International Journal on Applications in Electrical and Electronics Engineering
Volume 1:Issue 7: July 2015. pp 1-5. www.aetsjournal.com ISSN (Online) : 2395-3527

An Efficient Clock Gating Logic Using Matching Technique

I.Ribayabegam , S.Sumithra

Abstract— Gate level clock gating establish with netlist. So that it constraints much amount of power across flip-flop and gates. Circuit size enlarge complexity so that it attains execution speed is low. This paper dealt with conquers of these shortcomings by matching technique. In which factored forms of the gating functions are harmonized with factored forms of the Boolean functions of presented combinational nodes in the circuit. Strong match discover matches that are overtly present in the factored forms. Delay-matching uses gated cells whose timing characteristics are similar to that of their clock buffer (inverter) counterparts. This technique has been tested using Xilinx v 14.2 software packages using Very High Speed Integrated circuit hardware description language (VHDL), RTL and technology schematic are included to validate simulation results.

Index Terms — clock gating, factoring tree, gating logic, buffer logic.

I. INTRODUCTION

Power consumption has become one of the major concerns in modern chip design. The clock circulation network utilize 30-40% of power from the processor[1] .Power consumed by a CMOS circuit can be divided into two components: (1) static power, mainly caused by sub-threshold and gate leakage, and (2) dynamic power, Caused by switching activity[2]. Although in recent sub-micron designs static power is becoming ever more important, dynamic power is still a foremost component. One of the most extensively used techniques for dropping dynamic power is clock gating, a number of techniques to diminish the dynamic power have been urbanized, of which clock gating is predominant. Clock gating is a popular technique used in many synchronous circuits for reducing dynamic power dissipation. Clock gating saves power by adding more logic to a circuit to clip the clock tree. Clipping the clock disables portion of the circuitry so that the flip-flops in them do not have to switch states. Switching states consumes power. When not being switched, the switching power consumption goes to zero, and only leakage currents are acquired. Clock gating works by taking the enable conditions attached to registers, and utilizes them to gate the clocks. This clock gating process can also save significant die area as well as power, since it removes large numbers of muxes and replaces them with clock gating logic. This clock gating logic is generally in the form of "Integrated clock gating" (ICG) cells. However, note that the clock gating logic will alter the clock tree structure, since the clock gating logic will sit in the clock tree.

I.Ribayabegam, PG Student(M.E-Communication And Networking) , Pavendar Bharathidasan College of Engineering and Technology, Trichy, Tamilnadu, India. (Email: er.ribayabegam@gmail.com Contact no:8489202078)

S.Sumithra , Professor, Department of ECE, Pavendar Bharathidasan College of Engineering and Technology, Trichy, Tamilnadu, India. (Email: raghavi24@yahoo.com)

A. Motivation

Gate-level clock gating is an eventual approach is to simplify gating functions from gate-level netlist. A few techniques are discussed to simplify gating functions in the following section. This paper mainly focus on diminish circuit's power and reduce circuit's delay. In this paper develops a new technique to simplify gating functions

The key to this approach is to use the presented logic as far as achievable while the gating functions are produced. This is accomplished by matching factored forms of the gating functions with those of presented logic nodes, thus we name this method as factored form matching[3]. We will present two matching methods: 1) strong matching(SM),which looks for matches that are overtly present in an appearance of the logic circuit and 2) delay matching which uses gated cells whose timing characteristics are similar to that of their clock buffer (inverter) counterparts.It attains better slew and much smaller latency with comparable clock skew and less area

II. RELATED WORK

A. Related Work On Clok Gating Logic Simplification

To simplify clock gating logic two different aproaches are considered.

1)Approximation

The onset of *F* can be considered as a don't-care set, for the reason that the functionality of a circuit is not affected whether clock is truly gated or not gated while gating is probable(F=1).Therefore ,F can be estimated by any function F' whose onset is a subset of the onset of F. A appropriate F' can be implemented with fewer logic, but the gating probability will be diminished.

2)Division

The method of using a Boolean division [4], [5]to simplify gating functions is shown in Fig. 1. Let, *D* be a Boolean term for some internal node of a combinational logic circuit. If we execute Boolean division on *F* with *D* as a divisor, we can say *F* as follows:

$$F = D\,Q + R \qquad (1)$$

Where Q→ qutiotent,R→ remainder. Then,it is not only required to implement further logic for Q and R in addition to the AND and OR gates required to form(1).This can be require significantly less logic than a direct implementationf of

F.Runtime is an issue for boolean division because it must be executed numerous times

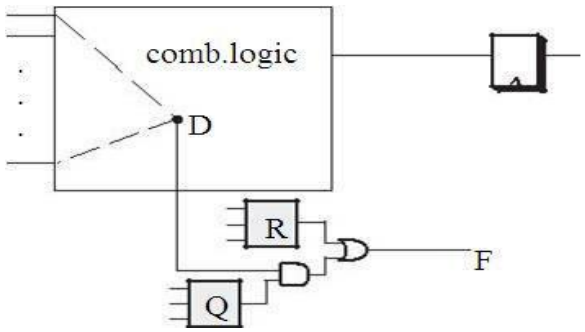

Fig 1.using Boolean division to simplify gating func.

B. Related Work on Boolean Matching

Boolean matching refers to the difficulty of resolving whether two Boolean functions are alike while some inputs may be permuted and some inputs may be complemented. Two techniques are proposed: 1) using a signature and 2) resorting to a canonical form.

Signature of Boolean function symbolizes some properties of function. Cofactor of minterms is used to decide matches under input negation and permutation [6], if number of minterms two or more cofactors are the same then it can't be applied. The numbers of unate (binate) [7] variables and interchangeable variables are also used [8], but the method experiences from large runtime If a few canonical form of Boolean function is defined, matching can be recast as transforming two Boolean functions into canonical forms, which are then compared.

C. Related work on matching factored forms

Factored form matching is to discover some obtainable logic of a combinational circuit, which, together with a few extra gates, implements a gating function. concurrently, we have to exploit the proportion of the gating function that is offered by existing logic, so as to reduce the number of extra gates [9] Because the obtainable logic can also be represented as a factoring tree, the problem can be recast as that of discovering the parts of a factoring tree of a gating function that can be restored by factoring trees taken from the obtainable logic. Week matching looks for matches that are hidden in the logic and thus are more complicated to discover [10]

III. MATCHING TECHNIQUE

Matching technique generally assist to simplify clock gating logic. In which two types of technique is considered. 1) Strong matching 2) Delay matching. Strong match discover matches that are overtly present in the factored forms. Delay-matching uses gated cells whose timing characteristics are similar to that of their clock buffer (inverter) counterparts.

A. Strong Matching

Consider the factoring tree of a gating function *F* shown in Fig. 2(a),and the factoring trees of three of the presented internal nodes shown in (b)–(d). The subtree of *F:* Sb_1 and internal nodes: n_1 are also equivalent[11]; in fact, these trees have the same structure, except for the ordering of children; we say that such trees are syntactically equivalent.Sb_2 and the tree n_2 are structured in different ways, but they actually represent the same Boolean expression $(b+a)c+ba$ we say that sb_2 and n_2 are equivalent. Finally, sb_3 and n_3 are syntactically equivalent, and children are ordered in the same way; rather obviously, we call these identical factored forms. Any of these three types of equality are said to provide a strong match in this paper.fig 3.shows proof of both sb_1 and n_1 are syntactically equivalent. It executes in a manner of bottom-to-up approach. fig 4 and fig 5 shows that VHDL logic for SE & SM

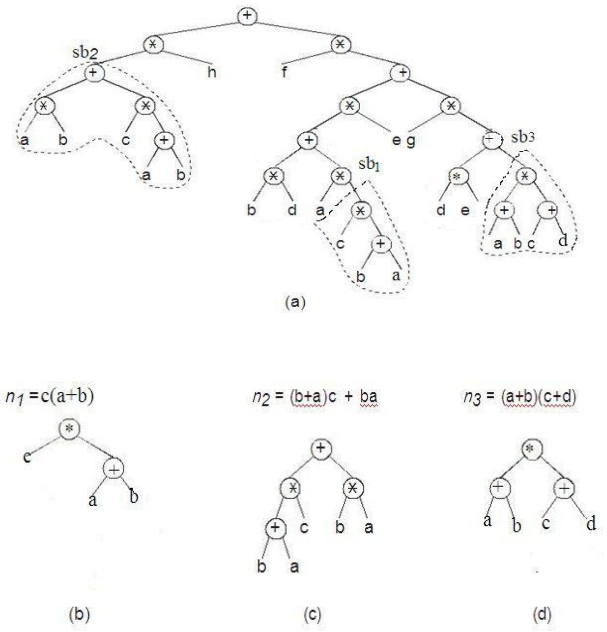

Fig 2.factoring tree of (a) gating function F:of internal nodes(b) which is syntactically equivalent to subtreesb1,(c) which is equivalent to sb2,(d)which is identical to sb3

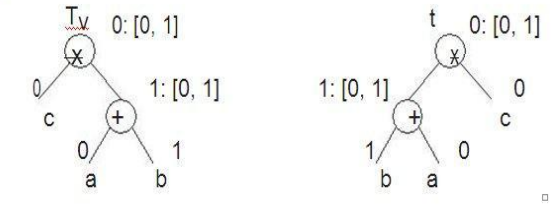

Fig3.checking whether two trees are syntactically equivalent by labeling.

VHDL for Syntatically_euivallent

----------SE begin

u1:dff port map(clk,a,s1); u2:dff port map(clk,b,s2); u3:dff port map(clk,c,s3);

```
u4:or_gate port map(s2,s1,p2);
u5:and_gate port map(s3,p2,p3); sb1_op<=
        p3;
u6:or_gate port map(s1,s2,p2);
u7:and_gate port map(s3,p2,p3);
        n1_op<=p3
    if(sb1_op=n1_op) then
        sb1_op<=n1_op end
  sb1_rtl
```

fig4.VHDL for checking syntactically equivalent tree

VHDL for Strong matching:

```
---------------------------------sm
library ieee;
use ieee.std_logic_1164.all; use
ieee.std_logic_arith.all; use
ieee.std_logic_unsigned.all;
entity sm is
port(
   a,b:in std_logic;


   c:out std_logic
        );
end sm;
architecture sm_rtl of sm is
begin
 process(a,b)
begin
   if(a=b)then
    c<=a;
   elsif(a/=b)then
    c<=b;
end if;
end process;
end sm_rtl;
```

fig 5.VHDL for find strong matching

Delay matching:

In a synchronous sequential circuit[12], the clock signal is used to define a relative time reference for the movement of data. The clock skew is the maximum difference among the clock latencies (i.e., clock delays) from the clock source to flip-flops. Since the clock skew degrades the circuit performance[13], the minimization of clock logic is always a very important topic in the design of synchronous sequential circuit. By taking fig 2. it utilize number of clock signal to perform the clock tree process. Because it uses clock signal for subtree and internal nodes and also it takes amount of time to find the similar matching part. There is an another issue is considered that multiple clock cycle process

.it creates complexity. To reduce delay in fig 2. There is another technique is proposed. To reduce delay level in tree structure clock gated cell timing match with clock buffer gate. [14] Thus we call this technique as delay matching. We know that buffer is a temporary storage. So that it passes that clock signal to another circuit. Fig 2 is modified by fig 6.

Consider fig 6.it executes in a manner of top to down approach. Single clock signal is given to fig 6 and fig 2 of gates (h, f, e, g) consider as control logic (i,e enable lines) and remaining gates modified as buffer gate. N_1, N2, N_3 are considered as subtree. With help of en1 we can execute N2 at a time. Moreover by enabling all enable line we can execute a tree at a time so that we can get multiple number of clock cycle. These multiple numbers of clock cycles is used to provide as input to the other circuit. [15]In this paper we use that multiple number of clock cycle as input of S-27 bench mark circuit shown in fig 8 .fig 7 shows that VHDL logic for buffer gate and subtree

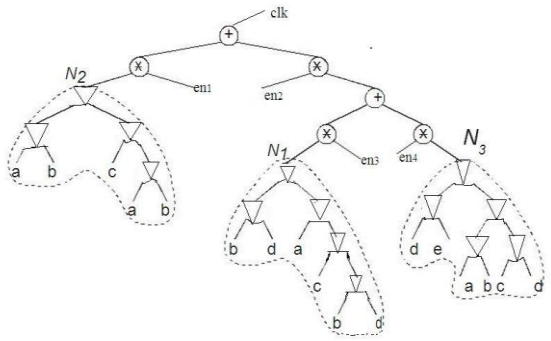

Fig 6.modified tree structure by buffer gate

VHDL for buffer_gate

```
------------------------------
port(a:inout  std_logic;  b:out
   std_logic);
end component;
```

VHDL for subtree N_2

```
----------------------------------
```

```
u1:and_gate      port      map(clk,en1,p1);
u2:buffer_gate   port      map(p1,p2);
u3:buffer_gate   port      map(p2,p3);
u4:buffer_gate   port      map(p2,p4);
u5:buffer_gate port map(p4,p5);

a<=(p3 & p3 & p4 & p5 & p5);
```

fig 7. VHDL for buffer and subtree

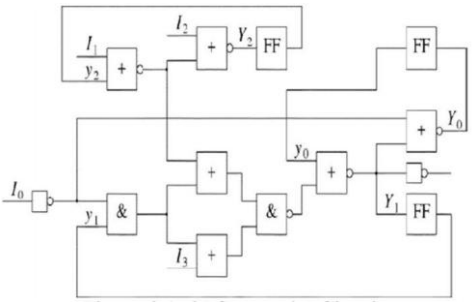

Fig 8. S27 bench mark circuit

Evaluation of Matching

Our matching technique is implemented in fig 2. And fig 6.to test and evaluate the result of matching technique.

A)Power analysis

Power obtained in matching technique is shown in fig 9.Through the help of strong matching technique number of flip-flops and gates are reduced .so that power obtained across those flip-flop and gates are discarded in that way total power of hierarchical structure is reduced. By delay matching technique number of checking function is diminished so that power dissipation is reduced.

B) Delay analysis

If circuit takes hold up time is large then it won't be effective for real time performance. Delay matching helps to reduce clock delay by using control logic. Circuit creates simplicity so that it attains better slew and much smaller latency is shown in fig 9.

matching	Power(W)	Delay(ns)
Strong matching	0.423	1.463
Delay matching	0.102	0.912

Fig 9. Power and delay of matching technique

C) RTL schematic and Technology schematic

Fig 10 and fig 11 shows that RTL and technology schematic diagram of strong matching and delay matching to validate the simulation

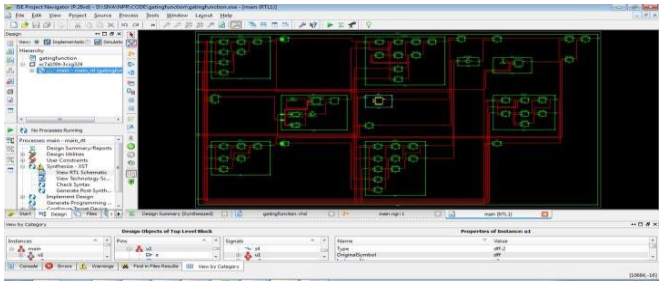

Fig 10 RTL schematic

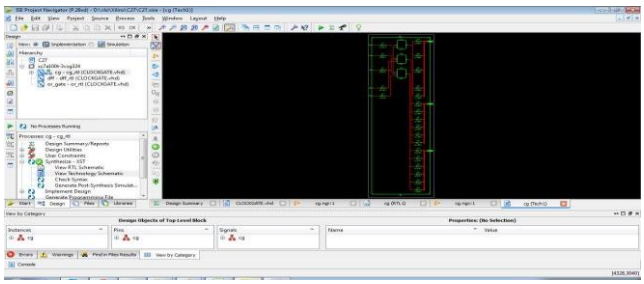

Fig 11.tecchnology diagram

D) Result of matching technique

Finally simulation result of strong math and delay match technique is shown in fig 12 and fig 13 using Xilinx tool 14.2 v.

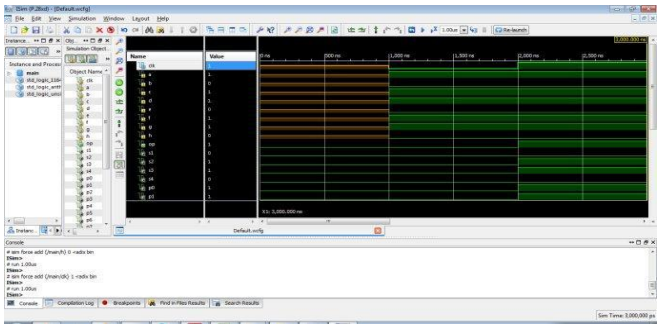

Fig 12.simulation result of strong match technique

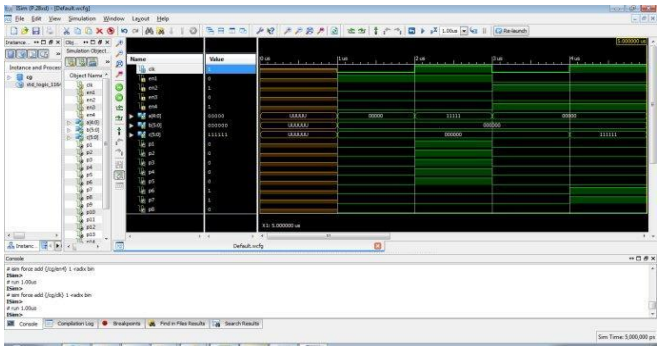

Fig 13.simulation result of delay matching technique

IV. CONCLUSION

Strong matching technique and delay matching technique are the suitable technique for reducing clock gating logic. These technique are used to reduce the gating function and the flip flop architecture. And to improve the system performance by reducing delay and power consumption.finally clock gating logic is simplified by matching technique. In future different type of matching technique is handled to optimize clock gating logic

REFERENCES

[1] A. Hurst, "Automatic synthesis of clock gating logic with con-trolled netlist perturbation," in *Proc. Design Autom. Conf.*, Jun. 2008, pp. 654–657.
[2] E. Arbel, C. Eisner, and O. Rokhlenko, "Resurrecting infeasible clock-gating functions," in *Proc. Design Autom. Conf.*, Jul. 2009, pp. 160–165.
[3] .Inhak Han and Youngsoo shin,"Simplifying clok gating logic by matching factored forms,"in IEEE Trans.Comput.,June 2014,vol.22 No.6.
[4] F. Theeuwen and E. Seelen, "Power reduction through clock gating by symbolic manipulation," in *Proc. Symp. Logic Archit. Design*, Dec. 1996,*pp.* 184–191.
[5] S. Kim, I. Han, S. Paik, and Y. Shin, "Pulser gating: A clock gating of pulsed-latch circuits," in *Proc. Asia South Pacific Design Autom. Conf.*, Jan. 2011, pp. 190–195.
[6] J. Mohnke and S. Malik, "Permutation and phase independent Boolean comparison," *Integr. VLSI J.*, vol. 16, no. 2, pp. 109–129, Dec. 1993.
[7] F. Mailhot and G. De Micheli, "Technology mapping using Boolean matching and don't care sets," in *Proc. Eur. Design Autom. Conf.*, Mar. 1990, pp. 212–216.
[8] F. Mailhot and G. De Micheli, "Algorithms for technology mapping based on binary decision diagrams and on Boolean operations," *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.*, vol. 12, no. 5, 599–620, May 1993
[9] R. Brayton, R. Rudell, A. Sangiovanni-Vincentelli, and A. Wang, "MIS: A multiple-level logic optimization system," *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.*, vol. 6, no. 6, pp. 1062–1081, Nov. 1987.
[10]I. Han and Y. Shin, "Synthesis of clock gating logic through factored form matching," in *Proc. Int. Conf. IC Design Tech.*, Jun. 2012, pp. 1–4.
[11]G. Hatchtel and F. Somenzi, *Logic Synthesis and Verification Algorithms*. Norwell, MA, USA: Kluwer, 1996.
[12]L. Benini, G. De Micheli, E. Macii, M. Poncino, and R. Scarsi, "Symbolic synthesis of clock-gating logic for power optimization of synchronous controllers," *ACM Trans. Design Autom. Electron. Syst.*, vol. 4, no. 4, pp. 351–375, Oct. 1999.
[13]C. M. Chang, S. H. Huang, Y. K. Ho, J. Z. Lin, H. P. Wang, and Y. S.Lu, "Type-Matching Clock Tree for Zero Skew Clock Gating," DAC, 714-719, 2008
[14]Shih-Jung Hsu, Rung-Bin Lin,"Clock Gating Optimization with Delay-Matching", 978-3-9810801-7-9/DATE11/©2011 EDAA.
[15]S.Nagaraju,Ch.N.L.Sujatha,J.S.S.Ramaraju "Test the S27 Benchmark Circuit by Using BuiltIn Self Test and Test Pattern Generation" Vol. 3, Issue 8, August2014