ISSN 2348-375X

Unique Journal of Engineering and Advanced Sciences

Available online: www.ujconline.net

Research Article

AN EFFICIENT H-TREE BASED CLOCK TREE DESIGN USING AGGLOMERATIVE CLUSTERING ALGORITHM

Radhika V[1*], Murugasami R[2]

[1]PG Scholar, Department of ECE, Nandha Engineering College, Erode-52
[2]AP, Department of ECE, Nandha Engineering College, Erode-52

Received: 17-12-2013; Revised: 16-01-2014; Accepted: 15-02-2014

*Corresponding Author**: Radhika.V**
PG Scholar, Department of ECE, Nandha Engineering College, Erode-52 Email:radhikavelusamy@gmail.com

ABSTRACT

Power consumption has become an important issue in high-performance circuits. Several techniques are used to reduce total power of a chip, such as multiple supply voltages, clock gating, and clock-tree minimization. Minimizing the size of a clock tree is known as an effective approach to reduce the power dissipation in modern circuit designs. Clock tree has different types of sinks like flip flops and pulsed latches. In this system latches are connected to the buffer at a limited position, each latch can communicate to nearby buffer this form the forms the tree based network. The clock tree with mixture of sinks is constructed to reduce power dissipation, but the load level of clock tree is increased. In the proposed system, load level of clock tree, the power and clock skew are reduced by using agglomerative clustering algorithm in the H-tree based architecture. Multi-corner multi-mode (MCMM) Clock tree synthesis (CTS) is used in H-Tree based topology to reduce the load level in clock tree. Experimentally load level of clock tree, the power and clock skew of the system is improved by using the agglomerative clustering algorithm in H tree based architecture. Xilinx tool will be used to carry out the proposed system. The proposed system will be implemented using Hardware Description Language (VHDL).

Keywords: Clock tree design, dynamic power reduction, Flip flops, Agglomerative clustering algorithm.

INTRODUCTION

Power consumption has become a crucial issue in high-performance circuits because the number of transistors has increased substantially. Several techniques are proposed to reduce total power of a chip, such as multiple supply voltages, clock gating, and Clock-tree minimization. Because of heavy pipeline designs and high-frequency signal switching, a clock tree is known to be a major contributor to power dissipation. The clock tree accounts for a significant portion of total power consumption and consumes 20%–40% of total power in synchronous Circuits. Therefore, the chip power can be greatly reduced by decreasing the clock-tree power. The power dissipation of a clock tree can be reduced by decreasing total (clock) wire capacitance. Hou et al. reduced the wire length in the placement stage by placing registers closer and grouping neighboring registers into groups[1-5].

A clustering algorithm that uses a minimum spanning tree to estimate the interconnect Capacitance and reduce the total wire capacitance in the routing Stage. However, existing methods of clock-tree minimization are primarily based on flip-flops and focus on wire length Minimization alone, which may limit achievable power Savings. In current circuit designs, the most common storage element is a D-type flip-flop that consists of two latches (master and Slave) triggered by a clock signal. This type of design makes it easier to apply static timing analysis (STA) for timing Verification. As transistor counts of a flip-flop are two times than that of a single latch, latches are superior to flip-flops in terms of area, transition time, and power dissipation. However, it is difficult to perform STA on latch-based circuits because of data transparency. A pulsed-latch-based design style was adopted for dynamic power reduction. Pulsed latches are latches triggered by a brief clock signal Generated from a pulse generator. When the pulse clock Waveform triggers a latch, the latch is synchronized with the Clock and its timing behavior is similar to an edge-triggered Flip-flop[6-9].

Hence, STA can be applied to the pulsed-latch clock Tree. Therefore, pulsed-latch designs have both advantages of Latches and flip-flops: they offer easier timing verification and less power consumption. As in, a 20% reduction in total dynamic power consumption can be achieved in practice. Although pulsed latches can effectively reduce power

Consumption, most current design flows are built for flip-flop designs. To adopt pulsed latches for a current design flow, Designers might change the circuit description in high-level Synthesis. However, this modification incurs excessive costs and causes high complexity in physical-synthesis stages. Therefore, an efficient pulsed-latch Migration approach in physical design to minimize the cost of utilizing pulsed latches under the current design flow[10-13].

In pulsed-latch designs, a pulse generator is indispensable to generate a clock pulse, but consumes more power than a pulsed latch and a buffer. Although pulsed latches can Reduce power dissipation, the total power of the clock tree May increase because of additional pulse generators. Thus, there is a tradeoff between the pulse-generator insertion and Pulsed-latch substitution. As the clock pulse is sensitive to Output load, it is essential to control the load of a pulse Generator for potential pulse degradation. Additionally, if designers do not limit the number of pulsed Latches driven by a pulse generator, the number of fan outs In the pulse generator may be too large, which could lead To routing congestion. Therefore, two major factors must be considered to control the output load: 1) the pulse-generator Driving load cannot exceed the maximum tolerable load Defined in the library and 2) the number of pulsed latches Driven by a pulse generator should be smaller than the maximum fan out number[14].

As pulse generators consume large amounts of power in Pulsed-latch circuits, it is critical to reduce the pulse-generator Power. Multi-type pulse generator insertion to reduce the unnecessary power dissipation. Considering Clock gating of pulsed-latch circuits, the largest-size pulse Generators are inserted to drive pulsed latches. Then, to further reduce power consumption, this method replaces the Largest-size pulse generators with smaller-size ones if there are no constraint violations. As a result, the power consumption can be further reduced. Two recent related papers have attempted to address the utilization of pulsed latches in clock-tree construction. Lin et al. proposed a clustering algorithm to cluster Pulsed latches and minimize the skew. However, this method re clusters pulsed latches from each flip-flop groups in an initial Clock tree, and the clustering result is limited by the local Search. In addition, the number of pulse generators increases unnecessarily, causing greater power dissipation[15].

A clustering algorithm is to minimize the number of pulse generators, and uses a minimum spanning tree to determine the sub tree topology. However, this method does not consider the skew problem when constructing a clock tree. Moreover, two related papers do not consider multi type pulse Generators and the mixture structure of pulsed latches and Flop-flops simultaneously, which are not applicable in practical Circuit designs. In this paper, to our best knowledge, this is the first Research to introduce the clock tree migration problem with a mixed structure of pulsed latches and flip-flops for Power optimization. We achieve this goal by altering a Clock-tree topology and distribute the driver load for timing[16].

PULSED LATCH SCHEMES

A. Our Contributions

Clock trees pose a growing challenge to advanced node IC design, particularly with regard to the chip power consumption. In this pulse generator (Clock generator) was done by three patterns as, Separate Pulse generation for each block (Small load but large power consumption), Single Pulse generation for overall FFs (Large load but small power consumption), and Separate Clusters of pulse generation for FFs (Small load and small power consumption). In pulsed-latch designs, a pulse generator is indispensable to generate a clock pulse, but consumes more power than a pulsed latch and a buffer. Although pulsed latches can reduce power dissipation, the total power of the clock tree may increase because of additional pulse generators[17].

Thus, there is a tradeoff between the pulse-generator insertion and pulsed-latch substitution. As the clock pulse is sensitive to output load, it is essential to control the load of a pulse generator for potential pulse degradation. Additionally, if designers do not limit the number of pulsed latches driven by a pulse generator, the number of fan outs in the pulse generator may be too large, which could lead to routing congestion. Therefore, two major factors must be considered to control the output load: 1) the pulse-generator driving load cannot exceed the maximum tolerable load defined in the library and 2) the number of pulsed latches driven by a pulse generator should be smaller than the maximum fan out number. As pulse generators consume large amounts of power in pulsed-latch circuits, it is critical to reduce the pulse-generator power[18].

Paik proposed multi-type pulse generator insertion to reduce the unnecessary power dissipation. Considering clock gating of pulsed-latch circuits, the largest-size pulse generators are inserted to drive pulsed latches. Then, to further reduce power consumption, this method replaces the largest-size pulse generators with smaller-size ones if there are no constraint violations. As a result, the power consumption can be further reduced. Fig. 1 shows three methods of pulse-generator insertion. To prevent the distortion of a clock signal, a pulse generator connects a single pulsed latch like Fig 1(a). In this method, the load of each pulse generator is small, but the power consumption may be significantly high, because of the many inserted pulse generators[19].

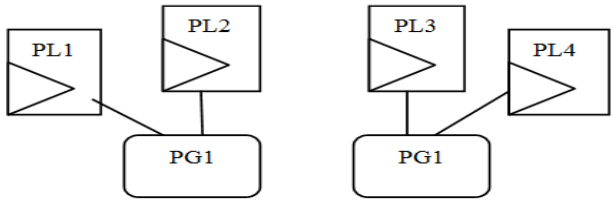

Figure 1: (a) Large Power Consumption & Smaller Load
(b) Large Load &Smaller Power Consumption

On the other hand, if we use a single pulse generator to drive all pulsed latches Fig 1 (b), then the power consumption is reduced by driver capacitance. However, the clock pulse waveform may be degraded because of a heavy output load or maximum fan out constraint. Considering the tolerable load of a pulse generator, the result with the best tradeoff in this example is to use three pulse generators to trigger pulsed latches Fig 1 (c). The two upper pulse generators *PG*1 and *PG*3 in Figure can be replaced by a smaller-size one because it

only drives two pulsed latches, further reducing power consumption.

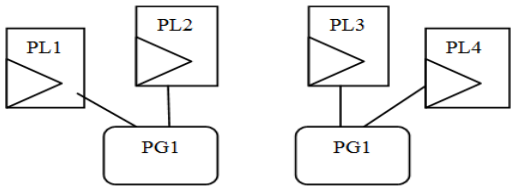

Figure 1: (c) Smaller Load & Power Consumption

B. Clustering Algorithm

Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters). It is a main task of exploratory data mining, and a common technique for statistical data analysis, used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, and bioinformatics[20].

Cluster analysis itself is not one specific algorithm, but the general task to be solved. It can be achieved by various algorithms that differ significantly in their notion of what constitutes a cluster and how to efficiently find them. Popular notions of clusters include groups with small distances among the cluster members, dense areas of the data space, intervals or particular statistical distributions. Clustering can therefore be formulated as a multi-objective optimization problem. The appropriate clustering algorithm and parameter settings (including values such as the distance function to use, a density threshold or the number of expected clusters) depend on the individual data set and intended use of the results. Cluster analysis as such is not an automatic task, but an iterative process of knowledge discovery or interactive multi-objective optimization that involves trial and failure. It will often be necessary to modify data preprocessing and model parameters until the result achieves the desired properties.

Besides the term clustering, there are a number of terms with similar meanings, including automatic classification, numerical taxonomy and typological analysis. The subtle differences are often in the usage of the results: while in data mining, the resulting groups are the matter of interest, in automatic classification primarily their discriminative power is of interest. This often leads to misunderstandings between researchers coming from the fields of data mining and machine learning, since they use the same terms and often the same algorithms, but have different goals. Clustering algorithms can be categorized based on their cluster model, as listed above. The following overview will only list the most prominent examples of clustering algorithms, as there are possibly over 100 published clustering algorithms. Not all provide models for their clusters and can thus not easily be categorized[21].

PROPOSED METHODS

We proposed optimized network topology to addresses both power and timing. In this system, we use multi-corner multi-mode (MCMM) CTS with smart clock gate handling, slew shaping, register clumping, and other advanced techniques for reducing power, skew, area, and buffer count in H-Tree based topology.

1. Initialize Latch

The D flip-flop tracks the input, making transitions with match those of the input D. The D stands for "data"; this flip-flop stores the value that is on the data line. A master–slave D flip-flop is created by connecting two gated D latches in series, and inverting the enable input to one of them. This flip-flops gets single bit as input and generate output as Q and Q`. At a time any one of D flip-flop was activated according to clock pulse generation. Each FF handle single bit input for separated clock generator.

2. Check distance

Here check distance between the neighboring latches. This distance calculation was done by Manhattan distance function. Normally, flip flops and pulsed latches in a logic circuit were handling single bit input with separated clock generation. This may consume more power. This can reduce by merging of flip flops. In this module, they find the distance between the neighboring flip flops which gets the distance between each flip flops. The Manhattan distance function computes the distance that would be traveled to get from one data point to the other if a grid-like path is followed. The Manhattan distance between two items is the sum of the differences of their corresponding components.

The formula for this distance between a point X=(X1, X2, etc.) and a point

Y= (Y1, Y2, etc.) is,

$$\mathbf{D}=\sum_{i=1}^{n}|\boldsymbol{X_i}-\boldsymbol{Y_i}| \qquad (1)$$

Where n is the number of variables, and **X**i and **Y**i are the values of the *i*th variable, at points ***X*** and ***Y*** respectively. The following Figure illustrates the difference between Manhattan distance and Euclidean distance,

Figure 2: (a) Manhatten distance (b) Euclidean Distance

3. Cluster Formation

Agglomerative hierarchical clustering is a bottom-up clustering method where clusters have sub-clusters, which in turn have sub-clusters, etc. The classic example of this is species taxonomy. Gene expression data might also exhibit this hierarchical quality (e.g. neurotransmitter gene families). Agglomerative hierarchical clustering starts with every single object (gene or sample) in a single cluster. Then, in the each successive iteration, it agglomerates (merges) the closest pair of clusters by satisfying some similarity criteria, until all of the data is in one cluster. This method form clusters of latches according to the possible distance range. Here implementing HIRARCHICAL TREE FORMATION to form cluster of latches.

The hierarchy within the final cluster has the following properties: Clusters generated in early stages are nested in

those generated in later stages and Clusters with different sizes in the tree can be valuable for discovery. This method form clusters of latches according to the possible distance range. Here implementing HIRARCHICAL TREE FORMATION to form cluster of latches[22].

4. Merge Flip Flop

When a flip-flop should be merged, then flip flops to update the merging group of feasible regions. The sequence of adding a FR into merging group can be determined according to the same criterion of flip-flops selection rule.

RESULTS AND DISCUSSION

The cluster using 2 FF's is implemented in the VHDL language and the experiments are performed on a Linux machine Xilinx with inteli3 processor with 4GB memory. The timing information can be derived by using STA. Two flip flops are formed by using pulsed D-latch to reduce the power. In this there two inputs are used that D and D2 and the clock signal is used to triggering the flip flips depending the clock signal the flip flops are enabled. The Figure 6.2 shows the RTL schematic of cluster using 2 FF's. The two master slave flip flops are used. The master and slave flip flops using the D-latch. There four latches are used. The clock signal is used enable the flip flops. The cluster using 2 FF's is implemented in the VHDL language and the experiments are performed on a Linux machine Xilinx with inteli3 processor with 4-GB memory. Timing information can be derived by using STA. Two flip flops are formed by using pulsed D-latch to reduce the power. When the clock input is low there is no change in the output. If clock is high there is a change in the output signal depending on the input signal. In this D flip flops are used to design the cluster. The clusters are designed using 5-FF. The proposed system is designed using the mixture circuit. The mixture circuits consist of pulsed latch and flip flop circuit.

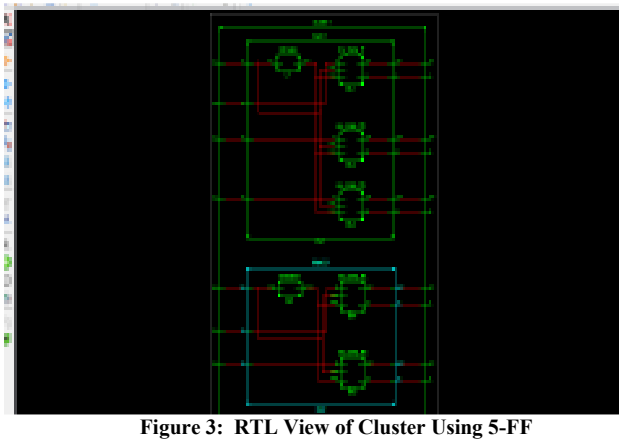

Figure 3: RTL View of Cluster Using 5-FF

The Fig 3 shows the RTL view of cluster using five flip flops. The three flip flops are connected in one pulse generator and remaining is connected with one pulse generators. There are two cluster are formed in this system. By using this method the power in the system is reduced. The maximum tolerable load of pulse generator should be much smaller than that to avoid pulse degradation. To provide a reasonable setting, we set the maximum tolerable load and maximum fan out constraint of the multi-type pulse generator. To prevent pulse distortion, the total load of a pulse generator cannot exceed the defined tolerable load and the maximum fan out constraint during the migration process.

A. System Power Comparison

The cluster using 2 FF's is implemented in the VHDL language and the experiments are performed on a Linux machine Xilinx with inteli3 processor with 4GB memory. The timing information can be derived by using STA. Two flip flops are formed by using pulsed D-latch to reduce the power. The power in the system is reduced by reducing the load level in pulse generator. Here for five flip flops only two pulse generators are reduced. The Fig. 4 shows the power of clock tree using 5-FF. the mixture circuit reduces the power consumption of clock tree. The pure flip flop circuit designed using only flip flops. The power consumption of pure flip flop circuit, pulsed latch circuits and mixture circuits for different design is shown in the Table I. The pure pulsed latch circuit using the only pulsed latch circuits. The proposed migration approach is using the both flip flop circuit and pulsed latch circuit. The power consumption in the proposed migration circuit is varied from the pure flip flop circuit and pure pulsed latch circuit.

The pure flip flop circuit using the only flip flops to design the circuit. The power consumption of pure flip flop, pure pulsed latch circuit and mixture circuit different design is shown in Figure

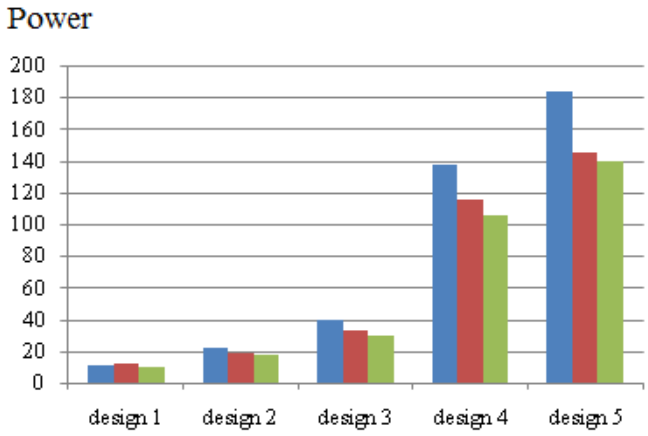

Figure 4 (a): System Power Comparison Graph

The pure pulsed latch circuit using the only pulsed latch circuits. The proposed migration approach is using the both flip flop circuit and pulsed latch circuit. The power consumption in proposed circuit is less than using the pure flip flop circuit and pure pulsed latch circuits. The power consumption variation of mixture circuit is less than the pure. flip flop circuit and pure pulsed latch circuits.

B. System Timing Information Comparison

The timing information of pure flip flop circuit, pure pulsed latch circuit and proposed circuits are varied according to clock skew and latency. The pure flip flop circuit designed using only flip flops.

The pure pulsed latch circuit using the only pulsed latch circuits. The proposed migration approach is using the both

flip flop circuit and pulsed latch circuit. The pure flip flop circuit using the only flip flops to design the circuit. The clock skew of pure flip flop circuit, pure pulsed latch circuit and mixture circuit for different design is shown in Fig.4 (b).

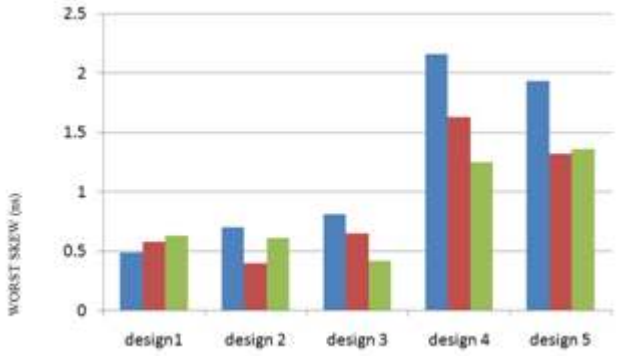

Figure 4 (b): System Clock Skew Comparison Graph

The pure pulsed latch circuit using the only pulsed latch circuits. The proposed migration approach is using the both n the proposed circuit is less compared to pure flip flop circuit and pure pulsed latch circuit. The pure flip flop circuit designed using only flip flops. The pure pulsed latch circuit using the only pulsed latch circuits. The proposed migration approach is using the both flip flop circuit and pulsed latch circuit.

The pure flip flop circuit using the only flip flops to design the circuit. The clock skew of pure flip flop circuit, pure pulsed latch circuit and mixture circuit for different design is shown in Fig. 5.

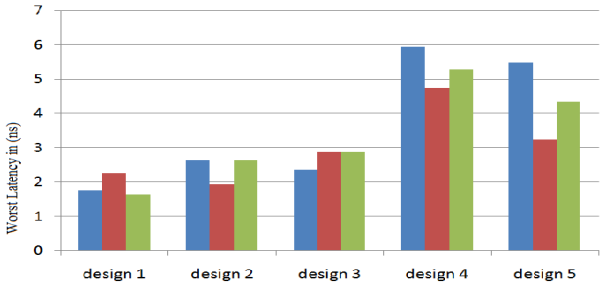

Figure 5: System

circuit. The clock skew in the proposed circuit is less compared to pure flip flop circuit and pure pulsed latch circuit.

CONCLUSION Latency Comparison Graph

The pure pulsed latch circuit using the only pulsed latch circuits. The proposed migration approach is using the both flip flop circuit and pulsed latch AND FUTURE WORK

CONCLUSION

To prevent pulse degradation, the tolerable load of a pulse generator and the number of pulsed latches driven by a pulse generator were considered during pulse-generator insertion. To further reduce the power dissipation of pulse generators, we enabled multi-type pulse generators and identified the pulse generators with suitable size to drive pulsed latches. Considering the tradeoff between the additional power of multi-type pulse-generator insertion and the power savings of pulsed-latch substitution, not all flip-flops were replaced. This allows the migrated clock tree to be constructed with a mixed structure of latches and flip-flops. To determine the topology configuration and simultaneously minimize the wire length and load balance, we applied a minimum-cost maximum-flow formulation to solve the pulsed-latch-clustering problem.

FUTURE WORK

In this system the load level of clock tree, the power consumption and clock skew are reduced by using agglomerative clustering algorithm in the H-tree based architecture. Multi-corner multi-mode (MCMM) Clock tree synthesis (CTS) is used in H-Tree based topology to reduce the load level in clock tree. Experimentally load level of clock tree, the power and clock skew of the system is improved by system. The proposed system will be implemented using using the agglomerative clustering algorithm in H tree based architecture. Xilinx tool will be used to carry out the proposed Hardware Description Language (VHDL).

REFERENCES

1. Usami K and Horowitz,M Clustered voltage scaling technique for low-power design, in Proc. Int. Symp. Low Power Electron. Design, 1995; 3–8.
2. Wu Q Pedram M and Wu X, Clock-gating and its application to low power design of sequential circuits," IEEE Trans. Circuits Syst. I, Reg. Papers, 2000; 47 (3): 415–420.
3. Farrahi AH, Chen C, Srivastava A, Tellez G and Sarrafzadeh M, Activity-driven clock design, IEEE Trans. Comput. Aided Design, 2001; 20(6): 705–714.
4. Oh J and Pedram MM, Gated clock routing for low-power microprocessor design, IEEE Trans. Comput. Aided Design, 2001; 20 (6): 715–722.
5. Pangjun J and Sapatnekar MM, Low-power clock distribution using multiple voltages and reduced swings, IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 2002; 10 (3): 309–318.
6. Venkatraman A, Garg R and Khatri SP, A robust, fast pulsed flip-flop design, in Proc. Great Lakes Symp. VLSI, 2008; 119–122.
7. Hou W, Liu D and Ho PH, Automatic register banking for low-power clock trees, in Proc. Qual. Electron. Design, 2009; 647–652.
8. Shelar RS, An efficient clustering algorithm for low power clock tree synthesis," in Proc. Int. Symp. Phys. Design, 2007; 181–188.
9. Chuang YL, Kim S, Shin Y and Chang YW, Pulsed-latch-aware placement for timing-integrity optimization, in Proc. 47th ACM/IEEE Design Autom. Conf., 2010; 280–285.
10. Kim SI, Han I, Paik S and Shin Y, Pulser gating: A clock gating of pulsed-latch Circuits," in Proc. Asia South Pacific Design Autom. Conf., 2011; 190–195.
11. Li HC, Chen MC and Ho KM, System and method of replacing flip-flops with pulsed latches in circuit designs," U. S. Patent, 2010; 7: 694 242.
12. Partovi H, Burd R, Salim U, Weber F, Di Gregorio V and Draper D, Flow-through latch and edge-triggered

flip-flop hybrid elements, in IEEE Int. 42nd Solid-State Circuits Conf. Dig. Tech. Papers, 1996; 138–139.
13. Shibatani S and Li A, Pulse-latch approach reduces dynamic power," EE Times, 2006.
14. Lin HT, Chuang YL and Ho TY, Pulsed-latch-based clock tree migration for dynamic power reduction," in Proc. Int. Symp. Low Power Electron. Design, 2011; 39–44.
15. Paik S, Nam G and Shin Y, Implementation of pulsed-latch and pulsed-register circuits to minimize clocking power," in Proc. IEEE/ACM Int. Conf. Comput. Aided Design, 2011; 640–646.
16. Paik S, Han I, Kim S and Shin Y, Clock gating synthesis of pulsed-latch circuits," in Proc. IEEE Trans. Comput. Aided Design Integr. Circuits Syst., 2012; 1019–1030.
17. Fortune S, A sweepline algorithm for Voronoi diagrams, in Proc. Annu. Symp. Comput. Geometry, 1986; 313–322.
18. Edahiro M. A clustering-based optimization algorithm in zero-skew routings, in Proc. 30th Conf. Design Autom., 1993; 612–616.
19. Cherkasssky B, Efficient algorithms for the maximum flow problem, Math. Methods Solution Econ. Problems, 1977; 7: 117–126.
20. Lin KY, Lin HT, Ho TY and Tsai CC, Load-balanced clock tree synthesis with adjustable delay buffer insertion for clock skew reduction in multiple dynamic supply voltage designs," ACM Trans. Design Autom. Electron. Syst., 2012;17 (3): 34.
21. Paik S and Shin Y "Pulsed-latch circuits to push the envelope of ASIC design," in Proc. IEEE Int. SoC Design Conf., Nov. 2010; 150–153.
22. Lee H, Paik S and Shin Y, Pulse width allocation and clock skew scheduling: Optimizing sequential circuits based on pulsed latches," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., 2010; 29 (3): 355–366.

Source of support: Nil, Conflict of interest: None Declared