CAD Optimization Technique in Reconfigurable Computing System using Hybrid Architecture

Sunil Kr. Singh
Ph.D, Research scholar
Uttarakhand Technical University
Uttarakhand, INDIA

R. K. Singh
Professor
Uttarakhand Technical University
Uttarakhand, INDIA

M. P. S. Bhatia
Professor
NSIT, University of Delhi
New Delhi, INDIA

ABSTRACT
Design automation or computer-aided design (CAD) for reconfigurable computing system is giving a new concept of research and development in system design for present and future technological environment. The basic ability of reconfigurable computing is to perform computations in hardware to increase performance, while retaining the flexibility of application software. The purpose of this paper is to meet the demand of a suitable design flow for a reconfigurable computing system using Hybrids architecture. 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. In this paper, we try to propose computer-aided design (CAD) optimization technique for Hybrid Reconfigurable Computing Architecture (HRCA), which combines FPGAs and CPLDs. 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. The new architecture HRCA offers significant savings in total logic area comparison with an architecture containing only LUTs. It also offers some improvements in speed performance. This paper focuses on suitable optimization techniques and design flow that will cover all major steps in system design which includes: routing and placement, circuit clustering, technology mapping and architecture-specific optimization, physical synthesis, RT-level and behaviour-level synthesis.

Keywords: Computer-aided design (CAD), Reconfigurable Computing, FPGA, CPLD, Hybrid Architecture.

1. INTRODUCTION
Research in architecture of computer systems has always been a central concern of the computer science and computer engineering communities. In the computer and electronics world, we are used to two different ways of performing computation: hardware and software. Computer hardware, such as application-specific integrated circuits (ASICs), provides highly optimized resources for quickly performing critical tasks, but reconfigurable computing is to perform computations in hardware to increase performance, while retaining the flexibility of application software, such hardware know as reconfigurable devices.

Field Programmable devices (FPDs) have emerged as a competitive alternative to Application Specific Integrated Circuits (ASICs) to implement designs and their popularity have grown in recent years. For the past twenty years, programmable devices have been widely used in digital systems. There are two main types of programmable devices, Field Programmable Gate Arrays (FPGA) and Complex Logic Device (CPLD), are both extensively used. Each device contributes particular strength in the development of reconfigurable system. In this article, we try to evaluate CAD optimization technique for our proposed Hybrid Reconfigurable Computing Architecture (HRCA), which combines FPGAs and CPLDs[8].

2. RECONFIGURABLE COMPUTING
Reconfigurable Computing is emerging as an important replacement for computing algorithms. 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 Field Programmable devices(FPDs), which helps in determining functionality of the system from programmable configuration bits. Modern high-end FPDs can have tens of millions of configuration points.

Conventional computing for the execution of algorithm has primary two methods, one is using Von-Neumann computing in which programming of microprocessor/microcontroller using S/W. the second method is making application specific processor or integrated circuits (figure: 1).

The first one is more flexible solution with performance degradation. in the later method the system has been designed for particular application , may not be cost effective to modify or add more feature. So the use of reconfigurable computing system (RCS) which fill the gap between H/W(ASIC/ASIP) & S/W(Microprocessor) approaches. Because conventional computing system have fixed H/W and variable algorithm (S/W) to implement a system but In RCS, it has both H/W as well as algorithm(S/W) are variables[7].

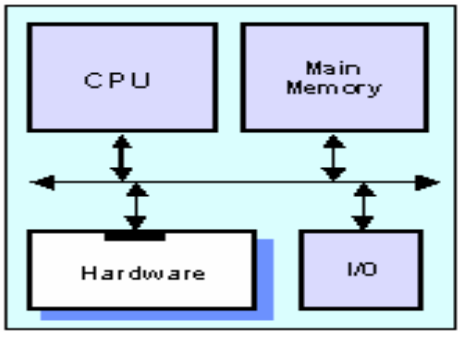


Fig 1: A general ASIC computational system

Reconfigurable computing has demonstrated greater performance than the currently available microprocessors and General-Purpose Processors.

The different applications of reconfigurable systems are owing to its various advantages like:

- **Flexibility:** The reconfigurable system can adapt to varied system architectures.

- **Cost:** System cost is reduced as the number of components is reduced and the same hardware is provided with multiple functionalities.
- **Performance:** The performance is increased many folds with the use of reconfigurable system as direct interfacing is used.

3. ARCHITECTURE OF FPGA

FPGA (Field programmable Gate Array) is a programmable device consisting of three main parts, which are programmable functional units or logic blocks or configurable logic blocks, Programmable interconnection network and set of input and output cells (Figure 2). A circuit is implemented on an FPGA by configuring the logic blocks (functional block) and the routing appropriately. Most FPGAs use static RAM (SRAM) cells to achieve their programmability [3].

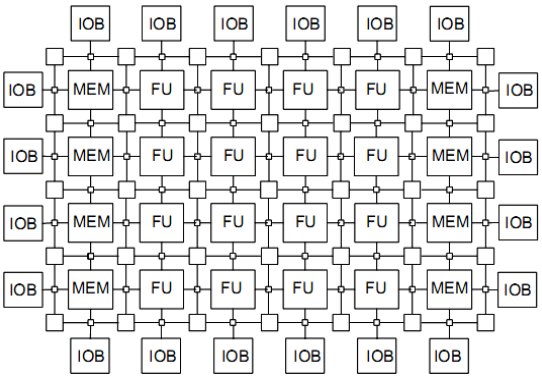
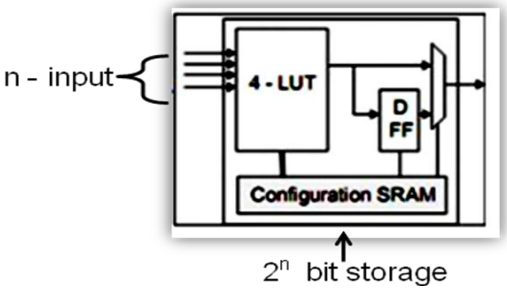


Fig 2: General FPGA Architecture

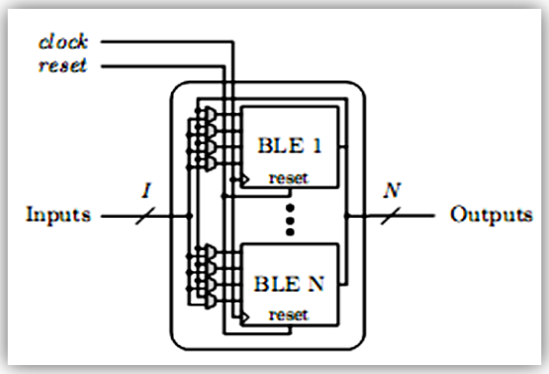
FPGAs can be programmed once or several times depending on the technology used. There are two types of technologies used in FPGAs programming: antifuse and SRAM based. Whereas the antifuse concept is limited to the recognition of interconnections, the memory-based concept (SRAM) well used for the computation as well as the interconnections of circuit [6].

3.1 BLEs

In FPGA, Logic blocks are made up of basic logic elements (BLEs). Basic logic element(BLE) usually consists of one or more SRAM based n input look up tables(LUTs) and one or more flip flop (where n is a number between three and six). 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. Therefore, an n-input LUT (Look up table) must provide 2^n SRAM cells for storing the possible values of an n-input function [3]. So the number (N) of required BLEs for I input function is N= (I-2)/2. (figure: 3 a, b)



(a)



(b)

Fig 3: (a) Logic Blok (b) Logic Cluster

3.2 SRAM

SRAM bit decides that circuit switch is closed or open, If SRAM bit is '1', then switch is closed and connection is made If SRAM bit is '0'then switch is open and connection remain open[5]. The netlist (bit stream) file developed by the software tool and directly configures the connection by placing the corresponding bits into the SRAM (Figure 4).

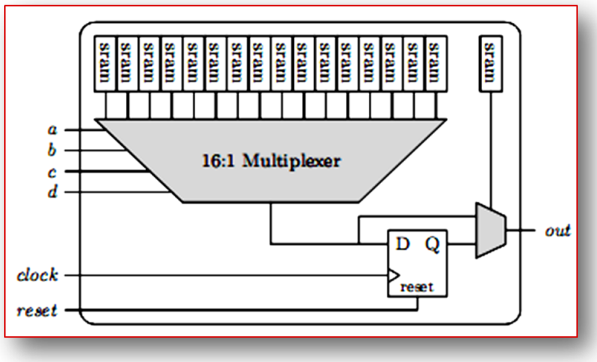


Fig. 4: LUTs Architecture with SRAM

4. ARCHITECTURE OF CPLD

PALs and PLAs are only available in small sizes, equivalent to a few hundred logic gates. Generally, It used for small logic circuit as simple programmable logic devices (SPLD). For large logic circuits, complex programmable logic devices (CPLD) can be used. A CPLD consists of a set of macro cells, input/output blocks and an interconnection network each macro cell typically contains several either PLA or PAL circuit arrangement, it builds on SPLD architecture and creates a much larger design.(Figure 5)

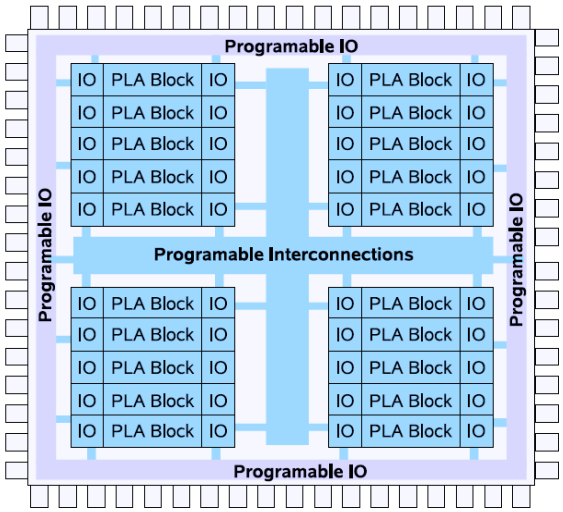


Fig 5: General CPLD Architecture

CPLD architectures integrate multiple SPLDs onto a single chip and provide programmable interconnect SPLD blocks together [3][8]. CPLD provide logic capacity up to 50 SPLD devices, but it is somewhat difficult to extend these architectures to higher densities. In CPLDs, generally flash-EPROM and EEPROM technologies are used for circuit switching.

4.1 PLA/ PALs

Programmable logic arrays (PLA) and programmable array logic (PAL) consist of a plane of AND-gates connected to a plane of OR-gates. The inputs signals as well as their negations are connected to the inputs of the AND-gates in the AND-plane. The outputs of the AND-gates are use as input for OR-gate in the OR-plane whose outputs correspond to the required hardware circuit [8]. The connections in the two planes are programmable by the user in PLA but in PAL only AND -plane can be programmed.(figure 6)

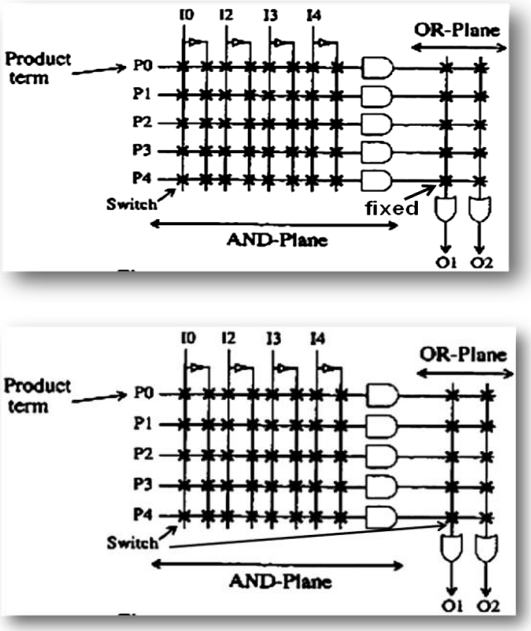


Fig 6: PAL and PLA Architecture

5. HYBRID RECONFIGURABLE COMPUTING ARCHITECTURE (HRCA)

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** [4]. 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][5].

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 Box (LB) are used. So a large percentage of chip area is wasted. To rectify above draw backs, it has been tried to develop a new structure of connection box (CB) which will facilitate to work in LUTs mode or PLAs mode for algorithm computation depend upon circuit fan-in. In this way we need a proper CAD optimization technique for new way logic synthesis and optimization of hybrid architectures (Combine FPGA & CPLD) to maximize the computation performance and decrease the logic concentration(figure 7)[2].

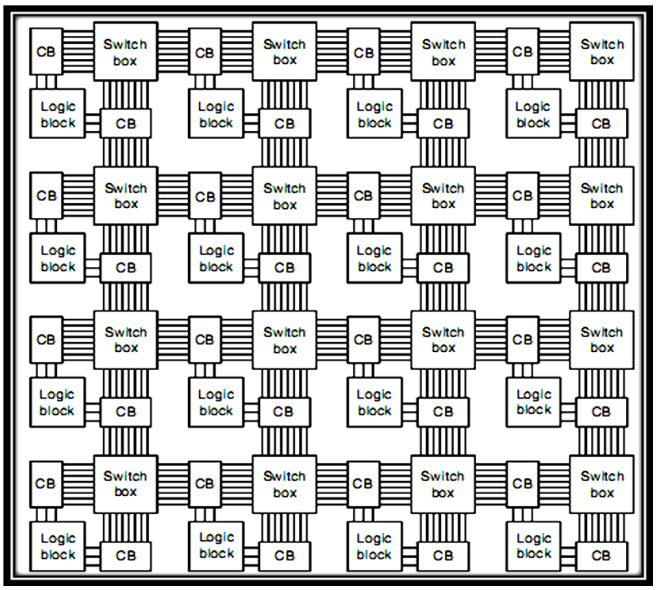


Fig 7: HRCA Architecture

6. CAD OPTIMIZATION TECHNIQUE

6.1 CAD for FPGA

FPGAs have become an increasingly popular medium due to the high costs associated with application specific integrated circuit (ASIC) implementations. FPGA Implementations usually enjoy reduced development and verification times, and avoid the large costs involved in chip fabrication. Computer Aided Design (CAD) software makes it easy to implement a desired logic circuit by using a programmable logic device, lile field-programmable gate array (FPGA) chip. To apply a circuit on the current generation FPGAs, CAD tools are needed which can produce the netlist for the SRAM cells of LUTs. generally the circuit design is provided using, VHDL or Verilog, or any other HDL. Presently, The available CAD tools for the FPGAs read this input from HDL programming for FPGA design, the

necessary four tasks in CAD flow for implementing a FPGA digital system are Synthesis, fitting, placement and routing[11]. As shown in figure 8.

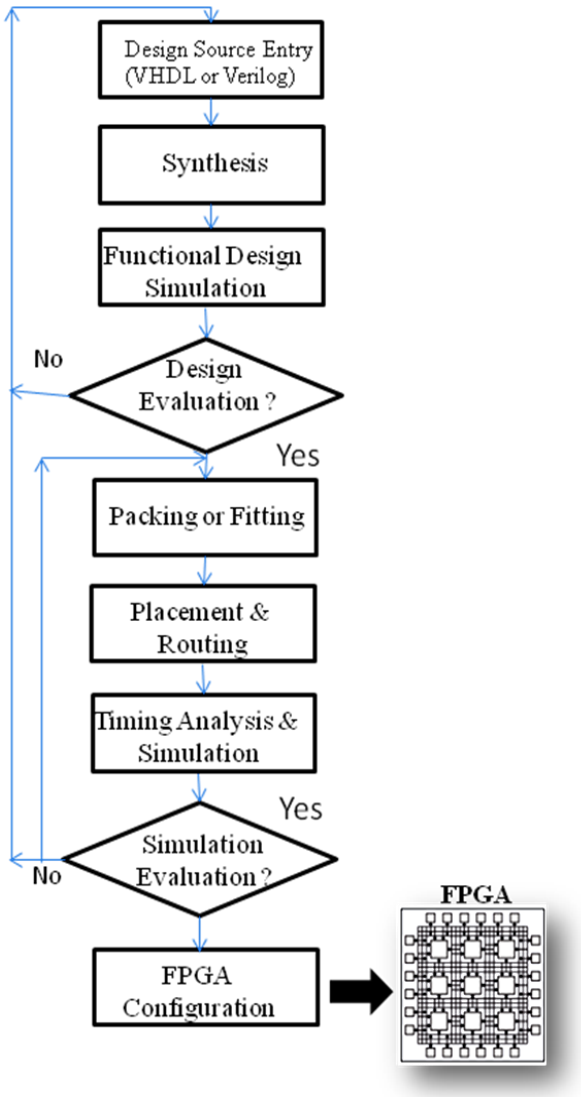


Fig 8: CAD flow for FPGA Architecture

6.2 CAD HRCA

The CAD tools used for HRCA are VPR (Versatile Placement and Route), for placement and routing, and T-V Pack for clustering of the Logic blocks and connection block (CBs) of proposed architecture [9] [10]. The key difference between an FPGA and HRCA is in the structure of the logic concentration. In an FPGA, each logic block (LUTs) element determines the functionality of application but in HRCA, available logic bloc and connection block determine the functionality of application. In this way of modification, architecture(HRCA) will offers significant savings in total chip area (size) and enhance total computational speed of system (minimize delay). therefore, there is significant need for flexible CAD tools and flow that can target on wide variety of FPGA architectures efficiently like HRCA, (as given in Figure 9) which typically evaluates the utility of HRC architectural features experimentally and hence allow fair comparisons of the available architectures.

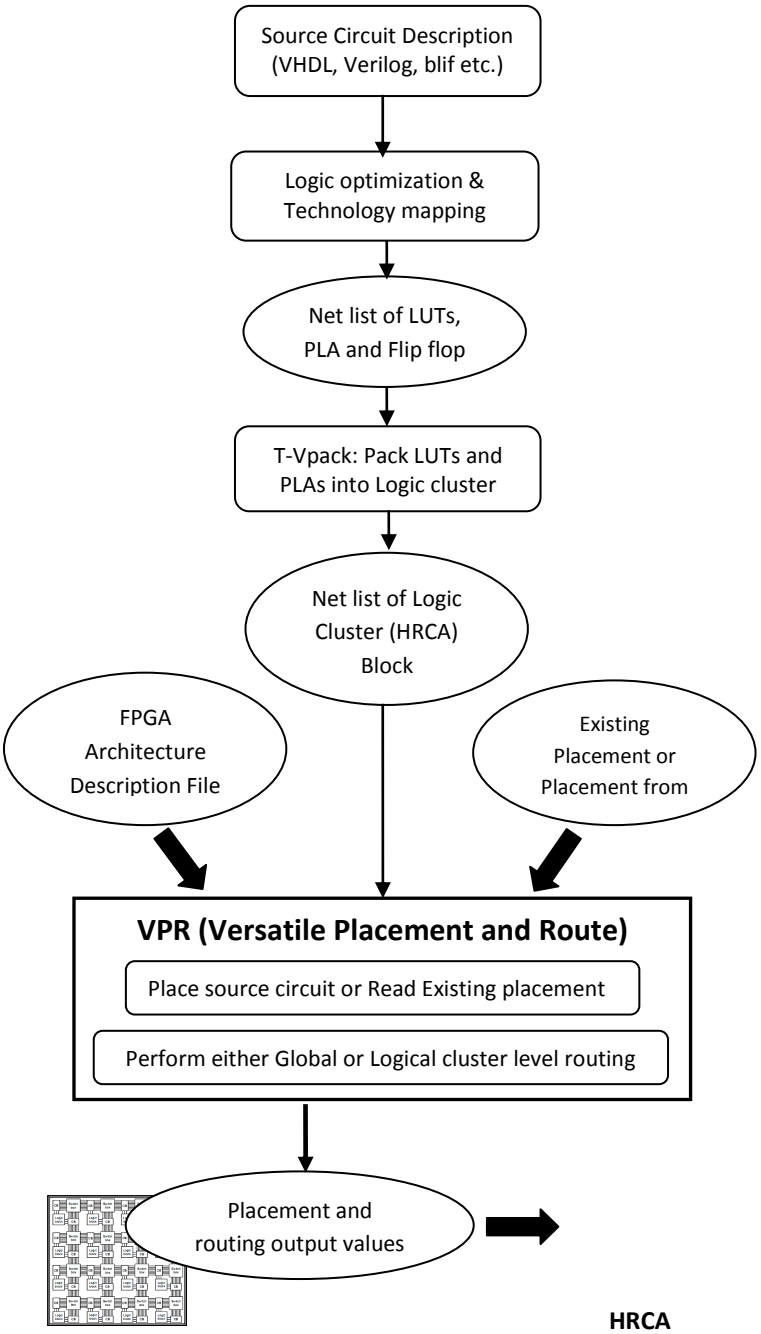


Fig 9: CAD flow for HRCA Architecture

MCNC benchmark circuits would be use for logic synthesis, technology mapped, packing, placed and routed and optimization onto hybrid reconfigurable computing architectures (HRCA) using FPGA [5]. It measures the architecture's excellence, like computational speed and area.

7. CONCLUSION

Complex Programmable Logic Devices (CPLDs) and Field Programmable Gate Arrays (FPGAs) have become a critical part of every system design. The ability to combine LUTs and PLAs into existing system design and to test designs, fixes bugs, are the major challenges. This standards device shows the significant outcome in logic concentration and computational speedup of digital system which is attractive field of all researchers in system design. In this way, a feasible

CAD flow for HRCA using FPGAs was presented. The CAD flow modifies the existing FPGA CAD flow in a number of ways. A new step VPR, adaptation, is introduced into the CAD flow. Technology mapping and optimization which was previously called once is now called twice, once before adaptation and once after adaptation. Besides, since the structure of an HRCA using FPGA circuit is considerably different from an FPGA circuit, so new CAD flow and optimization are essential. The HRCA using FPGA architecture and CAD flow were studied in detail and in the future to evaluate its performance experimentally.

8. REFERENCE

[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] Li-Guang Chen, Kan-Wen Wang et al, A Novel Hybrid FPGA Architecture, State Key Lab of ASIC & System, Fudan University, Shanghai 201203, China.

[5] M. Nadjarbashi, s. M. Fakhraia et al, On routing architecture for Hybrid architecture, scientia Iranica, Vol. 11 No. 3, page 159-164, 2004.

[6] Scott Hauck and Andre DeHon, Reconfigurable Computing: The theory and Practice of FPGA , Elsevier Book, 2008.

[7] Sunil Kr. Singh , R. K. Singh , M. P. S. Bhatia, System level Architectural Synthesis & Compilation Technique in Reconfigurable Computing System, World Comp - 2010 (ESA'10), Las Vegas Navada USA, page 109-115,2010.

[8] Stephen Brown and Jonathan Rose, Architecture of FPGAs and CPLDs: A Tutorial, University of Toronto.

[9] VPR and T-VPack1 User's Manual (Version 4.30), 2000.

[10] VPR and T-VPack1 User's Manual Summer, 2008.

[11] Xiaoxiang Shi, FPGA CAD Research: An Introduction, Department of Computer Science , Xidian University, 2005.