

# Pulsed-Latch Circuits to Push the Envelope of ASIC Design

Seungwhun Paik and Youngsoo Shin  
Department of Electrical Engineering, KAIST  
Daejeon 305-701, Korea

**Abstract**—The use of the slow and power-consuming flip-flops is one of the factors that cause a large gap between custom and ASIC designs. A pulsed-latch, which is a latch driven by a brief pulse clock, inherits the advantage of latch while allowing us to use a simple timing model similar to that of flip-flop. As a result, it offers the opportunity of higher performance and lower power consumption within the conventional ASIC design environment. We address challenges and problems specific to pulsed-latch ASIC, and review potential solutions. Some quantitative results are provided to assess the effectiveness of pulsed-latch circuits.

## I. INTRODUCTION

Most ASIC designs use an edge-triggered flip-flop as a sequencing element. This is because of simple timing model it offers; each combinational block between two flip-flops can be considered isolated. This enables timing analysis at higher design abstraction, which then supports timing-driven synthesis. An example is technology mapping, which receives Boolean expressions and determines corresponding connection of logic gates. The mapping can be performed while assuming that the computation of each expression is finished within a clock period.

However, flip-flops take an appreciable portion of clock period, total power consumption, and circuit area. The proportion of sequencing overhead of flip-flop (i.e. sum of clock-to-Q delay and setup time) in a clock period increases as clock frequency increases due to ever-increasing demand on high performance. The sequencing overhead of a typical flip-flop is 6 FO4 delay [1]; this is 13% of 46 FO4 delay of clock period, and 17% and 21% when clock period becomes 35 and 29 FO4 delay, respectively [2]. Clock distribution network often contributes more than half of total power consumption; almost half of the clocking power is consumed by flip-flops [3].

Since flip-flops are commonly designed by cascading two transparent latches, their sequencing overhead is about twice that of latches. This explains why latches are frequently used in high-performance custom designs. Latches also have smaller clock load due to smaller number of clocked transistors. However, time borrowing, along with requirement of two-phase non-overlapping clocking, complicates the timing analysis of latches, which makes them difficult to use in ASIC designs. In addition, data has to be held for a longer period of time, increasing the likely number of hold-time violations.

A pulsed-latch is a latch driven by a pulse clock. The amount of time borrowing that can be exploited is very small due to the brief pulse width, and is typically ignored in ASIC

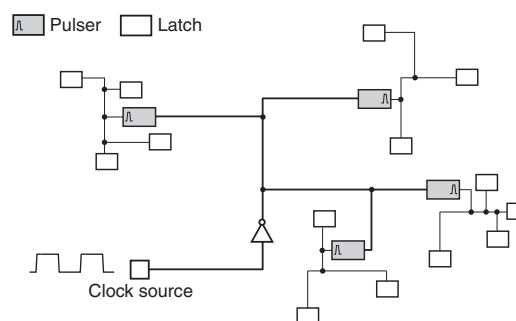


Fig. 1. A pulsed-latch circuit.

design to simplify the timing model. Consequently, a pulsed-latch can be approximated as a faster and smaller flip-flop taking advantages of both flip-flops and latches. This enables a simple migration of a flip-flop circuit to a pulsed-latch version by substituting all (or some) sequencing elements [4], which offers the opportunity of saving in clock period and power consumption. A new design methodology and tools are required for a complete environment of ASIC design based on pulsed-latches, which is a subject of this paper.

## II. CONVERSION OF FLIP-FLOP TO PULSED-LATCH CIRCUITS

In a pulsed-latch circuit, a normal clock is delivered from a clock source to multiple pulse generators (called pulsers), which are dispersed in a placement region. Each pulser then delivers a pulse to more than one latch to amortize the cost of pulser; the latches must be placed close to the pulser since pulse may be distorted when routed over a long distance. Fig. 1 shows an example of a pulsed-latch circuit.

### A. Resolving Hold Time Violations

A main drawback of pulsed-latch circuits is increasing risk of hold time violations due to a large hold time. A data launched at the rising edge of pulse is not allowed to arrive at capturing latch before hold time past falling edge; more hold time violations occur, therefore, with increasing pulse width. The violations are typically removed by inserting delay buffers [5]–[7]. Increasing the delay of short path through re-synthesis [8] is another method to fix the violations.

In our experiments with test circuits, the difference between buffer insertion and re-synthesis in total area turned out to be small, which are compared in Fig. 2, even though their netlists are very different. Fig. 2 shows the area overhead,

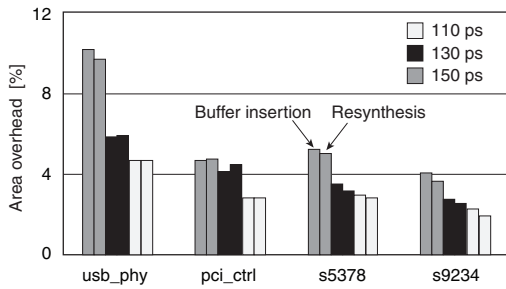


Fig. 2. Area overhead to fix hold-time violations by inserting delay buffers (left bars) and re-synthesis (right bars) under various pulse widths.

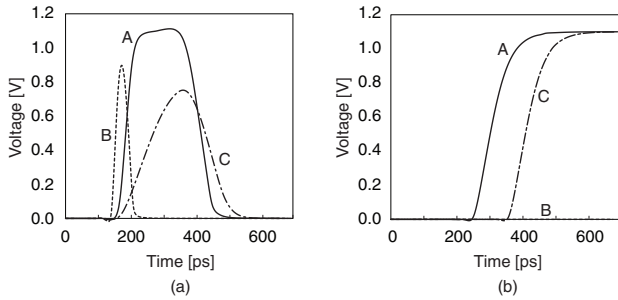


Fig. 3. (a) Various pulse clocks applied to a latch, and (b) waveforms of latch output when rising input data is applied.

which is the proportion of area increase over the initial area, to fix hold-time violations under several pulse widths. We tried three different pulse widths (i.e. 110, 130, and 150 ps), where 110 ps is a minimum value recommended in 45-nm technology we used. Clearly, the area overhead increases as pulse becomes wider. Even in 110 ps, the area overhead is not negligible, which signifies the importance of hold-time violations in pulsed-latch circuits.

### B. Pulser: Timing Integrity, Insertion, and Placement

1) *Timing Integrity*: A distortion in the shape of pulse clock can deteriorate the timing integrity of pulsed-latch circuits. The timing integrity of pulsed-latch circuits can be assessed in two perspectives: failure to capture input data at a latch and impaired timing parameter when data is safely captured. This is illustrated using a SPICE simulation in Fig. 3. Three different pulse clocks shown in Fig. 3(a) are applied to a standard latch; Fig. 3(b) shows the waveform of latch output. When pulse B is applied, the data is not captured due to narrow width. When pulse C is applied, which is quite distorted in magnitude and slew, the latch can capture the data but the clock-to-Q delay is much slower than the case when a normal pulse A is applied; this may cause a timing problem.

Fig. 4(a) shows an experiment setup to test the timing integrity of pulsed-latch circuits. We varied the number of latches and the value of  $C_w$ , which represents a wire capacitance, and determined failure or success; if data is not captured or the clock-to-Q delay exceeds  $1.1 \times$  of nominal value at any of the latches, we regard it as a failure. The shmoo plot of the result is shown in Fig. 4(b).

The load capacitance of a pulser consists of wire capacitance

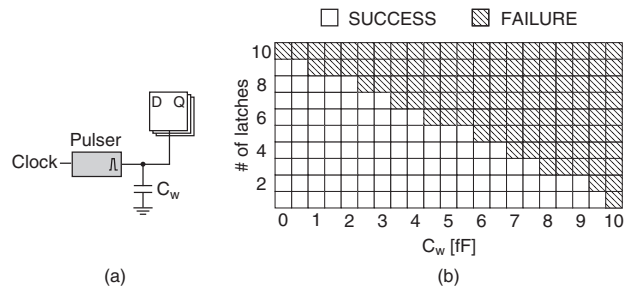


Fig. 4. (a) A setup to test timing integrity and (b) resulting shmoo plot.

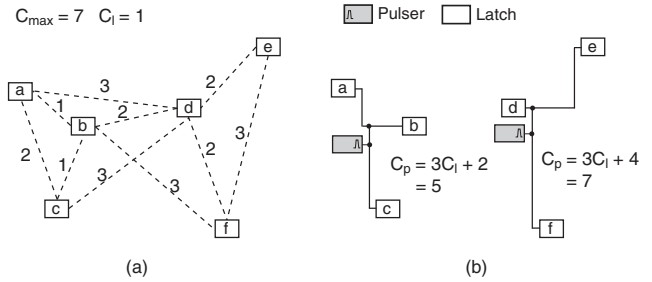


Fig. 5. (a) Latch locations with their wire capacitance between each other and (b) an example of solution.

$C_w$  and clock input capacitance of latches:

$$C_p = C_w + nC_l, \quad (1)$$

where  $n$  is the number of latches. The value of  $C_p$  at the boundary of success and failure of Fig. 4, denoted by  $C_{max}$ , turns out to be quite consistent, ranging from 9.9 fF to 11.1 fF in a 45-nm technology we used. This allows us to approximate  $C_{max}$  as a constant, for example 9.9 fF. As a result, for a given number of latches  $n$  that are driven by a single pulser, we can derive the value of  $C_w$  such that  $C_p \leq C_{max}$ , i.e. the timing integrity is guaranteed; the value then can be used to assess a particular placement of a pulser and its latches, which we call pulser group, by estimating the wirelength of their connections.

2) *Pulser Insertion*: To insert pulsers, we need to determine the grouping of latches so that the timing integrity of each pulser group is guaranteed. An arbitrary grouping of latches without considering their physical information is likely to yield a bad physical design, because latches in the same group will end up in a localized region, which severely constrains overall placement.

Pulser insertion, therefore, should be done after initial placement and latch locations are determined to comply with conventional placement objectives; the problem is to find a minimum number of pulsers such that each pulser group satisfy  $C_{max}$  of pulser. This is illustrated using an example in Fig. 5(a). The connection between each pair of latches is associated with wire capacitance, and the connections that cause  $C_{max}$  to be violated are dropped; there is no edge between  $a$  and  $e$  since the wire capacitance to connect them is larger than 5, which exceeds  $C_{max}$  when added to  $2C_l$ . Fig. 5(b) shows an example solution.

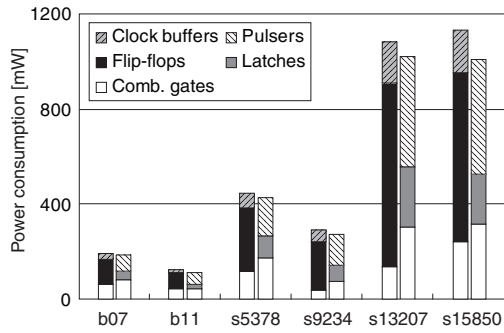


Fig. 6. Power consumption of flip-flop circuits (left bars) and pulsed-latch circuits (right bars).

3) *Placement*: Once pulser groups are identified, a whole design should be placed again, either in incremental fashion or as a completely new placement step. A simple heuristic method to rely on a conventional placement tool is to assign a higher net weight between each group of latches and their pulser [9]. A more systematic method is to design a new placement algorithm; the connection between pulser and latch [10] can be explicitly constrained by introducing extra barrier force into the conventional analytic placer.

### C. Benefits in Performance and Power

A simple migration of a flip-flop circuit to pulsed-latch version can benefit from improved performance; it is reported that the clock period is reduced by 5% due to the small sequencing overhead and is further reduced by 2.5% due to time borrowing [2].

Dynamic power is also reduced by using pulsed-latches; it is reported that mixture use of pulsed-latches allows a 20% saving in the dynamic power [4]. Fig. 6 illustrates the result of the migration to pure pulsed-latch version for some test circuits in 45-nm technology. Overall power consumption is reduced by 7.3% on average (minimum of 4.3% and maximum of 11.2%), even though it involves the inclusion of pulsers and delay buffers. It is assumed that leaf-stage clock buffers in flip-flop circuits are unnecessary in pulsed-latch counterparts due to the existence of pulsers. Notice the difference of power consumption between flip-flops and latches with pulsers. A standard D-type flip-flop consumes about  $1.6 \mu\text{W}$ , a latch consumes  $0.5 \mu\text{W}$ , and a pulser consumes  $7.2 \mu\text{W}$ ; when a single pulser drives 10 sequencing elements is reduced from  $16.0 \mu\text{W}$  to  $12.2 \mu\text{W}$ . The power consumption of combinational gates is increased in pulsed-latch circuits due to extra delay buffers.

The proportion of pulsers in total power consumption is significant in Fig. 6; this implies that there is a room to save more power if we design a pulser to consume less power, or to drive more number of latches, which, however, may constrain placement too much.

## III. OPTIMIZING PULSED-LATCH CIRCUITS

### A. Time Borrowing via Pulse Width Allocation

Even though a pulse is very short in pulsed-latches, a small amount of time borrowing is still possible. This possibility is

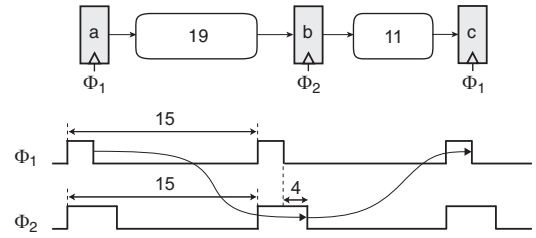


Fig. 7. Time borrowing from using different pulse widths.

ignored in ASIC design to simplify the timing model; pulsed-latches are approximated to trigger at rising edge (or falling edge) of clock.

If we use more than one pulse width in a design, the difference of pulse width between launching and capturing latches can be exploited, which is another form of time borrowing, to improve performance. This is illustrated in Fig. 7. The maximum combinational delay between latches  $a$  and  $b$  is 19 and that between  $b$  and  $c$  is 11. The pulse applied to  $b$  ( $\phi_2$ ) is wider by 4 than that applied to  $a$  and  $c$  ( $\phi_1$ ); the period of both pulse clocks are set to 15. In this setting, the block between  $a$  and  $b$  effectively borrows 4 time unit from the block between  $b$  and  $c$ , thereby working correctly even though its delay is larger than clock period. Note that clock period has to be set to 19 if  $\phi_1$  is applied to all three latches. A problem to assign pulse width to each latch is called pulse width allocation (PWA) [11].

The result of applying PWA to test circuits is shown in Fig. 8 (as circles). The clock period of initial pulsed-latch circuit ( $P_{ini}$ ) and minimum clock period ( $P_{min}$ ), which is obtained by clock skew scheduling [12] while assuming that arbitrary amount of skew can be realized, are also shown. It is clear that PWA alone cannot achieve clock period close to  $P_{min}$ , when three different pulse widths (130 ps, 190 ps, and 250 ps) are assumed to be available. The use of wider pulse width in PWA is necessarily limited, because increasing pulse width causes more risk of hold time violations.

### B. Combined PWA and Sequential Optimization

The limitation of PWA can be alleviated if PWA is used together with sequential optimization techniques such as clock skew scheduling (CSS) and retiming.

The results of combined PWA and CSS (PWCS for brevity) [9], in which skew is allowed up to 10% of  $P_{min}$ , and combined PWA and retiming (PWR for brevity) [13] are also shown in Fig. 8. In all circuits, either PWCS or PWR achieves clock period close to  $P_{min}$ . PWCS is not very effective in circuits b04 and b07; this is because of limited amount of skew and time borrowing while  $P_{min}$  is far from  $P_{ini}$  thus requires large extent of optimization. PWR, on the other hand, goes much further than PWCS in these circuits because retiming can always be performed as long as it can be applied, but at the cost of increased number of latches, 58% and 43%, respectively. The average number of latches from PWR is increased by 13%, which is much smaller than the increase

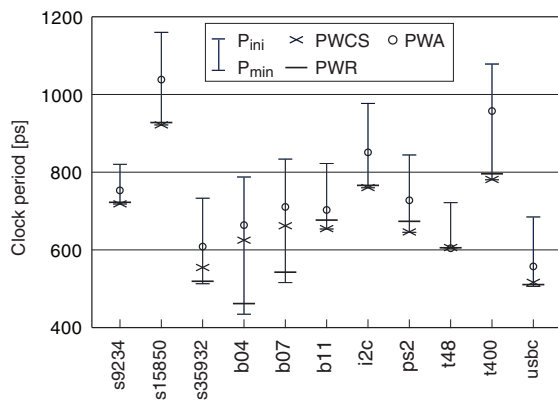


Fig. 8. Comparison of clock period obtained from PWA, combined PWA and CSS (PWCS), and combined PWA and retiming (PWR).

due to retiming alone.

When compared with the initial pulsed-latch circuits, the area, which is the sum of the areas of all the cells, of designs obtained from applying PWCS is increased by 2.1% on average; this is due to more pulsers (0.4%) and more delay buffers (1.7%). For designs obtained from PWR, the area is increased by 7.7% on average, which is due to more latches (2.6%), more pulsers (0.8%), and more delay buffers (4.3%).

### C. Clock Gating of Pulsed-Latch Circuits

To further reduce the power consumption of pulsed-latch circuits, we can consider clock gating, which has become a standard practice to reduce power consumption of a clock distribution network. A key problem in the conventional clock gating is to identify a group of flip-flops that can be gated at the same time (and as often as possible) and implement gating function for each group, which can be either specified by designers at architectural level, or automatically determined at RTL [14] or from a gate-level netlist [15].

Clock gating of pulsed-latch circuits can be implemented via pulsers, by feeding gating function to the enable pin of the pulsers, which shall be called pulser gating [16]. This implies a new problem, in which we identify a group of latches that can be driven by the same pulser (thus, they are placed nearby) as well as gated at the same time.

Fig. 9 shows a preliminary result of solving pulser gating problem given a gate-level netlist [16]: extract a gating condition of each latch (as a Boolean expression), perform initial placement to obtain latch locations, and identify groups of latches and insert pulsers, in which each pulser is enabled/disabled by the consensus of gating conditions of latches within its pulser group. Power saving is largely determined by the average gating probability, where average is taken over all latches. Notice the power reduction in pulsers, in particular, when gating probability is high.

## IV. SUMMARY

ASIC designs can benefit from adopting pulsed-latches in that they can be used in the standard ASIC design flow without major change while it still offers the opportunity

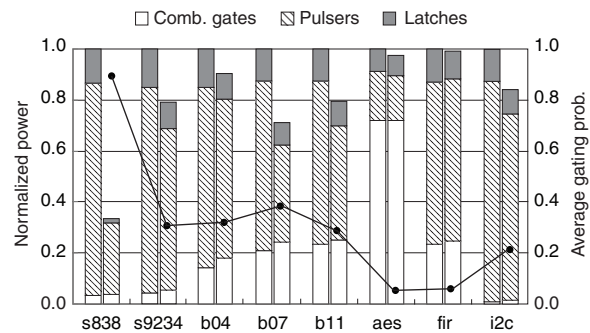


Fig. 9. Power consumption of initial pulsed-latch circuits (left bars) and that of circuits obtained by performing pulser gating (right bars).

of performance improvement and power saving. The key to success of adopting pulsed-latch circuits is to guarantee the timing integrity, which should be considered especially during pulser insertion and placement.

## REFERENCES

- [1] D. Chinnery and K. Keutzer, *Closing the Gap Between ASIC & Custom*. Kluwer Academic Publishers, 2002.
- [2] T. Baumann, D. Schmitt-Landsiedel, and C. Pacha, "Architectural assessment of design techniques to improve speed and robustness in embedded microprocessors," in *Proc. Design Automation Conf.*, July 2009, pp. 947–950.
- [3] R. S. Shelar, "An efficient clustering algorithm for low power clock tree synthesis," in *Proc. Int. Symp. on Physical Design*, Mar. 2007, pp. 181–188.
- [4] S. Shibatani and A. Li, "Pulse-latch approach reduces dynamic power," July 2006, EE Times.
- [5] N. Shenoy, R. Brayton, and A. Sangiovanni-Vincentelli, "Minimum padding to satisfy short path constraints," in *Proc. Int. Conf. on Computer-Aided Design*, Nov. 1993, pp. 156–161.
- [6] C. Lin and H. Zhou, "Clock skew scheduling with delay padding for prescribed skew domains," in *Proc. Asia South Pacific Design Automation Conf.*, Jan. 2007, pp. 541–546.
- [7] Y. Sun, J. Gong, and C. Chen, "Method and apparatus for fixing hold time violations in a circuit design," U.S. Patent 7278126 B2, Oct. 2007.
- [8] P. Kotecha, F. Musante, V. Pureswaran, L. Trevillyan, and P. Villarrubia, "Method of minimizing early-mode violations causing minimum impact to a chip design," U.S. Patent 2010/0042955 A1, Feb. 2010.
- [9] H. Lee, S. Paik, and Y. Shin, "Pulse width allocation and clock skew scheduling: optimizing sequential circuits based on pulsed latches," *IEEE Trans. on Computer-Aided Design*, vol. 29, no. 3, pp. 355–366, Mar. 2010.
- [10] Y. Chuang, S. Kim, Y. Shin, and Y. Chang, "Pulsed-latch-aware placement for timing-integrity optimization," in *Proc. Design Automation Conf.*, June 2010, pp. 280–285.
- [11] S. Paik, L. Yu, and Y. Shin, "Statistical time borrowing for pulsed-latch circuit designs," in *Proc. Asia South Pacific Design Automation Conf.*, Jan. 2010, pp. 675–680.
- [12] J. Fishburn, "Clock skew optimization," *IEEE Trans. on Computers*, vol. 39, no. 7, pp. 945–951, July 1990.
- [13] S. Lee, S. Paik, and Y. Shin, "Retiming and time borrowing: optimizing high-performance pulsed-latch-based-circuits," in *Proc. Int. Conf. on Computer-Aided Design*, Nov. 2009, pp. 375–380.
- [14] L. Benini and G. De Micheli, "Automatic synthesis of low-power gated-clock finite-state machines," *IEEE Trans. on Computer-Aided Design*, vol. 15, no. 6, pp. 630–643, June 1996.
- [15] E. Arbel, C. Eisner, and O. Rokhlenko, "Resurrecting infeasible clock-gating functions," in *Proc. Design Automation Conf.*, July 2009, pp. 160–165.
- [16] S. Kim, I. Han, S. Paik, and Y. Shin, "Pulser gating: A clock gating of pulsed-latch circuits," in *Proc. Asia South Pacific Design Automation Conf.*, Jan. 2011, to be published.