A Novel FPGA Architecture Using Hybrid LUT/MUX

R.K.Gayathri (Me Vlsi Design), Dr. M. DhamodaranM.E,Ph.d,

ECE Department, Associate professor/ECE,
Syed Ammal Engineering College, Ramanathapuram.

ABSTRACT: *In this paper,Field programmable gate arrays (FPGAs) has emerged as a promising solution for approximate computing. FPGA used for recognition, mining, and search applications. Approximate computing achieve the energy efficiency,it produces inaccurate result rather than the guaranteed accurate result. Compared with most prior works on approximate computing, which target approximate processors and arithmetic blocks. Hybrid configurable logic block architectures for field programmable gate arrays that contain a mixture of lookup tables and hardened multiplexers are evaluated toward the goal of higher logic density and area reduction.*

I INTRODUCTION

FPGA provide the next generation in the programmable logic devices. The word Field in the name refers to the ability of the gate arrays to be programmed for a specific function by the user instead of by the manufacturer of the device. The word Array is used to indicate a series of columns and rows of gates that can be programmed by the end user. As compared to standard gate arrays, the field programmable gate arrays are larger devices. The basic cell structure for FPGA is complicated than the basic cell structure of standard gate array. The programmable logic blocks of FPGA are called Configurable Logic Block (CLB).

The FPGA architecture consists of three types of configurable elements-

(i) IOBs – a perimeter of input/output blocks
(ii) CLBs- a core array of configurable logic blocks
(iii) Resources for interconnection

The IOBs provide a programmable interface between the internal; array of logic blocks (CLBs) and the device's external package pins. CLBs perform user specified logic functions, and the interconnect resources carry signals among the blocks.

A configurable program stored in internal static memory cells determines the logic functions and the interconnections. The configurable data is loaded into the device during power-up reprogramming function. FPGA devices are customized by loading configuration data into internal memory cells. The FPGA device can either actively read its configuration data out of an external serial or byte-wide parallel PROM (master modes), or the configuration data can be written to the FPGA devices (slave and peripheral modes).

The MUX-based logic blocks for the FPGAs have seen success in early commercial architectures, such as the Actel ACT-1/2/3 architectures, and efficient mapping to these structures has been studied [12] in the early 1990s. However, their use in commercial chips has waned, perhaps partly due to the ease with which logic functions can be mapped into LUTs, simplifying the entire computer aided design (CAD) flow. Nevertheless, it is widely understood that the LUTs are inefficient at implementing MUXs, and that MUXs are frequently used in logic circuits. To underscore the inefficiency of LUTs implementing MUXs, consider that a six input LUT (6-LUT) is essentially a 64 to-1 MUX (to select 1 of 64 truth-table rows) and 64-SRAMconfiguration cells, yet it can only realize a 4-to-1 MUX.4 to 1 MUX consist of four data inputs and two select inputs.

In this paper consist of six-input LE based on a 4-to-1 MUX, MUX4, that can realize a subset of six-input Boolean logic functions, and a new hybrid complex logic block (CLB) that contains a mixture of MUX4s and 6-LUTs. A Hybrid configurable logic block architectures for field programmable gate arrays that contain a mixture of lookup tables and hardened multiplexers are evaluated toward the goal of higher logic density and area reduction.

The rest of this brief introduces the FPGAs in section I. Next the related works and an approximate computing methodology for FPGA-based designare considered in Section II. Then, in Section III, the proposed is presented. Section IV and V presents a experimental Results and performance analysis to illustrate the effectiveness of the hardware security approach. Finally, the conclusions are summarized in Section VI.

II RELATED WORKS AND AN APPROXIMATE COMPUTING METHODOLOGY

The memoization architecture generator contains a memorization wrapper generator that generates the architectural blocks needed for memoization. It takes similarity measure, threshold, user inputs, and user selection, which are present in the configuration file, as the input. The similarity measure is defined by the user to compare two temporally spaced inputs; threshold is the threshold value within which the result of similarity measurement must exist.

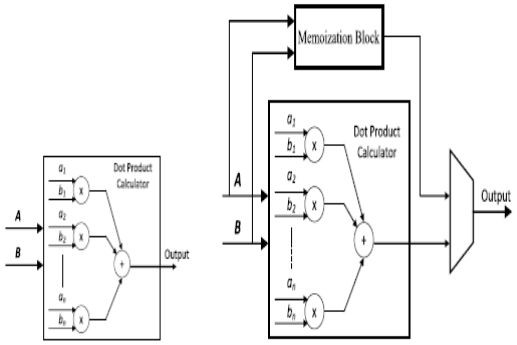

Fig.1:memorization block diagram

The memoization method consist of with memoization and without memorization.For with memorization consist dot product calculator.The two inputs A and B given in to themultipierdevice,and its provide the approximate result.

III HYBRID LUT/MULTIPLEXER FPGA LOGIC ARCHITECTURES

Multiple hybrid configurable logic block architectures, both non-fracturable and fracturable with varying MUX:LUT logic element ratios are evaluated across two benchmark suites (VTR and CHStone) using a custom tool flow consisting of LegUp HLS, Odin-II front-end synthesis, AB logic synthesis and technology mapping, and VPR for packing, placement, routing, and architecture exploration. Technology mapping optimizations that target the proposed architectures are also implemented within ABC.

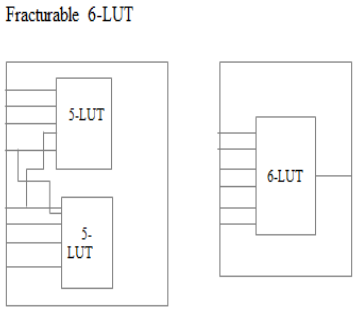

Fig.2:6-LUT that can be fractured into two 5-LUTs with two shared inputs.

For the fracturable architecture consist of eight-input LE, closely matched with the adaptive logic module in recent Altera Stratix FPGA families. A 6-LUT that can be fractured into two 5-LUTs using eight inputs is shown in Fig.2. Two five-input functions can be mapped into this LE if two inputs are shared between the two functions. If no inputs are shared, two four-input functions can be mapped to each 5-LUT. For the MUX4 variant, Dual MUX4, we use two MUX4s within a single eight-input LE. In the configuration, the two MUX4s are wired to have dedicated select inputs and shared data inputs.

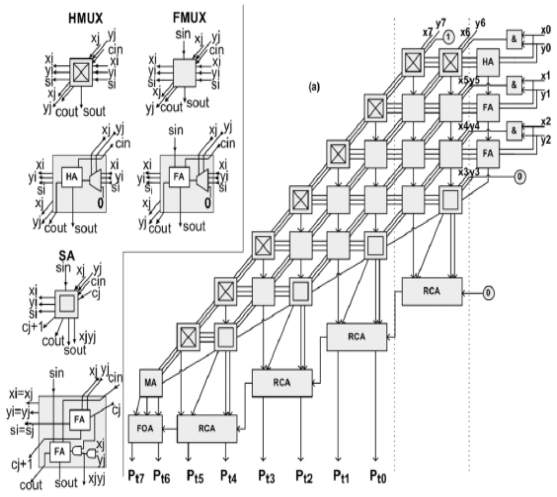

Fig.3 MUX based multiplier

The proposed new error compensation circuit by using the dual group minor input correction vector to lower input correction vector compensation error. By modifying half adder addition like both sum and carry is one when input A and B equal to one. By

constructing the error compensation circuit mainly from the "outer" partial products, the hardware complexity only increases slightly as the multiplier input bits increases.

IV EXPERIMENTAL RESULTS

The proposed circuit are simulated and synthesized by using modelsim and xilinx12.1 .which occurs low area than the existing. The experimental results are given in Table 1 and the simulation results of layout and the waveforms are shown in the fig.4 and fig.5.

Table.1 comparison table

s.no	Parameter	Existing	Proposed
1	Slice	89	53
2	LUT	158	98
3	IOB	37	29

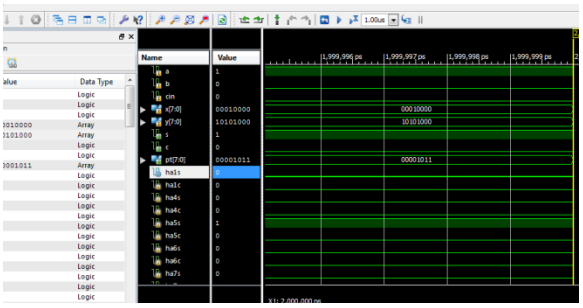

Fig.4 simulation results

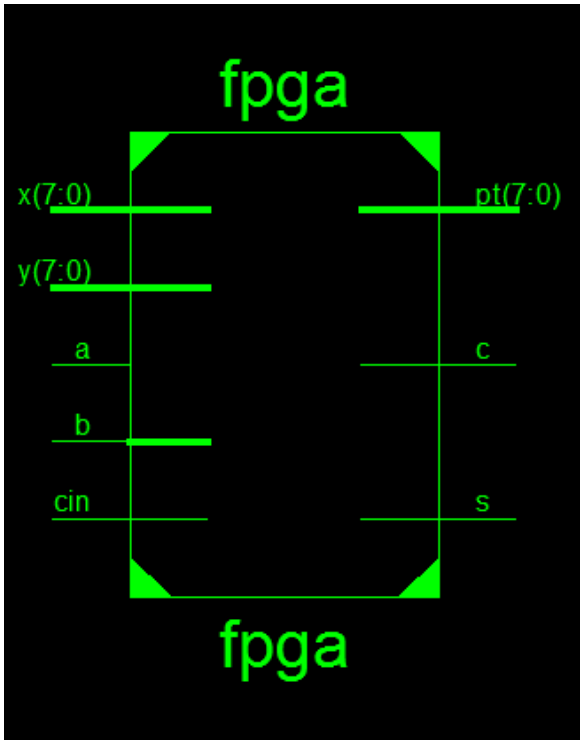

Fig.5 RTL schematic

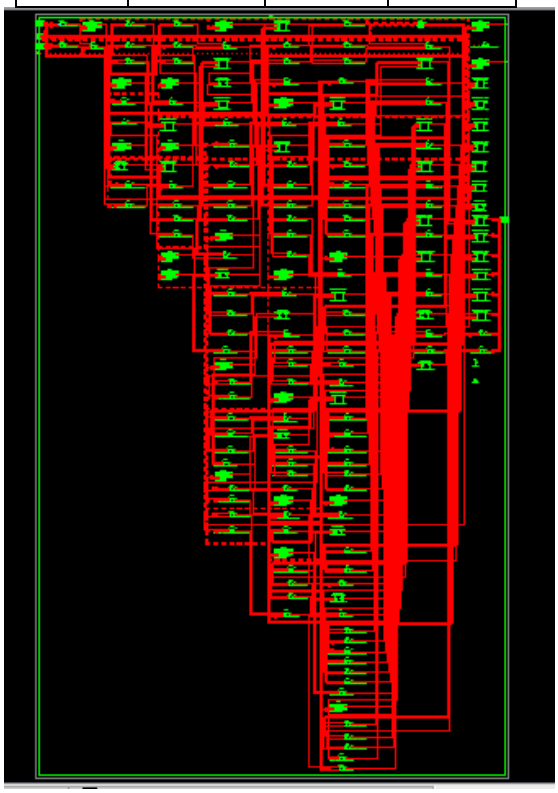

Fig 6: gatelevelnetlist

V PERFORMANCE ANALYSIS

The Figure given below is shown that there is a considerable reduction in time and area based on the implementation results which have been done by using Spartan-3 processor. The proposed algorithm

significantly reduces area consumption when compared to the existing system.

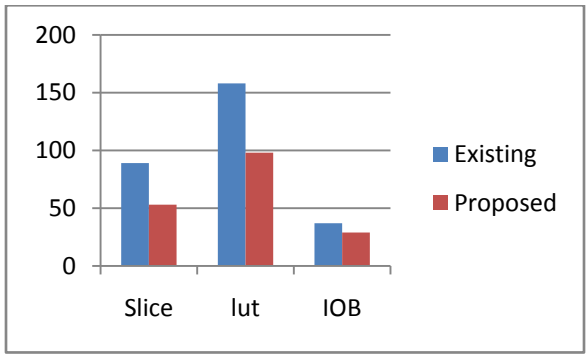

Fig.7 performance analysis of existing and proposed method

VI CONCLUSION

A new hybrid CLB architecture containing MUX4 hard MUX elements are proposed and it shows the techniques for efficiently mapping to the architectures. New multiplexer based truncation scheme with area reduction and less routing overhead than existing methods.For both non fracturable and fracturable architectures, minimal impact on timing performance for the architectures with best area efficiency**.**The addition of MUX4s to FPGA architectures minimally impact MUX and show potential for improving logic-density in nonfracturable architectures and modest potential for improving logicdensity in fracturable architectures.

REFERENCES

[1] Kuon and J. Rose, "Measuring the gap between FPGAs and ASICs," in Proceedings of the 2006 ACM/SIGDA 14th international symposiumon Field programmable gate arrays. ACM New York, NY, USA, 2006, pp. 21–30.

[2] Wu K. and Tsai Y., "Structured ASIC, Evolution or Revolution," April 2004, pp. 103–106**.**

[3] Martínez C.A., Adrián J.C.S., and Cortés M.V.(2012), 'Dynamic tolerance region computing for multimedia' ,IEEE Trans. Comput., Vol. 61, No. 5, pp. 650–665.

[4] Coussy P., Gajski D.D., Meredith M., and Takach A. (2011), 'An introduction to high-level synthesis', IEEE Design Test Comput., Vol. 26, No. 4, pp. 8–17.

[5] H. Parvez, Z. Marrakchi, and H. Mehrez, "ASIF: Application Specific Inflexible FPGA," in International Conference on Field-ProgrammableTechnology, 2009. FPT 2009, 2009, pp. 112–119.

[6] Anderson J., and Wang Q.(2009) "Improving logic density through synthesis- inspired architecture," in Proc. IEEE FPL ,pp. 105–111.

[7] Kuon I., and Rose J. (2007), 'Measuring the gap between FPGAs and ASICs', IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., Vol. 26, No. 2, pp. 203–215.

[8] Hu Y., Das S., Trimberger S., and He L. (2007), 'Design, synthesis and evaluation of heterogeneous FPGA with mixed LUTs and macro-gates', in Proc. IEEE ICCAD, pp. 188–193.

[9] Alvarez C., Corbal J., and Valero M. (2005), 'Fuzzy memoization for floating- point multimedia applications', IEEE Trans. Comput., Vol. 54, No. 7, pp. 922–927

[10] Rose J., Francis R., Lewis D., and Chow P. (2004), 'Architecture of field- programmable gate arrays: The effect of logic block functionality on area efficiency', IEEE J. Solid-State Circuits, Vol. 25, No. 5, pp. 1217–1225.

[11] Shim B., Sridhara S., and Shanbhag N.(2004) "Reliable low-power digital signal processing via reduced precision redundancy," IEEE Trans. VLSI Syst.,Vol. 12, No. 5, pp. 497 – 510.

[12] Hegde R., and Shanbhag N.R.(2001) "Soft digital signal processing," IEEE Trans. VLSI Syst., Vol. 9, No. 6, pp. 813–823.