Timing Analysis of Dual-Edge-Triggered Flip-Flop Based Circuits with Clock Gating

Chungki Oh, Sangmin Kim, and Youngsoo Shin
Department of Electrical Engineering, KAIST
Daejeon 305-701, Korea

Abstract— **Dual-edge-triggered flip-flop (DETFF) allows us to use half the clock frequency while maintaining the same throughput, thereby cutting the clock power in half. DETFF-based design, however, requires multiple runs of timing analysis, which is critical for short turn-around time; to make it worse, the number of analysis increases if we use clock gating and multiple clock gating logic, both of which are typical in practical designs. Alternative approach is to perform analysis once assuming the tightest timing condition, which turns out to be too pessimistic. Timing analysis method for DETFF-based circuit with clock gating is proposed for the first time. The method is based on identifying a cluster of nets that have to be associated with multiple required arrival times (RATs); the remaining nets having a single RAT then can be processed by conventional timing analysis. Experiments with several benchmark circuits in 65-nm technology demonstrate that, at 50% point of cumulative slack histogram, the slack from our analysis was 1.78× on average of the slack from conventional timing analysis assuming the tightest timing condition, and 1.30× at 90% point.**

I. Introduction

Synchronous sequential circuits, such as FSM controllers and pipelined circuits, consist of (clocked) storage elements and combinational logic. Latches and flip-flops, two examples of storage elements, are typically activated at single polarity of clock, e.g. latch that is transparent when clock is high but opaque when clock is low or flip-flop that captures data at rising edge but not at falling edge of clock. Flip-flop, in particular, is called single-edge-triggered (SETFF) if it is triggered either at rising or at falling, but not at both.

Dual-edge-triggered flip-flop (DETFF), on the other hand, is triggered at both clock edges, thereby launching and capturing data at a rate twice that of SETFF for the same clock period. Employing DETFFs, therefore, allows us to use half the clock frequency while keeping the same throughput, implying that the switching power at clock network is cut in half; this can be significant in overall power consumption since clock network takes large proportion of total power in high-performance microprocessors as well as in ASICs, e.g. 20% to 40% [1], [2]. This, of course, comes at a cost: DETFF-based circuits are affected by uncertainty of clock duty ratio as well as uncertainty of clock period [3] as opposed to SETFF-based ones that are affected by clock period alone, meaning that we need more timing guardband.

Another limitation of DETFF-based circuits is the difficulty of timing analysis. Any combinational block between two DETFFs is expected to work under two different timing conditions: data launched at rising-edge and captured at falling-edge, and data launched at falling-edge and captured at rising-edge. Note that the amount of time allowed to combinational block in these two conditions is different when duty ratio is not 0.5 and timing parameters of DETFF at rising-edge of clock do not match those at falling-edge. There are even more number of conditions we have to check if we employ clock gating for DETFF, which is very different from clock gating of SETFF as described in Section II. We may simply check the timing under the tightest timing condition, which turns out to be very pessimistic.

We propose timing analysis method for DETFF-based circuits employing clock gating. We show that, to remove pessimism, multiple required arrival times (RATs) have to be derived for a cluster of nets that are close to primary outputs, which cannot be processed by using conventional static timing analysis (STA); the remaining nets having a single RAT, once identified, then can be submitted to conventional STA. In experiments on sequential benchmark designs, we compared cumulative slack histograms, one obtained from proposed method and the other from conventional STA assuming the tightest timing condition. At 50% point of histogram, the slack from our method was 1.78× on average, meaning that 78% more slacks in 50% of nets, even though they exist, cannot be discovered by using conventional STA alone; at 90% point of histogram, the slack from our method was 1.30×.

The remainder of this paper is organized as follows: in the next section we briefly review DETFF and clock gating of DETFF-based circuits. In Section III, the timing analysis algorithm for DETFF-based circuits with clock gating is described. Experimental results are presented in Section IV, and we draw conclusions in Section V.

II. Preliminaries

A. DETFF

Since DETFF launches and captures data at both rising- and falling-edge of clock, its timing parameters, namely the setup time T_{su}, clock-to-Q delay T_{cq}, and hold time T_{hd}, are defined with respect to two clock edges, denoted by a superscripted sign, e.g. T_{su}^{+} for setup time against rising-edge and T_{su}^{-} against falling-edge. There are various implementations of DETFFs [4]–[8], but the timing parameters typically have different values against different edges [3], e.g. T_{su}^{+} and T_{su}^{-} are different.

In sequential circuits based on DETFFs, timing analysis has to be performed twice as we note in Section I. Alternative approach would be to perform timing analysis just once

978-1-4244-2934-9/09/$25.00 ©2009 IEEE

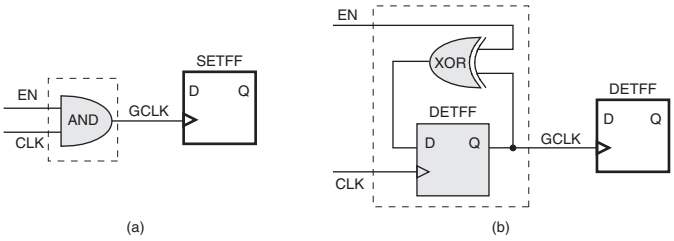

Fig. 1. Clock gating in (a) SETFF and (b) DETFF [10].

under the tightest timing condition. In specific, we assume DETFFs are single-edge-triggered with their setup time being the maximum of T_{su}^{+} and T_{su}^{-} (similarly for clock-to-Q delay and hold time), set the clock period to the minimum of two clock phases (clock being high and clock being low), and run the conventional STA. This, however, is pessimistic, i.e. even though STA reports timing violation, a circuit may still work without any violation.

B. Clock Gating of DETFF-Based Circuits

Clock gating is widely used to reduce clock power consumption. Fig. 1(a) illustrates a concept of clock gating in rising-edge-triggered SETFF. It consists of AND gate with one input from clock source (CLK) and the other from clock gating logic (EN); gating (EN=0) forces gated clock (GCLK) to be low, and un-gating (EN=1) lets GCLK follow CLK (with the delay of AND gate). Note that EN can change its value only when CLK is low (otherwise any glitch from EN may propagate to GCLK when CLK is high), which is realized via filtering out EN by using a negative-level-sensitive latch.

In DETFF, when gating is applied, GCLK has to maintain its current value rather than to be forced to some fixed logic value as in SETFF (note that GCLK is always forced to 0 when EN=0 in Fig. 1(a)), otherwise it may trigger DETFF to capture data, which may cause a malfunction. In un-gating, there are two approaches: force GCLK to follow CLK [9], or let GCLK to follow CLK or $\overline{\text{CLK}}$ whichever comes first [10]. In the first approach, if the value of GCLK, which is hold now, is different from that of CLK when un-gating is performed, GCLK has to wait until it becomes the same as CLK. In the second approach shown in Fig. 1(b), which we consider in this paper due to its faster response and simpler implementation, GCLK follows CLK (with clock-to-Q delay of DETFF within the dotted box) when un-gating is applied while CLK and GCLK have the same value; otherwise it follows $\overline{\text{CLK}}$ (again with clock-to-Q delay of DETFF).

III. Timing Analysis of DETFF-Based Circuits with Clock Gating

A. Problem Formulation

If we employ clock gating shown in Fig. 1(b) for DETFF-based circuits, GCLK either follows CLK or $\overline{\text{CLK}}$ with some delays as shown in Fig. 2. The amount of delay is determined by clock-to-Q delay of DETFF within dotted box of Fig. 1(b), which typically varies depending on the polarity of CLK it

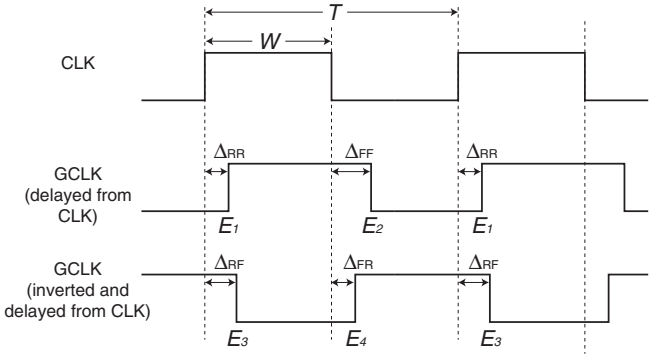

Fig. 2. Waveforms of gated clock (GCLK) in DETFF-based circuits with clock gating, where T is clock period, W is clock width (the period of clock being high), and E_i denotes GCLK edge with its delay from CLK also shown.

is triggered by and the value of D-input, e.g. Δ_{RR} =182 ps, Δ_{FF} =114 ps, Δ_{RF} =142 ps, and Δ_{FR} =72 ps in our SPICE simulation of [7] as our DETFF with 65-nm commercial technology. This, in conjunction with duty ratio W/T when it is not exactly 0.5, imposes various timing conditions that have to be checked for any combinational paths between two flip-flops.

Consider a combinational block between flip-flop i and j, denoted by $i \rightsquigarrow j$. If i and j are driven by the same clock gating logic, i.e. if they are connected to the same GCLK, there are four different timing conditions we have to check for $i \rightsquigarrow j$, namely from E_1 to E_2, from E_2 to E_1, from E_3 to E_4, and from E_4 to E_3 (see Fig. 2). In other words, the maximum delay of $i \rightsquigarrow j$, denoted by D_{ij}, has to satisfy

$$T_{cq}^{+} + D_{ij} + T_{su}^{-} \le d(E_1, E_2), \tag{1}$$
$$T_{cq}^{-} + D_{ij} + T_{su}^{+} \le d(E_2, E_1), \tag{2}$$
$$T_{cq}^{-} + D_{ij} + T_{su}^{+} \le d(E_3, E_4), \tag{3}$$
$$T_{cq}^{+} + D_{ij} + T_{su}^{-} \le d(E_4, E_3), \tag{4}$$

where $d(E_p, E_q)$ denotes the time interval between E_p and E_q, which is determined by clock period, duty ratio, and the delay between CLK and GCLK edges, e.g.

$$d(E_1, E_2) = W + \Delta_{FF} - \Delta_{RR}. \tag{5}$$

The conditions from (1) to (4) can be combined into

$$D_{ij} \le \min[\min(d(E_1, E_2), d(E_4, E_3)) - T_{cq}^{+} - T_{su}^{-}, \\ \min(d(E_2, E_1), d(E_3, E_4)) - T_{cq}^{-} - T_{su}^{+}]. \tag{6}$$

The right-hand side of (6), therefore, forms the timing constraint imposed on $i \rightsquigarrow j$, which is denoted by TC_{ij}. Hold time constraint can be established in a similar way.

When i and j are driven by different clock gating logic (i.e. they belong to different functional blocks that are gated and un-gated in different times or if we employ clock gating in fine granularity), there are four more timing conditions we have to consider: from E_1 to E_4, from E_4 to E_1, from E_3 to E_2, and E_2 to E_3; timing constraint TC_{ij} can be defined similarly as (6) if we include these additional conditions.

978-1-4244-2934-9/09/$25.00 ©2009 IEEE

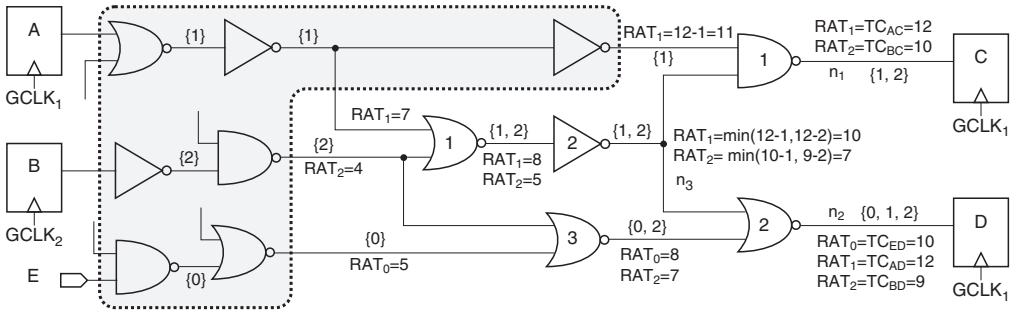

Fig. 4. An example for timing analysis.

Algorithm *STA_DETFF*
begin
L1 Derive TC_{ij} for all $i \rightsquigarrow j$
L2 Propagate GCLK tags from PIs to POs
L3 Initialize AT of each PI to 0
L4 Set RATs of each PO to corresponding TC_{ij}s
L5 Propagate RATs from POs to PIs, until nodes of single GCLK tag can be met
L6 Identify a region of nodes with single GCLK tag, and run STA on it
L7 Propagate ATs from nodes of a single GCLK tag to POs
end

Fig. 3. Pseudo-code of timing analysis of DETFF-based circuits with clock gating.

B. Timing Analysis Algorithm

The algorithm *STA_DETFF* shown in Fig. 3 performs timing analysis of DETFF-based circuits with clock gating; Fig. 4 will be used as an example. We first derive timing constraint TC_{ij} (see (6)) for each combinational block between i and j (L1); combinational blocks between circuit input and circuit output, between circuit input and flip-flop, and between flip-flop and circuit output are also considered (e.g. TC_{ED} at n_2 in Fig. 4, which is between circuit input and flip-flop). We propagate from primary inputs (PIs) to primary outputs (POs) an identifier of GCLK signal, called GCLK tag (L2). In Fig. 4, flip-flop A receives $\texttt{GCLK}_1$ and B receives $\texttt{GCLK}_2$, thus we propagate tags 1 and 2; note that E is circuit input thus is not clock gated, but it is also identified with tag 0. The GCLK tags shown within curly brackets at each net indicate a list of GCLK signals for launching flip-flops that affect the net, e.g. n_1 can be reached from flip-flops A and B, which are driven by $\texttt{GCLK}_1$ and $\texttt{GCLK}_2$, respectively.

The arrival time (AT) of each PI is initialized to 0 (L3); this is relative arrival time, since real AT at flip-flop output, which is T_{cq}, or the one at circuit input, which is specified by designers as a constraint, is embedded in TC_{ij} itself (see (6)), which we consider as RAT at PO. The RATs of each PO are set to TC_{ij}s according to GCLK tags (L4), for example of net n_1, TC_{AC} between A (corresponding to tag 1) and C and TC_{BC} between B (tag 2) and C. Note that TC_{AC}, which is determined by a single GCLK waveform, and TC_{BC}, which is determined by two independent GCLK waveforms, are different. The number of RATs at a particular node is thus equal to its number of GCLK tags.

In L5, we propagate RATs from POs to PIs, but only those corresponding to the tags associated with each net. Consider n_3; to compute RAT_1, the RAT corresponding to tag 1, we take RAT_1s from n_1 and n_2, subtract the delay of corresponding timing arc (which is denoted within a gate), and take a minimum; we repeat the same process for RAT_2. These RATs are propagated until we reach the nets of a single GCLK tag, i.e. until there is a single RAT, because propagating a single RAT can be done by conventional STA. A sub-circuit consisting of a single RAT is identified (within dotted outline in Fig. 4) and submitted to STA; once ATs are returned from STA, they are propagated toward POs and slack is derived at each net.

IV. Experimental Results

Timing analysis algorithm presented in Section III-B was implemented in SIS [11]. Experiments, based on 65-nm commercial technology, were carried out on a set of sequential circuits taken from the ISCAS and ITC benchmarks together with some circuits extracted from several open cores [12], which are summarized in the first three columns of Table I. We compared two methods of timing analysis: conventional STA assuming the tightest timing condition (see Section II-A) and the proposed STA (denoted by conservative and proposed, respectively, in Table I).

We first compared the maximum frequency achievable under two timing analysis methods, which is shown in columns 4-5. Conventional STA cannot discover the whole amount of timing slacks due to its pessimism, especially in timing critical paths, which is why its maximum frequency is smaller than the one obtained by the proposed method.

In the second experiment, we fixed the frequency of each circuit to the one in the fourth column, the frequency found by conventional STA. Then, we derived two cumulative slack histograms corresponding to each STA method. Example histograms are shown in Fig. 5 for two benchmark circuits (for example of sbc, about 80% of nets have slacks less than 170 ps in conventional STA). We see the clear shift of histogram to the right with the proposed STA meaning that more timing slacks, which cannot be discovered with conventional STA, can be

978-1-4244-2934-9/09/$25.00 ©2009 IEEE

TABLE I

COMPARING CONVENTIONAL- AND PROPOSED-STA FOR MAXIMUM ACHIEVABLE CLOCK FREQUENCY, AND THE SLACK AT 50% AND 90% POINT OF CUMULATIVE SLACK HISTOGRAM

Benchmark			Maximum frequency [MHz]		50% cumulative slack [ps]		90% cumulative slack [ps]	
Name	# Gates	# FFs	Conservative	Proposed	Conservative	Proposed	Conservative	Proposed
b03	275	30	800	888	61	123	213	275
b07	601	42	573	618	72	146	296	365
b09	170	28	852	952	41	101	140	202
b11	543	30	653	710	68	141	207	292
b12	2083	119	535	573	103	174	251	337
b14	10490	245	274	286	124	201	568	659
b17	22457	1414	205	211	659	721	1575	1633
b22	19840	703	271	280	272	336	723	789
s400	146	21	936	1058	32	104	164	233
s838	245	32	386	386	712	777	847	938
s1423	831	74	585	593	70	131	374	454
s5378	1289	160	826	831	99	139	247	388
s13207	2870	474	497	500	457	525	619	680
can_fifo	6220	877	449	464	172	234	507	543
irda_fifo	2830	527	711	753	45	118	256	330
sbc	661	27	838	849	105	153	212	415
Average			1.00	1.05	1.00	1.78	1.00	1.30

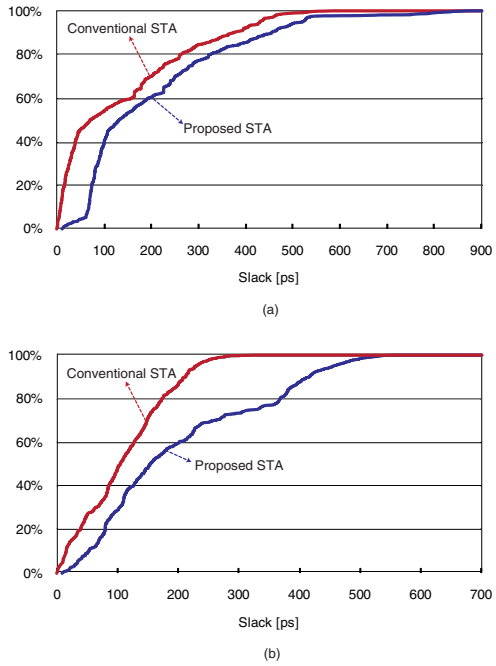

Fig. 5. Cumulative slack histogram of (a) s1423 and (b) sbc benchmark circuits.

extracted. In columns 6-7 of Table I, the slacks at 50% point of histograms are compared showing that the slacks with the proposed method is 1.78× on average, which means that 78% more slacks in 50% of nets, even though they exist, cannot be discovered by using conventional STA alone. At 90% point of histogram shown in the last two columns, the slack from our method is 1.30×.

V. CONCLUSION

DETFF has significant advantage over SETFF in that clock power is cut in half. It however has limited applications. There are two reasons: it asks for more timing guardband, i.e. less amount of time is allowed to combinational logic; multiple timing runs are required that increases turn-around time, which takes one of the highest priority in ASIC designs. In order to resolve the second issue, we have proposed a timing analysis method that does not need multiple runs in DETFF-based circuits without assuming any pessimism, thus is accurate.

REFERENCES

[1] J. Friedrich, B. McCredie, N. James, B. Huott, B. Curran, E. Fluhr, E. Chan G. Mittal, D. Plass Y. Chan, S. Chu, H. Le, L. Clark, J. Ripley, S. Taylor, J. Dilullo, and M. Lanzerotti, "Design of the Power6 microprocessor," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2007, pp. 96–97.

[2] D. Chinnery and K. Keutzer, *Closing the Gap Between ASIC & Custom*, Kluwer Academic Publishers, 2002.

[3] V. G. Oklobdzija, V. M. Stojanovic, D. M. Markovic, and N. M. Nedovic, *Digital System Clocking: High-Performance and Low-Power Aspects*, John Wiley & Sons, Inc., 2003.

[4] M. Afghahi and J. Yuan, "Double edge-triggered D-flip-flops for high-speed CMOS circuits," *IEEE Journal of Solid-State Circuits*, vol. 26, no. 8, pp. 1168–1170, Aug. 1991.

[5] A. Gago, R. Escano, and J. A. Hidalgo, "Reduced implementation of D-type DET flip-flops," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 3, pp. 400–402, Mar. 1993.

[6] R.P. Llopis and M. Sachdev, "Low power, testable dual edge triggered flip-flops," in *Proc. Int. Symp. on Low Power Electronics and Design*, Aug. 1996, pp. 341–345.

[7] M. Pedram, Q. Wu, and X. Wu, "A new design for double edge triggered flip-flops," in *Proc. Asia South Pacific Design Automation Conf.*, Feb. 1998, pp. 417–421.

[8] W. M. Chung, T. Lo, and M. Sachdev, "A comparative analysis of low-power low-voltage dual-edge-triggered flip-flops," *IEEE Trans. on VLSI Systems*, vol. 10, no. 6, pp. 913–918, Dec. 2002.

[9] J. W. Tschanz, D. Somasekhar, and V. K. De, "Gating for dual edge-triggered clocking," Mar. 2006, U.S. Patent 7109776.

[10] R. P. Llopis, "Electronic circuit with dual edge triggered flip-flop," Oct. 2000, U.S. Patent 6137331.

[11] E. M. Sentovich et al., "SIS: a system for sequential circuit synthesis," May 1992, Tech. Rep. UCB/ERL M92/41.

[12] "Opencores," http://www.opencores.org/.

978-1-4244-2934-9/09/$25.00 ©2009 IEEE