, a schematic outline of which is introduced in Figure 2.8. Intel DSP squares are coarser than the Xilinx ones. They can execute two 18×19 bit marked duplications, or one 27×27 bit increases. Past ages of gadgets had much progressively flexible squares, being equipped for executing three 9×9 bit duplications, two 18×18 bit marked increases, two 16×16 bit augmentations, or one 27×27 piece duplication, or a 36×36 piece duplication utilizing two squares. It merits referencing that the arrangement 10 of gadgets presents coordinate help for skimming point tasks in their DSP squares. A DSP can be utilized for executing one single-exactness expansion or increase, and a few squares can be utilized related to actualize twofold accuracy activities.

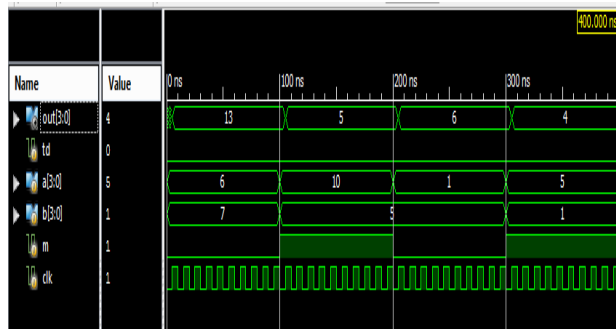


Fig.3.3. pipelined adder subtractor

IV. CONCLUSION

We have shown first class, motorized FPGA structures of calculating focuses with yield FFs, following the standard of unrefined instantiation and obliged position. The logic ideally suits to circuits where the organized reason parts are underutilized, or the possibility of the circuit in itself permits certain structure express changes, for instance, reshuffling of information sources or need encoding for expansion of breadth FFs with no hardware overhead. No proportion of changes in the decision settings for any association or enhancement target and effort for the lead designs can facilitate to our proposed plan, both to the extent zone and speed.

V. REFERENCES

- [1]. M. L. Bushnell and V. D. Agrawal, Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits. Kluwer Academic Publishers, 2000.
- [2]. A. Tiwari and K. A. Tomko, "Scan-chain Based Watch-points for Efficient Run-Time Debugging and Verification of FPGA Designs," in ASP-DAC, 2003, pp. 705-711.
- [3]. T. Wheeler, P. Graham, B. Nelson, and B. Hutchings, "Using Design-Level Scan to Improve FPGA Design Observability and Controllability for Functional Verification," in Field-Programmable Logic and Applications, ser. LNCS, vol. 2147, 2001, pp. 483-492.

- [4]. A. K. Verma, P. Brisk, and J. P. Ienne, "Challenges in Automatic Optimization of Arithmetic Circuits," in 19th IEEE International Symposium on Computer Arithmetic, June 2009, pp. 213-218.
- [5]. A. Ehliar, "Optimizing Xilinx designs through primitive instantiation," in Proceedings of the 7th FPGA World Conference, 2010, pp. 20-27.
- [6]. A. Palchoudhuri and R. S. Chakraborty, High Performance Integer Arithmetic Circuit Design on FPGA: Architecture, Implementation and Design Automation. Springer India, 2015.
- [7]. M. R. Stan, A. F. Tenca, and M. D. Ercegovac, "Long and Fast Up/Down Counters," IEEE Transactions on Computers, vol. 47, no. 7, pp. 722-735, July 1998.
- [8]. U. Meyer-Baese, "Digital Signal Processing with Field Programmable Gate Arrays", 3rd ed. Springer Series on Signals and Communication Technology, 2007.