Co-Synthesis of Data Paths and Clock Control Paths for Minimum-Period Clock Gating

Wen-Pin Tu[1], Shih-Hsu Huang[1, 2], and Chun-Hua Cheng[1]

[1]Department of Electronic Engineering
Chung Yuan Christian University
Chung Li, Taiwan, R.O.C.

[2]Information and Communications Research Labs
Industrial Technology Research Institute
Hsin Chu, Taiwan, R.O.C.

g9802602@cycu.edu.tw, shhuang@cycu.edu.tw, g9402601@cycu.edu.tw

***Abstract*—**Although intentional clock skew can be utilized to reduce the clock period, its application in gated clock designs has not been well studied. A gated clock design includes both data paths and clock control paths, but conventional clock skew scheduling only focus on data paths. Based on that observation, in this paper, we propose an approach to perform the co-synthesis of data paths and clock control paths in a nonzero skew gated clock design. Our objective is to minimize the required inserted delay for working with the lower bound of the clock period (under clocking constraints of both data paths and clock control paths). Different from previous works, our approach can guarantee no clocking constraint violation in the presence of clock gating. Experimental results show our approach can effectively enhance the circuit speed with almost no penalty on the power consumption.

Keywords—Clock Period Minimization, Delay Insertion, Clock Gating, Data Path Synthesis.

I. INTRODUCTION

It is well known that intentional clock skew can be utilized as a manageable resource to reduce the clock period [1]. The minimum-period clock skew scheduling problem [2] is to find the minimum clock period and the corresponding clock skew schedule under clocking constraints.

Generally, it is assumed that the path delay has been minimized and cannot be further reduced during the stage of clock skew scheduling. As a result, setup constraints give a lower bound on the clock period [3]. On the other hand, hold violations can be resolved by applying the delay insertion. Therefore, several research efforts [4-7] have been paid to study the combination of clock skew scheduling and delay insertion for further clock period reduction. Moreover, since the delay insertion is an engineering change order (ECO) process, there is also a demand to reduce the required inserted delay. In [7], Huang et al. propose a linear program (LP) to formally draw up the minimization of the required inserted delay.

In addition to clock period minimization, since the clock signal is the most active signal in the circuit, it is also very important to distribute the clock signal with low power. Clock gating [8,9] is a widely-used design technique to save the power consumption. However, traditionally, gated clock designs [8,9] are developed under the assumption of zero clock skew circuits. The impact of clock gating on clock skew scheduling has not been well studied. In this paper, we present the first work to deal with this problem. We have the following two observations on this problem.

(1) Previous clock skew scheduling researches [1,2] only focus on clocking constraints of data paths (i.e., clocking constraints between data signals and the clock signal). However, in a gated clock design, clocking constraints of clock control paths (i.e., clocking constraints between enable signals and the clock signal) may also limit the utilization of intentional clock skew.

(2) Previous delay insertion researches [4-7] only resolve hold violations in data paths. In a gated clock design, to resolve hold violations in clock control paths, the delay insertion in clock control paths is needed.

Based on those above observations, in this paper, we are motivated to perform the co-synthesis of data paths and clock control paths for gated clock designs via the simultaneous application of clock skew scheduling and delay insertion.We use a LP to formally draw up this problem. Our objective is to minimize the required inserted delay for working with the lower bound of the clock period (under clocking constraints of both data paths and clock control paths). Compared with previous works [1-7] that only focus on data paths, our approach can guarantee no clocking constraint violation in the presence of clock gating with almost no penalty on the power consumption.

II. PRELIMINARIES

A. Clocking Constraints of Data Paths

An edge-triggered circuit is composed of registers and logic gates, with wires connecting them. Let's use circuit ex1 shown in Figure 1 for example. This circuit has two registers and four logic gates. We use the notation U1/A to denote pin A of logic gate U1. A *timing arc* of a logic gate refers to the signal propagation from its input pin to its output pin. For example, in circuit ex1, logic gate U1 has a timing arc from U1/A to U1/Y. A *timing path* is defined as a path of signal propagation from an output pin of a register to an input pin of another register. A *data path* $R_i \rightarrow R_j$ refers to the combinational logic from the data output pin of register R_i to the data input pin of register R_j. Using circuit ex1 as an example, data path R1→R2 has two timing paths: R1/Q→U2/A→U2/Y→U3/A→U3/Y→R2/D and R1/Q→ U3/B→U3/Y→R2/D.

978-3-9815370-0-0/DATE13/©2013 EDAA

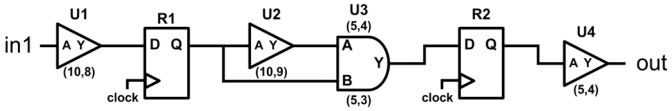

Figure 1: Edge-triggered circuit ex1.

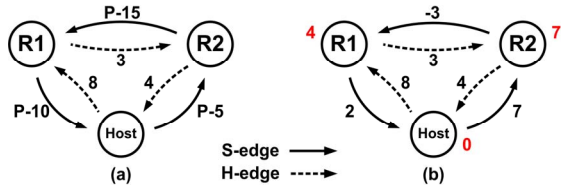

Figure 2: Constraint graphs of circuit ex1.

In Figure 1, each timing arc from pin a to pin b is associated with a pair of delay values ($D_{ARCa,b(max)}$,$D_{ARCa,b(min)}$), where the notation $D_{ARCa,b(max)}$ and the notation $D_{ARCa,b(min)}$ denote its maximum delay and its minimum delay, respectively. For each data path $R_i \rightarrow R_j$, we use the notation $D_{DPi,j(max)}$ and the notation $D_{DPi,j(min)}$ to denote its maximum delay and its minimum delay, respectively.

For each data path $R_i \rightarrow R_j$, there are two types of clocking constraints: the setup constraint and the hold constraint. Let the notation T_{Ci} represent the designated clock arrival time of register R_i. The setup constraint corresponds to $T_{Ci}-T_{Cj} \leq P-D_{DPi,j(max)}$, where P is the clock period, and the hold constraint corresponds to $T_{Cj}-T_{Ci} \leq D_{DPi,j(min)}$. In a constraint graph, each vertex represents a register, and each directed edge represents a clocking constraint. A special vertex, called the *Host*, is used for the synchronization of primary inputs and primary outputs. The setup constraint of data path $R_i \rightarrow R_j$ is modeled as an S-edge $e_s(R_j,R_i)$, which is from register R_j to register R_i and associated with a weight $P-D_{DPi,j(max)}$. The hold constraint of data path $R_i \rightarrow R_j$ is modeled as an H-edge $e_h(R_i,R_j)$, which is from register R_i to register R_j and associated with a weight $D_{DPi,j(min)}$. A constraint graph works with clock period P, provided that it has no negative cycle. Several graph-based algorithms [2] have been proposed to solve the minimum-period clock skew scheduling problem in polynomial time complexity. Figure 2(a) gives the constraint graph of circuit ex1. By applying minimum-period clock skew scheduling, we find that the minimum clock period is 12 and the corresponding clock skew schedule is $T_{Host}=0$, $T_{C1}=4$, and $T_{C2}=7$. Figure 2(b) gives the constraint graph when the clock period is 12. For the convenience of readers, in Figure 2(b), we label T_{Ci} for each vertex R_i.

B. The Lower Bound of Clock Period

During the stage of clock skew scheduling, it is generally assumed that the path delay has been minimized. Therefore, setup constraints give a lower bound on the clock period [3]. We use the notation P_{set} to denote this lower bound. Using circuit ex1 shown in Figure 1 for example, we have $P_{set}(ex1) = 10$. Due to the limitation of hold constraints, minimum-period clock skew scheduling often does not achieve this lower bound. In circuit ex1, the minimum-period clock skew scheduling only achieves 12. With an analysis to Figure 2(b), we find H-edge $e_h(R1,R2)$ is in the critical cycle.

Next, we address the limitation of delay insertion. We say the delay difference of timing path p is $D(p)-d(p)$, where $D(p)$ and $d(p)$ are the maximum delay and the minimum delay of timing path p, respectively. From [7], we know: if a clock skew schedule has satisfied all setup constraints, the largest delay difference among all timing paths gives a lower bound of clock period for inserting delays to satisfy all hold constraints without affecting the circuit speed. We use the notation P_{ins} to denote this lower bound. Take circuit ex1 as an example, the largest delay difference among all timing paths is 2. Consequently, we have $P_{ins}(ex1) = 2$.

Let the notation P_{LB} denote the lower bound of the clock period that the combination of clock skew scheduling and delay insertion can achieve. From [7], we know that P_{LB} = maximum (P_{set},P_{ins}). Furthermore, Huang et al. [7] propose a linear program to minimize the required inserted delay for working with clock period P_{LB}. Take circuit ex1 for example. We have $P_{LB}(ex1)$ = maximum(10,2) = 10. By applying the linear program proposed in [7] to circuit ex1, a delay element, whose delay value is 2, is added into the wire from R1/Q to U3/B. As a result, we obtain circuit ex2 as displayed in Figure 3. Note that circuit ex2 works with clock period 10 under $T_{Host}=0$, $T_{C1}=0$, and $T_{C2}=5$.

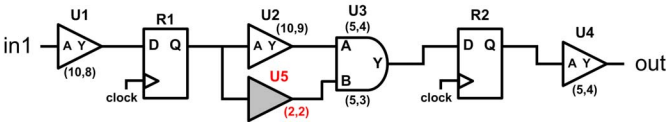

Figure 3: Edge-triggered circuit ex2.

III. MOTIVATION

A. Clocking Constraints of Clock Gating

There are two clock gating styles: latch-free and latch-based. In Figure 4, we use D-type flip-flop as an example. The latch-free clock gating uses a simple AND gate as shown in Figure 4(a). The latch-based clock gating adds a level-sensitive latch as shown in Figure 4(b).

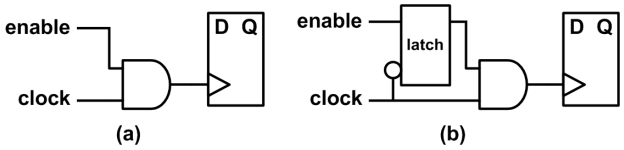

Figure 4: Clock gating. (a) Latch-free. (b) Latch-based.

For the convenience of discussing clock gating, we let the notation A_{Ci} to be the designated clock arrival time of the AND gate of register R_i. Furthermore, for the AND gate of register R_i, we let the notation $D_{ANDi(max)}$ and the notation $D_{ANDi(min)}$ denote its maximum delay and its minimum delay, respectively. Note that, if register R_i has no clock gating, both the values of $D_{ANDi(max)}$ and the values of $D_{ANDi(min)}$ are assumed to be 0. Obviously, for each register R_i, the value

of T_{Ci} is within the range between $A_{Ci}+D_{ANDi(min)}$ and $A_{Ci}+D_{ANDi(max)}$. Then, for each data path $R_i \rightarrow R_j$, the setup constraint is re-written as $(A_{Ci}+D_{ANDi(max)})-(A_{Cj}+D_{ANDi(min)}) \leq P-D_{DPi,j(max)}$, and the hold constraint is re-written as $(A_{Cj}+D_{ANDi(max)})-(A_{Ci}+D_{ANDi(min)}) \leq D_{DPi,j(min)}$.

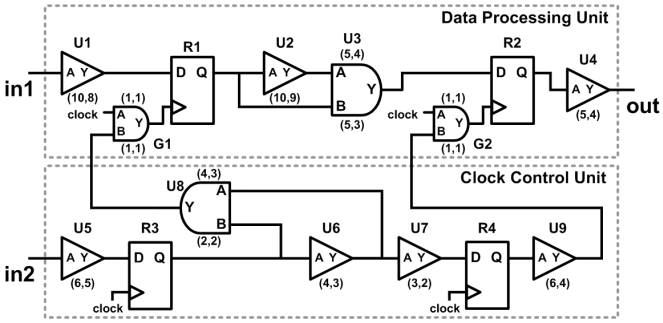

Figure 5: Gated clock design ex3.

In addition to data paths, a gated clock design also has clock control paths. A *clock control path* $R_i \rightarrow R_j$ refers to the combinational logic from the data output pin of register R_i to the enable input pin of the AND gate of register R_j. For each clock control path $R_i \rightarrow R_j$, we use the notation $D_{CPi,j(max)}$ and the notation $D_{CPi,j(min)}$ to denote its maximum delay and its minimum delay, respectively. Take gated clock design ex3 shown in Figure 5 for example. For the clock control path R3→R1, we have $D_{CP3,1(max)} = 8$ and $D_{CP3,1(min)} = 2$. In the following, we discuss clocking constraints of latch-free clock gating and latch-based clock gating, respectively.

(1) *Clocking constraints of latch-free clock gating*. To generate the complete clock pulse of gated clock, the enable signal should be held constant from the rising edge of the clock until the falling edge of the clock. Since the enable signal (for the following clock pulse) should arrive at the AND gate earlier than the rising edge of the following clock pulse, the setup constraint is $(A_{Ci}+D_{ANDi(max)})-A_{Cj} \leq P-D_{CPi,j(max)}$. Since the enable signal (for the following clock pulse) should arrive at the AND gate later than the falling edge of the current clock pulse, the hold constraint is $A_{Cj}-(A_{Ci}+D_{ANDi(min)}) \leq D_{CPi,j(min)}-\alpha \times P$, where α is a constant that denotes the duty cycle and $0 < \alpha < 1$.

(2) *Clocking constraints of latch-based clock gating*. The latch captures the enable signal and holds it until the complete clock pulse has been generated. Thus, the enable signal needs only be stable around the rising edge of the clock. Since the enable signal (for the following clock pulse) should arrive at the AND gate earlier than the rising edge of the following clock pulse, the setup constraint is $(A_{Ci}+D_{ANDi(max)})-A_{Cj} \leq P-D_{CPi,j(max)}$. Since the enable signal should arrive at the AND gate later than the rising edge of the current clock pulse, the hold constraint is $A_{Cj}-(A_{Ci}+D_{ANDi(min)}) \leq D_{CPi,j(min)}$.

In a gated clock design (no matter latch-free clock gating or latch-based clock gating), clocking constraints of clock control paths may limit the utilization of intentional clock skew. We use circuit ex3 as an example. Without loss of generality, we assume that latch-free clock gating is used and the duty cycle is 1/2 (i.e., 50%). We consider the following two constraint graphs.

(1) Figure 6(a) gives the constraint graph in which only data paths are considered. The minimum clock period is 12. Figure 6(b) gives the constraint graph when the clock period is 12.

(2) Figure 7(a) gives the complete constraint graph in which both data paths and clock control paths are considered. The minimum clock period is 14. Figure 7(b) gives the constraint graph when the clock period is 14.

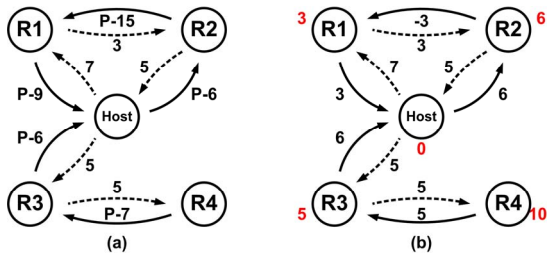

Figure 6: Constraint graph when only data paths.

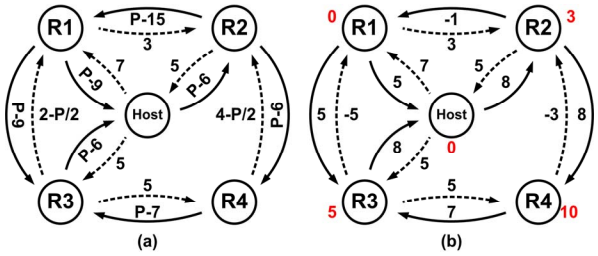

Figure 7: Complete constraint graph of circuit ex3.

In this example, we find that clocking constraints of clock control paths limit the utilization of intentional clock skew. The minimum clock period should be 14 (instead of 12). Since previous clock skew scheduling researches [1,2] only consider clocking constraints of data paths, in the presence of clock gating, they will derive a clock skew schedule that has clocking constraint violations in clock control paths.

Further, if we only consider clocking constraints of data paths, the clock skew schedule is $A_{Host}=0$, $A_{C1}=3$, $A_{C2}=6$, $A_{C3}=5$, and $A_{C4}=10$. This clock skew schedule cannot work with any clock period under both clocking constraints of data paths and clock control paths. We give the proof below.

(1) From the constraint graph shown in Figure 8(a), we know the hold constraint of clock control path $R_3 \rightarrow R_1$ (i.e., H-edge $e_h(R3,R1)$) is $A_{C1}-A_{C3} \leq 2-P/2$. Since $A_{C1}=3$ and $A_{C3}=5$, we have the requirement that the clock period should be less than or equal to 8.

(2) With an analysis to the constraint graph shown in Figure 8(a), we have $P_{set}(ex3) = 10$. Thus, we have the requirement that the clock period should be greater than or equal to 10.

Since the above two requirements conflict with each other, the clock skew schedule A_{Host}=0, A_{C1}=3, A_{C2}=6, A_{C3}=5, and A_{C4}=10 cannot work with any clock period under both clocking constraints of data paths and clock control paths.

B. Delay Insertion in Clock Control Paths

With an analysis to Figure 7(a), we have P_{set}(ex3) = 10. With an analysis to Figure 5, we have P_{ins}(ex3) = 2. Therefore, we have P_{LB}(ex3) = maximum(10,2) = 10.

On the other hand, the minimum clock period of circuit ex3 obtained by minimum-period clock skew scheduling is 14. With an analysis to Figure 7(b), we find that the minimum clock period is limited by the critical cycle consisting of S-edge e_s(R1,R3) and H-edge e_h(R3,R1). Therefore, if we perform the delay insertion in clock control path R3→R1, the minimum clock period can be further reduced.

However, previous delay insertion researches [4-7] is that they only focus on clocking constraints of data paths. Based on that observation, in this paper, we are motivated to perform the delay insertion in clock control paths. For circuit ex3, our approach will derive circuit ex4 as shown in Figure 8 to work with clock period P_{LB}(ex3) (i.e., the lower bound of the clock period). Compared with circuit ex3, in circuit ex4, a delay element U10 (whose delay value is 2) is added into the data path R1→R2, and a delay element U11 (whose delay value is 2) is added into the clock control path R3→R1.

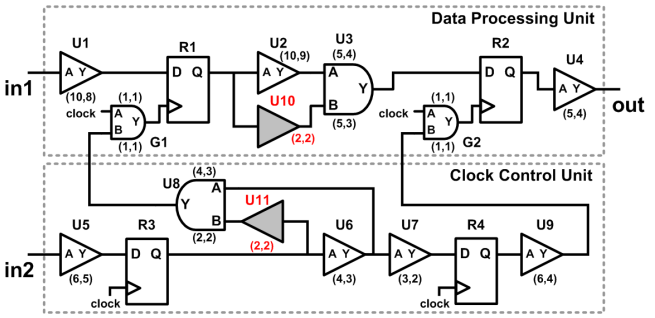

Figure 8: Our gated clock design ex4.

IV. OUR APPROACH

In this section, we propose a linear program LP to formally draw up our approach.

A. Proposed Linear Program

In our LP, the value P_{LB} (the lower bound of the clock period) is a constant. Our objective is to find a clock skew schedule that uses the minimum amount of the required inserted delay for working with clock period P_{LB}.

First, we introduce the variables used in our LP. For each register R_i, we define a variable A_{Ci} to be the designated clock arrival time of the AND gate of register R_i. Note that, if register R_i has no clock gating, we let both the value of $D_{ANDi(max)}$ and the value of $D_{ANDi(min)}$ to be 0. For each pin u, we define a variable EA_u and a variable LA_u to denote its earliest data arrival time and its latest data arrival time, respectively. For each wire from pin u to pin v, we define a variable $X_{u,v}$ to denote its delay insertion. Then, the objective function of our LP is to minimize the summation of delay insertions.

We describe the constraints of our liner program below. For every wire from pin u to pin v, its delay insertion should be non-negative. As a result, we have the following constraint:

$$0 \le X_{u,v} \qquad \text{(Formula 1)}$$

For every timing arc (of a logic gate) from pin u to pin v, the earliest data arrival time of pin v should not be greater than the earliest data arrival time of pin u plus the minimum delay of this timing arc, while the latest data arrival time of pin v should not be less than the latest data arrival time of pin u plus the maximum delay of this timing arc. As a result, we have the following two constraints:

$$EA_v \le EA_u + D_{ARCu,v(min)} \qquad \text{(Formula 2)}$$

$$LA_u + D_{ARCu,v(max)} \le LA_v \qquad \text{(Formula 3)}$$

For every wire from pin u to pin v, the earliest data arrival time of pin v should not be greater than the earliest data arrival time of pin u plus the delay insertion of this wire, while the latest data arrival time of pin v should not be less than the latest data arrival time of pin u plus the delay insertion of this wire. As a result, we have the following two constraints:

$$EA_v \le EA_u + X_{u,v} \qquad \text{(Formula 4)}$$

$$LA_u + X_{u,v} \le LA_v \qquad \text{(Formula 5)}$$

Suppose that pin u is the starting pin of data path $R_i \rightarrow R_j$. The earliest (latest) data arrival time of pin u should be the earliest (latest) clock arrival time of register R_i. Therefore, we have the following two constraints:

$$EA_u = A_{Ci} + D_{ANDi(min)} \qquad \text{(Formula 6)}$$

$$LA_u = A_{Ci} + D_{ANDi(max)} \qquad \text{(Formula 7)}$$

Suppose that pin v is the ending pin of data path $R_i \rightarrow R_j$. Due to the hold constraint, the earliest data arrival time of pin v should not be earlier than the earliest clock arrival time of register R_j. Thus, we have the following constraint:

$$A_{Cj} + D_{ANDj(min)} \le EA_v \qquad \text{(Formula 8)}$$

Suppose that pin v is the ending pin of data path $R_i \rightarrow R_j$. Due to the setup constraint, the latest data arrival time of pin v should not be later than the latest clock arrival time (for the following clock pulse) of register R_j. Therefore, we have the following constraint:

$$LA_v \le A_{Cj} + D_{ANDj(max)} + P_{LB} \qquad \text{(Formula 9)}$$

There are two clock gating styles: latch-free and latch-based. Suppose that pin v is the ending pin of clock control path $R_i \rightarrow R_j$. If register R_j is in the style of latch-free, due to the hold constraint, the earliest enable signal arrival time of pin v should not be earlier than the falling edge of the current clock pulse. Thus, we have the following constraint:

$$A_{Cj} + \alpha \times P_{LB} \le EA_v \qquad \text{(Formula 10)}$$

On the other hand, if register R_j is in the style of latch-based, due to the hold constraint, the earliest enable signal arrival time of pin v should not be earlier than the rising edge of the current clock pulse. Thus, we have the following constraint:

$A_{Cj} \leq EA_v$ (Formula 10')

Suppose that pin v is the ending pin of clock control path $R_i \rightarrow R_j$. Due to the setup constraint, the latest enable signal arrival time of pin v should not be later than the arrival time of clock rising edge (for the following clock pulse). Therefore, we have the following constraint:

$LA_v \leq A_{Cj} + P_{LB}$ (Formula 11)

Since our formulas are necessary and sufficient, we can solve this problem exactly and optimally. Further, it should be mentioned that a LP can be solved in polynomial time complexity [7].

We use circuit ex3 shown in Figure 5 for illustration. After solving the LP, we find that $X_{R1/Q,U3/B} = 2$, $X_{R3/Q,U8/B} = 2$, and the delay insertions of other wires are 0. As a result, we obtain circuit ex4 shown in Figure 8.

B. Generalization

In this subsection, we make a generalization to explain our approach can be easily applied to any complex clock gating. First, let's consider the condition that several registers share the same AND gate for clock gating. Obviously, for these registers, these AND gate clock arrival times should be the same. Using Figure 9(a) as an example, since register R1 and register R2 share the same AND gate, we add an extra constraint $A_{C1} = A_{C2}$ into our LP.

Next, let's consider the condition that a register R_i uses more than one AND gates for clock gating. In this condition, the constraint for $EA_{Ri/D}$ (i.e., Formula 6), the constraint for $LA_{Ri/D}$ (i.e., Formula 7), the constraint for $EA_{Ri/Q}$ (i.e., Formula 8), and the constraint for $LA_{Ri/Q}$ (i.e., Formula 9) should be modified according to the clock gating structure of register R_i. Further, suppose that pin v is the ending pin of a clock control path from a register to register R_i. The constraint for EA_v (i.e., Formula 10 or Formula 10') and the constraint for LA_v (i.e., Formula 11) should also be modified according to the clock gating structure of register R_i.

Let's use Figure 9(b) as an example. In this example, register R1 uses two AND gates for clock gating. Let the notation A_{C1} denote the designated clock arrival time of the AND gate U1. Let the notation $D_{U1(max)}$ and the notation $D_{U1(min)}$ denote the maximum delay and the minimum delay of AND gate U1, respectively. Then, the constraint for $EA_{R1/D}$ becomes $EA_{R1/D} = A_{C1} + D_{U1(min)} + D_{U2(min)}$; the constraint for $LA_{R1/D}$ becomes $LA_{R1/D} = A_{C1} + D_{U1(max)} + D_{U2(max)}$; the constraint for $EA_{R1/Q}$ becomes $A_{C1} + D_{U1(min)} + D_{U2(min)} \leq EA_{R1/Q}$; and the constraint for $LA_{R1/Q}$ becomes $LA_{R1/Q} \leq A_{C1} + D_{U1(max)} + D_{U2(max)} + P_{LB}$.

Besides, in Figure 9(b), we observe that pin U1/B and pin U2/B are the ending pins of clock control paths from registers to register R1. Suppose that register R1 uses latch-free clock gating. Then, we have the constraints $A_{C1} + \alpha \times P_{LB} \leq EA_{U1/B}$, $LA_{U1/B} \leq A_{C1} + P_{LB}$, $A_{C1} + D_{U1(max)} + \alpha \times P_{LB} \leq EA_{U2/B}$, and $LA_{U2/B} \leq A_{C1} + D_{U1(min)} + P_{LB}$.

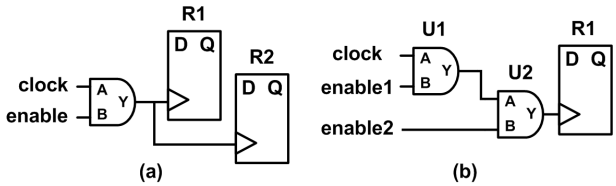

Figure 9: Complex clock gating conditions.

V. EXPERIMENTAL RESULTS

We use seven circuits, which are targeted to TSMC 0.13μm process technology, to test the effectiveness of our approach. Circuits S13207, S15850, S35932, S38417, and S38584 are adopted from ISCAS'89 benchmark suite. Originally, these ISCAS'89 circuits are not gated clock designs. For the purpose of experiment, we arbitrarily create their clock control units. Circuits IND1 and IND2 are real-life industry gated clock designs.

In our experiments, we use Extended-LINGO Release 11.0 as the linear program solver (running on Intel Core i7 processor). In each circuit, our CPU time is within very few minutes. We compare the following five design methods.

(1) Method *ZS*. The gated clock design is implemented as a zero skew circuit.

(2) Method *MCSS*. We apply minimum-period clock skew scheduling based on only clocking constraints of data paths but determine the minimum clock period value under both clocking constraints of data paths and clock control paths.

(3) Method *DI*. We apply [7] based on only clocking constraints of data paths but determine the minimum clock period value under both clocking constraints of data paths and clock control paths.

(4) Method *CMCSS*. We apply minimum-period clock skew scheduling based on both clocking constraints of data paths and clock control paths.

(5) Method *Ours*. The gated clock design is implemented by our approach.

Table 1 and Table 2 tabulates the comparisons on the minimum clock periods (in nanoseconds) of these five methods under the assumption of latch-free clock gating and latch-based clock gating, respectively. If the clock skew schedule obtained by method MCSS or method DI cannot work with any clock period under both clocking constraints of data paths and clock control paths, we use the notation *NA* to denote the clock period. We have the following three observations:

(1) In each circuit, no matter latch-free clock gating or latch-based clock gating, our approach achieves clock period P_{LB}.

(2) The hold constraint of latch-based clock gating is looser than the hold constraint of latch-free clock gating. Therefore, for the same circuit, when methods MCSS, DI, or CMCSS are used, latch-based clock gating has a smaller minimum clock period than latch-free clock gating.

(3) In circuits S13207 and S35932, their minimum clock periods obtained by method MCSS and method DI are denoted as *NA* because their clock skew schedules cannot work with any clock period.

Table 3 tabulates the required inserted delays of our approach and method DI (here we do not analyze circuits S13207 and S35932, since their clock skew schedules obtained by method DI do not work with any clock period). Our approach requires a larger amount of delay insertion than method DI because our approach performs delay insertion in both data paths and clock control paths.

A delay insertion may be implemented by buffer insertion or gate downsizing. If a delay insertion is implemented by gate downsizing, the power consumption is often decreased; if a delay insertion is implemented by inserting an extra buffer, the power consumption is often increased. To save the power, we try to use gate downsizing as possible. Table 4 tabulates the comparisons on the power consumptions between method *DI* and our approach (under the minimum clock period of method DI). The average overhead of our approach on the power consumption is only 1.16%.

VI. CONCLUSIONS

This paper studies the simultaneous application of clock skew scheduling and delay insertion for the co-synthesis of data paths and clock control paths in a gated clock design. Our work is the first attempt for the development of nonzero skew gated clock designs. We propose a LP to formally draw up this problem. Our approach can achieve both the lower bound of the clock period and the lower bound of the required inserted delay. Note that our approach is applicable to any complex clock gating. Experimental results show that our approach can effectively enhance the circuit speed with almost no penalty on the power consumption.

ACKNOWLEDGMENT

This work was supported in part by the National Science Council of Taiwan, R.O.C., under grant number NSC 99-2221-E-033-061-MY3.

REFERENCES

[1] J.P. Fishburn, "Clock Skew Optimization", IEEE Trans. on Computers, vol. 39, no. 7, pp. 945—951, 1990.

[2] R.B. Deokar and S.S. Sapatnekar, "A Graph-Theoretic Approach to Clock Skew Optimization", Proc. of IEEE ISCAS, vol. 1, pp.407—410, 1994.

[3] M.C. Papaefthymiou, "Understanding Retiming through Maximum Average-Delay Cycles", Mathematical Systems Theory, vol. 27, pp. 65—84, 1994.

[4] S.H. Huang and Y.T. Nieh, "Clock Period Minimization of Non-Zero Clock Skew Circuits", Proc. of IEEE/ACM ICCAD, pp. 809—812, 2003.

[5] S.H. Huang, Y.T. Nieh, and F.P. Lu, "Race-Condition-Aware Clock Skew Scheduling", Proc. of IEEE/ACM DAC, pp. 475—478, 2005.

[6] B. Taskin and I.S. Kourtev, "Delay Insertion Method in Clock Skew Scheduling", IEEE Trans. on CAD, vol. 25, no.4, pp. 651—663, 2006.

[7] S.H. Huang, C.H. Cheng, C.M. Chang, and Y.T. Nieh, "Clock Period Minimization with Minimum Delay Insertion", Proc. of IEEE/ACM DAC, pp. 970—975, 2007.

[8] Q. Wu, M. Pedram, and X. Wu, "Clock Gating and Its Application to Low Power Design of Sequential Circuits", IEEE Trans. on CAS – I: Fundamental, Theory, and Applications, vol. 47, no. 103, pp. 415—420, 2000.

[9] J. Srinivas, M. Rao, S. Jairam, H. Udayakumar, and J. Rao, "Clock Gating Effectiveness Metrics: Applications to Power Optimization", Proc. of IEEE ISQED, pp. 482—487, 2009.

Table 1: Comparisons on clock periods (latch-free gating).

Circuit	ZS	MCSS	DI	CMCSS	Ours
S13207	2.55	NA	NA	2.40	1.76
S15850	8.04	8.02	7.53	7.68	7.48
S35932	12.06	NA	NA	11.11	9.46
S38417	11.06	11.06	10.22	9.98	8.04
S38584	19.06	18.95	18.83	17.57	16.23
IND1	2.84	2.44	2.34	2.44	1.84
IND2	3.28	2.52	2.42	2.52	1.90

Table 2: Comparisons on clock periods (latch-based gating).

Circuit	ZS	MCSS	DI	CMCSS	Ours
S13207	2.55	NA	NA	1.76	1.76
S15850	8.04	7.92	7.53	7.68	7.48
S35932	12.06	NA	NA	11.11	9.46
S38417	11.06	11.06	10.16	9.98	8.04
S38584	19.06	18.95	18.77	17.57	16.23
IND1	2.84	2.44	2.34	2.44	1.84
IND2	3.28	2.52	2.32	2.42	1.90

Table 3: The required inserted delays (nanosecond).

Circuit	Latch-free		Latch-based	
	DI	Ours	DI	Ours
S15850	0.36	0.36	0.21	0.21
S38417	1.02	1.95	1.02	1.95
S38584	0.78	1.34	0.89	1.34
IND1	0.14	8.93	2.13	6.28
IND2	1.90	4.03	1.90	4.03

Table 4. Comparisons on the power consumptions (mW).

Circuit	Latch-free			Latch-based		
	DI	Ours	Overhead	DI	Ours	Overhead
S15850	8.33	8.32	-0.12%	9.15	9.14	-0.11%
S38417	21.58	21.4	-0.83%	23.85	23.59	-1.09%
S38584	19.42	19.44	0.10%	21.32	21.35	0.14%
IND1	19.02	20.11	5.73%	22.26	22.98	3.23%
IND2	32.63	32.88	0.77%	33.71	34.97	3.74%