

Performance Evaluation of Hybrid Reconfigurable Computing Architecture over Symmetrical FPGA

Sunil Kr. Singh *Member AIRCC*¹, R. K. Singh², M. P. S. Bhatia³,

¹*Ph.D, Research scholar, Uttarakhand Technical University, Uttarakhand, INDIA*
Email: anujsunilsingh@yahoo.co.in,

² *Professor, Uttarakhand Technical University, Uttarakhand, INDIA*

³ *Professor, Netaji Subhash Institute of Technology (NSIT), New Delhi.,INDIA*

Abstract

For last few decades, reconfigurable devices have been extensively used in digital systems. Reconfigurable computing using FPGA devices provide a method to utilize the available logic resources on the chip for various computations. The basic ability of reconfigurable computing is to perform computations in hardware to increase performance, while retaining the flexibility of application software. The two main types of programmable logic devices, field-programmable gate arrays (FPGA) based on LUTs technology and complex programmable logic device (CPLD) based on PLAs technology. They are both widely used and each contributing particular strengths in the area of reconfigurable system design. We identified Hybrid LUTs/PLAs architectures as Hybrid Reconfigurable Computing Architectures (HRCA). The purpose of this paper is to evaluate the performance of HRCA over regular FPGA device for reconfigurable computing by mixing of Look up tables (LUTs) and Programmable logic arrays (PLAs) architecture. The basis of the HRCA is that some parts of digital circuits are well-suited for execution with LUTs, but other parts help more from the PLAs structures. For several classes of high performance applications, HRCA offers significant savings in total computational delay comparison with a symmetrical FPGA which contain only LUTs. It also offers some improvements in logical area and power consumption. Experimental results based on MCNC benchmark circuit were performed on implemented HRCA CAD and compare between HRCA and symmetrical FPGA. Initially results indicate that noteworthy saving in computational delay and logic area of HRCA over symmetrical FPGA.

Key word

Reconfigurable computing, FPGA, CPLD, HRCA, CAD.

1. Introduction:

Progress in technology induces paradigm shifts in computing. Reconfigurable architectures have made a new area of research and innovation in reconfigurable computing because of its large flexibility and performance potential. Basically, there are two main principles in conventional computing for the execution of high performance applications. The first is Application Specific Integrated Circuit (ASIC) which designed for a specific application with fixed functionality, due to this it is very fast and efficient. Microprocessors are a far more flexible solution. Processors execute a set of instructions to perform a computation. By changing the software instructions, the functionality of the system is altered without changing the hardware. As we know that the microprocessors are the heart of most high performance computing architecture or platform. They provide a flexible computing platform and able to executing a large category of applications. Unfortunately, this flexibility decreases the performance of application. ASICs give an alternate solution which solves the performance issues of general purpose microprocessors. Hence, every

DOI : 10.5121/ijesa.2012.2312

ASIC has fixed functionality with greater performance over controlled set of applications. The second is reconfigurable computing which fill the gap between hardware and software. Reconfigurable computing utilizes hardware resources that can be tailored at run-time to give greater flexibility without compromising on performance. Reconfigurable computing devices agree to assemble both requirements i.e. flexibility and performance. Reconfigurable systems have evolved from Field Programmable Gate Arrays (FPGAs). Field-Programmable Gate Arrays (FPGAs) have experienced wonderful expansion in recent years and have become a multi-billion dollar industry. FPGAs are at least three times slower and demand more than ten times the silicon area when implementing the same function on a chip when compared to Standard Cells or Masked-Programmable Gate Arrays .

This happens because Standard Cells use simple wires to make interconnections between logic gates but in FPGAs, gates are connected with programmable switches. These switches have much larger resistance and capacitance and hence are slower than wires. In full fabricated FPGA chips, 10% of the FPGA is made of logic blocks and other 90% of the FPGA is made of the programmable interconnects network, which form a programmable routing architecture that provides random wiring between the logic blocks i.e. FPGA use only 10% logical resources for computation. So, ideally to improve the performance of an FPGA, we would like to use more logical chip resource for computation for any given circuit. Hybrid reconfigurable computing architecture (HRCA) combines the technology of FPGA (LUT) and CPLD (PLA). In this architecture, it can distribute the computations between different components (LUT/PLAs) of the system to improve the overall computing performance and logic concentration.

2. Related Research Work:

Field Programmable Devices (FPDs) face many challenges from lower speed-performance and less logic capacity in comparison to custom manufactured technologies, such as mask-programmed gate arrays. However, a lot of research has been devoted to improving FPD architecture. New architecture continues to emerge as main research in industry and academia with advanced total logic capacity and better speed-performance. A highlight of some recent research efforts on FPGA logic blocks is presented here.

Logic synthesis targeting FPGAs has been research extensively and numerous technology mapping approaches for LUT based FPGAs have been developed. These approaches have two main objectives, area and delay minimization [5].

In many reconfigurable embedded systems size, power and cost optimizations are the central goals. In those systems, the growing need of more computation power that contradict with size, power and cost optimization put a lot of pressure on researchers who must discover a good balance of all contradicting goals. In the July 2005 edition of the Altera Stratix Device Handbook[6], Stratix is an SRAM-based island-style FPGA containing many mixed computational elements. The main element is the logic array block (LAB), which contain 10 logic elements (LEs). The general architecture of the LE is much related to the structure that we try use to develop an HRCA, i.e. single 4-LUT function generator and a programmable register. In the FPGA research done by J.Rose, he focused on logic-block density. Assuming a LUT-based architecture, authors change the number of inputs to a LUT to measure the effects on implementation of a benchmark circuit set. Their conclusion is that LUTs with 4 or 5 inputs yield the best results in terms of chip area. We try to apply this result by using 4-LUTs, which are also found in commercial FPGAs such as the Altera Flex 10K, and the Xilinx Virtex. [7]. Xilinx 4000 Family was a popular first generation FPGA device family with 2,000 to 180,000 usable gates but Xilinx Virtex FPGAs, each CLB now contains four circuits similar to the earlier 4000 CLB in which interconnection network contains varying length row, column, and

neighbouring CLB interconnect structures that increases the logical area utilization for large fan in circuits [19]. Multicontext programming bits, a scheme that promises some savings in area efficiency and reconfiguration time for FPGAs proposed by E. Tau in their research [8]. In the research of Altera industry, they have recently introduced the new series of field programmable devices (FPDs) known as APEX (Advanced Programmable Embedded Matrix). Their main characteristic is the combination of LUTs and PLA like blocks on the same chip. APEX architecture contains embedded system blocks that can be configured to support pterms, memory blocks and content addressable memory cells (CAMs). The first APEX devices will offer 500,000 gates, but in future, they plan to include some more devices up to 2 million programmable gates [9]. In the study of J. He and J. Rose, called Heterogeneous FPGAs, they investigated FPGA architectures with logic blocks of two different sizes to see the effects on area efficiency of LUT-based FPGAs, in the same chip. In the result, they provide a saving of 15% in chip area by mixture of LUTs [12]. Most of the research focused on FPGAs based FPDs rather than CPLD based FPDs. There are very small effort has been published in the area of CPLDs research. Though, in the study of J. L. Kouloheris and A. El Gamal, they investigated and built FPDs using PLA based logic blocks. According to authors, an FPD based PLAs with 10 inputs, 12 Pterms, and 3 outputs achieves about the same level of logic density as FPGAs based on 4-LUTs, but this unchanging share decreases the flexibility of FPGA. We are not aware of any industrial product that is based on such FPDs based PLAs. [13]. S. Wilton et. al. explain the memory modules with variable aspect-ratio that could be incorporated as separate blocks in an FPGA. This design is not orthogonal to the Hybrid FPGA, and so memory blocks could also be included in our architecture for delay minimization [14]. Li- Guang investigated Hybrid FPGA architecture, which modify the CBs which implement the AND plane of PLA. They found that a mixture of LUTs and PLAs provide and area saving of about 46% but did not explain about computational delay [20]. Jason found that Flowmap technology mapping algorithm reduces the maximum no of LUT for delay optimization [20].

In the Altera data sheet, implementations of partial logic circuits with PLAs are also founded in commercial FPGAs. An Embedded System Block (ESB) of Altera APEX20K can be configured as a PLA with 32 inputs, 32 product terms(ptersms) and 16 outputs [10]. In the same way, in every Xilinx Virtex II slice has a devoted OR gate named ORCY, with which a Configurable Logic Block (CLB) can implement 2 product terms with 16 inputs [11]. These architectures benefit to the logic density of FPGA. However, they are not optimized specifically for implementing PLA. Microelectronics Center of North Carolina (MCNC) benchmark suite is used as logic synthesis and optimization benchmark. The benchmark suite has standardized libraries with representative circuit designs ranging from simple circuits to advanced circuits. MCNC benchmarks are very popular in academic research. The FPGA researcher relies a lot on benchmarks to evaluate performance of their hardware and software solutions. Hence, standard and fair benchmarking practices are essential to calculate FPGA architecture, design, configuration, verification and validation of FPGA device and verify their potential to support target applications. In this paper, We try to use some MCNC bench mark to evaluate the architectural performance of HRCA.

3. Reconfigurable Computing:

Reconfigurable computing is a relatively new field of research and development, the first research beginning in the late 1980's. It is an effort to overpass the traditional gap between hardware and software within the computing field. Reconfigurable Computing is emerging as an important replacement for computing algorithms evolved from FPGA. The key feature of this is that it incorporates the performance of ASIC and flexibility of GP (General-Purpose) processors. These devices are composed of logic device FPGA (Field-Programmable Gate Array), which helps in determining functionality of the system from programmable configuration bits. Modern high-end FPGAs can have tens of millions of configuration points.

FPGA consist of matrix of logic blocks and interconnection network between the logic block. The functionality of logic blocks and its interconnection can be customized by downloading bits of reconfiguration data on the given hardware[3][4]. In FPGA, the resources needed for the computation of an application will be built as components to be downloaded to the device at run-time. The generation of such components is called logic synthesis. Reconfigurable devices are usually used in three different ways:

3.1 Rapid prototyping: The reconfigurable device is used as an emulator for another digital device, generally an ASIC.

3.2 Non- regularly reconfigurable systems: The reconfigurable device is integrated in a running system where it is used as an application-specific processor. These systems are usually individual systems.

3.3 Regularly reconfigurable systems: This third category comprises systems, which are frequently reconfigured. Those systems are usually coupled with a host processor, which is used to reconfigure the device and control the complete system

Depending on the time at which the configuration sequences are defined, the computation and configuration flow of data on reconfigurable devices can be classified into two categories:

- a) **Compile-time reconfiguration:** In this case, the computation and configuration sequences as well as the data exchange are defined at compile time and never change during a computation.
- b) **Run-time reconfiguration:** The computation and configuration sequences are not known at compile time. Request to implement a given task is known at run-time and should be handled dynamically.

4. Architecture of CPLD/FPGAs:

The two main types of programmable devices, field programmable gate arrays (FPGA) and complex programmable logic devices (CPLD), are both extensively used. Each device contributes particular strength in the development of reconfigurable system. FPGAs programmed with static RAM technology are usually based on lookup tables. A look up tables (LUT) is a group of memory cell, which contain all possible results of a given function for a given set of input values. An n-input LUT can be used to implement up to 2^{2^n} different functions, each of which can take 2^n possible values[3].

In FPGA, an LUT physically consists of a set of SRAM cells to store the value and decoder that is used to access the correct SRAM location to retrieve the result of functions. Their main strengths are very high logic capacity in the range of hundreds of thousands of equivalent logic gates and good speed performance up to 20-50 MHz system clock rates. SRAM based LUT is used in the most commercial FPGA as function generator[2][18].

On the other hand, CPLDs consist of a set of macro cells, input/output blocks and an interconnection network. A macro A macro cell typically contains several PLAs and flip flops. Programmable logic arrays (PLA) consist of a plane of AND-gates connected to a plane of OR-gates and both the plane can be programmed by the user.

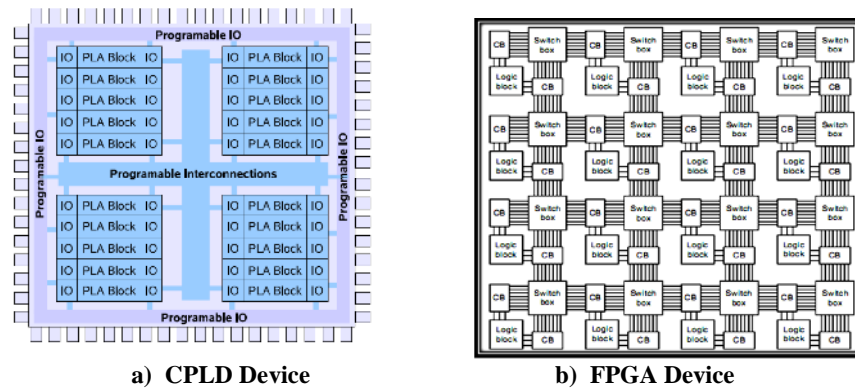


Fig. 1: Structure of programmable logic devices

Their characteristics include medium capacity, in the range of a few thousand gates, and ultrahigh speed performance, sometimes in excess of a 140-180 MHz system clock rate[16]. HRCA merge the two common technologies used in programmable logic devices: lookup table (LUT) based FPGA and PLA – like logic cell based CPLD. The most important initiative is to find out what function are suitable to be implemented on which logic resources. Logic resources include LUTs as well as product term (pterm) like PLA logic cells. In this paper, we try to analysis the computational and logic performance of HRCA in comparison with an architecture containing only LUTs. It indicates that the modified architecture offers significant savings in computational delay as well as total chip area. Also, the HRCA can reduce the depth of circuits implemented in the FPGA, which may provide improvements in data communication speed and performance.

5. Design Process of HRCA:

Reconfigurable computing systems regularly show remarkable performance and strength in the term of high speed, reduced energy and power consumption. The advances in high-performance computing and reconfigurable computing, based on field programmable gate arrays (FPGAs), form the basis for a new paradigm, called reconfigurable supercomputing. This can be achieved through hybrid of LUTs and PLAs of programmable logic devices.

FPGA programmed with SRAM technology are usually based on Look-Up Tables (LUTs). For implementing random logic circuits in LUT based FPGA, the cost of LUTs increases exponentially according to the inputs of circuits. So LUT is suitable for **low fan-in logic circuits**. In CPLDs are based on Programmable Logic Arrays (PLAs). The PLA usually have tens of inputs and is appropriate for **high fan-in logic circuits**. CPLDs are typically faster and have more predictable timing than FPGAs because FPGAs are generally more dense and contain more flip flops and registers than CPLDs[1][15].

As in most of the applications, due to fine granularity of FPGA, most of the Configuration Box's (CB) are never used. Because of this, many Logic Blocks (LB) are used to implement logic functions. So a large percentage of chip area is wasted. To rectify above draw backs, it has been tried to modify the structure of connection Box (CB) which will facilitate to work in LUTs mode or PLAs mode for algorithm computation depend upon circuit fan-in. The modified CB architecture is shown in Fig 2. For this architecture, Xilinx Virtex-E Configurable Logic Blocks (CLB) may used to evaluate the performance of HRCA. The basic building block of the Xilinx Virtex-E Configurable Logic Blocks (CLB) is the logic cell (LC). An LC includes a 4-input function generator, carry logic, and a storage element. The output from the function generator in each LC drives both the CLB output and the D input of the flip-flop. Each Virtex-E CLB contains

four LCs, organized in two similar slices, A devoted LUT C was designed which may implement the OR plane of PLA and CBs of FPGA are engaged to implement the AND plane of PLA to make a complete PLA operation

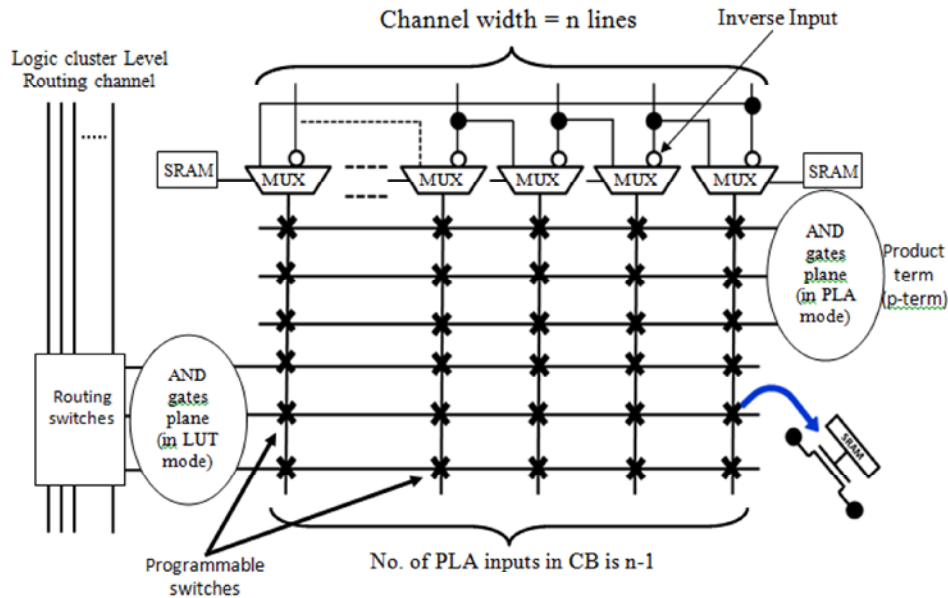


Fig. 2: Modified architecture of Connection Box (CB) in HRCA

6. Computational Optimization factor for HRCA:

HRCA based on the mixture of two existing configuration technologies: Field Programmable Gate Arrays (FPGAs) based on Look Up Tables (LUTs), and Complex Programmable Logic Devices based on PALs/PLAs. As we know PLAs are suitable for the operation of large fan-in logic circuits, while LUTs are used to execute low fan-in logic circuits. As in most of the applications, due to fine granularity of FPGA, most of the connections in connection box (CB) are never used and many logic blocks (LB) are used. So a large percentage of chip area is wasted. So, the logic blocks and routing switches strongly influences an FPGA's computational speed and logic density. In the subsequent section, the effect of computation speedup, logic concentration and power optimization are discussed. These effects have a significant impact on overall performance of HRCA.

6.1 Computation Speedup:

In order to raise the algorithm computation speedup, the concentration of the circuit must be reduced. There are three primary definitions of speed depending on the context of the problem: throughput, latency, and timing. In the context of processing data in an FPGA, throughput refers to the amount of data that is processed per clock cycle. A common metric for throughput is bits per second. Latency refers to the time between data input and processed data output. The typical metric for latency will be time or clock cycles. Timing refers to the logic delays between sequential elements. When we say a design does not "meet timing," we mean that the delay of the critical path, that is, the largest delay between flip-flops (composed of combinatorial delay, clk-to-out delay, routing delay, setup timing, clock skew, and so on) is greater than the target clock period. So as we know CPLDs (PLA) are characteristically faster in computation and have more

predictable timing than FPGAs (LUTs). So for large fan in , we utilize the CB circuit in PLA mode to maximize the computation speedup of HRCA.

6.2 Logic Concentration:

With the integration of the FPGA and CPLD cores technology and logic circuits, the area of the chip is decreased. Circuit-level reduction as performed by the synthesis and layout tools refers to the minimization of the number of gates in a subset of the design and may be device specific. Preliminary results indicate that compared to LUT-based FPGAs the Hybrid offers savings of more than a factor of two in terms of chip area. As we know FPGAs are generally denser and contain more flip flops and registers than CPLDs. If we distribute the application according to its fan-in into Hybrid architecture so that logic circuits can be configured as either PLAs or LUTs with slightly area reduction. Proposed HRCA design show that major chip area is reduced using HRCA CAD [17].

6.3 Power Optimization

FPGAs are power-hungry beasts and are typically not well suited for ultralow-power design techniques. A number of FPGA vendors do offer low-power CPLDs (complex programmable logic devices), but these are very limited in size and capability and thus will not always fit an application that requires any respectable amount of computing power. HRCA will offer significant savings in power consumption in circuit by using the proper mixing of CPLD and FPGA techniques.

In CMOS technology, dynamic power consumption is related to charging and discharging parasitic capacitances on gates and metal traces. The general equation for current dissipation in a capacitor is

$$I = V * C * f$$

where I is total current, V is voltage, C is capacitance, and f is frequency. Thus, to reduce the current drawn, we must reduce one of the three key parameters. In Hybrid FPGA design, the voltage (V) and frequency (f) are usually fixed. This leaves the parameters C to manipulate the current. The capacitance (C) is directly related to the number of gates that are toggling at any given time and the lengths of the routes connecting the gates and registers.

As we know that FPGAs are generally denser and contain more logic gates and registers than CPLDs. So power consumption is more in FPGA than CPLD. So HRCA shows the remarkable power saving as compare to the regular FPGA.

7. Performance evaluation of HRCA over regular FPGA:

In the performance evaluation of HRCA, the modified architecture requires a CAD tools for performing technology mapping, routing and placement of the circuits. We used HRCA CAD for this performance evaluation experiment[17]. FlowMap[21] algorithm is used for technology-mapping to map each circuit into 4-LUTs and flip flops. PLAs Mapping algorithm is used to map gate level network into PLA block. T-VPACK program then maps this netlist of 4-LUTs and flip flops into logic clusters. Finally, the VPR placement and routing tool [17] is used to place and globally route the circuit.

The HRCA CAD tool flow transforms an input circuit, usually described in an HDL like VHDL or Verilog, into a netlist (bitstream) suitable for the HRCA. Previous research has showed the

effect of LUT input on FPGA logic efficiency and it finished that 4 inputs LUT provide good result. Our method is experimental; we use 20 of the largest MCNC benchmark circuit's[22] to obtain HRCA area and critical-path delay estimations over regular FPGA. The MCNC benchmarks in the format of BLIF are optimized by SIS.

These papers also see the effect of logic cluster size in HRCA performance. The logic cluster blocks area is implemented by BLE. Figure 3 show that the logic clusters containing between 4 to 10 BLEs are all achieve good delay performance . Figure 4 and 5, show the computational delay and logic area of the benchmark circuit. In both cases, the size of logic cluster is 4. These tools are providing a rough calculation of accepted delay logic gain in HRCA as compare to a 4-LUT-based FPGA.

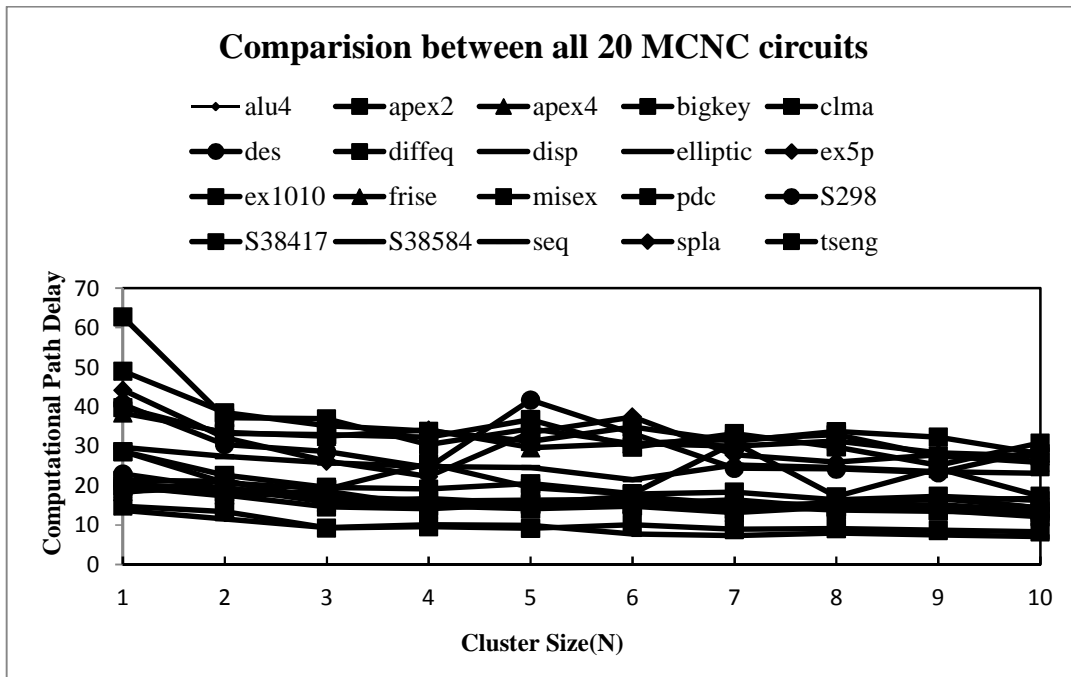


Figure 3: Effect of Cluster Size over computational delay

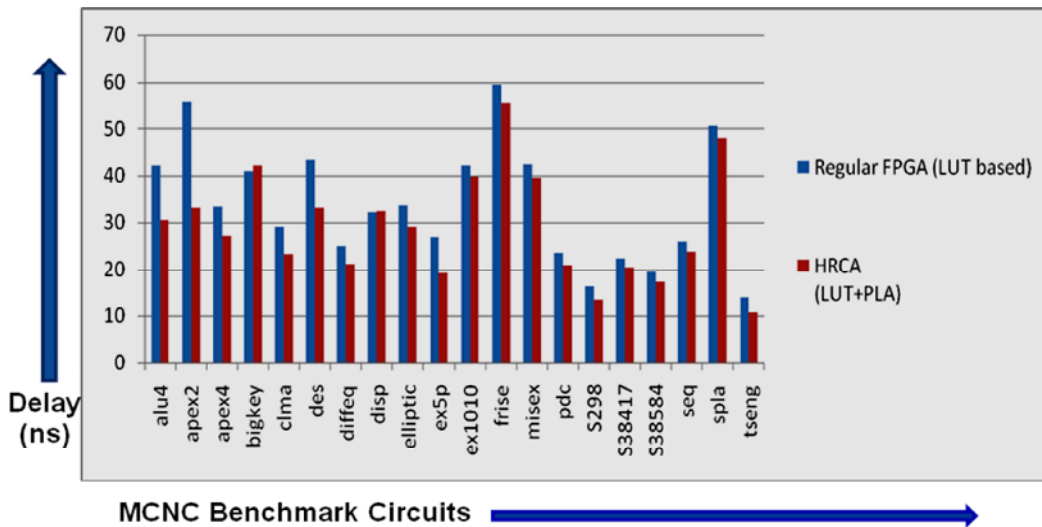


Figure 4: Circuit delay of MCNC circuit's for regular FPGA and HRCA.

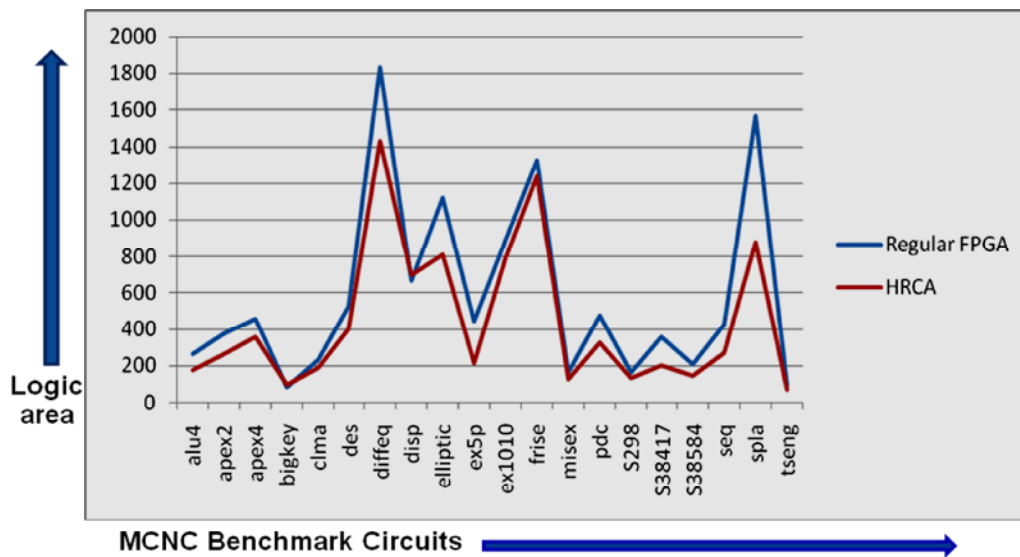


Figure 5: Logic area of MCNC circuit's for regular FPGA and HRCA.

8. Conclusion

We have reviewed the factors which determine the overall performance of hybrid reconfigurable computing architecture over traditional FPGA. We determined the delay and logic optimization by using FlowMap and VPR tool on various MCNC benchmark circuits and reviewed other factors also, namely Speed-up, Area and Power. We also discuss the logic cluster size for circuit delay for our proposed HRCA architecture. In our experimental results, HRCA can provide not only excellent delay trade-off curve but also in area optimization. In future, we would be evaluating the runtime power optimization approach of our proposed HRCA with traditional FPGA using MCNC benchmarks circuit

References:

- [1]. Alireza Kaviani and Stephen Brown, The Hybrid Field Programmable Architecture, IEEE Design & Test of Computer, page 74-83(1999).
- [2]. Bob Zeidman, Designing with FPGAs and CPLDs, CMP Books Canada,(2002).
- [3]. Christophe Bobda, Introduction to Reconfigurable Computing Architectures, Algorithms, and Applications, University of Kaiserslautern, Germany, Springer Book,(2007).
- [4]. Scott Hauck and Andre DeHon, Reconfigurable Computing: The theory and Practice of FPGA , Elsevier Book,(2008).
- [5]. J. Cong and Y. Ding, "Survey Paper - Combinational Logic Synthesis for LUT based Field Programmable Gate Arrays," ACM DAES, vol. 1, no. 2, pp.145-204,(1996).
- [6]. The Altera Stratix Device Handbook, 2005 (available online at <http://www.altera.com>).
- [7]. J. Rose et al., "Architecture of FPGA: The Effect of Logic Block Functionality on Area Efficiency," IEEE J. Solid-State Circuits, Vol. 25, No. 5, pp. 1217-1225(1990).

- [8]. E. Tau et al., "A First Generation FPGA Implementation," Workshop Proc. on FPD, Montreal, pp. 138-143(1995).
- [9]. A. Stansfield and I. Page, "The Design of a New FPGA Architecture," Proc. in Field-Programmable Logic and Applications, Univ. of Oxford, London,(1995).
- [10]. Altera Corporation, APEX20K PLD Family Data Sheet(2004).
- [11]. Xilinx Corporation, Virtex-E Platform FPGAs Complete Data Sheet (2005).
- [12]. J. He and J. Rose, "Advantages of Heterogeneous Logic Block Architectures for FPGAs," Proc. in Custom Integrated Circuits, San Diego, pp. 7.4.1 - 7.4.5.(1993).
- [13]. J. L. Kouloheris and A. El Gamal, "PLA-based FPGA Area vs. Cell Granularity," Proceedings of the Custom Integrated Circuits Conference,(1992).
- [14]. S. Wilton, J. Rose, Z. Vranesic, "Architecture of Centralized Field-Configurable Memory," ACM International Symposium on FPGA , Monterey Bay, CA,(1995).
- [15]. M. Nadjarbashi, s. M. Fakhraia et al, On routing architecture for Hybrid architecture, scientia Iranica, Vol. 11 No. 3, page 159-164,(2004).
- [16]. Sunil Kr. Singh , et al, System level Architectural Synthesis & Compilation Technique in Reconfigurable Computing System,(ESA'10), USA, page 109-115,(2010).
- [17]. Sunil Kr. Singh , et al , IJCA, Vol. 24, No. 4, pp 50-54,June(2011).
- [18]. Stephen Brown and Jonathan Rose, Architecture of FPGAs and CPLDs: A Tutorial, University of Toronto.
- [19]. James O. Hamblen, Tyson S. Hall, Michael D. Furman, Rapid prototyping of digital systems - Quartus II ed, Springer Book (2006).
- [20]. Li-Guang et . al, A Novel Hybrid FPGA Architecture, IEEE (2006).
- [21]. Jason Cong, FlowMap: An Optimal Technology Mapping Algorithm for Delay optimization in LUTbased FPGA Designs, IEEE Transaction on CAD, vol. 13, No. 1.(1994).
- [22]. S. Yang, "Logic Synthesis and Optimization Benchmarks, Version 3.0," Tech. Report, Microelectronics Center of North Carolina (1991).