Int. J. Elec&Electr.Eng&Telecoms. 2014

International Journal of Electrical and Electronic Engineering & Telecommunications

ISSN 2319 – 2518 www.ijeetc.com
Vol. 3, No. 1, January 2014

Research Paper

FIELD PROGRAMMABLE GATE ARRAYS AND THEIR APPLICATIONS

S M Shashidhara[1]*

*Corresponding Author: S M Shashidhara, ✉ shashi.com@gmail.com

---

In recent years, Field-Programmable Gate Arrays (FPGAs) have become increasingly fascinating as digital logic engines, sometimes used alone and sometimes in conjunction with a processor chip. The largest FPGA vendors, Altera and Xilinx, have invested heavily in developing DSP-oriented chips and development tools. This article presents insights focusing on the evolving role of FPGAs in embedded systems.

Keywords: FPGA, Embedded systems, Integrated circuit, Reconfigurable, Logic block

INTRODUCTION

Since a very long time, simulations and prototyping have been a very significant part of the electronics industry. Before steering in for the real fabrication of a dedicated hardware, one would desire to be certain that what their making will work the way they want it to. Over the past few decades electronics companies provided dedicated hardware in their products. Hence, it was not conceivable for the end user to reconfigure them to his own needs. This need resulted in the emergence of a new market segment of customer configurable integrated circuits called Field Programmable Gate Arrays or FPGAs.

HISTORY

The FPGA shares a common history with most Programmable Logic Devices. The first of this kind of devices was the Programmable Read Only Memory. Further driven by need of specifically implementing logic circuits, Philips invented the Field-Programmable Logic Array (FPLA) in the 1970s. This consisted of two planes, a programmable wired AND-plane and the other as wired OR. It could implement functions in the Sum of Products (SoP) form.

To subdue the difficulties of speed and cost, Programmable Array Logics were developed which had a programmable 'AND' plane fed into fixed OR gates. PALs and PLAs are

---

[1] Proudhadevaraya Institute of Technology, Hospet, Karnataka, India.

Figure 1: Field Programmable Gate Array Chip

classified as Simple Programmable Logic Devices (SPLDs). These SPLDs were incorporated onto a single chip and interconnects were provided to connect the SPLD blocks through programming. These were called Complex PLDs and were first initiated by Altera.

Then another category of Electronic devices, Mask-Programmable Gate Arrays comprising of transistor arrays which could be connected using custom wires propelled the design of the FPGAs. Transistors broke way to Logic Blocks and the customization could now be executed by the user on the field and not in the manufacturing laboratory. The credit to evolve the first commercially practicable FPGA goes to Xilinx co-founders Ross Freeman and Bernard Vonderschmitt. The XC2064 was devised in 1985 comprising of 64 Configurable Logic Blocks and 3 Look Up Tables.

By the end of 1990, a lot of competition sprung up in manufacturing FPGAs when Xilinx's market share started to downslope. Players like Actel, Altera, QuickLogic, Lattice, Cypress, Lucent and SiliconBlue started entering this domain and carving their niche in the world FPGA Market. Then, Xilinx, started gaining acceptance in applications like Digital Signal Processing and Telecommunications. In 1997, Adrian Thompson succeeded in blending a genetic algorithm technology with FPGA and started a new age of Evolvable hardware.

Regardless of the different makers and slightly different architectures and features, most FPGA's have a common general approach. The main constitutional blocks of any FPGA are a flexible programmable 'Configurable Logic Block' (CLB), enclosed by programmable 'Input/Output Blocks.' The hierarchy of routing channels interconnect different blocks on the board. In addition, these may consist of Clock DLLs for clock distribution and control and Dedicated Block RAM memories.

CONFIGURABLE LOGIC BLOCK

The primary building block of a Configurable Logic Block is the logic cell. A logic cell may comprise of an input function generator, carry logic and a memory element. The function generators are carried out as Look Up Tables dependent on the input. A Xilinx Spartan II has 4 inputs LUT where each LUT can cater a 16 x 1 bit Synchronous RAM which can in addition be multiplexed using multiplexers. An LUT may also be used as a Shift register. The storage elements may be used as edge triggered flip-

Figure 2: Configurable Logic Blocks

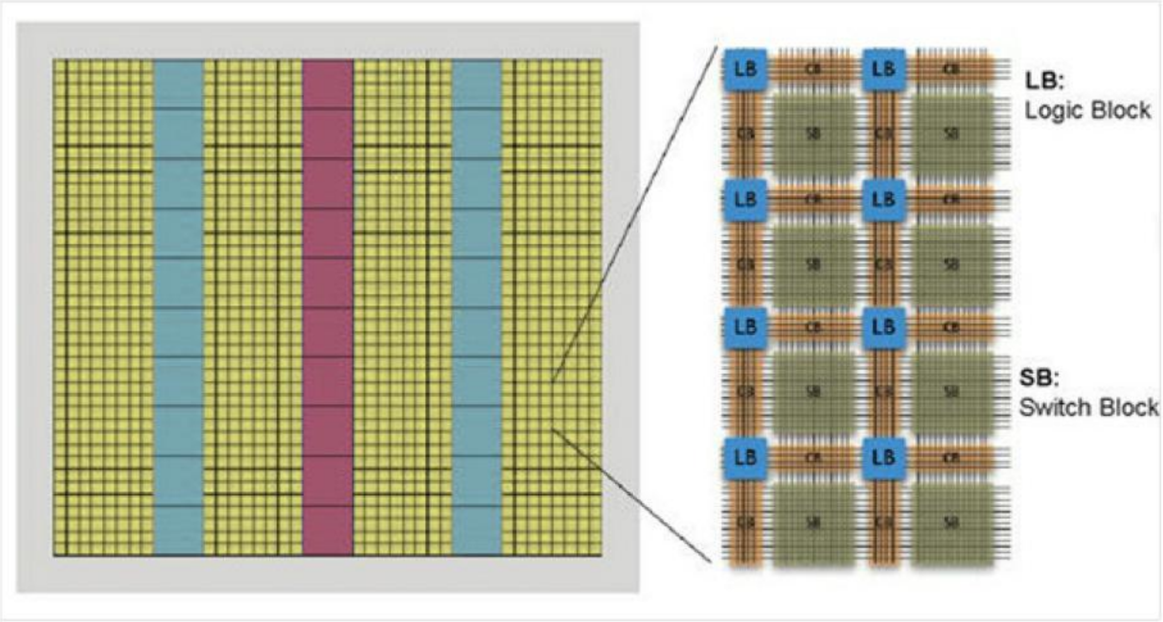
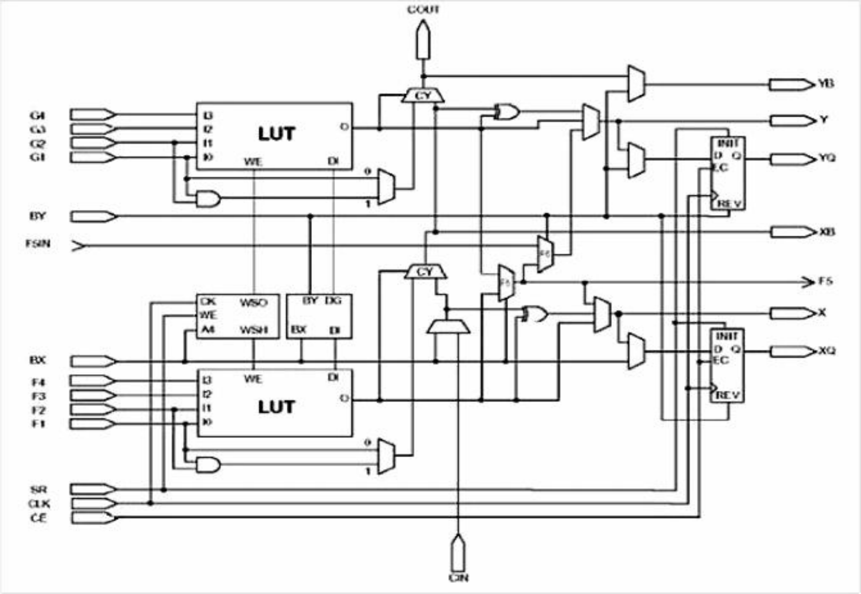


Figure 3: FPGA Slice

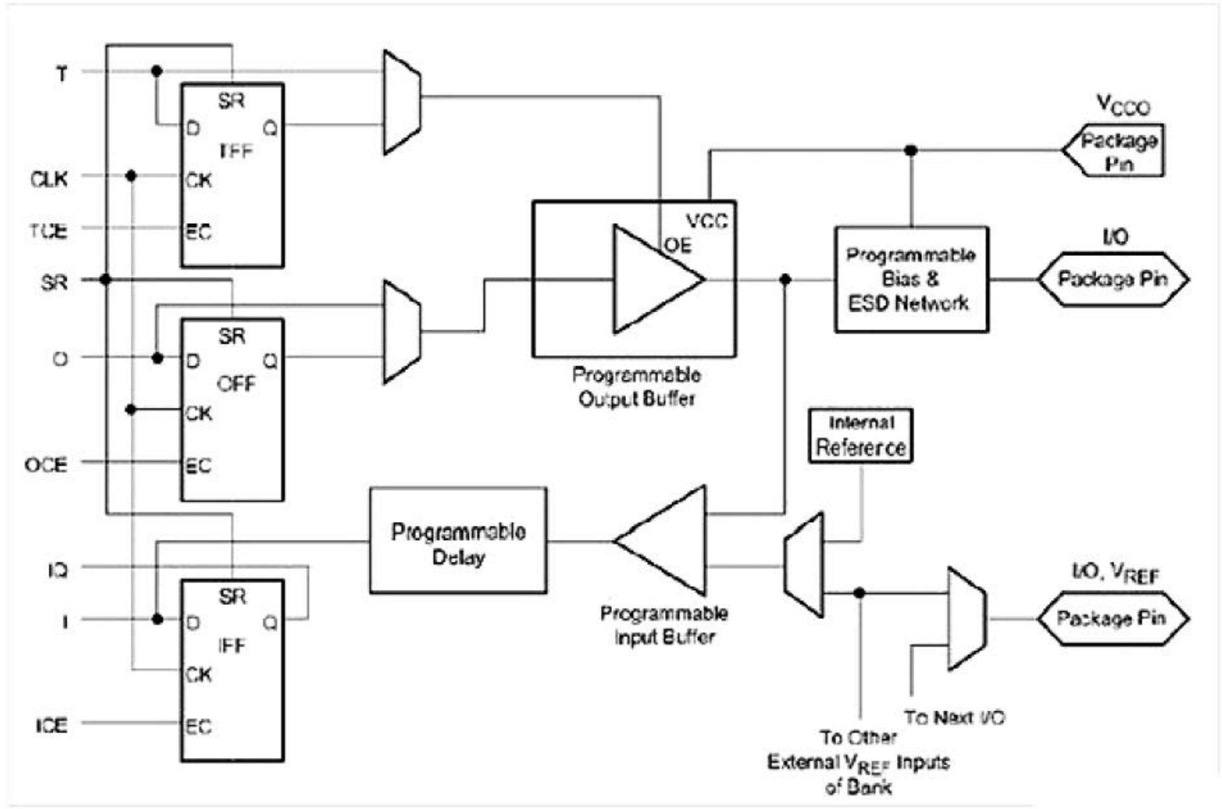


flops or level responsive latches. The arithmetic logic consists of an XOR gate for full-adder operation along with dedicated carry logic channels. The figure below demonstrates an FPGA slice:

INPUT/OUTPUT BLOCK

This block features inputs and outputs bearing a wide range of signaling standards and interfaces. A basic Input/Output block is shown below:

Figure 4: Input/Output Block



The buffers in the Input and output paths route the input and output signals to the internal logic and the output pads either direct or via a flip-flop. The buffer can be set to conform to various supported signaling standards which might even be user defined and externally set.

ROUTING MATRIX

In any assembly line it is often the slowest section which sets the overall output rate. Practically, it is the route that takes the longest delay that finally determines the performance of the entire electronic system. Thus routing algorithms are brought into place for the design of the most effective paths to extradite optimal performance. Routing is done on various levels like Local, between LUTs, flip-flops and General Routing Matrix, General Purpose Routing between various CLBs, I/O Routing between I/O Blocks and CLBs, Dedicated Routing for a certain categories of signals for maximizing performance and Global Routing for distributing clocks and other signals with very high fan-out.

CLOCK DISTRIBUTION

High speed, low skew clock distribution is provided in most FPGAs using Primary Global Routing resources. With each clock input buffer, there is a digital Delay-Locked Loop which eliminates skew between clock input pad and internal clock input pins by adjusting the delay element.

FPGA families now also have large block RAM structures to complement the distributed

RAM Look-Up Tables, sizes varying for different FPGA devices.

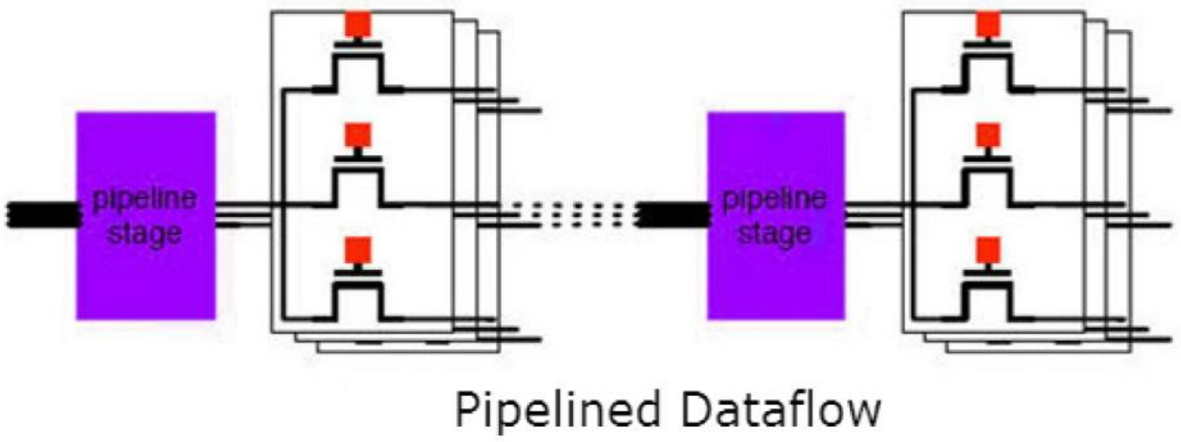
The design of an FPGA follows mostly the similar approach as any VLSI system, the main steps being Design Entry, Behavioral Simulation, Synthesis, Post-Synthesis Simulation, translation, mapping and routing, and further analysis like Timing simulation and Static Timing Analysis.

In order to increase the performance of FPGAs, more transistors could be used. The area overhead involved with FPGAs is higher than ASIC and could possibly do with more density now that 28 nm processes are also being applied on them. Placing more transistors also means that bigger designs would be imaginable. Use of asynchronous FPGA architecture has also shown better effects matched with pipelining technology which reduces global inputs and betters throughput.

Security used to be a major concern as the code needed to be revealed every time it was loaded on to the FPGA's, thus making FPGA's flexibility a likely threat to malicious alterations during fabrication. But, bitstream encryption has arrived to the rescue of FPGAs.

Often the inexperienced designers confront this dilemma that how much powerful FPGA would be appropriate for their application. Manufacturers specify metrics like 'Gate count'. For example, Xilinx uses 3 metrics to assess capacity of FPGA, Maximum Logic Gates, Maximum Memory Bits and normal Gate Range. As long as these cited metrics are agreeable, migration between families is somewhat easy, but it rarely offers subtle comparability between different sellers because of the deviation of architectures and as a result of which, performance alters. A better metric is to compare the number and type of logic resources rendered. In addition to it, the designer must be amply aware of what is precisely required of the device as sellers might be boasting of features which would be of little importance to the job. For example, Altera's Stratix II EP2S180 features about 1,86,576 4-Input LUTs while Xilinx Virtex-4 XC4VLX200 carries 1,78,176. However if the design needs only 175K LUTs, the last mentioned would suffice. If RAM is the desired

Figure 5: Pipelined Dataflow in FPGA



metric for the designer, neither the Xilinx XC4VLX200's 6 Mbits nor the Altera's EP2S180's 9 Mbits would be preferred over the lesser publicised, older model of XC4VFX140 bearing 9.9 Mbits. So it requires complete understanding on the part of the user who ultimately needs it and not what the cutting-edge product on the shelf offerings.

COMMON FPGA APPLICATIONS

FPGAs are used in all walks of life in innumerable number of embedded systems; On an average there are atleast 50 household embedded systems, many of them using FPGA as the processing engine.

Aerospace and Defense: Avionics/DO-254, Communications, Missiles and Munitions, Secure Solutions, Space.

ASIC Prototyping: Before fabricating Application Specific ICs, there functionality is checked on an FPGA.

Audio: Connectivity Solutions, Portable Electronics, Radio, Digital Signal Processing (DSP).

Automotive: High Resolution Video, Image Processing, Vehicle Networking and Connectivity, Automotive Infotainment.

Broadcast: Real-Time Video Engine, EdgeQAM, Encoders, Displays, Switches and Routers.

Consumer Electronics: Digital Displays, Digital Cameras, Multi-function Printers, Portable Electronics, Set-top Boxes.

Distributed Monetary Systems: Transaction verification, BitCoin Mining

Data Center: Servers, Security, Routers, Switches, Gateways, Load Balancing.

High Performance Computing: Servers, Super Computers, SIGINT Systems, High-end RADARS, Data Mining Systems, High-end Beam Forming Systems.

Industrial: Industrial Imaging, Data Mining Systems, Industrial Networking.

Medical: Ultrasound, CT Scanner, X-ray, PET, X-ray, PET, MRI, Surgical Systems.

Security: Industrial Imaging, Secure Solutions, Image Processing.

Video and Image Processing: High Resolution Video, Video Over IP Gateway, Digital Displays, Industrial Imaging.

Wired Communications: Optical Transport Networks, Network Processing, Connectivity Interfaces.

Wireless Communications: Baseband, Connectivity Interfaces, Mobile Backhaul, Radio.

FPGA ADVANTAGES

Systems based on FPGAs provide many advantages over conventional implementations:

Long Time Availability: Field Programmable Gate Arrays (FPGAs) enable customer independent from component manufacturers since the functionality is not given by the device itself but in its configuration. The configuration can be programmed to be portable between miscellaneous FPGAs without any adaptations.

Can be Updated and Upgraded at Customer's Site (Field Programmable): FPGAs in contrast to traditional IC chips are completely configurable. Updates and feature enhancement can be carried out even after delivery at customer's site.

Extremely Short Time to Market: Through the use of FPGAs, the development of hardware prototypes is significantly quickened since a larger part of the hardware development process is shifted into IP core design, which can go on in parallel. In addition, because of the early availability of hardware prototypes, time-consuming activities like the start-up and debugging of the hardware are advanced concurrently to the overall development.

Performance Gain for Software Applications: Complex tasks are often handled through software implementations in combination with high-performance processors. In this case FPGAs provide a competitive alternative, which by means of parallelization and customization for the specific task even gives an additional performance gain.

Fast and Efficient Systems: With FPGAs, systems can be developed that are precisely customized for the intended task and for this reason works extremely efficient.

Massively Parallel Data Processing: The amount of data in contemporary systems is ever increasing which leads to the problem that systems working sequential are no longer able to process the data on time. Particularly by means of parallelization, FPGAs provide an answer to this problem which in addition scales excellently.

Real Time Applications: FPGAs are perfectly suited for applications in time-critical systems. In contrast to software based solutions with real time operating systems, FPGAs provide real deterministic behavior. By means of the featured flexibility even complex computations can be accomplished in extremely short periods.

CONCLUSION

This paper reviewed some of the recent advances in FPGA technology. As higher performance becomes necessary and ASIC costs, risk adversity and time to market pressures continue to increase, FPGAs will continue to grow in importance.

REFERENCES

1. Article on available at FPGA http://www.af-inventions.de
2. BDTI's New Report (2013), "Special Preview: BDTI's FPGAs for DSP", *EE Times Jounal*, 2nd Edition.
3. Field-programmable gate array Artcle available at http://www.en.wikipedia.org/wiki/Field-programmable_gate_array
4. FPGA Applications Article, available at http://www.engineersgarage.com