The Effect of the System Specification on the Optimal Selection of Clocked Storage Elements

Christophe Giacomotto, *Student Member, IEEE*, Nikola Nedovic, *Member, IEEE*, and Vojin G. Oklobdzija, *Fellow, IEEE*

***Abstract*—This paper represents a departure from the conventional methods of design and analysis of clocked storage elements that rely on minimizing a fixed energy–delay metric. Instead it establishes a systematic comparison in the energy–delay design space based on the parameters of the surrounding blocks. We define the composite energy-efficient characteristic over all storage element topologies and identify the most efficient storage element depending on its position on the composite characteristic relative to other topologies within a pipeline stage. Thus, we show that an optimal design could use a mixed variety of clocked storage elements (CSEs) depending on their placement in the pipeline and critical path. Since a well-designed system has hardware intensities balanced for a given cycle, a CSE choice will be made depending on the pipeline and path intensities. We show that a meaningful comparison can be carried out only by acknowledging that the optimal design and choice of the clocked storage elements depends heavily on the application, and by analyzing the energy and delay of the clocked storage elements in context of this application. The analysis in the energy–delay space allows us to understand some intuitive design choices in a quantitative way and to identify the optimal storage element topologies for an arbitrary system specification.**

***Index Terms*—Clocked storage elements, energy delay optimization, flip-flops, VLSI, power consumption, registers, circuit tuning, circuit optimization, delay effects, circuit analysis, integrated circuit design, CMOS digital integrated circuits, energy measurement.**

I. Introduction

THE performance scaling over past two decades was enabled by the wide use of the pipelining concept in the processor architecture. The principle of the pipelining technique is to divide the logic needed to perform an operation (e.g., instruction execution, floating point multiplication, etc.) into N stages, and to perform the operation in N clock cycles. Each stage performs the portion of the logic operation based on the output of the previous stage(s), computed in the previous cycle(s). The computed results are stored for the next cycle, or, depending on the strategy, for a portion of the cycle in *clocked storage elements* (flip-flops or latches). In this way, N consecutive data sets evaluate at the same time, each in a separate pipeline stage. As a result, the pipeline throughput—defined as the rate of producing consecutive results—is increased by a factor slightly less than N. At the same time, the latency of the operation, or the delay between applying the input data to the pipeline and producing the results, is increased due to the delay of the storage elements introduced in the signal path.

As the performance targets in the processor market cannot be accommodated by the technology scaling only, the designers routinely resort to reducing the number of logic gates per pipeline stage, deepening the pipeline, and increasing the clock frequency. In such designs, the clocked storage elements (CSEs) used for pipeline synchronization occupy an increasingly large portion of the clock cycle and total power. Since the CSEs are not performing any useful logic function, this method quickly results in diminishing returns in terms of performance. Hence, this trend provides a strong motivation for considerable research interest in the performance and power consumption of the CSEs [1]–[8]. In practice, the design direction of increasing the number of pipeline stages and reducing the numbers of gates per stage has several fundamental problems. As shown in [9], deep pipelines are operating at a power-inefficient design point and the CSEs are responsible for most of the overheads in both energy and delay. However, in a typical high-performance design, the computer architects tend to choose the pipeline depth that promises the highest performance, or merely the highest achievable frequency in a given technology based on ad hoc design rules. If needed, the processor power is usually addressed only at a later stage of the design by scaling voltage or frequency down until the design falls below the power budget limit. It was shown [9] that this approach is drastically suboptimal, i.e., that other designs exist that deliver the same performance (not necessarily the same clock frequency) at a much lower power, or much higher performance at the same power.

In addition, the interconnects performance improvements has been consistently smaller than that of the transistor over the last few years. Because of this interconnect inefficiency, the energy needed for clock generation, distribution, and CSEs becomes an increasingly significant factor in the total energy break–up, making the circuit design of the clocking subsystem a decisive factor of the overall system performance [9]. Yet, there does not exist a basic understanding of the optimum CSE design point as a function of the particular logic design, clock frequency, and underlying technology. Typically, the CSE topologies are compared either by the smallest possible delay, or preferably, by minimizing some energy–delay metrics,

Manuscript received January 1, 2006; revised February 27, 2007. This work was supported by Semiconductor Research Corporation grants and Fujitsu Ltd.

C. Giacomotto and V.G. Oklobdzija are with ACSEL Laboratory, Department of Electrical and Computer Engineering, University of California, Davis, CA 95616 USA, and Integration Corp., Berkeley, CA 94708 USA, respectively (e-mail: christophe@acsel-lab.com; vojin@acsel-lab.com).

N. Nedovic is with Fujitsu Laboratories of America, Sunnyvale, CA 94085 USA (e-mail: nikola.nedovic@us.fujitsu.com).

Digital Object Identifier 10.1109/JSSC.2007.896516

such as energy–delay product. Once the "best" candidates are identified, the CSE may be used at the minimum metric point, or further design of the CSE may involve transistor-size tuning to minimize energy while achieving specified delay at given fixed input size and output load. As indicated by Zyuban *et al.* [10], the CSEs should be optimized for the metric that maximizes the overall performance depending on the energy and delay break-up between the CSEs and the associated logic block within a pipeline stage. Since the optimum break-up conditions are unknown in advance, we propose to extend this work to address the most important question in the design of the clocked storage elements: given the application and the set of candidate storage elements, what is the best choice of the CSE topology and what is its optimal design point in terms of transistor sizing? In subsequent works, Zyuban [7] proposes comparing the CSEs based on energy efficient characteristics with fixed input size and output load which is a significant improvement versus metric-based comparisons. However, this work does not capture the effect of the loading conditions on the CSE comparison which is necessary for determining the Energy and Delay break-up between CSEs and logic. Heo and Asanovic [11] propose a CSE comparison with varying loads; however, they do not proceed to study the effect of the pipeline stage and cycle time specification on the optimum CSE choice. Dao *et al.* [12] study the dependence of the optimum sizing of the pipeline stages on the load interface, but do not extend their work on the analysis and design choice of the CSEs within a pipeline stage. This paper presents a comparison and analysis of the CSEs based on their energy–delay characteristics and a particular application. The notion of CSE performance is extended by using the composite energy–delay characteristics over an entire collection of CSEs [7] and by formulating the natural target application of each individual CSE. A quantitative method for optimal cycle time break–up is defined on the system level and based on practical environment and system parameters constraints, that is, the combined effect of the varying delay target for the CSE and varying CSE load due to the changing operating point of the logic block with the target cycle time must be accounted for to achieve a meaningful quantitative analysis.

Section II presents the new comparison methodology by showing how the analysis of a single CSE needs to be carried out to be compared with other CSEs. In Section III, we present a selection of widely used CSEs in the industry, and also in this section we explore how various key design choices can be made early on in the design process for certain CSE topologies. Section IV shows a comparative analysis of CSEs in a pipeline stage under two different system conditions. These two cases point out how CSE selection can be affected by the system. Additionally, we show quantitatively how a classic energy–delay product (EDP) comparison relates to this work.

II. Comparison Methodology

In this work, a new CSE comparison methodology is introduced: Instead of using isolated test bench conditions for the CSEs compared, we propose to quantitatively evaluate the performance of each particular CSE topology within a given pipeline stage. The procedure to achieve this is explained as follows.

1. Extract the energy characteristic of a single CSE with varying internal transistor sizes but with fixed input and output loads.
2. Reproduce this characteristic for a range of input and output conditions. This set of characteristics represents all the necessary data to evaluate the energy–delay performance of that particular CSE. (Note that some comparative work can be executed at this stage for clearly less efficient designs as shown in Section III.)
3. Extract the energy efficient characteristic of the pipeline stage logic block of interest. To simplify the comparison procedure, the minimum energy envelope of energy–delay curves can be used.
4. Combine the data of each CSE characteristic with each point of the logic characteristic to create a single pipeline stage characteristic with its delay being the clock cycle.
5. Reproduce steps 1 to 4 for each CSE analyzed and plot all the pipeline stage characteristics together to create the composite characteristics of the candidate CSEs for that pipeline stage.

By choosing the clock frequency target or the energy consumption target, this technique would reveal which CSE topology is best, which internal transistor sizing is best for that topology, what is the optimum interface load between the CSE and the logic, and, consequently, what should be the electrical effort on the logic. Effectively, this method matches the hardware intensities [10] of the pipeline register and logic block, which in return leads to a meaningful CSE comparison.

A. Energy-Efficient Characteristic Extraction

In order to understand the energy–delay tradeoff of a CSE, we observe its *energy-efficient characteristic*, or *dominant characteristic* (Fig. 1), which consists of all points in the energy–delay space that yield the smallest energy of all points with same delay, or equivalently, all points that yield the smallest delay of all points with the same energy for a fixed input size and fixed output load [10]. Energy–delay characteristics can be obtained by varying technology, circuit, and architectural parameters such as threshold voltage, transistor sizes, and supply voltage. In this paper, we will use only the transistor sizing as the independent parameter for the energy–delay tradeoff. In Fig. 1, each UltraSPARC flip-flop design (USPARC, [13]) evaluated is represented by a point with its delay being the minimum D-to-Q delay at the optimum setup time and its energy being the average energy of the CSE at 25% data activity. The simulation details and assumptions are presented in the Appendix. The points on the steep part of the energy-efficient characteristic, labeled "high energy sensitivity region", are obtained using larger and more aggressive transistor sizing. Similarly, the points in the flat part of the energy–delay (ED) characteristic, labeled "high delay sensitivity region", are obtained using smaller transistor sizing configurations. Potentially, any one of the energy-efficient design points can be the optimum point since each one achieves different delay and energy results for this input and output configuration. Hence, the minimum

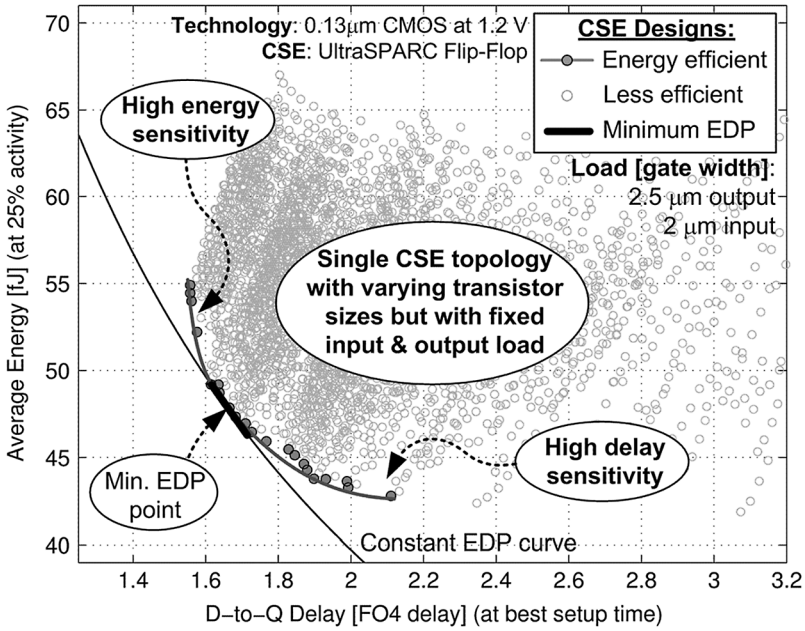

Fig. 1. Energy-efficient characteristic for a single CSE topology with fixed input size and fixed output load.

EDP design may not be the particular sizing solution that is representative of that CSE for comparison purposes [1].

B. Set of Energy-Efficient Characteristic Under Various Input and Output Loads

Previous CSE topology comparisons based on a simple metric such as ED^XP—where X is a constant and if $X = 1$ the metric is then EDP—or even energy-efficient characteristics assume similar gain and load conditions but show widely different energy and delay performance results. However, because a CSE is a small structure with usually about 15 to 20 transistors, the ED performance is greatly dependent on both input and output loading conditions [11]. So, in order to make use of the CSE in the design of a pipeline stage, the energy-efficient characteristics should be generated for varying input sizes and output loads. Furthermore, within a pipeline block, the optimum choice of a logic stage gain and load is dependent on the other logic stages' gain and load because it directly determines the ED performance of that logic stage [12]. Since comparing design points with different energy and delay performance implies different gain and load conditions for each point, comparing CSEs under same gain and load can be misleading. When the energy characteristics are produced for a range of input and output load as shown in Fig. 2, a complete data set of the potentially good designs is obtained. It becomes clear from Fig. 2 that the usage of any metric is impractical since all the characteristics cover a wide range of delay and energy targets.

C. Energy-Efficient Characteristic of the Logic Block

To thoroughly optimize the pipeline stage, the type of data produced in Fig. 2 must also be provided for the logic. However, because the objective of this work is to compare CSEs, we limit the logic block analysis to a single output load and the logic is optimized for minimum energy consumption [12]. The resulting information is an envelope of logic block energy–delay curves shown as the energy-minimized points in Fig. 3. To give an idea of the energy range of this logic block, the results of the logical effort design strategy [14] are also shown in gray. Along the energy-minimized envelope, the maximum input capacitance of the logic block varies. The logic block example for this work is a 32-bit Kogge–Stone adder (KSA, [15]). For the purpose of comparing CSEs, it is necessary to have an understanding of the ED behavior versus input/output load of the logic attached to the CSEs. To be able to carry out a meaningful comparison, the logic ED characteristic can be a rough estimate.

D. Pipeline Stage Energy–Delay Characteristic for a Particular CSE Topology

The objective is to combine the information from a particular CSE topology (Fig. 2) and a logic block of interest (Fig. 3) to get the ED characteristic of the pipeline stage for one CSE topology as shown in Fig. 4. Each design point for the adder, from the minimum energy envelope shown in Fig. 3, is combined with the CSE characteristic at the respective interface load. The interface load is the CSE output load which, in the case of the Fig. 4 pipeline stage, is also the adder input load. The stage delay is simply the addition of the CSE D-to-Q delay and the logic critical path delay; however, the energy evaluation is more complex. Because logic blocks such as adders are never perfectly regular, each bit of the adder has a different input capacitance. If exactly the same CSE is used for every bit, including the critical path bit(s), a large amount of energy would be spent by the registers unnecessarily. Theoretically, each CSE of the pipeline register should be tailored for each bit of the adder. In high-end pipelined processors, the registers usually contain a maximum of 8–10 different sizes. To reduce the complexity in this work, we assume the registers contain only a maximum of 2–3 different sizes of the same CSE topology. The pipeline register energy quantification can be done by directly looking up the CSE design which achieves the same delay target as the critical path in Fig. 2, but

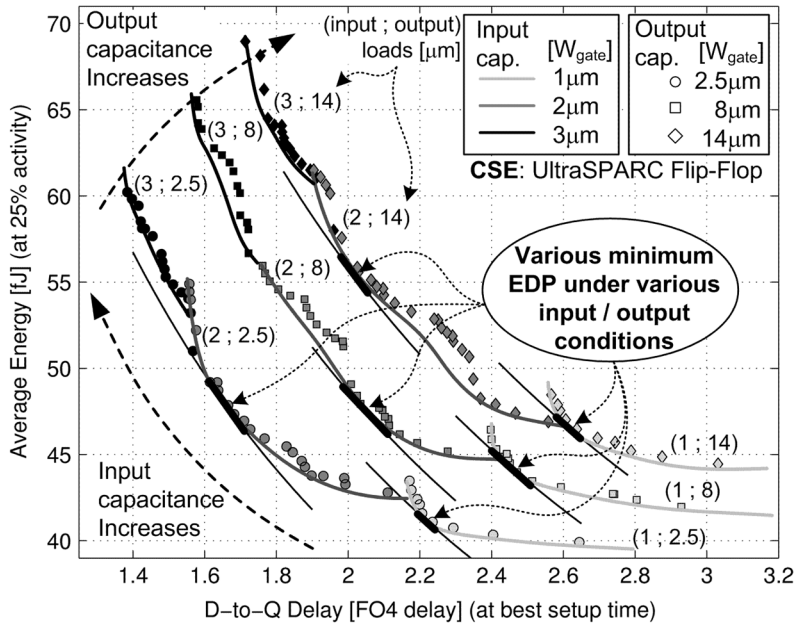

Fig. 2. Set of energy-efficient characteristic for a single CSE topology under various input and output loads.

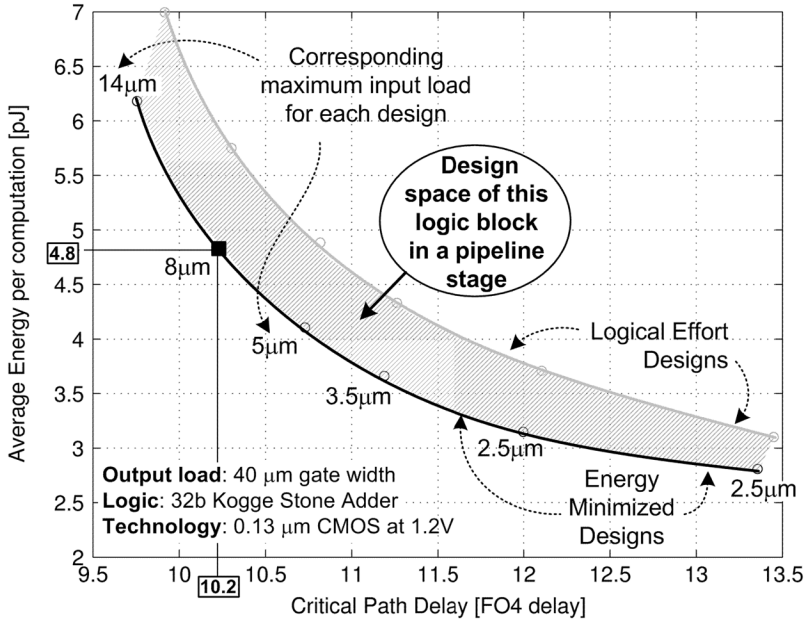

Fig. 3. Energy–delay characteristics of a Kogge–Stone adder.

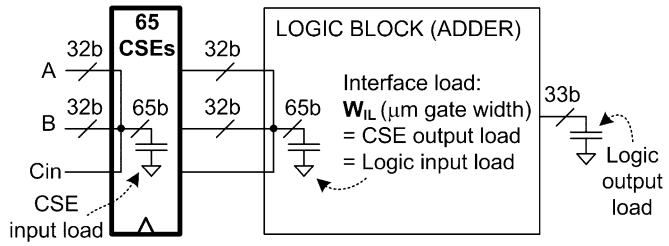

Fig. 4. Pipeline stage diagram.

on the ED characteristic with the appropriate interface load. For example, to combine the KSA design at the 8-μm interface load (10.2 FO4, 4.8 pJ point in Fig. 3) with the CSE shown in Fig. 2 with a 3-μm maximum input (all the design points shown by squares), the delay of each energy-efficient CSE design of the 8-μm interface load characteristic is simply offset by 10.2 FO4. However, for the energy, 42 inputs of the KSA adder have a load between 8 μm and 5 μm, 18 between 5 μm and 3.5 μm, and five below 3.5 μm. Then the register contains, respectively, only 42 CSEs from the 8-μm CSE characteristic, 18 from the 5-μm characteristic, and five from the 3.5-μm characteristic. The designs chosen from the 5-μm and 3.5-μm characteristics (not shown in Fig. 2 for clarity) must achieve at least the same delay as the 8-μm design but yield significantly lower energy consumption. For this reason, the logic input load distribution can skew the

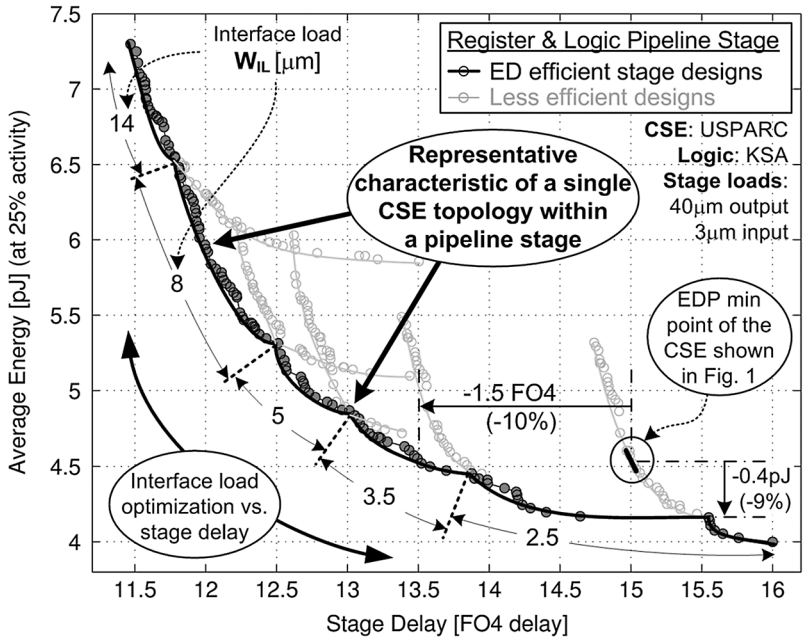

Fig. 5. Pipeline-stage energy–delay characteristic for a particular CSE topology with varying interface load.

energy results depending on the logic topology. The combination of this KSA logic design point with the 8-μm CSE energy characteristic results in one characteristic of the pipeline stage with a fixed interface load ($W_{\mathrm{IL}} = 8$ μm, shown in Fig. 5).

Once each logic block design point is paired with the appropriate CSE energy–delay characteristics, only some of the resulting designs are actually of interest, as shown by the bold points in Fig. 5. In fact, along this envelope of characteristics, the interface load is changing and effectively being optimized for the clock frequency or energy budget of choice. This happens because only a subset of the minimum energy designs are actually efficient for a specific interface load, as shown in black versus gray in Fig. 5. This black envelope represents accurately what a particular CSE topology can do in terms of energy–delay performance for a particular logic block within a pipeline stage. For example, with a stage delay target of 15 FO4, if we limit our design choices to the USPARC flip-flop for the CSE and KSA for the logic, the results in Fig. 5 indicate the following: the adder input load should be minimum (2.5 μm), the CSE should be minimum size, the adder delay should be reduced to meet the delay target while keeping its 2.5-μm input load and the energy penalty for doing this is the best compromise. If we assume the optimum interface load is already known, this method saves 9% of the total energy versus the classic EDP minimum design choice for this CSE structure. Reciprocally, for an energy target of 4.5 pJ, the clock cycle can be reduced by 10% by increasing the interface load to 3.5 μm and choosing the CSE sizing that meets the energy target. If the CSE topology choice is extended, not only can the best sizing for a CSE be chosen, but the best topology can also be chosen. So, by using this method for several representative CSEs, a set of final CSE characteristics can be produced and compared fairly on a single energy–delay plot.

In this section, we have shown how to obtain a meaningful representation of the energy–delay performance of a single CSE which can be compared to other CSEs. This CSE performance

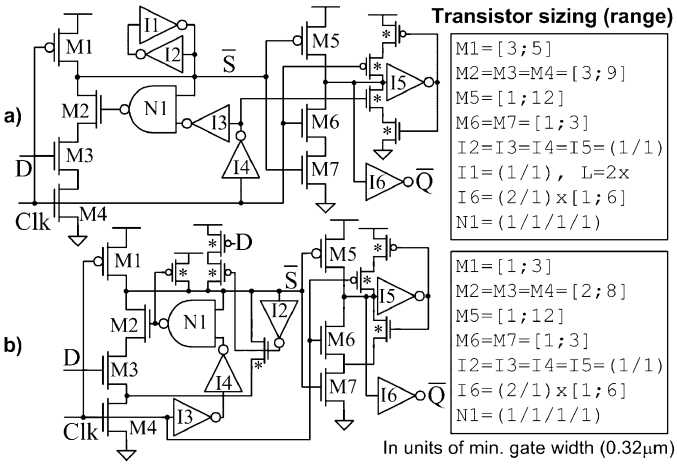

Fig. 6. Circuit and transistor sizing of (a) semi dynamic flip-flop (SDFF), and (b) UltraSPARC implementation of the SDFF (USPARC).

representation is bound to a particular pipeline stage and a fair comparison can only be carried out in the case studies presented in Section IV. However, a stand-alone analysis of a particular CSE can be performed as shown in Fig. 2. In the next section, we show that such analysis can be sufficient to either discard a topology or indicate quantitatively key design choices.

III. Energy-Efficient Storage Elements, Design and Quantitative Analysis

In this section, we describe several representative clocked storage elements commonly used, as well as several recently published structures. Single-ended CSEs fall into three major groups: dynamic structures, explicitly pulsed latches, and master–slave latches [2]. The advantage of our quantitative

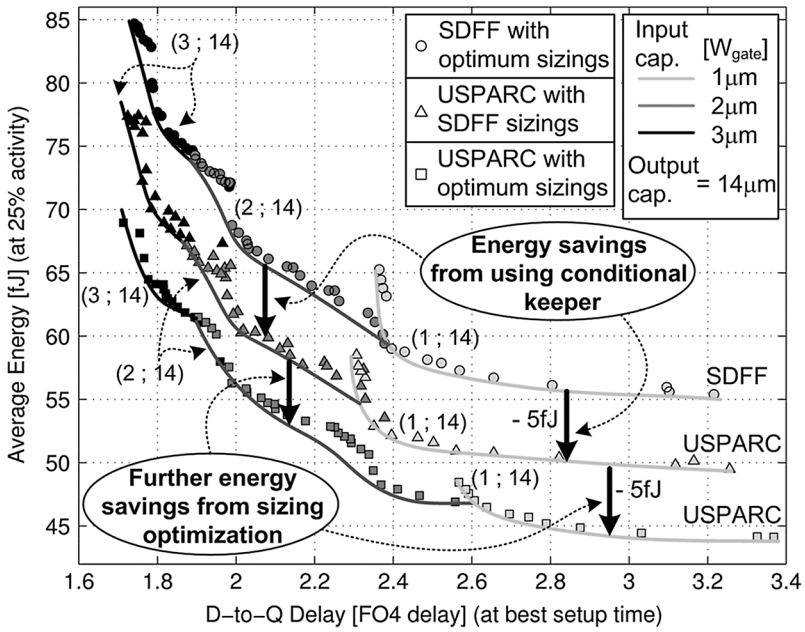

Fig. 7. SDFF versus USPARC topologies: Quantitative analysis.

evaluation, as shown in Figs. 2 and 5, is the design space understanding of various circuit features from an ED performance perspective.

A. Semi-Dynamic Flip-Flop and Its UltraSPARC Implementation

The semi-dynamic flip-flop (SDFF, [16], Fig. 6(a)) is the concept flip-flop used in Sun UltraSPARC-III microprocessor. Its operation is based on generating the short timing window after the rising edge of the clock, determined by the delay through the two inverters I4 and I3, and the NAND gate N1. The first stage of the SDFF is a dynamic logic-like circuit that keeps the voltage of the node $\overline{S}$ at the level evaluated in the transparency window until the clock *Clk* switches low. This allows the faster operation and use of a simpler TSPC-style [17] dynamic-to-static latch in the second stage. Its dynamic circuit design yields high speed and a limited logic embedding capability, highly desirable in high-performance applications. The small delay of the SDFF is paid for by its large energy, mainly consumed for switching the clock pulse generator and high-activity highly loaded dynamic node $\overline{S}$.

The actual SDFF implementation (USPARC, [13], Fig. 6(b)) was redesigned to reduce the impact of soft error hazards. The modifications mainly consist in making the dynamic node keeper conditional, which has the added benefit of having it not fighting the first stage. This feature decreases the energy consumption by 5 fJ if we keep the same transistor sizing as shown in Fig. 7. However, by optimizing the transistor widths further (Fig. 6(b)), another 5 fJ can be saved (Fig. 7). Additionally, the 2x input characteristic becomes more energy efficient, which is the reason why the 1x input characteristic seems less efficient for the USPARC than the SDFF. This happens because the first stage does not need to fight the keeper in the USPARC, hence, more transistor width can be assigned to the evaluation path rather than the pre–charge path while keeping the same capacitive load level on $\overline{S}$. This comparison on the same plot was possible because of the similarities between the two structures and mainly because the impact of the topology differences happens in terms of energy only.

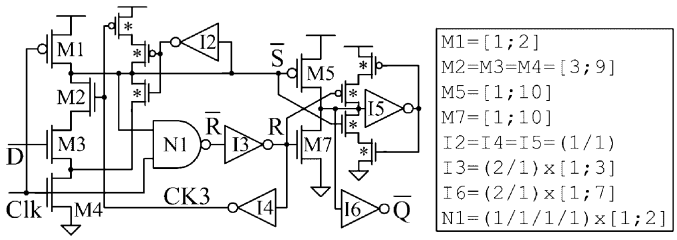

Fig. 8. Circuit and transistor sizing of the implicitly pulsed flip-flop (IPP).

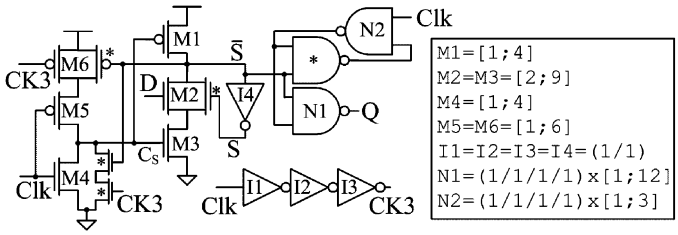

Fig. 9. Circuit and transistor sizing of the skew-tolerant flip-flop (STFFSE).

B. Other Dynamic Structures

Two other dynamic flip-flop variants are considered in this work: the implicitly pulsed flip-flop with push-pull latch (IPP, [5], Fig. 8) which presents low energy characteristics, and the single-ended skew-tolerant flip-flop (STFFSE, [18], Fig. 9) which presents high-speed properties.

The IPP explicitly generates both set $\overline{S}$ and reset $\overline{R}$ signals for the second-stage latch from the single-ended pulse generator. The circuit for the generation of the reset signal is at the same time used as part of the shut-off circuit in the pulse generator.

This reuse of the latch reset simplifies the implementation of the flip-flop versus the USPARC and also reduces the switching activity of the nodes CK1, R, and CK3 in Fig. 8. This conditional shut-off feature reduces total energy of the flip-flop, with no effect on the functionality since the node $\overline{\mathrm{S}}$ has already switched low. However, the critical path of the CSE is now through R which induces a delay penalty versus the USPARC, but this is largely compensated by the fact that R is driving a single-stack nMOS (M7) with no short-circuit current from the pMOS (M5). The main advantage of the IPP over other flip-flops remains the large gain achievable by the second-stage latch. This large latch gain allows for smaller load presented to the first-stage pulse generator, and consequently lower energy consumption for the same delay.

The STFFSE is based on the regenerative pulse at the node $\mathrm{C_S}$ that is asserted after each falling edge of the clock. If the input D is high at the time the pulse is asserted, the node $\overline{\mathrm{S}}$ is discharged, which in turn extends the pulse at the node $\mathrm{C_S}$ by opening the pull-up path through transistors M4 and M6. If the input D is low at the time the pulse at the node $\mathrm{C_S}$ is asserted, the node $\overline{\mathrm{S}}$ stays high and the node $\mathrm{C_S}$ quickly switches low after the delay through inverters I1–I3. In this way, the flip-flop is transparent to the transition at the input D during the duration of the regenerative pulse, which enables the soft clock edge property and allows for clock uncertainty absorption [2]. The second stage of the flip-flop is the dynamic-to-static latch that consists of the NAND gates N1 and N2, with clock input as the default reset. In addition to the soft clock edge property, the single-ended skew-tolerant flip-flop is fast as the critical path consists of a domino-like gate with an nMOS stack of only two transistors and a simple high-gain latch. However, the high speed is traded for the large energy consumption due to the generation of the regenerative clock pulses and the high activity of the heavily loaded nodes $\overline{\mathrm{S}}$ and $\overline{\mathrm{R}}$.

C. Transmission-Gate Pulsed Latch

The transmission-gate pulsed latch (TGPL, Fig. 10), used in several generations of Intel processors [19], is the straightforward implementation of the pulsed latch topology [2], [4]. It consists of a clock pulse generator that provides a pulse and its complement to a conventional transmission-gate transparent latch. The pulse generator creates a short $0 \rightarrow 1 \rightarrow 0$ pulse at the node CP after the rising edge of the clock *Clk*. The duration of this pulse is determined by the propagation delay through the three inverters in Fig. 10. The TGPL is regarded to be among the fastest storage elements, as its critical path consists of a single transparent latch. This short delay is obtained at the expense of large hold time and relatively large power consumption, dominated by the power of the pulse generator. When designing a pulsed latch, particular care must be paid to the pulse generator. In order to prevent pulse distortion and ensure proper operation, a typical fanout-of-2 (FO2) slope is maintained on $\overline{\mathrm{CP}}$. Furthermore, it is typically required that the pulse is generated locally to avoid pulse shrinkage and to reduce the effects of the noise. For the same reason, the pulse generator is somewhat overdesigned, in order to ensure that the pulse has sufficient width over all process corners, supply voltages, and operating temperatures.

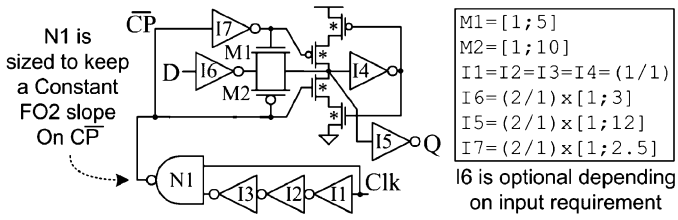

Fig. 10. Circuit and transistor sizing of the transmission-gate pulsed latch (TGPL).

The TGPL is typically shown without the input inverter I6 and presents the passgate to the input D. This becomes a problem when evaluating the input capacitance of the CSE since it varies depending on the clock. When the pulse occurs, all the transistors on the data path from D-to-Q load directly the input D. Hence, setting a fixed input size basically fixes the size of the CSE. For example, in this technology, the drain/source capacitance is equivalent to the gate capacitance and this topology has eight transistors connected to the input. With a 3-μm maximum input, yielding 9 minimum-size units of gate capacitance, fixing the input to 3 μm would make the CSEs all minimum sized. Furthermore, fixing the input capacitances to 1 μm or 2 μm would not be possible. However, by allowing the input capacitance to be large, this CSE presents significant advantages in terms of speed since it is only one stage. This advantage is further amplified with a small gain since the heavy parasitic capacitance, due to the passgate and the keeper on the input, limits the gain efficiency of the output inverter. To illustrate this behavior quantitatively, the TGPL without the input inverter I6 was also analyzed but with an input capacitance of up to 9 μm.

Fig. 11 shows the energy–delay performance behavior versus input and output capacitance when the input inverter I6 is removed. In the case of 14-μm output load and 3 μm input load, the TGPL with the input inverter achieves up to 1 FO4 delay improvement with similar energy spending. In the case of 2.5 μm output load and 3-μm input load, the TGPL without the input inverter achieves better delay. Additionally, if the input capacitance is increased, the TGPL without I6 can achieve 1 FO4 D-to-Q delay with a 2.5-μm output and 1.4 FO4 D-to-Q delay with a 14-μm output. This implies the TGPL is likely to be optimal under high CSE input capacitance. Yet, the ED performance of the TGPL with and without the input inverter is largely dependent on the input/output conditions and no clear comparative conclusion can be drawn. If the energy–delay information of the subsequent logic block is combined with the various fixed input/output characteristics shown in Fig. 11, a clear single characteristic can be drawn. This characteristic would keep only the best TGPL designs; hence the full pipeline analysis is necessary here.

D. Master–Slave Latches

The master and slave latches are clocked with complementary clock phases, generated locally (for the same reason as described for TGPL) in order to allow a fair comparison with other storage elements. The transmission-gate master–slave latch (TGMS, [20], Fig. 12) is a conventional master–slave

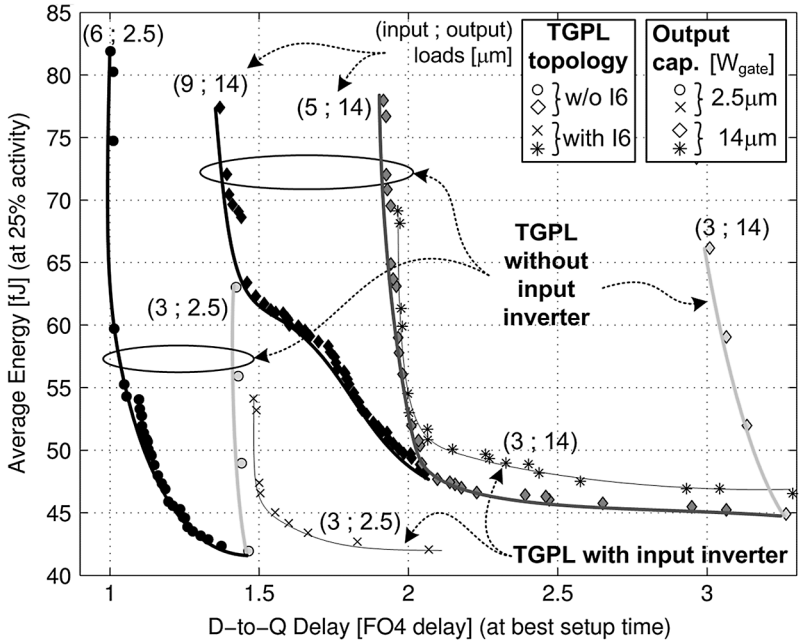

Fig. 11. Effect of removing the input inverter of the TGPL on its energy–delay tradeoff versus input/output capacitance.

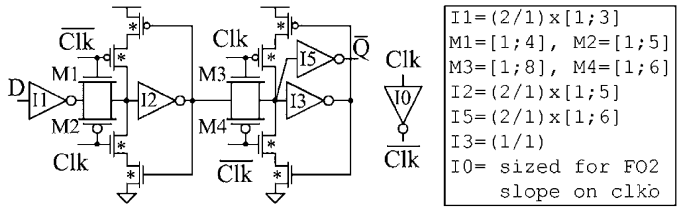

Fig. 12. Circuit and transistor sizing of the transmission-gate master–slave latch (TGMS).

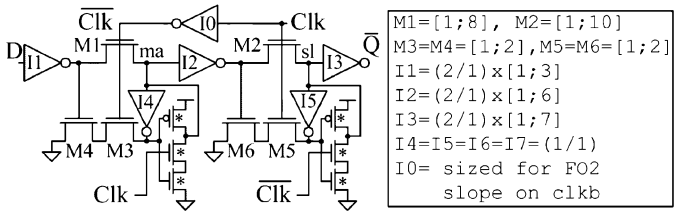

Fig. 13. Circuit and transistor sizing of the write-port master–slave latch (WPMS).

topology in which both master and slave latches are implemented using the transmission gates [2]. It is used in the PowerPC 603 low-power processor [20], and it is generally considered the most energy-efficient general-purpose storage element topology. For the same reasons explained in the previous section, the inverter I1 has been added for small input specifications. Although, when the TGMS performs better without I1 for the same input capacitance, the inverter is removed. The ED space results for this CSE are shown in Fig. 14.

The write-port master–slave latch (WPMS, [21], Fig. 13) works also like a classic master–slave structure. The implementation of each latch is inspired by a standard SRAM 6T cell. Each side of the keeper is controlled by a single nMOS which is driven by $\overline{Clk}$ or Clk if the latch is, respectively, master or slave. When the clock opens both nMOS transistors in a wordline manner, the keeper is push-pulled from each side to change its state. The advantage of this structure is the removal of the pMOS from the passgates, which decreases the clock load and the parasitic capacitance on the datapath. However, since the pMOS is missing, the keepers cannot be conditional on the pull-up in order to bring the nodes *ma* and *sl* from $V_{dd} - V_t$ to V_{dd} when logic high is needed. The C^{2}MOS master–slave latch has also been proposed [22], but was shown to be inferior to TGMS [1], [2] in all cases [23].

Fig. 14 shows the WPMS results in comparison to the TGMS results. This comparison can be made here because the WPMS is either similar or worse than the TGMS in all cases. In the high energy sensitivity region of both the high and low output load conditions, the two master–slave structures achieve equivalent energy–delay performance. On the other end, in the high delay sensitivity region, the WPMS is consistently worse in energy than the TGMS by at most 3.5 fJ. This occurs because at the minimum sizing condition, or close to it, the cost in energy for nonconditional level-high keepers is greater than the savings provided by the nMOS-only passgate. These results are consistent with the high and low ends of the output load range as well as with other input restrictions (not shown in Fig. 14 for clarity). In this case and range of system conditions, the WPMS topology can be discarded without combining the data with the logic. However, because this CSE structure reduces the clock load by increasing the energy cost of changing the keeper state, the WPMS is more appropriate than the TGMS for very low input switching activities.

IV. CSE Comparative Results for a Pipeline Stage Case

The set of representative energy-efficient characteristics of a CSE, as shown in Fig. 2 for USPARC, was generated for each

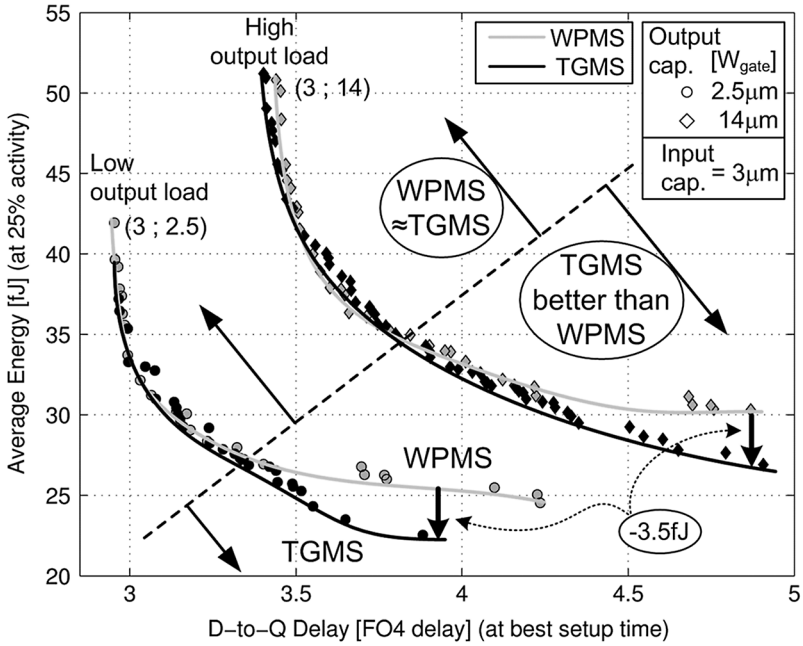

Fig. 14. Direct comparison between TGMS and WPMS.

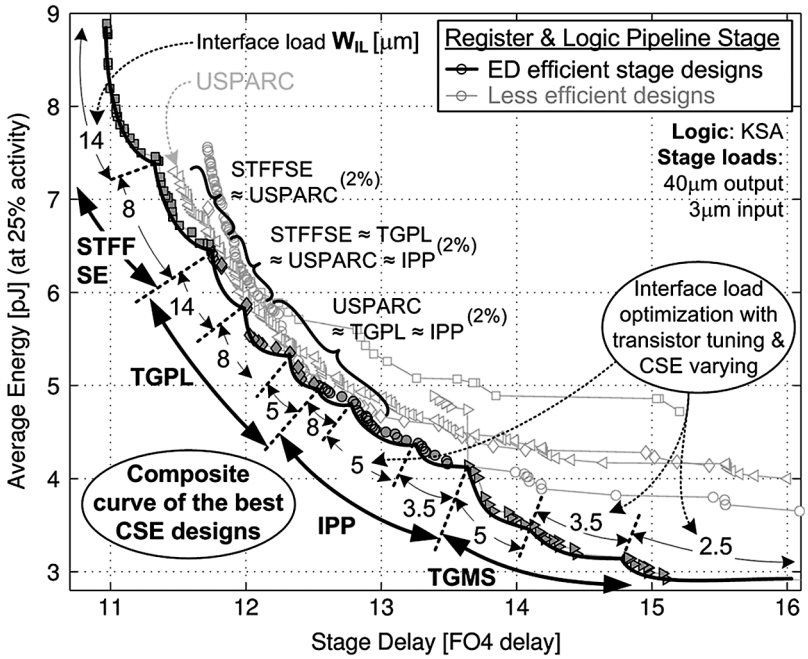

Fig. 15. Pipeline stage case study: Choice of the best CSE and interface load with a small input capacitance restriction to the stage.

CSE presented in Section III besides WPMS and SDFF. Then, following the method in Section II, each CSE result was combined with the logic (KSA shown in Fig. 3) as shown in Fig. 5. The presented comparison between the CSEs is true only in this pipeline stage example; other system conditions or logic can completely change the optimum CSE selection. The example in Fig. 15 consists of a pipeline stage as shown in Fig. 4 with the input capacitance fixed to at most 3-μm gate width and an output load of 40 μm on the logic.

Fig. 15 shows that the composite curve of the best CSE designs is made of four CSEs: STFFSE, TGPL, IPP, and TGMS. For a 3-μm input specification, the TGPL with the input inverter is more efficient than without, as seen in Fig. 11; on the other hand, TGMS is more efficient without it. The best TGPL and TGMS combinations are reported in Fig. 15. In general, because the interface load is allowed to change, each CSE operates best for a different range of interface load. STFFSE and TGPL are expected to provide small delays, consequently their interface load is expected to be high. Similarly, the IPP and TGMS are typically low energy and slower design, thus the interface load should be low. This intuitive result is proven to be true in both cases in Fig. 15. Since this method explores all the best pos-

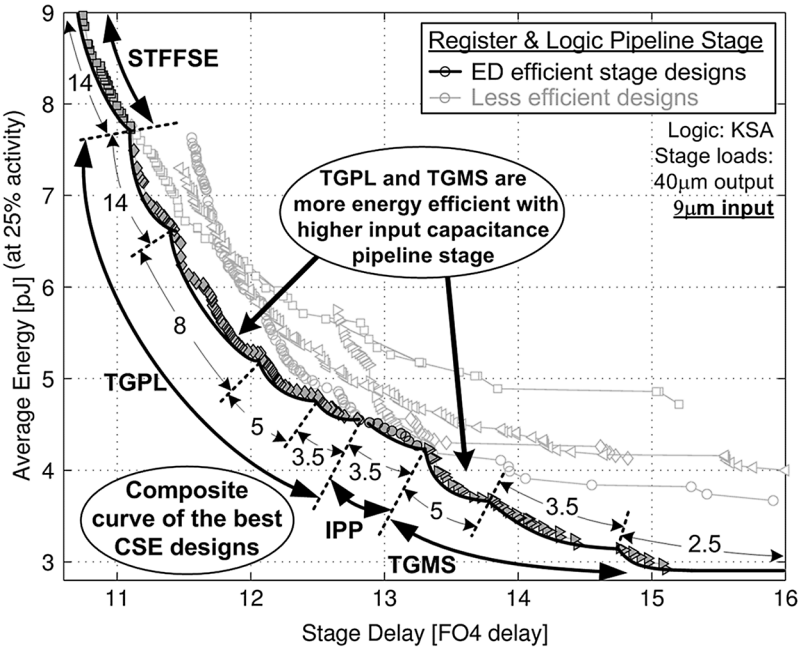

Fig. 16. Pipeline stage case study: Choice of the best CSE and interface load with a large input capacitance restriction to the stage.

sible combinations between the logic and the CSE, the final results yield surprisingly similar ED performance for several CSE topologies. Between 11.5 FO4 and 13 FO4, the dynamic structures and the TGPL provide essentially the same ED performance results within 2% of the best case, as shown in Fig. 15. There are many factors responsible for this situation, but the main controlling factors are the following.

- The USPARC and IPP have similar structures, and the benefit of having a conditional clock in IPP makes sense only for low energy designs, above 13 FO4 in Fig. 15. Because the IPP critical path is longer than the USPARC, the energy savings are compensated by the extra energy cost to improve the IPP speed versus the USPARC. This results in similar ED performance for IPP and USPARC in the high energy sensitivity region (below 13 FO4). Similarly, the STFFSE includes extra logic to increase its speed, which makes it best suited for the high energy sensitivity region. However, when it is sized down, the ED performance becomes similar to the other dynamic structures (IPP, USPARC).
- The 3-μm input limitation of the pipeline stage represents an important penalty for the TGPL, which thrives on large input capacitance as shown in Fig. 11. This constraint basically limits its speed because an additional inverter is necessary on the input to build the gain through the latch datapath. This basically holds back the TGPL in term of delay to the same level as the USPARC.

In a second example, we increased the allowed input size of the pipeline stage showed in Fig. 15 from 3 μm to 9 μm. The results in Fig. 16 shows that the TGPL occupies a much larger portion of the composite curve, indicating that it is more energy efficient under larger input capacitance constraint. There are two reasons for this.

- By having a 9-μm input limitation, the input inverter of the TGPL can be removed because the latch no longer needs to build gain. Also, a larger input allows a better transistor tuning flexibility for the first stages of both the TGPL and the TGMS. This yields improvements in both energy and delay.
- The dynamic structures cannot keep up with the speed improvement because the first stage of this type of flip-flop is a footed domino gate and increasing the input implies increasing the clock load. Thus, the speed gained from a faster drive is compensated by a large clock energy consumption cost, resulting in non-energy-efficient designs.

In both Figs. 15 and 16, no single CSE represents an optimal choice for all frequency or energy targets. Depending on the input conditions, the distribution of the best CSE selection on the composite curve varies greatly. However, from a comparison standpoint, it is necessary to relate the proposed method to the commonly assumed EDP or single point analysis. Fig. 17 shows the corresponding design points chosen based on EDP to their location within the energy–delay space of the pipeline stage example of Fig. 15. The first limitation of a single point comparison is the lack of information for intermediate design targets. For example, if our stage delay target is 13 FO4 and only the 14-μm interface load EDP analysis is available, the designer will choose the IPP EDP design point since the TGMS cannot achieve such speed. However, for the 13 FO4 delay target, the IPP designs optimized for the 5-μm interface load can achieve up to 23% energy saving (Fig. 17). Also, the EDP chosen design may not even be optimal in any case. For example, the TGMS and IPP designs used in the Fig. 17 EDP analysis are unrelated to the actual energy-efficient designs. As shown in Fig. 15, the IPP does not make sense for interface loads above 8 μm and the TGMS does not make sense for interface loads above 5 μm. Hence, any IPP or TGMS design optimized for 14 μm cannot

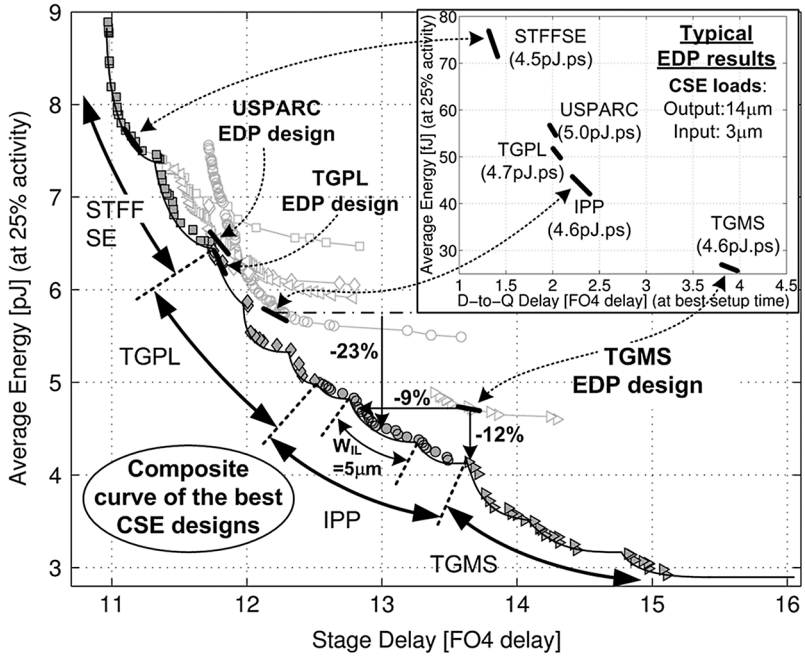

Fig. 17. Energy–delay performance relationship between an EDP comparison and this work for a set of CSE within a pipeline stage.

be optimal for that pipeline stage. Furthermore, if we look at the TGMS EDP point in Fig. 17, both energy and delay targets can be improved by at least 10% by selecting the IPP and by reducing the interface load. This type of design decision can be counterintuitive since IPP is a dynamic structure. On the other hand, the STFFSE and TGPL EDP designs are part of the composite characteristic of the energy-efficient designs (shown in bold in Fig. 15 and Fig. 17). This is expected since the EDP comparison was made with a 14-μm CSE output load, and this interface load happens to be optimum in this region for that pipeline. This brings us to the essential problem with metric- or fixed-interface-based comparisons: Without the knowledge of the optimum logic interface load, a set of CSE designs compared based on any fixed input and output load will yield several irrelevant design options.

V. Conclusion

This paper has presented a consistent method for analyzing CSEs in the entire energy–delay design space. It has shown that the conventional comparison approaches based on the fixed energy–delay metrics are incapable of identifying an optimal design choice as they do not reflect any particular target application and exact specification of the CSEs. We have shown that the optimal delay budget of the CSE is very sensitive to the cycle time and the characteristic of the logic block in the pipeline stage. We defined the composite energy-efficient characteristic over all storage element topologies and interface loads that allow us to define the natural target application for all CSEs. Our analysis studied the effects of the delay target, output load, and input load restriction to the CSE performance in a quantitative manner. This analysis approach was demonstrated on a group of state-of-the-art CSEs used in modern microprocessors and a group of experimental CSEs in the context of a practical application. Due to their fundamental structural advantage over master-slave latches, flip-flops tend to offer the most energy-efficient solution and the best gain for high- and medium-speed targets, while master–slave latches, which benefit from a simpler structure and low internal switching activity, perform the best in the low-power region. The pulsed latch can also show energy efficiency, but only in pipeline stages where the CSE input can be large. However, in the system perspective, in all but the highest performance design targets, the low-power CSEs such as TGMS or IPP tend to be the optimal or near-optimal choice. In the most demanding applications, the high gain of the output stage seems to be an important design parameter due to the large loads that the CSE needs to drive. Although, for non-dynamic structures, a large CSE input can mitigate the effect of a less gain-efficient output stage. By using our comparison methodology versus a single point comparison, we have shown that at least a 10% improvement in either energy or delay can be achieved in most cases, and the energy of the whole pipeline stage can be reduced by up to 23% in some cases.

We have shown that the increased complexity of our analysis is well justified by the overall energy improvements it provides for a given cycle time over the design methodologies based on fixed metric and fixed delay budget. This paper has shown that no single energy–delay metric is optimal for CSE comparison.

Appendix
Simulation Methodology and Assumptions

The primary goal of the simulation is to extract an accurate set of energy-efficient configurations for each CSE over a range of input sizes and output loads. Extracting these characteristics includes layout and wire parasitic capacitance estimates, which are re-evaluated for each combination of transistor sizes. The technology model used is a 130-nm process with a fanout-of-4

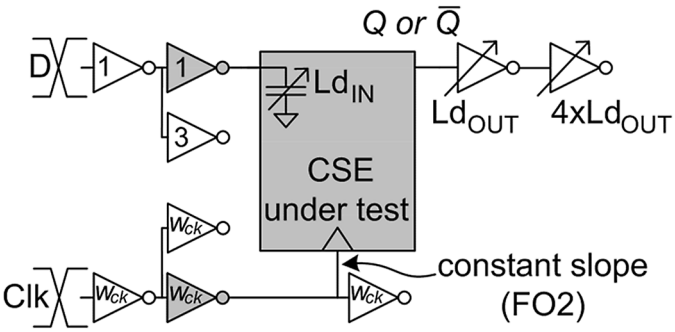

Fig. 18. Load flexible CSE simulation setup.

(FO4) delay of 45 ps. The granularity of the transistor width is set to 0.32 μm, which is equal to the minimum transistor width in this technology. However, in some cases a 0.16 μm granularity may be chosen, especially when the energy–delay performance is sensitive to a specific transistor like the passgate in the TGPL. The HSPICE simulations are fully managed and automated by a tool written in PERL which provides a complete set of energy-efficient characteristics for a particular CSE. This task is usually performed by circuit optimizers such as in [7], [9], and [10]. Our tool also checks for output noise, when a certain combination of transistor sizes generates unacceptable glitches the tool rejects the design.

During the simulations, the clock load varies because the slope of the clock input to the CSE is kept constant to a FO2 clock slope (Fig. 18) and the size of the clock driver is automatically updated for each simulated sizing configuration. This constant clock slope policy is typically used to maintain clocking uncertainties within system specifications. However, to moderate the discrepancies, the energy spent by the clock driver is accounted for as an approximation of the energy impact on the clock distribution due to the internal CSE clock load.

A. Delay Quantification

Since the delay of a CSE depends on the delay between the data and clock arrivals [1], the simulation procedure must determine the setup time for each transistor size combination. Nedovic *et al.* [18] showed that a minimum D–Q delay zone is flat for at least 10 ps of data-to-clock variation for all CSEs presented in Section III in the same technology. The granularity chosen for the simulations performed in this work was set to 5 ps, which yields a negligible minimum D-to-Q delay error versus setup time.

B. Energy Quantification

The energy is measured by integrating the supply current of the CSE, $i_{a\to b}(Int)$, the clock driver, $i_{a\to b}(Clk)$, and the data driver(s), $i_{a\to b}(In)$, over the clock cycle time T_{CLK} at the nominal supply voltage. The elements of this breakdown are shown in gray in Fig. 18 as well as in (1) for a transition from a to b logic level, where $a, b \in \{0, 1\}$. Note that t_{TRIG} in (1) stands for the latching edge time, rising or falling depending on the CSE, and U stands for the CSE optimum setup time.

$$E_{a\to b} = V_{DD} \cdot \int_{t_{TRIG}-|U|}^{t_{TRIG}+T_{CLK}-|U|} \{i_{a\to b}(Int) + i_{a\to b}(In) + i_{a\to b}(Clk)\} \cdot dt$$
$$E_{a\to b} = E_{a\to b}(Int) + E_{a\to b}(In) + E_{a\to b}(Clk) \quad (1)$$

$$E_{CSE}(\alpha) = \frac{1-\alpha}{2} \cdot (E_{0\to 0} + E_{1\to 1}) + \frac{\alpha}{2}(E_{0\to 1} + E_{1\to 0}) \quad (2)$$

The total energy for any desired activity factor α is obtained by combining the four transition cases ($0 \to 0$, $0 \to 1$, $1 \to 0$, and $1 \to 1$) with appropriate weight factors, as shown in (2). The cycle time of 1 ns is chosen for the simulation. For this technology node, the offset in energy due to leakage is negligible.

ACKNOWLEDGMENT

The authors would like to thank B. Zeydel for his suggestions on adder and pipeline stage design.

REFERENCES

[1] V. Stojanovic and V. Oklobdzija, "Comparative analysis of master-slave latches and flip-flops for high-performance and low-power systems," *IEEE J. Solid-State Circuits*, vol. 34, no. 4, pp. 536–548, Apr. 1999.

[2] V. G. Oklobdzija, V. M. Stojanovic, D. M. Markovic, and N. M. Nedovic, *Digital System Clocking: High-Performance and Low-Power Aspects*, 1st ed. New York: Wiley/IEEE Press, 2003.

[3] V. G. Oklobdzija, "Clocking and clocked storage elements in a multi-gigahertz environment," *IBM J. Res. Devel.*, vol. 47, pp. 567–584, Sep./Nov. 2003.

[4] J. Tschanz, S. Narendra, C. Zhanping, S. Borkar, M. Sachdev, and V. De , "Comparative delay and energy of single edge-triggered and dual edge-triggered pulsed flip-flops for high-performance microprocessors," in *Proc. Int. Symp. Low Power Electronics and Design (ISLPED)*, 2001, pp. 147–152.

[5] N. Nedovic, "Clocked storage elements for high-performance applications," Ph.D. dissertation, University of California, Davis, 2003, p. 353.

[6] N. Nedovic, W. W. Walker, and V. G. Oklobdzija, "A test circuit for measurement of clocked storage element characteristics," *IEEE J. Solid-State Circuits*, vol. 39, no. 8, pp. 1294–1304, Aug. 2004.

[7] V. Zyuban, "Optimization of scannable latches for low energy," *IEEE Trans. Very Large Scale Integrat. (VLSI) Syst.*, vol. 11, no. 10, pp. 778–788, Oct. 2003.

[8] D. Markovic, B. Nikolic, and R. W. Brodersen, "Analysis and design of low-energy flip-flops," in *Proc. Int. Symp. Low Power Electronics and Design (ISLPED)*, 2001, pp. 52–55.

[9] V. ZyubanD. Brooks, V. Srinivasan, M. Gschwind, and P. BoseP. N. StrenskiP. G. Emma, "Integrated analysis of power and performance for pipelined microprocessors," *IEEE Trans. Comput.*, vol. 53, no. 8, pp. 1004–1016, Aug. 2004.

[10] V. Zyuban and P. Strenski, "Unified methodology for resolving power-performance tradeoffs at the microarchitectural and circuit levels," in *Proc. Int. Symp. Low Power Electronics and Design (ISLPED)*, 2002, pp. 166–171.

[11] S. Heo and K. Asanovic, "Load-sensitive flip-flop characterizations," in *Proc. IEEE Computer Society Workshop on VLSI*, 2001, pp. 87–92.

[12] H. Q. DaoB. R. ZeydelV. G. Oklobdzija, "Energy optimization of pipelined digital systems using circuit sizing and supply scaling," *IEEE Trans. Very Large Scale Integrat. (VLSI) Syst.*, vol. 14, no. 2, pp. 122–134, Feb. 2006.

[13] R. Heald *et al.*, "A third-generation SPARC V9 64-b microprocessor," *IEEE J. Solid-State Circuits*, vol. 35, no. 11, pp. 1526–1538, Nov. 2000.

[14] I. E. Sutherland and R. F. Sproull, *Logical Effort: Designing for Speed on the Back of an Envelope*. Cambridge, MA: MIT Press, 1991.

[15] P. M. Kogge and H. S. Stone, "A parallel algorithm for the efficient solution of a general class of recurrence equations," *IEEE Trans. Comput.*, vol. C-22, pp. 786–793, Aug. 1973.
[16] F. Klass, "Semi-dynamic and dynamic flip-flops with embedded logic," in *Proc. Symp. VLSI Circuits*, 1998, pp. 108–109.
[17] J. Yuan and C. Svensson, "High-speed CMOS circuit technique," *IEEE J. Solid-State Circuits*, vol. 24, no. 2, pp. 62–70, Feb. 1989.
[18] N. NedovicV. G. OklobdzijaW. W. Walker, "A clock skew absorbing flip-flop," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2003, pp. 342–352.
[19] S. D. Naffziger and G. Hammond, "The implementation of the next-generation 64 b Itanium microprocessor," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2002, pp. 344–472.
[20] G. Gerosa *et al.*, "A 2.2 W, 80 MHz superscalar RISC microprocessor," *IEEE J. Solid-State Circuits*, vol. 29, no. 12, pp. 1440–1454, Dec. 1994.
[21] D. Markovic and J. Tschanz, "Transmission-gate based flip-flop," U.S. patent 6,642,765, Nov. 2003.
[22] Y. SuzukiK. OdagawaT. Abe, "Clocked CMOS calculator circuitry," *IEEE J. Solid-State Circuits*, vol. SC-8, no. 6, pp. 462–469, Dec. 1973.
[23] C. Giacomotto and N. NedovicV. G. Oklobdzija, "Energy-delay space analysis for clocked storage elements under process variations," in *Proc. 16th Int. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, Montpellier, France, 2006, pp. 360–369.

Christophe Giacomotto (S'01) was born in Cannes, France, in 1978. He received the M.S.E.E. degree from the Ecole Superieure d'Ingenieurs en Electronique et Electrotechnique (ESIEE), Paris, France, in 2001. He is currently pursuing the Ph.D. degree in the Department of Electrical and Computer Engineering, University of California at Davis.

From 2001 to 2003, he was a research staff member with Fujitsu at HAL Computer Systems and at Fujitsu Processor Technology where he worked on technology issues and projections for the SPARC processors product line. He provided technological expertise in various areas such as electromagnetic noise, circuit design rules and interconnect optimizations, on which he holds several patents. His current research interests include clocking techniques and systems for low-power and high-performance digital systems as well as low-power circuit design techniques.

Nikola Nedovic (S'99–M'03) received the Dipl.Ing. degree in electrical engineering from the University of Belgrade, Belgrade, Yugoslavia, in 1998, and the Ph.D. degree from the University of California at Davis in 2003.

In 2001, he joined Fujitsu Laboratories of America, Sunnyvale, CA, where he works in the area of high-speed communications and high-performance and low-power VLSI circuits. His research interests include analog and mixed-signal circuits for wireline communications, clock and data recovery, and circuit design and clocking strategies for high-speed and low-power digital applications.

Dr. Nedovic received the Anil K. Jain Prize for the Best Doctoral Dissertation in Electrical and Computer Engineering from the University of California at Davis in 2003.

Vojin G. Oklobdzija (M'82–SM'88–F'96) received the Dipl. Ing. degree in electrical engineering from the University of Belgrade, Belgrade, Yugoslavia, in 1971, and the Ph.D. degree from the University of California at Los Angeles in 1982.

From 1982 to 1991, he was with the IBM T. J. Watson Research Center, Yorktown Heights, NY, where he made contributions to the RISC processors and superscalar computer design resulting in several patents, most notably on register renaming, which enabled a new generation of computers. From 1988 to1990, he was an IBM visiting faculty at the University of California at Berkeley, and from 1991 to 2006, Professor of computer engineering at the University of California at Davis. He has served as a consultant to Sun Microsystems, Bell Laboratories, Hitachi, Fujitsu, SONY, Texas Instruments, Intel, Samsung, and Siemens/Infineon Corporation, where he was a Principal Architect for the Infineon TriCore processor. He is President and CEO of Integration Corp. and an independent consultant. Prof. Oklobdzija holds 14 US and 18 international patents. He has published more than 150 papers, five books, and a dozen book chapters in the areas of circuits and technology, computer arithmetic and computer architecture. His book *Computer Engineering* won the Outstanding Academic Title award in 2002. He has given over 150 invited talks and short courses in the U.S., Europe, Latin America, Australia, China, Korea, and Japan. He directs the ACSEL Laboratory (http://www.acsel-lab.com), which is involved in digital circuit optimization for low-power and ultra low-power, high-performance system design and sensor nodes.

Prof. Oklobdzija is a Distinguished Lecturer of the IEEE Solid-State Circuits Society. He serves as Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II, *IEEE Micro*, and the *Journal of VLSI Signal Processing*. He served as Associate Editor of the IEEE TRANSACTIONS ON COMPUTERS (2001–2005), IEEE TRANSACTIONS ON VERY LARGE SCALE OF INTEGRATION (VLSI) SYSTEMS from 1995 to 2003, the ISSCC digital program committee from 1996 to 2003, first A-SSCC in 2005, International Symposium on Low-Power Electronics, ISLPED, Computer Arithmetic Symposium, ARITH and numerous other conference committees. He was a General Chair of the 13th Symposium on Computer Arithmetic in 1997.