AENSI Journals

Australian Journal of Basic and Applied Sciences

ISSN:1991-8178

Journal home page: www.ajbasweb.com

Minimizing Delay and Power Using Cramming Adjournment Algorithm

[1]Vijayakumar Sundararaju, [2]Dr. J. Sundararajan Jayapal, [3]Dr. P. Kumar, [4]Nithya Kandasamy

[1]Assistant Professor, Dept. of Electronics and communication Engineering, Paavai Engineering College, Namakkal, 637018, TN, India. +919894445344,svijiece@gmail.com.
[2]Principal, Dept. of Electronics and communication Engineering, Pavai College of Technology, Namakkal, TN, India.
[3]Professor, Dept. of Electronics and communication Engineering, K.S. Rangasamy College of Technology, Tiruchengode, TN, India.
[4]Assistant Professor, Dept. of Electronics and communication Engineering, Paavai College of Engineering, Namakkal, TN, India.

ARTICLE INFO

Article history:
Received 26 December 2013
Received in revised form 25 February 2014
Accepted 26 February 2014
Available online 15 March 2014

Keywords:
Global Router, Routing Slice, Critical Path, Congestion, Spanning Tree.

ABSTRACT

Background: In recent years, the design for low power has grown to be one of the greatest challenges in FPGA. As a consequence, many techniques have been introduced to minimize the power. The FPGAs are extensively utilized in software/hardware embedded purposes due to their benefits. But a few drawbacks are there in using FPGAs during hardware design approaches. a few of the disadvantages are, bonding the hardware design tools into software tools, due to routing along with placement length of FPGA is boosted, FPGAs design has a lesser amount of portability. **Objective:** In this paper we described a new routing fabric for reducing power along with Cramming Adjournment algorithm for reducing the delay. The power consumed in a FPGA interior consists of mutually static and dynamic components. Static power adds only 10% of the entire power in a FPGA. in contrast, dynamic power put in excess of 90% of the whole power consumed moreover it is the foremost source for their power disorganization. This paper enlightens how we can decrease the power of FPGA by dropping the routing delay. The routing delay knows how to be reduced by establishing a new routing fabric method designed for field programmable gate array. **Results:** In this effort, we focus on accomplishing 2.30 times minor consumption of dynamic power plus average net delays reduction is 1.8 times. **Conclusion**: Enrichments in power and delay reduction can be realized by reducing the routed length via small interconnect segments as well as reducing interconnect segment loading due to programming overhead relative to the baseline FPGA devoid of compromising routability. When the proposed technique is compared with the conventional techniques, it reduces the power and delay considerably.

To Cite This Article: Vijayakumar Sundararaju, Dr. J. Sundararajan Jayapal, Dr. P. Kumar, Nithya Kandasamy., Minimizing Delay and Power Using Cramming Adjournment Algorithm. ***Aust. J. Basic & Appl. Sci.,*** *8(2): 152-161, 2014*

INTRODUCTION

Field-programmable gate arrays (FPGAs) are becoming an increasingly important implementation medium for digital logic. FPGA are much less power efficient when weight against with cell-based ASIC. This power inefficiency restricted the application of FPGA in low power area. But FPGA has advantage that it is well-matched to the changing need and short design cycles. Hence reducing power consumption is essential in FPGA. Dynamic power consumption rooted by signal modifications. Higher operating frequencies show the way to the boosted transistor action which means further dynamic power Dissipation. The huge source of dynamic power spending in a FGPA is commencing of charging and discharging of capacitor.

Signal transitions which openly determine dynamic power are divided into two types they are functional transition and spurious transitions or glitches. Functional transitions occur when there is a transition required to do the logic function between two consecutive clock cycles. Glitches is short duration electrical pulse, usually it produces fault result particularly in a digital circuit. In FPGA, glitch power plays a most important role in total dynamic power. For this reason reducing glitches is important.

In this article, we give attention to reducing glitch power by examine the path to inputs of look up table. So that signals of the same Look - Up table arrive at the same time and no glitches generated. Here we finding a substituted routes for early arriving signals so that the delays of the new routes causes the signals to arrive at the balanced times.

Corresponding Author: Vijayakumar Sundararaju, Assistant Professor Dept. of Electronics and communication Engineering, Paavai Engineering College, Namakkal, 637018, TN, India.
Ph: +919894445344, E-mail: svijiece@gmail.com.

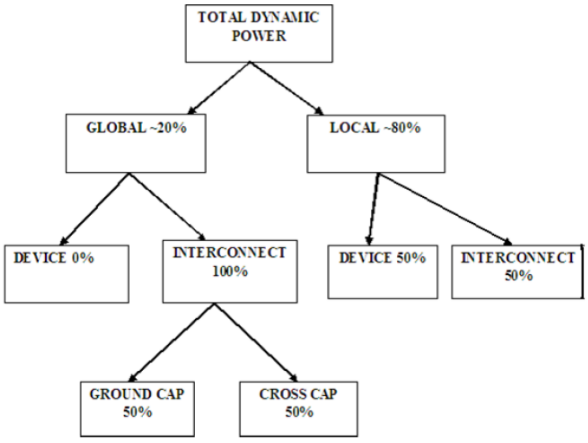


Fig. 1: Power Consumption types.

A. Associated work:

Numerous techniques proposed to shrink power and Delay which consist of

In (Lin and Gamal,2008) their effort they have shown that TORCH based on simulated annealing procedure to find an optimized segmentation based on an average delay-power product. In every iteration, segmentation is incrementally changed, for new segmentation the benchmark designs routed into the FPGA using Versatile place route, the performance metric restructured, and the new segmentation is either accepted or rejected. Because of infrequent placements, Run time is much condensed. TORCH outputs have an optimized mix of track segment lengths and an ordering of the segmented tracks in the channel.

In (Vijayakumar *et al*., 2013) their approach mobbing deferral evasion algorithm was implemented in 2-D FPGA. The glitch problem also reduced in this approach. Net results of the approach are power reduction 1.80 times and average net delay reduction is 1.50 times in various wire lengths.

The (Chen *et al*., 2003) algorithm revealed that every net in the circuit are repeatedly rips-up and re-routed, and gradually resolves routing congestion by gradually increasing the cost of overused routing resources. The slack of each connection used to decide the congestion avoidance delay minimization trade-off to use for that connection.

In (V. Betz and J. Rose,1997)Versatile place route algorithm exposed that the inputs to versatile place route consist of a technology routed net list and a text file describing the FPGA architecture. Versatile place route can place a pre-existing placement, or the circuit. Versatile place route can then do either a global route or a detailed route of the placement. Versatile place route output consists of the placement and routing, as well as statistics details useful in assessing utility of a FPGA architecture, which includes routed wire length, track count, and largest net length.

In (Lin *et al*., 2004) this work publicized that there are two types of routing tracks based on consumption of voltage: High tracks and Low tracks. Both tracks are differs from using switches. High tracks get high supply voltage and faster than the low tracks. The paths having zero values can use the faster High tracks and the other paths can use the slower .Low paths to save power.

At (Lin and Gamal,2009) this point they revealed that preliminarily, nets routed individual at a moment by means of the shortest path via considering interrelate segment or else logic block pin overuse. Every iteration of the router consists of sequential net rip-up as well as reroute according to the lowest cost path existing. The cost of using a routing source is a present overuse and any overuse that occurred in preceding routing iterations. By gradually increasing the cost of an oversubscribed routing resource, the algorithm forces nets with alternative routes to avoid using that resource, leaving it to the net that most needs it.

Above discussed algorithm analysed only the channel segments and most of which not discussed about the power consumption of look up table. We propose a similar technique that targets FPGA power and Delay. In addition to that we concentrated on Glitch reduction which is major part of power consumption.

Our Contribution,

- The new routing fabric for reducing overall power and delay with the help of short segment.
- Dynamic power condensed by reducing glitches through path balancing.
- The new algorithm to locate a shortest path between source nodes and sink node with desired delay.

The paper is ordered as follows segment 2 provides new routing architecture .In segment 3 we described the method of reducing the dynamic power. Segment 4 we described about routing algorithm for finding shortest path and we concluded the paper in segment 5.

Routing Fabrics for 2D FPGA:

In new routing fabrics, logic blocks were combined and arranged in an array format in the midst of horizontal and vertical routing channel overlay. Routed net lengths reduced by using only short interconnect segment in the routing channels. The routing block provides connectivity for logic block inputs and outputs as well as that it integrate the functions of connection and switch boxes. The routing points used to

- Form local connections among neighbouring logic blocks without going to channels.
- Connect routing block inputs and outputs to channel segments and Chain channel segments together to form longer segments without entering routing blocks.

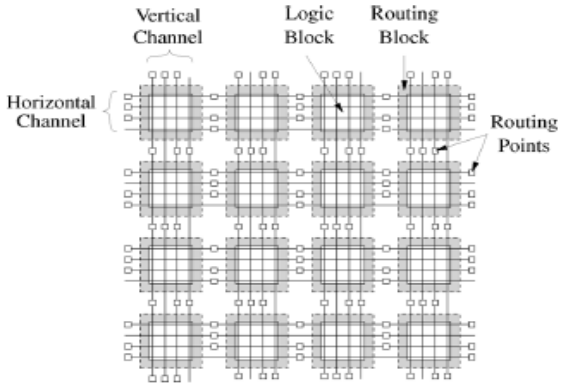


Fig. 2: Modified 2D FPGA.

A. Link among Logical Block and Routing Block:

Routing block perform occupation of connection and switch boxes. Logical block contain look up tables, flip-flops and programming overhead. Every routing block can connect to n_i LB input such that each LB and routing block connected to bypass transistor switches. By choosing a value of n_i such that each LB input connect to the same number of routing block inputs. The loading on a routing block segment is lower than on a routing block input segment in the baseline fabric.

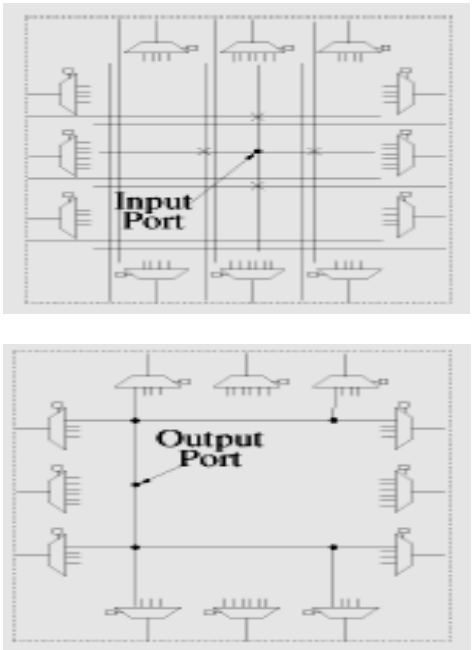


Fig. 3: Logical block input & output connected to routing block.

In addition to the connection through switch points, routing block architecture allows for extended switching width. In which a signal in a routing block looped back twice into it and exit to a perpendicular direction if it cannot do so directly. This extended switching much improves the efficiency of routing.

B. Connection between Routing Block and Routing Channel Overlay:

Every routing channel comprises of single and double segmented tracks. This segment has two unidirectional wires. The input and output connected by channel segment using the routing points. Segments joined together to form a longer segments, which called bypass interconnect. The segments can also be connected via routing points to routing blocks to connect to LB inputs and outputs, make bends, or fan-out. Two types of net connections: Local and bypass connection. In local connection output of LB are already routed inputs of its neigh boring LB without using routing channel segment .In order to route a longer net without entering intermediate routing block.

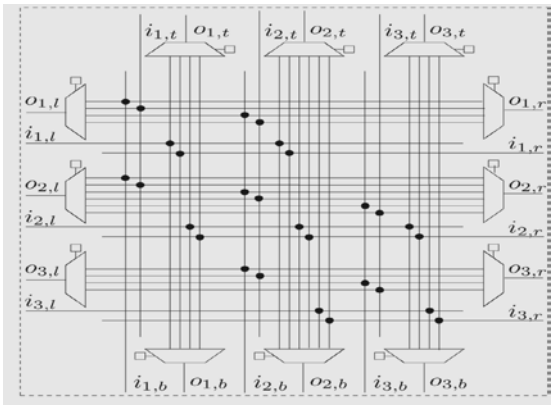


Fig. 4: Routing Block.

The routing fabric allocated a two types of net connections: local A local connection make use of route of one output of an LB to an input of its nearby LB directly without using routing channel segments. A bypass connection employs to route a longer net without entering midway routing blocks. Fig.5a Shows how a bypass interconnect is implemented. Remember that by turning off the two pass transistor switches S1 and S2, a local connection can be concurrently made among the two routing blocks. This routing point resource contributes in getting better routability. Turning off these pass transistors also diminishes the loading on the bypass interconnect.

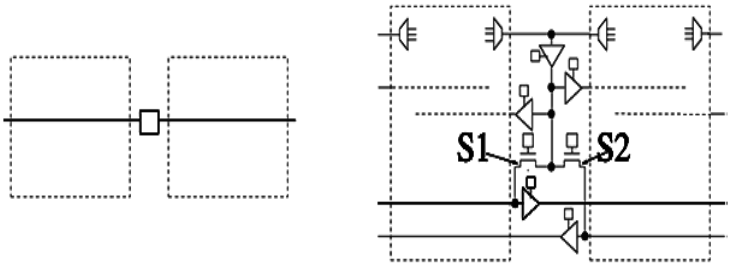


Fig. 5a: Bypass inter connect.

Fig. 5b and Fig. 5c., respectively, demonstrates how an LB input and output can be coupled to a segment. Note that only buffers that are required to establish the connection are turned on. This again helps in reducing the loading on the connection, thus reducing its delay and power consumption.

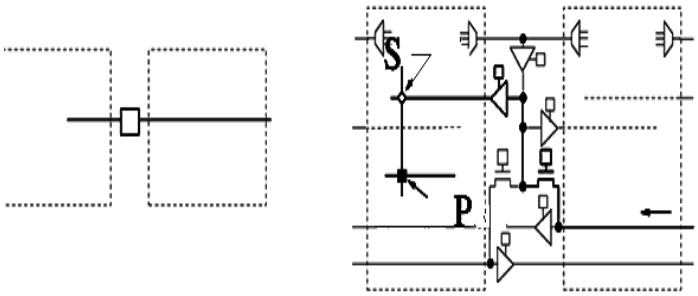


Fig. 5b: Connection from a segment to an LB input.

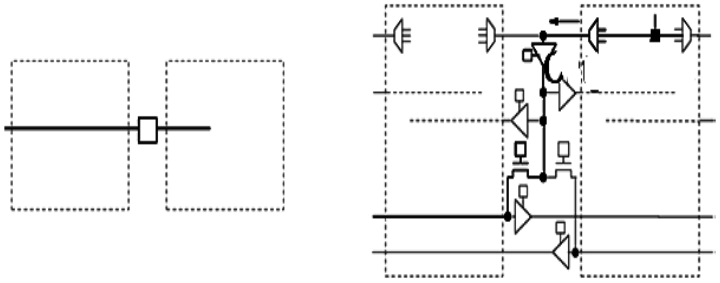
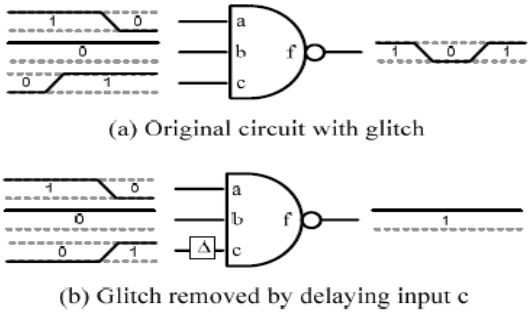


Fig. 5c: Connection from an LB output to a segment.

C. Dynamic Power Consumption:

In FPGA, glitches generated at the output of a LUT when signals transition takes place at different times. The pulse width of these glitches depends on how uneven the input signal arrival times are. Due to the limited connectivity of FPGA routing resources FPGA glitches are wider than ASIC glitches. We avoided the glitches by adding programmable delay elements within the logic blocks of the FPGA such that we delay the early arriving signals to align the edges on each LUT input signals, thereby reducing some glitches on the output of each LUT. The method demonstrated in Fig. 6 by delaying the input signal of c, the output glitches eliminated since only the early arriving signals delayed, the overall critical path of the circuit is not increased.



(a) Original circuit with glitch

(b) Glitch removed by delaying input c

Fig. 6: Removing Glitches by delaying early arriving signal.

1) Programmable delay element:

The delay element circuit consists of two inverters. The first inverter composed of pull-up and pull-down resistor for controlling the delay of the circuit. The second inverter has large channel lengths to decrease short-circuit power. Both pull-up and pull-down resistor have n stages with a resistor and a bypass transistor which controlled by an SRAM bit. Control bits used to double the value of resistor in later stages.

The control bit planned to produce any delay

$$\Delta \in \{k, \tau + k, 2\tau + k, 3\tau + k, \ldots, (2n\text{-}1)\tau + k\} \quad (1)$$

Where τ is the delay produced by a resistor R to charge or discharge the capacitor C and k is the delay produced by the delay produced by the bypass resistances and inverters.

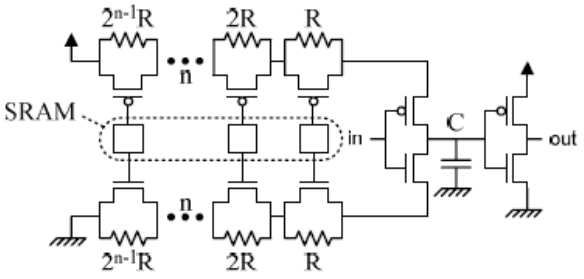


Fig. 7: Programmable delay elements.

D. Cramming/Adjournment algorithm for reducing delay:

Initially without considering interconnect segment or logic block pins, nets routed one at a time using the shortest path. Several iterations carried out for finding shortest path. According to the lowest cost path nets are

ripping up and rerouted at each iterations. The cost of mapping resource is function of its current overuse and any overuse that occurred in preceding mapping iteration. If the resources overused then algorithm forces nets with alternative routes to avoid using that resource.

Fig.8 shown below is the placing route graph for routing block. Here each routing block input signal and output represented by node. When routing algorithm applied to the routing graph shown below obtained. Fig. 9 shows the shortest path between the source n_1 and n_2. Solid line represents the direct connection and dashed line represents the extended connection.

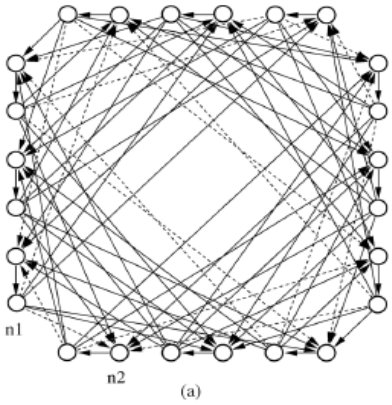


Fig. 8: Routing graph for routing block.

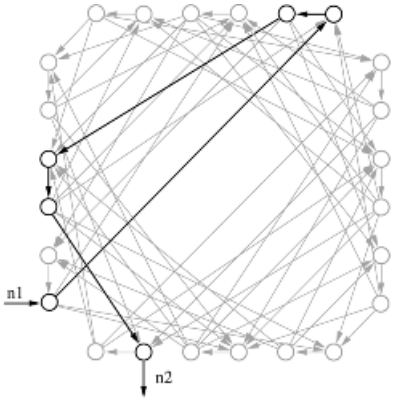


Fig. 9: Routing graph for routing block.

E. Cramming/Adjournment algorithm:

- K_{xy} is critical connection from the source of net i to one of its sinks j;
- D_I is the intrinsic delay of routing node n;
- C_C is the present congestion cost of node n.

Algorithm:

1: $K_{xy} \leftarrow 1$ for each signal net x and each sink y
2: **while** shared routing nodes exist **do**
3: **for all** nets x **do**
4: rip up routing tree RT_x
5: initialize the queue AB
6:**for all** sinks d_{xy}**do**
7: enqueue each node n in RT_x at costs K_{xy} D_I toAB
8:**while**$d_{xyj}$is not found **do**
9:dequeue node m with the lowest cost from AB
10:**for all**fanout node n of m **do**
11: **if** node n is unseen **then**
12: mark node n as seen
13: enqueue n to AB with the cost of $K_{xy} D_I + (1- K_{xy}) D_I C_C$
14: **end if**
15: **end for**
16: **for all** node n in the routed path d_{xy} to s_j**do**

17: update the cost of node n
18 : add n to RT_x
19: **end for**
20:**end while**
21:**end for**
22:mark all nodes in AB as unseen
23:update K_{xy} for net x
24:**end for**
25:**end while**

Fig.10 Contains the details of the delay-based routing algorithm. The K_{xy}'s are initialized to 1 (step1). Thus for the duration of the first iteration the global router discovers the minimum-delay route for all signal. In a style equivalent to the congestion-based router, a precedence queue execution of a breadth first search be carried out to locate sinks. For nodes previously in RTx the cost is now the delay time (step 7); for every single one it is the amount of the delay along with congestion. The net effect is that nodes that are previously in the routing tree will not have a congestion component. Having previously be owed to signal x they are "free" from a congestion viewpoint. Sinks are routed one by one and decreasing K_{xy} .Spontaneously, sinks with the highest slack ratios(and as a result the majority time-critical) must have the most significance in shaping the tree structure whereas sinks with a low slack ratio will comprise extra flexibility.

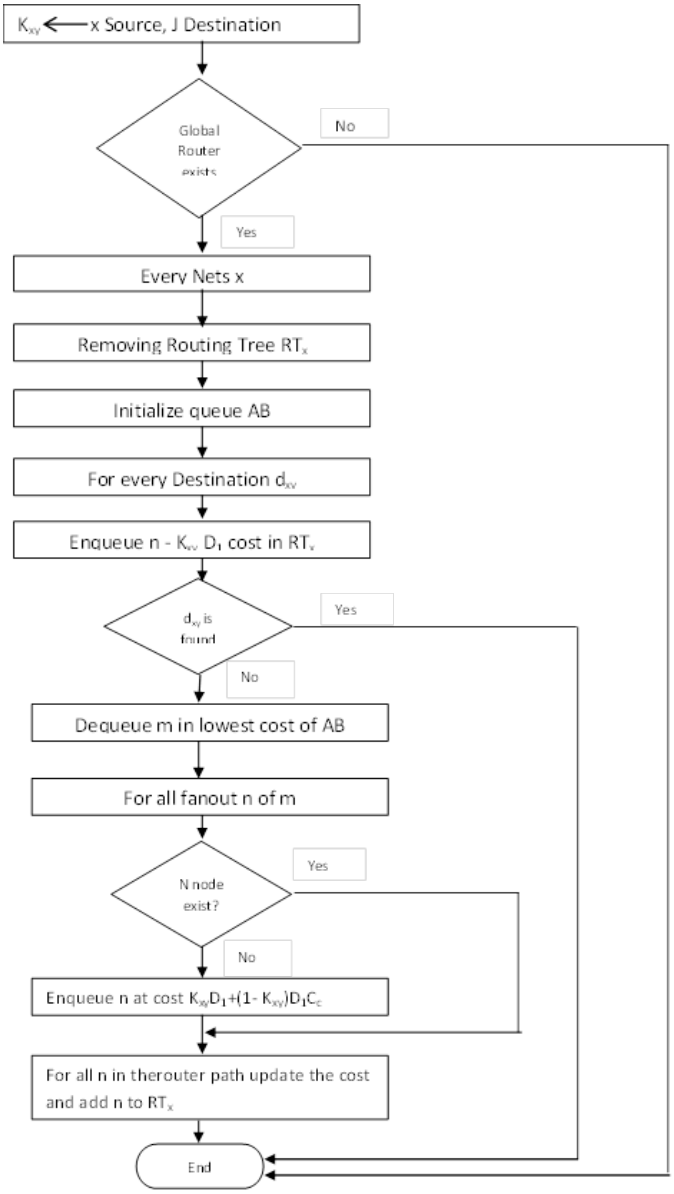


Fig. 10: Flowchart of Cramming/ Adjournment.

This mechanism work fine in exercise, which is explained with model where sinks having lower K_{xy} have to be routed initially since their nearness towards good paths to sinks with higher K_{xy}. The frequent difficulty of discovering a minimum spanning tree focus to delay limitations is a multifarious trouble. Finding the accurate minimum spanning tree, given a set of delay restrictions also congestion values, is not essential to this algorithm, now locating the optimum Steiner points is not important to the congestion-based algorithm. If we "support" the routing tree to assemble the delay limitations during the use of the K_{xy}, the global router will try to alter the congestion values to meet on a routing tree that has denial shared nodes. At conclusion of every iteration (step 21) the path delays and K_{xy} are recalculated. The global router finish when no collective resources exist. Note that by means of recalculating the K_{xy} in all iteration, we remain a fixed region on the critical path. In excess of iterations, the critical path increases only to the scope essential to determine congestion.

Results:

Fig 11. Show that the power reduction in times based on four technology nodes, and Fig 12. Show that the delay ratio increases with increase in technology nodes. This is because of increase in the parasitic wires.

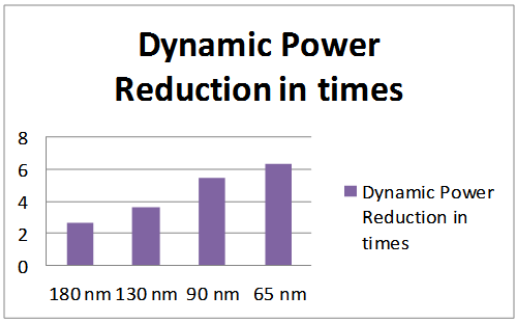


Fig. 11: Technology generation versus power reduction in times.

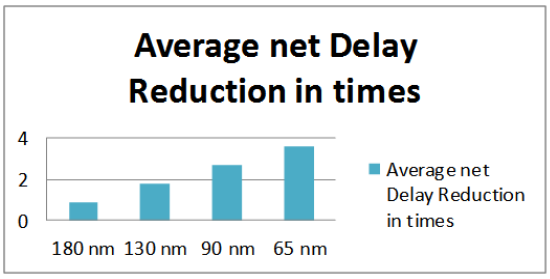


Fig. 12: Technology generation versus Delay reduction in times.

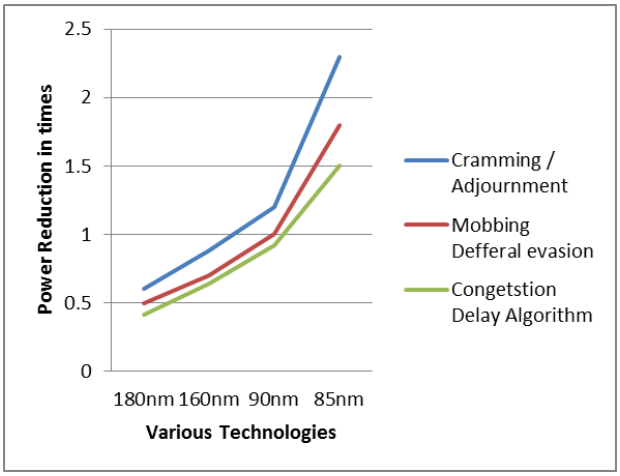


Fig. 13: The existing and proposed power comparison in routing fabrics.

From this analysis the delay has been increased by increasing the delay weight.

Discussion:

The power usage is decreased, but it is in very lighter way. The Cramming Adjournment algorithm method is reducing the power and delay, when it is compared with the existing method which is shown in Fig.13 and Fig 14.The power reduction is 2.30 times when it compared with existing methods shown in Fig 13.

Fig.14 shows that the relation between delay and routing length. The Cramming Adjournment algorithm method reduces the delay reduction of 1.80 times. From the above analysis the delay reduction in routing fabric is improve the overall performance of the FPGA systems.

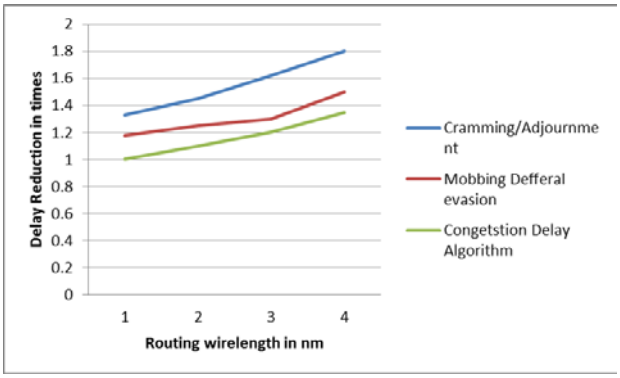


Fig. 14: The existing and proposed delay comparison in routing fabrics.

Conclusion:

The power inefficiency of FPGA is a major problem. By reducing routed net length and programming overhead the power consumption reduced. Routed net length reduced by inter connect the segments by shortest routing channels. By decreasing the switch box and connection box flexibilities programming overhead reduced. We developed a new routing fabrics and algorithm FPGA can do 2.30 times reduction in the overall dynamic power consumption and 1.80 time reduction in average net delays.

REFERENCES

Betz, V. and J. Rose, 1997. "VPR: A new packing, placement and routing tool for FPGA research," in *Proceeding 1997 7th International Workshop Field-Programmable Logic Application*, pp: 213-222.

George, V., 2000. "Low energy field-programmable gate array," Ph.D. dissertation, University of California, Berkeley.

Joshi, A.R. and K.C. Saraswat, 2003. "Nickel induced crystallization of α-Si gate electrode at 500C and MOS capacitor reliability," *IEEE Transactions on Electron Devices*, 50(4): 1058-1062.

Li, F., D. Chen, L. He and J. Cong, 2003. "Architecture evaluation for power-efficient FPGAs," in *Proceeding 2003 ACM/SIGDA 10th International Symposium Field-Programmable Gate Arrays*, pp: 175-184.

Li, F., Y. Lin, L. He and J. Cong, 2004. "Low-power FPGA using pre-defined dual-Vdd/dual-Vt fabrics," in *Proceeding 2004 ACM/SIGDA 12th International Symposium Field-Programmable Gate Arrays*, pp: 42-50.

Li, F., Y. Lin, L. He, D. Chen and J. Cong, 2006. "Power modelling and characteristics of field programmable gate arrays," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 24(11): 1712-1724.

Lin, M. and A. El-Gamal, 2007. "A routing fabric for monolithically stacked3-D-FPGAs," in *Proceeding 2007 ACM/SIGDA 15th International Symposium Field-Programmable Gate Arrays*, pp: 3-12.

Lin, M., A. El Gamal, Y.C. Lu and S. Wong, 2006. "Performance benefits of monolithically stacked 3-D-FPGA," in *Proceeding2006*, "Performance benefits of monolithically stacked 3-D-FPGA," in *Proceeding2006 ACM/SIGDA 14th International Symposium Field-Programmable Gate Arrays*, pp: 113-122.

Mingjie Lin and Abbas El Gamal, 2008. "TORCH: A tool for segmented routing channel design in FPGAs",*Proceeding 2008 ACM/SIGDA 16thInternational Symposium Field-Programmable Gate Arrays*, pp: 131-138.

Mingjie Lin and Abbas El Gamal, 2009. "A Low-Power Field-Programmable Gate Array Routing Fabric" in *IEEE Transactions on Very Large Scale Integration (VLSI)Systems*, 17: 1481-1494.

Shang, L., A.S. Kaviani and K. Bathala, 2002. "Dynamic power consumption in Virtex-II FPGA family," in *Proceeding 2002 ACM/SIGDA 10th International Symposium Field-Programmable Gate Arrays*, pp: 157-164.

Tuan, T., S. Kao, A. Rahman, S. Das, and S. Trimberger, 2006. "A 90 nm low power FPGA for battery-powered applications," in *Proceeding 2006 ACM/SIGDA 14th International Symposium Field-Programmable Gate Arrays*, pp: 3-11.

S.Vijayakumar, J. Sundararaju, P. Kumar and K. Nithya, 2013. “Delay and power reduction in new routing fabrics,” *American journal of applied sciences*, 10(12): 1537-1545.