Transactions Briefs

High-Performance Energy-Efficient D-Flip-Flop Circuits

Uming Ko and Poras T. Balsara

***Abstract*—This paper investigates performance, power, and energy efficiency of several CMOS master–slave D-flip-flops (DFF's). To improve performance and energy efficiency, a push–pull DFF and a push–pull isolation DFF are proposed. Among the five DFF's compared, the proposed push–pull isolation circuit is found to be the fastest with the best energy efficiency. Effects of using a double-pass-transistor logic (DPL) circuit and tri-state push–pull driver are also studied. Last, metastability characteristics of the five DFF's are also analyzed.**

***Index Terms*—Circuit optimization, D-flip-flop, energy efficient, high performance.**

I. Introduction

D-flip-flops (DFF's) are one of the major functions in finite-state machines (FSM), which in turn form a critical part of control logic. It has been reported in [1] that the control logic of a microprocessor can utilize 20% of the processor's power. As more advanced architecture concepts, such as register renaming and out-of-order execution in a superscalar microprocessor [2], continue to prevail, the control logic is likely to become more complex, and its power dissipation is likely to grow beyond its current level. In addition, to boost processor clock frequency, modern processors typically adopt superpipelined execution [2], which also makes heavy use of DFF's. Enhancing DFF speed can either lead to a higher clock rate or allow more logic depths between two pipeline registers. In this paper, we first compare the area, speed, and power dissipation of existing DFF implementations with a conventional low-risk DFF [3], a low-area DFF, and a low-power DFF [4]. Then we propose two energy-efficient DFF designs: a push–pull DFF for performance and a push–pull isolation DFF (PPI-DFF) for performance and energy efficiency. These are then compared with the existing designs and demonstrated to be better for high-performance, energy-efficient applications. Discussion is then extended to the use of double-pass-transistor logic (DPL) for speed and tri-stated circuit for reducing short-circuit power dissipation. Last, characteristics of metastability of all the five DFF implementations are analyzed.

II. Design Techniques and Comparison for Energy Efficiency

In this section, we compare the existing and proposed DFF's for speed, power, and area. The flip-flop schematic diagrams are depicted in Fig. 1 and results of evaluation are tabulated in Table I. All the designs were individually optimized for speed by appropriately sizing the transistors. The evaluation was done using a 0.6-μm CMOS technology with a supply voltage of 3.3 V. The time-averaged power dissipation was characterized from SPICE simulation under the condition of 100-MHz clock frequency and maximum switching activity at the circuit nodes. The energy consumption was derived as a product of power and delay. A conventional negative edge-triggered DFF consisting of two-level sensitive latches or 16 MOSFET's is illustrated in Fig. 1(a) [3]. The speed of this conventional DFF is limited by two gate delays [one transmission gate and one inverter in Fig. 1(a), or 245 ps in Table 1] after the clock signal C transitions from logic 1 to 0. The advantage of this DFF design is that it involves minimum design risk. A common approach for reducing area overhead of the conventional DFF is to remove the two feedback transmission gates. This low-area DFF is depicted in Fig. 1(b). Although the strength of feedback inverters has been weakened to minimize short-circuit power dissipation due to voltage contention, this low-area DFF still consumes 18% more total power and is 42% slower (or has 76% more energy, Table I) compared to the conventional DFF.

One approach to optimize for power dissipation is to replace the inverter and transmission gate in the feedback path of Fig. 1(a) with a single tri-state inverter. This approach is referred to as a low-power DFF and was recently proposed by Gerosa *et al.* in [4]. Fig. 1(c) depicts this low-power DFF. The tri-state inverter eliminates short-circuit power dissipation from the feedback path and yields only 1% reduction in total power and 3% (Table I) slower speed when compared to the conventional DFF. Considering area and energy efficiency, the low-power DFF is comparable to the conventional DFF.

In order to improve performance of a conventional DFF, we propose addition of an inverter and transmission gate between the outputs of master and slave latches to accomplish a push–pull effect at the slave latch, i.e., input and output of the output inverter (which drives the signal Q directly) will be driven to opposite logic values during switching. This push–pull DFF is depicted in Fig. 1(d). This adds four MOSFET's, but reduces the clock-to-output (C-to-Q) delay from two gates in a conventional DFF to one gate. One method to reduce the transistor count is to use an nMOSFET for latch input. However, since the output of an nMOSFET can only reach a voltage level of $V_{\rm dd} - V_t$ when it is at logic 1, it results in increased power dissipation. Therefore, a full transmission gate is kept in the push–pull DFF. To offset the four added MOSFET's for a push–pull DFF, we propose the elimination of two transmission gates from the feedback paths, as shown in Fig. 1(d). Compared to the conventional DFF, this push–pull DFF is 31% faster, but has a 22% power overhead.

To optimize the proposed push–pull DFF for energy usage, we add two pMOSFET's to isolate the feedback path, as illustrated in Fig. 1(e). This PPI-DFF increases the transistor count to 18, but achieves a 16% reduction in total power dissipation and a speedup of 25% (Table I) relative to the previous push–pull DFF in Fig. 1(d). Compared to the conventional DFF, PPI-DFF improves speed by 56% at an expense of 6% more power. Energy efficiency of this PPI-DFF is enhanced by 45–122% when compared to the previous four DFF's. Further adding two nMOSFET's to the feedback path of PPI-DFF (to make its feedback path identical to that of the conventional DFF) increases the transistor count to 20 and the load on clock signals. This derivative increases the total power dissipation by 9% and slows down C-to-Q delay by 12% relative to the PPI-DFF. Applying a DPL input [Fig. 1(f)] to the PPI-DFF can result in a 20% reduction in the setup time. However, when D is at logic 1 and C switches from logic 1 to 0, a dc path exists (*INV2-P2-P1-C*), leading to a 60% power overhead. Another option is to use a tri-state inverter to replace the push–pull driver of PPI-DFF,

Manuscript received June 29, 1995; revised April 8, 1999.

U. Ko is with Texas Instruments, Inc., Dallas, TX 75266 USA.

P. T. Balsara is with the Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX 75083 USA.

Publisher Item Identifier S 1063-8210(00)00767-8.

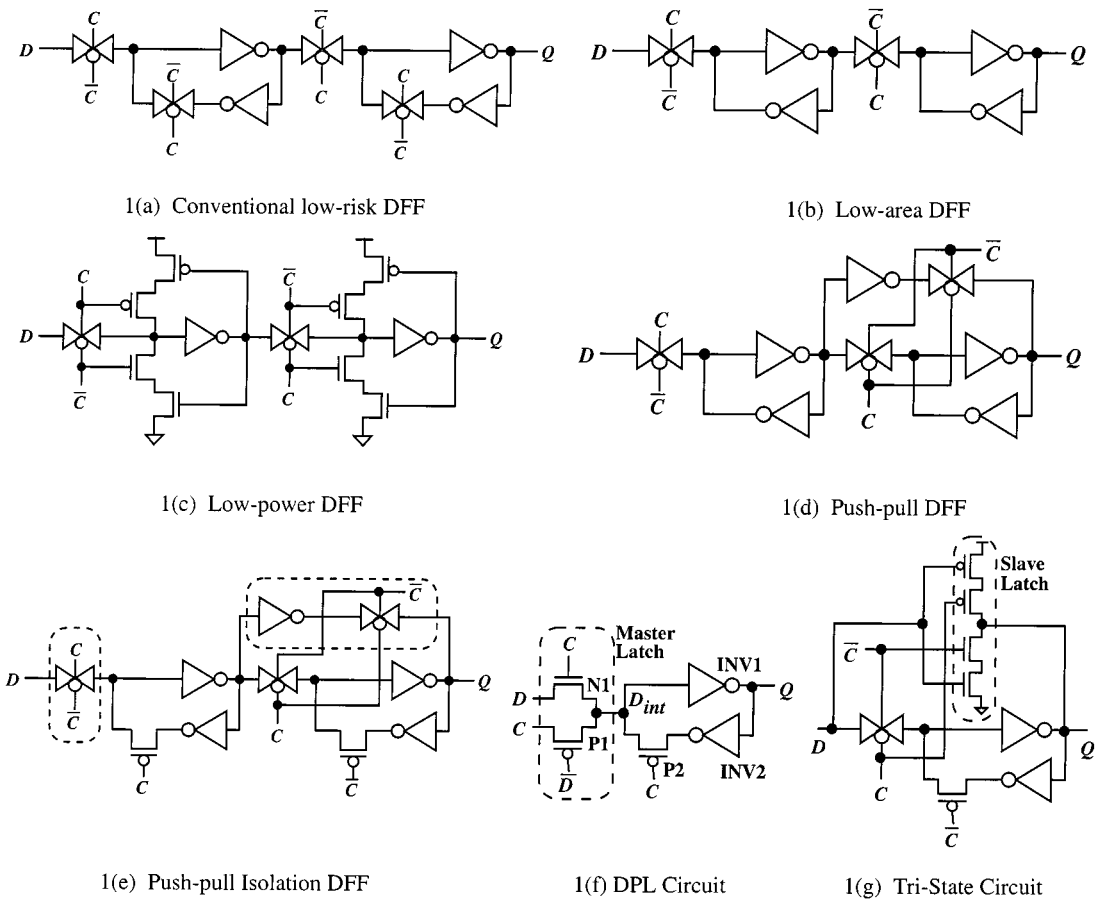

Fig. 1. Schematic diagrams of different DFF's.

TABLE I
COMPARISON OF VARIOUS D-FLIP-FLOPS

Parameters: 3.3V, 100MHz	Conventional	Low-area	Low-power	Push-pull	Push-pull Isolation	Unit
# of transistors	16	12	16	16	18	tr.
Total tr. width	45.8	40.2	46.8	43.0	48.0	μm
Power, avg.	122.9	146.9	121.7	152.6	131.4	μW
Percentage	94.0	112.0	93.0	116.0	100.0	%
Delay, C-to-Q	245.0	311.5	250.0	195.5	157.0	ps
Percentage	156.0	198.0	159.0	125.0	100.0	%
Energy	30.11	45.75	30.41	29.83	20.63	fJ
Percentage	146.0	222.0	147.0	145.0	100.0	%

TABLE II
METASTABILITY CHARACTERISTICS FOR lh TRANSITION

Nominal	Conventional	Low-area	Low-power	Push-pull	Push-pull Isolation	Unit
t_{meta}-t_{clk}	-0.454	-0.775	-0.464	-0.757	-0.429	ns
τ	56.030	99.341	65.694	86.653	71.682	ps
T_0	73.629	2.790	32.386	9.149	38.70	ns
$T_W(t_r$=1ns)	0.001	0.119	0.008	0.089	0.034	ps

as shown in Fig. 1(g). Though this approach reduces the short-circuit power of the push–pull driver, it weakens the drive strength due to stacked MOSFET's and is 10% less efficient in energy compared to the PPI-DFF. Hence, compared to all the DFF's and their derivations discussed above, the PPI-DFF turns out to be the most energy efficient.

TABLE III
METASTABILITY CHARACTERISTICS FOR hl TRANSITION

Nominal	Conventional	Low-area	Low-power	Push-pull	Push-pull Isolation	Unit
t_{meta}-t_{clk}	-0.360	-0.394	-0.330	-0.379	-0.312	ns
τ	81.861	109.57	82.153	99.339	94.878	ps
T_0	20.000	32.623	11.423	45.991	7.310	ns
$T_W(t_r$=1ns)	0.099	3.537	0.059	1.954	0.193	ps

The above flip-flops were also evaluated to observe the impact of supply voltage scaling on minimum data pulse width (see Fig. 2). Values of this minimum data pulse width were obtained by adding the DFF setup and hold times from SPICE simulations. At 1.5 V, the minimum data pulse width for the low-area and push–pull DFF's degrades to 3 ns due to voltage contention from the nonisolated feedback paths, while the other three DFF's maintain a pulse width of about 0.7 ns. This strongly suggests that the former should be avoided in low-voltage high-performance applications.

III. METASTABILITY ANALYSIS FOR DIFFERENT FLIP-FLOP CIRCUITS

Metastability is the phenomenon where a bistable element requires an indeterminate amount of time to generate a valid output, i.e., it can get stuck in a third illegal output state for some time. Metastability

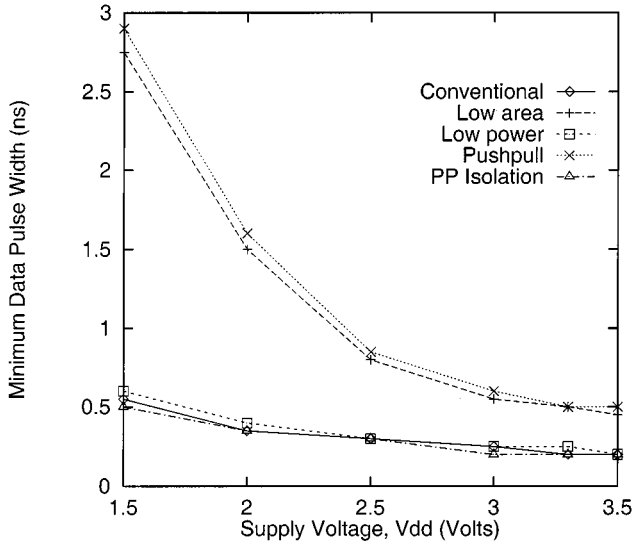

Fig. 2. Minimum data pulse width $(=t_{\rm setup} + t_{\rm hold})$ versus $V_{\rm dd}$.

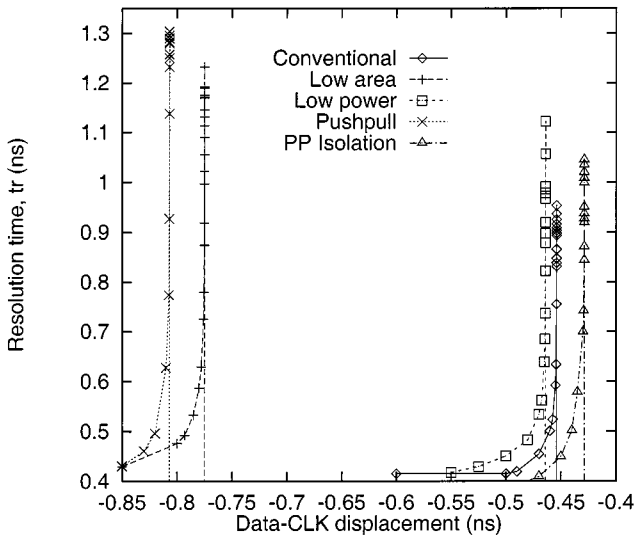

Fig. 3. t_r(lh) versus data-clock time displacement.

occurs in any systems where the data can change randomly with respect to the system clock [5]–[9] and can cause problems in a system because nonbinary signals are propagated to storage elements, hence leading to intermittent errors. Portmann *et al.* [7] recently discussed the equations for mean time between failures (MTBF) as a function of data transition frequency (f_D), clock frequency $(f_{\rm CLK})$, resolution time (t_r), resolution time constant (t), and the asymptotic width (T_0) of the metastability window(T_W) with zero resolution time. The MTBF and T_W equations are given in (1) and (2). In this section, we use these equations and the methodology in [7] to discuss metastability characteristics of the flip-flop circuits described earlier, under loaded conditions

$$\text{MTBF} = \frac{e^{(t_r/\tau)}}{f_D \cdot f_{\rm CLK} \cdot T_0} = \frac{1}{f_D \cdot f_{\rm CLK} \cdot T_W} \tag{1}$$

$$T_W = T_0 \cdot e^{(-t_r/\tau)} \tag{2}$$

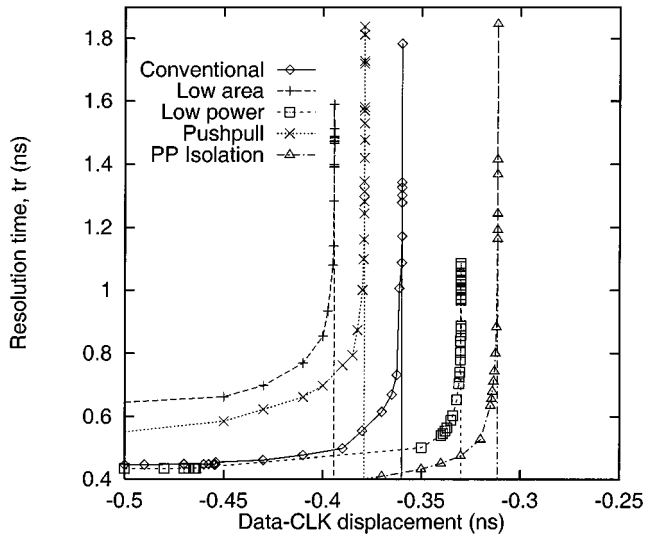

Fig. 4. t_r(hl) versus data-clock time displacement.

where

f_D	data transition frequency;
$f_{\rm CLK}$	clock frequency;
t_r	resolution time (including DFF clock-to-output time);
T_0	asymptotic width, the time window T_W normalized to $t_r = 0$;
T_W	metastability window;
τ	regenerative (resolution) time constant.

Equations (1) and (2) indicate that the higher the frequency for data transition(f_D) or clock $(f_{\rm CLK})$, the shorter the mean time between failure. Also, the larger the metastability window (T_W), the shorter the mean time between failure. Since both f_D and$f_{\rm CLK}$ are application specific and are primarily determined by systems, we focus the discussion on analyzing fundamental metastability characteristics of t_r, τ, T_0, and T_W. SPICE simulations were conducted on five flip-flop circuits under the conditions of 3.3-V supply voltage, 25°C, and a 0.6-μm nominal process model. An input slew rate of 0.5 ns is assumed for both the clock and data input signals (of flip-flops), which are both buffered with inverters. The flip-flop output has a fanout of three active standard loads. The standard load is defined as gate input capacitance of the minimum size inverter for the process technology under consideration.

Figs. 3 and 4 depict the output low-to-high (lh) and high-to-low (hl) resolution time(t_r) versus time displacement between the input data and clock signals, respectively. The resolution time is measured from falling edge of the clock input to the Q output signal (Fig. 1). The time displacement is measured from the point of change in the data input to the falling edge of the clock signal. Theoretically, as a transition of the input data signal is approaching the metastability point $(t_{\rm meta})$, t_r will grow exponentially and be infinite when it reaches $t_{\rm meta}$ when the input logic value is no longer latched by the flip-flop [7]. In SPICE simulation, this exponential relationship is observed for each flip-flop implementation and is plotted in Figs. 3 and 4. When time displacement between the input data and clock signals is at a distance from $t_{\rm meta}$ (e.g., 50 ps from $t_{\rm meta}$ or larger), t_r is primarily the propagation delay from clock to output Q, and PPI-DFF has the smallest t_r for either lh or hl transition. As this time displacement gets closer to $t_{\rm meta}$, t_r grows exponentially. In Figs. 3 and 4, a vertical line is used to highlight when each flip-flop reaches its $t_{\rm meta}$. For each flip-flop curve in these two figures, if we extend downwards to a point where it corresponds to a

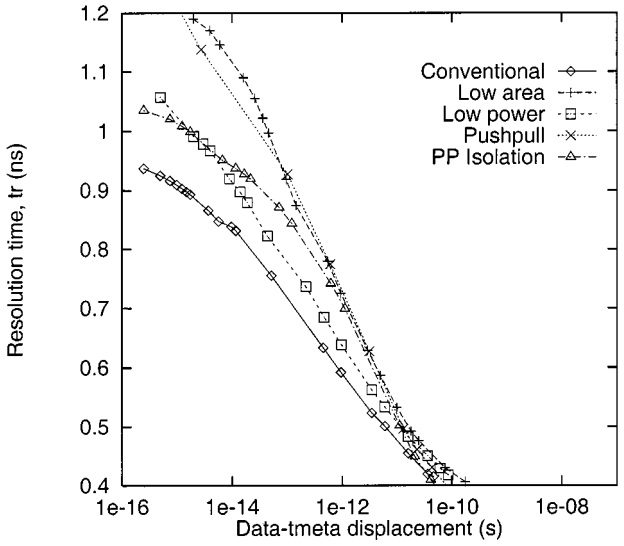

Fig. 5. $t_r(\mathrm{lh})$ versus data-t_{mcta} time displacement.

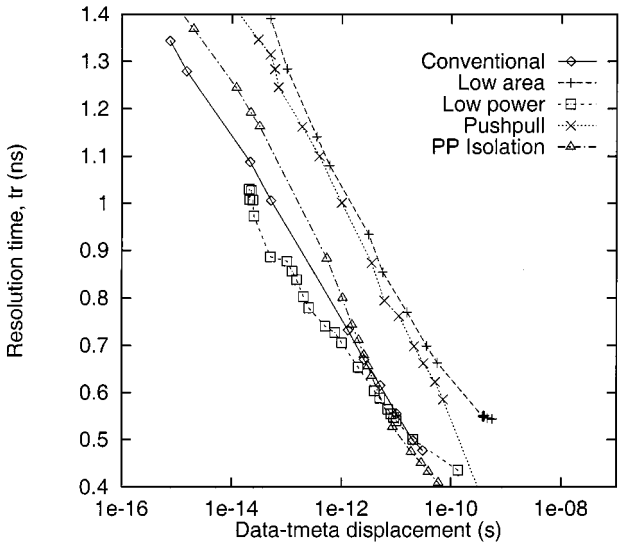

Fig. 6. $t_r(\mathrm{hl})$ versus data-t_{mcta} time displacement.

resolution time of zero, the distance between the X-axis intercept and t_{meta} is half of the asymptotic width, i.e., $T_0/2$ [7]. We should note that this zero resolution time is used to derive metastability windows mathematically and is meaningless from a practical point of view, as a finite propagation delay exists between the clock falling edge and a valid output signal [7].

From (2), if we plot resolution time t_r versus time displacement of input data transition and t_{meta} on a semilog scale (linear scale on Y-axis and log scale on X-axis), we should theoretically get a linear curve. The slope of this line is the resolution time constant τ, and the X intercept is the logarithm of $T_0/2$, or $\log(T_0/2)$. From the first-order approximation, τ is the inverse of the gain-bandwidth product and is related to the DFF's capability to resolve intermediate voltage level [7]. These semilog plots are illustrated in Figs. 5 and 6 for lh and hl flip-flop output transitions, respectively. To be conservative in our measurements, we use the largest slope for each curve in extracting τ, as the exponential curves in Figs. 3 and 4 decay slower with larger value of τ [7]. The value of τ and T_0 are summarized in Tables II and III for lh and hl transitions, respectively. Plugging τ and T_0 into (2), the metastability window T_W can be plotted in a semilog scale with respect to the flip-flop resolution time t_r (Figs. 7 and 8). Among the five flip-flop implementations compared under a given resolution time of 1.0 ns, both low-area and push–pull DFF's have a metastability window of one to two orders of magnitude larger than the other three flip-flops in either the lh or hl transition. This is due to the lack of feedback gating transistors in the master latch of both the low-area and push–pull DFF implementations, which causes voltage contention on the input nodes. Between these two, the push–pull DFF performs slightly better than the low-area implementation, as the push–pull effect helps to latch the logic value in the slave latch before input of the master latch switches. Compared to the conventional DFF, the metastability window of the PPI-DFF is 0.5–1.5 orders of magnitude larger (for $t_r = 1.0$ ns), since it lacks the nMOSFET in the feedback path and has an additional inverter load at output of its master latch. Compared to the conventional DFF, the low-power DFF has about the same order of metastability window, as both have nMOSFET and pMOSFET in the feedback path. The effects of voltage scaling on metastability parameters have been recently published by Portmann *et al.* in [7], and they suggest that the degradation of metastability parameters for reduced supply voltages is largely cancelled when the effects of longer on-chip delays are factored in.

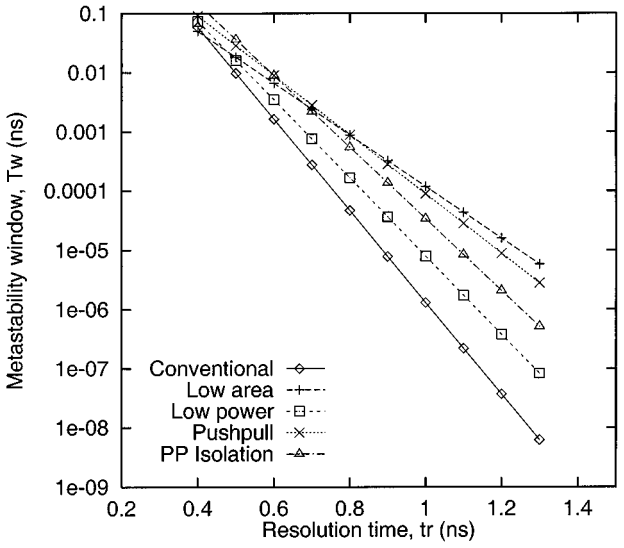

Fig. 7. Metastability window T_W versus $t_r(\mathrm{lh})$.

IV. Conclusion

In this paper, we have proposed DFF designs for energy-efficient applications and compared them with existing designs. Though the low-area DFF uses up to 33% fewer transistors, the internal voltage contention consumes up to 122% more energy than the rest of DFF's. Compared to a conventional DFF, a low-power and a push–pull DFF improve power dissipation by 1% and delay by 31%, respectively, but end up with a comparable energy efficiency. Our proposed PPI-DFF improves speed by 56% at the expense of only 6% more power, when compared to a conventional DFF. Energy efficiency of this PPI-DFF is 45–122% better than that of the other DFF's. Compared to the existing low-power DFF [4], our PPI-DFF uses 47% less energy. This may result in a 30–40% reduction in the overall energy consumption of control logic. On the issue of metastability, the lack of feedback transistors

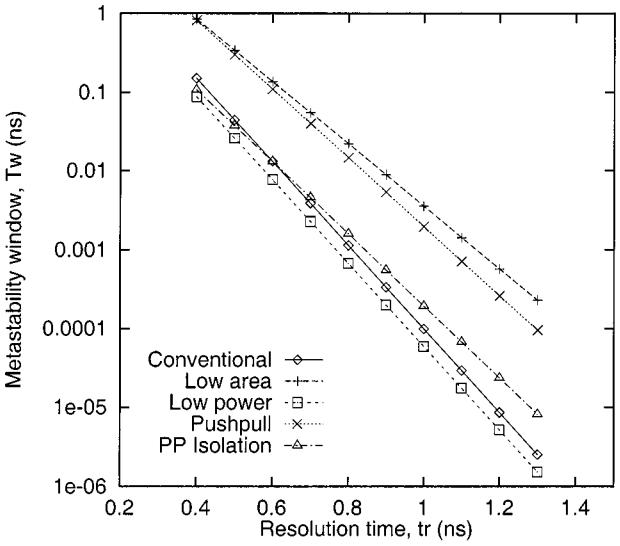

Fig. 8. Metastability window T_W versus $t_r(\mathrm{hl})$.

in both low-area and push–pull DFF's causes them to have a metastability window one to two orders of magnitude larger than the other three DFF's. In the applications where synchronizing circuits are required, the inclusion of both nMOSFET and pMOSFET devices in the feedback path of a DFF can minimize the metastability window and the resolution time constant, hence reducing their susceptibility to failure.

REFERENCES

[1] *Blue Lightning Technology Preview*, IBM, 1993.
[2] M. Johnson, *Superscalar Microprocessor Design*. Englewood Cliffs, NJ: Prentice-Hall, 1991.
[3] N. Weste and K. Eshraghian, *Principles of CMOS VLSI Design: A Systems Perspective*. Reading, MA: Addison-Wesley, 1993.
[4] G. Gerosa, S. Gary, C. Dietz, D. Pham, K. Hoover, J. Alvarez, H. Sanchez, P. Ippolito, T. Ngo, S. Litch, J. Eno, J. Golab, N. Vanderschaaf, and J. Kathle, "2.2 W, 80 MHz superscalar RISC processor," *IEEE J. Solid-State Circuits*, vol. 29, pp. 1440–1454, Dec. 1994.
[5] T. J. Chaney and C. E. Molnar, "Anomalous behavior of synchronizer and arbiter circuits," *IEEE Trans. Comput.*, vol. C-22, pp. 421–422, Apr. 1973.
[6] H. J. M. Veendrick, "The behavior of flip-flops used as synchronizers and prediction of their failure rate," *IEEE J. Solid-State Circuits*, vol. SC-15, pp. 169–176, Apr. 1980.
[7] C. L. Portmann and T. H. Y. Meng, "Metastability in CMOS library elements in reduced supply and technology scaled applications," *IEEE J. Solid-State Circuits*, vol. 30, pp. 39–46, Jan. 1995.
[8] F. Rosenberger, C. Molnar, and R. Dutton, "Comments, with reply, on 'Metastability of latch/flip-flop'," *IEEE J. Solid-State Circuits*, vol. 27, pp. 128–130, Jan. 1992.
[9] J. Hohl, W. Larsen, and L. Schooley, "Prediction of error probabilities for integrated digital synchronizers," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 236–244, Apr. 1984.

Design of VHDL-Based Totally Self-Checking Finite-State Machine and Data-Path Descriptions

C. Bolchini, R. Montandon, F. Salice, and D. Sciuto

***Abstract*—This paper presents a complete methodology to design a totally self-checking (TSC) sequential system based on the generic architecture of finite-state machine and data path (FSMD), such as the one deriving from VHDL specifications. The control part of the system is designed to be self-checking by adopting a state assignment providing a constant Hamming distance between each pair of binary codes. The design of the data path is based on both classical methodologies (e.g., parity, Berger code) and *ad hoc* strategies (e.g., multiplexer cycle) suited for the specific circuit structure. Self-checking properties and costs are evaluated on a set of benchmark FSM's and on a number of VHDL circuits.**

***Index Terms*—Checker circuits, detecting codes, finite-state machine and data path (FSMD), state encoding, totally self-checking, VHDL.**

I. INTRODUCTION

Self-checking (SC) devices are increasingly becoming a suitable approach to the design of complex systems, to cope with the growing difficulty of on-line and off-line testing. The design of self-checking systems allows the detection of transient and permanent faults, providing the identification of an erroneous behavior as soon as it is observable. An SC circuit consists of a functional circuit generating encoded data and a checker, which verifies that no fault has occurred (in the circuit or in the checker itself) by controlling data that belong to the adopted code. When combinational circuits are considered, an SC design encodes the circuit outputs; when sequential circuits are considered, the encoding applies to outputs and state [1].

This paper considers the design process for SC circuits, oriented toward control-dominated systems based on finite-state machines and data-path (FSMD) descriptions, typical of VLSI devices specified in VHDL. The goal is to present a complete methodology for the design of SC systems specified in VHDL. The FSM state assignment constitutes a novelty. The main difference from other approaches ([1]–[5]), is that the assigned state code does not constitute a codeword itself, but the sequence *present state code—next state code* is the element of the defined code. When the data path is combinational, without specific arithmetic units (dealt with in [6] for instance), the totally self-checking (TSC) methodology applies the traditional encoding of the data, Berger, or parity code. When the data path is sequential, a specific innovative approach has been defined, based on the identified structures composing the network.

The methodology is integrated in the design process of VLSI system, described in VHDL [7]. This formalism (constituting the entry point of the design methodology together with the KISS format for MCNC benchmarks) allows the integration of the proposed approach in a standard industrial design flow. One of the main advantages of the proposed methodology is that it is, to the authors' knowledge, the only complete approach for the realization of TSC complex systems starting from a VHDL description. The elements of the TSC design methodology based on the aforementioned aspects are the following ones.

Manuscript received November 5, 1997; revised November 10, 1998.

The authors are with the Dipartimento di Elettronica e Informazione, Politecnico di Milano, Milano 32–20133, Italy (e-mail: bolchini@elet.polimi.it; montandon@elet.polimi.it; salice@elet.polimi.it; sciuto@elet.polimi.it).

Publisher Item Identifier S 1063-8210(00)00764-2.