International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Low Power FPGA Architecture

Abhijeet Khandale[1], Dr. H R Bhagyalakshmi[2]

M.Tech Student, Department of ECE, BMS College of Engineering, Bangalore, India

Associate professor, Department of ECE, BMS College of Engineering, Bangalore, India

Abstract: ***A comprehensive analysis and implementation of FPGA architecture for low routing power and clock gated CLBs has been presented in this paper. The power consumption in FPGAs is more in routing and in clock network. As the FPGA has thousands of logic blocks and hard embedded micros spread across the FPGA chip, more numbers of routing lines and switch boxes are required. Also the clock network is built with same routing resources. The Configurable logic blocks with clock gating will allow reducing the dynamic power. The logical equivalence of CLB inputs will help to reduce the routing congestion and also improve the timing of the design.***

Keywords: FPGA,VTR, clock gating, CLB

1. Introduction

Field Programmable Gate Array (FPGA) is the most popular reconfigurable computing technology, which is an ideal for various applications. Field Programmable Gate Array (FPGA) technology became a viable target for implementation of reconfigurable designs. FPGAs generally consist of a system with configurable logic blocks consist of LUTs, flip-flops and hard embedded blocks like RAM, DSP block, arithmetic blocks like multiplier, all placed in the vast array of interconnects. The FPGA can be reconfigured to a particular logic circuit using hardware description language like VHDL, Verilog and system Verilog. The FPGA architecture allows large variety of logic designs for real time application.

The FPGA architecture needs to be modified for higher performance and low power consumption. The power dissipation happens more in the routing and clock networks, designing FPGA such a way that less congestion will occur in the routing. Such experiment of FPGA architecture can be carried out using open source CAD like VTR or QFLOW

The understanding of new programmable architectures, and the development of new algorithms required to synthesize designs into FPGAs requires a complex software flow that allows experimentation.

In this paper we describe the modified FPGA architecture which has impact on power. Section A describes the FPGA architectural challenge which is used to analyze the requirements of the new FPGA architecture and algorithm implementation. Section B describes the necessity of the VPR tool which performs logic optimization and technology mapping on the soft-logic portion of the BLIF and using this synthesized BLIF it performs physical synthesis and power analysis. The later on sections give the modified architecture of FPGA and its advantages. Section III provides the results of the modified FPGA architecture. Section IV describes the conclusion and future work.

2. Design Requirement and Implementation

A. FPGA Architectural challenges

Field-Programmable Gate Arrays (FPGAs) are one of the most promising devices for digital circuit implementation media over the past decade. The most important part of their creation is their architecture, which improves their programmable logic functionality and their programmable interconnect. FPGA architecture has a crucial effect on the quality and the performance of the final device's speed performance, power consumption, and is efficient. The challenging areas of FPGA are Logic Block architecture, routing architecture, Input/output architecture and capabilities [1].

B. VPR

The Verilog-to-Routing (VTR) is an open source FPGA CAD tool which provides a complete, open-source framework for the implementation of FPGA architecture and CAD research and development [2]. The software flow of the tool starts with a Verilog hardware description of digital circuits, and a file describing the target FPGA architecture, and elaborates, synthesizes, packs, places and routes the circuit, and performs timing analysis on the result. The study of advance FPGA architectures and algorithms might be a difficult task, because of the effort required to conduct quality experiments. A good FPGA architecture or algorithm experiment requires a good benchmark designs, advanced architectures, and CAD tools that can efficiently map those designs to the architectures [3]. The VTR open source tool enables such experiments by providing a new FPGA architectural language with a flexible and robust CAD flow for FPGAs [4].

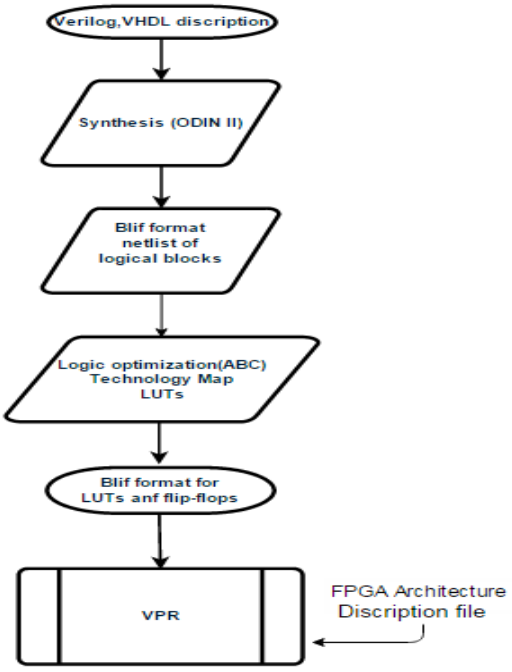

Figure 1: VPR Flow

C. LUT Architecture with fully populated crossbar

The CLB architecture has many LUTs inside it which has inputs corresponding to each LUT. These input connections made configurable to the LUTs. The below figure shows the overall CLB architecture with LUTs [5].

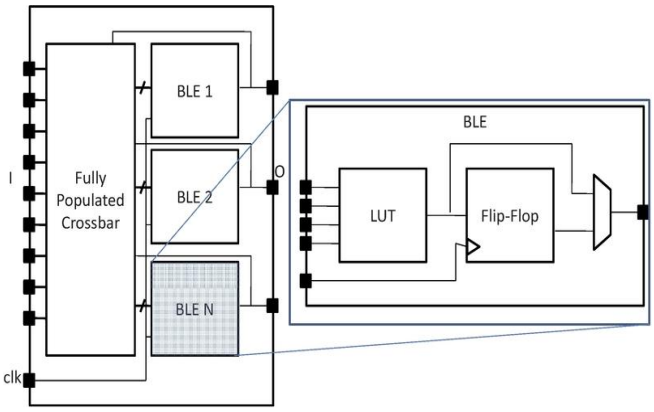

Figure 2: LUT with populated crossbar

The internal structure of the crossbar is as shown in below diagram.

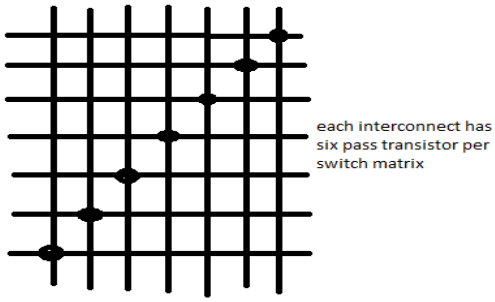

Figure 3: Populated cross bar structure

The advantage of the programmable interconnection between LUT and inputs is the routing congestiongets reduced. The configurable LUT inputs reduce the long paths being routed. This is implemented in VPR using FPGA architectural language. We need to apply logical equivalence to the inputs and outputs so that tool will understand that connections to those pins can be swapped without changing its functionality.

<pb_type name="clb">
<input name="I" num_pins="XX" equivalent="true"/>
<output name="O" num_pins="XX" equivalent="true"/>

D. Clock gating to CLBs

The clock network power consumption is around 30% in the FPGA total dynamic power consumption. As the clock pin in CLBs could not make as logical equivalence, the best way to put hardcoded clock gating options so that clock power consumption can be reduced. The following diagram shows the clock gating LUTs [5].

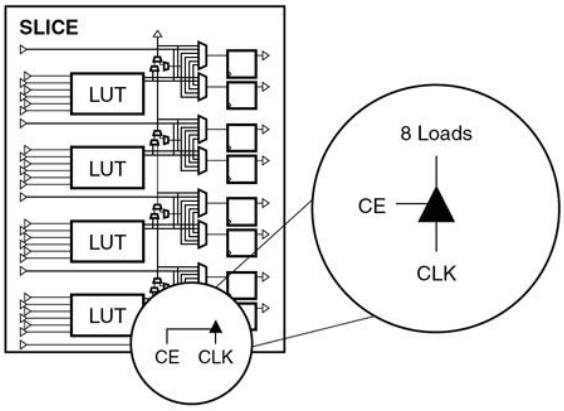

Figure 4: LUT clock gating buffer

The clock enable signal produced is allowed changing in the non-active part of the clock. This can be achieved by introducing latch having opposite sense of the clock edge. The flowing diagram shows the clock gating.

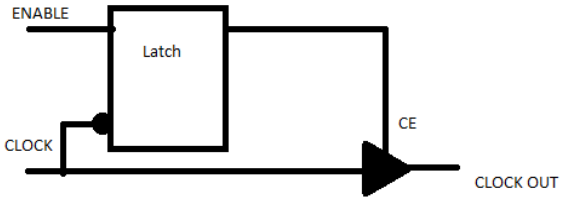

Figure 5: Glitch free Clock gating

This clock gating with LUT is implemented in the VPR which is discribed in the follwing architectural language [5].

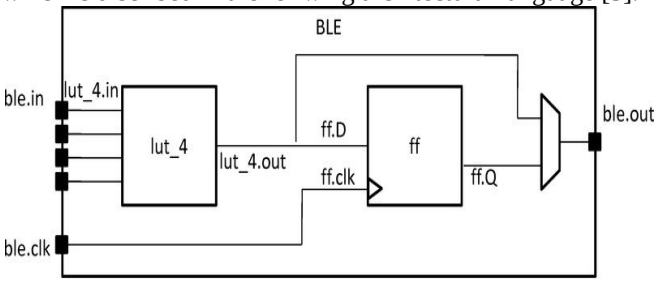

Figure 6: 4 input LUT

```
<pb_type name="ff" blif_model=".clockgate>
<interconnect>
<direct input="clk" output="latch.D"/>
<direct input="clk" output="buff_in"/>
<direct input="enable" output="latch.D"/>
<direct input="latch.Q" output="buff_CE"/>
<direct input="clock_out" output="ble.clk"/>
<direct input="lut_4.out" output="ff.D"/>
<direct input="ble.in" output="lut_4.in"/>
<mux input="ff.Q lut_4.out" output="ble.out"/>
<direct input="ble.clk" output="ff.clk"/>
</interconnect>
</pb_type>
```

E. LUT Architecture for efficiency and performance

A bigger LUTs can be implemented from smaller LUTs and one or more multiplexers. Similarly a 5-LUT can be built from two 4-LUTs and a multiplexer, while a 6-LUT can be built with two 5-LUTs and a multiplexer. The problem with the smaller LUT architecture is that logic circuit built from it are inefficient and result in unused resources when implementing smaller functions. There is anotherimportantissue is the replication of routing to the smaller LUTs when building a larger LUT and the creation of extra delays between LUTs which results in a non-optimized logic structure [6].

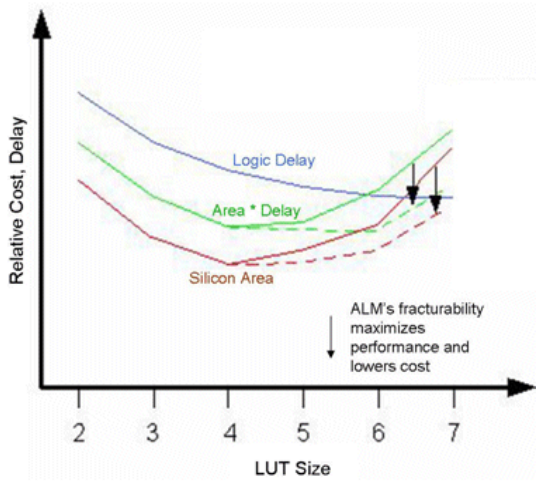

Figure 7: Cost-logic delay tradeoff with varying LUT sizes

The above graph shows the delay and area of the LUTs with different sizes and as shown in above figure the minimun area is for 4 LUT architecture and as 5-LUt and 6-LUt cand be implemented using 4-LUT area wise and routing wise it is efficient to use 4-LUt for FPGA architecture.

F. Timing Driven Routing and Placement

VPR supports the placement and routing of the design driven by the timing. The constraints are written in the standard SDC format.

```
create_clock -period 3 -waveform {1.25 2.75} clk
create_clock -period 2 clk2
create_clock -period 1 -name input_clk
create_clock -period 0 -name output_clk
set_clock_groups -exclusive -group input_clk -group clk2
set_output_delay -clock output_clk -max 1[get_ports {out*}]
```

3. Implementation Result

The routing power of the design reduced due to populated cross bar FPGA CLB architecture. Also the hardcoded clock gating helps to reduce the clock network dynamic power consumption and 4-LUT architecture is more efficient with respect to area and delay. The timing of overall design was improved.

Table 1: Device:3s500efg320-4 utilization

No of CLBs	*FPGA architecture without clock gating and Populated crossbar*	*FPGA architecture with clock gating and Populated crossbar*	*Timing*
4930	*58.77%*	*56.43%*	*322.32MHz*
4342	*33.48%*	*27.22%*	*345.72MHz*

4. Conclusion

FPGA are the most promising devices for the modern digital design. The architecture plays crucial role in the reprogrammable devices. It impacts the timing, Speed and power performance of the design. Hardcoded clock gating helps to improve the dynamic clock power consumption. The 4-LUT is the best option for the FPGAs due its less power and area consumption with good efficiency to implement digital design with large number of variables.

The IO pin configuration can also be made logically equivalence to reduce the routing congestion. Also the heterogeneous routing wire architecture, pipe lined architecture and LUTs input/output pin rearrangement techniques can be used to improve the FPGA performance [7].

5. Acknowledgment

This research is supported by the BMS College of Engineering, Bangalore. The authors wish to thank BMS college of Engineering for supporting this work by encouraging and supplying thenecessary tools.

References

[1] I. Kuon, R. Tessier, and J. Rose, \FPGA Architecture: Survey andChallenges," Foundations and Trends in Electronic Design Automation,vol. 2, no. 2, pp. 135-253, 2007.
[2] J. Luu, I. Kuon, P. Jamieson, T. Campbell, A. Ye, and W. M. Fang,\VPR 5. 0: FPGA CAD and Architecture Exploration Tools withSingle-Driver Routing, Heterogeneity and Process Scaling," Computer Engineering, pp. 133-142, 2009.
[3] J. Rose, J. Luu, C. Yu, O. Densmore, J. Goeders, A. Somerville,K. Kent, P. Jamieson, and J. Anderson, \The vtr project: architecture and cad for fpgas from verilog to routing," in Proceedings of theACM/SIGDA international symposium on Field Programmable GateArrays, pp. 77-86, ACM, 2012.

[4] Mr. V. Betz and J. Rose, \VPR: A New Packing, Placement and RoutingTool for," Technology, pp. 1-10, 1997.
[5] "VPR 6.0 Beta Architecture Description Language",https://code.google.com/p/vtr_verilog_to_routing/wiki/VPR_Arch_Language.
[6] Singh, S.,"The effect of logic blocks architecture on FPGA performance",Solid-State Circuits, IEEE Journal of Volume: 27, Issue: 3,Mar. 1992, pp. 277-282.
[7] P. Jamieson and J. Rose, \A verilog RTL synthesis tool for heteroge-neous FPGAs," in Field Programmable Logic and Applications, 2005.
[8] P. Leong, W. Luk, and S. Wilton, \Floating-Point FPGA: Architectureand Modeling," IEEE Transactions on Very Large Scale Integration(VLSI) Systems, vol. 17, pp. 1709{1718, Dec. 2009.

Author Profile

Abhijeet Khandalereceived his B.E. degree in Electronics and telecommunication engineering from Mumbai University, Maharashtra, India, in 2011. Currently obtaining MTech degree in Electronics from BMS College of engineering, VTU, Karnataka. His research interests include Digital circuits and logic design, FPGA Programming, Image processing, advanced computer architecture, low power VLSI.

Dr.H.R.Bhagyalakshmi received her B.E. degree in Electronics and communication engineering from Bangalore University, Karnataka, India, in 1985. Later on obtained the ME degree in Electronics in 1995 from Bangalore university, Karnataka. Currently she is an Associate professor in the department of Electronics and communication B.M.S College of engineering, Bangalore. She obtained her Ph.D. degree from the Visveswaraya technological university, Belgaum. Her research interests include Digital circuits and logic design, reversible logic and synthesis, multiple valued logic, advanced computing techniques, low power VLSI.