

Chapter 2

Architecture of Target FPGA Platform

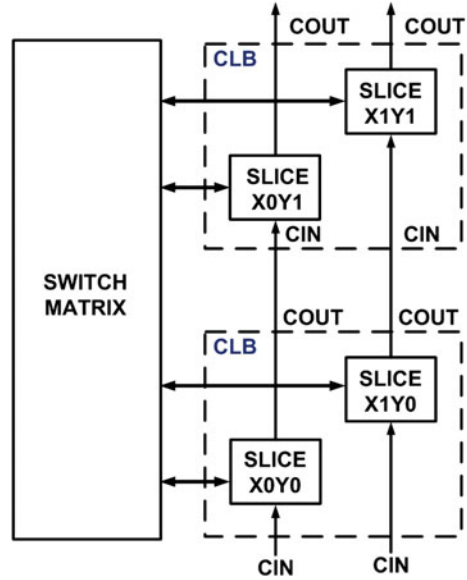
Abstract This chapter provides an insight into the architecture of Configurable Logic Blocks (CLBs), the basic building blocks of a FPGA, including details of the Look-Up Tables, wide function multiplexers, carry chains, flip-flops, and DSP slices. It also gives an overview of the different modes of implementation supported by Xilinx ISE to realize arithmetic functions.

2.1 Introduction

Current FPGAs such as the advanced Virtex and Spartan families of Xilinx FPGAs (e.g., Virtex-5, Virtex-6 and Spartan-6), promise immense hardware logic support, at higher integration, lower power consumption, and maximum performance. Maximum system performance requires a balanced mix of performance-efficient FPGA components: logic fabric (Look-Up Tables (LUTs), special functions like carry chains and dedicated multiplexers, flip-flops (FFs)), on-chip RAM, DSP blocks, and I/Os. Virtex-5 FPGAs have been the first FPGA device fabricated at the 65 nm CMOS technology node. Switching from 90 nm (for Virtex-4 FPGAs) to 65 nm (for Virtex-5 FPGAs) [4] have promised the above-mentioned advantages. Spartan-6 FPGAs, on the other hand, are built on a mature 45 nm low-power copper process technology [3] that delivers the optimal balance of cost, power and performance, whereas Virtex-6 FPGAs are built using a 40 nm state-of-the-art copper process technology, and are a programmable alternative to custom ASIC technology [9]. Xilinx 7 series FPGAs leverage the unprecedented power, performance, and capacity enabled by TSMC's (Taiwan Semiconductor Manufacturing Company Limited) 28 nm [2] high-k metal gate (HKMG), high performance, low power (HPL) process technology, and the unparalleled scalability afforded by the FPGA industry's first scalable, optimized architecture to provide a comprehensive platform base for next-generation systems.

The Configurable Logic Blocks (CLBs) of FPGAs are the main logic resources for implementing sequential as well as combinatorial circuits. Each CLB element is connected to a switch matrix for access to the general routing matrix as shown in Fig. 2.1. Each CLB element for Virtex-5 and Virtex-6 series of FPGAs (that have been our target platform for implementation) contain a pair of slices. These two slices do not have direct connections to each other, and each slice is organized as a column.

Fig. 2.1 Arrangement of slices within the CLB [5]



The Xilinx tools designate slices with the following definitions [5]. An “X” followed by a number identifies the position of each slice in a pair as well as the column position of the slice. A “Y” followed by a number identifies a row of slices. The number remains the same within a CLB, but counts up in sequence from one CLB row to the next CLB row.

The rest of the chapter is organized as follows. In Sect. 2.2, we present the slice architecture for Xilinx Virtex-5 FPGAs. In Sect. 2.3, we present the additional and modified features that Virtex-6 FPGAs offer in comparison to Virtex-5 FPGAs. A brief overview of the DSP slice architecture has been presented in Sect. 2.4. The different modes of implementation—fabric and DSP slice logic have been discussed in Sect. 2.5. We conclude in Sect. 2.6.

2.2 Fabric Slice Architecture for Virtex-5 FPGAs

The CLBs of Xilinx FPGA are the main logic resources for implementing combinational and sequential circuits. A typical CLB of Virtex-5 FPGA contains 2 “slices,” with each slice (called a “SLICEL” or “SLICEM” in Xilinx terminology depending on the nature of LUTs) comprising of four 6-input logic-function generators or LUTs, four storage elements or FFs, three wide function multiplexers, and a length-4 carry chain comprising of multiplexers and XOR gates [1, 5] as shown in Fig. 2.2. All these elements are used by the slices for realization of arithmetic, logic, and memory functions.

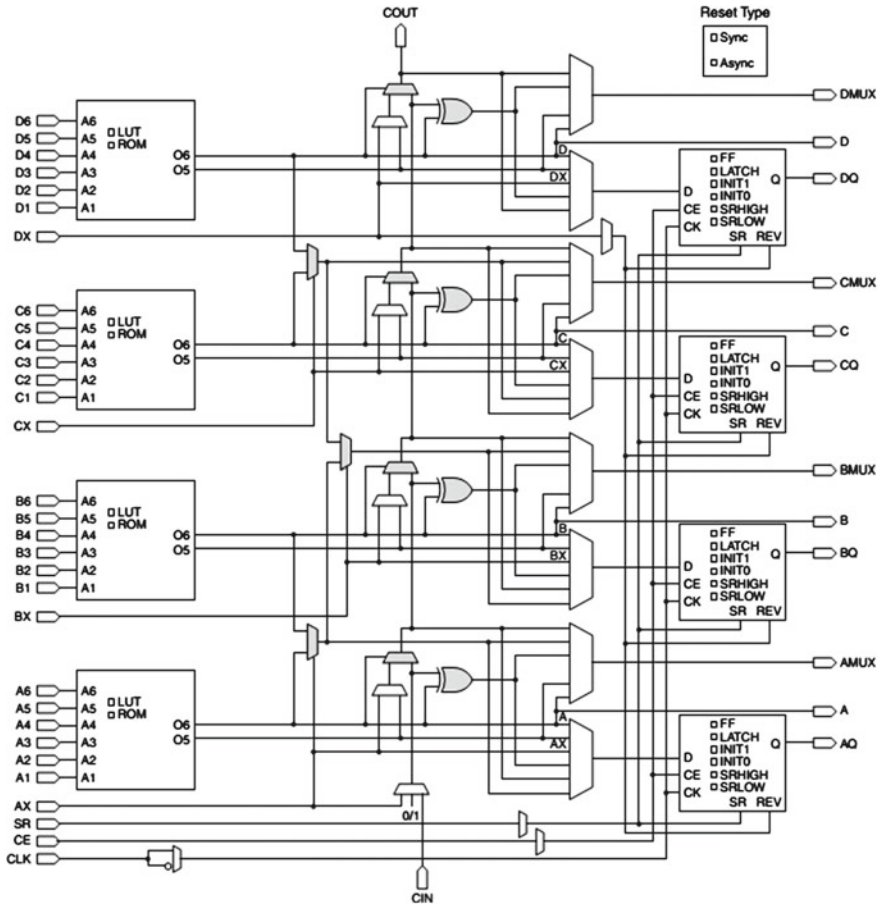


Fig. 2.2 Xilinx Virtex-5 slice architecture [5]

The Xilinx Virtex-5 family has been the first FPGA platform to offer a true 6-input LUT, with fully independent (not shared) inputs. This allows the implementation of functions with higher operand width and reduces the number of logic levels between registers. The 6-input LUT can also be configured as a 5 (or less) input, 2-output logic function with shared inputs, thereby reducing the requirement in the number of LUTs from two to one for certain logic expressions elaborated in Chap. 3. The LUTs present in SLICEL can implement any arbitrary combinational logic, whereas the LUTs in SLICEM can be implemented as a synchronous RAM resource called a distributed RAM element. The carry chain represents the fast carry propagation logic and the LUTs in the slice can be optionally connected to the carry chain via dedicated routes to implement complex logic functionality [7]. The storage elements in a slice can be configured as either edge-triggered D-type FFs or level-sensitive latches. Each FF can be controlled using the control signals *set*, *reset*, *clock*, and *clock enable* signals.

greater number of serial transceivers that can deliver at a higher Gbps rate, along with a faster global clocking with lower skew, improved jitter and faster clock trees in comparison to Virtex-5 FPGA platforms.

2.4 DSP Slice Architecture for Virtex-5 and Virtex-6 FPGAs

DSP slices in FPGAs are typically designed for low power applications as it significantly avoids fabric routing, but provides a reasonable speed of operation. The Virtex-5 FPGA DSP48E slice, as shown in Fig. 2.4, supports several independent arithmetic functionalities. Such functional units include a 25×18 two's complement multiplier, multiply accumulate (MACC) unit, multiply adder, three-input adder, barrel shifter, wide-bus multiplexer, magnitude comparator, bitwise logic functions, pattern detector, and wide counter. The slice has internal pipeline stages which must be used for achieving maximum performance up to 550 MHz.

For Virtex-6 FPGAs, the DSP slice available, DSP48E1, has all the features of a Virtex-5 FPGA DSP48E slice with certain additional features [10]. When all pipeline stages are used, Virtex-6 DSP slices can achieve a 600MHz speed of operation. It supports an additional 25-bit pre-adder and register with another additional control unit.

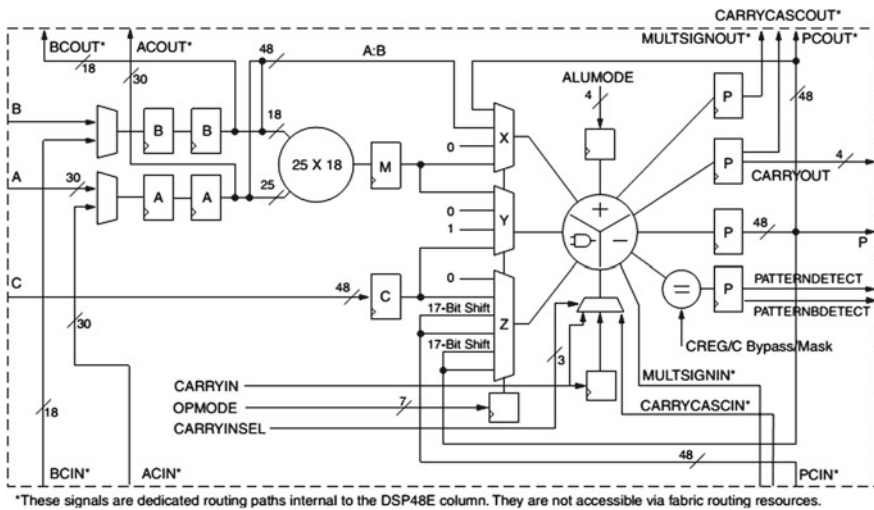


Fig. 2.4 Xilinx Virtex-5 DSP48E slice [6]

2.5 Implementation Overview

Native platform-dependent primitives such as 6-input LUT and carry chain can be directly instantiated in the HDL circuit description, and they appear unchanged in the final implementation mapped on the FPGA. For efficient and high performance design, the designer must ensure maximum utilization of the logic elements within each slice, and place the logically related slices into adjacent locations.

Designs targeted toward Virtex-5 and Virtex-6 FPGAs can also exploit IP cores available specifically for various arithmetic and logic functions. The Xilinx *Logi-CORE*[®] hard IP provides two modes of implementation: *Fabric* and *XtremeDSP*[®]. Fabric implementation involves utilization of LUTs, FFs, multiplexers and carry chains, whereas *XtremeDSP* implementation involves utilization of the special *DSP* slice which can also be instantiated as a primitive. DSP slice-based implementations guarantee lower power consumption in comparison to fabric logic, but as we would find, cannot always match the performance achievable by circuits implemented using constrained placement of Xilinx fabric logic.

Most of our performance results have been reported using the Virtex-5 FPGA as the implementation platform. However, for certain circuits, like the cellular automata-based pseudorandom binary sequence generator or the matrix multiplication circuit, where it is necessary to register the dual outputs of the LUTs, Virtex-6 have been chosen as the implementation platform to ensure a compact implementation and pack more registers into a single slice, thereby freeing up resources that would otherwise have spanned across multiple slices and depleted adjacent slices of their register resources.

The circuits described in Chap. 4–7 were implemented either on a Xilinx Virtex-5 FPGA, device family XC5VLX330T, package FF1738 and speed grade -2 or Xilinx Virtex-6 FPGA, device family XC6VLX550T, package FF1760 and speed grade -2 using the Xilinx ISE 12.4 design environment. The speed of operation, resource utilization, and power–delay product (PDP) of the architectures have been compared with those reported in existing literature (if any) and with different modes of implementation have been tabulated. The designs have been evaluated in terms of speed, resource consumption in terms of FFs, LUTs, slices and DSP hard macros (whenever applicable), and power–delay product (PDP). Power–delay product has been calculated as the product of the power dissipation (sum of clock, logic, signal, and DSP power dissipation), the (minimum) clock-period (toggle rate of 12.5 %), and the latency (in terms of the number of clock cycles required to complete the computation).

Functions implemented using the DSP slices consume less power than those implemented in general FPGA fabric [6], and this would be evident from the results. However, it would also be evident that the proposed methodology based on fabric logic, combined with careful and constrained placement can outperform the DSP slice-based design with respect to speed. To achieve maximum performance using the DSP slices, it is desirable to use all the pipeline stages within the DSP slice.

2.6 Summary

In this chapter, we have introduced the modern and advanced families of Xilinx FPGAs that provide immense logic integration facilities and architectural support for high-performance implementations. The slice architectures for Virtex-5 and Virtex-6 FPGA families were described. An overview of the different modes of implementation were presented.

The next chapter will address a fabric component-based approach for realization of arithmetic circuits on modern FPGA families, where certain guidelines for manipulation and decomposition of Boolean logic level equations describing the implemented circuits will be discussed so that they can be easily and efficiently mapped to the physical fabric logic primitives of the target FPGA platform. Such an approach also allows the designer to predict the overall hardware cost and ensure a careful and compact placement of the logic architectures on the FPGA fabric. Certain examples of useful and practical circuits have also been described to illustrate the application of such guidelines for logic design.

References

1. Cosoroaba, A., Rivoallon, F.: Xilinx Inc., White Paper: Virtex-5 Family of FPGAs, Achieving Higher System Performance with the Virtex-5 Family of FPGAs, WP245 (v1.1.1). http://www.xilinx.com/support/documentation/white_papers/wp245.pdf. Cited 7 July 2006
2. Mehta, N.: Xilinx Inc., Xilinx 7 series FPGAs: the logical advantage, WP405 (v1.0). http://www.xilinx.com/support/documentation/white_papers/wp405-7Series-Logical-Advantage.pdf. Cited 6 Mar 2012
3. Xilinx Inc.: Spartan-6 family overview, DS160 (v2.0). http://www.xilinx.com/support/documentation/data_sheets/ds160.pdf. Cited 25 Oct 2011
4. Xilinx Inc.: Virtex-5 family overview, DS100 (v5.0). http://www.xilinx.com/support/documentation/data_sheets/ds100.pdf. Cited 6 Feb 2009
5. Xilinx Inc.: Virtex-5 FPGA user guide, UG190 (v5.4). http://www.xilinx.com/support/documentation/user_guides/ug190.pdf. Cited 16 Mar 2012
6. Xilinx Inc.: Virtex-5 FPGA xtremeDSP design considerations user guide, UG193 (v3.5). http://www.xilinx.com/support/documentation/user_guides/ug193.pdf. Cited 26 Jan 2012
7. Xilinx Inc.: Virtex-5 libraries guide for HDL designs, UG621 (v11.3). http://www.xilinx.com/support/documentation/sw_manuals/xilinx11/virtex5_hdl.pdf. Cited 6 Sep 2009
8. Xilinx Inc.: Virtex-6 FPGA configurable logic block, UG364 (v1.2). http://www.xilinx.com/support/documentation/user_guides/ug364.pdf. Cited 3 Feb 2012
9. Xilinx Inc.: Virtex-6 family overview, DS150 (v2.4). http://www.xilinx.com/support/documentation/data_sheets/ds150.pdf. Cited 19 Jan 2012
10. Xilinx Inc.: Virtex-6 FPGA DSP48E1 slice user guide, UG369 (v1.3). http://www.xilinx.com/support/documentation/user_guides/ug369.pdf. Cited 14 Feb 2011

High Performance Integer Arithmetic Circuit Design on
FPGA

Architecture, Implementation and Design Automation

Palchaudhuri, A.; Chakraborty, R.S.

2016, XVII, 114 p. 56 illus., Hardcover

ISBN: 978-81-322-2519-5