International Journal of Electronics and Communication Engineering (IJECE)
ISSN 2278-9901
Vol. 2, Issue 1, Feb 2013, 69-76

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

A STUDY OF ARCHITECTURE OF FPGA ROUTING

B. HARIKRISHNA[1] & S. RAVI[2]

[1]Research Scholar, Sathyabama University, Chennai, India

[2]Professor & Head, Department of Electronics Engineering, Dr. M.G.R. University, Chennai, India

ABSTRACT

An FPGA is a device that consists of a matrix of reconfigurable gate array logic circuitry. A single FPGA can replace thousands of discrete components by incorporating millions of logic gates in a single integrated circuit (IC) chip. The internal resources of an FPGA chip consist of a matrix of configurable logic blocks (CLBs) surrounded by a periphery of I/O blocks. Signals are routed within the FPGA matrix by programmable interconnect switches and wire routes. The programmable routing in an FPGA provides connections among logic blocks and I/O blocks to complete a user-designed circuit. It consists of wires and programmable switches that form the desired connections. To accommodate a wide variety of circuits, the interconnect structure must be flexible enough to support widely varying local and distant routing demands together with the design goals of speed performance and power consumption. This paper presents an overview of FPGA routing architectures. In commercial architectures, the routing consumes most of the chip area, and is responsible for most of the circuit delay. As FPGAs are migrated to more advanced technologies, then the routing fabric becomes even more important. Thus, there has been a great deal of recent interest in developing the efficient FPGA routing architectures.

KEYWORDS: FPGA, I/O Blocks, IC, Logic Blocks, Routing

INTRODUCTION

An FPGA [1] is a device that contains a matrix of reconfigurable gate array logic circuitry. When a FPGA is configured, the internal circuitry is connected in a way that creates a hardware implementation of the software application. Unlike processors, FPGAs use dedicated hardware for processing logic and do not have an operating system. FPGAs are truly parallel in nature so different processing operations do not have to compete for the same resources. As a result, the performance of one part of the application is not affected when additional processing is added.

Also, multiple control loops can run on a single FPGA device at different rates. FPGA-based control systems can enforce critical interlock logic and can be designed to prevent I/O forcing by an operator. However, unlike hard-wired printed circuit board (PCB) designs which have fixed hardware resources, FPGA-based systems can literally rewire their internal circuitry to allow reconfiguration after the control system is deployed to the field. FPGA devices deliver the performance and reliability of dedicated hardware circuitry. In this paper we present an overview of routing architectures of FPGA. Routing [2] architecture design is a very challenging because the best value for each of the parameters above depends on complex trade-offs. For example, in an FPGA with too many short wires, then some long connections will be constructed using several short wire segments connected in series, resulting in poor speed. If an FPGA includes too many long wires, then some short connections will be forced to use long wire segments, degrading speed and wasting area.

STRUCTURE OF FPGA

A single FPGA can replace thousands of discrete components by incorporating millions of logic gates in a single integrated circuit (IC) chip[3]. The internal resources of an FPGA chip consist of a matrix of configurable logic blocks

(CLBs) surrounded by a periphery of I/O blocks shown in Figure 1. Signals are routed within the FPGA matrix by programmable interconnect switches and wire routes.

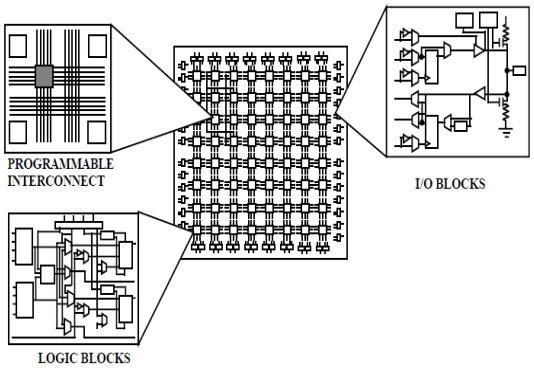

Figure 1: Internal Structure of FPGA

In an FPGA logic blocks are implemented using multiple level low fan-in gates, which gives it a more compact design compared to an implementation with two-level AND-OR logic. FPGA provides its user a way to configure:

- The intersection between the logic blocks and
- The function of each logic block.

Logic block of an FPGA can be configured in such a way that it can provide functionality as simple as that of transistor or as complex as that of a microprocessor. It can used to implement different combinations of combinational and sequential logic functions. Logic blocks of an FPGA can be implemented by any of the following:

- Transistor pairs
- Combinational gates like basic NAND gates or XOR gates
- n-input Lookup tables
- Multiplexers
- Wide fan-in And-OR structure.

Routing in FPGAs consists of wire segments of varying lengths which can be interconnected via electrically programmable switches. Density of logic block used in an FPGA depends on length and number of wire segments used for routing. Number of segments used for interconnection typically is a tradeoff between density of logic blocks used and amount of area used up for routing. Simplified version of FPGA internal architecture with routing is shown in Figure 2.

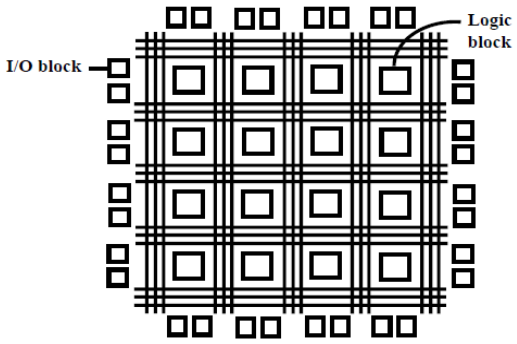

Figure 2: Simplified Internal Structure of FPGA

FPGA ROUTING ARCHITECTURES

Routing architecture comprises of programmable switches and many wires. Routing provides connection between logic blocks, I/O blocks, and between one logic block and another logic block. The type of routing architecture decides area consumed by routing as well as density of logic blocks. Routing techniques used in an FPGA largely decides the amount of area used by wire segments and programmable switches as compared to area consumed by logic blocks.

Hierarchical Routing Architecture

Hierarchical routing architectures separates FPGA logic blocks into distinct groups [4][5]. Connections between the logic blocks within a group can be made using wire segments at the lowest level of the routing hierarchy. Connections between the logic blocks in distant groups require the traversal of one or more levels (of the hierarchy) of routing segments. As shown in Figure 3, only one level of routing (Level 1) directly connects to the logic blocks. Programmable connections are represented with the crosses and circles. Generally, the width of routing channels is widest at levels furthest from the logic blocks. This hierarchical global routing architecture has been used in a number of commercial FPGA families which includes Altera Flex10K [6], Apex [7], and Apex II [8] architectures.

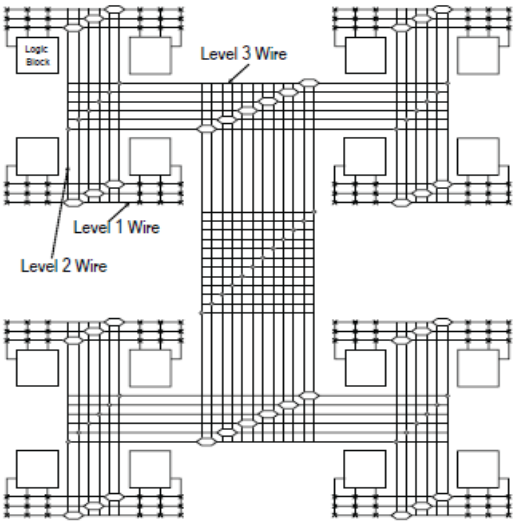

Figure 3: Example of Hierarchical FPGA

Xilinx Routing Architecture

In Xilinx routing [9], connections are made from logic block into the channel through a connection block. As SRAM technology is used to implement Lookup Tables, connection sites are large. A logic block is surrounded by connection blocks on all four sides. They connect logic block pins to wire segments. Pass transistors are used to implement connection for output pins, while use of multiplexers for input pins saves the number of SRAM cells required per pin. The logic block pins connecting to connection blocks can then be connected to any number of wire segments through switching blocks. Figure 4 shows the Xilinx routing architecture. There are four types of wire segments available:

- General purpose segments, the ones that pass through switches in the switch block.
- Direct interconnect : ones which connect logic block pins to four surrounding connecting blocks
- Long line : high fan out uniform delay connections
- Clock lines : clock signal provider which runs all over the chip.

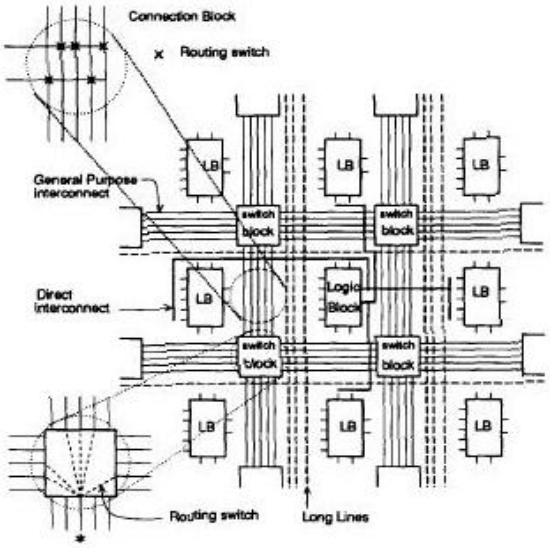

Figure 4: Xilinix Routing Architecture

Altera Routing Architecture

Altera routing architecture [10] has two level hierarchy. At the first level of the hierarchy, 16 or 32 of the logic blocks are grouped into a Logic Array Block (LAB); structure of the LAB is very similar to a traditional PLD. The connection is formed using EPROM- like floating-gate transistors. The channel here is set of wires that run vertically along the length of the FPGA. Figure 5 shows Alter Max 5000 routing architecture. Tracks are used for four types of connections:

- Connections from output of all logic blocks in LAB.
- Connection from logic expanders.
- connections from output of logic blocks in other LABs
- connections to and from Input output pads

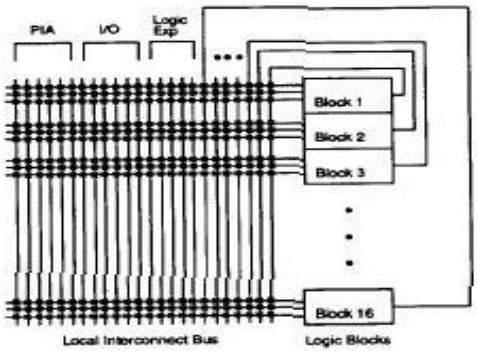

Figure 5: Alter Max 5000 Routing Architecture

All four types of tracks connect to every logic block in the array block. The connection block makes sure that every such track can connect to every logic block pin. Any track can connect to into any input which makes this routing simple. The intra-LAB routing consists of segmented channel, where segments are as long as possible. Global interconnect structure called programmable interconnect array(PIA) is used to make connections among LABs. Its internal structure is similar to internal routing of a LAB. Advantage of this scheme is that regularity of physical design of silicon allows it to be packed tightly and efficiently. The disadvantage is the large number of switches required, which adds to capacitive load.

Island-Style Routing Architecture

As shown in Figure 6, island-style FPGAs logic blocks are arranged in a two dimensional mesh with the routing resources evenly distributed throughout the mesh. An island-style global routing architecture typically has the routing channels on all four sides of the logic blocks. The number of wires contained in the channel, W, is pre-set during fabrication, and is one of the key choices made by the architect. Island-style routing architectures generally employs wire segments of different lengths in each channel in an attempt to provide the most appropriate length for each given connection. They also typically stagger the starting point of the wire segments so that each of the logic blocks has a chance of connecting at the beginning of a wire of the most appropriate length. Currently, most commercial SRAM-based FPGA architectures [11][12] uses island-style architectures. Altera architecture uses two levels of hierarchy. At the first level of the hierarchy, 16 or 32 of the logic blocks are grouped into a Logic Array Block.

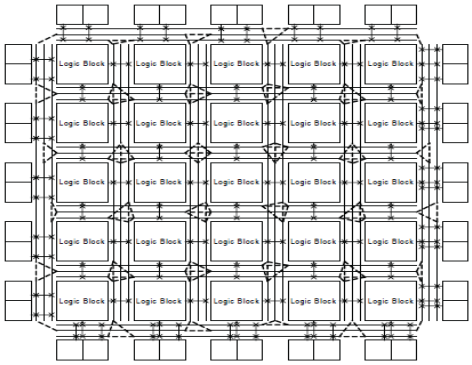

Figure 6: Island-Style Routing Architecture

Actel Routing Architecture

Actel's design (figure 7)[13] has more wire segments in horizontal direction than in vertical direction. The input pins connect to all tracks of the channel that is on the same side as the pin. The output pins extend across two channels above the logic block and two channels below it. Output pin can be connected to all 4 channels that it crosses. The switch blocks are distributed throughout the horizontal channels. All vertical tracks can make a connection with every incidental horizontal track. This allows for the flexibility that a horizontal track can switch into a vertical track, thus allowing for horizontal and vertical routing of same wire. The drawback is more switches are required which add up to more capacitive load.

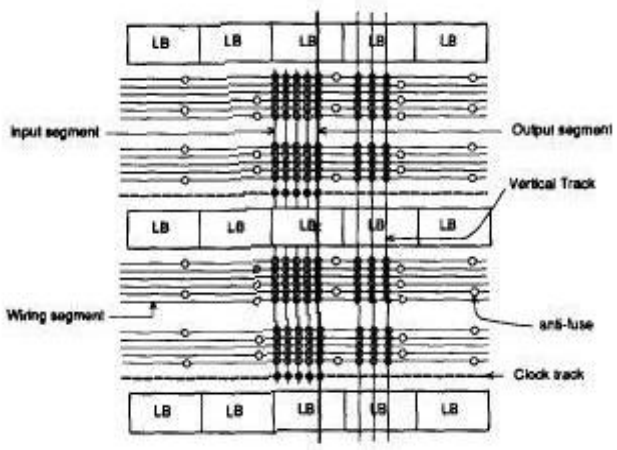

Figure 7: Actel Routing Architecture

CONCLUSIONS

Conceptually FPGA can be considered as an array of Configurable Logic Blocks (CLBs) that can be connected together through a vast interconnection matrix to form complex digital circuits. FPGA architecture consists of programmable logic elements and a programmable routing fabric. Routing in FPGAs consists of wire segments of varying lengths which can be interconnected via electrically programmable switches. Density of logic block used in an FPGA depends on length and number of wire segments used for routing. Number of segments used for interconnection typically is a tradeoff between density of logic blocks used and amount of area used up for routing. In this paper we present different architectures based on FPGA routing. Hierarchical routing architectures separate FPGA logic blocks into distinct groups. Connections between logic blocks within a group can be made using wire segments at the lowest level of the routing hierarchy. In Xilinx routing, connections are made from logic block into the channel through a connection block. As SRAM technology is used to implement Lookup Tables, connection sites are large. Island style architecture is the most commonly used architecture among academic and commercial FPGAs. It is called the island-style architecture because in this architecture configurable logic blocks look like islands in a sea of routing interconnect. In this architecture, the configurable logic blocks (CLBs) are arranged on a 2D grid and are interconnected by a programmable routing network. Actel's design has more wire segments in horizontal direction than in vertical direction. The input pins connect to all tracks of the channel that is on the same side as the pin. The output pins extend cross two channels above the logic block and two channels below it.

REFERENCES

1. A. Singh and M. Sadowska, FPGA interconnect planning. In Proceedings of the 2002 International Workshop on System-Level Interconnect Prediction. ACM, New York, 23–30.
2. M.Alexander and G. Robins, New Performance driven FPGA routing algorithms, 32nd ACM/IEEE Design Automation Conference, San Francisco,CA, 1995, pp. 562-567.
3. Andrew Wylie (2009). "The first integrated circuits". Retrieved 14 March 2011.
4. A. Aggarwal and D. Lewis, “Routing architectures for hierarchical field programmable gate arrays,” in IEEE International Conference on Computer Design, pp. 475–478, October 1994.
5. W. Tsu, K. Macy, A. Joshi, R. Huang, N. Walker, T. Tung, O. Rowhani, G. Varghese, J. Wawrzynek, and A. DeHon, “HSRA: High-speed, hierarchical synchronous reconfigurable array,” in Proceedings: ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, pp. 125–134, February 1999.
6. Altera Corporation, “FLEX 10K embedded programmable logic device family, DS-F10K-4.2,” http://www.altera.com/literature/ds/dsf10k.pdf, January 2003.
7. Altera Corporation, “APEX 20K programmable logic device family data sheet, DS-APEX20K-5.1,” http://www.altera.com/literature/ds/apex.pdf, March 2004.
8. Altera Corporation, “APEX II programmable logic device family, DSAPEXII- 3.0,” http://www.altera.com/literature/ds/ds ap2.pdf, August 2002.
9. Xilinx Inc, Virtex II Data Book, 2004.

10. Altera Corporation, Stratix II Device Handbook, Volume 1, Jul 2004. International Conference on computer design pp.457-478 October 1994
11. Altera Corporation, "Stratix II device handbook SII5V1 3.1,"http://www.altera.com/literature/hb/stx2/stratix2handbook.pdf, July 2005.
12. Altera Corporation, "Stratix III device handbook, ver 1.0," http://www.altera.com/literature/hb/stx3/stratix3 handbook.pdf, November 2006.
13. Actel Inc, Axcelerator family FPGA, 2004.

AUTHOR'S DETAILS

Mr. B. Hari Krishna is presently a research scholar in Sathyabama University, Chennai and is pursuing his research in the area of Autonomous Restructuring Systems, he is presently working as Assoc.professor in BITS, Khammam. He has 7 years of experience. His interested areas are Image Processing, Embedded systems and FPGA testing.

Dr. S. Ravi was born on July 24th, 1971 at Chennai, India and got A.M.I.E. (Electronics Engineering) from Institution of Engineers, Calcutta in the Year 1991, M. Tech. (Communication systems) from Anna University, Chennai in the Year 1994 and Ph.D. from Anna University in the year 2004. He has got 17 Years of Teaching Experience and 3 Years of Industrial Experience. He attended Summer and Winter School Programs at Anna University, Chennai, IITM Chennai and Indian Institute of Science, Bangalore. Presently, he is working as Professor and Head of the Department of Electronics Engineering in Dr. M.G.R. University, Chennai. From the Year 1994 onwards he is working as faculty in the above college. He has so far published nearly twenty papers in referred International Journals and successfully guided three Ph.D. scholars. Dr. S. Ravi is a Life Member and Charted Engineer in Institute of Engineers (I) and Life Member of Indian Society for Technical Education.