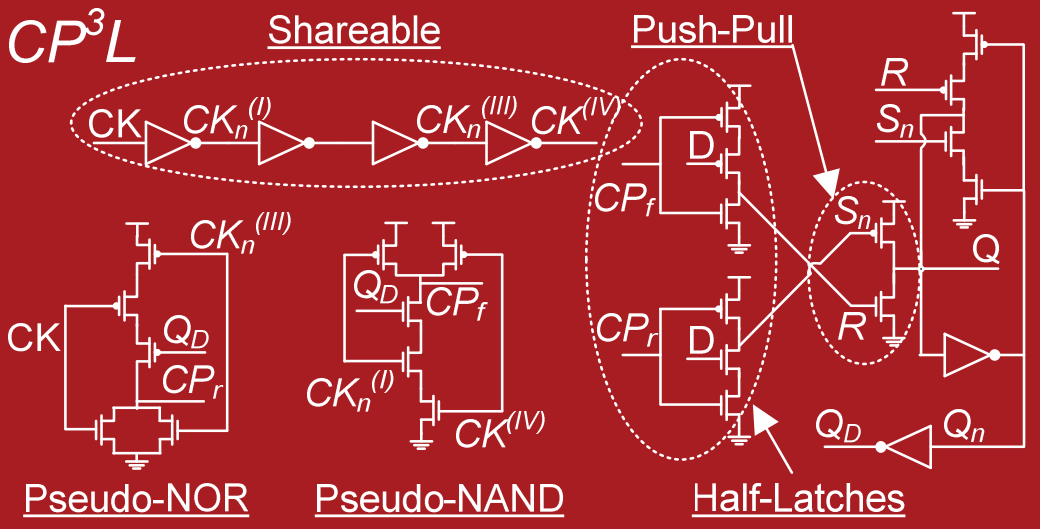
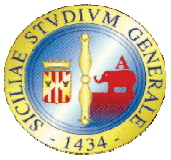


Nanometer CMOS Clocked Storage Elements: Optimization Techniques, Comparison and Novel Energy-Efficient Design Solutions

Elio Consoli

Ph.D. in Electronic, Automation and Control of Complex Systems Engineering (XXIV cycle)
University of Catania, Italy



Nanometer CMOS Clocked Storage Elements: Optimization Techniques, Comparison and Novel Energy-Efficient Design Solutions

by
Elio Consoli

B.S. (University of Catania, Italy) 2005
M.S. (University of Catania, Italy) 2008

A dissertation submitted in partial satisfaction of the requirements for the degree of
Doctor of Philosophy
in
Electronic, Automation and Control of Complex Systems Engineering
(XXIV cycle)

Dept. of Electrical, Electronic and Information Engineering
Faculty of Engineering
University of Catania, Catania, Italy

Coordinator:
Prof. Luigi Fortuna
Tutor:
Prof. Gaetano Palumbo

2011

To

(mom) Francesca
(dad) Maurizio
Laura

ABSTRACT

Clocked storage elements are among the most important elements in the design of digital systems, such as microprocessors, since they allow to synchronize and regulate the entire flow of digital data within the system.

With the aim of obtaining conspicuous performance increments at each process generation, dimensional scaling has been supported by the reduction of the number of logic stages within each pipeline stage. Therefore, an increasing impact of the timing overhead due to clocked storage elements on the clock period can be observed. Moreover, the continuous increase in energy consumption has become the major concern limiting the speed performances of VLSI integrated circuits, insomuch as, even for high-speed systems, designs undergo a power limited regime and the achievement of energy-efficiency becomes the primary target.

The topics of energy-efficient design, analysis, comparison and selection of suitable clocked storage elements topologies for applications in nanometer technologies have been the focus of the research activity carried out by the candidate in pursuit of the Ph.D. degree. The aim of this thesis is to provide a deep understanding of the challenges relative to clocked storage elements design and selection when including all the above mentioned aspects, as well as to propose novel energy-efficient solutions at the transistor- and micro-architectural design levels.

The basic theoretical foundations are provided to set the stage for the comprehension of analyses and results. Exhaustive methodologies are presented and many analytical derivations are included, since they allow to gain an insight on the main dependencies of relevant parameters on circuital properties. Finally, several results, which have been derived by carrying out extensive simulation analyses and measurements on an integrated chip prototype are reported to emphasize the practical perspective of the work.

ACKNOWLEDGEMENTS

Firstly, I want to thank my Ph.D. supervisor, Prof. Gaetano Palumbo, and my research advisor, Prof. Massimo Alioto, for their technical and inspirational support. All the results achieved during my Ph.D. experience are shared with them and ensue from their technical teachings, constant motivation and invaluable guidance. Because of their wonderful human approach, which I am extremely grateful for, they are not only scientific tutors but also life mentors.

I wish to express my most sincere gratefulness to Prof. Salvatore Pennisi, Prof. Gianluca Giustolisi and Prof. Giuseppe Di Cataldo, whose human and cultural worthiness I admire. They have always provided me precious helpfulness and careful suggestions.

I would like to acknowledge the Ph.D. coordinator, Prof. Luigi Fortuna, for its motivating guide and all the other Professors, Researchers and Research Assistants at DIEEI, whose organizational skills have made attending my Ph.D. program a fruitful and exciting experience.

Part of my research activity has been carried out at the Berkeley Wireless Research Center, UC Berkeley, and hence I want to thank Prof. Jan Rabaey, all BWRC staff and, again, Prof. Massimo Alioto for giving me the wonderful opportunity to work in such an astonishing research environment. The collaboration with BWRC culminated in the realization of an integrated prototype and I thank STMicroelectronics for chip fabrication.

I am also grateful to my lab mates at DIEEI, Melita Pennisi, Dario Mita, Walter Aloisi and David Cristaldi for their enjoyable company. A special thank goes to Davide Marano for his sincere friendship.

I want to extend my thanks to all friends, relatives and colleagues, who have accompanied me throughout this three-year experience.

Finally, I am immensely grateful to my family.

I wish to thank my sister for her support during hard working periods and for helping me to preserve my confidence and reference points when worrying about the choices for the future.

I will be endlessly indebted to my father, who has always supported me by any means. His experience and unconditional love have been the sources of indispensable suggestions, which have represented the strong and encouraging foundation for all the decisions that I have taken.

I can hardly express my gratitude to my mother, which my deepest thanks goes to. Her enormous and endless effort to sustain, encourage and make me live a fully happy life is undoubtedly the biggest gift I have ever received. I am who I am thanks to her infinite love and she will always be the main reference for any future achievement.

Least but not last, I want to thank my lovely fiancée Laura, who changed my life by giving me infinite joy and happiness. Day by day, Laura has made me know the real meaning of the word “love” and, while preparing my future next to her, I am truly conscious that she is the most important person in my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

The design of the clock network represents a crucial aspect when dealing with CMOS VLSI integrated circuits, as it strongly affects not only the chip speed, but also its overall energy consumption.

Independently from the basic nature (fully synchronous, globally asynchronous locally synchronous) of the systems where it is employed, any clock network can be subdivided into three main parts. Indeed, similarly to the structure of a tree, we can identify a root constituted by the circuits devoted to clock generation, branches constituted by the wires and circuits devoted to clock distribution and, as the final leaves, the clocked storage elements, i.e. latches and flip-flops.

In particular, clocked storage elements are among the most important elements in the design of digital systems, such as microprocessors. They separate the various stages which a pipeline is made up by, maintain the present logic state and prevent the transition towards a new one until the "right" instant occurs. On the whole, they allow to synchronize and regulate the entire flow of digital data within the system.

With the aim of obtaining conspicuous performance increments at each process generation, dimensional scaling has been supported by the reduction of the number of logic stages (logic depth) within each pipeline stage. Therefore, an increasing impact of the timing overhead due to clocked storage elements on the clock period can be observed. On the other hand, due to the high switching activity featuring clocked gates, the overall dissipation of the clock network can be as high as 30-50% of the overall chip energy budget. Moreover, the fraction of this contribution due to clocked storage elements tends to increase again due to the decreasing logic depth and, in some cases, because of the adoption of novel low energy clock distribution techniques.

The above issues, together with the needs for an high operation robustness and for the capability to deal with clock signal uncertainties, make the clocked storage elements design a key aspect in the VLSI systems domain, and, in order to account for all the mentioned aspects, it may be also quite complex. For this reason, these circuits have been extensively studied in the past and a significant effort has been devoted to provide guidelines for the identification of new circuital solution and the selection of the most suitable clocked storage elements according to the requirements of the application.

Moreover, the task of carrying out topologies selection and optimized design strategies is now made more complicated since two main challenges arise when adopting the current technologies: the increasing relevance of energy consumption and the effects arising at nanometer scale.

In particular, the continuous increase in energy consumption (due to the raising impact of leakage dissipation) has become the major concern limiting the speed performances of digital VLSI integrated circuits, insomuch as, even for high-speed systems, designs undergo a so-called power limited regime. Therefore, since the achievement of energy-efficiency must be the primary target, a deep understanding of the energy-delay tradeoff and the related design issues is crucial.

Secondly, when entering the nanometer scale, several effects have to be considered, such as the impact of layout parasitics associated with interconnects, degrading both speed and energy, and leakage, affecting energy both in active and in standby operation modes. Such effects, which once could be neglected, have now become prominent.

The topics of energy-efficient design, analysis, comparison and selection of suitable clocked storage elements topologies for applications in nanometer technologies have been the focus of the research activity carried out by the candidate in pursuit of the Ph.D. degree. The aim of this thesis is to provide a deep understanding of the challenges relative to clocked storage elements design and selection when including all the above mentioned aspects, as well as to propose novel energy-efficient solutions at the transistor- and micro-architectural design levels. This target is accomplished by organizing the candidate's recent results as well as the other ones reported in the literature.

The basic theoretical foundations are provided to set the stage for the comprehension of analyses and results. Exhaustive methodologies are presented and many analytical derivations are included, since they allow to gain an insight on the main dependencies of relevant parameters on circuital properties. Finally, several results, which have been derived by carrying out extensive simulation analyses and measurements on an integrated chip prototype are reported to emphasize the practical perspective of the work.

The outline of the work is as follows.

The first three chapters contain all the basic theoretical elements, including some novel results, that are exploited in the remaining part of the book. Chapter 1 describes the well known Logical Effort method, which is extensively adopted throughout the book, both as a modeling approach and as a methodology to design circuits for delay minimization. It is shown that, when designing digital circuits in the energy-delay space, Logical Effort method allows to derive practical design constraints.

Chapter 2 reports consideration on the energy consumption of digital circuits and the theory concerning their efficient design in the energy-delay space. The adoption of suitable figures of merit and the concept of energy-efficient curve are discussed, since they are exploited when deriving a novel optimization methodology that takes into account the energy-delay tradeoff.

Chapter 3 provides an overview of clocking and clocked storage elements operation and of the main parameters related to their timing and energy features. The main topological classes are presented, together with their basic properties.

A detailed and extensive design strategy for nanometer CMOS clocked storage elements through circuital optimization in the energy-delay space is presented in Chapter 4. The methodology accounts for the impact of parasitics due to interconnects, which strongly increases in nanometer technologies, and widely exploits the theories of Logical Effort and of energy-efficient design described in the first two chapters.

The results of a wide comparison among 19 clocked storage elements topologies, selected among the most representative and best known previously proposed ones, are reported in Chapter 5. Besides the exploration of the energy-delay one, several other tradeoffs involving leakage, area and clock load are investigated, thereby allowing to compare clocked storage elements in a more general framework.

Chapter 6 contains novel results concerning the optimization of clock distribution at the clock domain level. The energy-delay tradeoff is again examined by observing the joint performances of clock buffers and clocked storage elements when locally varying the clock slope.

Novel ultra-fast clocked storage elements topologies are presented in Chapter 7, together with measurements results extracted from a chip prototype in a 65-nm CMOS technology. These novel circuits belong to the Pulsed Latches class, which, from the analysis in Chapter 5, is recognized as the most promising one in nanometer technologies. Overall, the proposed topologies achieve the best speed and energy-efficiency performances in the high-speed energy-delay region that have ever been reported.

Finally, conclusions of the work are reported.

Chapter 1

THE LOGICAL EFFORT METHOD

This chapter describes the Logical Effort approach [SSH98], which represents the state of the art as concerns the modeling and optimization of CMOS digital circuits from the point of view of their speed performances. Such a methodology is often exploited throughout the following chapters, especially when dealing with the optimized clocked storage elements sizing and when searching for practical design space bounds.

1.1 An RC Model for the Delay of Logic Gates

When searching for a model of the delay of logic gates, it is necessary to recur to some simplifications that can allow the development of back-of-the-envelope though useful calculations. From their basic structure and except for some cases, it is evident that CMOS logic gates can be simply modeled as decoupled RC blocks [H84]. As shown in Fig. 1.1, each block consists of supply(V_{DD})-to-output and ground-to-output alternately activated resistive paths, corresponding to pull-up (PUN) and pull-down (PDN) networks, respectively, and the output is capacitively self-loaded and externally loaded. Whether the PUN or PDN is activated depends on the logic value stored at the output of the previous block, whose external load is the input capacitance of the gate corresponding to the considered block. Hence, once suitable R and C values are found, an effective model for the delay estimation of CMOS logic gate can be easily developed.

The equivalent resistance R of a MOS can be evaluated by averaging out the derivative $(\partial I_D/\partial V_{DS})^{-1}$ in the voltage range of interest. Anyhow, the most important consideration is that, independently from working in triode or saturation, the resistance of a MOS transistor is inversely proportional to

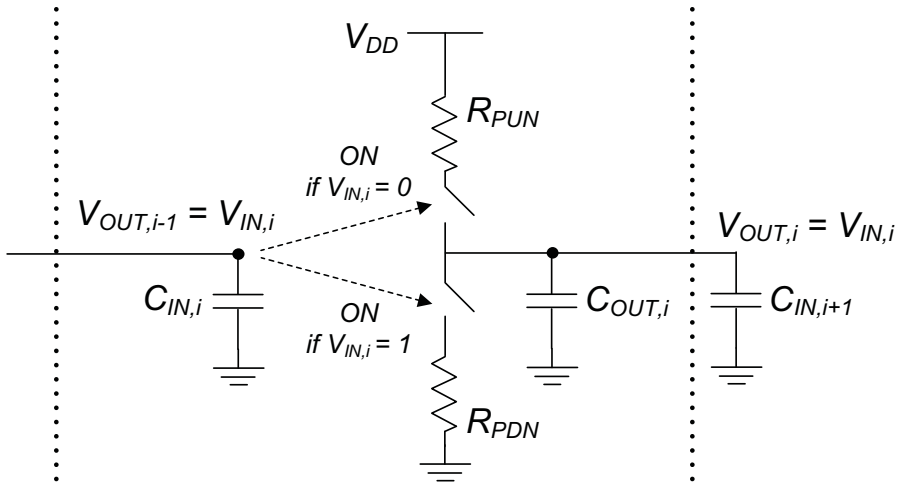


Fig. 1.1. CMOS logic gates seen as decoupled RC blocks.

its width W (for simplicity, by neglecting the impact of normal and reverse narrow width effects [CH99]). When considering complex CMOS gates, the evaluation of the total equivalent resistance of PUN and PDN can be approximately performed by summing the resistances of stacked blocks of transistors and by summing the conductances (which are proportional to W) of parallel blocks of transistors that are conducting the current at the same time [RCN03].

The equivalent capacitance at the input of a MOS transistor, C_G, can be evaluated by averaging out the sum of C_{GS} (gate-source), C_{GD} (gate-drain) and C_{GB} (gate-bulk) contributions in the voltage range of interest. The resulting value is proportional to WL and typically nearly equal to $C_{ox}WL$ (being C_{ox} the gate oxide capacitance per unit area) [RCN03].

The self-loading in a CMOS gate is due to the drain-bulk (and source-bulk in internal nodes for stacked transistors) diffusion capacitances. Such capacitances can be expressed as [OK06][1]

$$C_D = C_{D,A}WL_d + C_{D,P}(2W + 2L_d) \tag{1.1}$$

where L_d is the length of drain/source diffusions and $C_{D,A}$ ($C_{D,P}$) are the capacitances per unit area (perimeter) of drain-bulk and/or source-bulk

[1] Note that sometimes the sidewall capacitance is not counted for the side of diffusion adjacent to the channel [RCN03]. In this case, the second term of equation (1.1) becomes equal to $C_{D,P}(W + 2L_d)$.

junctions, evaluated by averaging out in the voltage range of interest[2]. By neglecting the $2L_d C_{D,P}$ term, C_D can be considered nearly proportional to W.

Summarizing, by considering

a) a CMOS gate with all stacked transistors of the same size;

b) only one conductive branch among the existing ones;

c) a constant ratio between the size of PMOS and NMOS;

one has that

$$C_{IN} \propto WL \tag{1.2a}$$
$$C_{OUT} \propto W \tag{1.2b}$$
$$R_T \propto L/W \tag{1.2c}$$

where C_{IN} is the input capacitance of the gate terminal where the critical input is applied, C_{OUT} is the total diffusion capacitance at the output (also including contributions from internal nodes) and R_T is the resistance of PUN or PDN.

Usually, the channel lengths are all minimum, and the considered gate can be seen as a version scaled by a factor α (in terms of channel width) of a reference gate of the same type, called the "template" gate (typically considered of minimum size). Such a gate exhibits parameters $C_{IN,ref}$, $C_{OUT,ref}$ and $R_{T,ref}$ and the following relationships hold [OK06]

$$C_{IN} = \alpha C_{IN,ref} \tag{1.3a}$$
$$C_{OUT} = \alpha C_{OUT,ref} \tag{1.3b}$$
$$R_T = R_{T,ref}/\alpha \tag{1.3c}$$

Hence, any timing parameter of the considered gate can be expressed as [H84]

$$t_D = KR_T(C_{OUT} + C_L) = K\left(R_{T,ref}C_{IN,ref}\frac{C_L}{C_{IN}} + R_{T,ref}C_{OUT,ref}\right) \tag{1.4}$$

---

[2] By considering the large signal behavior of reverse-biased junction capacitances, $C_{D,A}$ can be equaled to $C_{j0}K_j$, being C_{j0} the value under zero bias condition, and

$$K_j = \frac{\phi^m}{V_2 - V_1}\left[\frac{(\phi - V_1)^{1-m}}{1-m} - \frac{(\phi - V_2)^{1-m}}{1-m}\right]$$

where ϕ is the built-in potential across the junction, m is the grading coefficient of the junction, and V_1 and V_2 are the minimum and maximum direct voltages across the junction, respectively [RCN03].

$C_{D,P}$ is equal to $C_{D,A}x_j$, being x_j the depth of the diffusion.

where C_L is the external output load and K depends on the kind of timing parameter (delay, fall/rise times) and on the slope of the input. For instance, when considering the propagation delay under a step input, $K = 0.69$.

An equivalent model can be derived considering the current provided by the gate instead of equivalent resistances (see the short Appendix at the end of this chapter).

It is worth noting that especially the evaluation of equivalent resistances requires several approximations to manage the various effects arising in deep-submicron technologies and influencing the I-V behavior of MOS transistors. Some of these effects are mobility degradation, carriers velocity saturation, channel length modulation, drain-induced barrier lowering (DIBL), short-channel and narrow-width effects (e.g., V_{TH} roll-off) and so on (see [T03] and [TN09] for a thorough discussion).

For instance, according to a well-known short-channel model [TKM88], the classic inversely proportional dependence of MOS current from channel length L is damped because of velocity saturation, i.e.

$$I_D \propto \begin{cases} \frac{1}{V_{DS}+LE_{cr}} & in\ triode\ region \\ \frac{1}{(V_{GS}-V_{TH})+LE_{cr}} & in\ saturation\ region \end{cases} \quad (1.5)$$

where V_{TH} is the threshold voltage and E_{cr} is the electrical field for which velocity saturation is observed. Meanwhile, due to short-channel effects, V_{TH} decreases when lowering L [Y74]. Hence, on the whole, an $R \propto L$ approximation is still feasible.

Another example is given by the case of n stacked MOS transistors, which classically exhibit a total equivalent resistance equal to nR, being R that of a single transistor. When considering velocity saturation, this effect again grows fainter since transistors enter current saturation for smaller voltages [SN91]. Meanwhile, channel length modulation and DIBL effects have a severe impact and increase the dependence of saturation current from V_{DS}.

1.2 The Logical Effort Model

The RC model in (1.4) was revisited in [SSH98] to obtain a new one normalized to (i.e., independent from) technology: the Logical Effort model. Basically, formula (1.4) is divided by $R_{INV}C_{INV}$, which is the product of the equivalent resistance and input capacitance of a symmetrical inverter, i.e. an inverter showing symmetric PUN and PDN driving capabilities (in current technologies, this is typically obtained by sizing the PMOS twice the size of

the NMOS). Note that, even if the absolute size of this inverter is varied, the product $R_{INV}C_{INV}$ is a constant dependent on technology.

Once normalized, the timing parameter of the considered gate, t_D, (which in the following is referred as delay without loss of generality) becomes

$$t_D = \tau(gh + p) \quad (1.6a)$$
$$t_D = \tau(f + p) = \tau d \quad (1.6b)$$

where the various quantities correspond to

$$\tau = KR_{INV}C_{INV} \quad (1.7)$$
$$g = \frac{R_{T,ref}C_{IN,ref}}{R_{INV}C_{INV}} \quad (1.8)$$
$$h = \frac{C_L}{C_{IN}} \quad (1.9)$$
$$p = \frac{R_{T,ref}C_{OUT,ref}}{R_{INV}C_{INV}} \quad (1.10)$$

The parameter τ allows to normalize the absolute delay t_D to technology and it represents the delay of symmetrical inverter loaded with an inverter of the same size and neglecting the self-loading due to diffusion capacitances.

The parameter g is called "Logical effort" and, except for some cases, is a feature dependent on the gate's topology and hence not affected by the "absolute" sizing of the gate but only by its "relative" sizing (by definition, $g = 1$ for a symmetrical inverter).

The logical effort g describes the driving capability of the gate topology and has a twofold interpretation:

1) under the assumption of equal C_{IN}, g indicates how much worse is the driving capability of the considered gate with respect to that of a symmetrical inverter;

2) under the assumption of equal driving capability, g indicates how much larger the considered gate has to be (in terms of C_{IN}) with respect to a symmetrical inverter.

The parameter h is called "Electrical effort" and it is equal to the fanout of the gate. It is independent from the topological characteristics of the gate, is affected only by the absolute gate sizing (i.e., it depends on α) and affects the normalized delay d as much as g. Obviously it increases for high C_L (heavier load) and decreases for high C_{IN} (larger driving capability).

The parameter p is called "Parasitic delay" and represents the intrinsic and unavoidable delay contribution due to the self-loading of the gate. As for g, except for some cases, p is a feature dependent on the gate's topology and hence it is not affected by the "absolute" sizing of the gate but only by its "relative" sizing. Indeed when enlarging the gate size to improve its driving

capability, also the capacitance C_{OUT} increases proportionally. In the case of an inverter, p is close to 1, since typically $C_G \approx C_D$.

Finally, the parameters f (equal to gh) and d are named "Stage effort" and "Normalized delay", respectively.

It is apparent that the normalized delay d is a linear function of h, as shown in Fig. 1.2. The logical effort g represents the slope of such a line, whereas the parasitic delay p is the minimum achievable delay extrapolated for $h = 0$, i.e. for zero external load or for $C_{IN} \gg C_L$.

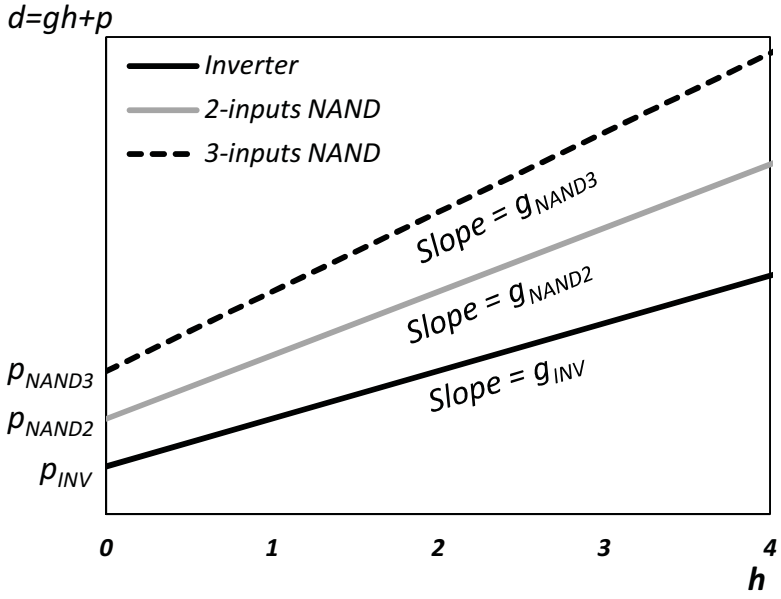


Fig. 1.2. Geometrical interpretation of logical effort and parasitic delay.

1.3 Limitations of the Original Logical Effort Model

The model described so far recurs to several simplifications and suffers from some limitations, which, however, appears necessary when trying to develop back-of-the-envelope calculations.

The model assumes MOS transistors behaving as equivalent resistances. However, for the major part of the transient state in the case of rise/fall times and always in the case of delays, MOS transistors behave as non ideal current generators. Such a phenomenon arises in deep submicron technologies, since the drain-to-source saturation voltage decreases with respect to the classic behavior [RCN03]. Anyhow, the results maintain their validity since the dependence from channel widths and lengths is basically

the same for the current in saturation region and for the conductance (inverse of resistance) in triode region (see the Appendix at the end of the chapter).

Being an RC model, the Logical Effort predicts a perfect exponential behavior of the output waveforms. However the total resistances of PUN and PDN vary with the output voltage. Anyhow, the general character of actual output waveforms is preserved when approximating with exponentials.

Both the self- and external loads vary with the output voltage in a complicated nonlinear manner. Again, the general character of actual output waveforms is preserved when adopting averaged constant load capacitances.

The delay and rise/fall times of CMOS gates both significantly depend on the input transition time (or slope) [HJ87], which is neglected in the Logical Effort model. As concerns rise/fall times, they always increase in a nonlinear manner for slower input transitions [DML95]. As concerns delays, they can both increase and decrease in a non monotonic fashion with slower input transitions [DA99]. Several attempts have been made to develop Logical Effort extensions in order to capture the effect of a non-zero input transition time, although they have resulted in quite complicated and non practical models, or in simpler ones whose applicability is however restricted only to some cases [LEW06], [HJR09], [WM09].

However, although the accuracy of the original Logical Effort approach as a delay model is somewhat weakened by this lack, in the following it is shown that Logical Effort is primarily used as an optimization method for minimizing delays through the equalization of the external load-dependent part of the delay d, i.e. gh. This approach leads to somewhat constant input and output slopes for CMOS gates in a path, and, under this condition, the original Logical Effort model is quite accurate. Moreover, the usefulness of Logical Effort method to size CMOS gates for maximizing speed is not invalidated by the typical accuracy of the underlying model.

The model in (1.4) and (1.6) deals with the self-loading effect through a single capacitance C_{OUT} that is placed in parallel to the external output load and charged/discharged through the single resistance R_T. In other words, only the capacitances insisting on the output nodes are taken into account. However, when the PUN and/or PDN are made up by stacked (blocks of) transistors, the capacitances in their internal nodes can give a further contribution to the parasitic delay (see Fig. 1.3a). In particular, this happens when the critical inputs is not applied to the transistors closest to the output node, as shown in Fig. 1.3. Thus implying that not all the internal capacitances are already charged/discharged [SN91].

A simple though inaccurate solution can be that of simply transferring to the output node all the internal nodes capacitances to be charged/discharged when the critical input arrives. A better approach is that of modeling each stacked group of transistors as an equivalent resistance and recurring to the Elmore delay model [E48].

The Elmore delay model allows to find an estimate of the delay introduced by an RC tree (Fig. 1.3b) and its insertion in the Logical Effort background is straightforward [MFG10]. In the following, it is shown that, when using the Logical Effort as an optimization method, parasitic delays (which are generally constant even when adopting more complex approaches such as the Elmore delay one) do not enter in the calculations. Nevertheless, an accurate estimation of parasitic delays can be necessary in other situations.

As mentioned in the first paragraph, the total resistance of n stacked transistors can be only roughly approximated to n times that of single transistor. Indeed, there exist many concurrent phenomena having different effects (carriers velocity saturation, body biasing, DIBL and channel length modulation). Therefore, due to their complexity, an accurate estimation of the total equivalent resistance (and hence of g and p) in stacked structures with different values of n should be addressed through simulations.

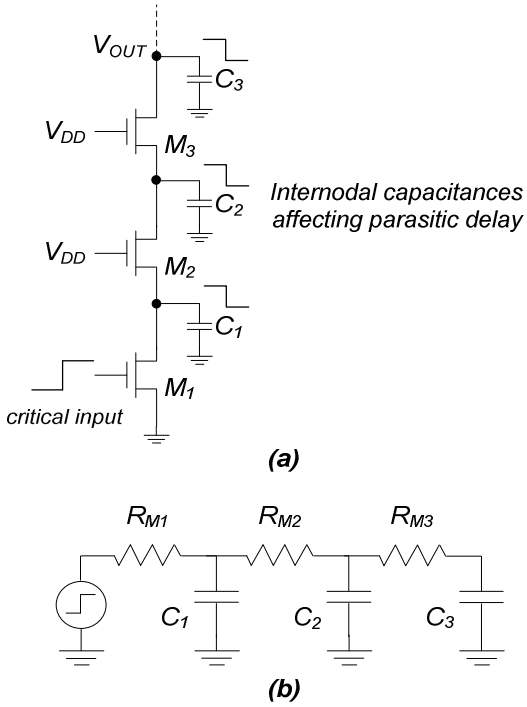


Fig. 1.3. Application of the Elmore delay theory to deal with stacked transistors and internodal capacitances (a) through an equivalent RC tree (b).

1.4 Basic Estimation of Logical Effort Parameters

As a necessary premise, in the rest of the work transistors widths and lengths are normalized with respect to the minimum values allowed by technology, called W_{min} and L_{min}. In particular, such normalized values are referred to as $w = W/W_{min}$ and $l = L/L_{min}$, being W and L the absolute values. Analogously, the absolute capacitances and equivalent resistances, C and R, are normalized to the values obtained for $W = W_{min}$ and $L = L_{min}$ (e.g., gate capacitances C_G are normalized to $C_{ox}W_{min}L_{min}$) and referred with lower case letters c and r, respectively.

The first step to carry out an evaluation of Logical Effort parameters (g, h and p) is to determine r_{INV} and c_{INV} (i.e., the normalized R_{INV} and C_{INV}).

As mentioned before, the product $r_{INV}c_{INV}$ remains constant whichever the absolute sizing of the reference inverter is, provided that such an inverter is symmetrical. In deep submicron technologies, the symmetry of an inverter is typically achieved by sizing the PMOS transistor 2 times larger than the NMOS transistor. For instance, one can refer to the so-called *minimum symmetrical inverter*, i.e. to an inverter with $w_P = 2$, $l_P = 1$, $w_N = 1$ and $l_N = 1$. Hence, such an inverter has an input capacitance $c_{INV} = 3$, while its resistance (equal for PMOS and NMOS) can be assumed as the reference unitary resistance, i.e. $r_{INV} = 1$, thus an overall product $r_{INV}c_{INV}$ equal to 3. Note that, if for instance one had considered a non minimum inverter with w_P and w_N equal to 8 and 4, respectively (both minimum length transistors), one would have found $c_{INV} = 12$ and remembering (1.3c) $r_{INV} = 1/4$, i.e. again $r_{INV}c_{INV} = 3$.

When defining g and p in (1.8) and (1.10), we resorted to the products $r_{T,ref}c_{IN,ref}$ and $r_{T,ref}c_{OUT,ref}$. As for the symmetrical inverter, for each CMOS static gate such products (and hence also g and p) remain constant whichever the absolute size is, but provided that the gate maintains its *skew*.

The skew of a CMOS static gate is defined as the ratio among the driving capabilities of PUN and PDN when applying the critical input (i.e. that leading to the output transition). By definition, a symmetrical gate has unitary skew, whereas gates with unbalanced rise/fall transitions has a skew greater or smaller than unit.

To obtain any desired skew, one has to properly size stacked and/or parallel (groups of) transistors. So far it has been pointed out that parallel transistors that are contemporarily turned on behave like parallel resistances, and the resistance of n stacked transistors can be roughly approximated to n times that of a single transistor.

By way of example, if one wants to make a gate symmetrical and under the usual assumption of considering only one branch conducting among the

possible parallel ones [RCN03], the PMOS transistors have to be sized $k = \mu(MP/MN)$ times the size of the NMOS, where:

- MN (MP) is the number of stacked transistors in the PDN (PUN), hence the ratio MP/MN compensate for the different weight of stacking in PUN and PDN;
- μ is the above mentioned factor equal to 2 that compensates for the different mobilities of electron and holes.

Under these assumptions, the sizing strategies leading to $S = 1$ skew are shown in Fig. 1.4 for Inverter and up to 3 inputs NAND and NOR gates (note that, as usually done in practice, there is no difference in the size of stacked transistors).

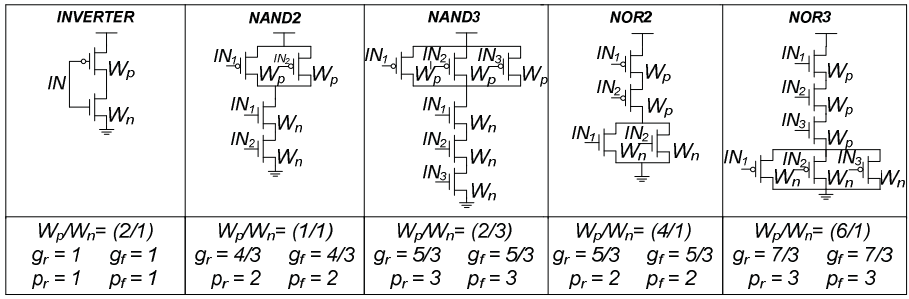


Fig. 1.4. Sizing of basic gates under unitary skew conditions.

Parameters g and p can be simply estimated by visual inspection of the capacitances (c_{IN} and c_{OUT}) and resistances (r_T) of the gate for any sizing leading to a certain skew (recall that g and p remain constant under the same skew). In particular:

$$c_{IN} = w_{N,i} l_{N,i} + w_{P,i} l_{N,i} \quad (1.11)$$

$$c_{OUT} = \sum_j w_{N,j} + \sum_k w_{P,k} \quad (1.12)$$

$$r_{T,N} = \sum_{m=1}^{MN} \frac{l_{N,m}}{w_{N,m}} \quad (1.13)$$

$$r_{T,P} = \sum_{m=1}^{MP} \frac{l_{P,m}}{w_{P,m}} \quad (1.14)$$

where:

- $w_{N,i}$ ($w_{P,i}$) and $l_{N,i}$ ($l_{P,i}$) are the width and length of the NMOS (PMOS) transistor driven by the i-th input of the gate, which is the one considered for delay estimation;
- $w_{N,j}$ ($w_{P,j}$) are the widths of all NMOS (PMOS) transistors contributing to the self-loading;
- MN (MP) is the number of stacked transistors in the PDN (PUN);

- $w_{N,m}$ ($w_{P,m}$) and $l_{N,m}$ ($l_{P,m}$) are the width and length of the m-th transistor in the PDN (PUN) stack.

Parallel transistors are neglected in this computation. Only when some of them are contemporarily conducting, their effect can be included by summing the correspondent conductances, i.e. the ratios w/l. Moreover, as previously stated when discussing the properties of an inverter, remember that relationship (1.12) assumes the transistor diffusion drain-bulk capacitance nearly equal to the gate input capacitance. Actually, the validity of this assumption strongly depends on technology and layout features [WH04]. Moreover, for the moment only the capacitances that are physically attached to the output node are considered, as in the original Logical Effort model (more accurate extensions are given in the following).

Once c_{IN}, c_{OUT}, $r_{T,N}$ and $r_{T,P}$ are computed for the specific input, g and p can be estimated from (1.8) and (1.10), i.e. dividing $c_{IN}r_{T,N}$ ($c_{IN}r_{T,P}$) and $c_{OUT}r_{T,N}$ ($c_{OUT}r_{T,P}$) by $r_{INV}c_{INV} = 3$.

It is apparent that, except for the cases of unitary skew (shown in Fig. 1.4), g and p are different for the falling and rising transitions. Hence we define g_f and p_f as the parameters referring to the falling transition and g_r and p_r as those referring to the rising transition. The values of g and p for the rising and falling transitions of the basic gates depicted in Fig. 1.4 are reported in Tab. I.I for generic inputs number, M, and skew, S.

Moreover, it is worth noting that, while p is constant for any input of the gate (at least when considering only the parasitic capacitances attached to the output node), g can vary according to the considered input because of the possible different input capacitance seen at each input. This happens when considering gates that employ combined stacked/parallel group of transistors and where NMOS (PMOS) are not equally sized as shown in Fig. 1.5[3].

Finally, the electrical effort h is simply estimated by transforming the external load C_L into a normalized equivalent width. This is easily done by visual inspection when the load is constituted by another CMOS gate. When, for some reason, one has to refer to an absolute capacitive value expressed in Farad, it can be again transformed into an equivalent normalized width, w_L, dividing it by the capacitance seen at the input of a single minimum transistor. The latter one is equal to one third of that at the input of a symmetrical minimum inverter.

[3] When considering basic gates such as NAND or NOR, the resistance exhibited by pull-up an pull-down network is nearly the same whichever is the applied critical input. On the contrary, for more complex gates it is not possible to size the pull-down (pull-up) networks so that they exhibit the same resistance for all input combinations. Therefore, the usual approach [RCN03] is to consider a worst-case where it is considered that only one among various parallel groups of transistors is conducting, as done for the sizing of the gate in Fig. 1.5.

TABLE I.I: LOGICAL EFFORT AND PARASITIC DELAY FOR NAND AND NOR GATES HAVING M INPUTS AND S RISE/FALL SKEW

M inputs S skew	***Inverter***	***M-inputs NAND***	***M-inputs NOR***
W_p/W_n	$2S$	$\frac{2S}{M}$	$2SM$
g_r	$\frac{2}{3}\left(1+\frac{1}{2S}\right)$	$\frac{2}{3}\left(1+\frac{M}{2S}\right)$	$\frac{2}{3}\left(M+\frac{1}{2S}\right)$
g_f	$\frac{1}{3}(1+2S)$	$\frac{1}{3}(M+2S)$	$\frac{1}{3}(1+2SM)$
p_r	$\frac{2}{3}\left(1+\frac{1}{2S}\right)$	$\frac{2}{3}\left(M+\frac{M}{2S}\right)$	$\frac{2}{3}\left(M+\frac{M}{2S}\right)$
p_f	$\frac{1}{3}(1+2S)$	$\frac{1}{3}(M+2SM)$	$\frac{1}{3}(M+2SM)$

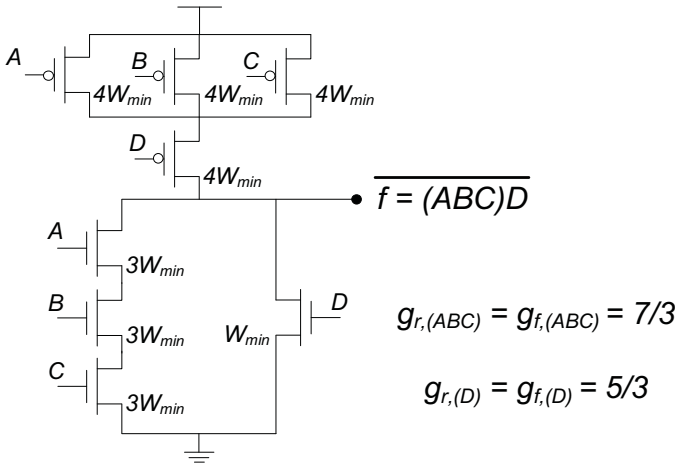


Fig. 1.5. Complex gate sized to exhibit $S = 1$ skew and delay equal to a minimum inverter: differences in g values according to the considered input.

1.5 Accurate Estimation of Parameters g and p

So far some simple strategies to estimate the logical effort and the parasitic delay have been discussed. The following more accurate approaches are required if one wants to obtain better quantitative delay estimations when dealing with stacked structures.

1.5.1 Internal nodes capacitances

The value of C_D capacitances depend on the layout features. Drain-bulk and source-bulk capacitances can correspond both to shared or unshared and contacted or uncontacted nodes [WH04]. For instance, stacked transistors

can share their diffusions thereby reducing parasitic diffusion capacitances. Moreover, also the gate-source and gate-drain capacitances of the transistors above that driven by the latest input contribute to the internodal capacitances, as shown in Fig. 1.6.

An accurate estimation of internodal capacitances is required when applying the Elmore delay model (as shown in the subsequent point) and can be based on the following rules:

- Diffusions shared among two transistors must be accounted only once;
- Contacted diffusions exhibit a capacitance that is actually nearly equal to the correspondent gate capacitance of the same transistor [WH04]. Their normalized value is hence equal to w_i, being w_i the width of the correspondent transistor(s);
- Uncontacted diffusions exhibit a lower capacitance. Their value is however still proportional to w_i, but according to a factor $\alpha < 1$;
- Gate source and gate-drain capacitances of transistors in the stack that are above (for a PDN) the transistor driven by the latest input have to be included. Since being relative to a turned on transistor, their sum is nearly equal to $C_{ox}WL$, i.e. to w_i in a normalized fashion. A good approximation is to split this value into half and assign a capacitance $w_i/2$ both to the nodes above and below the considered transistor.

Summarizing, the self loading effect in the internal nodes can be accounted for by still obtaining capacitive values that are proportional to w_i.

1.5.2 Elmore delay

As was introduce in Fig. 1.3, the parasitic delay of stacked structures can be more accurately expressed by exploiting the Elmore delay model [E48], which allows to estimate the delay due to an RC tree from a source node (where the voltage is applied) to a node k. Elmore delay consists on estimating the time constant

$$T_{E,k} = \sum_{i=1}^{N} C_i R_{ik} \tag{1.15}$$

where C_i is the i-th capacitance in the RC tree and R_{ik} is the total resistance shared by the paths between the source node and nodes i and k [E48].

It is apparent that stacked transistors can be approximated by an even simpler RC ladder structure (which is a particular case of an RC tree) and the source voltage is given by V_{DD} or ground nodes.

The resistances R_{ik} are the sums of stacked transistors resistances, and the capacitances C_i are the internodal capacitances.

Obviously, only the capacitances that are not already charged or discharged (i.e., for a PDN, only those in the nodes above the transistor driven by the latest input) have to be considered. Hence, it is easy to note that the worst-case parasitic delay occurs when the latest input drives the transistors closest to V_{DD} or ground nodes.

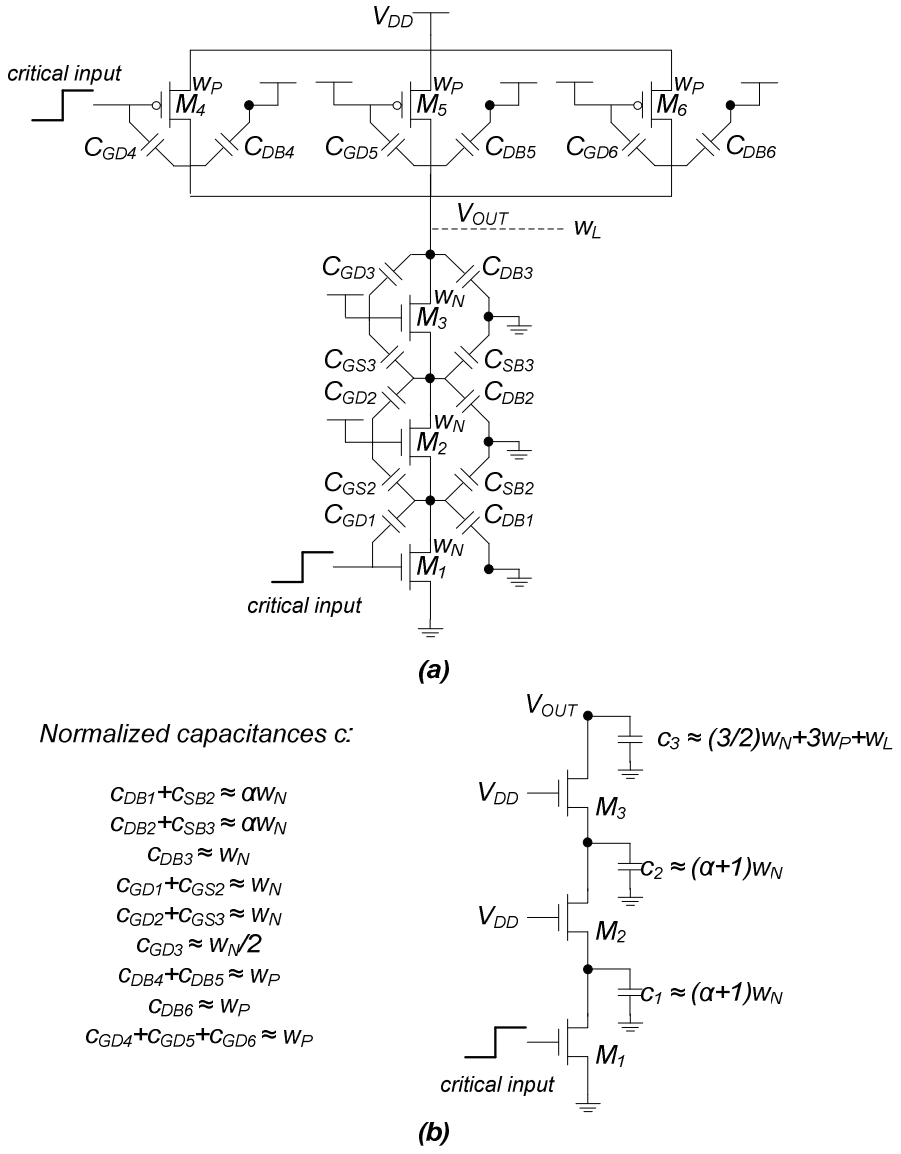


Fig. 1.6. Internodal capacitances in a NAND3 (a) and accurate estimation of their normalized values as functions of sizing (b).

By way of example, the Elmore delay is applied to estimate the worst-case falling delay of the 3-inputs, symmetrical (i.e., $w_N = 1.5w_P$), minimum lengths NAND gate, shown in Fig. 1.6a. The normalized Elmore time constant is equal to

$$d_{E,out} = \frac{c_3(r_1+r_2+r_3)+c_2(r_2+r_3)+c_1 r_3}{r_{INV}c_{INV}} \tag{1.16}$$

where (see Fig. 1.6)

$$c_3 = w_L + \frac{3}{2}w_N + 3w_P = w_L + \frac{7}{2}w_N \tag{1.17}$$
$$c_1 = c_2 = (\alpha + 1)w_N \tag{1.18}$$
$$r_1 = r_2 = r_3 = \frac{1}{w_N} \tag{1.19}$$

Note that the approximations described in subsection 1.5.1 have been applied. In particular, only two drain diffusions capacitances are considered for PMOS transistors, since two of them can share the same one. Moreover, the sum of the three gate-drain capacitances of the turned-off PMOS transistors is equaled to w_P since they are all small overlap capacitances.

By using (1.17)-(1.19), the delay (1.16) can be rewritten as

$$d_{E,out} = \frac{3w_L+\frac{21}{2}w_N+3(\alpha+1)w_N}{3w_N} = \frac{w_L}{w_N} + \left(\frac{9}{2} + \alpha\right) \tag{1.20}$$

that is the sum of a term dependent only on the external load, i.e. the classical gh term, and another one due to self-loading that is a more accurate parasitic delay estimation for this worst-case scenario.

It is worth noting that the resulting parasitic delay is slightly higher than that resulting with the traditional LE, equal to 3 for the 3-input NAND.

1.5.3 Parameters calibration

A third issue concerns the actual overall resistance exhibited by stacked transistor, which influences both g and p. Even if DIBL and channel length modulation effects somewhat compensate for velocity saturation, the nR approximation is not precise. Anyhow, given the complexity of the problem, no easy calculations can be carried out and the best way to estimate the actual value of g and p is through simulations.

Remembering that g is the slope and p is the intercept of the linear relationship between delay and electrical effort, the procedure to estimate them simply consists in evaluating the delay of the considered gate for increasing h values and normalizing with respect to the technology parameter τ.

Proper precautions have to be taken in order to provide the gate with realistic input waveforms and to avoid an unrealistic amplification of the C_{GD} external loading capacitances due to Miller effect [SSH98]. The first objective is achieved by driving the gate under test with the signal provided by two gates of the same type, while the second is achieved by imposing a sufficiently large load on the loading gate. Fig. 1.7 illustrates such a testbench to extract parameters g and p through simulations [WH04].

By extracting g and p in this way, for instance one finds that, in a 65-nm CMOS technology the ratio $R_{T,ref}/R_{INV}$ is equal to 1.85, 1.6, 2.7, 2.3 for minimum symmetrical NAND2, NOR2, NAND3 and NOR3 gates, respectively. This shows that, although DIBL and channel length modulation effect have a strong impact in nanometer technologies, the effect of velocity saturation is higher[4].

It is worth noting that the above calibration procedure allows to estimate the values of g and p by including the effect of non-zero input rise/fall times, although under the restriction that the output rise/fall times are equal to the input ones. Obviously, in real circuits this is not generally true. But, as anticipated in Section 1.3 and shown in the following one, this is nearly the case when using the Logical Effort as a method to minimize delay.

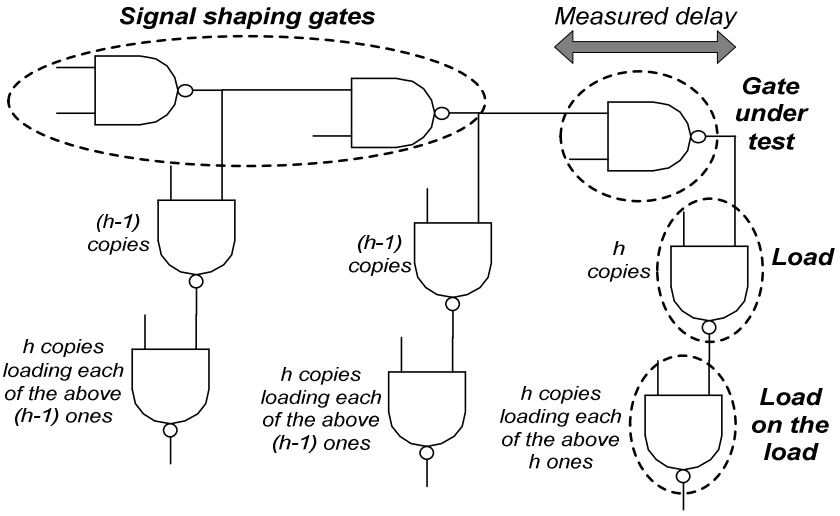


Fig. 1.7. Simulations testbench to extract g and p (h is the electrical effort).

[4] In the case of long-channel devices, NAND2/NOR2 and NAND3/NOR3 show a ratio $R_{T,ref}/R_{INV}$ equal to 2 and 3, respectively. The ratio increases because of DIBL and channel length modulation, whereas decreases due to velocity saturation.

1.5.4 Non-step input

Although it was previously stated that the attempts to model the delay variations due to input slope often result in complex models or in models having no general validity, a simple approach to deal with non-step input is that of characterizing the gates delay according to the following model [Z06]

$$d = gh + p + \eta d_{in} \tag{1.21}$$

where d_{in} is the normalized input rise/fall time (typically extracted by interpolating the points at $0.2V_{DD}$ and $0.8V_{DD}$), g and p are the logical effort and parasitic delays extracted under a step-input and η is an additional parameter which accounts for the impact of d_{in} in a linear way.

Actually, the dependence of delay (or rise/fall times) on the input rise/fall times is nonlinear. However, given that, to improve robustness, circuits are typically designed to avoid excessively slow transitions in their internal nodes, a linear approximation is feasible.

Nevertheless, the parameter η needs to be re-extracted when changing the skew featuring the gate, since, according to the relative strength between pull-up and pull-down networks, the behavior of the delay when increasing d_{in} can significantly change. For instance, the asymptotic delay for $d_{in} \to \infty$ can be a line with negative or positive slope according to the DC behavior of the gate, which obviously depends on skew [DML95]. This in turns has an impact on the d vs. d_{in} dependence also for small and moderate d_{in} values.

1.6 Multistage Logic Networks and Delay Minimization

1.6.1 Path parameters

In the following let us consider a multistage network comprising a path made up of N cascaded logic gates, the i-th of which is featured by a logical effort g_i, a parasitic delay p_i and an electrical effort

$$h_i = \frac{C_{L,i}}{C_{IN,i}} = \frac{C_{IN,i+1} + C_{off,i}}{C_{IN,i}} \tag{1.22}$$

where $C_{IN,i}$ and $C_{IN,i+1}$ are the input capacitances of the i-th and $(i+1)$-th gate in the considered path, while $C_{off,i}$ is the input capacitance of other gates loading the stage i but not belonging to the path under analysis (see Fig. 1.8).

It is convenient to manage and use the LE also on a path, at this purpose, let us define the "Path logical effort", G, the "Path parasitic delay", P, and the "Path electrical effort", H, as

$$G = \prod_{i=1}^{N} g_i \tag{1.23}$$
$$P = \sum_{i=1}^{N} p_i \tag{1.24}$$
$$H = \frac{C_{L,N}}{C_{IN,1}} \tag{1.25}$$

respectively, being $C_{L,N}$ and $C_{IN,1}$ in (1.25) the final load of the path and the input capacitance of the first stage, respectively.

By defining the "Branching effort" b_i of the i-th stage as the proportion between the total load of gate i and the fraction lying on the considered path,

$$b_i = \frac{C_{IN,i+1} + C_{off,i}}{C_{IN,i+1}} \geq 1 \tag{1.26}$$

one can also introduce the "Path branching effort", B, of the entire path through the following formulas

$$B = \prod_{i=1}^{N} b_i \tag{1.27}$$

whose product with the path electrical effort, H, results in the electrical effort product of the gates in the path (equal to H only when there are not branch in path)

$$HB = \prod_{i=1}^{N} h_i \tag{1.28}$$

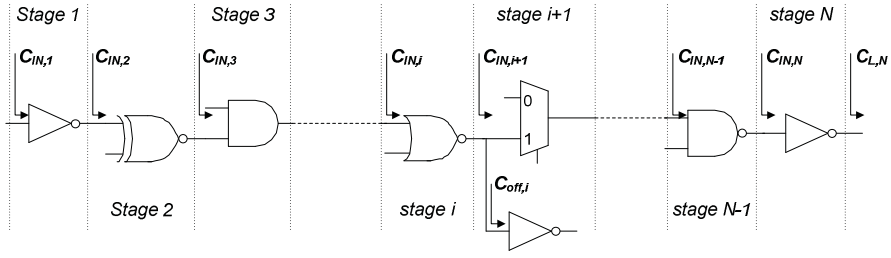


Fig. 1.8. Multistage path.

Finally, the "path effort" F is equal to

$$F = \prod_{i=1}^{N} g_i h_i = \prod_{i=1}^{N} f_i = GBH \tag{1.29}$$

and the total normalized delay of the considered path results

$$D = \sum_{i=1}^{N}(g_i h_i + p_i) \tag{1.30}$$

By inspection of (1.30) and considering that not only g_i and p_i, but also b_i, are constant parameters (although at the end of the chapter it is shown that this is not in general true), one has that D is a function only of the capacitive gains of the various stages on the path.

1.6.2 Optimized design

As previously anticipated, the Logical Effort representation model can serve also to develop an optimization method to minimize delay. In particular, considering that

$$h_1 = \frac{H}{h_2 h_3 \cdots h_N} \tag{1.31}$$

and assuming one knows *H*, the freedom degree of relationship (1.30) is reduced by one.

By using (1.31) into (1.30) and minimizing it allow to find the minimum path delay. Thus considering (1.30) function of only the electrical effort h_i for *i* from 2 to N, the minimum is find solving the below *N*-1 equations

$$\frac{\partial D}{\partial h_i} = \frac{\partial\left(g_1 \frac{H}{h_2 h_3 \cdots h_N} + \sum_{i=2}^{N}(g_i h_i + p_i)\right)}{\partial h_i} = g_i - \frac{g_1 H}{h_1 (h_2 h_3 \cdots h_N)} = 0 \tag{1.32}$$

whose solution gives

$$g_1 h_1 = g_i h_i \qquad \forall i \tag{1.33}$$

From (1.33) it is apparent that the stage effort has to be the same for all stages in the path. Moreover, according to (1.29), the optimum stage effort (i.e., the optimum value of $g_i h_i$) is equal to

$$f_{opt} = \sqrt[N]{GBH} \tag{1.34}$$

Note that as previously anticipated, parasitic delays do not enter in the optimization final result.

Considering that the final load and the input capacitance of the first stage are known, the minimum achievable delay of a path with fixed topology and stages number *N* is known a priori, since, from (1.30) and (1.34) it is equal to

$$D_{opt} = N\sqrt[N]{GBH} + P \tag{1.35}$$

where G, B and H have fixed value independently from the absolute sizing of the various stages (this is actually true only if parameters g_i, b_i and p_i can be assumed as constant).

As show above, by using the logical effort delay model one can design the path minimizing the delay. Indeed, in order to satisfy the conditions (1.32)-(1.34), it is sufficient to set

$$f_i = \sqrt[N]{GHB} \qquad \forall i \tag{1.36a}$$

$$h_i = \frac{\sqrt[N]{GHB}}{g_i} \qquad \forall i \tag{1.36b}$$

$$C_{IN,i} = \frac{g_i b_i C_{IN,i+1}}{\sqrt[N]{GHB}} \qquad \forall i \tag{1.36c}$$

which are a set of relationships that can be applied by starting from the N-th gate ($C_{L,N}$ is known) and proceeding backward along the path, or starting from the first gate ($C_{IN,1}$ is known) and proceeding onward along the path.

Since thanks to the logical effort delay model it has been achieved an optimized design procedure which minimize the path delay, this optimized procedure has been named "Logical effort" method.

It is worth noting that, according to the above considerations and by neglecting the contribution of parasitic delays, the minimum overall path delay is reached when all the gates in the path exhibit similar speed, i.e. when the input and output rise/fall times along the path are similar. Under this condition, the parameters g and p extracted as shown in Paragraph 1.5 are quite accurate since they account also for the impact of finite input slope.

1.7 Optimum Number of Stages

So far it has been discussed on how to size a path with fixed topology to minimize its delay which need equaling the various stage efforts. Actually, it is possible to consider the number of stage as a further degree of freedom for delay minimization.

Indeed, the path effort F does not change when introducing any number of inverters to the considered path (since the additional inverters are featured by $g = b = 1$ and $H = C_{L,N}/C_{IN,1}$ does not change).

Starting from an initial number of stages n_1 required to perform the logic operations, one can add n_2 inverter so that the total number of stages is now equal to $N = n_1 + n_2$.

By applying the delay minimization procedure described in the previous paragraph, the minimum delay of the path is

$$D_{opt} = N\sqrt[N]{F} + \sum_{i=1}^{N} p_i + (N - n_1)p_{INV} \tag{1.37}$$

where p_{INV} is the actual parasitic delay of an inverter (so far supposed to be equal to 1).

The best number of stages, N_b, can be calculated by setting the derivative of (1.37) to zero, i.e.

$$\frac{\partial D_{opt}}{\partial N} = -\sqrt[N]{F}\ln\left(\sqrt[N]{F}\right) + \sqrt[N]{F} + p_{INV} = 0 \tag{1.38}$$

Which is a non linear function of *N*.

A solution of (1.38) can be found in terms of the best stage effort which is equal to

$$\rho = \sqrt[N_b]{F} \tag{1.39}$$

Thus equation (1.38) can be written

$$\rho(1 - \ln\rho) + p_{INV} = 0 \tag{1.40}$$

The solution of (1.40) can be found by graphic inspection, as shown in Fig. 1.9. It is apparent that the best stage effort increases with p_{INV} since, intuitively, when the parasitic delay introduce by inverters increases it is no more convenient to add many of them.

In the typical case when $p_{INV} \approx 1$, it results $\rho \approx 3.59$, which tells us that the best stage effort to optimize circuits speed is close to 4. This was the main reason for the widespread adoption of the $FO4$ inverter delay metric [HHW97].

The $FO4$ delay is defined as the delay of an inverter loaded by four inverter of its same size. From the Logical Effort point of view, the $FO4$ delay corresponds to a technology normalized delay equal to 5 ($d = h + 1$).

Moreover, it is worth noting that, under the unrealistic assumption of negligible parasitic delays, the Logical Effort theory provides the classical result relative to tapered buffers sizing, since, if $p_{INV} = 0$, it results $\rho = e$ [MC79]. Thus, in general, ρ can be assumed in the range 2-4.

It is worth noting that with respect to the starting topology with a desired number of stages required to perform the desired logic function, it can be necessary to increase the number of stages to achieve the best delay. Such an optimum number of stages is equal to

$$N_b = \log_\rho(GBH) \tag{1.41}$$

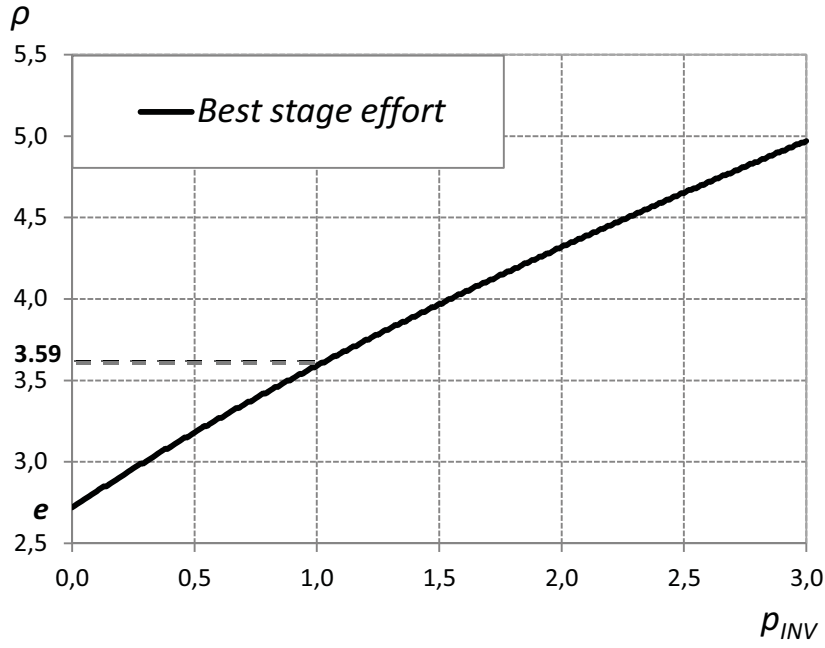


Fig. 1.9. Best stage effort vs. inverter parasitic delay.

Obviously, the actual number of stages must be an integer number. By defining $D_{opt,b}$ as the optimized best delay achieved through a stage effort equal to ρ, the trend of $D_{opt}/D_{opt,b}$ versus N/N_b can be analyzed. Such a curve has obviously a minimum equal to 1 for $N/N_b = 1$. Anyhow, because of the flatness of the curve near its minimum, even by choosing a stage number quite different from N_b such as $N = 2N_b$ ($N = 0.5N_b$), the optimized delay increases by a small quantity (only by a factor 1.26 (1.51) for $p_{INV} = 1$) with respect to the minimum achievable delay $D_{opt,b}$.

1.8 Extension of the Model to Non-Static Gates

The procedures described so far are valid also in the case of non-static gates, such as the dynamic ones and, under some conditions, those based on pass-transistors and transmission gates.

1.8.1 Dynamic and Domino gates with keeper

Let us consider a domino gate [KLL82] consisting of a dynamic NAND2, a cascaded static inverter and a PMOS keeper (introduced to improve the robustness of the structure) as shown in Fig. 1.10.

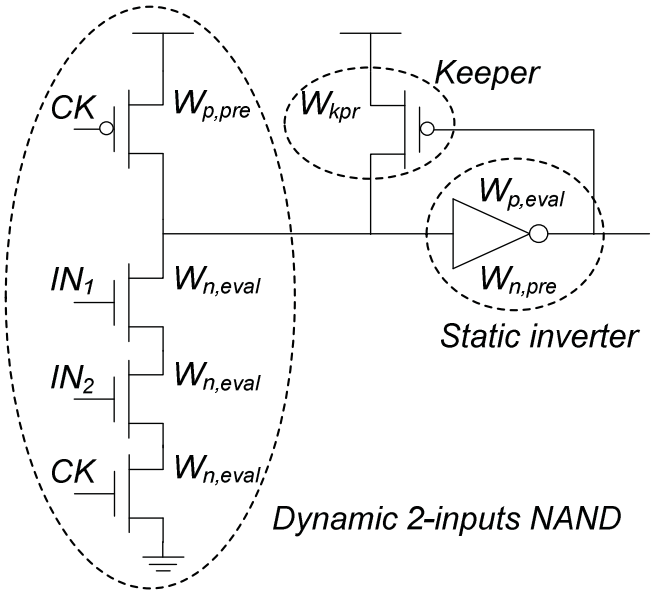


Fig. 1.10. Domino gate compound of a typically low-skewed dynamic NAND2, a typically high-skewed static inverter and a PMOS keeper.

The evaluation of parameters g, h, b and p follows the same strategy described for static gates. Note that the PMOS keeper introduces further loading effects (on both nodes X and OUT) that usually lead to non-constant branching efforts when varying the size of the dynamic gate and of the static inverter. Indeed, the PMOS keeper has to avoid an unintentional discharge of the internal node X[5] but, at the same time, it must not introduce a strong current contention. Therefore, it has typically a nearly fixed (and small) size. To estimate the impact of such a current contention between the PMOS keeper and the evaluation path in the dynamic gates, a multiplicative factor $r > 1$ is introduced in both parameters g and p. The value of r is [SSH98]

$$r = \frac{1}{1-\frac{r_{eval}}{r_{kpr}}} \tag{1.42}$$

where r_{eval} is the equivalent resistance of the evaluation PDN path in the dynamic gate and r_{kpr} is the resistance of the PMOS keeper. Note that, the constraint $r_{eval} < r_{kpr}$ has to be satisfied, otherwise the strength of the keeper would not allow to discharge the node X.

[5] Possibly due to due to charge-sharing effects, leakage currents, cross-talk noise and other sources of disturbances.

The example of the domino gate is also useful to highlight a typical case where a skewed design (i.e., non symmetrical gates) is required. Indeed, there is no point in making the dynamic gate symmetrical, since the rising/precharge transition can typically last up to an entire half-clock period. On the contrary the falling/evaluation transition belongs to the critical path and has to be fast. Therefore, the dynamic gate is usually "lo-skewed", meaning that the relative size between $w_{N,eval}$ and $w_{P,pre}$ is chosen to guarantee a falling delay smaller than the rising one.

For the same reasons, the cascaded static inverter is usually "hi-skewed", meaning that it is usually sized to speed up its rising transition (which is the one following the evaluation) thanks to a proper over-sizing of $w_{P,eval}$ with respect to $w_{N,pre}$.

Moreover the domino gate exemplifies the situation where different input signals have a different g because of the different input capacitance (see Fig. 1.5). Indeed, differently from the inputs IN_1 and IN_2, the clock signal CK drives also the precharge transistor and hence has a larger g_f. Also p_f is larger when the critical input is CK and, indeed, in real applications, the design is oriented to make IN_i the critical inputs [H00].

1.8.2 Logic with transmission-gates and pass-transistors

Transmission gates (TGs) and pass-transistors (PTs) can be straightforwardly introduced in the Logical Effort framework. The only limitation is that (a chain of) TGs (or PTs) have to considered in series to an initial gate with driving capability, i.e. connected to V_{DD} and/or GND, as shown in Fig. 1.11. Indeed, only in this way the classical simplified RC structure, or the more accurate RC tree one basing on the Elmore delay model, can be identified.

As concerns the estimation of the equivalent resistance of a TG, one has to consider that both its transistors are contemporarily conducting, and hence their resistance are in parallel. In particular, assuming that and NMOS PT exhibits a resistance equal to R (when transferring a logic "0"), a TG with equally sized PMOS and NMOS transistors exhibits a resistance nearly equal to R for both "1" and "0" inputs. Indeed, when a "0" is passing into the TG the PMOS it can be assumed with resistance 4R (remember that PMOS switched off when the output reach a voltage as low as a threshold voltage); while when a logic "1" is passing transistor TG NMOS and PMOS transistor can be both assumed with a resistance equal to $2R$ [SSH98][6].

[6] A simplifying approximation can be that of assuming the resistance of a PT doubled when transferring in its poor direction, i.e. in the direction that lead to lose a threshold voltage.

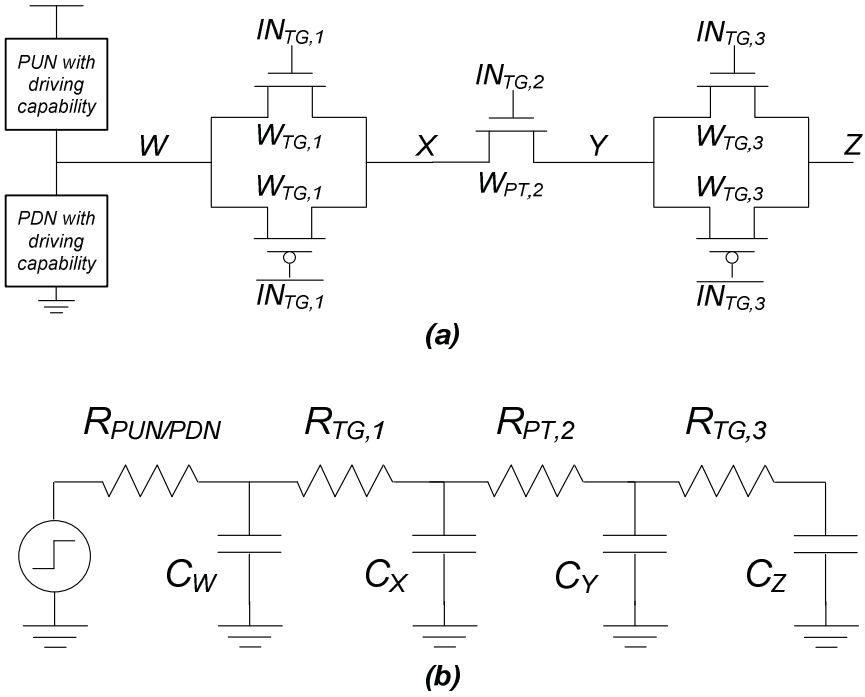


Fig. 1.11. Transmission-gates and/or pass-transistors network (a) and reduction to an equivalent RC tree (b).

It is worth noting that there is no point in doubling the size of the PMOS with respect to the NMOS (as usually done in static/dynamic gates with driving capability) since the TG resistance would become equal to $(2/3)R$ for both "1" and "0" at the input but the capacitances at the input and output of the TG would increase by 50%. Hence, the usual practice is to equally size PMOS and NMOS transistors in a TG, as shown in Fig. 1.11.

As concerns the estimation of internodal capacitances in Fig. 1.11, following criteria similar to those in Paragraph 1.5, it can be assumed that each PT (alone or composing a TG) contributes with a normalized capacitance nearly equal to $(3/2)w$ (w is its normalized width) on both its source and drain nodes.

Finally, note that when considering a structure such as that depicted in Fig. 1.11, the critical input can be one of those driving the PDN/PUN in the gate with driving capability, or one of those enabling a TG (or PT). In the latter case, as previously done for the case of stacked transistors, the capacitances in the nodes lying before the last TG (or PT) to be enabled can be considered as already charged/discharged. Obviously, the input

capacitance of the stage from the Logical Effort perspective is the gate capacitance of the PT (alone or composing a TG) driven by the critical input.

1.9 Nonlinearities and Need for Iterative Procedures

In Sections 1.5 and 1.6 it has been shown how Logical Effort can be employed as an optimization method to maximize speed other than a simple way to model delays. Nice and useful equations were derived, whose utility is however subject to the condition that g, b and p all have constant values.

In practical cases, this condition cannot be satisfied for several reasons, which are listed in the following.

- The correction factor for g and p in (1.42), which accounts for current contention due to keepers, can become a function of the gate and keeper absolute sizes when a constant ratio between their driving capabilities is not maintained;
- The branching effect in (1.26) experienced by the i-th gate in a path because of gate and/or diffusion capacitances of transistors outside the considered path can often be a function of the absolute size of the i-th gate itself (this happens when a constant proportion between the absolute values of $C_{IN,i+1}$ and $C_{off,i}$ is not maintained);
- Global interconnections and local interconnection.

In particular, regarding global interconnection, it can be modeled as equivalent RC ladder blocks and hence handled as done for stacked transistors and TGs/PTs. However, their length is normally fixed and hence the resistive and capacitive contributions that they introduce lead to g and b values that are functions of the absolute size of the gates driving such interconnections;

Differently from global interconnections, local interconnections associated with each of the internal nodes in a circuit can be simply modeled through additional parasitic capacitances. However, as reported in Chapter 4 where a methodology to estimate such parasitics within a clocked storage element is introduced, the overall local interconnections values can be subdivided in a contribution given by the gate driving the considered node, in a contribution given by the gates loading (through both gate and diffusion capacitances) the considered node and in a constant contribution. Thus, from the perspective of the gate driving the considered node, the first contribution lies on the parasitic delay, the second contribution modifies the capacitance imposed by the following gates and hence the electrical effort, while the latter contribution influences branching effect in a gate-size dependent way. Note that also the first two contributions depend on the gates sizes in complex non-linear ways and a linearization is not always feasible.

It is apparent from the discussion above that in all the considered cases several nonlinearities emerge and do not allow the optimization described in (1.32)-(1.35) to be straightforwardly applied. Therefore, in order to minimize the delay of paths including complex branching effects and the impact of interconnections, a need for iterative procedures arises, thereby weakening the logical effort handiness.

Appendix 1
Derivation of Logical Effort with a Current Approach

The Logical Effort model can be equivalently derived by assuming transistors as equivalent current generators instead of equivalent resistors. This modeling is becoming more suitable in submicron technology where during transition transistors spend much more time in the saturation region (i.e., they work for the great part of the transition as a current generator instead of providing a resistance behavior).

Indeed, under the above assumption, any timing parameter of the considered gate can be expressed as

$$t_D = K\frac{(C_{OUT}+C_L)}{I_T} \tag{A.1.1}$$

where I_T is the current provided by the PUN or PDN, and can be generally approximated with the saturation current. The inverse of I_T exhibits the same functional dependencies of the resistance R_T in (1.2c), i.e.

$$(I_T)^{-1} \propto L/W \tag{A.1.2}$$

Therefore, the Logical Effort parameters can be extracted as in (1.3)-(1.4), (1.6)-(1.10) as functions of the equivalent current behavior of the gate and results to

$$g = \frac{I_{INV}C_{IN,ref}}{I_{T,ref}C_{INV}} \tag{A.1.3}$$

$$h = \frac{C_L}{C_{IN}} \tag{A.1.4}$$

$$p = \frac{I_{INV}C_{OUT,ref}}{I_{T,ref}C_{INV}} \tag{A.1.5}$$

where I_{INV} is the equivalent current provided by a symmetrical inverter and $I_{T,ref}$ is the equivalent current provided by the template version of the considered gate. Both the equivalent currents can be simply approximated with the current given by PDN or PUN transistors in saturation region.

Chapter 2

DESIGN IN THE ENERGY-DELAY SPACE

Scaling trends have driven CMOS technology into a so-called power limited regime, where power/energy dissipation has become a prominent aspect and it is no more possible to focus solely on the optimization of circuit speed. This chapter deals with the design of energy-efficient digital circuits, i.e. to the achievement of the desired speed performances under the minimum energy consumption. Energy-delay models of logic gates and the theoretical background relative to the analysis of circuits in the energy-delay space are discussed, in order to identify the energy-efficient design criteria.

2.1 Energy Modeling

Being the optimization of circuits from the joint speed-consumption perspective the focus of this chapter, it is necessary to clarify the metrics that are used to quantify the consumption at the abstraction level this chapter deals with, i.e. the transistor-level one. In particular, two metrics are available: power and energy [R09].

Both metrics are actually interchangeable and choosing one or another is simply a matter of convention as long as transient (i.e., dynamic and short-circuit) and static (i.e., leakage) dissipative contributions are properly weighed [ACP10-1]. In the following, energy is chosen as the metric for circuits consumption. This implies that transient contributions relative to a generic circuit operation have to be simply summed, whereas static leakage-related power has to be multiplied by the time between successive operations (e.g., the duration of a clock cycle in a pipelined system) and summed to the previous transient contribution to obtain the overall energy dissipation.

In the following, a model accounting for the above contributions [ACP12-1], [ACP12-2] is reported. This model aims at the extraction of a factor χ featuring a logic gate and such that the overall gate energy, E, can be simply expressed as linearly proportional to the input capacitance, C_{IN}, i.e. to the gate size

$$E = \chi C_{IN} \tag{2.1}$$

Such a model intentionally exclude the energy dissipated in charging/discharging the load C_L, but includes that dissipated in charging/discharging C_{IN}. Again, it is simply a matter of convention.

Let us consider a static CMOS gate such as the 2-inputs NAND shown in Fig. 2.1, where also the various capacitive contributions determining the dynamic dissipation are depicted. One can distinguish among capacitances lying in the input nodes and switching according to the transition probability of the inputs, $\alpha_{sw,in}$, and capacitances lying in the output node (or in the internal ones featuring stacked structures) and switching according to the transition probability of the output (internal) node, $\alpha_{sw,out}$.

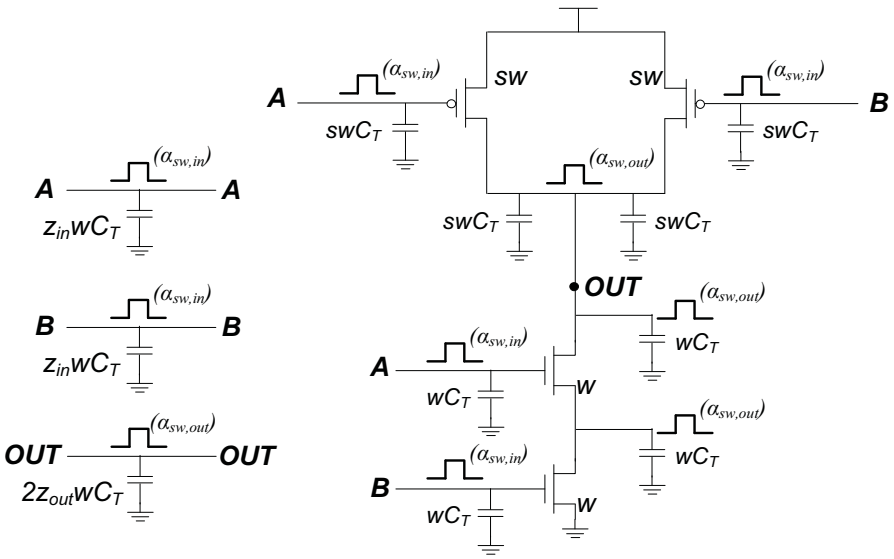


Fig. 2.1. Capacitive contributions determining dynamic energy in a gate.

Each of these capacitances is made up by transistors related contributions which can be assumes nearly equal [WH04]: gate capacitances for the input nodes and diffusions capacitances (drain-bulk and source-bulk) for the output and/or internal nodes;

In particular, defining

- w the normalized width (with respect to the minimum feasible value W_{min} imposed by the technology) of each NMOS transistor inside the gate (assuming that all NMOS have the same width and minimum lengths);
- C_T the gate capacitive contribution relative to a minimum sized transistor. It can be defined as $C_{INV}/3$, where C_{INV} is the input capacitance of a symmetrical minimum inverter (i.e., with $W_{PMOS} = 2W_{NMOS} = 2W_{min}$);
- s a multiplicative factor that defines the widths of PMOS (again all equal and with minimum lengths) with respect to the NMOS ones, thus leading to a certain skew in the speed of PUN and PDN [SSH98];

the average dynamic energy (in a clock cycle) of a CMOS gate is given by

$$E_{DYN} = \left[(1+s)\alpha_{sw,in} + (1+s)\alpha_{sw,out}\right] w C_T m V_{DD}{}^2 \tag{2.2}$$

where m is the gate inputs (equal to 2 for the NAND in Fig. 2.1), and it is assumed that each transistor contributes with a single gate and a single parasitic capacitance[1].

In order to also account for parasitic capacitances due to local wires at the input and at the output of the gate, let us introduce parameters z_{in} and z_{out} which weigh parasitic capacitive contributions through the gate size w[2]. Hence, the overall local wires capacitance in a generic node j, $C_{par,j}$, can be expressed [ACP12-1], [ACP12-2]

$$C_{par,j} = z_{out,i-1,j} w_{i-1} C_T + z_{in,i,j} w_i C_T \tag{2.3}$$

being j the node at the output and the input of the $(i-1)$-th and the i-th stage, respectively, and the average dynamic energy (2.2) becomes

$$E_{DYN} = \left[(1+s+z_{in})\alpha_{sw,in} + (1+s+z_{out})\alpha_{sw,out}\right] w C_T m V_{DD}{}^2 \tag{2.4}$$

A similar analysis concerning the static dissipation of a CMOS gate can be carried out. In particular, defining

[1] The approximation of considering a single intermodal capacitance for each stacked transistor is simple but reasonably accurate.

[2] Although the dependence of such parasitics on w is formally complex and nonlinear, linear fittings can be extracted without seriously compromising the estimation of lumped local wires capacitances

- $\rho_{sub,n}$ and $\rho_{sub,p}$ ($\rho_{gate,n}$ and $\rho_{gate,p}$) parameters depending on technology and approximately constant for any gate. They include the dependences of the sub-threshold (gate) leakage current of a single transistor on threshold voltage, on the applied biases (assuming $V_{GS} = 0$ and $V_{DS} = V_{DD}$), on the temperature and on technology parameters for a NMOS and PMOS, respectively;
- $T_{sub,n}$ and $T_{sub,p}$ ($T_{gate,n}$ and $T_{gate,p}$) factors that include the effect of the PDN and PUN topologies on their sub-threshold (gate) leakage currents, respectively (by averaging out the various currents for each inputs combination);
- $\beta_{sub,n}$ and $\beta_{sub,p}$ parameters that average the sub-threshold leakage currents of PDN and PUN according to static probabilities of logic values at input and output nodes of the gate ($\beta_{sub,n} + \beta_{sub,p} = 1$);

the average energy due to sub-threshold and gate leakage in a clock cycle having a period T_{CK} is given by

$$E_{STAT} = w\left(\beta_{sub,n}\frac{\rho_{sub,n}}{T_{sub,n}} + s\beta_{sub,p}\frac{\rho_{sub,p}}{T_{sub,p}} + \frac{\rho_{gate,n}}{T_{gate,n}} + s\frac{\rho_{gate,p}}{T_{gate,p}}\right)V_{DD}T_{CK}\theta \tag{2.5}$$

In (2.5) the parameter θ is included to account for the relation between the durations of active and inactive modes (or standby) for the part of the system where the considered gate lies. It is a correction factor leading to an effective clock period, $T_{CK}\theta$, which properly weighs the impact of static dissipation compared to dynamic one.

The above expressions (2.4) and (2.5) can be further complicated to more accurately model some effects while still remaining proportional to the parameter identifying the gate size, i.e. w. For instance, (2.4) and (2.5) can be easily generalized to deal with gates with non-minimum channel lengths, with non static (e.g. dynamic) gates, to more accurately weigh the impact of internodal capacitances on dynamic energy and of stacking effect on leakage, to consider the cases where some NMOS (PMOS) transistor within the PDN (PUN) has a width proportional but not equal to w, and so on. Hence, such models do not lead to any loss of generality. Furthermore, as already discussed for the Logical Effort model in the previous chapter, many of the parameters in (2.4)-(2.5) can be accurately characterized through simulations.

Once E_{DYN} and E_{STAT} have been found, the overall energy dissipation of the gate is

$$E = E_{DYN} + E_{STAT} \tag{2.6}$$

It is apparent that, according to the previous definitions, C_{IN} including the wire parasitic capacitance can be expressed as

$$C_{IN} = (1 + s + z_{in})wC_T \tag{2.7}$$

It is worth noting that this is the same value entering in the definition of Logical Effort parameters g and h, i.e. it is the input capacitance seen at one of the gate inputs.
And, hence, the parameter χ in (2.1) can be expressed as

$$\chi = \left(\alpha_{sw,in} + \alpha_{sw,out}\frac{(1+s+z_{out})}{(1+s+z_{in})}\right) m{V_{DD}}^2 + \\ + \frac{\left(\beta_{sub,n}\frac{\rho_{sub,n}}{T_{sub,n}} + s\beta_{sub,p}\frac{\rho_{sub,p}}{T_{sub,p}} + \frac{\rho_{gate,n}}{T_{gate,n}} + s\frac{\rho_{gate,p}}{T_{gate,p}}\right)}{(1+s+z_{in})C_T} V_{DD}T_{CK}\theta \tag{2.8}$$

It is worth noting that the above model neglects short-circuit dissipation. Given the increasing V_{TH}/V_{DD} ratios, this contribution tends to relatively decrease with technology scaling [RCN03]. Nevertheless, when the input rise/fall times are quite large, the impact of short-circuit energy can be non negligible.

Differently from the dynamic and leakage ones, short-circuit contribution cannot be approximated as linearly dependent on the gate size. Indeed, it increases with gate size for three reasons:

- the linear dependence of the PDN and PUN currents on w;
- the approximately proportional dependence on the input rise/fall time, i.e. on the output rise/fall time of the preceding gate [RCN03];
- the approximately inverse dependence on the output rise/fall time of the gate itself [RCN03].

The last two terms can be assumed (by neglecting the parasitic delays in the computation of input rise/fall times) as nearly linearly dependent on w.

Overall, the short-circuit dissipation can be equaled to

$$E_{SC} = \frac{d_{in}}{d_{out}}\rho_{sc}\left[\left(T_{sc,n} + sT_{sc,p}\right)\alpha_{sw,out}\right]w \tag{2.9}$$

where d_{in} and d_{out} are input and output rise/fall times according to Logical Effort model, while parameters $T_{sc,n}$ and $T_{sc,p}$ average the various possible output transition cases according to PDN and PUN topologies. Finally ρ_{sc} is a further parameter accounting for the impact of technology and V_{DD}.

2.2 Energy-Delay Space Analysis and Hardware-Intensity

2.2.1 The energy-efficient curve

For a digital circuit under a fixed supply voltage V_{DD} and whose last stage is loaded with a capacitance C_L, the "energy-efficient curve" (EEC) is made up by the design points exhibiting the minimum delay for a fixed energy dissipation or, equivalently, the minimum energy consumption for a fixed delay [ACP09-1], [ACP10-2], [PM02]. By definition, other design points above the EEC lead to a needlessly higher energy under the same speed performances, as shown in Fig. 2.2.

As previously stated, the conventions of considering the input capacitance of (the first stage of) the circuit, C_{IN}, as a further design variable to be optimized, and that of including (excluding) the energy dissipated in charging/discharging C_{IN} (C_L), are adopted. This assumption is different from that adopted in [OK06], [ZS02], [ZS03], [MSN04] and, while it was a simple matter of convention when referring to the modeling of the energy of a circuit, it is shown that it becomes a necessary care when the target is the full exploration of the E-D potentials of a topology.

In [PM02] it was predicted that the EEC of any circuit has an hyperbolic shape

$$(E - E_0)(D - D_0) = E_0 D_0 \tag{2.10}$$

being E_0 and D_0 the minimum energy and minimum delay asymptotes, as shown in Fig. 2.2. Actually, substantial deviation from (2.10) are found when analyzing real circuits and hence a correction factor γ (typically $0 < \gamma < 1$) can be introduced to fit real data [ZS02], [ZS03]

$$(E - E_0)(D - D_0) = \gamma E_0 D_0 \tag{2.11}$$

Despite our assumptions of including the dissipation related to a fully optimizable C_{IN} and excluding that relative to the load C_L differ from those in [ZS02], [ZS03], the general character of (2.11) is retained. In particular, looking at the generic EEC depicted in Fig. 2.2, there is a minimum energy value, E_{min}, that is achievable with the minimum transistors sizes allowing correct operation, hence the points between E_0 and E_{min} have not a physical correspondence (see Fig. 2.2).

Moreover, regarding delay, the value D_0 can be approached only asymptotically through transistor sizing, and measures the maximum speed potential of a specific topology. More specifically, one can indefinitely trade energy for delay by increasing C_{IN}. On the contrary, if C_{IN} is fixed [OK06] ,[ZS02], [ZS03], [MSN04], a minimum delay for a given load is actually

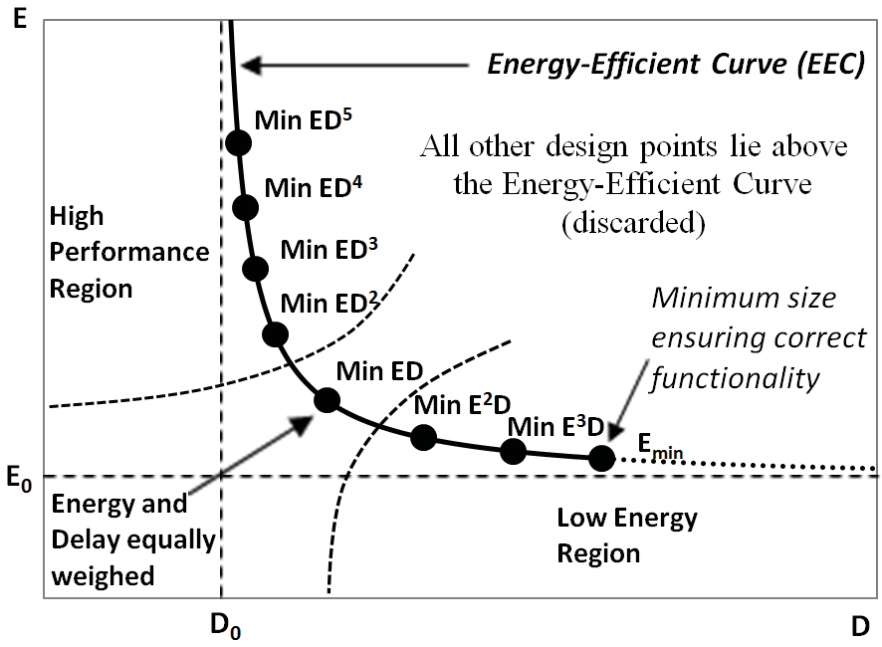


Fig. 2.2. Energy-efficient curve and designs optimizing the metrics $E^i D^j$.

reachable and corresponds to the Logical Effort sizing. Nevertheless, also the asymptotic value D_0 under a varying C_{IN} can be estimated through Logical Effort and it is the parasitic delay P.

As concerns parameter γ in (2.11) and the actual analytical expression of the EEC under our assumption, analytical calculations can be carried out only for a single logic gate [ACP12-1]. Indeed, according to Logical Effort model [SSH98], one has

$$\frac{D-D_0}{D_0} = \frac{gh}{p} = \frac{g}{p}\frac{C_L}{C_{IN}} \tag{2.12}$$

As concerns the energy, from (2.1) one gets

$$\frac{E-E_0}{E_0} = \frac{\chi C_{IN} - \chi C_{IN,min}}{\chi C_{IN,min}} = \frac{C_{IN} - C_{IN,min}}{C_{IN,min}} \tag{2.13}$$

being $C_{IN,min}$ the minimum input capacitance of the gate (i.e., when its transistors are all minimum sized).

By referring to (2.11) and using (2.12)-(2.13), γ is given by

$$\gamma = \frac{gC_L}{pC_{IN,min}} - \frac{D-D_0}{D_0} = \frac{gC_L}{D_0 E_0} - \frac{D-D_0}{D_0} \tag{2.14}$$

The above formula indicates that, under our assumptions, formula (2.11) can be applied with a value of γ that is dependent on the variable D, that is to say the EEC is not a pure hyperbole. However, γ can be approximated in a sufficiently accurate way by its first term, $gC_L/pC_{IN,min}$ as long as the delay is not much higher than $D_0 = p$.

Nevertheless, when dealing with circuits made up by more than one gate, no analytical expression can be determined for γ, and, in such a case, it is consistent to assume in (2.11) γ as a constant parameter.

2.2.2 Energy-delay metrics and hardware intensity

In the last two decades digital circuit designers have become familiar with the use of composite energy-delay metrics to effectively translate the more and more stringent constraints on the speed performances while not disregarding the energy dissipation.

The first (and at first glance the most appropriate) composite metric to be introduced is the simple ED product, which equally weighs the two quantities. Another popular metric is the ED^2 product where speed has priority over energy. The latter metric is claimed to have useful properties such as a nearly zero sensitivity on the supply voltage [M01].

However, although designs optimizing (i.e., minimizing) the above metrics are maximally efficient for a given delay (or energy), it is clear that a generalization is required when analyzing and/or designing a circuit over the entire spectrum of the delay (energy) values it can achieve.

Hence, the general class of metrics E^iD^j, or equivalently ED^η (being η equal to j/i) as originally presented in [PM02], are introduced. By varying the exponents $i \geq 0$ and $j \geq 0$ ($\eta \geq 0$), any tradeoff between energy and delay can be explored. The extreme cases are obtained when $j/i = 0$ ($\eta = 0$) and when $j/i = \infty$ ($\eta = \infty$), which, once optimized, represent the designs having the minimum possible energy and delay, respectively.

Turning back to the EEC introduced before, one has that a design solution minimizing a metric E^iD^j (ED^η), lies in the EEC [PM02], i.e. this curve is made up of all points that minimize E^iD^j (ED^η), for some i and j (η), as shown in Fig. 2.2.

The demonstration of this assertion is quite simple and intuitive. Indeed, considering a circuit under a fixed load and supply voltage, both its delay and energy are functions of its sizing W (W is an array containing the sizes of transistors in all circuit gates). A design minimizing an E^iD^j metric for some (i,j) has a delay D^* which is obtained with a certain size W^* (i.e., $D^* = D(W^*)$). Since the size W^* minimizes a product E^iD^j, in which the energy is taken into account with $i \geq 0$, the value $E^* = E(W^*)$ of this design is the minimum among all the designs exhibiting a delay $D = D^*$ and

thus it lies on the EEC. More rigorous analytical proofs can be found in [PM02].

From the above considerations, the indexes i and j (η) identify cost functions for optimizing hardware under a fixed load and supply voltage, and, according to [ZS02], [ZS03], [Z03], the value j/i (η) is defined "hardware intensity". Basically, j/i (η) quantifies the effort to be spent in sizing a circuit to optimize the speed of the circuit at the expense of its energy consumption. The higher j/i (η), the higher the effort to further optimize speed. The region of the E-D design space where metrics with $j > i$ ($\eta > 1$) are minimized is hence called the high-performance one, while the region where metrics with $j < i$ ($\eta < 1$) are minimized is called the low energy one. The former is featured by lower and lower delay gains achieved at the cost of larger and larger increments in energy as long as the delay itself diminishes. Analogous considerations are valid for the low energy region.

The graphical interpretation of hardware intensity is shown in Fig. 2.3 [ZS03], [Z03]. The solid line plots a typical EEC for a generic circuit. Dotted curves show several contours of the cost function E^iD^j for three values of the hardware intensity. The point in the E-D space at which the EEC tangents the lowest of the contours corresponds to the energy-efficient implementation of the circuit for that specific hardware intensity value [ZS02], [ZS03].

Accordingly, the analytical interpretation of hardware intensity is related to the energy-to-delay sensitivity evaluated in correspondence of the design points optimizing the E^iD^j (ED^η) metrics [ZS02], [ZS03], [ACP12-1].

Indeed, by referring to the former ones, the design point minimizing E^iD^j for a given (i,j) leads to a zero derivative of E^iD^j with respect to D and E [PM02], [ACP12-1]

$$\frac{\partial(E^iD^j)}{\partial D}_{E^iD^j{}_{min}} = \left(iE^{i-1}D^j\frac{\partial E}{\partial D} + jE^iD^{j-1}\right)_{E^iD^j{}_{min}} = 0 \tag{2.15}$$

$$\frac{\partial(E^iD^j)}{\partial E}_{E^iD^j{}_{min}} = \left(iE^{i-1}D^j + jE^iD^{j-1}\frac{\partial D}{\partial E}\right)_{E^iD^j{}_{min}} = 0 \tag{2.16}$$

Solving the set of equations (2.15)-(2.16), one finds

$$S^E_{D\,E^iD^j{}_{min}} = \left(\frac{\partial E}{\partial D}\frac{D}{E}\right)_{E^iD^j{}_{min}} = -\frac{j}{i} \tag{2.17}$$

When carrying out analogous calculations by referring to the ED^η metrics, the results is simply $-\eta$. Anyhow, the adoption of the two indexes i

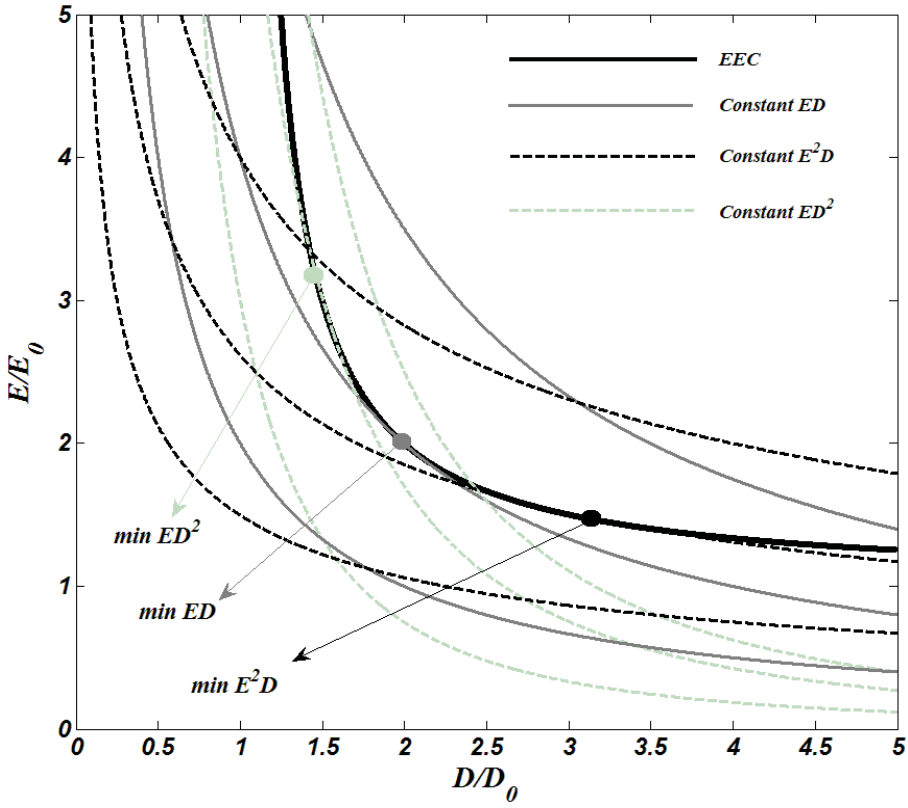


Fig. 2.3. Typical energy-efficient curve and constant cost function contours for $j/i = 1.0$, $j/i = 0.5$ and $j/i = 2.0$.

and j allows for better clarifying the E-D tradeoff when the generic E^iD^j FOM is minimized. Indeed, in the neighborhood of the optimum E^iD^j design, a j% speed increase is traded for a i% energy increment and vice versa. Finally, from (2.17) it is apparent that metrics leading to the same j/i ratio are not distinguishable.

2.2.3 Voltage intensity and generalization of the sensitivity criterion

So far the focus was on hardware, i.e. transistors sizing, optimization. However, other tuning variables, such as the supply voltage V_{DD} and the transistors threshold voltages, are available in the circuital level design.

As concerns supply voltage, by introducing the dimensionless derivatives of energy and delay with respect to V_{DD}, henceforth referred as ν,

$$E_\nu = \frac{\nu}{E}\frac{\partial E}{\partial \nu} \tag{2.18}$$

$$D_v = -\frac{v}{D}\frac{\partial D}{\partial v} \tag{2.19}$$

and taking their ratio, one can define "voltage intensity", β, as the energy-to-delay sensitivity relative to the variation of v at a fixed hardware intensity j/i [ZS02], [ZS03]. Hence, just like j/i represents the negative energy (delay) relative gain at the cost of a relative increase in delay (energy), achievable by restructuring hardware, i.e. sizing w, under a fixed v [ZS02], [ZS03], β represents the energy (delay) relative increase (decrease), achievable by increasing v under a fixed w [ZS02], [ZS03],

$$\beta = -\frac{\partial E}{\partial D}\frac{D}{E}\Big|_{v\ variable-\ w\ fixed} \tag{2.20}$$

The E_v and D_v values cannot be simply determined through classical $E \propto {V_{DD}}^2$ and $D \propto 1/(V_{DD} - V_{TH})^2$, given the impact of leakage and short-circuit currents on energy and the complexity of $I_D = f(V_{GS}, V_{DS})$ relationship featuring nanometer transistors. Therefore, it is necessary to develop comprehensive models of energy and delay as functions of the V_{DD} value [AP06] (similarly to those relative to transistors sizing that were discussed in the previous paragraph) or extract E_v and D_v for the various gates in a circuit through simulations. To have an idea of the main trend, according to experimental results [ZS03], the voltage intensity β almost linearly increase with V_{DD} for typical CMOS circuits.

The most important aspect of this discussion is that hardware and voltage intensities are related when optimizing a circuit in the E-D space.

If one considers a circuit (like a pipeline stage) that has to satisfy a given maximum delay constraint, such a requirement can be achieved at different combinations of the j/i and θ values. However, the energy-efficient implementation, i.e. that with the minimum energy, is the one featured by

$$j/i = \beta \tag{2.21}$$

Indeed, energy and delay are functions of the variables (w, v), and, by solving the problem of minimizing $E(w, v)$ under the constraint $D(w, v) = D^*$, one finds [ZS02], [ZS03], [DZO06]

$$\frac{\partial D}{\partial w}\frac{\partial E}{\partial v} = \frac{\partial D}{\partial v}\frac{\partial E}{\partial w} \tag{2.22}$$

which means $j/i = \theta$. Hence, for an optimal balance between the supply voltage and the transistors sizing, the relative speed gain achieved at the cost of a given relative energy increase due to an increment in the supply voltage must equal the relative speed gain achieved at the cost of a given relative

energy increase due to a larger transistors sizing [ZS03]. This result disproves the common misconception that the lowest energy can be achieved by designing circuit for the highest speed and then reducing the power supply up to the lowest value that satisfies the delay requirement.

Further generalizing the above analysis to any kind of design variable, e.g. like threshold voltages [LS93], [GGH97], and to the sensitivity of energy to delay with respect to a change in that variable, as in (2.18)-(2.19), the minimum energy under a given delay constraint is achieved when [MSN04]

$$S_x(X) = \frac{\partial E}{\partial D}\frac{D}{E}\Big|_{x\ variable} = S_y(Y) = \frac{\partial E}{\partial D}\frac{D}{E}\Big|_{y\ variable} \qquad \forall x, y \qquad (2.23)$$

being x and y design variables, i.e., the energy-efficient corresponds to the design with $x = X$, $y = Y$ and so on.

2.3 Energy-Efficient Design of Digital Circuits

In this section practical optimization techniques to achieve the energy-efficient design of digital circuits at the circuit level, by considering various levels of complexity, are discussed. In particular, some preliminary remarks are first provided concerning the role played by the input capacitance of the circuit and the definition of design space bounds, both essential regardless of the actually employed optimization technique. Then, the case of simple basic blocks whose complexity allows a simulations-based optimization is considered by ending with large designs that can be dealt with by resorting to convex optimization and exploiting simple E-D models.

2.3.1 The role of the input capacitance

As shown in recent works [OK06], [DZO06], when dealing with the issue of energy-efficient design, the input capacitance, C_{IN}, of a logic circuit cannot be simply assumed as fixed. Granted that the adopted C_{IN} value is also related with the architectural-level design strategies [DZO06], differently from [ZS02], [ZS03], [MSN04], [OK06], here C_{IN} (i.e., the transistors sizes determining its value) is considered as an additional design variable to be fully optimized like all the other transistors sizes. Indeed, an effective exploration of the E-D space to achieve the required E-D tradeoff strongly depends on C_{IN}.

A second assumption, differently to [ZS02], [ZS03], [MSN04], [OK06], [DZO06] is that of including the energy dissipated in the charge and discharge of the C_{IN} and to exclude the energy dissipated in the charge/discharge of the external output load, C_L. Indeed, the first term is inherently related to the adopted circuit sizing (here C_{IN} is a further design

knob), whereas the latter term does not depend on the features of the topology [SO99], [GNO07].

It is worth opportunely addressing the consequences of the C_{IN} optimization within a wide range of exploration [ACP10-2], which can find benefit in both high speed and low power design. In general, a throughput increment can be achieved by means of an increase in the degree of parallelism and/or a more critical sizing of all the gates in the logic paths (e.g., when the serial part of code is dominant and parallelization is not so effective). In the latter case, if C_{IN} is increased with respect to medium values, it means that the topology is being sized to achieve a high speed (increasing the energy consumption). Even if the circuit imposes a larger load on the preceding logic stage (e.g. in a pipeline), in high-speed applications the speed penalty of the preceding logic stages could be exceeded by the speed improvement in the considered topology. This tradeoff cannot be explored if one does not assume a fully variable C_{IN}.

Conversely, when sizing to achieve low-power, low-speed operation, C_{IN} can be strongly reduced. Indeed, granted that the above tradeoff is still valid, the low-power applications are typically featured by long cycle times and hence can easily tolerate slower stages and high logic depths (e.g., when no parallelization is adopted and the processing is actually done serially through single deep paths). In such a context, a slower topology can be tolerated in favor of its smaller energy dissipation.

Obviously, there always exist practical limits on the adoptable C_{IN} values. Nevertheless, once the full EEC is extracted, the designer can easily select the portion of interest according to practical constraints in terms of maximum allowed C_{IN}.

Finally, it is worth highlighting that, when referring to the "first stage", we mean the first gate in the path of the circuit whose sizing is assumed as a reference in terms of timing criticality. Indeed, several input-to-output paths coexist in a circuit composed by more than a single gate and, being the delay of the circuit identified with the maximum among the delays in its various input-to-output paths, the target must obviously be that of equaling these delays. Among the various paths, it is then possible to identify one (typically the longest) that can be used as the reference to identify the C_{IN} of the circuit. Note that, since C_{IN} is fully varied and the optimization targets the equality of the various concurrent delays, the input capacitances of the first stages of all the other paths are optimized and fully explored as well.

2.3.2 Definition of design space bounds

Regardless of the methodology actually employed for EEC extraction, one first needs to define practical design space bounds allowing to limit the space of solutions. As it is shown successively, this issue is particularly important in the case of simulations-based procedures and nonlinear

optimizations. In these cases, a larger and larger computational effort is required if the design space bounds are not properly defined. On the contrary, this issue becomes less relevant when one adopts simple E-D models leading to a convex optimization problem.

At the same time, one must be sure to catch the optimum sizings actually leading to the desired energy-delay tradeoffs, i.e. one must guarantee that the selected bounds strictly contain the searched optimum sizings.

In [DZO06] it is shown that Logical Effort designs lie above the EEC, i.e. they are not the most efficient possible designs. Even if, unlike [DZO06], the C_{IN}-related dissipation is here included and C_{IN} is assumed as a design variable, the same result still holds[3]. Nevertheless, the energy-to-delay sensitivity of Logical Effort designs can be exploited to determine design space bounds.

More specifically, one can be interested in the portion of the EEC up to a certain minimum-E^iD^j design point with $j/i = X$, i.e. the portion of the EEC made up by energy-efficient designs that minimize FOMs with j/i less or equal than X [4]. In such a case, the design bounds can be defined through the "limiting" Logical Effort sizing exhibiting an energy-to-delay sensitivity with respect to C_{IN} equal to X., i.e. the upper bound of C_{IN}, $C_{IN,max}$, is the value which satisfies [ACP12-1]

$$S_D^E\big|_{C_{IN}} = \frac{S_{C_{IN}}^E}{S_{C_{IN}}^D} = -\frac{j}{i} = -X \tag{2.24}$$

The definition of $C_{IN,max}$ leads also to the definition of the upper bounds for the other design variables (i.e. transistors sizes) that are determined by the Logical Effort sizing with $C_{IN} = C_{IN,max}$.

The (2.22) can be analytically evaluated thanks to the property of Logical Effort designs. In particular, as discussed in Chapter 1, given C_{IN} and C_L, the minimum normalized path delay D_{TOT} of a circuit simply made up by a path of N cascaded gates is

---

[3] Indeed, as explained in the previous section, the minimum energy under a given speed constraint is reached when the sensitivity with respect to "all" the tuning variables is the same. Logical Effort designs are featured by an infinite energy-to-delay sensitivity with respect to the sizes of internal transistors (since delay cannot be further reduced given a fixed C_{IN}), but not with respect to the size of transistors defining C_{IN}. Hence, the condition in (2.23) is not satisfied for Logical Effort designs, which thus are not energy-efficient [DZO06]. Only when C_{IN} approaches infinity, the Logical Effort design is featured by an equal (and infinite) energy-to-delay sensitivity with respect to all the tuning variables.

[4] It is worth noting that if the searched X is not large enough (say, smaller than 3), the bounds determined through Logical Effort is not much close to the minimum-E^iD^j design with $j/i = X$.

$$D_{TOT} = N\sqrt[N]{GBH} + P = N\sqrt[N]{F} + P \tag{2.25}$$

which can be rewritten as

$$D_{TOT} = P(1 + k) \tag{2.26}$$

where

$$k = \frac{N\sqrt[N]{GB}\sqrt[N]{C_L}}{P\sqrt[N]{C_{IN}}} \tag{2.27}$$

is the relative delay increment with respect to the ideal and practically inaccessible minimum path delay (i.e., the path parasitic delay P).

From (2.26)-(2.27), the sensitivity of the optimized path delay D_{TOT} to C_{IN}, is given by

$$S_{C_{IN}}^{D_{TOT}} = \frac{\partial D_{TOT}}{\partial C_{IN}} \frac{C_{IN}}{D_{TOT}} = -\frac{1}{N} \frac{k}{k+1} \tag{2.28}$$

which is a function of the only C_{IN}.

As for the path delay D_{TOT}, it is possible to univocally determine the energy E_{TOT} of a single path circuit sized through Logical Effort for a given C_{IN} and C_L. According to (1.33) and (1.34), the input capacitance, C_N, and the energy, E_N, of the N-th gate are respectively given by

$$C_N = \frac{g_N b_N \sqrt[N]{C_{IN}}}{\sqrt[N]{GBC_L}} C_L \tag{2.29}$$

$$E_N = \chi_N C_N \tag{2.30}$$

By iterating the above reasoning and going backward through the path, one finds that, the input capacitance and energy of the i-th gate (for the Logical Effort design) are

$$C_i = \frac{\left(\prod_{j=i}^{N} g_i\right)\left(\prod_{j=i}^{N} b_i\right)(C_{IN})^{\frac{N-i+1}{N}}}{(GBC_L)^{\frac{N-i+1}{N}}} C_L \tag{2.31}$$

$$E_i = \chi_i C_i \tag{2.32}$$

and $C_1 = C_{IN}$.

Therefore, the overall dissipation of the reference path is

$$E_{TOT} = \sum_{i=1}^{N} \left[\chi_i \frac{\left(\prod_{j=i}^{N} g_i\right)\left(\prod_{j=i}^{N} b_i\right)(C_{IN})^{\frac{N-i+1}{N}}}{(GBC_L)^{\frac{N-i+1}{N}}} C_L \right] \tag{2.33}$$

Although one cannot attain to a simple expression like (2.28), also the sensitivity of the overall energy E_{TOT} to C_{IN} can be again expressed as a function of the only C_{IN}

$$S_{C_{IN}}^{E_{TOT}} = \frac{\sum_{i=1}^{N}\left[\chi_i \frac{\left(\prod_{j=i}^{N} g_i\right)\left(\prod_{j=i}^{N} b_i\right)(C_{IN})^{\frac{N-i+1}{N}}}{(GBC_L)^{\frac{N-i+1}{N}}}\left(\frac{N-i+1}{N}\right)C_L\right]}{\sum_{i=1}^{N}\left[\chi_i \frac{\left(\prod_{j=i}^{N} g_i\right)\left(\prod_{j=i}^{N} b_i\right)(C_{IN})^{\frac{N-i+1}{N}}}{(GBC_L)^{\frac{N-i+1}{N}}}C_L\right]} \tag{2.34}$$

Finally, (2.28) and (2.34) can be combined to evaluate (2.24) and determine $C_{IN,max}$.

Unfortunately, formula (2.24) cannot be always applied straightforwardly given that g_i, h_i, b_i and p_i are often not available in a closed-form as functions of C_{IN} [5]. Rather, g_i, h_i, b_i and p_i themselves can be found only by numerically solving a set of complex non-linear equations when applying the Logical Effort method for a given C_{IN} (see note 5).

Furthermore, when the circuit is not simply made up by a single path, also the energy of the circuit is not simply that in (2.32) (see note 5), and it is not always possible to find closed form relationships describing the energy of the other gates as functions of C_{IN}. Nevertheless, one has to keep in mind that, when sizing for maximum speed, the energy still depends on the only variable C_{IN}.

---

[5] There are three main reasons for this issue.

1) The various sources of nonlinearities listed in the second paragraph, which imply the need for iterative procedures to be solved to determine the Logical Effort sizing.

2) The fact that not all the transistors in the circuit have to be considered as variables to be optimized. Actually, only transistors lying in input-to-output paths should represent variables to be optimized in the E-D space, since they affect both consumption and speed. On the contrary, there can exist some parts of the circuit whose size must be simply the minimum one guaranteeing a correct operation, since they affect only energy. This is the case for instance of keepers, pulse generators, and so on. However, these gates have a size dependent on the design variables to be optimized (to guarantee the correct operation) and hence affect the branching effort b_i in a non-linear way.

3) The possible presence of reconvergent paths or multiple outputs. Indeed, transistors in the paths that lie nearby to the path assumed as the reference one affect speed too, since, as previously explained, they must be sized so that all concurrent paths exhibit the same delay (for this reason, their sizes must be considered as design variables to be optimized in the E-D space exploration successive to the definition of design space bounds). When formulating the Logical Effort equations, besides satisfying (1.35a) for stages in the reference path, additional equations arise that are relative to the equality of the various concurrent paths delays. This makes the problem of finding the minimum delay design even more complex and nonlinear.

Therefore, the need for iterative procedures arises. For instance, one can adopt the following cycle for increasing C_{IN} [ACP10-2], [ACP12-1]:

a) under the current C_{IN} (re)apply the Logical Effort method to find the transistor sizes leading to the minimum delay of all the concurrent paths in the circuit (a non-linear set of equations must be solved, see note 5);
b) (re)simulate energy and delay;
c) (re)extrapolate the E_{TOT}vs.C_{IN} and D_{TOT}vs.C_{IN} fitted curves and (re)compute the sensitivity (2.24) around the current C_{IN} value;
d) (re)compare such sensitivity with the desired one $-j/i$. If $\left|S_{D_{TOT}}^{E_{TOT}}\right|_{C_{IN}}\Big| < \left|\frac{j}{i}\right|$, C_{IN} is increased and cycle comes back to a). Otherwise, cycle stops and $C_{IN,max}$, together with the overall design space bounds, is found.

To exemplify the above procedure, let us consider a 4-bit Ripple-Carry Adder in a 65-nm technology, whose schematic is shown in Fig. 2.4, under a load equal to 16 minimum inverters and $V_{DD} = 1V$. In Fig. 2.5 the energy-to-delay sensitivity relative to the variation of C_{IN} is shown. The x-axis corresponds to the value of the transistor width w_1 (normalized to the minimum W_{min}) determining the size of the first stage of the circuit, i.e. C_{IN}, while other four transistors widths are selected as further tuning variables, $w_2 - w_5$ (see Fig. 2.4 and [ACP12-1] for details).

From figure inspection, according to the above discussed procedure, one has that the minimization of the ED^3 metric requires $w_1 > 15$, while the minimization of the ED^4 metric requires $w_1 > 31$. The corresponding bounds on the other variables $[w_2, w_3, w_4, w_5]$ are [17,18,17,7] for the ED^3 metric and [31,30,25,9] for the ED^4 metric [ACP12-1]. These bounds are very close to the transistors sizes actually optimizing the two metrics, which are equal to [15,17,17,16,6] and [29,30,30,18,10], respectively [ACP12-1].

Summarizing, these results confirm the effectiveness of such a procedure, which aims at practically bounding the design space through the analysis of the energy-to-delay sensitivity relative to the variation of C_{IN} in minimum delay (i.e. Logical Effort based) designs.

2.3.3 Simulations based optimization of small size circuits

When dealing with small circuits featured by few design variables (i.e., simple basic circuit blocks), the energy-efficient optimization can be carried out by employing a simulations based procedure, allowing to evaluate both energy and delay with the maximum possible degree of accuracy [ACP10-2], [ACP12-1], [ACP11-1], [ACP11-2]. Obviously, given that simulations are time consuming, the accuracy in E-D estimation is traded for a non extensive exploration of all the possible design solutions and hence some sort of algorithm have to be applied to reduce the computational effort but still allowing to reach the optimum points.

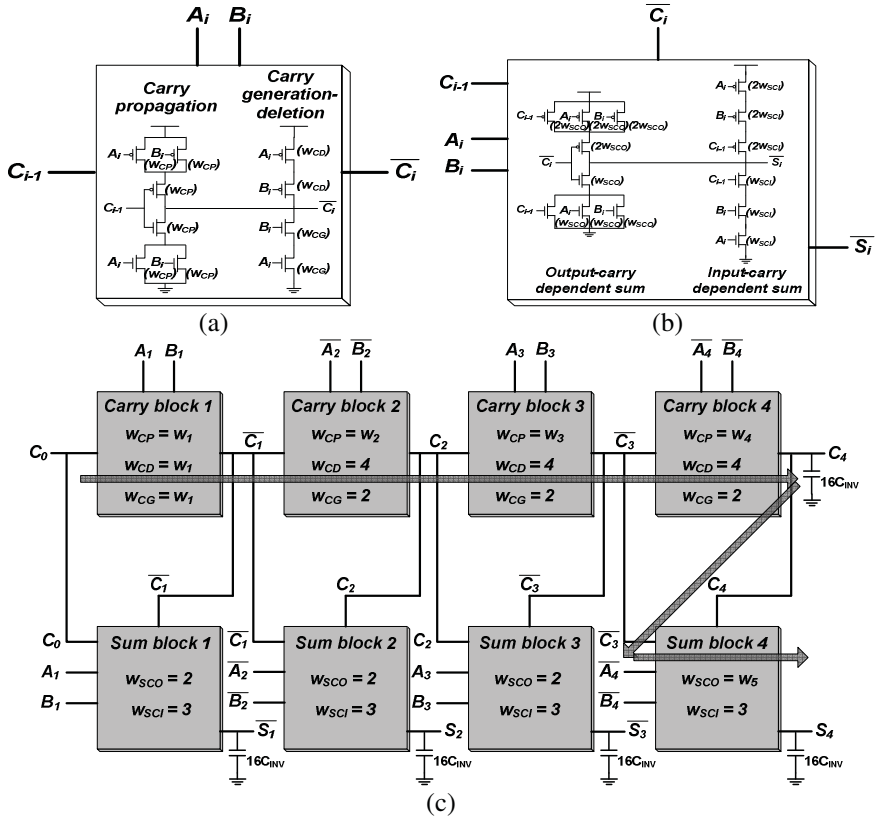


Fig. 2.4. 4-bit RCA: carry block (a), sum block (b), whole structure (c).

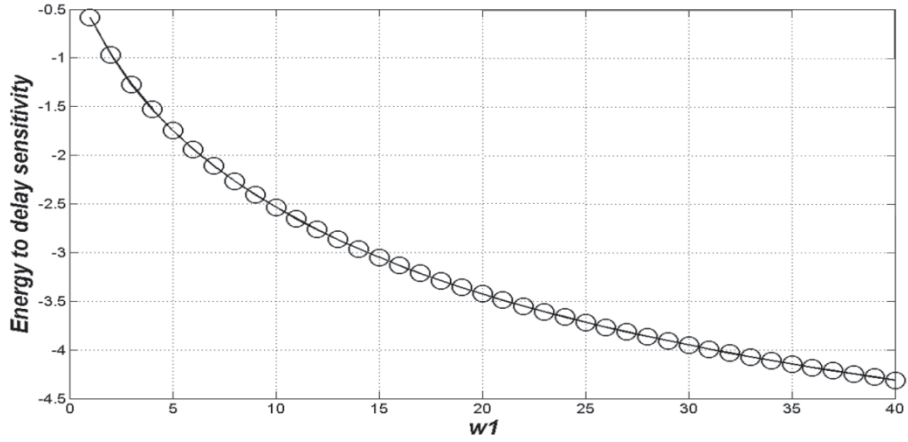


Fig. 2.5. 4-bit RCA: energy-to-delay sensitivity of Logical Effort designs as a function of the first stage size.

As a useful consequence of the properties of the E^iD^j metrics discussed in the previous paragraph, from a practical perspective the EEC of a circuit can be extracted by simply minimizing E^iD^j for a limited number of pairs (i,j) and interpolating such optimum points.

In particular, a binary search can be employed to identify minimum-E^iD^j designs because in a simulations-based framework it is worth assuming that E^iD^j functionals are nearly convex in the design space [ACP10-2] (anyhow, more complex search criteria can be adopted as well). Moreover, the design space to be explored can be progressively reduced. Indeed, assuming $j_1/i_1 < j_2/i_2$, a design optimizing $E^{i_1}D^{j_1}$ is always featured by a sizing smaller than that optimizing $E^{i_2}D^{j_2}$. Therefore, one can start from the metric with an highest j/i ratio, and, once it is optimized with a sizing W', the optimization of the successive (in terms of decreasing j/i value) FOM is constrained by bounding the design space with the sizing W', and so on [ACP10-2].

To exemplify the above search algorithm, the results relative to the simulations-based extraction of the EEC for the 4-bit adder previously mentioned are reported. In Fig. 2.6 the design points explored in the search space are depicted with small circles, while the energy-efficient ones minimizing some E^iD^j metrics are highlighted. It is apparent that the explored designs crowd near the EEC, thus highlighting the search algorithm effectiveness.

As a further validation, the energy-to-delay sensitivity in the minimum E^iD^j points is also evaluated and compared with the theoretically expected $-j/i$ value, as shown in Tab. II.I. Results again confirms that the described search algorithm allows to fairly well identify the minimum E^iD^j points.

2.3.4 Nonlinear and convex optimization of large size circuits

When dealing with circuits of large size, that is to say featured by several tens to several thousands design variables, a simulations based optimization becomes infeasible because of its prohibitive computational effort and a design space exploration based on compact E-D models is required.

To give an idea, the full E-D space exploration of a simple buffered 2:1 multiplexer, featured by five design variables (transistors widths swept with a W_{min} step), takes nearly a minute on a current desktop computer when using the E-D models in the second paragraph and the previous procedure to determine the design space bounds. The tens of millions designs explored are shown in Fig. 2.7. Considering larger circuits, the complexity grows exponentially and a full exploration soon becomes infeasible.

If the objective function to be minimized (e.g., energy) and constraints functions to be satisfied (e.g., delay related) have not any special feature (e.g., convexity), the optimization problem is said a "nonlinear optimization"

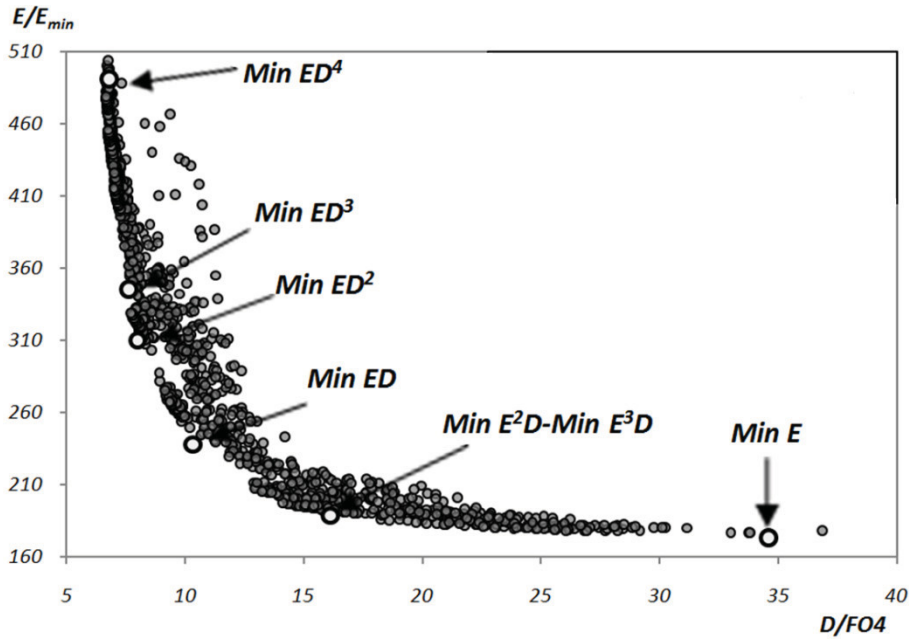


Fig. 2.6. $E - D$ space exploration for the 4-bit RCA ($C_L = 16C_{IN}, V_{DD} = 1V$).

TABLE II.I: 4-BIT RCA: MINIMUM $E^i D^j$ DESIGNS

Sizing ↓	*D [FO4]*	*E [E_{min}]*	S_D^E	*-j/i*
Min ED⁴	6.79	490.89	-4.24	-4.00
Min ED³	7.62	345.12	-2.78	-3.00
Min ED²	7.99	310.12	-1.85	-2.00
Min ED	10.32	238.00	-0.92	-1.00
Min E²D	16.11	188.62	-0.37	-0.50
Min E³D	16.11	188.62	-0.37	-0.33
Min E	34.59	173.43	-0.08	-0.00

or a "nonlinear programming" [BV03]. This is actually the case when both energy and delay are very accurately modeled by accounting for several effects even in complex ways (e.g., short-circuit currents, impact of input slope on the delay, dependence of leakage on the threshold voltages, etc.).

As long as the design variables are no more than several tens, global optimization algorithms, ensuring that the true global optimum solution is found, can be applied while still maintaining the computational effort feasible, i.e. from hours to no more than few days [BV03]. Obviously, the accuracy in E-D estimation is not maximum as in the simulations based case, but, on the other hand, a much broader exploration of the design space can be performed in a comparable time [OZD05]. Note that in such a case, the

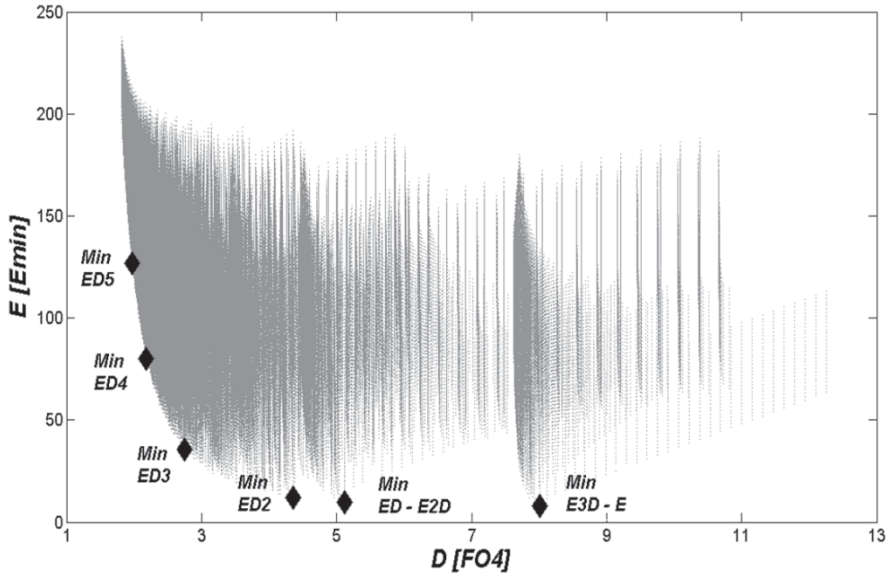


Fig. 2.7. Full $E - D$ space exploration for a buffered 2:1 multiplexer.

definition of proper design space bounds, which can be accomplished by resorting to the previously described method, has still a great importance as in the simulations based case.

When dealing with circuits featured by more than one hundred design variables, a nonlinear programming does not anymore allow to reliably determine the optimum solution of the optimization problem. Therefore, the focus must be on the adoption of the most accurate possible E-D models leading to optimization problems that can be reliably solved (i.e., assuring the global optimum is found) in a feasible time.

A class of problems that can be reliably and fast solved is the "convex optimization", where both the objective and constraint functions are convex [BV03] (see the Appendix at the end of the chapter). There is in general no analytical formula for the solution of convex optimization problems, but there are very effective methods for solving them like interior-point methods [BV03] or other custom methods. For instance, the method proposed in [JB08] is claimed to size circuit with a million gates in nearly one hour. Furthermore, thanks to the properties of the above solving methods, the definition of practical design space bounds as well as that of the initial point from which start the optimization, become irrelevant.

Hence, it is apparent that as long as the optimization problem can be formulated in a convex form, the required computational effort is incomparably lower than that required in the previous cases. The other side of the coin is that the formulation itself requires a simplification of the E-D

models that lowers the accuracy in their estimation. Nevertheless, this is the only feasible approach when the circuit size is large enough.

A class of convex optimization problems that really well suits the problem of digital gate sizing (e.g., to determine the energy-efficient designs as in our case) is that called "generalized geometric programming (GGP)", where the objective and constraint functions take the special form of "generalized posynomials" ("monomials" for the equality constraints). A brief overview on convex optimization and GGP is reported in the Appendix of the chapter, while a detailed and full mathematical treatment can be found in [BV03] and [BKP05]. In particular, in [BKM05] a comprehensive list concerning the applicability of GGPs to the design of digital circuits can be also found in. It includes:

- the minimization of energy/power (or area) of logic circuits under speed (e.g., delay, clock frequency) constraints, i.e. the energy-efficient design;
- wires sizing in RC tree networks;
- statistical optimization under PVT variations.

As previously discussed, energy and delay have to be modeled as most accurately as possible through generalized posynomials. As concerns delay, RC based models linearly including the impact of input slope, as that shown in the second paragraph, are typically adopted [FD85], [CEM99], [PK], while energy is typically modeled as proportional to gates sizes, as in (2.1).

2.4 Design of Energy-Efficient Pipelined Systems

When dealing with custom datapaths, the design of energy-efficient pipelined systems is essential to achieve the desired throughput (or clock frequency) while paying the lowest possible energy consumption.

Convex optimization methods allow to deal with any kind of digital circuit featured by several concurrent constraints, as in the case of pipelined systems. However, simply formulating the problem as (for instance) a GGP and solving it by relying upon the related mathematics, makes one lose sight of the relevant aspects pertinent to the design of an energy-efficient pipeline. In such sense, the state of the art is represented by the papers from Zyuban & Strenski's [ZS02],[ZS03] and a subsequent work [DZO06] drawing inspiration from the former ones and attempting to solve the related issues.

In this paragraph, we refer to pipelines that are made up of pipeline stages (e.g., fetch, decode, execute stages in a processor). In turn, pipeline stages are made up of circuit blocks of different complexity (e.g., a flip-flop, an adder, a multiplier, etc.). Finally, a block is constituted by a number of basic logic gates (e.g., inverters, NAND gates, NOR gates, etc.).

2.4.1 Zyuban & Strenski's hardware-voltage intensity criteria

According to (2.21), the minimum energy of a single circuit under a given delay constraint is achieved when hardware, η, and voltage, β, intensities are equal. The analysis can be extended to the cases of:

a) A composite pipeline stage made up of several blocks (see Fig. 2.8a). The speed constraint is expressed in terms of the overall stage delay, as in the case of a single circuit. However, here the energy and delay contributions from the various underlying blocks are separately targeted.
b) A multistage pipeline with composite stages (see Fig. 2.8b), i.e. various pipeline stages subject to the same delay constraint.
c) A multistage pipeline with composite stages, i.e. various pipeline stages subject to the same delay constraint, where the energy and delay contributions from the various underlying blocks are separately targeted.

a) A composite pipeline stage

In any conventional pipeline, at least two independent blocks (latches and logic) can be distinguished, and these are usually designed and tuned independently of each other. Consequently, different blocks in the same pipeline stage may have different values for the optimal hardware intensity.

Assuming the pipeline stage is made up of M blocks, one has to minimize the overall energy

$$E(w_1, w_2, \dots, w_M, v) = \sum_{i=1}^{M} E_i(w_i, v) \tag{2.35}$$

being w_i the sizes of the various blocks and v the supply voltage, under the constraint that the overall delay is equal to a given value

$$D(w_1, w_2, \dots, w_M, v) = \sum_{i=1}^{M} D_i(w_i, v) = D_r \tag{2.36}$$

The solution of the problem can be easily found by using Lagrange multipliers [DZO06], and corresponds to the condition

$$\frac{e_i}{d_i}\eta_i = \beta \quad \forall i = 1 \dots M \tag{2.37}$$

where $e_i = E_i/E$ and $d_i = D_i/D$ are the energy and delay percentages of the i-th block relative to the entire pipeline stage, η_i is the hardware intensity of the i-th block and β is the stage voltage intensity, i.e.

$$\eta_i = -\frac{\partial E_i}{\partial D_i}\frac{D_i}{E_i}\Big|_{w_i \text{ variable } / \ (w_1, w_2, \dots, w_{i-1}, w_{i+1}, \dots, w_M, v) \text{ fixed}} \tag{2.38}$$

$$\beta = -\frac{\partial E}{\partial D}\frac{D}{E}\Big|_{v \text{ variable } / \ (w_1, w_2, \dots, w_M) \text{ fixed}} \tag{2.39}$$

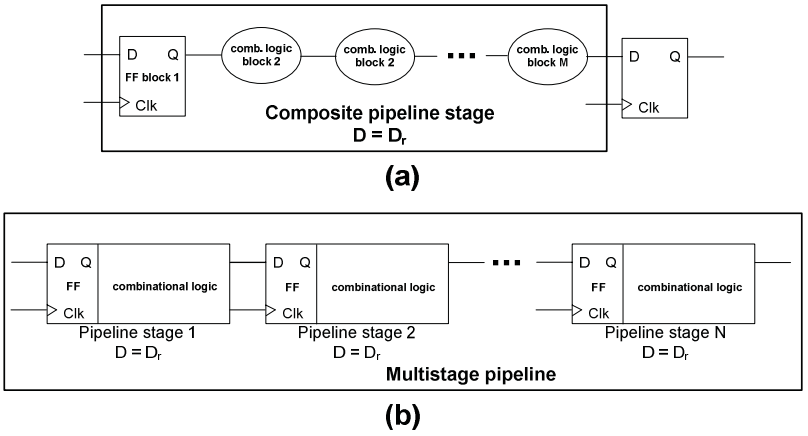


Fig. 2.8. Composite pipeline stage (a) and multistage pipeline (b).

Thus, in a pipeline stage with multiple blocks designed independently, blocks that have lower energy weight and higher delay weight should be designed more aggressively than blocks with lower delay weight and higher energy weight.

The aggregate hardware intensity of the whole pipeline stage cannot be in general related to the hardware intensities of the underlying blocks, given that one has [ZS03]

$$\frac{\partial E}{E} = -\sum_{i=1}^{M}\left[\frac{e_i}{d_i}\eta_i\frac{\partial D_i}{D}\right] \tag{2.40}$$

However, when condition (2.37) is satisfied, from (2.40) one finds that the aggregate hardware intensity of the whole pipeline stage is equal to those of the various blocks, i.e.

$$\eta = -\frac{\partial E}{\partial D}\frac{D}{E}\Big|_{(w_1,w_2,\ldots,w_M)\ variable\ /\ v\ fixed} = \frac{e_i}{d_i}\eta_i = \beta \quad \forall i = 1 \ldots M \tag{2.41}$$

b) A multistage pipeline

Practically, different stages of the pipeline usually have different amounts of complexity, and it would be incorrect to tune all of them for the same value of hardware intensity.

Assuming the pipeline is made up of N stages, one has to minimize the overall energy

$$E(W_1, W_2, \ldots, W_N, v) = \sum_{i=1}^{N} E_i(W_i, v) \tag{2.42}$$

being W_i the sizes of the various stages, under the constraint that the delays of the various stages are all equal to a given value

$$D_i(W_i, v) = D_r \quad \forall i = 1 \dots N \tag{2.43}$$

Note that each i-th stage is in turn made up of M_i blocks and hence the sizing W_i should be more properly expressed as

$$W_i = \left(w_{i,1}, w_{i,2}, \dots, w_{i,M_i}\right) \tag{2.44}$$

The solution of the problem can be again easily found by using Lagrange multipliers [DZO06], and corresponds to the conditions

$$\sum_{i=1}^{N} e_i \eta_i = \beta \quad \forall i = 1 \dots N \tag{2.45}$$

The above relationship can be used to reevaluate the choice of the power-supply voltage and the clock-cycle target, and possibly the partitioning of the pipeline into stages.

This time the aggregate hardware intensity of the whole multistage pipeline can be computed from the hardware intensities of the various stages and corresponds to the left side of (2.45) equation [ZS03], i.e.

$$\eta = -\frac{\partial E}{\partial D}\frac{D}{E}\Big|_{(W_1, W_2, \dots, W_N)\ variable\ /\ v\ fixed} = \sum_{i=1}^{N} e_i \eta_i \tag{2.46}$$

c) A multistage pipeline with composite stages

Assuming the pipeline is made up of N composite stages and the i-th stage is made up of M_i blocks, one has to minimize the overall energy

$$E\left(w_{1,1}, w_{1,2}, \dots, w_{1,M_1}, w_{2,1}, w_{2,2}, \dots, w_{2,M_2}, \dots, w_{N,1}, w_{N,2}, \dots, w_{N,M_N}, v\right) = \sum_{i=1}^{N}\left\{\sum_{j=1}^{M_i}\left[E_{i,j}\left(w_{i,j}, v\right)\right]\right\} \tag{2.47}$$

where the subscripts i and j refer to the i-th pipeline stage and to the j-th block within it, under the constraint that the overall delays of the various stages are all equal to a given value

$$D_i\left(w_{i,1}, w_{i,2}, \dots, w_{i,M_i}, v\right) = \sum_{j=1}^{M_i}\left[D_{i,j}\left(w_{i,j}, v\right)\right] = D_r \tag{2.48}$$

The solution of the problem, as in the previous cases, can be found by using Lagrange multipliers and corresponds to the conditions

$$\frac{e_{i,j}}{d_{i,j}}\eta_{i,j} = \frac{e_{i,k}}{d_{i,k}}\eta_{i,k} \quad \forall j,k = 1 \dots M_i \tag{2.49}$$

$$\sum_{i=1}^{N} \frac{e_{i,j}}{d_{i,j}}\eta_{i,j} = \beta \quad \forall j = 1 \dots M_i \tag{2.50}$$

Again, the aggregate hardware intensity of the whole pipeline stage cannot be in general related to the hardware intensities of the underlying blocks, given that one has [ZS03]

$$\frac{\partial E}{E} = -\sum_{i=1}^{N}\left\{\sum_{j=1}^{M_i}\left[\frac{e_{i,j}}{d_{i,j}}\eta_{i,j}\frac{\partial D_{i,j}}{D}\right]\right\} \tag{2.51}$$

However, when condition (2.49) is satisfied, from (2.51) one finds that the aggregate hardware intensity of the whole multistage pipeline is equal to

$$\eta = \sum_{i=1}^{N} \frac{e_{i,j}}{d_{i,j}}\eta_{i,j} \quad \forall j = 1 \dots M_i \tag{2.52}$$

2.4.2 Practical guidelines to design energy-efficient pipelines

The optimal criteria given by Zyuban and Strenski have two primary limitations: their hard-to-use coarse-tuning approach and the restricted assumption of energy and delay dependency among blocks/stages [DZO06].

Indeed, the optimal criteria are difficult to apply and their application is mainly suited for the verification of design optimality, since, given a design solution, this criteria can be used to determine if the design is optimal. However, if the design is not optimal, the criteria may suggest modifications to energy, delay, hardware intensity, or supply voltage, but it is not immediately clear how to change each of these quantities [DZO06].

The other limitation arises since these optimal criteria are derived assuming that changes in a particular block/stage do not affect the energy and delay of neighboring ones. While this assumption can be justified in coarse tuning of circuits, it is generally not true for a pipeline stage where the input (output) capacitances of each stage/block affect the performance of the preceding stage/block (of the stage/block itself). Therefore, the energy and delay dependencies between adjacent blocks/stages should be added to the previous derivations. However, due to the non-analytical form of these dependencies, their inclusion does not lead to an analytical solution [DZO06].

To partially overcome the above issues, a thorough methodology consisting of several iterative steps has been proposed in [DZO06]. This methodology targets the minimization of the energy of a multistage pipeline under a given delay constraint and without neglecting the mutual influence

between the design of the various stages. In this case, the stages are treated as unique blocks, i.e. the previous analysis relative to the energy-efficient design of a stage considered as the composition of several blocks is not considered.

The convention adopted in [DZO06] is to exclude the energy dissipation related with the charge/discharge of the input capacitance of a stage and to include that related with the charge/discharge of the output load capacitance.

The iterative procedure leading to the optimum designs of all the combined pipeline stages is based on the optimization of the stages themselves under various input/output capacitances conditions. Indeed, three different optimizations can be performed on a single stage:

1) The stage can be designed to achieve the minimum energy under a given delay constraint and with a fixed input and load capacitances. This is the problem discussed in the rest of this chapter and can be dealt with by resorting to simulations- or models-based optimizations (e.g., with generalized geometric programming). When exploring different delay constraints, an energy-efficient curve can be extracted and it reaches a well-defined minimum delay point corresponding to the Logical Effort design. This case is exemplified in the case of a 64-bit Kogge-Stone adder in Fig. 2.9 [DZO06].
2) Given the convention adopted on input capacitance related dissipation, the delay of the stage can be improved without worsening energy by simply increasing the input capacitance as shown in the case of the 64-bit Kogge-Stone adder in Fig. 2.10 [DZO06]. Obviously, such an increase negatively affects the delay of the stage preceding the considered one given that its load increases.
3) Given the convention adopted on input capacitance related dissipation, the energy of the stage can be improved without worsening delay by simply increasing the input capacitance as shown in Fig. 2.10 [DZO06]. Indeed, a larger input capacitance allows to reach the same delay with a smaller sizing (and hence a smaller dissipation) of the other gates within the stage. Obviously, such an increase negatively affects the energy of the stage preceding the considered one given that its load increases.

According to 2) and 3), for a fixed output load and a variable input capacitance an energy-efficient design region comes out and, as shown in Fig. 2.10, it is located between the minimum energy and minimum delay points [DZO06]. Given a delay constraint, the maximum and minimum values for the input capacitance are found and correspond to the minimum energy and minimum delay point in Fig. 2.10.

The key for overall energy optimization is the analysis for each stage of the sensitivities of the optimized energy to the input capacitance, C_{IN}, under a fixed output load, C_L, and to the output load under a fixed input capacitance

$$\sigma_{E,C_{IN}} = -\frac{\partial E}{\partial C_{IN}}\Big|_{C_{IN}\ variable\ /\ C_L\ fixed} \tag{2.53}$$

$$\sigma_{E,C_L} = \frac{\partial E}{\partial C_L}\Big|_{C_L\ variable\ /\ C_{IN}\ fixed} \tag{2.54}$$

Indeed, in this way one can deal with the improvement of the performance of a stage when increasing its input capacitance and decreasing its output load, and the corresponding decrease in the performance of the preceding and succeeding stages.

The general trends are shown in Fig. 2.11, where it is shown that [DZO06] the sensitivity of the optimized energy of the stage to C_{IN} under a fixed C_L asymptotically decreases for larger values of C_{IN} itself. The maximum value of C_{IN} leading to the lowest stage energy corresponds to the minimum energy point in Fig. 2.10 and increases for larger C_L. On the contrary, the sensitivity increases when moving towards the minimum delay point (Logical Effort design) by decreasing the C_{IN} value. Again, the minimum delay point is achieved with a larger C_{IN} when C_L increases. Moreover, the optimized energy of the stage under a fixed C_{IN} is a nearly linear increasing function of C_L both when considering the minimum energy and minimum delay points.

It is easy to show that, when considering a multistage pipeline, the overall minimum energy is reached when the sensitivities of the energy of all stages to their input capacitances and output loads are all equal [DZO06].

Basing on the above considerations, an iterative procedure to determine the energy-efficient sizing of a multistage pipeline comes out [DZO06]:

1) A set of initial values for the capacitances at the boundaries between the various stages is chosen.
2) The various stages are optimized for minimum energy given the delay constraint under a fixed input capacitance and output load (the capacitances at the boundary are fixed). This optimization can be performed with any of the methods discussed in this chapter.
3) The sensitivities in (2.53)-(2.54) are computed for each stage.
4) If the sensitivities are not equal for all the stages, the values of the capacitances at the boundaries between the various stages are properly updated and the procedure comes back to 2). Otherwise the energy-efficient design for the multistage pipeline is found and procedure ends.

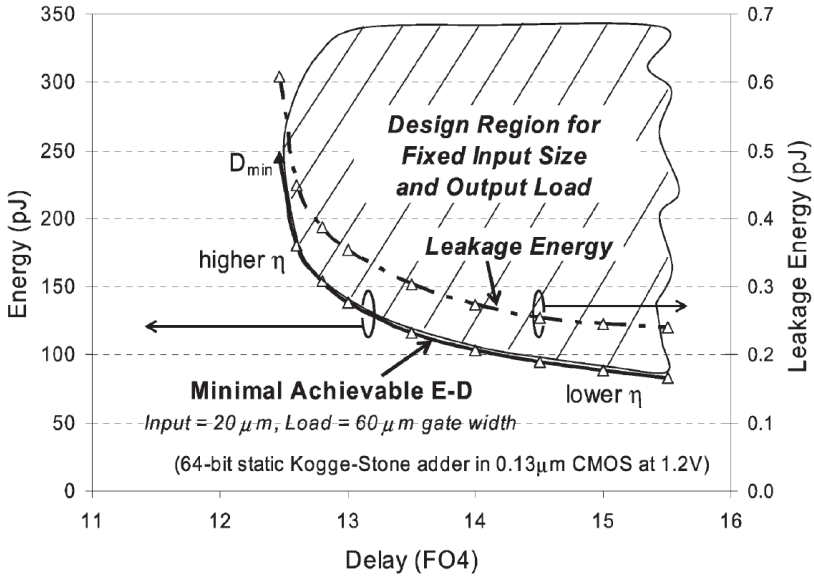


Fig. 2.9. 64-bit Kogge-Stone adder: energy-delay optimization under fixed input capacitance and output load [DZO06].

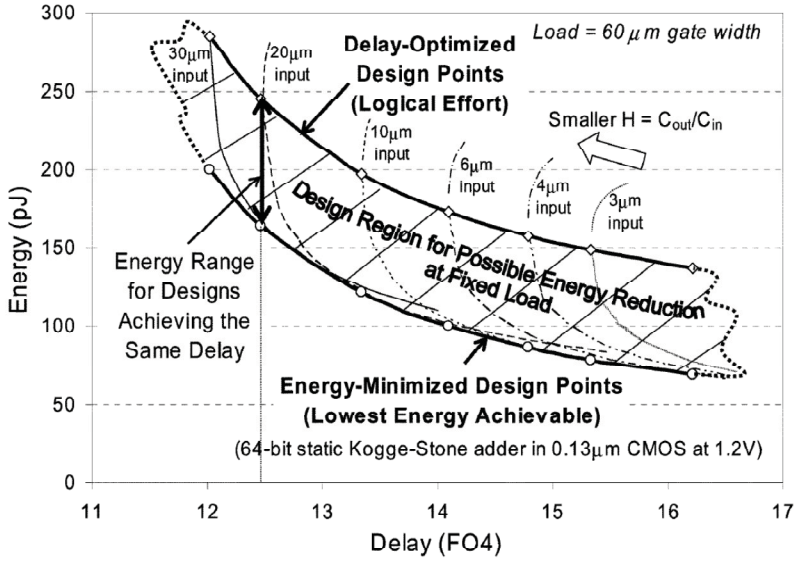


Fig. 2.10. 64-bit Kogge-Stone adder: design region for possible energy-delay reduction under varying input capacitance and fixed output load [DZO06].

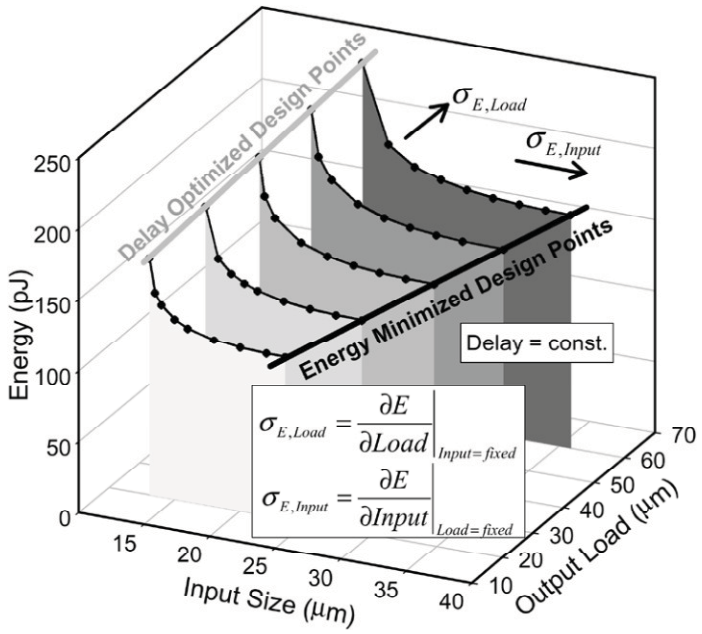


Fig. 2.11. Optimized energy of a pipeline stage versus input capacitance under fixed load and versus load under fixed input capacitance [DZO06].

Appendix 2
Convex Optimization

When dealing with circuits of large size, that is to say featured by more than one hundred design variables, a nonlinear programming does not anymore allow to reliably determine the optimum solution of the optimization problem. Since a nonlinear programming and the related solving algorithms were necessary because of the complexity of quite accurate E-D models, the usage of the latter ones cannot be afforded anymore. On the contrary, the focus must be on the adoption of the most accurate possible E-D models leading to optimization problems that can be reliably solved (i.e., assuring the global optimum is found) in a feasible time.

Two classes of optimization problems that can be reliably and fast solved are "least-squares" problems, where there are no constraints and the objective function is a sum of square terms, and "linear programming", where both the objective and constraint functions are linear [BV03]. Both cases are special subclasses of "convex optimization", where both the objective and constraint functions are convex [BV03]. In short, by defining the objective function $f_0: \mathrm{R}^n \rightarrow \mathrm{R}$ and the constraint functions $f_i: \mathrm{R}^n \rightarrow \mathrm{R}$ for $i = 1 \ldots m$, a convex optimization problem is one of the form [BV03]

$$\begin{array}{ll} minimize & f_0(x) \\ subject\ to & f_i(x) \le b_i \quad i = 1 \ldots m \\ where & f_i(\alpha x + \beta y) \le \alpha f_i(x) + \beta f_i(y) \end{array} \tag{A.2.1}$$

for all $x, y \in \mathrm{R}^n$ and all $\alpha, \beta \in \mathrm{R}$ with $\alpha + \beta = 1$ and $\alpha, \beta \ge 0$.

There is in general no analytical formula for the solution of convex optimization problems, but, as with linear programming, there are very effective methods for solving them like interior-point methods [BV03] or other custom methods [JB08] allowing to easily solve problems with hundreds of variables and thousands of constraints on a current desktop computer, in at most a few tens of seconds. For instance, the method proposed in [JB08] is claimed to size circuit with a million gates in nearly one hour. Furthermore, thanks to the properties of the above solving methods, the definition of practical design space bounds as well as that of the initial point from which to start the optimization, become irrelevant.

Hence, it is apparent that as long as the optimization problem can be formulated in a convex form, the required computational effort is incomparably lower than that required in the previous cases. The other side of the coin is that the formulation itself requires a simplification of the E-D models that lowers the accuracy in their estimation. Nevertheless, this is the only feasible approach when the circuit size is large enough.

A class of optimization problems that really well suits the problem of digital gate sizing (e.g., to determine the energy-efficient designs as in our case) is that called geometric programming [BKM05].

Given a vector $x = (x_1, x_2, \ldots, x_n)$ of positive optimization variables (e.g., transistors sizes), a function g of the form

$$g(x) = c {x_1}^{\alpha_1} {x_2}^{\alpha_2} \cdots {x_n}^{\alpha_n} \tag{A.2.2}$$

where $c > 0$ and $\alpha_i \in \mathrm{R}$, is called a "monomial" [BKP05]. Monomials are closed under product, division, positive scaling, power, inverse. A sum of monomials, i.e.

$$f(x) = \sum_{k=1}^{t} c_k {x_1}^{\alpha_{1k}} {x_2}^{\alpha_{2k}} \cdots {x_n}^{\alpha_{nk}} \tag{A.2.3}$$

where $c_k > 0$ and $\alpha_{ik} \in \mathrm{R}$, is called a "posynomial" [BKP05]. Obviously, a monomial is also a posynomial. Posynomials are closed under sum, product, positive scaling, product and division by monomial and positive integer power. A "generalized posynomial" can also be introduced as a function that can be obtained also by taking the positive fractional power of posynomials or the maximum of some posynomials [BKP05]. Generalized posynomials are hence closed under sum, product, positive scaling, product and division

by monomial, positive power and maximum.

A "(generalized) geometric programming", (G)GP is an optimization problem with the form [BKP05]

$$\begin{array}{ll} minimize & f_0(x) \\ subject\ to & f_i(x) \le 1 \quad i = 1 \dots m \\ & g_i(x) = 1 \quad i = 1 \dots p \end{array} \tag{A.2.4}$$

where f_i are (generalized) posynomials and g_i are monomials. Several extensions, such as $f(x) < g(x)$, $f(x) < a$ or $g_1(x) = g_2(x)$ constraints, can be readily handled. It is also possible to maximize a nonzero monomial objective function, by minimizing its inverse (which is also a monomial).

GPs are not convex optimization problems. However, they can be converted into nonlinear but convex optimization problems basing on a logarithmic change of variables and a logarithmic transformation of the objective and constraint functions [BV03], [BKP05]. Variables x_i are substituted by $y_i = \log(x_i)$ and the log of posynomials and monomials is taken leading to the following optimization problem

$$\begin{array}{lll} minimize & \log[f_0(e^{y_i}, e^{y_2}, \dots, e^{y_n})] & \\ subject\ to & \log[f_i(e^{y_i}, e^{y_2}, \dots, e^{y_n})] \le 0 & i = 1 \dots m \\ & \log[g_i(e^{y_i}, e^{y_2}, \dots, e^{y_n})] = 0 & i = 1 \dots p \end{array} \tag{A.2.5}$$

where the objective and inequality constraints are now smooth convex functions and the equality constraint are now affine functions, i.e. linear plus a constant.

Chapter 3

CLOCKED STORAGE ELEMENTS

This chapter deals with the theory of clocked storage elements, i.e. latches and flip-flops. The crucial role played by these circuits within synchronous digital systems, which can be as complex as microprocessors, is first outlined. Then the operating principles, the most important properties and parameters characterizing clocked storage elements and a brief description of the main topological classes are reported. Much of this discussion is an excerpt from [OSM03].

3.1 Clocking in Synchronous Digital Systems

The concept of clocking and the related issues are among the most important aspects in the design of a synchronous digital system. The latter always comprises synchronous memory elements and combinational logic, which together build finite state machines [OSM03]. In a finite state machine each event is dictated by the changes that happen on input signals and/or on the clock signal, which is signal providing synchrony. In particular, as depicted in Fig. 3.1, the next state, S_{T+1}, is a function of the present state, S_T, and of the value of the N input signals, $X_{[1-N]}$, while the clock, in conjunction with the mode of operation of synchronous memory elements, henceforth Clocked Storage Elements (CSEs), determines "when" the change of the state and of the M outputs, $Y_{[1-M]}$, have to occur [OSM03].

Therefore, the clock provides the temporal references for state and outputs transitions and hence regulates the flow of data. This is the basic working principle of a synchronous digital system. In the following, unless explicitly stated, let us assume the rising clock transition as the above described temporal reference [PCB01]. For the sake of completeness, it is

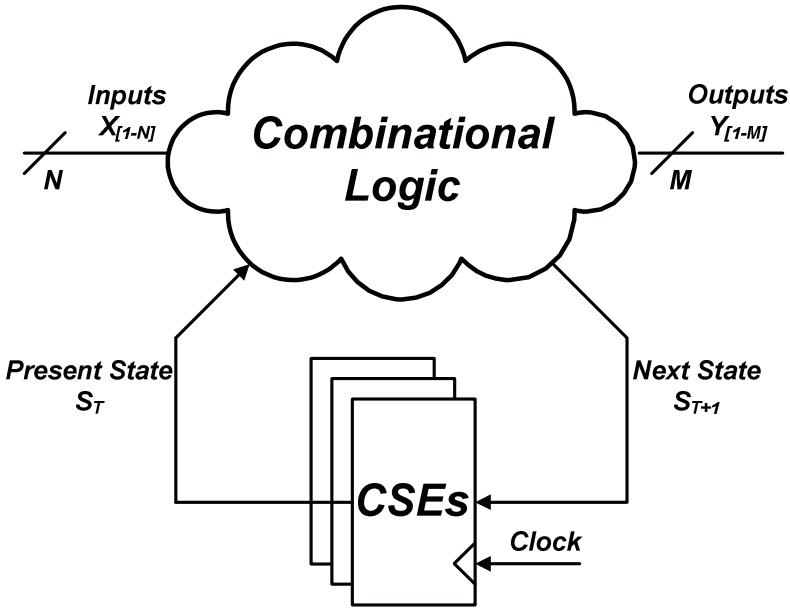


Fig. 3.1. Finite state machine.

worth noting that an asynchronous approach, where a clock signal is not present, can also be adopted. Although this approach could in principle bring to a reduction of some sources of energy consumption, its practical robustness and performances are not comparable to those of the synchronous approach.

It often happens that the delay through a complex combinational logic block is excessive and, starting from a rising clock edge, the changes on its input signals (S_T and $X_{[1-N]}$) do not have the time to propagate through the block and the CSEs before the successive rising clock edge occurs. In such case it is said that the "critical path" requirements are not satisfied or also that a critical path violation has happened, thereby invalidating the system functionality given that the inputs and present state do not have the correct effect on the next state and outputs [OSM03]. Since there are many concurrent paths within the combinational logic, the critical path is in general defined as the longest/slowest one.

To solve the above issue, the finite state machine is split into subsequent "stages" so that all the transitions launched from the outputs of the CSEs of stage i reach the inputs of the CSEs of stage $i + 1$ in a time shorter than a clock period [OSM03]. This approach is called "pipelining" and by extending the diagram of a finite state machine into subsequent pipeline stages the graphical representation in Fig. 3.2 is obtained.

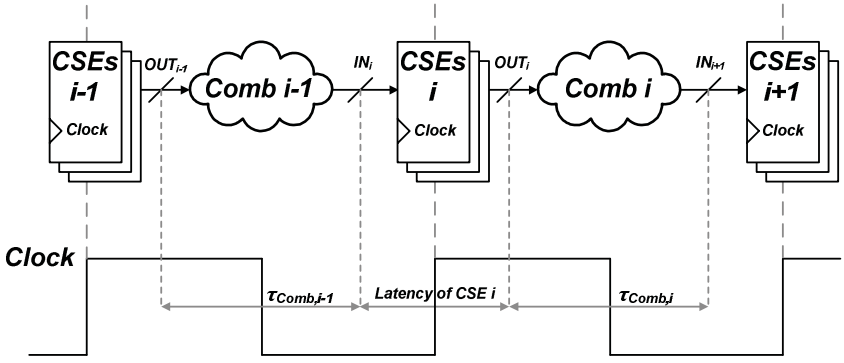


Fig. 3.2. Pipelining.

To meet the specific speed demands in microprocessors (and in digital systems in general), the duration of the clock period, which is the time interval required to execute simple instructions, decreases over time. According to the previous argumentation, there is a consequent increase in the number of pipeline stages (deep pipeline) [O02]. This in turn causes a reduction in the number of combinational logic stages in a single pipeline stage, i.e. the number of gates between two CSEs. The graph in Fig. 3.3 shows an increase over time of the clock frequencies used in some representative microprocessors and, correspondingly, the decrease in the number of logic levels in a pipeline stage [O02].

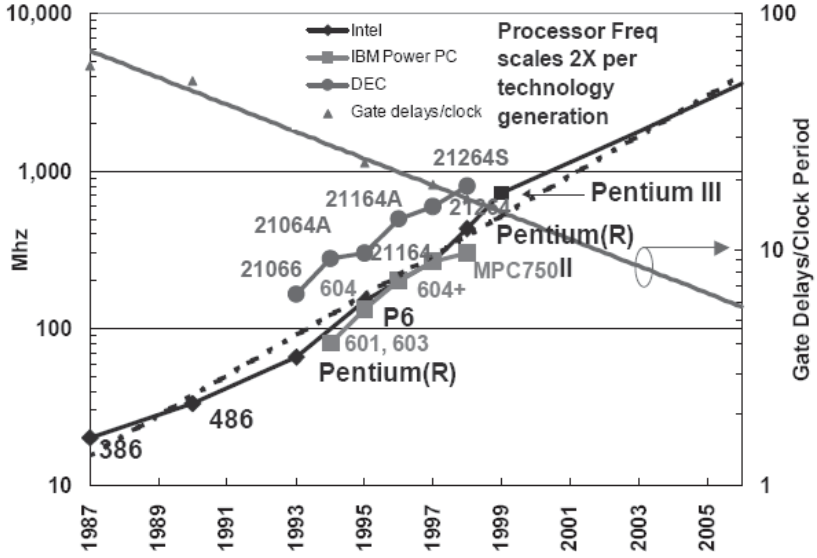


Fig. 3.3. Clock frequencies and logic depths in microprocessors [O02].

As a consequence, the fraction of the clock period that is occupied by latency of CSEs proportionally grows. It is hence easy to understand that all the timing overheads related to clocking and to the delay of CSEs strongly influences the performances and design of deeply pipelined microprocessors.

3.2 Features of the Clock Signal

Although the issues related to clock generation and distribution are not treated here, given that they lie outside the purposes of this work, is necessary to describe some of the main properties featuring the clock signal, which are tightly correlated with the usage and performances of CSEs.

A first aspect is the so called "clock scheme", which can be "single phase" or "multi-phases" [OSM03]. Data transfer through CSEs is in general obtained using an "active phase" (clock equal to ′1′ or ′0′) or a specific "active" edge of the clock (rising or falling). In order to prevent that data propagate over a desired point and to avoid an undesired transparency, the clock phases are separated in time and they are said to be not overlapping. In the past multi-phases clock scheme were used and the CSEs were synchronized by pulses of short duration. With increasing clock frequencies, the feasibility of such a solution becomes prohibitive because it is more and more difficult to control the temporal relationships between the various clock phases and hence to properly distribute them. Therefore, the single phase scheme is preferred for the distribution of the clock signal throughout the system and, where necessary, two phases can be generated locally by the use of appropriate circuits. In Fig. 3.4 the different schemes are shown.

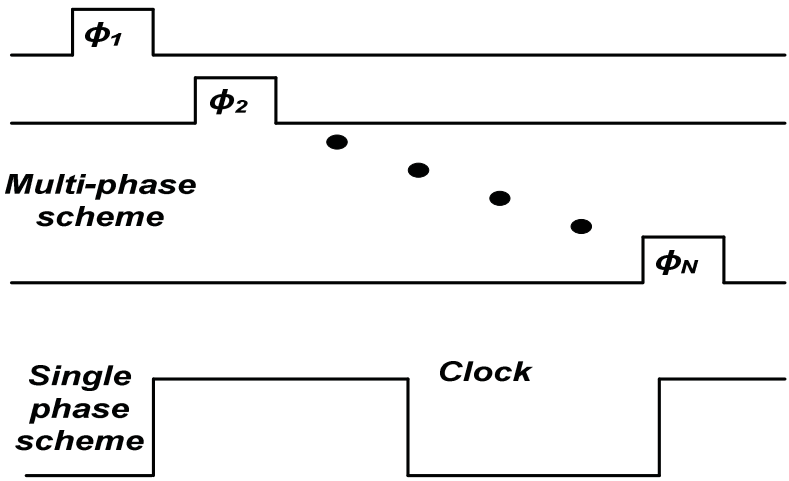


Fig. 3.4. Multi-phases and single-phase clocking schemes.

Other parameters characterizing the clock signal are its period (frequency), T_{CK} ($f_{CK} = 1/T_{CK}$), the clock width, W_{CK}, which is the duration of the time window in which the clock is said to be active, the duty cycle, W_{CK}/T_{CK} (typically equal to 50%) and its rising/falling times, t_r/t_f. Fig. 3.5 depicts the above quantities.

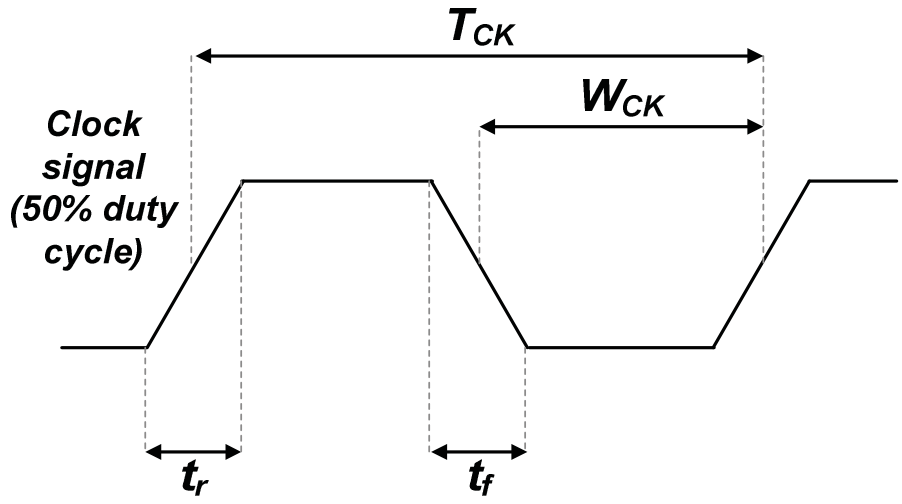


Fig. 3.5. The clock signal and its parameters.

Two other important parameters are the clock "skew" and "jitter", since they represent sources of uncertainty [RCN03].

The skew is a spatially distributed (within the system) delay between the clock signal at any point of the system and a reference point, where, for example, the clock distribution network starts. Clock skew hence manifests as clock is distributed throughout the system and is due to the different characteristics of the distribution paths and to the different loading effects seen by the clock signal at different points. The "global skew" is defined as the maximum delay among the clock signal versions synchronizing any two CSEs in the whole system. The "local skew" is relative to two adjacent CSEs and is particularly meaningful for the possible problems of "data race-trough", i.e. the anticipated and undesired transmission of between two adjacent CSEs that have no (or very small) combinational logic between them.

The jitter is instead is instead a temporally and randomly distributed variation of the times when clock signal transitions occur with respect to the ideal reference times. The "edge-to-edge jitter" is the variation relative to two consecutive clock edges and mainly influences the constraints related to the CSEs delays, while the "long-term jitter" refers to a large number of

clock cycles. The jitter is due to process variations and non-idealities especially related to the clock generation circuitry. Note that jitter translates into variations of the values of clock width and duty cycle thereby further contributing to overall uncertainty.

Clock skew and jitter are depicted in Fig. 3.6.

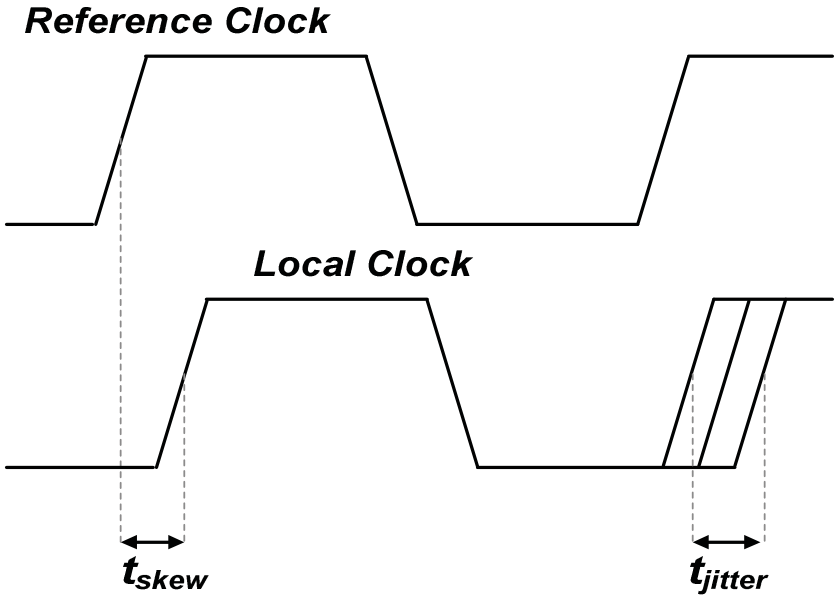


Fig. 3.6. Clock skew and clock jitter.

3.3 Clocked Storage Elements: Latches, Master-Slave Flip-Flops and Pulsed Topologies

The simplest feasible memory element is made up by two inverters connected back to back. The positive feedback allows to preserve the stored bit and hence this simple structure, shown in Fig. 3.7a, is called a "keeper" [RCN03]. The only way to change the stored bit is to "force" a change in the logic level on one of the two nodes for a time sufficient for the extinction of the current contention that is temporarily created.

To avoid the energy consumption and the robustness issues related to this operation, it is necessary to introduce additional nodes that help to change the stored logic state and this can be simply done by replacing the inverters with NAND or NOR gates, thus realizing a so called "Set-Reset" memory element, as shown in Fig. 3.7b. The Boolean equations that relate the next output value, Q_{n+1}, to the present output, Q_n, and to the inputs S and R can be straightforwardly derived. By focusing on the NAND implementation in

Fig. 3.7b, note that by setting the inputs to complementary values, the stored bit can be changed, while setting both of them to $'1'$ the bits are maintained (the $[0,0]$ configuration is instead forbidden).

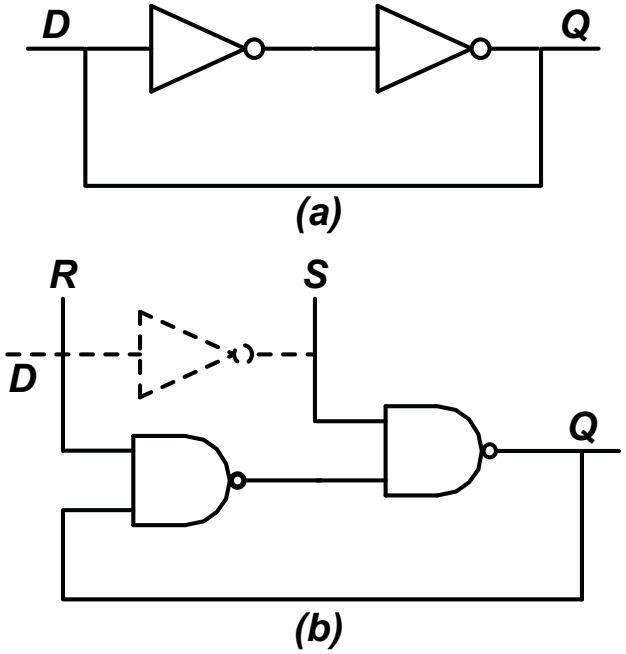


Fig. 3.7. Keeper (a) and NAND-based Set-Reset (b) memory element.

The Set-Reset memory element just described changes the value of its output according to the sole inputs values and hence, in order to translate this structure in one compatible with a synchronous design, it is necessary to introduce additional logic that allows to change the output only during a specific clock phase or edge.

The simple clocked "D latch" is shown in Figure 3.8a and requires the addition of two NAND gates driven by the complementary inputs and by the clock signal. Basically, this circuit is transparent to the input D-value (meaning that this value is transferred to the output Q) only when the clock is equal to $'1'$. The transparency during an entire active clock phase is the essence of a latch. Another implementation of a D latch, shown in Fig. 3.8b, is based on the employment of a clocked transmission gate and a keeper to maintain the stored data when the latch is opaque.

Although it is possible to build synchronous systems by using only latches [H00], the typical approach is to employ CSEs that capture the input data (ideally) on the occurrence of a single instant. Such an instant can

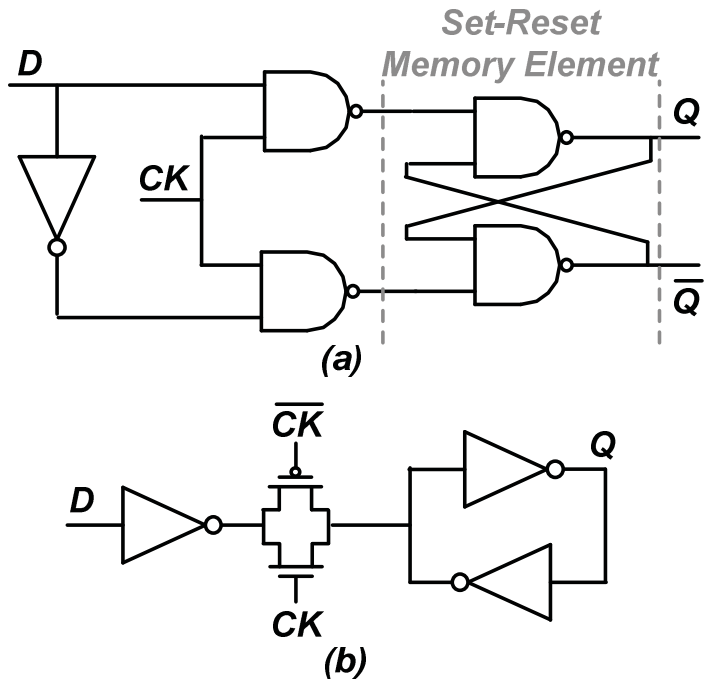


Fig. 3.8. Clocked D latches: NAND- (a) and transmission-gate (b) based.

coincide with the rising or falling clock edge and the arising temporization strategy is hence called "Positive or Negative Edge-Triggered" [PCB01]. CSEs that are (ideally) transparent only during a clock edge belong to two main classes: Master-Slave CSEs, often said Flip-Flops (FFs), and Pulsed CSEs.

A Master-Slave FF, shown in Fig. 3.9, is made up by two latches that are alternately enabled by two complementary clock signals [OSM03]. For instance the first latch, called the Master, is transparent during the low clock phase ($CK = '0'$) while the second latch, called the Slave, is transparent during the high clock phase ($CK = '1'$). In this way the input D is transferred to the Master output as long as $CK = '0'$ while the Slave is opaque and maintains its output (which is the global flip-flop output, Q). At the rising clock edge, the Master becomes opaque and the Slave captures the Master output, which is ideally equal to the D value in correspondence of the rising clock edge. Summarizing, the whole Master-Slave circuit is transparent to the input D only during an instant, the rising clock edge, i.e. it is a Positive Edge-Triggered FF. The sensitivity of the Master to the D-value during an entire clock phase (the Master is a latch) can cause energy dissipation if D makes several transitions, but does not invalidate the correct operation as

long as (ideally) the desired D value is stable just before the rising clock edge. Actually, it is the Master output that has to be stable during the rising clock edge and, as subsequently shown, this translates in a so called setup time requirement for the input D. As concerns the requirement of two clock phases, actually a single clock phase, CK, is normally distributed and its complementary version, $\overline{CK}$, is only locally generated as shown in Fig. 3.9. In order to ensure a proper operation, CK and $\overline{CK}$ must not be overlapped in order to avoid a direct bit transfer from D to Q in instants different from the desired clock edge.

One of the first Master-Slave circuit to be ever proposed is the C^2MOS FF [SOA73], which is however outperformed by the classic Transmission-Gate FF (TGFF) reported in [GGD94], [MNB01]. A classification of several Master-Slave FFs can be found in [KB95], [KB00]. A few different Master-Slave topologies with good energy-efficiency are deeply investigated in Chapters 4 and 5. In general, Master-Slave topologies do not exhibit an high speed but are featured by a low energy consumption thanks to the absence of nodes (e.g., the precharged ones) that commutate independently from data switching activity.

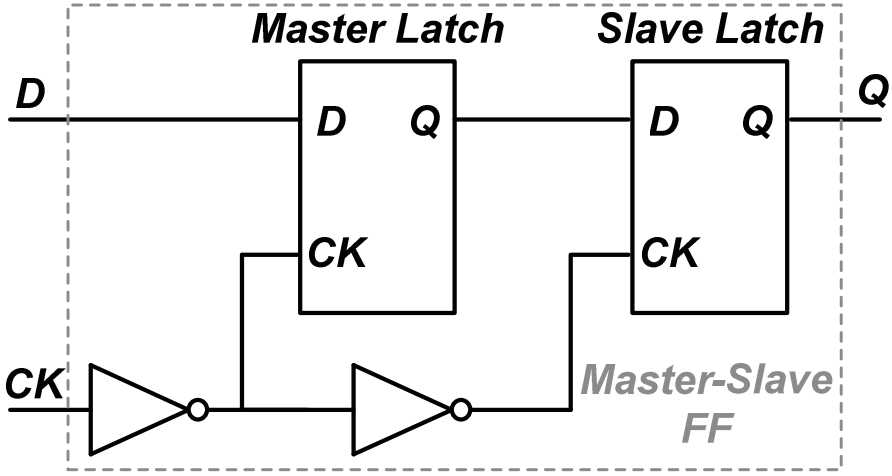


Fig. 3.9. Master-Slave CSEs (FFs).

A variant of the Master-Slave scheme was proposed in [AS90][YS89][YS96], which is conceptually a Master-Slave structure but does not need two complementary clock phases. Indeed, by mixing static and dynamic logic gates and evaluation pull-up and pull-down paths, only a single clock phase is needed and for this reason this technique is called "True-Single-Phase-Clock (TSPC)". Despite of the advantage of actually working with a single clock phase (e.g., because one does not need to worry

about the disoverlap between different phases), TSPC FFs exhibit lower performances with respect to other Master-Slave topologies and hence are not considered in the rest of the work.

A conceptually (but not necessarily practically) simpler implementation than a Master-Slave CSE is a Pulsed CSE. Actually, one can distinguish among "Explicitly Pulsed" and "Implicitly Pulsed" CSEs.

As shown in Fig. 3.10, Explicitly Pulsed CSEs are made up of a simple latch synchronized by a pulsed clock, i.e. a clock signal featured by a small clock width (and duty cycle), and are hence also defined "Pulsed Latches". The "Pulse Generator" circuit generates a clock pulse of arbitrary duration in correspondence of the desired clock edge and hence the latch is transparent only during a narrow time window [K96]. Since the Pulse Generator can be responsible for significant area and energy overheads it can be shared among several latches. However, it is only locally possible given that it is not possible to distribute a pulsed clock in a large area since it would be filtered by the resistive and capacitive parasitics of the distribution network. The Explicitly Pulsed approach is certainly topologically simpler than the Master-Slave one and allows to reach higher speed thanks to the small latch delay. However, the criticality of the design is moved to the Pulse Generator since the transparency window must be large enough to ensure the capture of the correct D-values but this comes at the cost of an increased risk of data race-through. Dealing with this tradeoff is particularly difficult because of the impact of process variations.

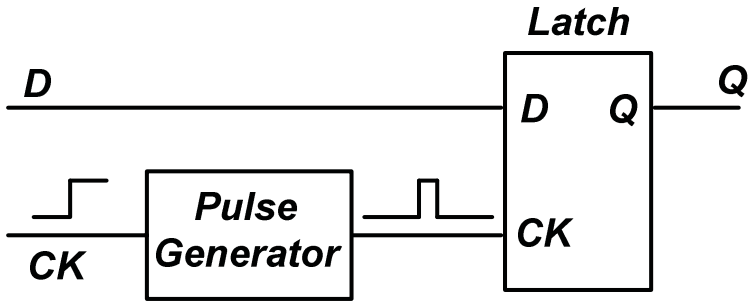


Fig. 3.10. Explicitly Pulsed CSEs (Pulsed Latches).

An Implicitly Pulsed CSE (Fig. 3.11) is conceptually constituted by a first stage that generates a pulse, which is a function of D values and CK transitions, and this pulse is then captured by a latch or by a non-clocked asynchronous memory element. Again, overall, the behaviour is quite similar to that of a Pulsed Latch given that the CSE is transparent during a regulable time window, but they are more typically referred as an "Hybrid Latch / Flip-Flop" since they integrate the Pulse Generator within the stage driven

also by the D input [PBS96]. There are several different Implicitly Pulsed CSE topologies, but all share the same above described data capturing mechanism. Compared with the Explicitly Pulsed solution, Implicitly Pulsed CSEs have typically slightly more complex topologies.

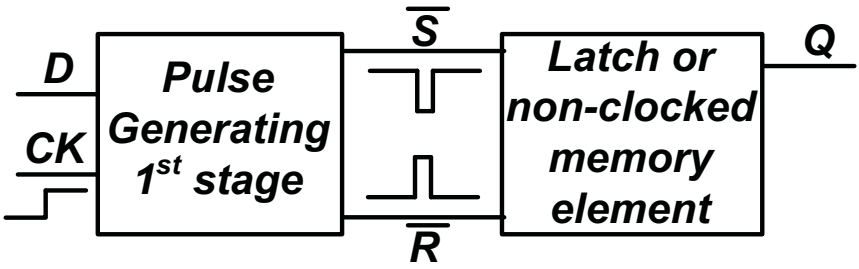


Fig. 3.11. Implicitly Pulsed CSEs.

A classification of several Explicitly and Implicitly Pulsed CSEs can be found in [TNC01] and [ZDB02]. Moreover, in Chapters 4 and 5 several Pulsed topologies are deeply investigated. In general, Pulsed CSEs are faster than Master-Slave CSEs but are featured by an higher energy consumption due to the presence of precharged nodes and pulse generators that always commutate independently from data switching activity.

3.4 Timing Features of Clocked Storage Elements

Formally, both latches and Master-Slave/Pulsed circuits are CSEs. However, according to the previous considerations, the timing operation of latches is different from that of Master-Slave or Pulsed CSEs. Nevertheless, when identifying in the following Master-Slave and Pulsed CSEs timing parameters, it is possible to extend the definitions also to latches.

For a Master-Slave or Pulsed CSEs, henceforth simply referred as CSEs, the clock-to-output delay, τ_{CQ}, is the time measured from the clock edge enabling the data, D, acquisition and the output, Q, transition. τ_{CQ} is measured by considering the instants when the clock, CK, and Q reach 50% of their swing (typically $V_{DD}/2$, being V_{DD} the power supply).

The Q transition (τ_{CQ}) occurs in a finite time (has a finite value) as long as certain timing constraints are satisfied concerning the timing relationship between CK and D, i.e. according to the clock-to-data delay, τ_{CD} (the latter is often considered in its negative version, i.e. by referring to data-to-clock delay, τ_{DC}). These timing constraints are called “setup” and “hold” requirements and, accordingly, specific τ_{CD} values defined as setup time, t_{setup}, and hold time, t_{hold}, are identified [OSM03]. Accordingly, it can be

said that, in order to transfer D to Q and have a finite τ_{CQ}, D has to remain stable to the new value that has to be captured at least for a time t_{setup} (t_{hold}) before (after) the CK edge [RCN03]. Actually, this is an hard and somewhat unsuitable definition for t_{setup} and t_{hold}. Indeed, let us consider the trend of τ_{CQ} vs. $\tau_{CD} = -\tau_{DC}$ for a Master-Slave FF in Fig. 3.12.

By figure inspection, τ_{CQ} has two (equal) horizontal asymptotes defining its minimum value, $\tau_{CQ,min}$, when D is setup to the desired value much time before CK and it is hold to this value for a long time after the CK edge [OSM03]. As long as D is made vary closer to the CK edge (both from setup and hold perspectives), τ_{CQ} slowly increases at the beginning until a critical timing region is entered where τ_{CQ} is much more sensitive to τ_{CD} value and starts to increase in a much faster way [OSM03]. Then a so called meta-stable region is entered where τ_{CQ} approaches infinity, i.e. the CSE does not capture the D value anymore [OSM03].

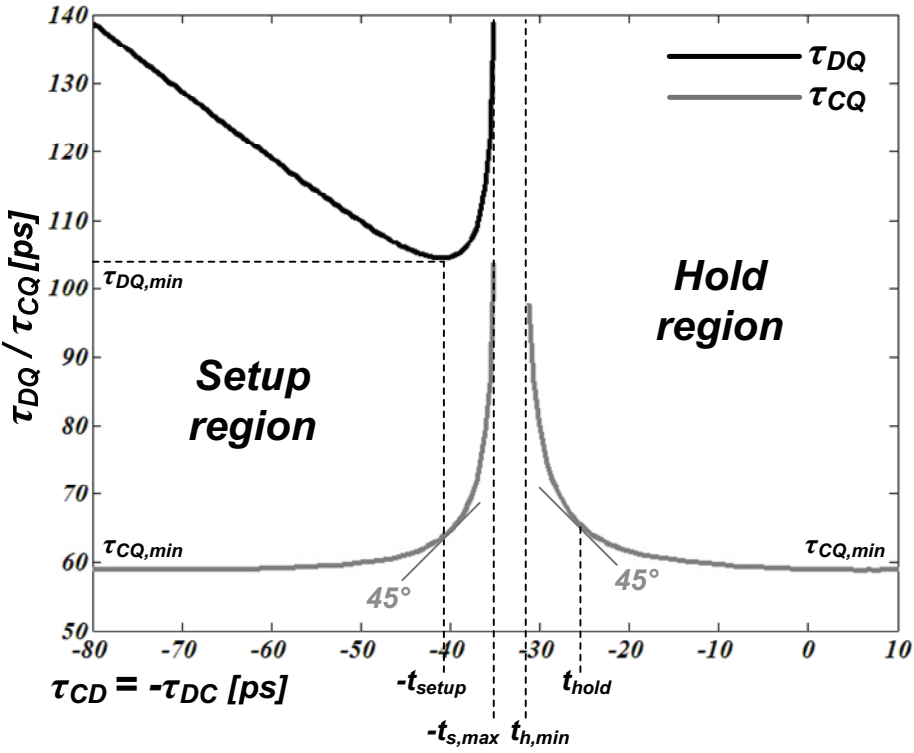


Fig. 3.12. τ_{CQ} / τ_{DQ} vs. $\tau_{CD} = -\tau_{DC}$ timing curves in a Master-Slave FF.

Note that, from setup perspective, increasing τ_{CD} over the metastable region does not cause τ_{CQ} to be brought back to finite values as could be wrongly inferred from Fig. 3.12. Indeed, if D assumes the desired values after the setup vertical asymptote, $\tau_{CD} = t_{s,max}$, the setup constraint is violated and the CSE will never transfer the desired D value to Q. Basically, this means that the setup and hold constraints have to be separately described (although contemporarily satisfied).

In particular, the background condition to describe the hold constraint is that the setup one has to be preliminary satisfied, i.e. D has to assume its new desired value a time $t_{s,max}$ before the CK edge and then one can examine what happens when D is made vary again, i.e. when D is brought back to the value already stored in the CSE. Equivalently, the D transition defining τ_{CD} in the hold region is opposite to that defining τ_{CD} in the hold region (e.g., assuming a $'0'$ is stored in the CSE, D first goes to $'1'$ with $\tau_{CD} < t_{s,max}$ and then comes back to $'0'$). This can be intuitively justified given that the setup constraint comes before the hold one in terms of temporal sequence and hence its preliminary satisfaction is a necessary condition to discuss the hold requirement. From Fig. 3.12, once the setup constraint is satisfied, if D is made change again to its old value too soon, i.e. before the hold vertical asymptote with $\tau_{CD} < t_{h,min}$, then Q will be left unchanged (τ_{CQ} becomes infinite). To exemplify the above discussion, a timing diagram is depicted in Fig. 3.13 to show conditions where setup and hold constraints are satisfied or not in a positive edge-triggered CSE.

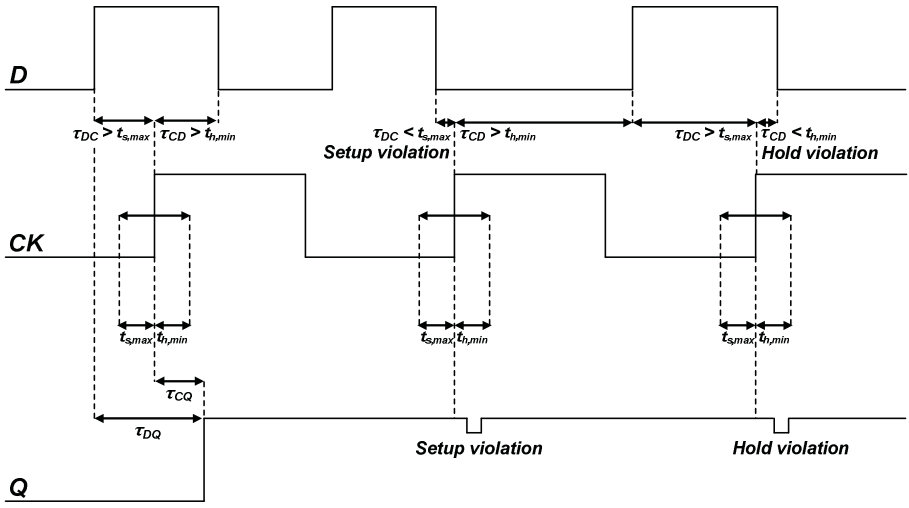


Fig. 3.13. Timing diagram and setup/hold times violations in a positive edge-triggered CSE.

According to the vertical asymptotic trend shown in Fig. 3.12, it is now clear why defining t_{setup} and t_{hold} according to the finite τ_{CQ} condition (i.e. t_{setup} slightly before $t_{s,max}$ and t_{hold} slightly above $t_{h,min}$) is not an effective choice. Indeed, the sensitivity of τ_{CQ} close to $t_{s,max}$ and $t_{h,min}$ leads to unacceptably low speed performances and to transition failures implying a very low yield given the unavoidable process variations [OSM03]. This means that another definition, targeting the speed issue as a primary concern, is required for t_{setup} and t_{hold}.

To this purpose, let use consider the impact that a CSE has on the period of the clock employed in the system where the CSE is inserted. According to the pipelined scheme, the minimum clock period requirement is determined by the latency of combinational logic and CSEs. The latency of a CSE is in turn the sum of the setup time, i.e. of a certain $\tau_{DC} = -\tau_{CD}$ value and of τ_{CQ} [SO99]. The sum of these two delays is the data-to-output, τ_{DQ}, delay, whose trend versus $\tau_{CD} = -\tau_{DC}$ was also depicted in Fig. 3.12. By figure inspection, while τ_{CQ} has a monotonic trend versus τ_{DC}, τ_{DQ} exhibits a minimum value, $\tau_{DQ,min}$ for a well specified τ_{DC} value. This is because, starting from the left of the curve, τ_{CQ} does not significantly increase as long as τ_{DC} is decreased and hence their sum also decreases. At a certain point, τ_{CQ} will start to significantly increase and hence a minimum τ_{DQ} is reached. The slope of the τ_{CQ} curve in correspondence of the τ_{DC} value determining $\tau_{DQ,min}$ is $45°$ since one has that

$$\left.\frac{\partial \tau_{DQ}}{\tau_{DC}}\right|_{\tau_{DQ}=\tau_{DQ,min}} = \left.\frac{\partial(\tau_{DC}+\tau_{CQ})}{\tau_{DC}}\right|_{\tau_{DQ}=\tau_{DQ,min}} = 1 + \left.\frac{\partial \tau_{CQ}}{\tau_{DC}}\right|_{\tau_{DQ}=\tau_{DQ,min}} \Rightarrow$$
$$\left.\frac{\partial \tau_{CQ}}{\tau_{DC}}\right|_{\tau_{DQ}=\tau_{DQ,min}} = -1 \qquad (3.1)$$

After this minimum point, due to the rapid τ_{CQ} increase also τ_{DQ} will increase and tend to infinity for $\tau_{DC} = -\tau_{CD} = t_{s,max}$.

Since the CSE latency from clock period perspective is determined by τ_{DQ}, it is reasonable to define t_{setup} as the τ_{DC} value leading to $\tau_{DQ,min}$ [SO99]. This definition is the most suitable one when designing high-performance circuits since, once (3.1) condition is satisfied, the maximum clock frequency is achieved. More specifically, by considering a pipeline made up of combinational logic blocks separated by CSEs, the minimum T_{CK} is given by

$$T_{CK} > \tau_{logic,max} + \tau_{DQ,min} + t_{skew/jitter} \qquad (3.2)$$

being $\tau_{logic,max}$ the maximum among the delays of the various combinational logic blocks and $t_{skew/jitter}$ the worst case (skew and jitter are random parameters) value of clock skew and jitter. If (3.2) is satisfied, each pipeline stage succeeds to perform the required computation in a clock cycle, or, equivalently, the setup time is satisfied. Note that a setup time violation does not imply an incorrect operation but "only" a slower speed of the system since some operations will require two clock cycles instead of one to be completed.

The above criterion for setup time definition can be employed also for hold time, i.e. t_{hold} is related to the τ_{CD} value leading a 45° slope in the τ_{CQ} versus τ_{CD} hold characteristic [NWO04]. Actually, by convention t_{hold} is defined as the τ_{CD} value that leads to 45° slope.

While the setup time impact on performances can be easily understood by analyzing its effect on the minimum clock period requirement, the hold time does not have an analogous impact on speed performances but rather on the correct operation of the pipeline itself. For instance, let us consider two adjacent CSEs (A and B) separated by a block of combinational logic with τ_{logic} delay as shown in Fig. 3.14.

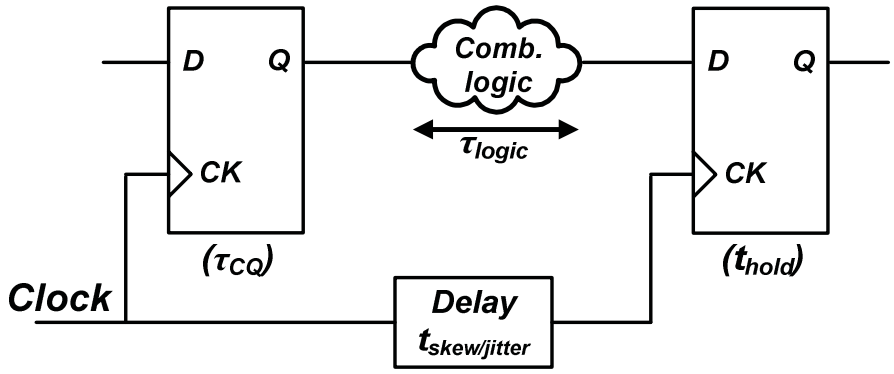


Fig. 3.14. Pipeline stages in a datapath.

If the sum of τ_{logic} and τ_{CQ} of CSE A is less than t_{hold} of CSE B, then the latter will not capture the correct data, i.e. instead of transferring D_A on Q_A and D_B (i.e., the previous Q_A elaborated through combinational logic) on Q_B in correspondence of the clock edge, one has that D_A would be directly (after combinational logic elaboration) transferred to Q_B in correspondence of a single clock edge. This condition is called a "critical race" or a "data race-through" and implies an incorrect pipeline operation. The condition that has to be hence satisfied is

$$\tau_{CQ} + \tau_{logic} > t_{hold} + t_{skew/jitter} \tag{3.3}$$

being $t_{skew/jitter}$ the worst case (skew and jitter are random parameters) value of clock skew and jitter.

From the above discussion, it is clear that the hold time requirement is critical for logic paths with small or no combinational logic between CSEs, i.e. for "fast paths" as opposed to critical paths that can be conversely affected by setup time violations. Expression (3.3) allows to determine a minimum combinational logic delay between CSEs given by

$$\tau_{logic,min} > t_{hold} + t_{skew/jitter} - \tau_{CQ} \tag{3.4}$$

where, obviously, if $\tau_{logic,min} < 0$ it means that no logic delay is needed to avoid hold time violations. Moreover (3.3) allows to define an additional CSE related parameter called "race immunity" R

$$R = \tau_{CQ} - t_{hold} \tag{3.5}$$

which, in the case where $\tau_{logic} = 0$, determines the amount of skew/jitter that can be tolerated by the CSE without incurring in an hold time violation.

As anticipated at the beginning of the paragraph, analogous quantitites ($\tau_{CD} = -\tau_{DC}$, τ_{CQ}, τ_{DQ}, t_{setup} and t_{hold}) can be defined for a latch and, in this case, they are more conveniently referred to the clock edge that closes the transparency phase of the latch (e.g., the clock falling edge for a latch transparent during the high clock phase) [OSM03]. Note also that, differently from what happens when an edge triggered scheme (i.e., Master-Slave or Pulsed CSEs) is employed, in systems employing pure latches (level sensitive clocking schemes) the CSEs latency is again determined by τ_{DQ} if the new D value arrives when the transparency phase has already been entered, but is determined by τ_{CQ} if the new D value is set prior to the transparency phase [OSM03]. The two cases are depicted in Fig. 3.15.

This ambiguity is resolved by considering that level sensitive clocking schemes inherently exploit the "time borrowing" concept, which is later explained in Paragraph 3.6.

Finally, let us examine the topological differences between Master-Slave and Pulsed CSEs from setup and hold times perspective:

1) In Master-Slave CSEs (or FFs), the clock edge enabling the Slave is the active one. For this reason, it can be understood that the setup time requirement is related to the delay that D experiences when traversing the Master to reach the Slave input, i.e. $t_{setup} > 0$ always. As can be seen from Fig. 3.12, Master-Slave CSEs are also featured by an high τ_{DQ} sensitivity in

the minimum τ_{DQ} region (this is due to the alternate enabling of Master and Slave). As concerns the hold time, again due to the Master delay experienced by D before reaching Slave input, after having assumed a new value, D can come back to the old one even before the clock edge and the CSE can still change its state. This means that $t_{hold} < 0$ in Master-Slave CSEs. Note however that this does not imply a total immunity to critical races given the impact of skew/jitter in (3.3).

2) As previously explained, Pulsed CSEs are basically latches that are enabled by a pulsed clock. By assuming the clock edge generating the pulse as the active clock edge, typical setup and hold curves for a Pulsed CSE are depicted in Fig. 3.16. Since the inner latch is transparent for the whole pulse duration, D can change its value well after the active clock edge and still be captured. This means that, unless for unrealistically high latch delays, normally $t_{setup} < 0$ in Pulsed CSEs. Moreover, the transparency during the pulse duration implies a flat minimum τ_{DQ} region where τ_{DC} and τ_{CQ} can be traded but their sum (τ_{DQ}) remains constant and is equal to the delay of the transparent latch. As concerns the hold time, it is obviously positive since the new D value has to be maintained up to the end of the transparency window.

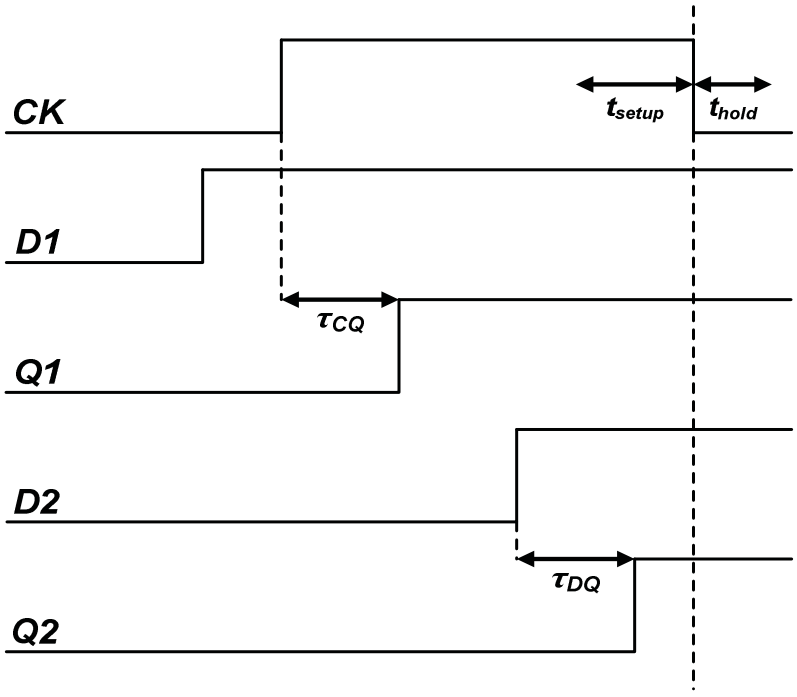


Fig. 3.15. Timing diagrams of a latch transparent during high clock phase.

As a final remark, note that for both Master-Slave and Pulsed CSEs a tradeoff exists between t_{setup} and t_{hold}. Master-Slave (Pulsed) CSEs suffer (take advantage) from a $t_{setup} > 0$ (< 0) and are hence less (more) suitable for critical paths. Conversely, Master-Slave (Pulsed) CSEs take advantage (suffer) from a $t_{hold} < 0$ (> 0) and are hence more (less) suitable for fast paths.

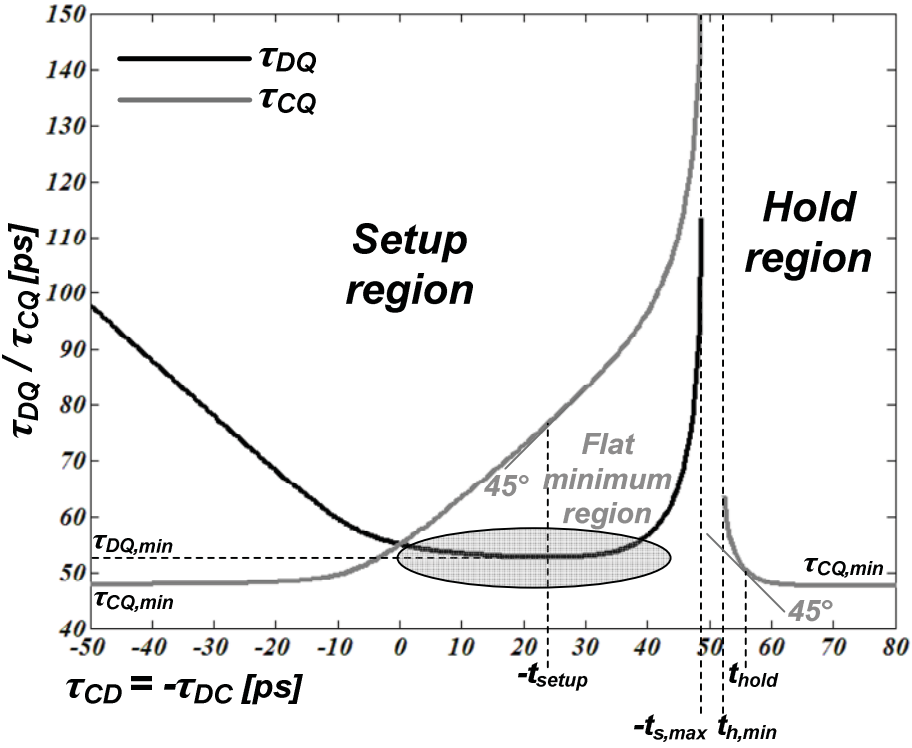


Fig. 3.16. τ_{CQ} / τ_{DQ} vs. $\tau_{CD} = -\tau_{DC}$ timing curves in a Pulsed CSE.

3.5 Clock Uncertainties Absorption and Time Borrowing

The sources of uncertainty related to the clock, i.e. skew and jitter, affect high-performances systems significantly. For instance, their impact on the timing constraint in (3.2) can be as high as $1 \div 2FO4$, hence strongly affecting the minimum clock period, which one wants to be close to $10FO4$ in very high-speed microprocessors. It is hence necessary to adopt techniques helping to achieve such high clock frequencies in despite of the presence of these clock uncertainties. The clock skew can be reduced with an appropriate design of the clock distribution network and through active

de-skewing techniques. The clock jitter, which nowadays can weigh as much as skew, is more related to the clock generation circuitry and to the noise injected from the power network. In any case, the application of complex techniques to reduce skew and jitter is possible up to a certain point and they have to be taken into account as an unavoidable overhead on cycle time.

However, CSEs prove to be extremely useful to reduce skew and jitter impact on performances when they exhibit the so called "soft clock edge property" [OSM03]. Basically, the topologies having a transparency window like the Pulsed ones (see Fig. 3.16) behave as latches during such a time window, thus being less sensitive to the $\tau_{CD} = -\tau_{DC}$ delay (which varies due to skew and jitter). Quantitatively, by referring to Fig. 3.16, one can define a maximum allowable τ_{DQ} increment, $\Delta\tau_{DQ}$, with respect to $\tau_{DQ,min}$ and a correspondent τ_{DC} window, $\Delta\tau_{DC}$, within which $\tau_{DQ} \leq \tau_{DQ,min} + \Delta\tau_{DQ}$. The clock uncertainties absorption factor, α_{CU}, is hence defined as [OSM03]

$$\alpha_{CU} = \frac{\Delta\tau_{DC} - \Delta\tau_{DQ}}{\Delta\tau_{DC}} = 1 - \frac{\Delta\tau_{DQ}}{\Delta\tau_{DC}} \tag{3.6}$$

This parameter represents the fraction of clock uncertainties (τ_{DC} variation) that is not transformed into a correspondent τ_{DQ} variation. It obviously depends on the choice of allowable $\Delta\tau_{DQ}$ delay degradation and, under a fixed topology, it decreases (increases) by increasing (decreasing) $\Delta\tau_{DC}$. Different CSE topologies have to be obviously compared under equal $\Delta\tau_{DC}$ and $\Delta\tau_{DQ}$ conditions and, the higher α_{CU}, the better the CSE from clock uncertainties absorption perspective. The above discussion is graphically exemplified in Fig. 3.17.

The flatness of τ_{DQ} vs. $\tau_{CD} = -\tau_{DC}$ characteristic is hence essential to reach high α_{CU} values and, as anticipated, this is a peculiar feature of Pulsed CSEs, while, due to their high τ_{DQ}-to-τ_{DC} sensitivity, Master-Slave ones are unsuitable for clock uncertainties absorption. Since the latter is an essential requirement for circuits featured by high clock frequencies, this further proves that Pulsed CSEs are the optimum choice when speed is the primary concern. The transparency window of Pulsed topologies is increased by enlarging the pulse duration and this leads to more and more negative (positive) setup (hold) times. Therefore, one must keep in mind that, according to (3.3), a stronger soft clock edge property is unavoidably traded with a minor robustness to hold time related failures.

The lack of "hard" clock edges in Pulsed CSEs (differently from Master-Slave ones) allows a significant flexibility in the design of pipeline stages. Indeed, being Pulsed CSEs just like latches transparent for the whole pulse width, the D transition can arrive just before the end of the pulse (which occurs later than the clock edge generating the pulse itself) and still be

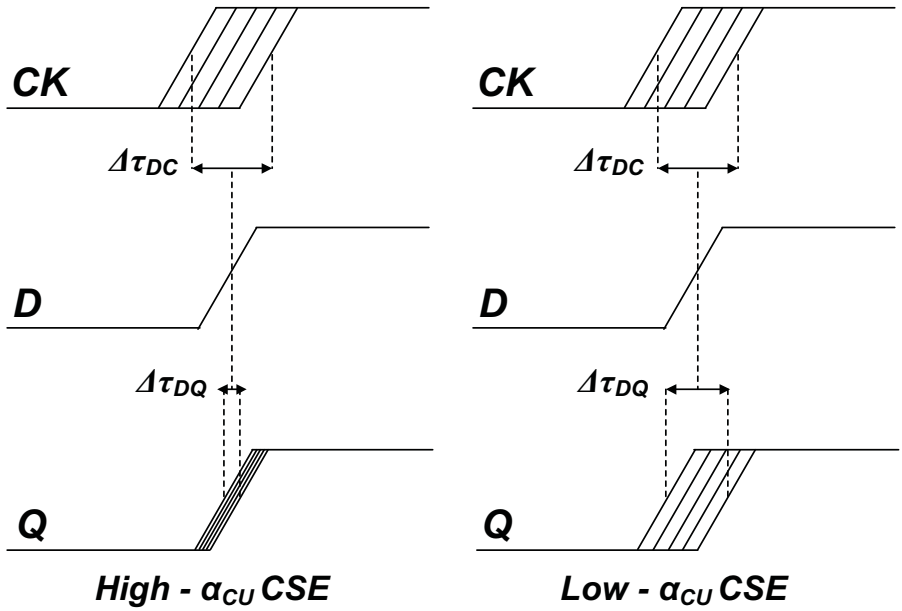


Fig. 3.17. Clock uncertainties absorption.

captured. This means that, by referring to the clock edge as the reference time instant, the pipeline stage providing the *D* transition can "steal time" from the subsequent stage [OSM03]. In this way, the flexibility in the pipeline design is increased since there can be stages employing more and less than a clock cycle to perform their operation. This technique is called "time borrowing" and allows to achieve higher clock frequencies since the minimum clock period is not determined by the maximum among the delays of the various unbalanced pipeline stages but rather by their mean [OSM03]. This mechanism is depicted in Fig. 3.18.

The one just described is actually called "dynamic time borrowing", since it is determined by the pipeline stages delay. For completeness, it is worth highlighting that a similar principle is exploited when the actual clock signals provided to the CSEs are deliberately delayed so that the critical pipeline stage has more time to complete its operation, and this is called "opportunistic skew scheduling" or "static time borrowing" [OSM03].

Note also that the above discussion can be easily extended to the case of level sensitive (instead of edge-triggered) clocking strategies employing latches instead of Pulsed CSEs. Actually, latches can even exploit a transparency window lasting for an entire half clock period and hence the flexibility in pipeline stages design when employing time borrowing is further increased. Nevertheless, as previously anticipated, level sensitive clocking is not as popular as the edge-triggered approach (based on Master-

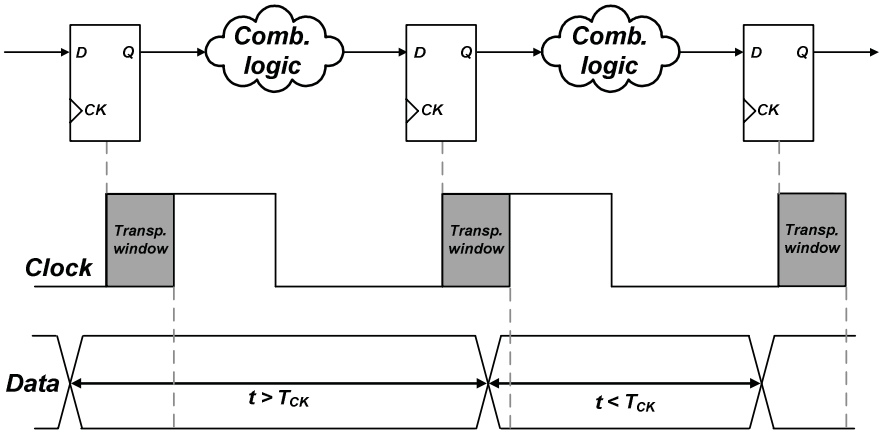


Fig. 3.18. Time borrowing.

Slave or Pulsed CSEs) and this is because, even if the maximum achievable speed performances (in terms of maximum clock frequencies) are higher in a latch-based system, the risk of data race-through is significantly increased [OSM03]. On the whole, the edge-triggered approach proves to be much more robust and reliable and, as a further merit, it is compatible with Design for Testability (DFT) methodologies [OSM03].

To conclude, both clock uncertainties absorption and time borrowing are based on the soft clock edge property of Pulsed CSEs (and latches), which in turn is related to a flat τ_{DQ} vs. $\tau_{CD} = -\tau_{DC}$ characteristic. In the former case, the variations occur on the time when the clock edge occurs. In the latter case, it is the D arrival time to vary. Pulsed CSEs, as well as latches, can hence be used to achieve both targets in very high speed systems [OSM03].

3.6 Energy Consumption in Clocked Storage Elements

The analysis of CSEs energy dissipation is of fundamental importance since they constitute a large portion of the clock system (made up also by clock generation and distribution circuits), which can contribute up to $30 - 50\%$ of the whole energy budget in high-performance microprocessors [NO05], since this fraction increases when considering higher clock frequencies and smaller logic depths. Moreover, the issue of energy consumption can be no more faced separately from that of speed performances, since, in practice, unavoidable constraints on the power-budgets subsist [OK06], [N08]. Hence, energy dissipation is as fundamental as speed in the design of high-performance microprocessors.

As for other kinds of digital circuits, also the average energy dissipation of a CSE in a clock cycle can be evaluated by integrating the simulated

supply current over the clock period and it is constituted by dynamic (capacitances charge/discharge), short-circuit (current contentions) and static (leakage) contributions.

Techniques to model this sources of dissipation for generic CMOS gates (by which also CSEs are made up) have been provided in Chapter 2 and are not repeated here. Moreover, due to their peculiar operation that involves the presence of a signal always commutating (the clock), a proper estimation of the transient (dynamic plus short-circuit) dissipation of CSEs requires deep considerations that are later provided in Chapter 5 when discussing characterization and optimization techniques.

Here, the focus is on identifying the causes that lead each of the three dissipation sources to increase or decrease according to topological features and the techniques that can be employed to achieve energy savings.

3.6.1 Dynamic energy dissipation and techniques for its reduction

The dynamic consumption increases with the supply voltage and the voltage swings, capacitances and switching activities associated to the CSE nodes [RCN03]. Obviously, being the clock, which is the signal with the highest switching activity within a chip, one of their inputs, CSEs suffer from dynamic dissipation more than combinational logic.

The simplest way to reduce dynamic energy is to lower the supply voltage V_{DD} due to its quadratic impact [RCN03]. This has obviously a strong effect on speed performances but the biggest issue is related to fast paths (hold time) requirements [OSM03]. Indeed, the setup time constraint can be satisfied by properly decreasing the clock frequency. However, the hold time constraints are not dependent on the clock period and care must be taken to guarantee that (3.3) is not violated by considering the impact that a V_{DD} reduction has on its various terms. Note that, the speed decrease in CSEs due to V_{DD} reduction is not as strong as that of buffers/inverters since CSEs extensively employ stacking in most of their critical stages (stacked structure takes relatively advantage of V_{DD} reduction since the further lowered DIBL is overtaken by the advantage in terms of reduced body effect).

Another approach is to reduce the voltage swing of some critical node. Typically, this approach is adopted on the clock signal since it exhibits the highest switching activity. Again, the tradeoff is with the lowered speed performances but in this case also an increased circuit complexity is required (as well as a low supply voltage has to be feasibly available). For instance, a local low-swing clock can be provided by adding specific low supply drivers for each CSE [KTS95] or a global low swing clock can be provided to all CSEs. In any case, in order to avoid an unacceptable static consumption, this low swing clock has to drive only NMOS transistors [MTD04] or PMOS transistors with a modified bulk voltage to increase the threshold have to be

employed [KS98]. In general however, low swing CSEs do not exhibit a good energy-efficiency and hence are not discussed in the rest of the work.

As concerns the impact of capacitances, it has to be considered in conjunction with the switching activity relative to each node. It is certainly true that CSEs with too many nodes (capacitances) are not energy-efficient. But it is also obvious that those nodes that are critical from speed perspective can exhibit a significant capacitance due to the required transistors over-sizing [WH04]. In general, compatibly with delay requirements, one should always reduce the capacitances of the nodes that frequently commutate. Note also that the relative impact of capacitive interconnects parasitics is extremely significant in nanometer technologies. Therefore, when comparing and selecting optimum CSE topologies, the layout related issues are of primary importance and cannot be simply considered a posteriori. This aspect is deeply investigated in Chapters 4 and 5.

Finally, two main techniques to reduce the switching activities of certain nodes can be employed: the clock gating and the conditional approach.

Clock gating is based on enabling the clock provided to CSEs through an additional signal as shown in Fig. 3.19 [OSM03]. Once the clock is disabled, several other (if not all) nodes transitions within the CSE are too.

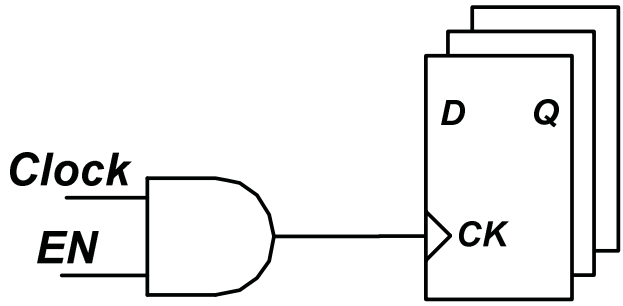


Fig. 3.19. Clock gating.

The clock gating can be global when several CSEs are all contemporarily disabled depending on external condition (e.g., the system goes in a standby mode). On the other hand local clock gating can be employed by integrating additional enabling logic within each CSE [NO98], [MNB01]. This logic is based on the comparison between the stored output and the upcoming data input and, only in the case where a difference is detected, an internal clock is enabled to capture the new data. Obviously, the setup time and the data-to-output delay can be significantly worsened since the data has to pass through additional gates. Although this techniques can lead to energy savings in low switching activity conditions, the increased circuit complexity can hide this advantage, as shown for the exemplificative cases discussed in Chapter 5.

Also the conditional approach allows to reduce the dynamic consumption in low switching activity conditions. Several mechanisms have been proposed to disable some internal transitions when the data input does not change its value. Several examples are discussed in Chapters 4 and 5, such as the conditional precharge [NAO01], the conditional capture [KKJ01] and the conditional discharge [ZDB04]. Unlike locally clock gated CSEs, the conditional ones typically do not suffer from a significant speed performances degradation because the additional logic typically lies outside the critical data-to-output path.

3.6.2 Glitches, short-circuit and static energy dissipation

Glitches are undesired and transient nodes transitions occurring because of relative delays between the commutations of transistors/gates. The glitches are said "propagated" if they are already present in the CSE inputs (data and clock) and then propagate in some CSE internal node or even to the output. On the contrary, they are said "generated" if the CSE inherently produce them. They manifest as positive or negative pulses (often not reaching the full V_{DD} voltage swing) on internal nodes and/or output, which should remain stable when data and/or clock make clean $0 \rightarrow 1$ or $1 \rightarrow 0$ transitions but do not. If glitches are generated on largely capacitive nodes, they bring a significant additional transient dissipation and make the topology energy-inefficient.

Short-circuit dissipation arises because of transient current contentions between pull-up and pull-down networks within CSEs. As for any other CMOS circuits, the impact of this contribution is reduced by avoiding slow signals with large rise/fall times (e.g., clock and data inputs provided to the CSEs have to be properly buffered) [RCN03]. Note that in CSEs this contribution can be significant because of the presence of keepers that are used to make precharged or output nodes static. When keepers are not gated, they must be properly weakened in order to avoid a large current contention (although they have a sufficient minimum strength to retain the node voltage in case of disturbances or to counteract leakages). A solution to this issue is to employ gated keepers (see Fig. 3.20) that are disabled by some internally generated signals when the node where they lie has to change its value (obviously this causes a slightly augmented complexity) [OSM03].

Static dissipation is due to leakage currents, i.e. to junctions, gate and sub-threshold leakages. The latter is the main contribution and, due to technology scaling, it is becoming one of the dominant one in modern microprocessors [CFB01], [N08]. Leakage current is always present and is closely related to the overall transistors width, i.e. to circuit complexity in terms of number of gates and transistors sizing. As will be shown in Chapter 5, leakage energy is not comparable to the transient one when the clock (CSE) is active. However, it must be considered that systems (or portions of

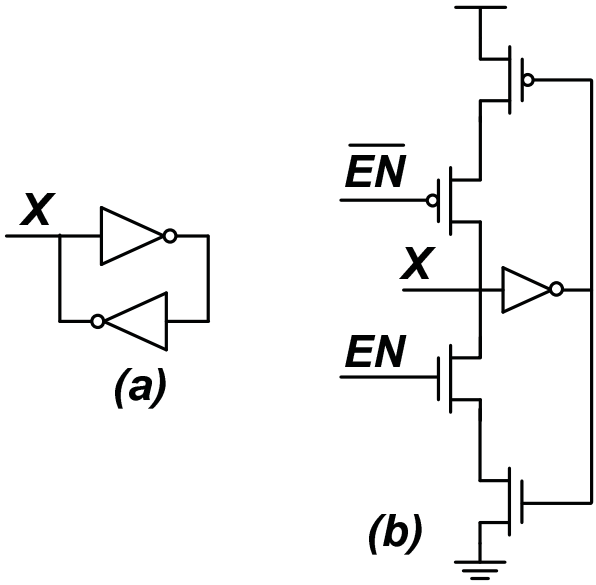


Fig. 3.20. Non-gated (a) and gated (b) keepers.

them) are often put in standby/idle modes where the clock is gated for long period of times and hence the relative weight of leakage energy (which is the only one to be present in standby) becomes significant also for CSEs. The techniques employed to reduce sub-threshold leakage are the same that are used for any other kind of digital circuits, like the usage of multi-thresholds and sleep transistors [KC01].

3.7 Differential and Dual Edge-Triggered Topologies

The two main CSE topological classes (Master-Slave and Pulsed) have been discussed in the previous paragraphs by referring to single output, Single Edge (positive or negative) Triggered (SET) CSEs.

In real systems it is often necessary to generate the negative version of the output signal, $\overline{Q}$. Although this could be done through the addition of a simple inverter, unbalanced delays would inherently arise. For this reason, fully differential CSEs, that exhibit symmetrical D-to-Q and D-to-$\overline{Q}$ paths are often employed. In many cases, these differential CSEs exploits regenerative feedback between the outputs of the first stage (see Fig. 3.21), thus achieving an operation that resembles that of sense amplifiers. Several versions of the so called Sense Amplifier FF (SAFF) have been proposed in the past [MB90], [GCM91], [MWA96], [OS01], being the Modified SAFF

(MSAFF) in [NSO00] the most energy-efficient one (this CSE is extensively discussed in Chapters 4 and 5).

It is worth highlighting that differential sense amplifier CSEs can be used to achieve low-to-high level shifting within the CSE (when data input is driven by gates with a V_{DD} lower than that powering the CSE) given the regenerative property of their input stages [HTA98], [ISN04].

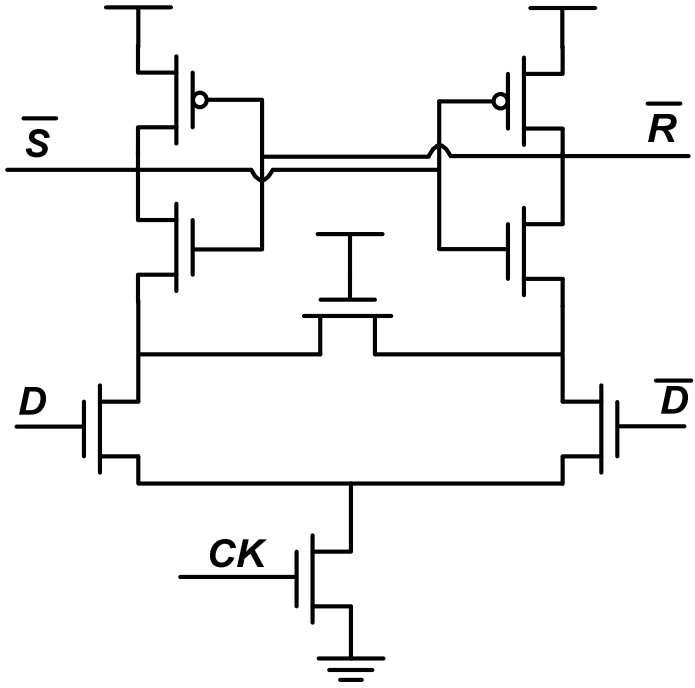


Fig. 3.21. Sense-amplifying input stage in a differential CSE.

A possible technique to reduce power dissipation is to employ Dual Edge-Triggered (DET) CSEs. A DET CSE captures the *D* value in correspondence of both (rising and falling) clock edges and is obtained by properly modifying the topology of a correspondent SET CSE [U81], [LE90], [GEH93], [HWA94], [SNC99], [NAO02], [NO05].

The great potential benefit with DET CSEs is that the same system throughput can be maintained while halving clock frequency and this implies power savings relative to the CSE dissipation induced by clock commutations. However, DET CSE are inherently more complex than their SET counterparts and this can lead to a worsening in delay and energy performances, i.e. to a lower overall energy-efficiency.

As for SET CSEs, also DET ones belong to three main classes that are depicted in Fig. 3.22:

1) Latch-Mux DET CSEs, where two latches, which are transparent in opposite clock phases, are connected in parallel and, at each moment, a final clocked multiplexer selects the output of the latch that is non transparent at that moment. Basically, the operation is analogous to that of Master-Slave CSEs and the difference is that there are two Masters that operate in parallel and the final clocked multiplexer that acts as the Slave. The main disadvantage of such a solution is that the area is almost doubled with respect to the SET counterpart.

2) Explicitly Pulsed DET CSEs, which are analogous to the SET case except for the pulse generator, which, this time, produces a synchronizing pulse in correspondence of both clock edges. The increase in area and complexity is limited to the pulse generator, which however has to be properly designed to achieve a robust and as much symmetrical as possible operation.

3) Implicitly Pulsed DET CSEs, where, analogously to the SET case, a first stage generates a pulse that is function of *D* and both clock edges, while a second stage (a clocked latch or an asynchronous memory element) captures the transition. In general, Implicitly Pulsed CSEs are significantly more complex than Latch-Mux or Explicitly Pulsed DET and hence exhibit a lower energy-efficiency.

Four energy-efficient DET CSEs, belonging to the above three classes, are deeply investigated in Chapters 4 and 5.

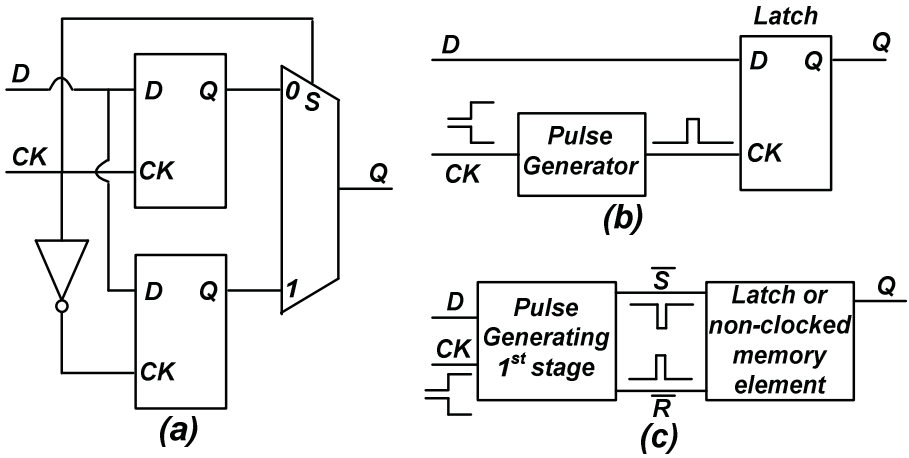


Fig. 3.22. Master-Slave (a), Explicitly Pulsed (b) and Implicitly Pulsed (c) Dual Edge-Triggered CSEs.

Chapter 4

ENERGY-EFFICIENT TRANSISTOR-LEVEL DESIGN OF CLOCKED STORAGE ELEMENTS

In this chapter, a general and complete transistor-level design flow for nanometer clocked storage elements is presented. The proposed design methodology permits to optimize these circuits under constraints within the energy-delay space through extensive adoption of the Logical Effort method. Transistor sizing is rigorously discussed by referring to cases that occur in practical designs and various interesting properties are derived from circuit analysis. In contrast to previous works, the impact of local interconnections is explicitly accounted for in the design loop, as is required in nanometer CMOS technologies. A case study is discussed in detail to exemplify the application of the proposed methodology.

Furthermore, in the large Appendix at the end of the chapter it is shown that, when dealing with transmission−gate based Master-Slave flip flops, a reconsideration of the classical approach for the delay minimization is worthwhile. By splitting such circuits in two sections that are separately optimized and then reconciling the results, the emerging design always outperforms the one resulting from the employment of a classical Logical Effort procedure.

4.1 A Comprehensive Design Approach

As explained in the previous chapter, the clocked storage elements (CSEs) selection is crucial in the design of synchronous VLSI systems (such as microprocessors), due to their high impact on overall timing and energy consumption [O03], [NO05]. For this reason, a number of CSE topologies have been studied and analyzed until now [SO99], [HA01], [MNB01],

[TNC01], [GNO07], [HKA07], but unfortunately no general methodology has been proposed for the circuit design of CSEs.

As concerns the transistor sizing methodology, due to the limited available power-budgets, real designs must belong to the set of points in the $E-D$ space with minimum energy (delay) for a given delay (energy) constraint, which, as explained in Chapter 2, is usually referred as the Energy-Efficient Curve (EEC). Up to now, typical approaches were based on extensive simulations to identify the CSE design that meets the delay requirement and belongs to the EEC [GNO07], which is computationally inefficient and forces the designer to arbitrarily discard potentially good design solutions. Instead, a better approach should exploit the properties of the EEC in order to preliminarily discard inefficient designs, and search only among the best and promising design points, as discussed in the following.

In the following, a general and complete CSE design methodology is proposed [ACP10-2], [ACP10-3]. Such a procedure is summarized in Fig. 4.1 and consists of four steps.

In the first step, independent design variables (i.e., transistor sizes) are identified. Indeed, it is no use considering all the transistors dimensions as independent variables, since not all of them affect the CSE speed performance (but all sizes impact the energy consumption). In particular, as is detailed in Paragraphs 4.2 and 4.3, let us adopt the following strategy:

- assume as Independent Design Variables (IDVs) only the aspect ratios of transistors lying in the input-to-output paths (first step in Fig. 4.1);
- assume the remaining Dependent Design Variables (DDVs) either equal to the minimum size which guarantees the correct circuital functionality, or functions of the size of the IDVs. Their expression is set in the second step in Fig. 4.1.

After the two above steps, the IDVs have been identified to reduce the size of the design space, thereby reducing the computational effort of the optimization algorithm.

In the third step discussed in Paragraph 4.4, the computational effort involved in the optimization is further reduced by appropriately bounding the region where the optimum designs are searched. Indeed, as it has been explained in Chapter 2, these bounds can be evaluated by estimating the transistor sizes needed to achieve the maximum speed (i.e., applying the Logical Effort (LE) method) under a varying and increasing input capacitance C_{IN}. In particular, according to the iterative procedure in Chapter 2, all IDVs (i.e., transistor sizes) are related to C_{IN} through the LE design approach. Hence, C_{IN} is progressively increased until the required condition on the energy-to-delay sensitivity with respect to input capacitance is achieved [ACP12-1], [ACP12-2].

Once the bounds for the IDVs are found, in the fourth step (see Paragraph 4.5) a search algorithm is applied to find the optimum value of the IDVs that

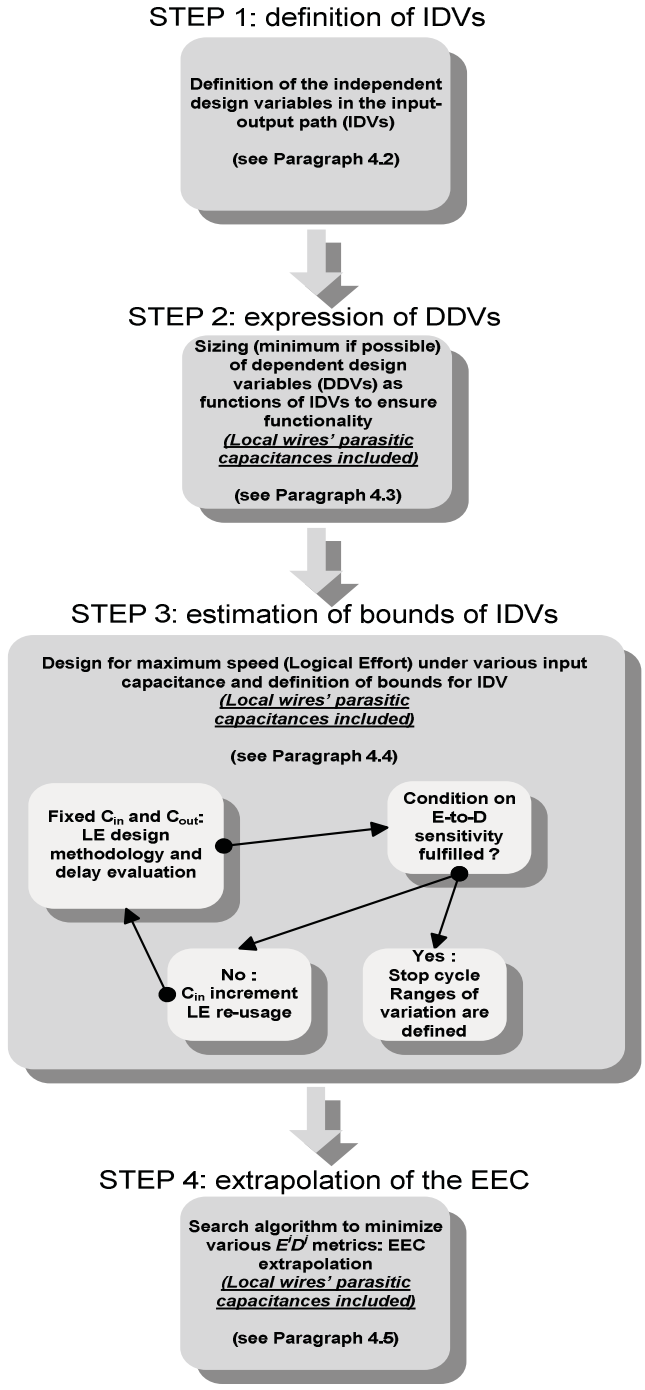


Fig. 4.1. Summary of the proposed design procedure.

minimize a few $E^i D^j$ metrics. This is repeated for various values of the pair (i, j) to find various points of the EEC, according to the discussion in Chapter 2. The complete EEC is then found by interpolating these points, which permits to evaluate the energy efficiency of a CSE topology under various speed constraint, as well as to compare different topologies.

It is worth noting that, in contrast to previous works, the proposed design procedure takes into account the local interconnections parasitics, which must be necessarily introduced within the transistor-level design loop in nanometer circuits, rather than consider them only subsequently [HMH01], [WH04]. In principle, the exploration of the design space accounting for local wires' parasitics would require one layout for each design point (i.e., transistor sizing) considered in the optimization, hence a very large number of layouts (typically thousands or more) should be drawn, which is impractical. To overcome this problem, a simple but reasonably accurate method to estimate the layout parasitics under an arbitrary transistor sizing is used from the second to the fourth step in Fig. 4.1, as discussed in the following. A detailed description of this method is reported in Paragraph 4.7.

4.2 Definition of Independent Design Variables - Step 1

In practical CSEs, automated transistors size optimization is computationally feasible only if proper circuit simplifications are introduced to reduce the design space size, due to the large number of transistors (and hence of design variables). To reduce the number of design variables, it should be first observed that the sizes of some groups of transistors can be unified into a single variable, as in the case of the series-connected transistors. Secondly, it is well known that the transistors that do not affect the CSE speed performance can be sized to a constant value ensuring correct functionality, or to a value that is a function of the size of some other transistor that impacts the CSE speed. Therefore, the sizes of these transistors are dependent design variables (DDVs), as opposed to the sizes that are regarded as independent design variables (IDVs).

In the following, criteria to identify IDVs are discussed for the various cases that occur in practical CSE topologies.

The IDVs are the sizes of transistors affecting the delay of input-to-output paths. As usual, the delay D of CSEs is defined as the well-known minimum data-to-output delay $\tau_{D-Q,min}$, [O02], [O03], [OSM03], hence the input is represented by the incoming data applied to the CSE. In order to reduce the computational effort required for the various design tasks, we assume that:

- the series-connected transistors are equally sized [WH04];
- all transistors in the data-to-output $(D - Q)$ paths have minimum sized channel lengths.

Thus, the IDVs are simply channel widths of transistors lying in the $D-Q$ path, which in the following will be normalized to the minimum value W_{min}, imposed by the technology, and will be referred to as w_i (i.e., w_i is the width of the i-th transistor normalized to W_{min}).

In general, with regard to the possible $D-Q$ paths, three different cases may occur depending on the CSE topology:

1) a single path;
2) two different paths, which at a certain point re-converge;
3) a first common path and a following bifurcation.

Each of these cases is discussed below, together with the procedure to identify the IDVs within these paths and the approach to be followed in the LE-based design (this is exploited in Paragraph 4.4, where the LE methodology is applied).

4.2.1 A single path

In this case, the rising and falling data transitions undergo the same logic gates to be passed through. Thus, in order to equalize the two delays, their pull-up and pull-down networks must be sized in order to compensate the different PMOS and NMOS driving capability.

This case is typical for the Latch-based structures such as Master-Slave FFs (e.g. Transmission-Gate FF (TGFF) [MNB01] or Write-Port Master-Slave FF (WPMS) [MTD03]) and Pulsed Latches (e.g. Transmission-Gate Pulsed Latch (TGPL) [NH02], [NCF02]). For instance, let us consider the TGFF in Fig. 4.2, where the data rising and falling paths are indicated. In this figure, the normalized channel widths w_i of all transistors lying in the path from D to Q are IDVs. As was anticipated, observe that also the first stage size w_1 (which in turn determines the CSE input capacitance) is considered as an IDV. Moreover, the transmission gates have transistors equally sized, as suggested in [SSH98].

With regard to the LE optimization, the transistors sizes in each stage are expressed as functions of the same parameters w_i and hence the LE method is simply applied to only one of the two paths (the other is automatically optimized as well).

4.2.2 Two different re-converging paths

The case of two different re-convergent paths occurs for example in the Implicitly Pulsed FFs (e.g. Hybrid Latch-FF (HLFF) [PBS96], Semi-Dynamic FF (SDFF) [KAD99], UltraSPARC SDFF (USDFF) [HAA00], Implicit Push-Pull FF (IPPFF) [N03], Conditional Precharge FF (CPFF) [NAO01]). Since the two paths have some logic gate in common, they must be jointly optimized for the common logic gates and separately optimized for the other gates. Hence, the previous symmetrical sizing strategy can be applied only for those gates belonging to both paths. For instance, when

considering the SDFF in Fig. 4.3, the two $D-Q$ paths are respectively made up by three and two topological stages having the IDVs $[w_1,w_2,w_4]$ and $[w_3,w_4]$, respectively. The two paths re-converge in the final inverter, which is thus symmetrically sized.

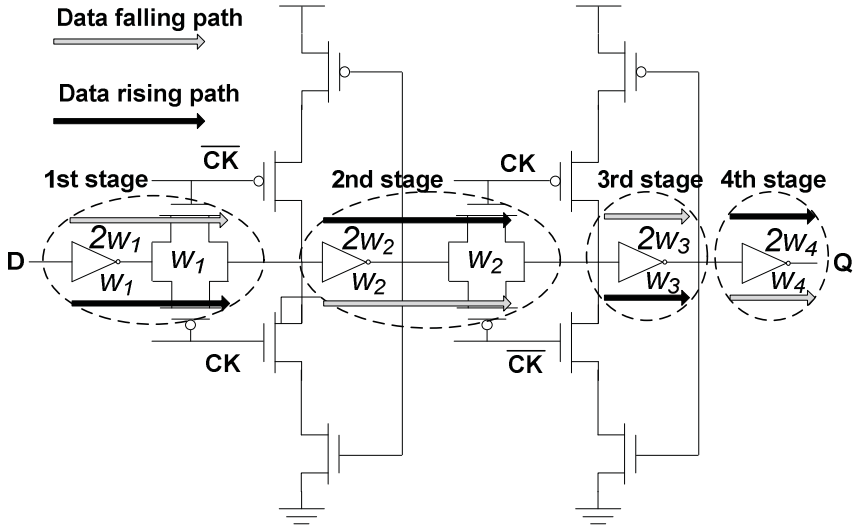


Fig. 4.2. $D-Q$ paths in the TGFF circuit (a single path).

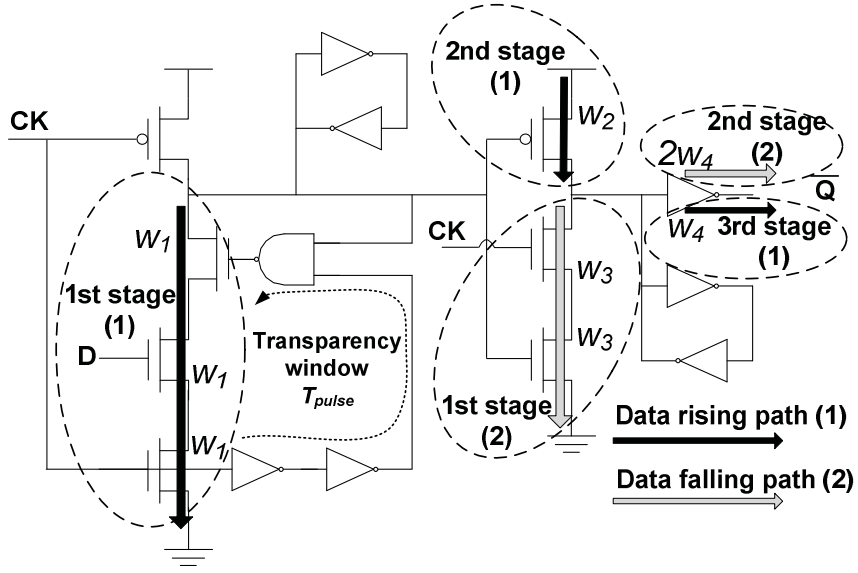


Fig. 4.3. $D-Q$ paths in the SDFF circuit (two re-converging paths).

In particular, regarding the application of the LE method, the longer path is first optimized, whereas the other one is sized to exhibit the same delay, given that, unless an intentional sizing strategy is applied, the longer path is inherently slower than the other one. Indeed, the CSE impact on the system speed is quantified through the maximum of its rising and falling delays and hence both paths have to be sized to somewhat guarantee a minimum common delay. Conversely, during the optimization in the $E - D$ space, the constrain is relaxed and IDVs are free to vary independently.

4.2.3 A bifurcating path

This case is typical of the differential topologies (e.g. Modified Sense-Amplifier FF (MSAFF) [NSO00], Conditional Capture FF (CCFF) [KKJ01] or Skew-Tolerant FF (STFF) [NOW03]), and has not the two paths related with different (rising and falling) inputs transitions, but to the different output nodes.

Even if this case is different from the previous one, the same approach can be followed. Indeed, during the LE optimization, the delay of the output charging and discharging paths must be equalized just like the data rising and falling paths previously discussed. An example is shown in Fig. 4.4 for the MSAFF. Note that only the pull-up networks of the inverters define an IDV (skewed inverters), since only their output rising transition is critical.

4.2.4 Other cases

Other possible topologies, such as clock-gated and Dual Edge-Triggered (DET) FFs, can be included in the previous cases. In particular, for the clock-gated FFs (e.g. Data-Transition Look-Ahead FF (DTLA) [NO98] or Gated Master-Slave FF (GMSL) [MNB01]), the parts of the additional gating logic belonging to the $D - Q$ paths, are considered as additional IDVs. With regard to DET structures, they exhibit four different $D - Q$ paths. However, in some topologies the four paths are equal in pairs, since have common sections and the other sections are replicated [LS96]). In other cases the topology is that of a Single-Edge Triggered FF and is simply driven by a DET Pulse Generator (PG) [ZDB04].

4.3 Sizing of Dependent Design Variables - Step 2

Once the IDVs have been identified, the remaining transistors sizes are dependent variables (i.e., DDVs) and must be set according to a proper strategy linking them to the IDVs.

Since the DDVs do not influence the CSE speed, but only its energy consumption, in general their sizes must be kept as low as possible, i.e. to the minimum value that guarantees correct CSE functionality. However, in

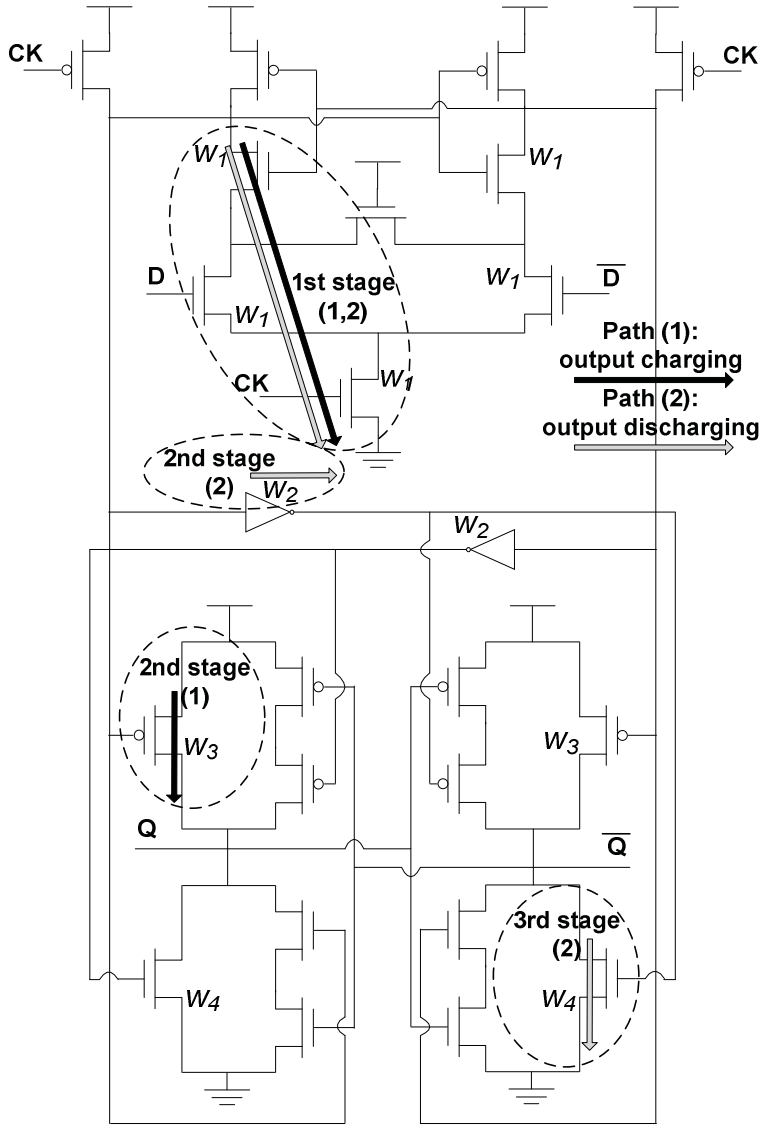


Fig. 4.4. $D-Q$ paths in the MSAFF circuit (a bifurcating path).

some of the cases discussed below a slightly more complex design strategy is required.

In the following we refer to transistors channel widths w (lengths l) normalized to the minimum size W_{min} (L_{min}). For the sake of simplicity, we consider only integer values of w, and $l \leq 2$.

The DDVs expressions are found in the following for the cases that can occur in practical designs.

4.3.1 Clocked precharge transistors

In most cases, PMOS precharge transistors are turned on only when all pull-down networks connected to the same node are turned off, i.e. they do not experience current contention with other transistors. In this usual case, for moderate clock frequencies and/or load values on the nodes to be precharged, precharge transistors are minimum-sized, (i.e., their normalized width is set to $w_{P,prech} = 1$). On the contrary, if the clock frequency requirement is very pressing and/or if the load on the precharged nodes is very high, there can be the need for larger precharge transistors sizing. Moreover, a sizing larger than minimum can be required also if there is some NMOS transistor in non-gated keeper that is simultaneously turned on during precharge. In this case, the PMOS precharge transistor must have a sufficient strength to counteract the non-gated keeper. Since NMOS transistors in non-gated keepers have (as discussed later on) minimum width and $l > 1$, $w_{P,prech}$ of the PMOS precharge transistor cannot be lower than 2 (see Fig. 4.5).

According to the LE model, a $FO4$ delay corresponds to a normalized delay equal to 5 [SSH98]. Hence, by normalizing to the $FO4$ delay, the half-period of the clock becomes

$$(T_{CK}/2)_{norm} = 5\frac{(T_{CK}/2)_{ps}}{(FO4)_{ps}} \tag{4.1}$$

In general, the normalized delay of the precharge PMOS, which has a $w_{P,prech}$ size, is given by

$$d_{P,prech} = \frac{1}{3}\frac{w_{P,prech}+\sum_i w_{gate,i}+\sum_j w_{drain,j}+3\frac{C_{par,int}}{C_{inv,min}}}{w_{P,prech}}\frac{1}{1-\frac{1}{l_{keeper}w_{P,prech}}} \tag{4.2}$$

where $w_{gate,i}$ and $w_{drain,j}$ are the widths of transistors whose gate and drain, respectively, are connected to the precharged node, $C_{par,int}$ is the parasitic lumped capacitance due to local wires related with the precharged node and $C_{inv,min}$ is the input capacitance of a symmetrical minimum inverter (i.e., with $W_P = 2W_N = 2W_{min}$). It is worth noting that the rightmost factor has the form $1/(1-r)$, where r is the ratio between the driving capabilities of a possible non-gated keeper connected to the precharged node and of the precharge PMOS (see Chapter 1).

A duration slightly smaller (e.g. by a 0.9 factor) than the half-period of the clock can be chosen to conservatively satisfy the clock period requirement. Thus, by assuming that rise time is roughly twice the delay, one can set $w_{P,prech}$ to the minimum value guaranteeing that

$$2d_{P,prech} < 0.9(T_{CK}/2)_{norm} \quad (4.3)$$

4.3.2 Keepers and noise immunity

When dealing with CSEs, the noise immunity is a concern mainly as regards floating nodes. This is the case of dynamic first stages in Pulsed FFs and of Master-Slave structures (since one of the two latch is disabled), where keepers and feedback paths are needed. Therefore, keepers lying on the floating nodes should be sized to guarantee a proper level of immunity to noise sources (such as leakage currents or capacitive crosstalk) [JKK02], but, at the same time, the current contention introduced by them should be as small as possible to avoid a speed/consumption degradation [NO00].

As concerns the leakage noise, one can adopt the methodology accurately described in [JKK02] (not reported for the sake of brevity). In [JKK02], simple mathematical manipulations involving the triode current of the keepers transistors and the subthreshold current of the pull-up or pull-down network which the keeper fights against are performed to size the keeper in order to guarantee the desired noise margin. It is worth noting that, in general, except for very large transistor sizings in the pull-up or pull-down paths (made up of stacked and almost never parallel transistors), minimum-sized keepers are sufficient to contrast leakage currents in CSEs.

The immunity towards capacitive crosstalk strongly depends on the coupling capacitances among the various nodes, which, in turn, strongly depend on the layout. Unlike for the estimation of grounded lumped capacitances in each node (which can be carried out through straight though enough accurate geometrical manipulations as shown in Paragraph 4.7), the estimation of the coupling capacitances is a task that needs the realization of the actual layout.

Again, a minimum-sized keeper is typically sufficient to avoid the corruption of the stored data. Anyhow, a check has to be carried out after post-layout simulations and, in the case of hazardous capacitive couplings, the keeper has to be slightly oversized.

When one can assume that leakage and crosstalk are not a significant concern (i.e., when minimum keepers are sufficiently strong), keepers transistors are sized in this way:

a) when the inverters that form the first stages of keepers do not belong to the $D-Q$ paths (see Fig. 4.5), they are always minimum sized to reduce the load on the CSE internal nodes;
b) keepers second stages, which restore the logic state at their output node, have always transistors with minimum widths, while their lengths, l_{keeper}, is in the order of a few units (e.g. 1 or 2), according to the desired driving strength. In particular, gated keepers usually have $l_{keeper} = 1$, since they are disabled during the charge/discharge of the internal nodes

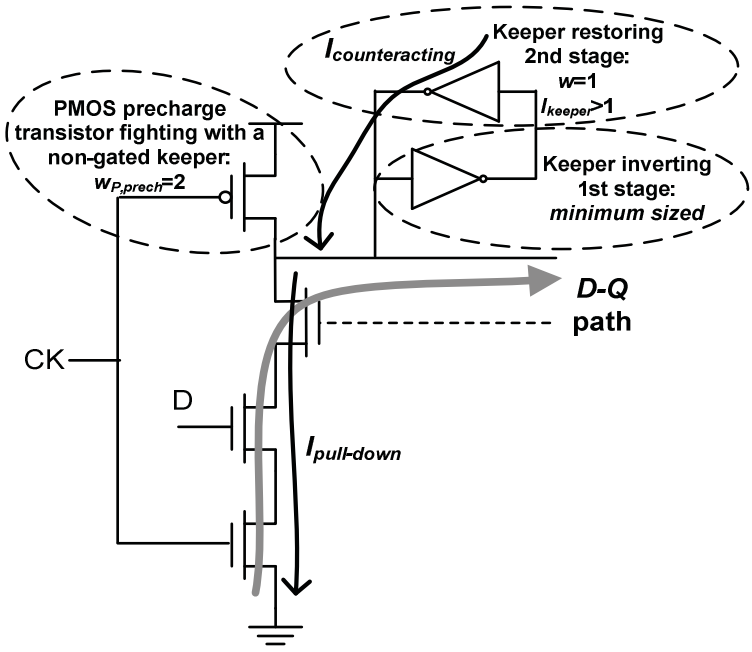


Fig. 4.5. Counteractive action due to non-gated keeper.

and hence do not determine current contention with other transistors. Instead, in order to guarantee a correct precharge or evaluation of the internal nodes (even when the transistors in $D-Q$ paths are very small), the restoring stages of non-gated keepers have $l_{keeper} > 1$ (see Fig. 4.5).

4.3.3 Feedback paths

Transistors involved in feedback paths (typically coming from the output) are used to disable/enable the transition of some internal node. If such feedback paths define the duration of the CSE "transparency window" [1] (e.g. in the CCFF [KKJ01]), the sizing of the transistors is carried out like for the pulsed generator, treated in the next case. For other feedback paths cases, transistors have minimum size (e.g. in the CPFF [NAO01]).

4.3.4 Pulse generators

Implicitly and Explicitly Pulsed CSEs [OSM03] always contain a pulse generator (PG) circuit that generates a pulse that defines the transparency window. The PG lies outside of the $D-Q$ paths, hence its transistors sizes are DDVs that must be tuned to:

[1] In Implicit-Explicit Pulsed FFs, the transparency window is the time period when the FF is transparent to the data input [OSM03].

a) achieve a sufficiently sharp edge at the beginning of the window;
b) keep the capacitance seen from the clock network as low as possible;
c) achieve the required pulse width, T_{pulse}.

For instance, by referring to the widely adopted PG implementation reported in Fig. 4.6, we can identify two main tasks: the design of the NAND gate providing the pulsed clock and the design of the inverter chain setting the transparency window duration, T_{pulse}.

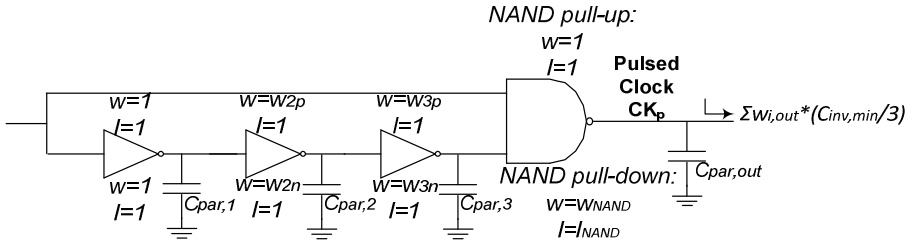


Fig. 4.6. Typical implementation of a pulse generator.

a) NAND design

In the circuit in Fig. 4.6, a sharp edge at the beginning of the window is ensured through proper sizing of the NAND gate.

Since the transparency window begins at the high-to-low transition of signal CK_p, the pull-up network is minimum-sized (to reduce the clock load), whereas the pull-down network must be sized larger. In other words, the resulting DDVs in the NAND gate are the width (length) w_{NAND} (l_{NAND}) of the transistors in the NAND pull-down network, which are usually set to achieve a fall time of FO3 (i.e., the fall time of an inverter loaded by three equal inverters) according to a well-known rule of thumb [SO99], [O02], [O03], [OSM03], [GNO07].

The LE model can be exploited to obtain a FO3 falling transition. To this end, we have to consider the size of each transistor driven by the PG, $w_{i,out}$ (i-th), together with the interconnections capacitance, $C_{par,out}$, at the NAND output (see Paragraph 4.7). All capacitive contributions can be converted into equivalent transistor widths by normalizing with respect to $C_{inv,min}/3$.

The logical effort g, electrical effort h, branching effort b and parasitic delay p **Errore. L'origine riferimento non è stata trovata.** for the NAND falling transition are

$$g_{f,NAND} = \left(\frac{w_{NAND} l_{NAND} + 1}{3}\right)\left(\frac{2 l_{NAND}}{w_{NAND}}\right) \tag{4.4a}$$

$$h_{f,NAND} = \left(\frac{\sum_i w_{i,out} + 3\frac{C_{par,out}''}{C_{inv,min}}}{w_{NAND} l_{NAND} + 1} \right) \quad (4.4b)$$

$$b_{f,NAND} = 1 \quad (4.4c)$$

$$p_{f,NAND} = \left(\frac{w_{NAND} + 2 + 3\frac{C_{par,out}'}{C_{inv,min}}}{3} \right) \left(\frac{2 l_{NAND}}{w_{NAND}} \right) \quad (4.4d)$$

where $C_{par,out}'$ is the fraction of $C_{par,out}$ depending on the size of the NAND gate itself (i.e., on w_{NAND} and l_{NAND}), thus affecting parasitic delay, whereas $C_{par,out}''$ depends on the sizes of the other CSE gates, thus affecting electrical effort.

By setting the falling transition time at the NAND output equal to FO3 (i.e., a normalized delay equal to 4 [SSH98]) one gets

$$d_{f,NAND} = g_{f,NAND} h_{f,NAND} b_{f,NAND} + p_{f,NAND} = 4 \quad (4.5)$$

thus finding w_{NAND} and l_{NAND} (both have discrete range of values and hence iterative cycles are sufficient to find them).

b) Inverters chain design

The pulse width (transparency window), T_{pulse}, determines the setup time and the hold time [RCN03] in Pulsed FFs. Therefore, T_{pulse} is tuned according to the time-borrowing requirement [OSM03] and/or to the necessity to avoid races in fast paths [OSM03].

The desired T_{pulse}, is achieved by properly sizing the chain made up by the three inverters in Fig. 4.6.

Once the NAND gate is sized, the clock load is minimized by adopting minimum size in the first inverter. Only the widths of the remaining two inverters are varied (i.e., they have minimum lengths) since four different widths (w_{2p}, w_{2n}, w_{3p}, w_{3n}) represent a sufficient number of degrees of freedom to achieve any practical T_{pulse} value.

Through calculations similar to (4.4)-(4.5), one gets

$$T_{pulse} = \frac{w_{2p} + w_{2n} + 2 + 3\frac{C_{par,1}}{C_{inv,min}}}{3} + \frac{2}{3} \frac{w_{3p} + w_{3n} + w_{2p} + w_{2n} + 3\frac{C_{par,2}}{C_{inv,min}}}{w_{2p}} + \frac{w_{NAND} l_{NAND} + 1 + w_{3p} + w_{3n} + 3\frac{C_{par,3}}{C_{inv,min}}}{3 w_{3n}} + d_{r,NAND} - d_{f,NAND} \quad (4.6)$$

where $d_{r,NAND}$ is the rising NAND delay, evaluated as in (4.4)-(4.5) and $C_{par,1-2-3}$ are the local wires' parasitic capacitances at inverters output nodes (see Paragraph 4.7).

The optimal solution in terms of w_{2p}, w_{2n}, w_{3p} and w_{3n} (see Fig. 4.6) is found by simple iterative cycles (the searched widths are integer numbers). Obviously, in order to reduce the energy consumption, these optimum widths should be searched within small-values ranges (i.e. a few units).

c) Different pulse generator topologies

A similar reasoning can be followed for other PG topologies, as in the case of the simple inverter chain usually employed in the Implicitly Pulsed FFs (e.g., in the HLFF [PBS96]). It is worth noting that, in some cases where the last stage of the delay chain is not an inverter (such as the CCFF [KKJ01] or the SDFF [KAD99]), such last stage is minimum sized to reduce the load on FF internal nodes and thus the CSE energy. In this case, in order to set T_{pulse}, only the size of the second inverter is varied by exploiting also its channel lengths, so that a sufficient number of degrees of freedom is maintained.

4.3.5 IDVs and DDVs in SDFF first stage

Since SDFF is chosen in Paragraph 4.6 as a case of study to validate the design methodology effectiveness, in Fig. 4.7 some IDVs and DDVs are shown in the case of the SDFF first stage (see Fig. 4.3) for exemplification.

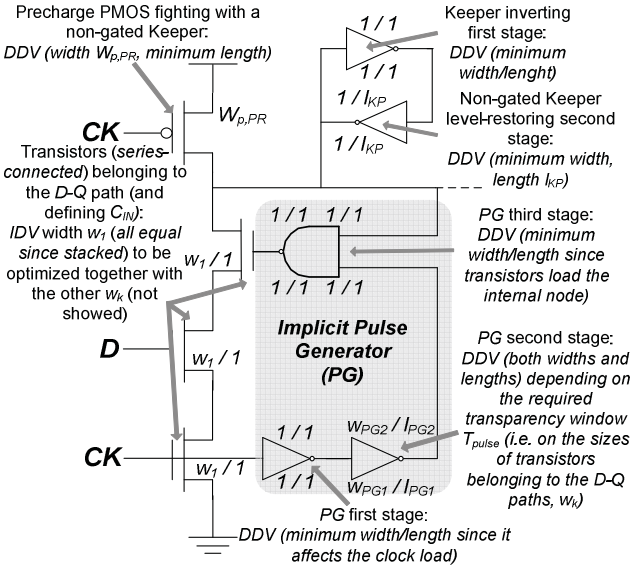


Fig. 4.7. SDFF: exemplification of IDVs and DDVs.

4.4 Estimation of Design Space (IDVs) Bounds - Step 3

The design strategies presented in the literature are generally discussed in papers that compare CSE topologies and assume a fixed input size or at most a narrow set of C_{IN} values [SO99], [HA01], [MNB01], [TNC01], [OSM03], [NO05], [HKA07], [GNO07].

However, as thoroughly explained in Chapter 2, this does not allow to extensively explore and compare the CSE potentials in terms of both energy and delay. For this reason, C_{IN} must also be considered as an IDV, in order to define a fair strategy to design and compare different CSE topologies.

Moreover, as discussed in Chapter 2, the definition of a maximum C_{IN} value can be exploited to determine design space bounds for all the other IDVs by referring to the minimum delay design through LE method and by setting an energy-to-delay sensitivity target value (with respect to C_{IN}) to be achieved.

This third step of the methodology, regarding the definition of practical design space (IDVs) bounds, is hence the same that has been presented in Paragraph 2.3.2 [ACP09-1], [ACP10-2], [ACP10-3], [ACP12-1], [ACP12-2] and is not repeated here for brevity.

4.5 Extrapolation of the Energy-Efficient Curve - Step 4

As for the previous paragraph, the fourth step of the methodology, regarding the extrapolation of the EEC once the design space bounds are defined, has already been discussed in Paragraph 2.3.3. For instance, to find some discrete points of the EEC, it is sufficient to minimize E^iD^j for a discrete set of (i,j) values (e.g. $[ED^3, ED^2, ED, E^2D, E^3D, E_0]$, where E_0 refers only to the energy minimization), by adopting a binary search algorithm (the topological complexity of CSE allows to assume the E^iD^j functionals as convex) and by progressively reducing the design space bounds once the design minimizing a specific figure of merit has been found. The EEC is finally extracted through a hyperbolical fitting that interpolates the discrete optimum points in the $E - D$ space, according to (2.11).

4.6 A Complete Design Example: the SDFF Case of Study

Let us consider the SDFF [KAD99], which is again reported in Fig. 4.8 and where the circles identify the various nodes in the circuit.

By applying the procedure described in Fig. 4.1, in the first step we have to identify the IDVs. This was already done in the SDFF example in Paragraph 4.2.2, which leads to the following list of IDVs: w_1, w_2, w_3 and

w_4. C_{IN} is given by $w_1(C_{inv,min}/3)$, i.e. w_1 is set by the assigned C_{IN}. In the following C_{IN} will be indifferently referred with w_1.

In the second step, by following the guidelines in Paragraph 4.3, the DDVs and their sizing are:

1) Transistors $M14$-$M15$ (first inverter of the delay chain that defines the transparency window), $M10$-$M11$, $M22$-$M23$ (the first inverters of the two non-gated keepers), $M18$-$M21$ (the NAND gate that is the last stage of the time-window delay chain) are all minimum sized.
2) Transistors $M12$-$M13$, $M24$-$M25$ (second inverters of the keepers) have minimum widths and $l = 2$ channel lengths.
3) Precharge PMOS $M9$ has $w_9 = 2$. Indeed, if it were minimum, it could not win the current conflict with $M12$.
4) Transistors $M16$-$M17$, which have to guarantee a transparency window duration T_{pulse} (together with the inverter $M14$-$M15$ and the NAND gate), are sized once the IDVs are known (i.e., for each w_i set of values) according to the principles similar to those suggested in Paragraph 4.3.4. In particular, the choice is that of setting the transparency window equal to the whole $D - Q$ delay, which can be analytically estimated as a function of the IDVs by using the LE method for each set of IDVs values.

In the third step, the upper bounds of IDVs must be found through the iterative LE-based procedure in Paragraph 2.3.2. As was explained in the SDFF example in Paragraph 4.2.2, one first optimizes the rising path r that

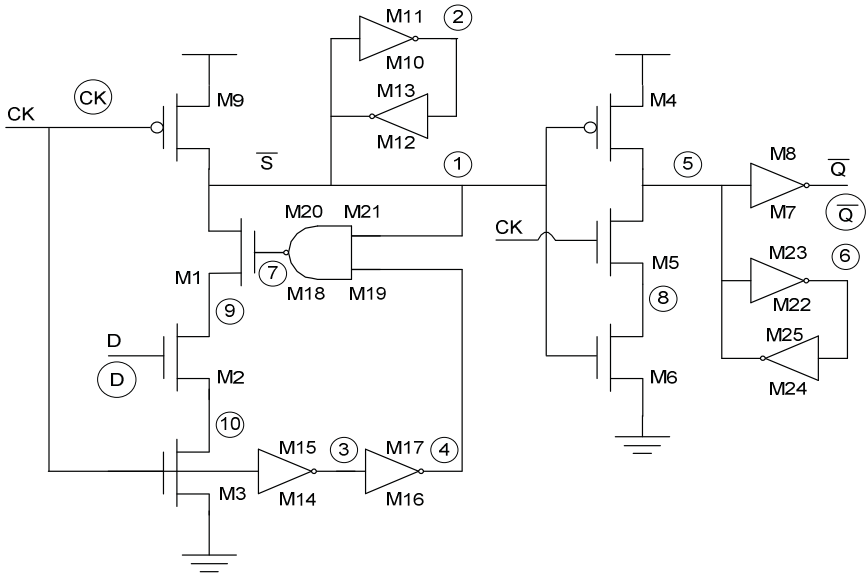


Fig. 4.8. SDFF schematic.

is made up by three stages. Then, one sizes the falling path f to have the same delay as the r path. Since w_1 is fixed for each cycle step (it is imposed by each C_{IN} value), one needs to evaluate only w_2, w_3 and w_4 with the LE method. This requires the knowledge of the LE parameters of each cascaded gate (logical effort g, electrical effort h, branching effort b and parasitic delay p) as a function of transistors widths, which are reported in Tab. IV.I, where the external load, C_L, is expressed in terms of equivalent normalized width w_L (i.e., $w_L = (3C_L)/C_{inv,min}$).

In regard to the parasitic interconnections capacitance at the generic i-th $C''_{par,i}$. The former are due to wire lengths depending on the size of the gate itself, whereas $C''_{par,i}$ depends on the other gates sizes. In Tab. IV.I, both capacitive contributions are converted into equivalent transistor widths by normalizing with respect to $C_{inv,min}/3$ (as for C_L).

Now, one can apply the LE method to find the optimum IDVs w_2, w_3 and w_4 as a function of w_1. Through a straightforward employment of LE and adopting the usual definitions ($G = g_{1r}g_{2r}g_{3r}$, $H = w_L/w_1$, $B = b_{1r}b_{2r}b_{3r}$ and the optimum stage effort $f_{OPT} = \sqrt[3]{GHB}$, [SSH98]), one gets

$$w_4 = \frac{w_L + 3\frac{C''_{par,\bar{Q}}}{C_{inv,min}}}{3f_{OPT}} \tag{4.7}$$

$$w_2 + w_3 = \frac{3w_4 + 4 + 3\frac{C''_{par,5}}{C_{inv,min}}}{f_{OPT}} \frac{4w_2}{3(2w_2 - 2)} \tag{4.8}$$

A third equation is then found by equalizing the r and f paths delays, which results to

$$g_{1r}h_{1r}b_{1r} + p_{1r} + g_{2r}h_{2r}b_{2r} + p_{2r} = g_{1f}h_{1f}b_{1f} + p_{1f} \tag{4.9}$$

The set of equations (4.7)-(4.9) must be iteratively solved to find the optimum value of w_2, w_3 and w_4 according to the values of w_1 and w_L (sub-step a in the iterative procedure in Paragraph 2.3.2). Once the IDVs are known, also the variable DDVs (sizes of M16-M17) are determined (sub-step a). After each simulation (sub-step b), the energy-to-delay sensitivity (with respect to C_{IN}) on (2.24) is evaluated on the E_{TOT}vs.C_{IN} and D_{TOT}vs.C_{IN} fitted curves (sub-step c).

The SDFF has been optimized under a $16C_{inv,min}$ load and a $V_{DD} = 1$ V supply voltage in the 65-nm STCMOS065 technology with $W_{min} = 120nm$, $L_{min} = 60nm$ and $C_{inv,min} = 410$aF.

The upper bounds of IDVs are found by first evaluating the maximum value of C_{IN} (or equivalently w_1) leading to an energy-to-delay sensitivity

TABLE IV.I: SDFF LE PARAMETERS: STAGE S = [1,2,3] / PATH P = [r,f]

	g	b
S=1 P=r	$\frac{4w_1}{4w_1-3}$	$\frac{w_2+8+w_3+3\frac{C''_{par,1}}{C_{inv,min}}}{w_2+w_3}$
S=2 P=r	$\frac{4w_2}{3(2w_2-2)}$	$\frac{3w_4+4+3\frac{C''_{par,5}}{C_{inv,min}}}{3w_4}$
S=1 P=f	$\frac{4w_3}{3(2w_3-1)}$	$\frac{3w_4+4+3\frac{C''_{par,5}}{C_{inv,min}}}{3w_4}$
S=2, 3 P=r, f	1	$\frac{w_L+3\frac{C''_{par,\bar{Q}}}{C_{inv,min}}}{w_L}$
	h	p
S=1 P=r	$\frac{w_2+w_3}{w_1}$	$\frac{w_1+3\frac{C'_{par,1}}{C_{inv,min}}}{w_1}\frac{4w_1}{4w_1-3}$
S=2 P=r	$\frac{3w_4}{w_2+w_3}$	$\frac{w_2+w_3+3\frac{C'_{par,5}}{C_{inv,min}}}{3w_2}\frac{4w_2}{2w_2-2}$
S=1 P=f	$\frac{3w_4}{w_2+w_3}$	$\frac{w_3+w_2+3\frac{C'_{par,5}}{C_{inv,min}}}{3w_3}\frac{4w_3}{2w_3-1}$
S=2, 3 P=r, f	$\frac{w_L}{3w_4}$	$\frac{3w_4+3\frac{C'_{par,\bar{Q}}}{C_{inv,min}}}{3w_4}$

value equal to $-j/i = -3$. Such a condition is fulfilled when $w_1 = 30$ (sub-step d in the iterative procedure in Paragraph 2.3.2). By employing the LE method for such a value of first stage size (i.e., input capacitance C_{IN}), one finds $w_2 = 20$, $w_3 = 10$ and $w_4 = 10$. Such a design is featured by a $2.12FO4$ $\tau_{DQ,min}$ delay (from simulations, $FO4 = 18.3$ps) and a $162E_{min}$ average energy per clock cycle (from simulations, $E_{min} = 0.202$ fJ is the transient energy dissipated in a $0 \rightarrow 1 \rightarrow 0$ output transition by an unloaded minimum symmetrical inverter).

With regard to step 4 in Fig. 4.1, the resulting points in the search space explored by the algorithm are depicted as small circles in Fig. 4.9. To develop an intuitive understanding regardless of the technology, in Fig. 4.9

the delay is normalized to $FO4$ and the energy is normalized to E_{min}. Note that the searched design points crowd around the EEC, as a proof of the effectiveness of the design space bounds determination and search algorithm employed strategies.

As expected, the asymptote D_0, which results to $30.2\text{ps} \approx 1.65FO4$, agrees well with the parasitic delay P (equal to the sum of parasitic delays in the $D-Q$ path stages), which is equal to $27.3\text{ps} \approx 1.5FO4$. This confirms the agreement of the theoretical results based on the LE optimization and that performed in the $E-D$ space.

In order to evaluate the impact of the local interconnections, the optimized designs with and without the interconnect parasitics (see Paragraph 4.7) are compared under the same conditions. The resulting optimum IDVs that minimize the metrics E^iD^j are summarized in Tab. IV.II, while the EECs and the location of the minimum E^iD^j points in the $E-D$ space are plotted in Fig. 4.10 (both the curves with the parasitics extracted as showed in Paragraph 4.7 and through post-layout simulations on actual layouts are reported).

By inspection of Tab. IV.II and Fig. 4.10, the optimum circuit sizes considerably change when the parasitics are taken into account or not. Indeed, as shown in Paragraph 4.7, the layout parasitics can increase the capacitances in the circuit nodes by more than a factor two. In particular, as expected the optimization leads to larger transistors sizes (up to 2X) when parasitics are accounted for, in order to compensate the resulting speed degradation. As a result, the energy increases both for the additional interconnections capacitances themselves and for the larger transistors sizes when interconnect parasitics are considered. From Tab. IV.II, the energy (delay) without parasitics is underestimated by a factor of up to 1.8 (1.4), compared to the evaluation with parasitics. Moreover, the energy/delay estimation when extracting the layout parasitics with the method in Paragraph 4.7 is highly accurate (see Fig. 4.10). Quantitatively, the average relative error on energy and delay with respect to the actual post-layout simulations is equal to -4.4% and -5.1%, respectively.

The above considerations confirm that interconnections parasitics must be necessarily considered from the beginning in the transistor-level optimization, as opposed to the approach adopted in most works that consider parasitics only subsequently (or do not consider at all).

Finally, to further validate the effectiveness of the whole design methodology, the energy-to-delay is also reported in Tab. IV.2. From this table, the assumptions adopted allow to find sensitivity values in good agreement with the theoretically predicted ratios $-j/i$. This again confirms the efficacy of the LE-based procedure used to bound the design space (Paragraphs 2.3.2 and 4.4) and of the optimization algorithm in the $E-D$ space (Paragraph 2.3.3 and 4.5).

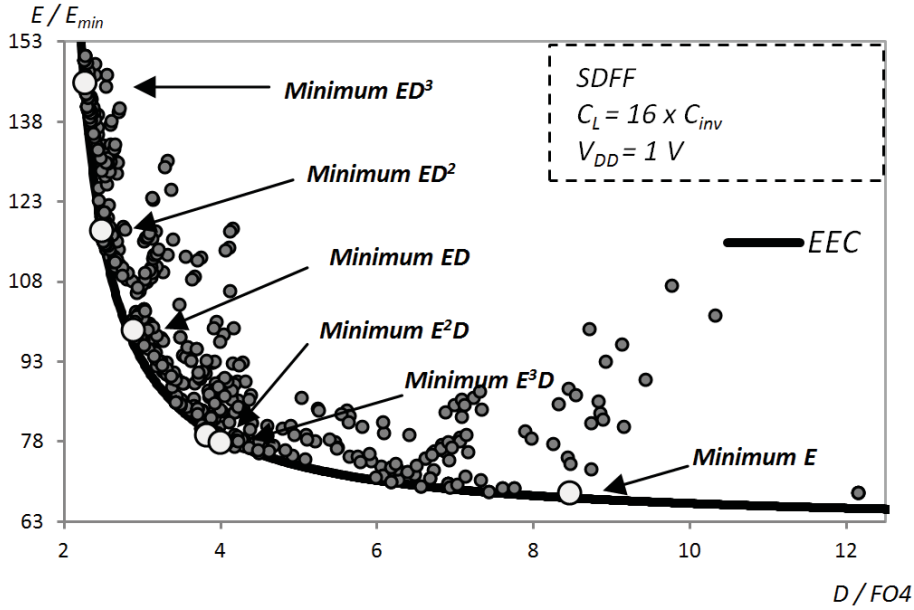


Fig. 4.9. $E - D$ space exploration for the SDFF.

TABLE IV.II: PROPERTIES OF THE MINIMUM $E^i D^j$ DESIGNS FOR THE SDFF IN PRESENCE AND IN ABSENCE OF LOCAL WIRES' PARASITICS INCLUSION

		w_1	w_2	w_3	w_4	E [E_{min}]	D [FO4]	S_D^E	$-\frac{j}{i}$
No C_{par}	Min ED^3	17	16	4	7	96.91	2.05	-3.09	-3.0
	Min ED^2	10	12	3	6	71.07	2.34	-1.82	-2.0
	Min ED	5	8	2	5	55.83	2.84	-0.94	-1.0
	Min E^2D	4	6	2	3	51.23	3.21	-0.53	-0.5
	Min E^3D	3	3	1	2	47.50	3.83	-0.34	-0.3
	Min E	1	2	1	1	44.58	6.13	-0.13	-0.0
With C_{par}	Min ED^3	24	20	5	9	145.30	2.26	-2.89	-3.0
	Min ED^2	16	17	4	7	117.63	2.46	-1.87	-2.0
	Min ED	10	15	3	6	99.08	2.87	-0.95	-1.0
	Min E^2D	5	10	2	4	79.38	3.82	-0.44	-0.5
	Min E^3D	4	9	2	3	78.02	3.98	-0.30	-0.3
	Min E	1	2	1	1	68.43	8.45	-0.10	-0.0

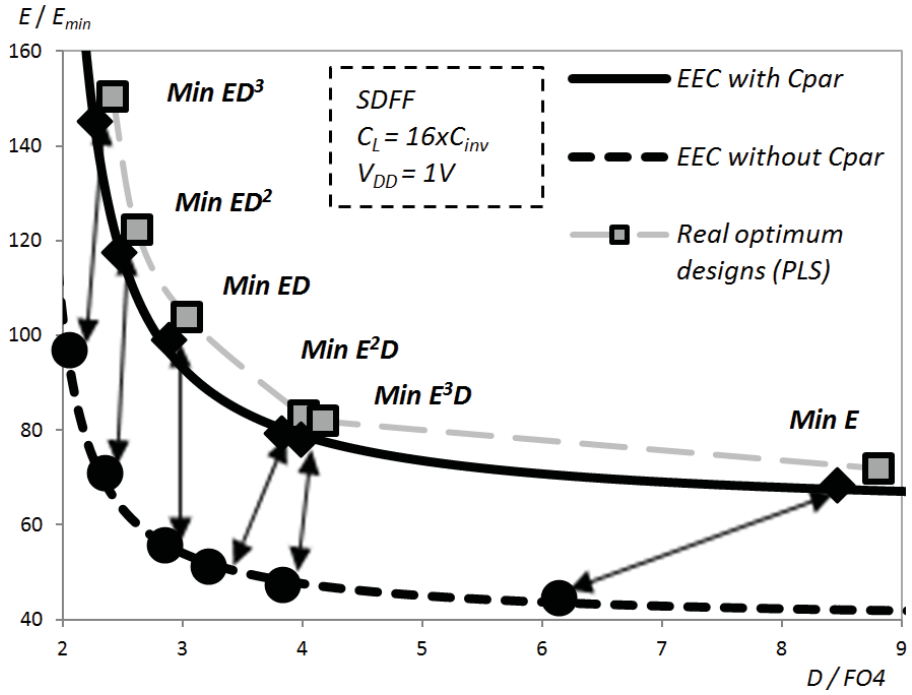


Fig. 4.10. EECs of SDFF with and without interconnect parasitics.

4.7 Estimation of Layout Parasitics in Transistor-Level Design Iterations

A simple though accurate methodology allowing for fast extraction of local wires' parasitic capacitances, which does not need the detailed layout drawing, is highly desirable to efficiently explore the space of design solutions [ACP10-2].

A primary and necessary step is the definition of the layout style which one refers to. In the following let us consider standard cell CSEs layouts, given that standard cells approach allows to simplify the physical design of complex ICs and easily automate the place & route [CK02], [SS02], [MC79]. As it is shown below, the use of stick diagrams [RCN03] relative to standard cell CSE layouts permits to fulfill the targeted parasitics extraction.

To improve the readability of this paragraph, it is split into three subsections. The first one describes the basic principles and the main assumptions adopted in the use of stick diagrams related to the standard cells. The second one contains a detailed example of typical formulations which one arrives to. The third one provides results relative to the SDFF

practical example to validate the accuracy of the method and its relevancy (already discussed at the end of Paragraph 4.6).

4.7.1 Estimation of layout parasitics from stick diagrams

A generic interconnection within a standard cell consists of several vertical and horizontal segments. Since the associated parasitic capacitance is proportional to the interconnection length (via the capacitance per unit length), its estimation is equivalent to evaluating the length of the vertical and horizontal segments (i.e., it becomes a geometrical problem). To avoid drawing a complete layout for each transistor sizing during the optimization, one can resort to stick diagrams [RCN03], which geometrically describe the relative position of each transistor to be connected with respect to the others. In particular, the length of horizontal tracts depends on the geometrical horizontal width of transistors structures which the interconnections runs over or in parallel. The effective (for parasitics extraction) length of vertical tracts is obtained subtracting the channel width of driven (by the specific interconnection) transistors to the whole vertical extension.

To simplify the geometrical analysis, let us refer to standard cell layouts with the following usual features:

a) Metal1 and Metal2 are used as interconnect layers within the cells ("over-the-cell" routing).
b) Vertical Poly lines and horizontal diffusion lines (having a greater density compared to Weinberger approach [RCN03]).
c) Height of the cell layout equal to 16 Metal2 tracks [CK02].
d) Width of V_{DD} and GND rails equal to 3 Metal1 tracks [SS02].
e) All the gates that can share their diffusions are connected together and the common diffusions among series-connected transistors are shared too. Also the Poly wires are shared among PMOS/NMOS transistors driven by the same input.
f) Cell height equally shared among PMOS and NMOS transistors (i.e., n- and p-well have the same height).
g) If the entire cell height is not needed, the transistors are located near the half cell height (not near the supply rails) to reduce Poly-input and Metal-output lines length.
h) For three series-connected transistors structures, the folded layout technique is applied by adopting the Euler Path method, in order to employ a unique diffusion line [RCN03].

From the above assumptions and layout design rules, the maximum (vertical, from assumption b) width w_{max} of the diffusion regions within an n-/p-well is immediately identified for NMOS/PMOS transistors. Hence, transistors whose width w is lower than w_{max} are implemented with a single poly/diffusion strip, as usual. On the other hand, transistors with $w > w_{max}$ must be implemented as multiple folded transistors in parallel, each of which

is implemented with a single poly/diffusion strip and has $w = w_{max}$. The number k of these parallel folded transistors is $k = w/w_{max}$. According to these observations, the geometrical height of a generic transistor is w if $w < w_{max}$, and equal to w_{max} if $w > w_{max}$. On the other hand, the geometrical width is obviously proportional to the number k of parallel folded transistors, since they are placed by abutment. A detailed example is reported in the next subsection.

Finally, the capacitances per unit length should be extracted from the design kit information by including the fringing contribution due to coupled lines between stacked buses, which is usually reported as the worst-case interconnect capacitance (as opposed to the best-case value obtained for an isolated interconnect above a ground plane). Inclusion of the fringing contribution is very important to correctly estimate layout parasitics, as the worst-case value can be typically 3 times the best-case value [ACP10-2].

4.7.2 A detailed example: geometrical width of folded transistors

To better understand how to evaluate the geometrical width of transistors, let us consider an exemplifying case depicted in Fig. 4.11, where a folded layout of a single PMOS with the source tied to V_{DD} is shown. By inspection of Fig. 4.11, assuming minimum-width wires and neglecting the small segments crossing the active regions, the overall length of the Poly (gate) and Metal1 (drain) wires result to:

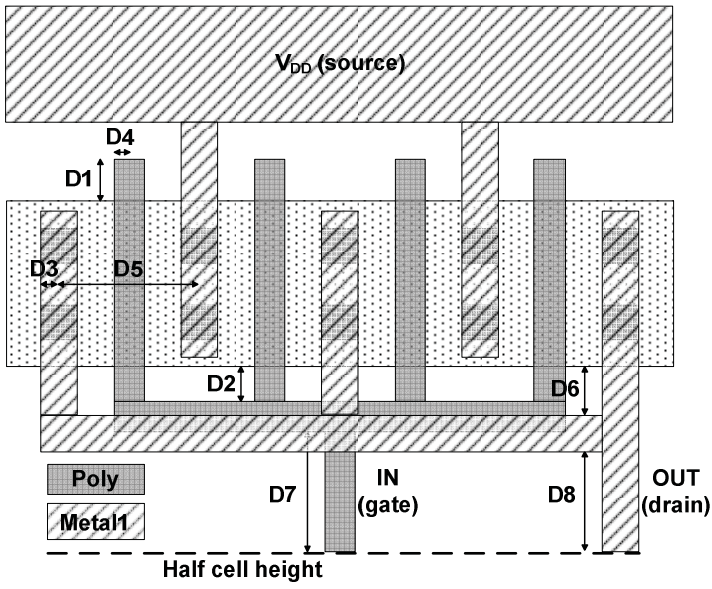


Fig. 4.11. PMOS with source at V_{DD} and local gate- and drain- wires.

$$L_{Poly} = [k]^{+}(D_1 + D_2) + 2D_4 + ([k]^{+} - 1)D_5 + D_7 \quad (4.10)$$

$$L_{Metal1} = 2D_3 + 2\left[\frac{k+1}{2} - \varepsilon\right]^{-} D_5 + \left(\left[\frac{k+1}{2} - \varepsilon\right]^{-} + 1\right) D_6 + D_8 \quad (4.11)$$

In particular:

- $[x]^{-}$ is the closest integer smaller than x (floor)
- $[x]^{+}$ is the closest integer larger than x (ceil)
- $D_1 - D_4$ are technology-dependent layout design rules
- $D_5 - D_8$ derive from the combination of the above design rules; in particular, $D_7 - D_8$ depend on the minimum distance between the active region and the half-cell height (see Fig. 4.11), which is dictated by the space around the half-cell height that is reserved for horizontal Metal wires to connect transistors.
- The Poly length over the diffusion, which defines the transistor gate capacitance, is not included in the computation since this contribution is already managed by the simulator even without the local wires' parasitics inclusion.
- To manage the folded-layout technique related constraints once that w_{max} and w_{min} are given, the number ε appearing in some of the above terms has to be smaller than $(w_{max} + w_{min})/w_{max}$. For instance the second and third terms in (4.10) count the number of D_5- and D_6-long Metal tracts and their general validity can be verified by visual inspection for any k value (in Fig. 4.11, $3 < k \leq 4$).

(4.10)-(4.11) clearly show how the parasitics extraction translates into a geometrical problem and the functional dependence of such a methodology on transistor widths (through the parameter k). The same procedure can be applied to evaluate the length of horizontal wires crossing single devices, two- and three- series-connected MOS structures, keepers, pass-transistors, transmission gates and so on.

4.7.3 The SDFF case of study

As an example, let us consider the SDFF sized for minimum ED product (under the same conditions considered in Paragraph 4.6). Its stick diagram and the corresponding layout are reported in Fig. 4.12 and Fig. 4.13. In the considered 65-nm CMOS technology, the capacitance per unit-length of poly, metal1 and metal2 (including fringing contributions) are $c_{Poly} = 282$ aF/μm, $c_{Metal1} = 231$ aF/μm and $c_{Metal2} = 189$ aF/μm. By applying the above procedure, the estimated interconnections capacitances at the circuit nodes are summarized in Tab. IV.III, together with the values extracted by the parasitic extractor tool from the layout in Fig. 4.13 (the node names are indicated in Fig. 4.8). By inspection of Tab. IV.III, the estimated capacitances are always within 18% of those extracted from the layout, and

typically within 10%. This confirms that the proposed approach is sufficiently accurate to estimate layout parasitics.

Finally, the impact of interconnections parasitics on the overall capacitance at each node is evaluated. To this aim, the sum of the transistor (both gate and drain) capacitances at each node is reported in the fifth column in Tab. IV.III. The ratio of the overall node capacitance (i.e., due to both transistors and interconnections) and the contribution of transistors is reported in the last column.

The latter shows that the layout parasitics can increase the capacitance by more than 2, which confirms that local wires' parasitics must be necessarily taken into account in the transistor sizing design phase. This also justifies why such a procedure was introduced, as opposed to previous works that completely neglected layout parasitics [SO99], [HA01], [MNB01], [TNC01], [O02], [O03], [OSM03], [NO05], [HKA07], [GNO07].

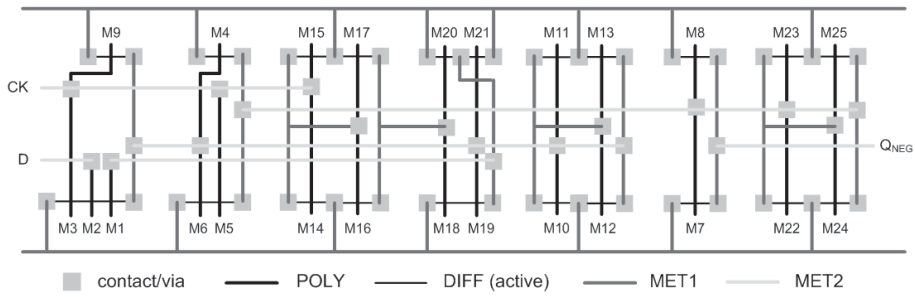


Fig. 4.12. Stick diagram of the SDFF.

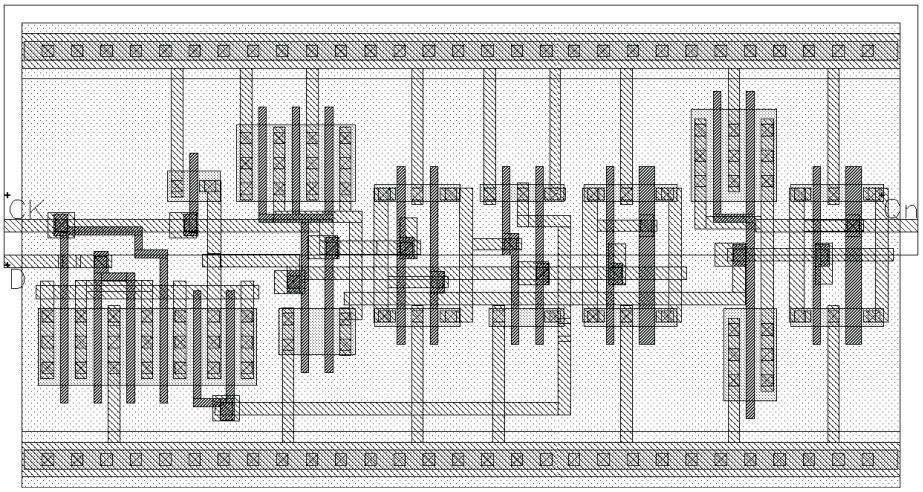


Fig. 4.13. Layout of the SDFF (min. *ED* product sizing).

TABLE IV.III: PARASITIC CAPACITANCES ESTIMATION IN SDFF

Node ↓	Estimated $C_{par,i}$ (fF)	Extracted $C_{par,i}$ (fF) (a)	Error (%)	Transistor cap. (fF) (b)	Cap. increase due to wires $(a+b)/b$
D	0.33	0.39	-14.1	1.37	1.29
CK	1.43	1.65	-13.7	2.32	1.71
Q_{NEG}	1.16	1.42	-18.0	9.02	1.16
1	3.08	3.48	-11.5	4.92	1.71
2	0.73	0.82	-11.0	0.82	1.99
3	0.60	0.72	-17.3	0.55	2.32
4	0.62	0.71	-12.5	0.55	2.30
5	3.11	3.43	-9.4	5.47	1.63
6	0.73	0.81	-10.0	0.82	1.98
7	1.81	2.22	-18.6	1.78	2.25
8	0.00	0.01	-	0.82	1.01
9	0.57	0.65	-11.8	2.73	1.24
10	0.34	0.40	-14.7	2.73	1.15

Appendix 4
Reconsidering High-Speed Design Criteria for Transmission-Gate Based Master-Slave Flip Flops

Transmission-gate (or pass-transistors) -based Master-Slave (TGMS) FFs are among the most popular and simplest CSE topologies, and many of them have been proposed in the past [SOA73], [GGD94], [LS96], [KB00], [MNB01], [MTD03], [HMA05]. They are featured by a small area occupation, by few internal nodes to be charged and discharged and by the absence of precharge. All these factors lead to a small dissipation and hence TGMS FFs can be effectively employed in energy-efficient microprocessors.

Traditionally, LE optimization is carried on by looking at the whole circuit as a unique uninterrupted path [OSM03], [SSH98]. Actually, for this specific class of circuits, the problem of delay minimization has to be looked from a different perspective by resorting to a novel approach [CPP11], [CPP12]. The LE basis is still exploited but, unlike the traditional methodology, TGMS FFs are split in two overlapping sections and two different paths that are separately optimized. In particular, the paths considered are the first part of the one considered in the traditional methodology and the CK-to-Q one. As shown in the following, breaking the D-to-Q path instead of considering it as a whole leads to the actual delay

minimization. Remarkably, also energy consumption and area occupation of the resulting designs are always significantly lower than those obtained with the traditional LE method [CPP11], [CPP12].

Therefore, this means that the actual path effort of TGMS FFs is more properly handled through such a new approach, whereas the traditional one fails to correctly catch it. These considerations can be practically exploited when sizing these circuits in the high-speed energy-efficient design region, i.e. as a base (or as a starting point) when accounting also for energy in the minimization of energy-delay products ED^j with $j > i$.

A.4.1 Timing behavior of TGMS flip-flops

As explained in Chapter 3, CSEs can be basically split into two topological categories: Pulsed CSEs and Master-Slave FFs [PCB01]. The former are featured by an internally or externally generated time window during which the CSE is transparent to the input data. Such a time-window implies a) a flat minimum region in the τ_{D-Q} vs. τ_{D-CK} curve, b) a negative t_{setup}, c) a continuous topological path from D to Q since D is the critical input when considering $\tau_{D-Q,min}$ as the figure of merit for CSE speed.

On the contrary, Master-Slave FFs are constituted by two latches that are alternately transparent according to the CK value. This implies: a) an high τ_{D-Q} to τ_{D-CK} sensitivity in the minimum region, b) a positive t_{setup}, c) the presence of two distinct paths from the input node D to the boundary node between Master and Slave sections, and from this node to the output Q.

To exemplify the above discussions, let us consider the generic structure of a transmission-gate, TG, (or pass-transistor, PT) –based Master-Slave (TGMS) FF shown in Fig. 4.14 (the depicted inverters can stand for generic combinational blocks, while keepers and/or feedback paths are not shown). The node X is the boundary between Master and Slave sections and the paths relative to τ_{D-CK}, τ_{CK-Q} and τ_{D-Q} delays are depicted with grey lines.

When τ_{D-CK} is sufficiently large, the input signal traverses the Master latch and stops at node X, waiting for the Slave TG to be enabled by the falling clock transition. After that, the input is transferred to the output.

On the contrary, when $\tau_{D-CK} = t_{setup}$, the last gate in the Master section (henceforth referred as block A, as shown in Fig. 4.14) transfers its input nearly contemporarily to the enabling of the TG (henceforth referred as block B, as shown in Fig. 4.14) in the Slave section. However, as shown in the following, the traditional assumption of an uninterrupted path from D to Q [OSM03], [SSH98] is not consistent.

An indication of such an incongruence arises since, assuming the union of blocks A and B as a single stage (they are performing logical operations at the same time), it is not clear if the critical input signal to be considered is the input of block A or the CK signal enabling block B.

Therefore, in order to optimize the speed of TGMS FFs in terms of $\tau_{D-Q,min}$, rather than applying the LE method to the whole $D-Q$ path, a different approach may be required. In particular, we will show that t_{setup} and $\tau_{CK-Q,opt}$ (i.e. τ_{CK-Q} delay when $\tau_{D-Q} = \tau_{D-Q,min}$) have to be separately handled.

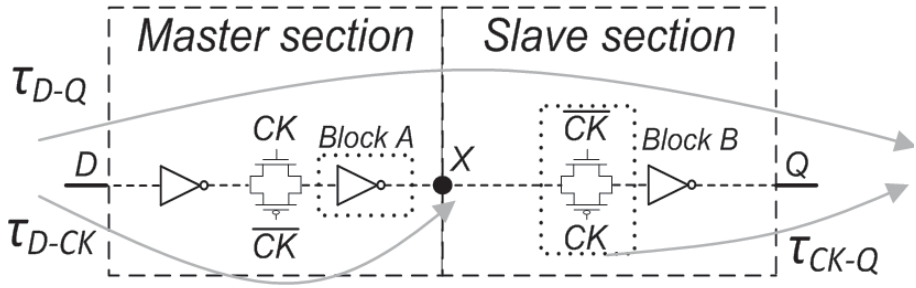


Fig. 4.14. Structure of a generic TG (or PT) –based FF.

A.4.2 High-speed design strategy for TGMS flip-flops

As explained in Chapter 1, the LE can be extended to the case where TGs (or PTs) are present, provided that TGs are incorporated to previous stages with driving capability.

Anyhow, the logical effort parameters g, h and p can be even more accurately extracted by applying the Elmore delay model, which is easily adaptable in a LE fashion [WH04], [MFG10]. For this reason, in the following the LE tool is employed by basing on the more accurate Elmore delay interpretation. Finally, it is worth noting that, as suggested in [SSH98], the most effective approach is to equally size the PMOS and NMOS transistors composing a TG[2].

The general rule when considering a transparent TG driven by a static gate is to size its transistors equally to those of the preceding gate. For instance, in the usual case where the previous gate is a simple inverter (INV), the input of such an inverter will be the critical signal (the TG is transparent). The highest speed and symmetrical rising/falling behavior of the whole block INV+TG is achieved by sizing the PMOS of the INV with a

[2] It can be assumed that equally sized PMOS / NMOS PTs exhibit a resistance equal to $4R$ / R when transferring a logic "0" and $2R$ / $2R$ when transferring a logic "1" [SSH98] (assuming NMOS mobility twice that of PMOS). Therefore, a TG with equally sized PMOS and NMOS transistors exhibits a resistance nearly equal to R for both "1" and "0" inputs. There is no point in increasing the size of the PMOS (as usually done in static/dynamic gates with driving capability) since, by sizing the PMOS twice the NMOS, the TG resistance is equal to $(2/3)R$ for both "1" and "0" at the input but the capacitances at the input and output of the TG increase by 50%.

width twice the NMOS (assuming NMOS mobility twice that of PMOS) and the transistors in the TG with the same width of the NMOS of the INV.

When sizing transistors at the boundary between Master and Slave, as in the case of blocks A and B (see Fig. 4.14), the purpose is still to achieve symmetrical and minimum rising and falling delays. But it is less evident how to set the relative size between the two blocks, since this time the TG can be enabled slightly before, contemporarily or slightly later than the time when combinational block (considered as a simple INV for simplicity) begins to transfer its input.

To resolve the doubt one can consider only the two blocks A and B, as shown in Fig. 4.15, and feed them directly with the D input, loading block B with a capacitance C_O. The minimum delay[3] from D to output O is analyzed for various sizes of the INV and various values of C_O, by varying the size W_{TG} of the TG (smaller, equal or larger than the NMOS width in the INV, W_{INV}). Again, it is found that a symmetrical and minimum delay is obtained by sizing the PMOS ($2W_{INV}$) twice the NMOS (W_{INV}) and $W_{TG} = W_{INV}$.

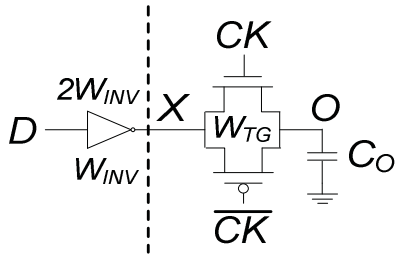


Fig. 4.15. Gates at the boundary between Master and Slave latches.

Once established that the blocks at the boundary between Master and Slave are identified by a single width, there is the need to set their absolute size, together with the size of the remaining stages in the TGMS FF, in order to minimize $\tau_{D-Q,min}$.

The traditional approach would be to consider a unique path from D to Q and apply the LE method with a number of stages N given by all the gates in Master and Slave sections. The normalized width (with respect to the minimum value W_{min}) of the first stage, w_1, as well as the equivalent normalized width of the load, w_L, are obviously assumed as known

[3] In the case of Fig. 4.12, the delay from D to O diminishes by decreasing the interarrival time between D and CK up to reaching a constant minimum. Conversely, in a whole TGMS FF the delay increased after having reached the minimum, since a too small D to CK interarrival time means that the input is not well captured (or not captured at all) before TG in the Master is disabled.

parameters. Keepers and feedback paths have typically a fixed size, and hence lead to non constant branching effects, i.e. to nonlinearities. Therefore, an iterative procedure is required to satisfy the LE optimum condition of an equal stage effort among all FFs stages.

The novel approach [CPP11], [CPP12] breaks up the optimization in two steps. In particular, two LE optimizations are carried out to minimize the delays from input D to node X (path 1) and from CK (enabling the Slave TG) to output Q (path 2), and then the results are reconciled.

Such an approach is intuitively justifiable since the signals coming from D and CK, which traverse block B, would experience a different effort according to the classic interpretation. Hence, though blocks A and B nearly contemporarily act when the condition $\tau_{D-CK} = t_{setup}$ is satisfied, two distinct overlapping paths can be identified. Such paths are not simply restricted to Master and Slave sections. Indeed, the first delay (up to node X) is influenced by the enabled block B in the Slave (and hence by the input capacitance of the gate that follows block B) and the second delay is influenced by the resistance introduced by block A.

According to the above point of view, the overall path effort is hence more appropriately broken into two separate contributions and, rather than according to

$$opt\left(t_{setup} + \tau_{CK-Q,opt}\right) \tag{A.4.1}$$

the minimum $\tau_{D-Q,min}$ is actually found according to

$$opt\left(t_{setup}\right) + opt\left(\tau_{CK-Q,opt}\right) \tag{A.4.2}$$

where the notation $opt(T)$ means that the delay T is optimized by applying the LE method.

According to (A.4.2), two sets of LE parameters have to be derived for the paths 1 and 2 and condition (1.36) is applied to both paths. Note that the input capacitance of the gate following block B is considered as the final load for path 1, while blocks A-B represent the first stage for path 2.

Further arrangements are necessary to properly define the LE equations according to the FF topology (the examples in the next paragraph clarify many practical aspects). Nevertheless, it is worth anticipating that, by separately optimizing path 1 and path 2 and then reconciling the results, a unique possible size for blocks A-B (and hence also for all the other gates) comes out, just like in the traditional LE approach.

Moreover, another aspect strengthening the above point of view concerns the role played by variability when TGMS FFs are employed in the critical paths of pipelined schemes. Indeed, due to their high τ_{D-Q} to τ_{D-CK}

sensitivity in the minimum region, TGMS FFs have to be actually operated with a τ_{D-CK} slightly larger than t_{setup} (which leads to the very $\tau_{D-Q,min}$ delay) in order to guarantee a sufficient margin to absorb the impact of process-environmental variations and external clock skew-jitter. Yet, when employed in critical paths, TGMS FFs still obviously work under the condition where there is a certain overlap between the operations of the blocks A and B , i.e. under the condition where the figure of merit for speed is still the τ_{D-Q} delay and not only the τ_{CK-Q} one (as in fast paths). Hence, a LE based optimization targeting the D-Q path is still consistent to minimize the impact of TGMS FFs timing on the clock period.

Given all of the above, right due to the margin that has to be provided on τ_{D-CK}, the assumption of splitting the D-Q path in two sections becomes even more consistent and justifiable with respect to the traditional one[4].

A.4.3 Design example: TGFF

To exemplify the novel approach, one can choose the typical TGFF [MNB01], already considered in Paragraph 4.2.1. It is a modified version of the well-known FF employed in the PowerPC 603 low power processor [GGD94]. In particular, an inverter is added to isolate the D input and provide better noise immunity. The input is transferred to the output with inverted polarity, $\bar{Q}$, and simple gated keepers are employed.

The normalized widths (with respect to the minimum value W_{min}) of the various stages are highlighted in Fig. 4.16a (the keepers are minimum sized). In particular, the first INV+TG block ($M1 - M4$) in the Master has the width w_1 given by the FF C_{IN}. Blocks A-B correspond to $M5 - M8$ and are identified by a width w_2, while INV $M9 - M10$ is identified by a width w_3[5].

If one considers the traditional LE approach, the LE parameters relative to the various stages are those in Tab. IV.IV (w_L is the equivalent load width). The stages corresponding to the LE parameters in Table I are shown in Fig. 4.16 for exemplification. In this case the LE method has to be applied by assuming a number of stages $N = 3$ (nonlinear equations arise due to branching and hence the solution has to be found iteratively).

As anticipated, the Elmore delay model is applied to determine the expressions of delays of blocks $M1 - M4$ and $M5 - M8$ (from which LE parameters are then extracted). Note that, in Table IV.IV, capacitive terms are between parentheses and are multiplied by the resistances from each node to V_{DD}/GND. Diffusion capacitance introduced by each transistor is

[4] When increasing τ_{D-CK} with respect to t_{setup}, one is getting closer to (but not really reaching) the condition in which block A fully completes its operation before block B is enabled. This reinforces the intuition according to which the paths up to and after node X have to be separately handled.

[5] PMOS $M2$, $M6$, $M10$ actually have widths $2w_1$, $2w_2$ and $2w_3$, respectively.

equaled to its gate capacitance under the same width [SSH98] (it has been verified that they are nearly equal).

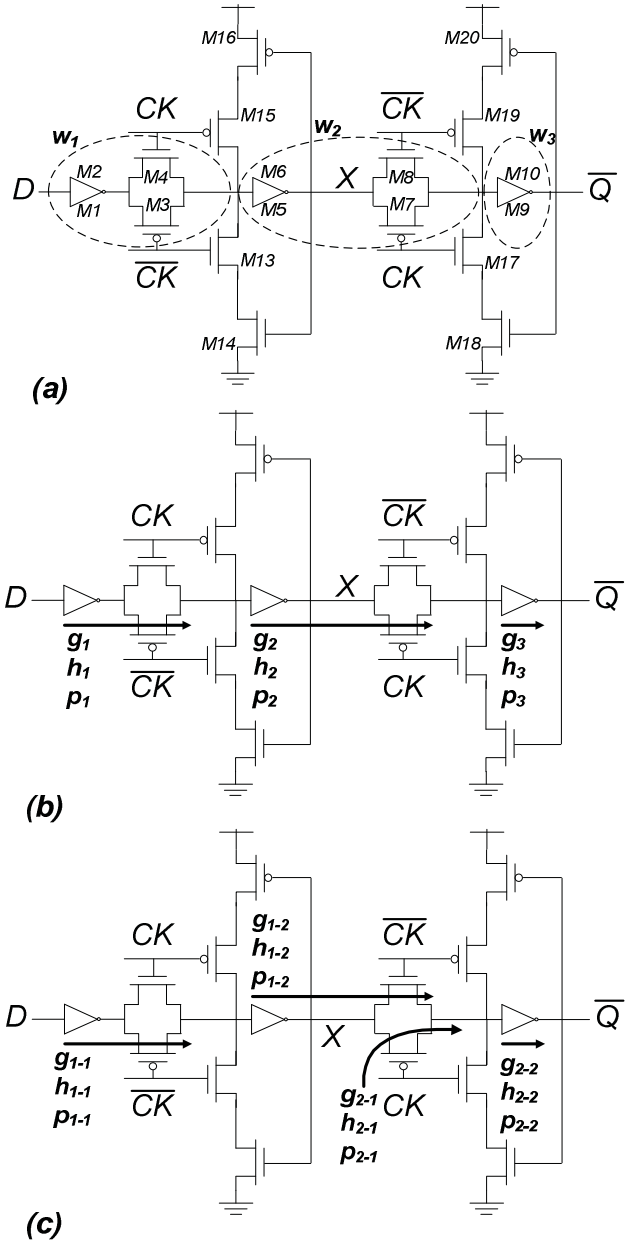


Fig. 4.16. Schematic of the TGFF (a) and LE parameters according to the traditional and proposed approaches.

TABLE IV.IV: LE PARAMETERS FOR THE TGFF CONSIDERED AS A WHOLE PATH (FROM D TO $\bar{Q}$) WITH $N = 3$ STAGES

Stage	***Normalized (in LE fashion) Elmore delay d***	***Logical effort g***	***Electrical effort h***	***Parasitic delay p***
1	$\frac{(5w_1)\frac{1}{w_1} + (2w_1 + 2 + 3w_2)\frac{2}{w_1}}{3}$	2	$\frac{2 + 3w_2}{3w_1}$	3
2	$\left[\frac{(5w_2 + 2)\frac{1}{w_2} + (2w_2 + 2 + 3w_3)\frac{2}{w_2}}{3} + \frac{(2w_2 + 2 + 3w_3)\frac{2}{w_2}}{3}\right]/2$	$\left[2 + \frac{2}{3}\right]/2$	$\left[\frac{3 + 3w_3}{3w_2} + \frac{2 + 3w_3}{w_2}\right]/2$	$\left[3 + \frac{4}{3}\right]/2$
3	$\frac{(3w_3 + 2 + w_L)\frac{1}{w_3}}{3}$	1	$\frac{2 + w_L}{3w_3}$	1

TABLE IV.V: LE PARAMETERS FOR THE TGFF CONSIDERED AS THE UNION OF TWO PATHS EACH WITH $N = 2$ STAGES

Path-Stage	***Normalized (in LE fashion) Elmore delay d***	***Logical effort g***	***Electrical effort h***	***Parasitic delay p***
1-1	$\frac{(5w_1)\frac{1}{w_1} + (2w_1 + 2 + 3w_2)\frac{2}{w_1}}{3}$	2	$\frac{2 + 3w_2}{3w_1}$	3
1-2	$\frac{(5w_2 + 2)\frac{1}{w_2} + (2w_2 + 2 + 3w_3)\frac{1}{w_2}}{3}$	1	$\frac{4 + 3w_3}{3w_2}$	$\frac{7}{3}$
2-1	$\frac{(2w_2 + 2 + 3w_3)\frac{2}{w_2}}{3}$	$\frac{2}{3}$	$\frac{2 + 3w_3}{w_2}$	$\frac{4}{3}$
2-2	$\frac{(3w_3 + 2 + w_L)\frac{1}{w_3}}{3}$	1	$\frac{2 + w_L}{3w_3}$	1

Moreover, the resistance reduction exhibited by stacked transistors due to velocity saturation is neglected, since, in the adopted 65-nm technology, it is nearly compensated by strong channel length modulation and DIBL effects.

Regarding the parameters of the second stage, they are derived by averaging out two different cases:

a) the case where input of INV $M5 - M6$ is the critical signal;
b) the case where CK enabling TG $M7 - M8$ is the critical signal.

It is verified that such an assumption leads to the best results for the traditional $N = 3$ LE procedure with respect to the case of simply assuming the input of INV $M5 - M6$ as the critical signal, and hence it is the fairest choice to point out any possible merit of the novel approach.

Considering the proposed approach, two sets of LE parameters are derived for the $N = 2$ paths 1 and 2 (referred through the first subscript) and are reported in Tab. IV.V. The stages corresponding to the LE parameters in Tab. IV.V are shown in Fig. 4.16c for exemplification. Note that, obviously, the first and last rows of Tabs. IV.IV and IV.V are equal.

As concerns the first delay, the Elmore model is applied to estimate the delay up to node X, while, as concerns the second delay, the capacitance at

node X is assumed as already charged or discharged through $M5 - M6$.

The application of condition (1.36) to both paths leads to

$$g_{1-1}h_{1-1} = g_{1-2}h_{1-2} = \sqrt{F_1} = \sqrt{G_1 B_1 H_1} \quad \text{(A.4.3)}$$

$$g_{2-1}h_{2-1} = g_{2-2}h_{2-2} = \sqrt{F_2} = \sqrt{G_2 B_2 H_2} \quad \text{(A.4.4)}$$

$$G_1 = g_{1-1}g_{1-2} \quad / \quad G_2 = g_{2-1}g_{2-2} \quad \text{(A.4.5)}$$

$$B_1 H_1 = h_{1-1}h_{1-2} \quad / \quad B_2 H_2 = h_{2-1}h_{2-2} \quad \text{(A.4.6)}$$

where g_{i-j} (h_{i-j}) is the logical (electrical) effort of the j-th stage in the i-th path. According to Tab. IV.V, (A.4.3)-(A.4.6) are solved by setting

$$w_2 = \frac{-6+\sqrt{36+18(4+3w_3)(3w_1)}}{18} \quad \text{(A.4.7)}$$

$$w_3 = \frac{-6+\sqrt{36+54(2+w_L)(w_2)}}{18} \quad \text{(A.4.8)}$$

The equations (A.4.7)-(A.4.8) have to be satisfied to contemporarily minimize t_{setup} and $\tau_{CK-Q,opt}$ according to (A.4.2). Practically, by substituting (A.4.8) into (A.4.7) (or vice versa), a single variable equation comes out and w_2 and w_3 can be easily identified (w_1 and w_L are given and a simple iterative cycle is sufficient to solve the arising nonlinear equation).

A.4.4 Simulation results

Starting from the LE parameters in Tabs. IV.IV-IV.V, the TGFF is sized according to the traditional and suggested approaches and the actual energy and delay are extracted by means of simulations. Various loading and input capacitance conditions are explored and, in particular, w_1 is varied in the range $[5 - 35]$, whereas w_L in the range $3 \times [5 - 35]$ (i.e., a load equal to $[5 - 35]$ minimum symmetrical inverters with $W_P = 2W_N = 2W_{min}$). For practical reasons, only the realistic cases where $w_L > w_1$ are considered.

The average delay (normalized to $FO4 = 18.3\text{ps}$) and the energy dissipation under a 0.25 data input switching activity (normalized to $E_{min} = 0.202\text{ fJ}$) are shown in Fig. 4.17a-b, respectively, for the TGFF optimized according to the proposed procedure. The relative differences on delay and energy between the proposed sizing strategy and the traditional one are shown in Fig. 4.18a-b, respectively[6].

By inspection of results in the figures, the suggested procedure always outperforms the traditional one in terms of speed performance, with quantitative improvements ranging from 1% to 23% and increasing with w_L.

[6] The relative differences are obtained as $(P_A - P_B)/[(P_A + P_B)/2]$, being P_A the parameter (delay or energy) relative to the traditional sizing strategy and P_B the parameter relative to the proposed sizing strategy.

Even more interestingly, the dissipation (and area) of the suggested approach is significantly lower than that of the traditional one, which reduces from 6% to 57% (increasing for larger w_L).

Indeed, as concerns the sizing, the w_2 and w_3 values found with the proposed methodology are always lower than the ones found with the traditional approach (nearly by 30% and 50% factors). For instance, when considering the case with load equal to 16 minimum symmetrical inverters and $w_1 = 4$, the traditional approach would lead to $[w_2, w_3] = [7.5/13.8]$, while the proposed one to $[w_2/w_3] = [5.4/6.8]$. Despite the smaller sizing, the proposed approach leads to 6% (40%) better delay (energy). The above results imply that, when optimizing $\tau_{D-Q,min}$, the assumption of two split paths is more consistent than that of a single path, which unnecessarily overestimates the actual path effort in the case of Master-Slave topologies.

In particular, by combining the above results, it is apparent that the energy-efficiency of the suggested sizing strategy is significantly improved. Therefore, the traditional sizing strategy, which assumes a TGMS FF as a whole path, does not actually correspond to the best solution in terms of an high-speed optimization that accounts for energy consumption too.

Given the above results, the suggested approach can constitute a base (or a starting point) for the optimization of this class of FFs in the high-speed region of the energy-delay space, that is the region where products $E^i D^j$ with j significantly larger than i are minimized.

To verify this statement, the sizing strategies obtained with the proposed approach are compared with those resulting by applying the simulations-driven optimization algorithm described in this chapter under the constraint of minimum ED^4 energy-delay product. The minimization of such a figure of merit exemplifies a design strategy that primarily targets speed. The optimizations are carried out by combining the ranges $[1 - 19]$ and $3 \times [1 - 19]$ for w_1 and w_L, respectively (some rows, relative to non-practical $w_L \ll w_1$ cases, are highlighted in grey).

The w_2 and w_3 values obtained through the traditional procedure, through the proposed one and through the simulations-driven optimization algorithm are reported in Tab. IV.VI. By inspection of results, the relative percentage error of the proposed procedure in the sizes of transistors is moderate and typically within 20% except for few cases (due to the very small w_i values). On the contrary, it is apparent that the traditional approach leads to an unnecessary strong over-sizing.

To further exemplify the energy-delay space region where it is worth using such an approach, in Tab. IV.VII the sizing, energy and delay are reported for the proposed approach, and for the minimum ED^4 and minimum ED designs arising from the optimization algorithm, with 13 loading inverters and $w_1 = [1,7,13,19]$.

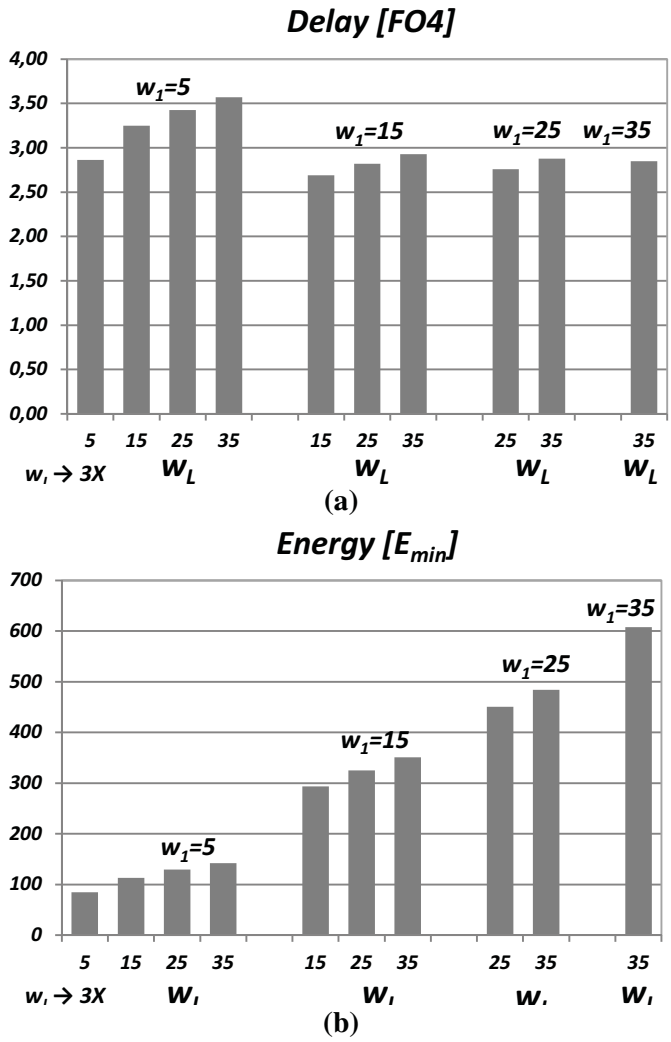


Fig. 4.17. TGFF: delay (a) and energy (b) obtained with the novel approach.

It is apparent that the designs found with the proposed methodology are close to the energy-efficient one in the high-speed region of the E-D space, i.e. results further demonstrates that the proposed procedure allows to closely approach the energy-efficient sizings minimizing the figures of merit where speed is the primary concern. As pointed out in this chapter, minimum delay designs represent a bound for the design space and can be used as starting point for the optimized search within it. For this reason, a proper revision of the traditional LE method is necessary when optimizing circuits employing TGMS FFs.

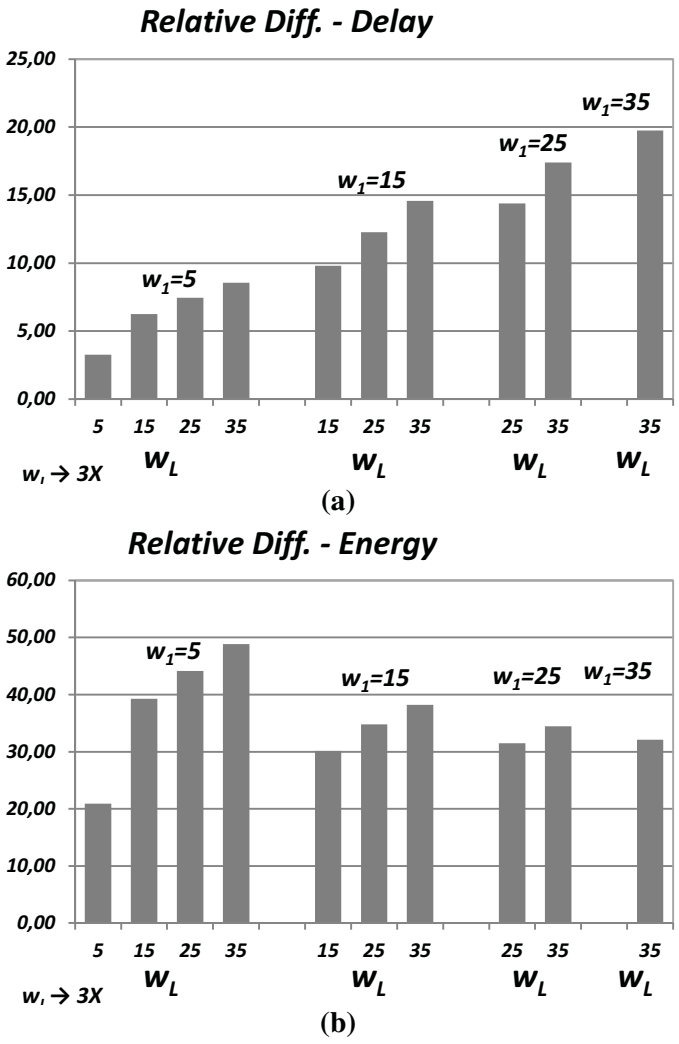


Fig. 4.18. TGFF: Relative percentage delay (a) and energy (b) differences between the traditional and proposed approaches.

TABLE IV.VI: ERROR BETWEEN MIN. ED^4 SIZINGS EXTRACTED WITH TRADITIONAL/PROPOSED PROCEDURES AND AN OPTIMIZATION ALGORITHM

First TGMS	*Traditional procedure*		*Proposed procedure*		*Optimization algorithm*		*Relative % error (proposed)*		*Relative % error (traditional)*	
$w_L - w_1$	w_2	w_3	w_2	w_3	w_2	w_3	w_2	w_3	w_2	w_3
3 – 1	**1.3**	**1.1**	**1.2**	**1.0**	**1**	**1**	**20.0**	**0.0**	**30.0**	**10.0**
3 – 7	**4.9**	**2.6**	**4.3**	**1.7**	**3**	**2**	**43.3**	**-15.0**	**63.3**	**30.0**
3 – 13	**7.5**	**3.4**	**6.4**	**2.2**	**6**	**2**	**6.7**	**10.0**	**25.0**	**70.0**
3 – 19	**9.6**	**3.9**	**7.8**	**2.4**	**7**	**2**	**11.4**	**20.0**	**37.1**	**95.0**
21 – 1	**2.1**	**4.5**	**1.4**	**1.6**	**1**	**2**	**40.0**	**-20.0**	**110.0**	**125.0**
21 – 7	**8.4**	**9.6**	**5.6**	**5.0**	**5**	**5**	**12.0**	**0.0**	**68.0**	**92.0**
21 – 13	**12.8**	**12.0**	**8.5**	**6.1**	**7**	**6**	**21.4**	**1.7**	**82.9**	**100.0**
21 – 19	**16.6**	**13.7**	**11.3**	**7.0**	**10**	**8**	**13.0**	**-12.5**	**66.0**	**71.3**
39 – 1	**2.6**	**7.1**	**1.9**	**3.5**	**2**	**4**	**-5.0**	**-12.5**	**30.0**	**77.5**
39 – 7	**10.2**	**14.6**	**7.5**	**7.3**	**6**	**6**	**25.0**	**21.7**	**70.0**	**143.3**
39 – 13	**15.6**	**18.2**	**11.3**	**9.1**	**10**	**9**	**13.0**	**1.1**	**56.0**	**102.2**
39 – 19	**20.2**	**20.8**	**14.6**	**10.4**	**13**	**9**	**12.3**	**15.6**	**55.4**	**131.1**
57 – 1	**2.9**	**9.2**	**2.1**	**4.6**	**2**	**5**	**5.0**	**-8.0**	**45.0**	**84.0**
57 – 7	**11.6**	**19.0**	**8.3**	**9.4**	**7**	**8**	**18.6**	**17.5**	**65.7**	**137.5**
57 – 13	**17.6**	**23.5**	**12.7**	**11.7**	**11**	**10**	**15.5**	**17.0**	**60.0**	**135.0**
57 – 19	**22.8**	**26.8**	**15.3**	**13.2**	**14**	**12**	**9.3**	**10.0**	**62.9**	**123.3**

TABLE IV.VII: SIZING, ENERGY AND DELAY FOR THE PROPOSED PROCEDURE, MIN. ED^4 AND MIN. ED SIZINGS IN SOME REFERENCE CASES

	Proposed procedure				*Minimum ED^4 sizing*				*Minimum ED sizing*			
w_L w_1	w_2	w_3	*Energy [E_{min}]*	*Delay [FO4]*	w_2	w_3	*Energy [E_{min}]*	*Delay [FO4]*	w_2	w_3	*Energy [E_{min}]*	*Delay [FO4]*
39 1	**2.1**	**3.4**	**48.05**	**4.09**	**2**	**3**	**47.31**	**4.08**	**1**	**2**	**22.8**	**5.0**
39 7	**8.8**	**8.1**	**156.07**	**2.87**	**8**	**7**	**148.12**	**2.90**	**3**	**3**	**55.6**	**4.0**
39 13	**12.8**	**10.3**	**241.45**	**2.67**	**11**	**9**	**222.48**	**2.71**	**4**	**5**	**104.9**	**3.4**
39 19	**15.4**	**12.0**	**321.00**	**2.60**	**15**	**11**	**314.29**	**2.64**	**6**	**6**	**160.7**	**3.2**

Chapter 5

ANALYSIS AND COMPARISON IN THE ENERGY-DELAY-AREA DOMAIN

In this chapter, results relative to an extensive comparison in a 65-nm CMOS technology of existing clocked storage elements (CSEs) classes and topologies are reported. In contrast to previous works, the analysis explicitly accounts for effects that arise in nanometer technologies and affect the energy-delay-area tradeoff (e.g., leakage and the impact of layout and interconnects). Compared to previous papers on CSEs comparison, the analysis involves a significantly wider range of CSE classes and topologies. The tradeoffs between leakage, area, clock load, delay and other interesting properties are extensively discussed. The investigation permits to derive several considerations on each CSE class and to identify the best topologies for a targeted application.

5.1 A Thorough Analysis and Comparison Strategy

Various classes of CSEs have been proposed to achieve a desired energy-delay ($E - D$) tradeoff and depending on the features of the application (high speed, low energy, low standby energy, etc.). Understanding the suitability of CSEs for a given application is difficult and so is their selection, since it involves a large number of existing topologies and depends on transistors sizing. In particular, an appropriate sizing methodology is necessary to get reliable results usable in practical designs [ACP10-2].

So far various analyses have been carried out, each focusing on aspects pertinent with FFs comparison [SO99], [PCB01], [HA01], [MNB01], [TNC01], [O03], [OSM03], [NO05], [HKA07], [GNO07]. Among these works, [SO99], [NO05] and [GNO07] are the most thorough in terms of

adopted figures of merit and evaluated parameters. However, previous comparisons exhibit (some of) the following lacks:

- they involve a limited number of topologies and/or do not cover the entire spectrum of applications and $E-D$ constraints that are observed in real designs (therefore, no uniform comparison is available for the wide range of existing CSE classes).
- The area-delay and leakage-delay tradeoffs are usually not considered.
- Circuit designers are typically accustomed to think in terms of minimum energy-delay products ED or ED^2. Instead, a fair comparison should take into account the CSEs behavior over the whole $E-D$ space.
- The CSE input capacitance, C_{IN}, is typically assumed fixed or at most swept as a parameter in a narrow range, whose extent is selected in a naïve manner. Hence, it is not clear if the adopted ranges cover the regions of the $E-D$ space involved in real applications. Moreover, this naïve choice does not permit to associate each value of C_{IN} to a well-defined point in the $E-D$ space.
- Till now, the most significant CSE analyses in the literature have not adopted sub-100 nm technologies, thereby neglecting:

- the leakage influence in active and in standby modes;
- the impact of layout parasitics associated with interconnects, which degrade both speed and energy.

In [ACP10-4], [ACP10-5], [ACP11-1], [ACP11-2], [ACP11-3], a novel analysis and comparison strategy is proposed, which suitably accounts for all the above-mentioned aspects to achieve fair and meaningful results. Such strategy is applied to compare a large number of CSE classes (4) and topologies (19) in a 65-nm CMOS technology. In particular:

a) The comparison is carried out by including local wires parasitics within the transistors sizes optimization by adopting the strategy discussed in Paragraph 4.7.
b) Leakage is separately evaluated and its impact is analyzed in both active and standby mode.
c) The $E-D$ space is explored by considering the points where E^iD^j products are minimized (i and j are widely varied to cover this space). Accordingly, every design is associated with a point in the $E-D$ space that has a clear meaning, which links results to the hardware intensity concept in [ZS02]. This allows for gaining a deeper insight into the $E-D$ tradeoff.
d) According to the motivations in Chapters 2 and 4, C_{IN} is a design variable allowing for further exploring the potentials of each topology in the minimization of different figures of merit, differently from [GNO07], where separate energy-efficient curves (EECs) were extracted under very few (three) different C_{IN} parametrical values.

e) In addition to the thorough investigation of the $E-D$ tradeoff, the interdependence of several other circuit parameters is analyzed, including leakage, silicon area and clock load. To this aim, appropriate figures of merit to rank the considered CSE classes and topologies are introduced.

5.2 Simulation Setup and Energy-Delay Estimation

5.2.1 Test bench circuit

Fig. 5.1 shows the setup used to test a generic CSE, which is similar to that proposed in [OSM03] but with some differences.

The clock signal fed to the CSE comes from a two-stage buffer, sized to attain a typical $FO3$ slope [O03] at the clock input node of the CSE. Hence, the size w_{clock} of the clock-driving inverter close to the CSE is set to get an electrical effort equal to 3 [SSH98]. When evaluating the CSE input capacitance seen from the clock terminal, both the transistors and local (i.e., internal to the CSE) interconnects capacitances are taken into account.

As concerns the CSE data input signal, a different approach is followed with respect to [OSM03], where another constant slope policy was adopted for simplicity. Indeed, in real pipelines, the speed of the logic block driving the CSE data input is obviously comparable to the CSE speed. Accordingly, the size of the data-driving inverter close to the CSE, w_{data}, is set so that the slope of the CSE data input signal (D in Fig. 5.1) is equal to the slope at the output of the CSE first stage that is driven by D. The latter slope is estimated by resorting to the Logical Effort (LE) model. Indeed, during the exploration of the design space, the sizes of all CSE transistors are known and LE model can be applied (also including the layout parasitics).

In the case of circuits that are driven by complementary clocks (e.g., Master-Slave FFs) or data (e.g., Differential CSEs) signals, both polarities are generated through buffers that are considered as external to the CSE, in order to avoid a penalty with respect to other circuits [HA01]. Moreover, it is assumed that the comparison of inverting and non-inverting CSEs does not require further arrangements and that neither of them is presumptively better than the other ones [HA01].

Finally, the output load is swept to test the CSE response under light, moderate and heavy loading conditions [HA01]. Typical reasonable loads are $[4, 16, 64]C_{inv,min}$. Greater loads are not considered since, according to LE, they usually require the insertion of a buffer at the CSE output, which alters the intrinsic energy-delay CSE features [HA01]. Observe that the first loading inverter in Fig. 5.1 that loads the CSE output is in turn loaded by another inverter, which is 4 times wider to avoid an unrealistically strong Miller effect in the gate-drain capacitances at the CSE output.

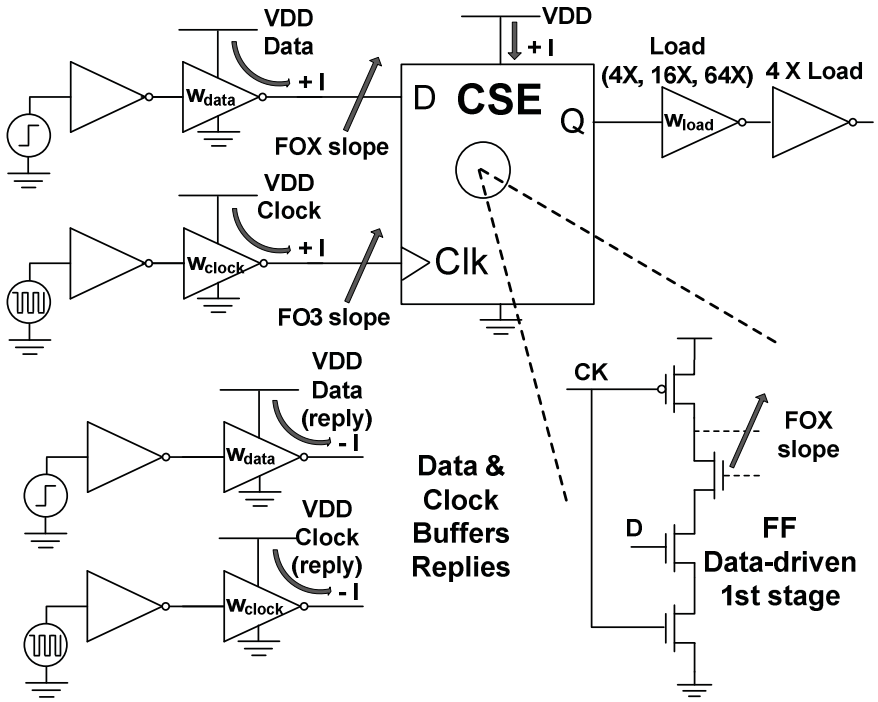


Fig. 5.1. Test bench circuit used to characterize a generic CSE.

5.2.2 Definition of timing figure of merit

The timing parameters characterizing a CSE are well-known and were accurately described in Chapter 3. They are:

1) the minimum data-to-output delay, $\tau_{DQ,min}$, which is obtained by selecting the optimum data-to-clock, $\tau_{DC} = t_{setup}$, delay;
2) the setup time, t_{setup}, which is the optimum τ_{DC} delay that leads to $\tau_{DQ} = \tau_{DQ,min}$;
3) the minimum clock-to-output delay, $\tau_{CQ,min}$, occurring when the data input transition occurs well before the clock transition;
4) the hold time, t_{hold}, which is the clock-to-data delay that leads to a -1 slope in the τ_{CQ} vs. τ_{DC} curve.

In the analysis of the CSE behavior within the $E - D$ space, the speed is quantified through the minimum achievable data-to-output delay, i.e. $D = \tau_{DQ,min}$, according to [GNO07]. Indeed, $\tau_{DQ,min}$ represents the CSE timing contribution to the cycle time when the CSE is placed into a critical path [SO99]. Moreover, every delay is evaluated by considering the greatest among all the possible data-to-output ($D - Q$) paths (namely two for the Single-Edge-Triggered (SET) CSEs and four paths for the Dual Edge-

Triggered (DET) CSEs). On the other hand, CSEs lying in fast paths do not affect system speed. Anyhow, data races must be avoided and the race immunity $R = \tau_{CQ,min} - t_{hold}$ is the parameter that defines the CSE sensitivity to races [OSM03].

As regards t_{setup} and t_{hold}, CSEs can be subdivided into two main categories:

a) CSEs where t_{setup} (t_{hold}) have positive (negative) values, such as the Master-Slave FFs. They always have $R < 0$ and hence are not prone to data race problems, although they do not allow time-borrowing because of their positive t_{setup} [SO99], [OSM03].
b) CSEs where t_{setup} (t_{hold}) have negative (positive) values, such as the Pulsed FFs. They are featured by an inherent tradeoff related to the duration of their transparency window. Indeed, by enlarging the width of the clock pulse, their soft-clock-edge and time-borrowing properties are improved thanks to an increasingly negative t_{setup} [SO99], but their race immunity diminishes because t_{hold} increases [OSM03].

In the first case, t_{setup} and t_{hold} are inherently related to the $\tau_{DQ,min}$ value. Hence, the only independent figure of merit concerning CSE timing is $\tau_{DQ,min}$. In the second case, t_{setup} and t_{hold} can be arranged regardless of the $\tau_{DQ,min}$ value. However, this tradeoff depends only on the specific requirements at the micro-architectural level and can be freely regulated through the pulse width. Hence, also in this case, the only real figure of merit concerning CSE timing is $\tau_{DQ,min}$.

5.2.3 Estimation of energy dissipation

The CSE energy dissipation is made up of transient (i.e., dynamic and short-circuit) and static (i.e., leakage) contributions.

In deep submicron technologies, the impact of leakage has to be considered not only in standby mode but also in active mode, as it is a sizeable fraction of the chip consumption [NC06]. For this reason, it is separately evaluated from the transient contribution, as discussed in the following. An average leakage current, $I_{Leak,avg}$, is estimated by averaging out the 8 possible values for the total CSE leakage current according to the different steady states of the CSE terminals. The generic current is defined I_{xyz} where the subscripts x, y and z stand for the clock, data and output static values (0 or 1). In order to correctly account for the gate leakage, one also needs to:

– add the leakage contribution at the data and clock inputs when they are at the high logical level (because this current is drawn by the CSE under analysis)

– subtract leakage currents that flow from the output when it is logically high (because they are drawn by the load).

Thus, the average static dissipation in a clock cycle is

$$E_{LKG} = \left(\frac{\sum_{x,y,z\,\epsilon\,\{0,1\}} I_{xyz}}{8}\right) V_{DD} T_{CK} = I_{Leak,avg} V_{DD} T_{CK} \qquad (5.1)$$

from which it is apparent that the value of the clock period T_{CK} has to be explicitly set to consistently add and compare the leakage and the transient energy. Equivalently, the impact of leakage in active mode depends on the choices made at the micro-architectural level, which set T_{CK} for a given technology. Actually, to express the impact of micro-architecture regardless of the adopted technology, it is more convenient to refer to the logic depth $T_{CK}/FO4$ instead of the absolute clock cycle (i.e., the equivalent number of cascaded stages with optimum stage effort equal to 4 [SSH98]). Typical values of $T_{CK}/FO4$ are respectively equal to 10, 40 and 80 for high-performance, energy-efficient and low-energy microprocessors, respectively [HJF02], [OK06].

To fairly compare DET and SET CSEs, they are analyzed by assuming the same throughput, i.e. by assuming that the clock cycle of the former is half that of the latter ones.

In regard to the temperature, the analyses are carried out by setting the temperature to realistic values encountered in real applications (i.e., in the order of $70°\,C$) instead of the room temperature [RCN03], [WH04]. This setting affects the speed of the circuits and the strongly temperature-dependent leakage currents in a realistic way.

Transient energy depends on the data input switching activity α_{sw} [HKA07], which is set to the typical values 0.10, 0.25 and 0.5. The evaluation of transient dissipation has been discussed in detail in [OSM03]. However, some modifications are required to more properly evaluate this contribution [ACP11-1]. Details about our estimation methodology are reported in the Appendix, where the average transient energy E_{TRAN} in a clock cycle is evaluated as a function of α_{sw}.

The proper evaluation of the energy dissipation related with the input, internal and output nodes transitions is a somewhat complicated task. In [SO99], [OSM03] some significant guidelines were outlined. For instance, the evaluation of the energy consumption must not include the energy dissipated in the charging/discharging of the external output load, since it is a value solely depending on the load dimension and not on the CSE features.

However, one should not simply subtract the magnitude of the current flowing from and towards the load, otherwise, the effect of some undesired output transitions would not be taken into account. To be more specific, some topologies (e.g. some semi-dynamic FFs) can suffer from glitches both

on the internal and output nodes. In this case, the energy dissipation due to output glitches must be included, since it is a shortcoming that worsens the CSE features right dependently on the load value.

The energy spent to charge the data and clock inputs has to be included in the computation [SO99], [OSM03], because it is a feature dependent on the CSE characteristics. The replicas of the data- and clock-driving buffers are inserted in the simulation setup (see Fig. 5.1), to subtract the energy due to the parasitic load of the same driving inverters [SO99], [OSM03].

Summarizing, defining I_{FF}, I_{clk}, I_{data}, $I_{clk,re}$ and $I_{data,re}$ as the currents drawn from the power supply by the CSE, by the data- and clock-driving inverters close to the CSE and by their unloaded replicas, the generic contribution to the transient energy (at the moment including the energy on the load) is

$$V_{DD} \int_{T_A}^{T_B} \left(I_{FF} + I_{clk} + I_{data} - I_{clk,re} - I_{data,re}\right) dt \tag{5.2}$$

where the definition of the integration limits, T_A and T_B, has to be properly done.

In [OSM03], the authors deal with the energy breakdown by referring to four energy contributions: $E_{0\to0}$, $E_{1\to1}$, $E_{1\to0}$ and $E_{0\to1}$. They are evaluated by considering a single clock period during which a single event on data occurs ($0 \to 0$, $1 \to 1$, $1 \to 0$ or $0 \to 1$). The authors state that it is possible to infer clocking, precharge and internal nodes energy contributions by simply combining the four terms according to transition probabilities and subtracting the energy spent on load.

However, the simple approach shown in [OSM03] does not allow to accurately separate and localize the various sources of energy consumption (clock, precharge, etc.), because the energy dissipation related with the transition of one signal is influenced by the values of the other signals. For instance, according to the CSE functionality, the transition of the data input can cause simply the charging of the input gate capacitance or the transition of internal nodes according to the state of the clock (e.g. in Master-Slave FFs). If, after the transition, the data always remained stable waiting for being transferred through the CSE, the argument made in [OSM03] would be completely correct and exhaustive. But, actually, the data can change during the opaque phase of the CSE and one needs to account for all the possible transition scenarios, in order to have the most general information about the transient energy.

Moreover, the integration of the supply current over the entire clock period includes the static energy due to leakage, whereas it should be separately evaluated and weighted according to the chosen logic depth $T_{CK}/FO4$.

For these reasons it is suggested to consider all the possible 12 transitions that arise according to all possible inputs combinations [ACP11-1]. The following notation is adopted: the first subscript refers to the input signal (clock or data) that is varying and to the specific transition ($1 \rightarrow 0$ or $0 \rightarrow 1$), the second subscript refers to the value of the other stable input signal ($= 0$ or $= 1$) and the third subscript is related with the output behavior, which can remain stable ($= 0$ or $= 1$) or can vary ($1 \rightarrow 0$ or $0 \rightarrow 1$). In the following, with no loss of generality, we will refer to a non-inverting Positive SET CSE. The first four contributions are those related with the clock transitions when the input and output are stable:

$$E_1 = E_{CK\ 1\rightarrow0\ ,D=0\ ,Q=0} \tag{5.3}$$
$$E_2 = E_{CK\ 0\rightarrow1\ ,D=0\ ,Q=0} \tag{5.4}$$
$$E_3 = E_{CK\ 1\rightarrow0\ ,D=1\ ,Q=1} \tag{5.5}$$
$$E_4 = E_{CK\ 0\rightarrow1\ ,D=1\ ,Q=1} \tag{5.6}$$

The second four contributions are related with the clock transitions when the input and output are different. In the case of a Positive SET CSE, this does (not) lead to an output change for the clock $0 \rightarrow 1$ ($1 \rightarrow 0$) transition (the situation is reversed for a Negative SET CSE):

$$E_5 = E_{CK\ 1\rightarrow0\ ,D=0\ ,Q=1} \tag{5.7}$$
$$E_6 = E_{CK\ 1\rightarrow0\ ,D=1\ ,Q=0} \tag{5.8}$$
$$E_7 = E_{CK\ 0\rightarrow1\ ,D=0\ ,Q\ 1\rightarrow0} \tag{5.9}$$
$$E_8 = E_{CK\ 0\rightarrow1\ ,D=1\ ,Q\ 0\rightarrow1} \tag{5.10}$$

Finally come the data input transitions, which can occur during the high or low clock phase:

$$E_9 = E_{D\ 1\rightarrow0\ ,CK=0\ ,Q=1} \tag{5.11}$$
$$E_{10} = E_{D\ 1\rightarrow0\ ,CK=1\ ,Q=1} \tag{5.12}$$
$$E_{11} = E_{D\ 0\rightarrow1\ ,CK=0\ ,Q=0} \tag{5.13}$$
$$E_{12} = E_{D\ 0\rightarrow1\ ,CK=1\ ,Q=0} \tag{5.14}$$

These 12 contributions are evaluated by integrating the supply current according to (5.2), assuming T_A to be the point of time where the input experiences a transition, and T_B to be the point of time where the slowest node within the CSE reaches 99% of its steady value[1]. In this way, the time

[1] Sometimes, the nodes voltages can take long times to reach the 99% of the steady value. Anyhow, when not employing simple pass-transistors that cause a threshold drop and when all transistors are properly sized according to the architectural

window $[T_A, T_B]$ is sufficiently wide to fully capture the dynamic and short-circuit energy contributions, whereas it is sufficiently narrow to neglect and the impact of leakage.

To determine the average transient energy in a clock cycle, the switching activity α_{sw} needs to be used. If at most one data transition occurs for each clock period (i.e. $\alpha_{sw} \leq 1$), the average transient energy can be written as

$$E_{TRAN} = \frac{(1-\alpha_{sw})}{2}(E_1 + E_2 + E_3 + E_4) + \frac{\alpha_{sw}}{2}(E_5 + E_6 + E_7 + E_8) + \frac{\alpha_{sw}}{4}(E_9 + E_{10} + E_{11} + E_{12}) - \frac{\alpha_{sw}}{2} C_L {V_{DD}}^2 \quad (5.15)$$

The quantity $\frac{\alpha_{sw}}{2} C_L {V_{DD}}^2$ in (5.15) is the energy required by the load capacitance, and is hence is subtracted because it depends only on the adopted load (i.e., it is not a feature of the considered CSE).

Some further arrangements are required when dealing with DET CSEs. In this case (5.7) and (5.8) change into

$$E_5 = E_{CK\ 1\to0\ ,D=0\ ,Q\ 1\to0} \quad (5.16)$$

$$E_6 = E_{CK\ 1\to0\ ,D=1\ ,Q\ 0\to1} \quad (5.17)$$

because both the clock transitions enable the data-transfer through the CSE. To fairly compare SET and DET CSEs, the same throughput must be assumed, which translates into an halved clock frequency for DET with respect to SET CSEs. Therefore, in order to consistently readjust the average transient energy evaluation, one has to refer to the transitions occurring in a half-cycle and hence (5.15) is changed into

$$E_{TRAN} = \frac{(1-\alpha_{sw})}{4}(E_1 + E_2 + E_3 + E_4) + \frac{\alpha_{sw}}{4}(E_5 + E_6 + E_7 + E_8) + \frac{\alpha_{sw}}{4}(E_9 + E_{10} + E_{11} + E_{12}) - \frac{\alpha_{sw}}{2} C_L {V_{DD}}^2 \quad (5.18)$$

It is worth noting that the proposed methodology also allows for straightforwardly taking data glitches into account. This was not possible in [OSM03], since contributions in (5.11)-(5.14) were not explicitly evaluated.

Finally, the average CSE energy dissipation in one clock cycle, E_{CSE}, is the sum of E_{TRAN} and E_{LKG} and thus depends on the input data statistics and micro-architectural choices through switching activity α_{sw} and logic depth $T_{CK}/FO4$.

$T_{CK}/FO4$ specification [ACP10-2], the 99% value can be closely approached in practically acceptable times. Nevertheless, a good estimation of transient energy comes out also considering slightly smaller values than 99% (e.g. 90%), and hence it is simply a matter of convention when characterizing a CSE.

5.3 Analyzed CSE Classes and Topologies

To cover the wide spectrum of adopted CSE topologies, in the following analysis the main four CSE classes are considered: Master-Slave (MS), Implicit-Explicit Pulsed (IP and EP), Differential and DET topologies. 19 CSE circuits, which are among the most representative and best known ones, are chosen for the four classes.

In particular, the considered MS topologies are (the latter two ones are also clock-gated structures):

- Transmission Gate FF (TGFF) [MNB01];
- Write-Port Master-Slave FF (WPMS) [MTD03];
- Gated Master-Slave FF (GMSL) [MNB01];
- Data Transition Look-Ahead FF (DTLA) [NO98].

The analyzed Pulsed topologies are seven (the first five ones are IP, whereas the remaining two are EP):

- Hybrid Latch FF (HLFF) [PBS96];
- Semi-Dynamic FF (SDFF) [KAD99];
- UltraSPARC Semi-Dynamic FF (USDFF) [HAA00];
- Implicitly Push-Pull FF (IPPFF) [N03];
- Conditional Precharge FF (CPFF) [NAO01];
- Static Explicit Pulsed FF (SEPFF) [ZDB02];
- Transmission Gate Pulsed Latch (TGPL) [NCF02].

The four Differential CSEs investigated are:

- Modified Sense-Amplifier FF (MSAFF) [NSO00];
- Skew-Tolerant FF (STFF) [NOW03];
- Conditional Capture FF (CCFF) [KKJ01];
- Variable Sampling Window FF (VSWFF) [SK05].

Finally, the four DET topologies are (the first is a MS, the second is IP and the other ones are EP):

- Transmission Gate Latch-Mux (DET-TGLM) [LS96];
- Symmetric Pulse Generator FF (DET-SPGFF) [NWO02];
- Static Pulsed Latch (DET-SPL) [TNC01];
- Conditional Discharge FF (DET-CDFF) [ZDB04].

In Fig. 5.2a-s the schematics of each CSE and the location of the independent design variables w_k lying in the $D-Q$ paths and that have to be optimized as explained in Chapter 4, are depicted. Note that, in the analysis of EP CSEs, one pulse generator (PG) for each latch is considered. Obviously, this is a somewhat conservative choice in the estimation of the energy consumption of such FFs, since, by sharing the PG among a few different latches [TNC01], [NCF02], [ZDB02], [WH04], the fraction of PG

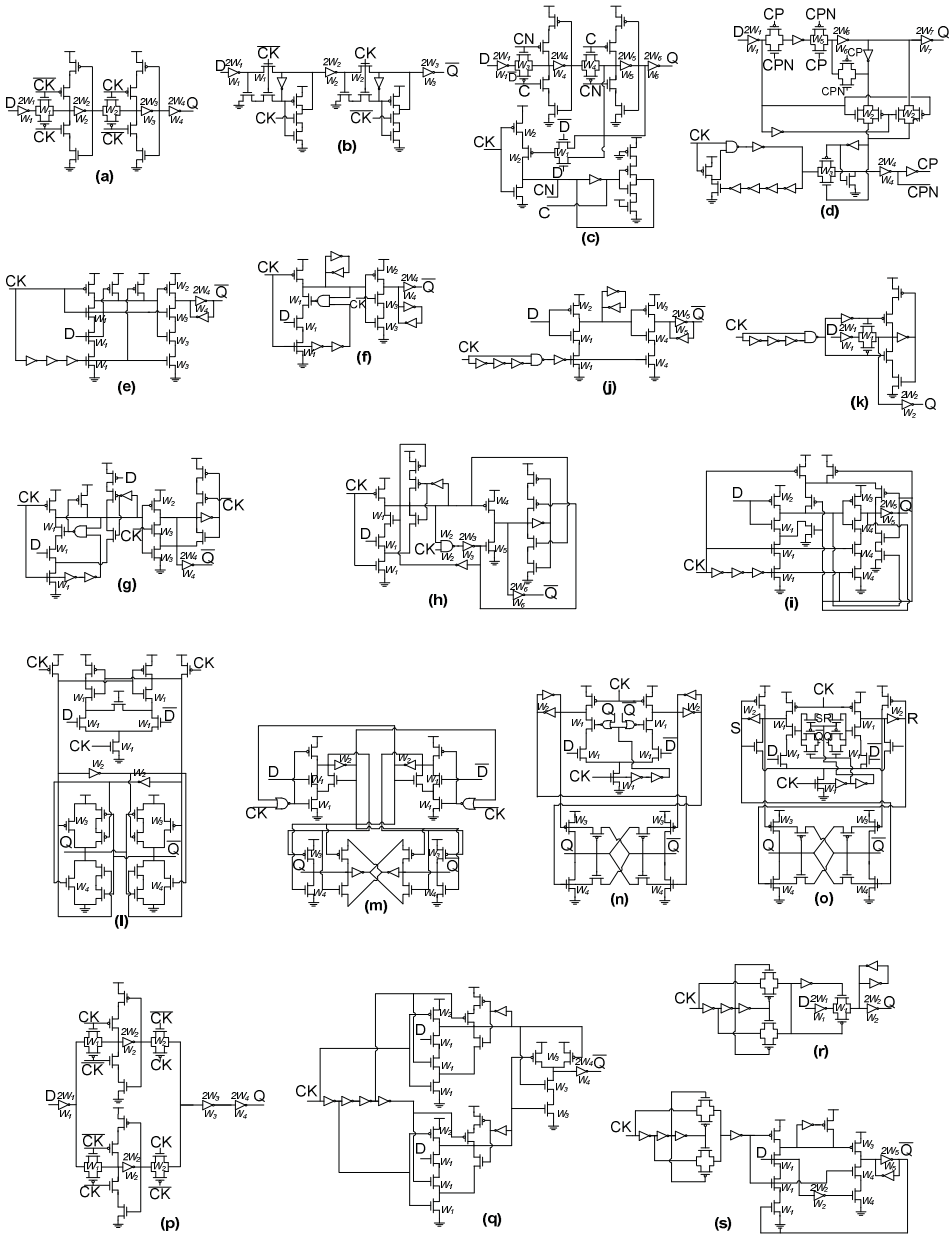


Fig. 5.2. Schematics of the analyzed CSEs and variable widths w_k to be optimized: TGFF (a), WPMS (b), GMSL (c), DTLA (d), HLFF (e), SDFF (f), USDFF (g), IPPFF (h), CPFF (i), SEPFF (j), TGPL (k), MSAFF (l), STFF (m), CCFF (n), VSWFF (o), DET-TGLM (p), DET-SPGFF (q), DET-SPL (r), DET-CDFF (s).

dissipation imputable to each latch slightly diminishes. This must be taken into account when comparing EP CSEs with other topologies.

As concerns the optimized figures of merit (FOMs) in the $E-D$ space, the following products are chosen (E_0 is the energy of the minimum sizing leading to a correct operation)

$$[ED^5 \,/\, ED^4 \,/\, ED^3 \,/\, ED^2 \,/\, ED \,/\, E^2D \,/\, E^3D \,/\, E_0] \tag{5.19}$$

The energy-efficient transistor level design methodologies in Chapter 4 are extensively employed and, as was done in the example in Paragraph 4.6, the consistency of the methodology assumptions is verified by comparing the energy-to-delay sensitivity in the minimum E^iD^j points with the theoretical values $-j/i$. The simulated and theoretical sensitivities are plotted in Fig. 5.3 for all the optimum designs found within this analysis under various conditions and for all the considered CSEs. Detailed numerical values of the mean, standard deviation, maximum and minimum sensitivity for each point of the energy-efficient curve (EEC) are reported in Tab. V.I. From Fig. 5.3 and Tab. V.I, it is apparent that the dispersion of the simulated values is very small and the resulting values agree very well with theoretical sensitivity, thereby confirming again the validity of the assumptions introduced in Chapter 4 (including the convexity of the functionals E^iD^j).

In order to have an idea of the layout complexity and the resulting impact of local wires, the layouts of the CSEs are shown in Fig. 5.4a-s for the minimum-ED sizings under typical values $C_L = 16C_{inv,min}$, $\alpha_{sw} = 0.25$ and $T_{CK}/FO4 = 40$.

5.4 Normalization to Technology

To gain an intuitive understanding of results independently of technology, the various quantities and data are properly normalized to reference technology values. In particular:

- capacitances are normalized to that of a symmetrical minimum inverter ($W_P = 2W_N = 2W_{min}$), $C_{inv,min} = 410\text{aF}$;
- delays are normalized to $FO4 = 18.3\text{ps}$ delay [HHW97], [WH04];
- energies are normalized to $E_{min} = 0.202\text{fJ}$ (see Chapter 4);
- leakage currents are normalized to the average leakage current of a symmetrical minimum inverter, $I_{Leak,min} = 35.4\text{nA}$;
- areas are normalized to $\chi^2 = 0.04\mu m^2$, where $\chi = 200\text{nm}$ is the minimum pitch of the Metal2 layer.

For all the analyses, a $VDD = 1$ V supply voltage is adopted.

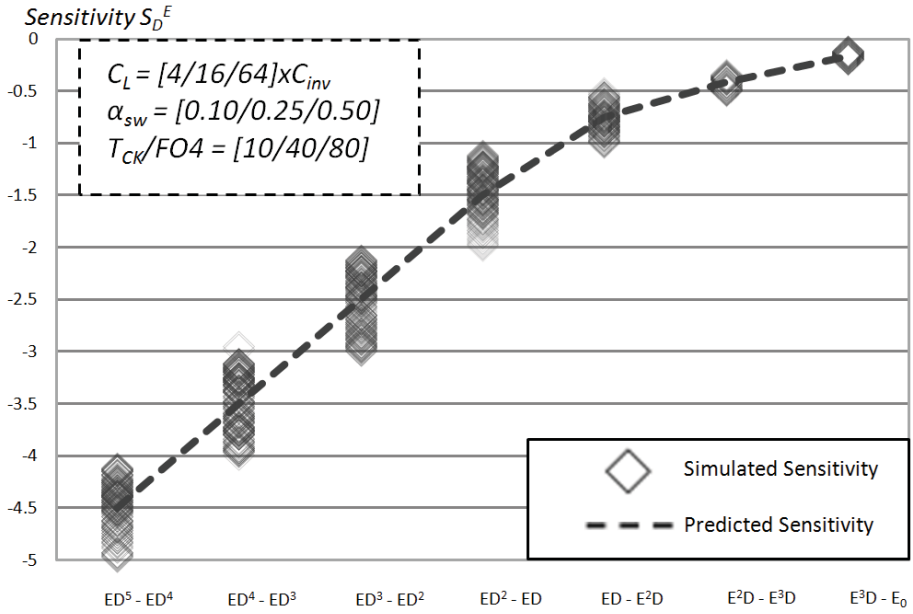
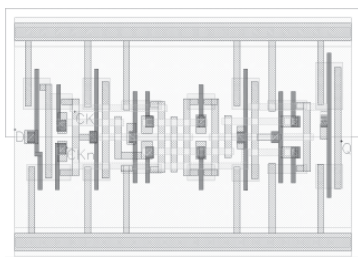


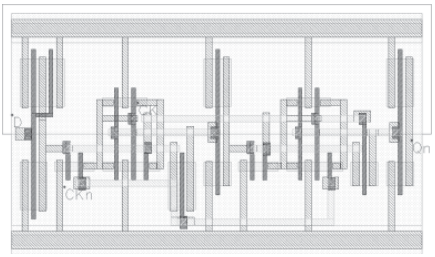
Fig. 5.3. Sensitivity analyses for the optimum designs.

TABLE V.I: ANALYSIS OF THE SENSITIVITY S_D^E

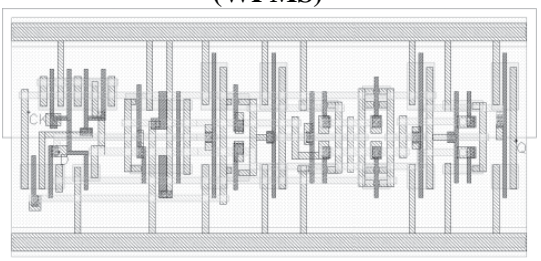
Sensitivity S_D^E					
FOMs	***Theoretical***	***Average***	***Standard Deviation***	***Maximum***	***Minimum***
ED^5 / ED^4	-4.500	-4.354	0.232	-3.986	-4.931
ED^4 / ED^3	-3.500	-3.423	0.291	-2.949	-3.990
ED^3 / ED^2	-2.500	-2.512	0.269	-2.115	-2.995
ED^2 / ED	-1.500	-1.434	0.206	-1.112	-1.983
ED / E^2D	-0.750	-0.760	0.119	-0.513	-0.999
E^2D / E^3D	-0.416	-0.420	0.005	-0.337	-0.500
E^3D / E_0	-0.166	-0.177	0.045	-0.111	-0.329



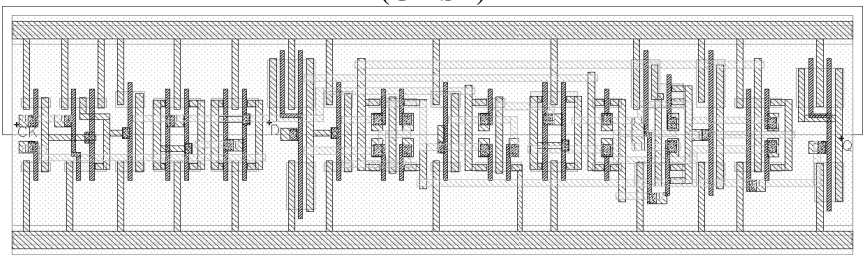
(TGFF)



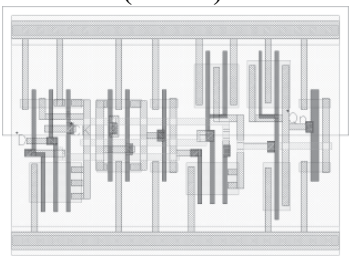
(WPMS)



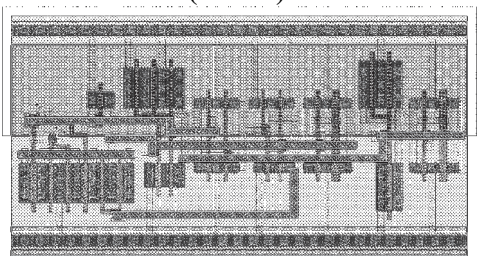
(GMSL)



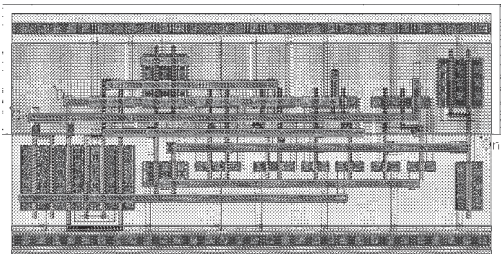
(DTLA)



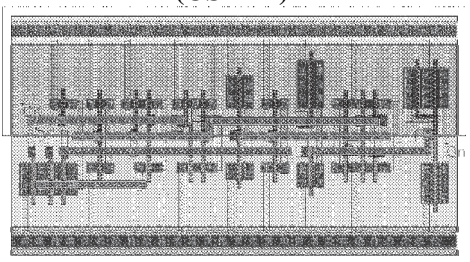
(HLFF)



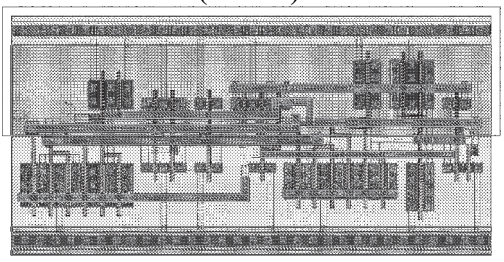
(SDFF)



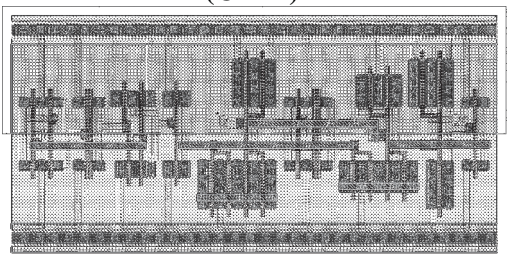
(USDFF)



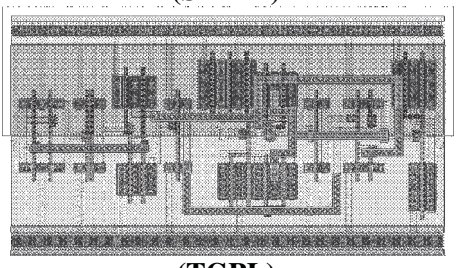
(IPPFF)



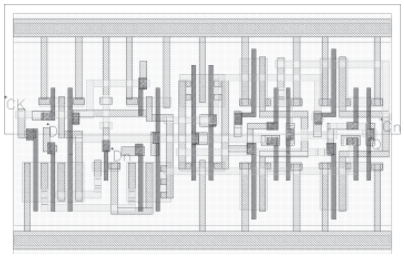
(CPFF)



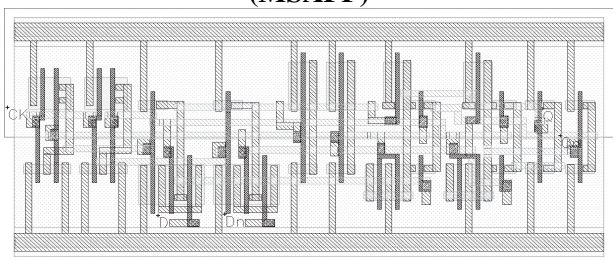
(SEPFF)



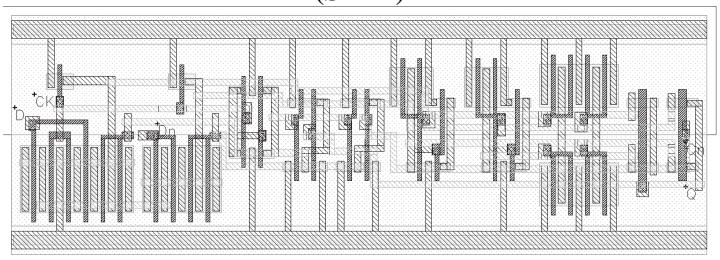
(TGPL)



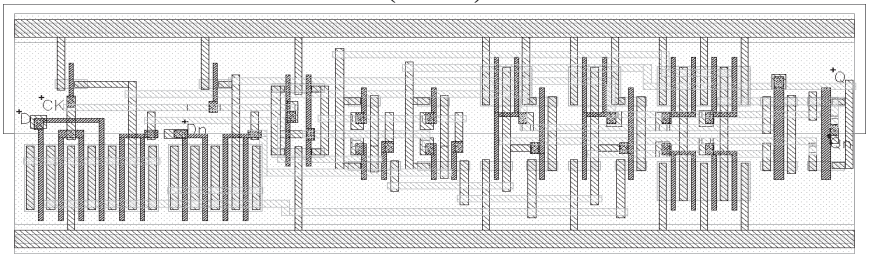
(MSAFF)



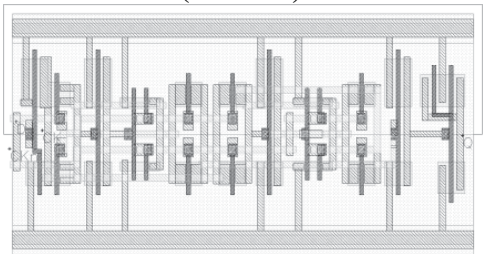
(STFF)



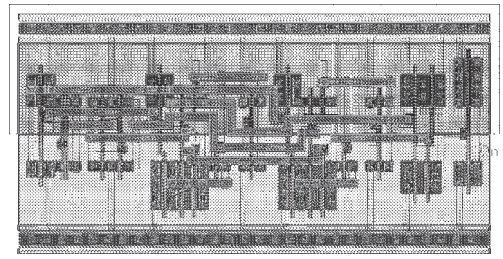
(CCFF)



(VSWFF)

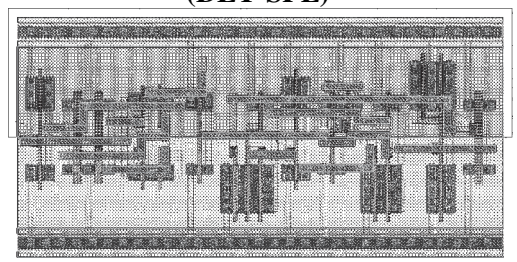


(DET-TGLM)



(DET-SPGFF)

(DET-SPL)



(DET-CDFF)

Fig. 5.4. Layouts of the analyzed CSEs (min. ED sizing).

5.5 Energy-Delay Tradeoff in Each Class

In this paragraph the tradeoff between the energy E_{CSE} and the delay $\tau_{DQ,min}$ is discussed by comparing the EECs for various CSE classes. The curves were extracted under different C_L and α_{sw} conditions and, since the ranking of topologies does not change significantly with the logic depth, the results for $T_{CK}/FO4 = 40$ are presented. In the following, a load capacitance $C_L = 16C_{inv,min}$ and switching activity $\alpha_{sw} = 0.25$ is assumed as the "reference case".

5.5.1 Single-Edge Triggered Master-Slave CSEs

The EECs of the SET MS CSEs, derived in the reference case, are reported in Fig. 5.5. Unlike the results in [GNO07], where the performances of TGFF and WPMS are similar in the high-speed and minimum ED region,

from Fig. 5.5 it is found that TGFF is more energy-efficient than WPMS in all $E-D$ regions (WPMS has minimum delay D_0, energy-delay product ED and minimum energy E_0 worse than the TGFF by a factor close to 1.5, and other E^iD^j FOMs are even worse).

This is partly due to the adoption of NMOS pass-transistors (vs. TGFF transmission gates) and partially non-gated keepers (vs. TGFF full-gated keepers), but also to the impact of the longer internal wires needed (WPMS area is $1.3-1.5$X greater than TGFF area in the various considered sizings).

Clock-gated CSEs (GMSL and DTLA) exhibit the worse performances throughout the $E-D$ space. Their high latency is obviously due to the high number of stages involved in the $D-Q$ paths, because of the additional gating logic with respect to the basic MS topologies. In regard to energy consumption, in principle clock-gated CSEs should have a low dissipation for low switching activity α_{sw} [NO98], [OSM03], [ACP11-1]. Actually, this holds given that GMSL (DTLA) nearly achieves 700% (200%) clock-related energy savings when working in gating (i.e., when $D=Q$) rather than in non-gating (i.e., when $D\neq Q$) condition. However, from an absolute point of view (i.e., when comparing to other CSEs), the E_0 of GMSL and DTLA are about 1.2X (1.8X) and 3.0X (3.0X) times greater than TGFF for $\alpha_{sw}=0.1$ (0.25). Again, this is due to the strong impact of layout parasitics that degrade the performances of clock gated CSEs, since they have a very complex layout (see layouts in Fig. 5.4).

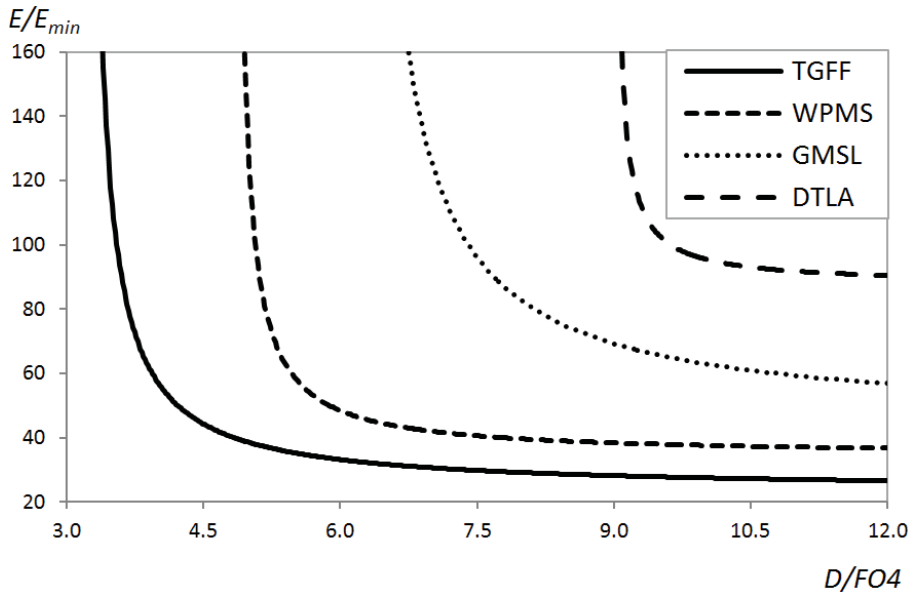


Fig. 5.5. EECs of MS CSEs: $C_L=16C_{inv,min}$, $\alpha_{sw}=0.25$, $T_{CK}/FO4=40$.

The ranking of the analyzed MS CSEs does not change for different α_{sw} and C_L values, and the TGFF largely remains the most energy-efficient MS CSE in all the $E - D$ space. For this reason, additional figures relative to this CSE class are not reported for the sake of brevity.

5.5.2 Single-Edge Triggered Implicitly-Explicitly Pulsed CSEs

The EECs of the SET IP-EP CSEs, derived in the reference case, are reported in Fig. 5.6. From this figure, the TGPL is clearly the most energy-efficient SET Pulsed CSE in the high-speed region and in the low-energy one up to the E^2D FOM. This was expected from the simplicity of the basic latch structure of TGPL (and hence the low impact of layout parasitics) [PCB01]. This good energy efficiency of TGPL is remarkable since here every latch is considered with its own PG, but actually energy may be further reduced by sharing PG among various latches. From Fig. 5.6, in the deep low-energy region (minimum E^3D and E_0 FOMs), the CPFF and IPPFF are the best SET Pulsed CSEs. Indeed, both are IP and hence do not require a PG. In addition, the CPFF employs a conditional technique to avoid unnecessary precharge [NAO01], while the IPPFF reduces the load on the precharged internal node by using a push-pull second stage. CPFF and HLFF also exhibit the best speed among SET IP CSEs.

SEPFF is fast, but dissipates more than TGPL in all conditions and hence

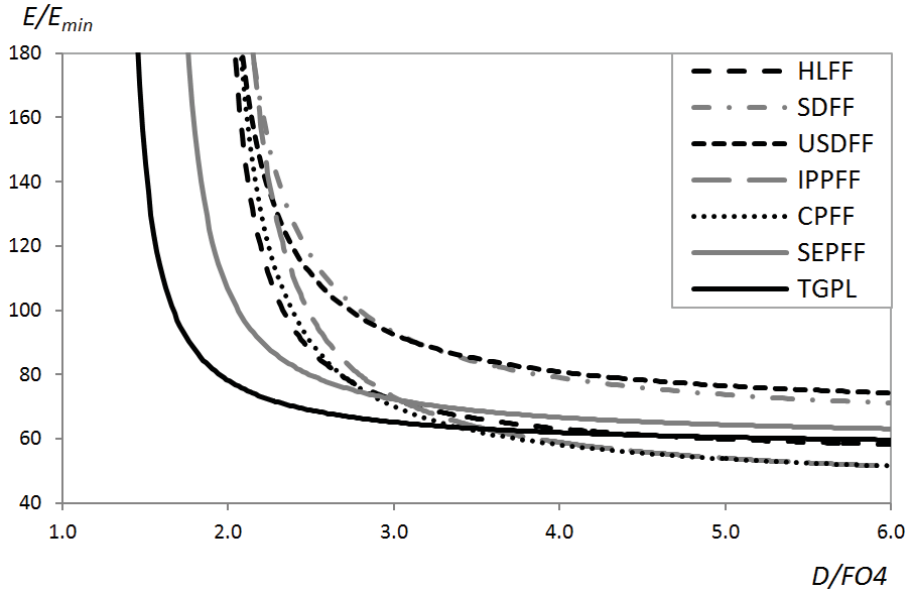


Fig. 5.6. EECs of IP-EP CSEs: $C_L = 16C_{inv,min}$, $\alpha_{sw} = 0.25$, $T_{CK}/FO4 = 40$.

is less energy-efficient. Its delay is also nearly 1.2X greater than TGPL in the various conditions. This is somewhat different from previous works [TNC01], which predicted the same speed for an average load (like $16C_{inv,min}$). Again, this is due to the heavier parasitic delay associated with interconnects, since SEPFF apparently has a slightly more complex layout compared to TGPL (see layouts in Fig. 5.4).

Among all the SET Pulsed CSEs, the semi-dynamic ones (SDFF and USDFF) exhibit the worst speed in the whole $E-D$ space. The reason is again related with the layout complexity, as is apparent from the comparison of the layouts in Fig. 5.4. In contrast with [KAD99], [SO99], [TNC01], where it is stated that such FFs have $E-D$ features very similar to the HLFF, it is found that the latter one is significantly more energy-efficient throughout the whole $E-D$ space (except in the very high-speed region where they are similar). Indeed, HLFF has a much simpler schematic and hence its layout has much shorter interconnects, thus reducing energy consumption. Moreover, in contrast to previous results [GNO07], USDFF does not outperform SDFF, again because of its more complex routing, although this can be only partly inferred from the inspection of the layouts in Fig. 5.4, which are relative to a single sizing strategy. Given the mirror-like structure of the two circuits, the local wires capacitances can be compared by averaging out the results for all the different nodes and for all the different sizing strategies considered in this work. On the average, it is found that local wires parasitics are nearly 60% larger for USDFF than SDFF.

All SET IP CSEs are slower than EP CSEs. In particular, by averaging out the delays correspondent to the various optimized FOMs, IP CSEs delays are nearly 1.3X greater than for EP CSEs. This happens because IP CSEs need stages with three stacked transistors in their critical path, whereas EP CSEs exploit a real pulsed signal and need stages with two stacked transistors. In particular, IPPFF has the worst minimum delay D_0 among IP CSEs, since it exhibits three and four stages paths for the rising and falling data transitions and this effect overcomes the advantages given by the push-pull stage [GNO07].

To understand the dependence of the above results on the load, the EECs of Pulsed CSEs for $C_L = 64C_{inv,min}$ and $C_L = 4C_{inv,min}$ are reported in Fig. 5.7**Errore. L'origine riferimento non è stata trovata.**a-b (in both cases $\alpha_{sw} = 0.25$ and $T_{CK}/FO4 = 40$). The ranking of IP CSEs does not change significantly, except for IPPFF that, having a greater number of stages in its $D-Q$ paths, becomes relatively faster for a large load, as is obvious from LE. As concerns EP CSEs, unlike [HA01], where the speed of a two stage CSE (TGPL) is overcome by that of a three stage topology (SEPFF) when the load is large enough ($64C_{inv,min}$), the SEPFF still shows a 1.1x (1.3x) delay increment even for $C_L = 64C_{inv,min}$ ($4C_{inv,min}$). When the load is

small ($4C_{inv,min}$), TGPL is the most energy-efficient Pulsed CSE up to E^3D, whereas it is dominant "only" up to ED for large load ($64C_{inv,min}$).

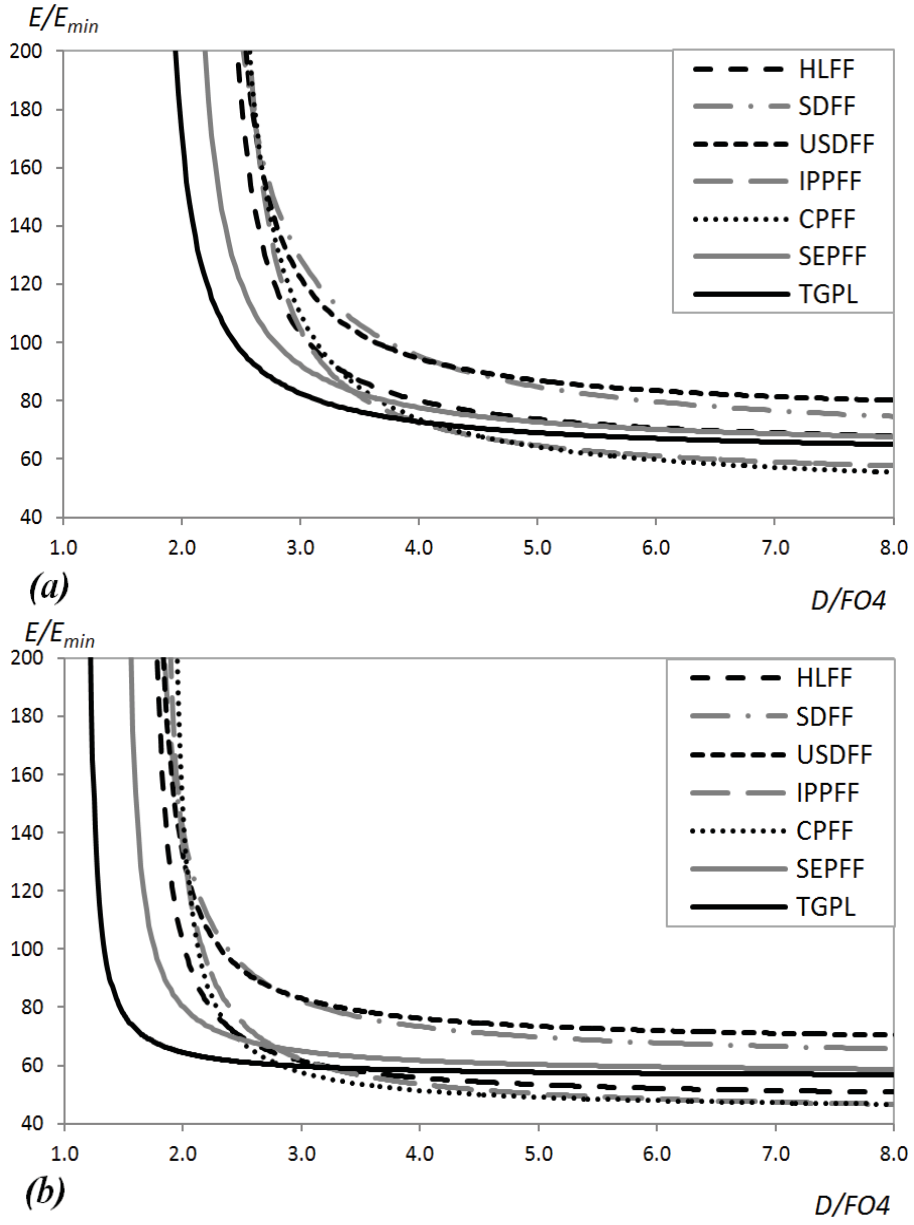


Fig. 5.7. EECs of IP-EP CSEs: $C_L = 64C_{inv,min}$ (a) and $C_L = 4C_{inv,min}$ (b) ($\alpha_{sw} = 0.25, T_{CK}/FO4 = 40$).

To understand the effect of switching activity on Pulsed CSEs, their EECs for $\alpha_{sw} = 0.1$ and a $\alpha_{sw} = 0.5$ are reported in Fig. 5.8a-b (in both cases $C_L = 16C_{inv,min}$ and $T_{CK}/FO4 = 40$). The main changes occur in the low-energy region, where the CPFF becomes more energy efficient for

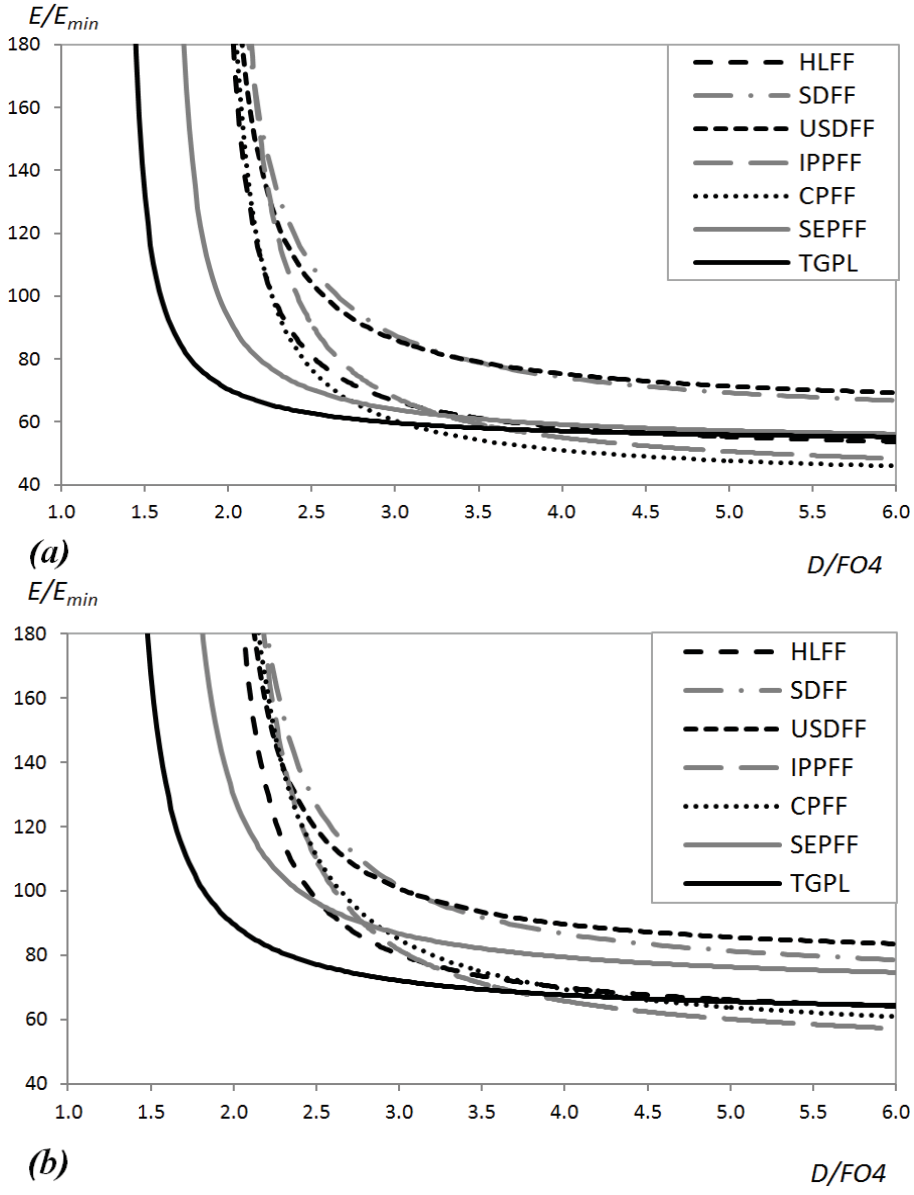


Fig. 5.8. EECs of IP-EP CSEs: $\alpha_{sw} = 0.1$ (a) and $\alpha_{sw} = 0.5$ (b) ($C_L = 16C_{inv,min}$, $T_{CK}/FO4 = 40$).

$\alpha_{sw} = 0.1$ from E^2D on, since it takes advantage of conditional precharge. Conversely, for $\alpha_{sw} = 0.5$, the IPPFF becomes the more energy-efficient Pulsed CSE in the low-energy region, whereas the CPFF and the SEPFF (both exhibiting pseudo-static first stages) suffer from a considerable increase in their dissipation due to the high data activity rate.

5.5.3 Single-Edge Triggered Differential CSEs

The EECs of the SET Differential CSEs in the reference case are reported in Fig. 5.9. From this figure, the $E - D$ space is split in two regions: the high-speed one (from D_0 to ED^2 FOMs), where the STFF is the most energy-efficient, and the low-energy one (from ED to E_0 FOMs), where the MSAFF is the best Differential CSE. In particular, STFF is the fastest among all the analyzed CSEs. For instance the D_0 of TGPL is 1.1X greater than the STFF, whereas those of MSAFF, CCFF and VSWFF are 1.8X, 1.3X and 1.4X greater, respectively.

These differences in the speed of such Differential CSEs can be explained as follows: all of them have equal second (skewed inverter) and third (push-pull) stages, which are very fast. As regards the first stage, the speed of MSAFF is affected by the load imposed by the cross-coupled inverters, whose NMOS transistors belong to the complementary critical paths (although the sense-amplifier nature is useful for level-restoring). The first

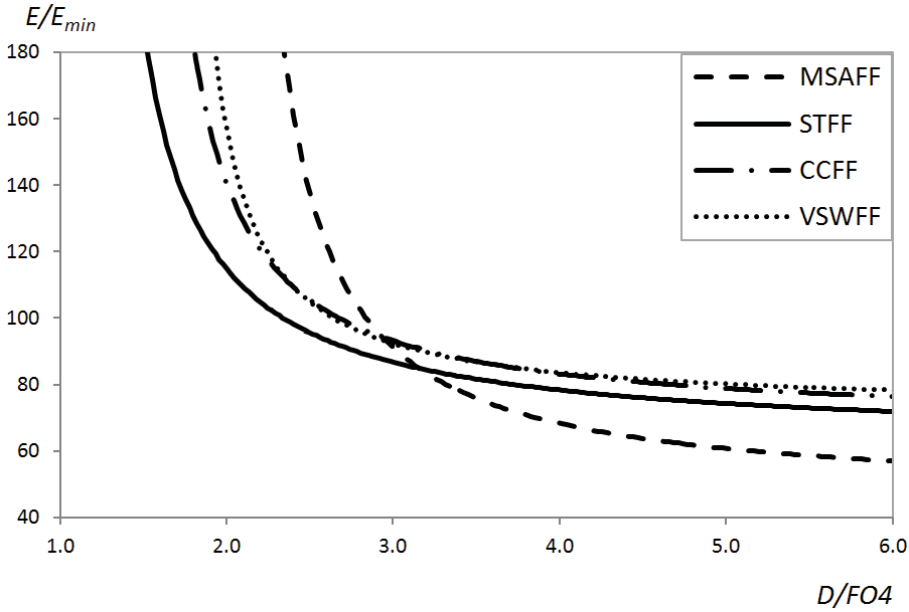


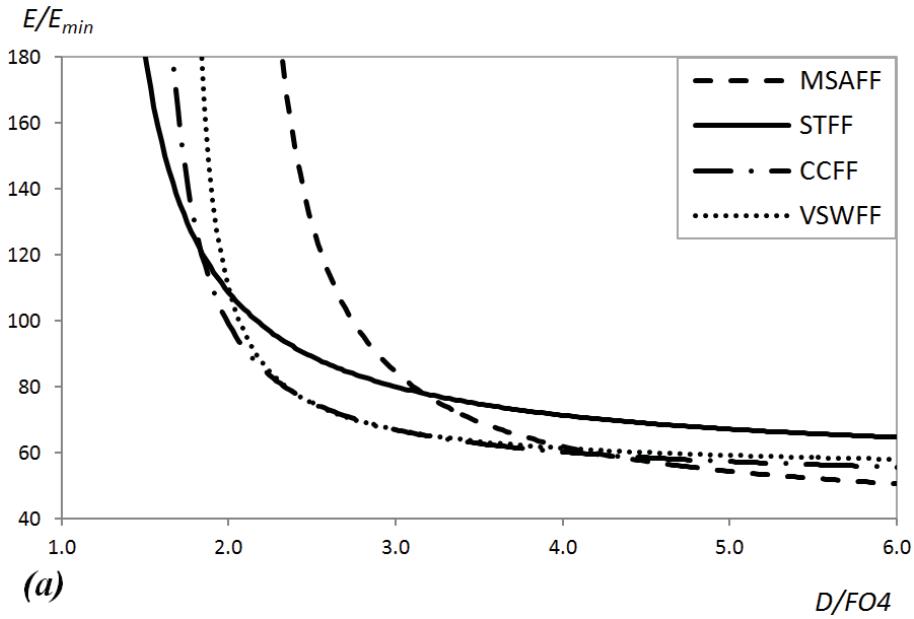
Fig. 5.9. EECs of Differential CSEs: $C_L = 16C_{inv,min}$, $\alpha_{sw} = 0.25$, $T_{CK}/FO4 = 40$.

stage of CCFF and VSWFF does not have this drawback and is significantly faster, but not as much as the first stage of STFF, where only two stacked NMOS are employed thanks to the use of additional driving NOR gates.

The high energy-efficiency of MSAFF in the low-energy region is due to its relatively simpler layout and to the lower impact of layout parasitics that allows for downsizing transistors with minor performances loss with respect to STFF, CCFF and VSWFF. As shown in the layout in Fig. 5.4, the high regularity of MSAFF layout leads to a very small area despite of its differential signaling.

For analogous reasons, CCFF and VSWFF, which have extremely complex layout and local wires (see the layouts in Fig. 5.4), are never the most energy-efficient. This is in contrast to what is claimed in many papers (especially as concerns CCFF) [NAO01], [KKJ01], [OSM03], [SK05], where the conditional capture property is praised as a very efficient technique to reduce energy at a negligible speed penalty. This is no longer true in nanometer technologies where the impact of local wires is considerable (in order to maintain good speed, CCFF and VSWFF need to be strongly oversized for a targeted speed).

Given the very similar topology of the considered Differential CSEs, the same ranking is obtained regardless of the load C_L. Instead, switching activity has a significant impact on the comparison, as is shown in Fig. 5.10a-b where the EECs derived for $\alpha_{sw} = 0.1$ and a $\alpha_{sw} = 0.5$ are plotted (in both cases $C_L = 16C_{inv,min}$ and $T_{CK}/FO4 = 40$). In detail, for $\alpha_{sw} = 0.1$, CCFF and VSWFF become the most energy-efficient from ED^2 to E^2D



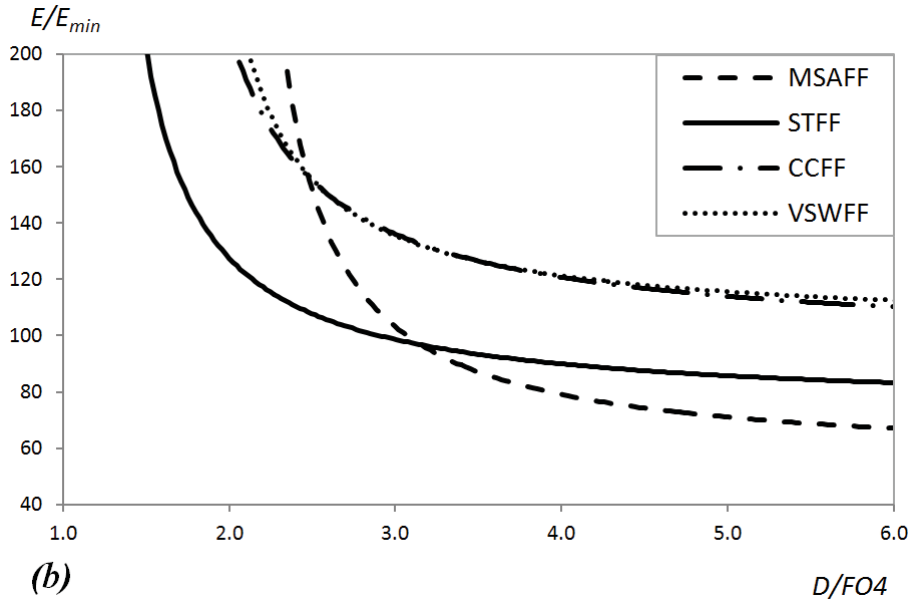


Fig. 5.10. EECs of Differential CSEs: $\alpha_{sw} = 0.1$ (a) and $\alpha_{sw} = 0.5$ (b) ($C_L = 16C_{inv,min}, T_{CK}/FO4 = 40$).

FOMs (thus including also the product ED). For $\alpha_{sw} = 0.5$ their EECs move far away from the MSAFF and STFF ones, in contrast to [KKJ01], where it is stated that conditional capture CSEs have a reasonable energy consumption even for such a data transition rate.

5.5.4 Dual-Edge Triggered CSEs

It is decided to put the selected DET topologies in a single class even if they have quite different basic operations and features. The EECs of the DET CSEs, derived in the reference case, are reported in Fig. 5.11. As for the Differential class, two topologies emerge as the most energy-efficient ones: DET-SPL in the high-speed region (from D_0 to ED^2 FOMs) and the DET-TGLM in the low-energy one (from ED to E_0 FOMs). In particular, DET-TGLM (which has a MS structure) dissipates less energy among all the analyzed CSEs. On average, DET-SPGFF, DET-SPL and DET-CDFF dissipate 1.7X, 2.4X and 2.1X more energy than the DET-TGLM. This is due to the combination of the DET and MS features, which both contribute to reduce energy consumption.

In general, DET CSEs can have rather complex layouts in those topologies where some parts of the circuit are replicated (as for DET-TGLM or DET-SPGFF). Instead, in other cases (DET-SPL or DET-CDFF), the DET functionality is simply accomplished by adopting a DET PG. Anyhow, in the

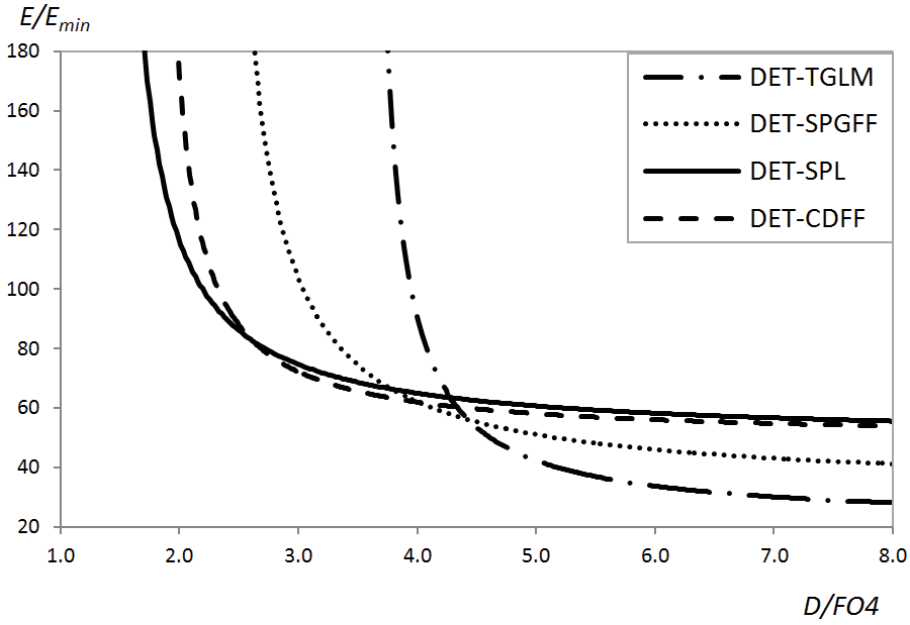


Fig. 5.11. EECs of DET CSEs: $C_L = 16C_{inv,min}$, $\alpha_{sw} = 0.25$, $T_{CK}/FO4 = 40$.

case of DET-TGLM, the more complex layout is compensated by the DET property that makes it the best CSE in the low-energy region with the TGFF.

DET-SPL, which has an EP structure, is the faster DET topology thanks to the simplicity of its $D - Q$ path. Also DET-CDFF has an EP structure and shows a good $E - D$ tradeoff but is competitive only for the ED FOM.

DET-SPGFF, which is an IP CSE, is never the most energy-efficient CSE because it suffers from a high layout complexity and also from the inclusion in the $D - Q$ paths of the clocked precharge transistors, which thus need to be oversized. This explains why results contrast with those in [NO05], where it was stated that the DET-SPGFF has a better ED product than DET-TGLM and DET-SPL in typical conditions.

The effect of load on DET CSEs is shown in Fig. 5.12a-b, where the EECs for $C_L = 64C_{inv,min}$ and a $C_L = 4C_{inv,min}$ conditions are plotted (in both cases $\alpha_{sw} = 0.25$ and $T_{CK}/FO4 = 40$). The speed of DET-SPL and DET-CDFF is nearly the same for large load (they have a nearly equal D_0), whereas the DET-SPL is significantly faster for small load since it has only two stages in the $D - Q$ paths. However, for large load, the DET-CDFF is more energy-efficient than DET-SPL from ED^3 FOM on, i.e. in almost all the $E - D$ space (differently from the previous similar discussion on the comparison of TGPL and SEPFF). This is because the conditional discharge

allows for considerably reducing energy (although DET-CDFF has a more complex layout than DET-SPL).

The effect of switching activity on DET CSEs is analyzed in Fig. 5.13a-b,

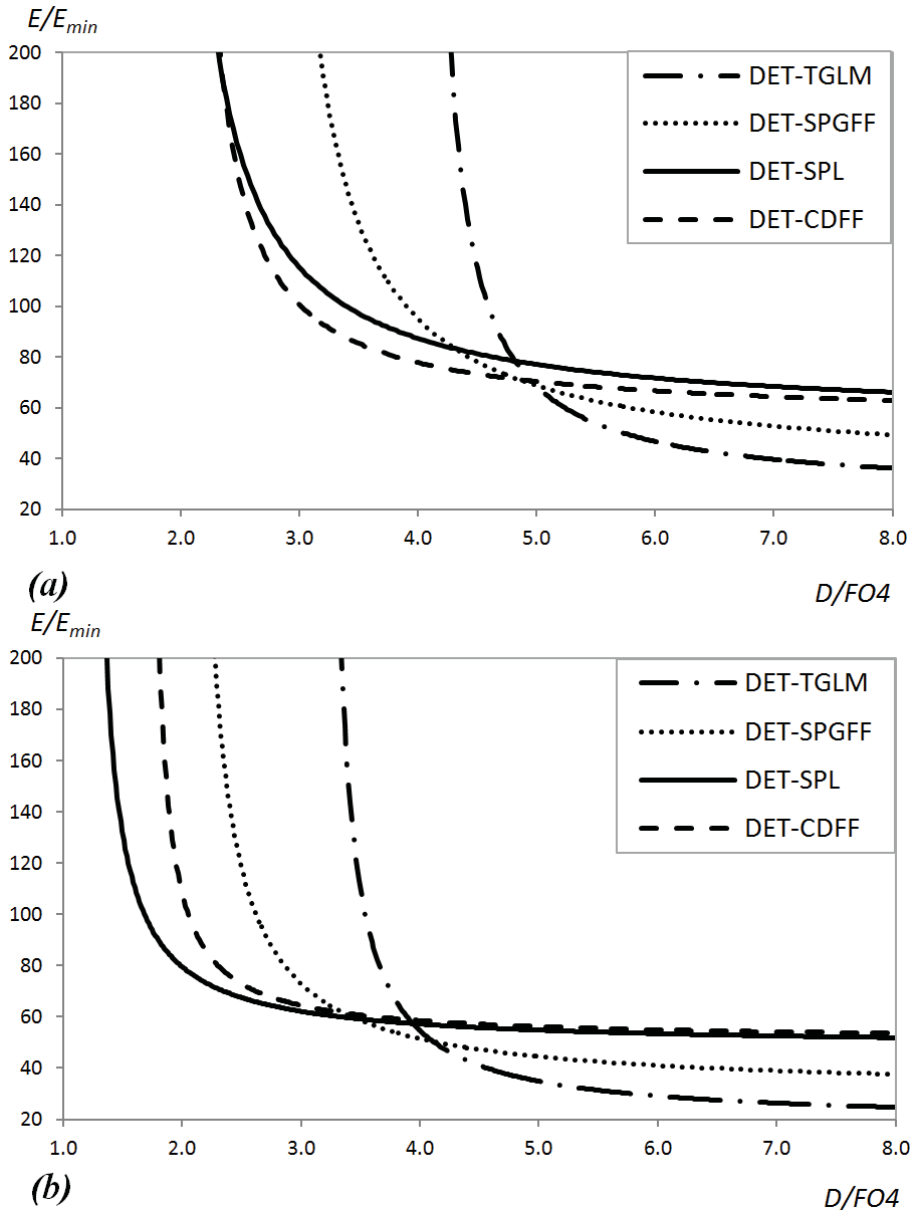


Fig. 5.12. EECs of DET CSEs: $C_L = 64C_{inv,min}$ (a) and $C_L = 4C_{inv,min}$ (b) ($\alpha_{sw} = 0.25$, $T_{CK}/FO4 = 40$).

where the EECs for $\alpha_{sw} = 0.1$ and $\alpha_{sw} = 0.5$ are reported (in both cases $C_L = 16C_{inv,min}$ and $T_{CK}/FO4 = 40$). Even if the DET-CDFF adopts the conditional discharge property, it is the most energy-efficient circuit only

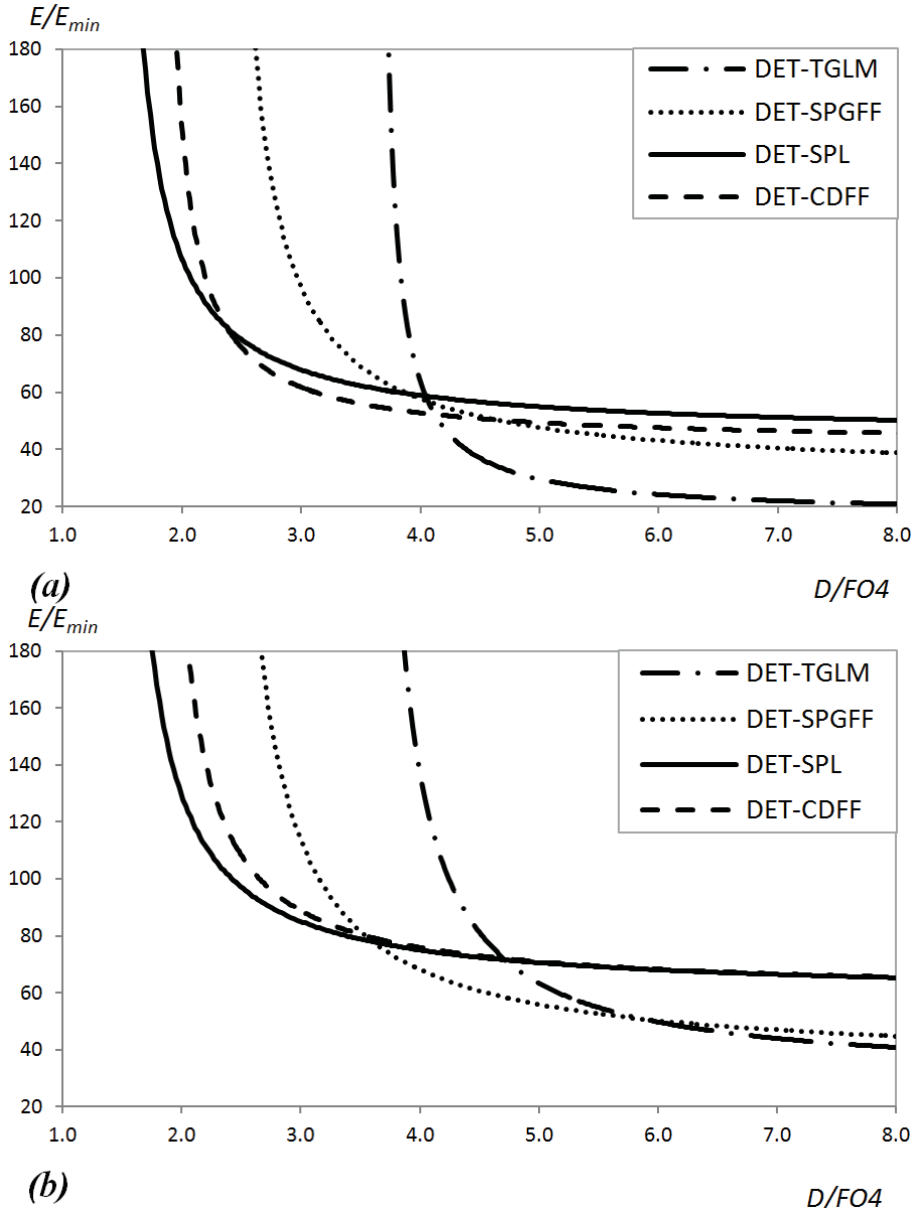


Fig. 5.13. EECs of DET CSEs: $\alpha_{sw} = 0.1$ (a) and $\alpha_{sw} = 0.5$ (b) ($C_L = 16C_{inv,min}$, $T_{CK}/FO4 = 40$).

around the ED^2 FOM. Indeed, for low switching activity, DET-TGLM takes advantage of the intrinsic absence of precharge, while DET-CDFF always suffers from the PG consumption. For analogous reasons, DET-SPGFF has an even higher energy due to precharge of internal nodes. For $\alpha_{sw} = 0.5$ DET-CDFF can no longer benefit from the conditional discharge and DET-TGLM suffers from the frequent transitions in its numerous internal nodes. On the other hand, DET-SPL is the most energy efficient in the region from D_0 to ED FOMs, and DET-SPGFF is the best at ED and E^2D FOMs. DET-TGLM is still the best circuit in the deep low-energy region.

5.6 Energy-Delay Global Comparison Among All CSEs

5.6.1 E^iD^j metrics

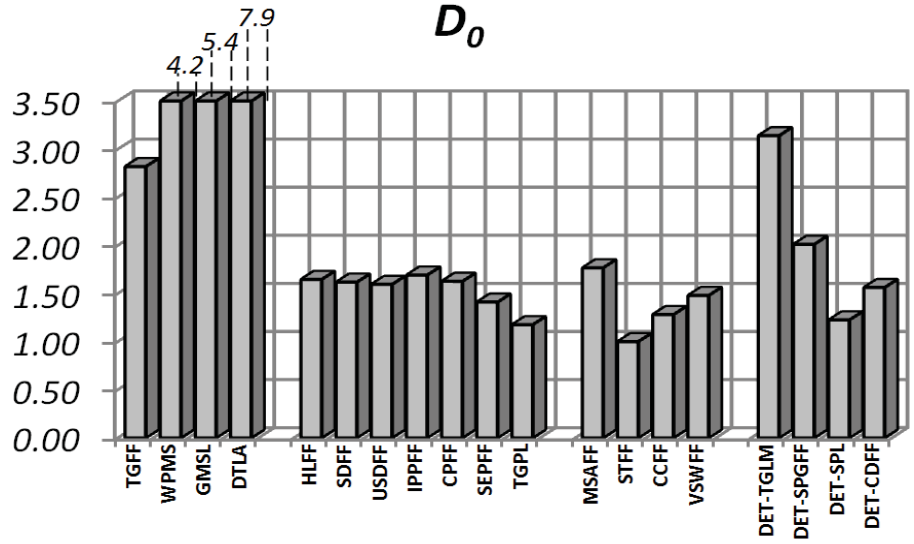
In Fig. 5.14a-e, the FOMs D_0, ED^3, ED, E^3D and E_0 of all CSEs, normalized to the best topology, are reported (again in the reference case). This permits to draw general conclusions on the comparison of the analyzed classes. It is apparent that Pulsed CSEs are the most energy-efficient in the high-speed region (D_0 and ED^3 FOMs) and the EP CSEs in particular result more energy-efficient than the IP CSEs in such a region. Since Pulsed CSEs are employed in real high-speed applications, EP topologies can be considered the best choice in such a case. The superiority of EP over IP CSEs is explained by considering that, in nanometer technologies, IP CSEs suffer from a complex routing between the stages involved in the $D-Q$ paths, which thus need to be oversized to avoid a speed penalty.

In particular, TGPL is the best circuit even in terms of ED product. Observe that, in high-speed applications, Pulsed CSEs can benefit from an even greater energy reduction when the PG is shared among various CSEs. The advantage of EP over IP CSEs no longer exists in the low-energy region (E^3D and E_0 FOMs).

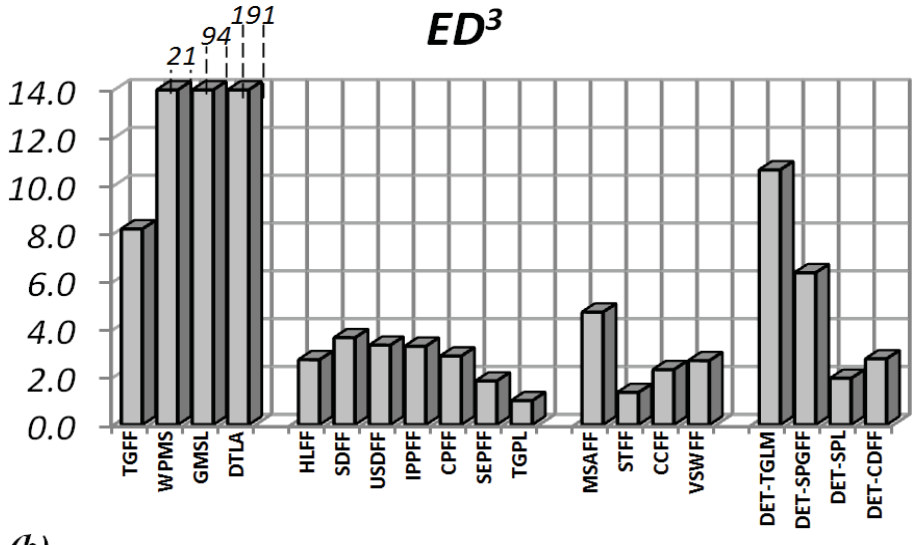
As expected, also Differential CSEs exhibit very good features in the high-speed region. Indeed their basic structures closely resemble those of Pulsed CSEs (STFF is the fastest topology among all those considered). Obviously the energy dissipation of Differential CSEs is high since they have to provide both polarities of the output. Some of them have a single-ended counterpart, like the STFF [NOW03] and the CCFF [KKJ01]. However, such single-ended versions (which are IP CSEs) are quite complex and it is found that their energy-efficiency is always worse than other analogous single-ended topologies (for this reason they are not included in the analysis). In the low-energy region, MSAFF is quite efficient since it achieves acceptable speed at the cost of a relatively low consumption.

MS CSEs are clearly the most energy-efficient ones in the low-energy region, whereas their speed is limited. Together with TGPL, TGFF and

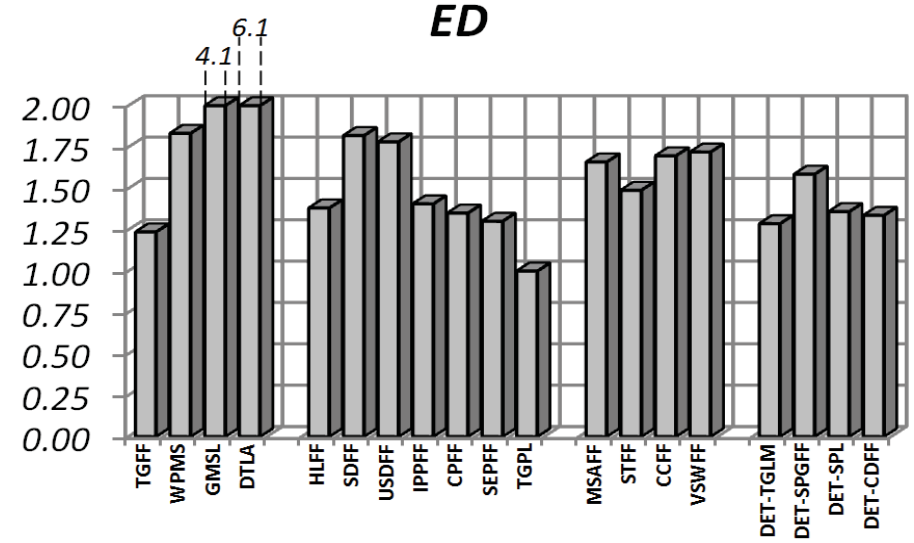
DET-TGLM offer also the best compromise in terms of *ED* product. Clock-gated CSEs are by far the worst circuits and have a degraded speed and energy compared to any other topology. Accordingly, Clock-gated CSEs are unsuitable for nanometer technologies. Among DET CSEs, the DET-TGLM represents the most energy-efficient solution in the deep low-energy region, together with TGFF. It is the DET counterpart of TGFF and they show similar performances since the greater layout complexity of DET-TGLM is compensated by the energy reduction due to the DET property.



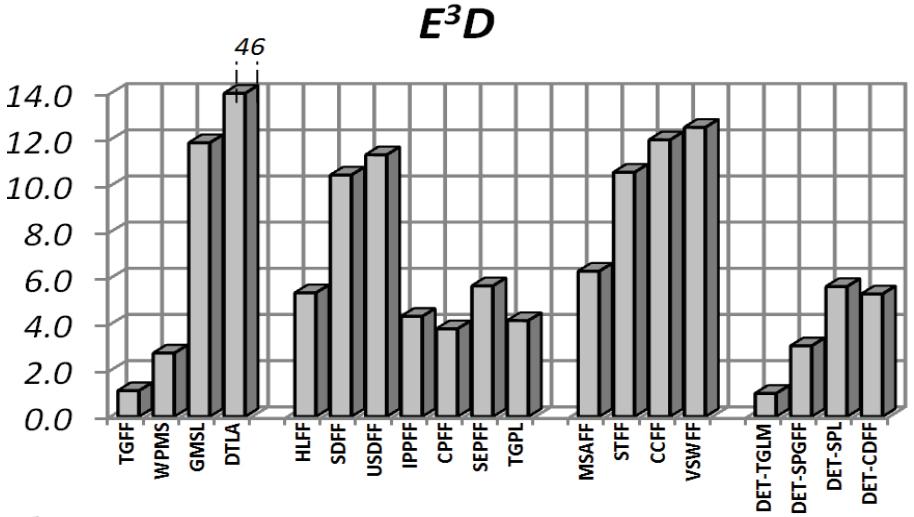
(a)



(b)



(c)



(d)

5.6.2 Selection of the most energy-efficient CSEs

An ideal EEC extrapolated by selecting the best circuits for each $E-D$ region is reported in Fig. 5.15. STFF exhibits the best D_0 and ED^5 products. TGPL is the best from ED^4 to ED FOMs. DET-TGLM has the best E^2D, E^3D and E_0 FOMs (TGFF has nearly the same performances in the low-energy region and is the best circuit in a narrow window between ED and E^2D FOMs). Hence, except for extreme high-speed designs, TGPL reveals itself as the most energy-efficient solution in a very wide region of practical

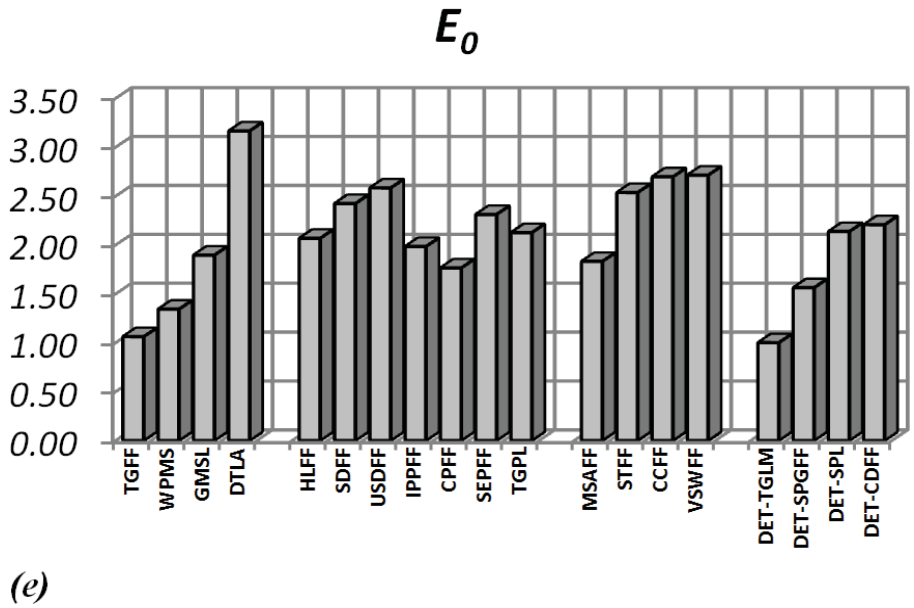


Fig. 5.14. D_0 (a), ED^3 (b), ED (c), E^3D (d), E_0 (e) normalized FOMs: $C_L = 16C_{inv,min}$, $\alpha_{sw} = 0.25$, $T_{CK}/FO4 = 40$.

applicability. MS CSEs based on transmission gates (TGFF, DET-TGLM) are the best when energy is the main concern.

Such results only partially agree with those on [GNO07] and lead to the different following considerations:

- the suitability of STFF for high-speed designs is now limited only to extremely high-speed applications ($j/i > 4$ in E^iD^j FOMs);
- the basic simplicity of TGPL leads to low layout parasitics (in general, topologies having simple layouts in their $D-Q$ paths are definitely favored);
- IP CSEs are no longer advantageous and, differently from the IPPFF in [GNO07], none of them is the most energy-efficient in any of the $E-D$ regions;
- in despite of the presumed ineffectiveness of DET topologies, DET-TGLM is slightly more energy-efficient than the TGFF in the low-energy region (in [GNO07] only TGFF is considered to extract the ideal EEC).

The above-mentioned CSEs are still the best ones even combining all the considered load and switching activity values. In general, a few differences emerge:

- For large load ($64C_{inv,min}$), STFF always achieves the best ED^4 FOM and TGPL does not always have the best ED product. By comparing

STFF and TGPL in the high-speed region, the delay of the latter one is nearly 1.03X, 1.10X and 1.20X greater in the small, average and large loading conditions, respectively. Instead, the energy consumption of STFF (which has a more complex routing and 1.3 ÷ 1.5X larger area) is nearly 1.9X greater in such a region.

- For small load ($4C_{inv,min}$), TGPL is always the best circuit in the extremely wide range $[ED^5 - ED]$.
- For low-switching activity (0.1), DET-TGLM is more efficient than TGFF. Indeed, it partly replicates the TGFF circuit but the increased nodes number is not a significant concern if the data rarely varies. For instance, DET-TGLM has the best overall ED for $C_L = 64C_{inv,min}$ and $\alpha_{sw} = 0.1$.
- For high-switching activity (0.5), TGFF replaces DET-TGLM as the best circuit in terms of E^2D, E^3D and E_0.

The significantly larger number of analyzed topologies and inclusion of the impact of layout parasitics are responsible for the above mentioned differences with respect to the results in [GNO07]. This is easily demonstrated by comparing the above results with those in Fig. 5.16, where layout parasitics are not considered at all. By comparing Fig. 5.16 with Fig. 15 in [GNO07], not surprisingly, the results are nearly coincident except

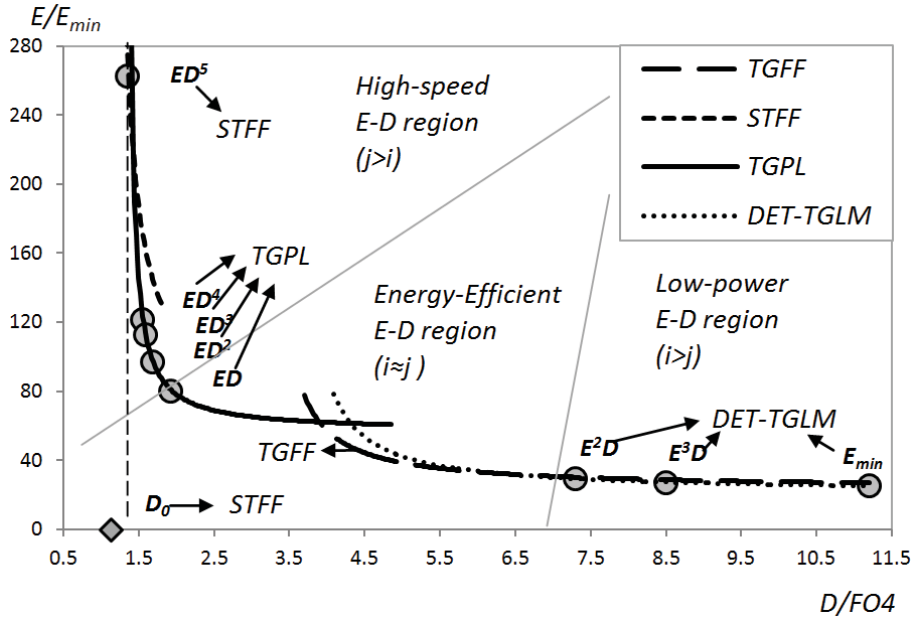


Fig. 5.15. Ideal EEC extracted selecting the most energy-efficient CSEs and minimum-E^iD^j designs.

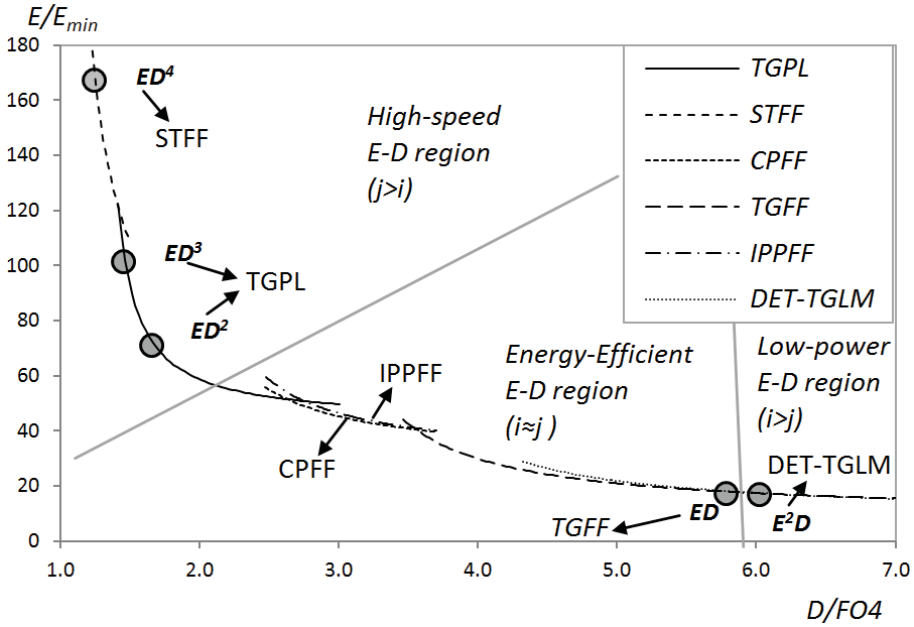


Fig. 5.16. Ideal EEC extracted selecting the most energy-efficient CSEs and minimum-E^iD^j designs (layout parasitics not included).

for CPFF and DET-TGLM that were not considered in [GNO07].

In particular, STFF shows now also the best ED^4 and is more suitable in a wider part of the high-speed region. TGPL no longer has the best ED (TGFF does) and IPPFF (together with CPFF) is the most energy-efficient in a non-negligible $E-D$ space window as in [GNO07]. The couple TGFF/DET-TGLM still exhibits the best features in the low-energy region.

5.7 Leakage

5.7.1 Leakage impact in active mode

By considering different logic depth values, it is found that the leakage energy in active mode does not influence the ranking of CSEs, although it can significantly impact the optimum transistor sizing. Thus, leakage cannot be merely considered only in standby mode (where it is the only source of dissipation). For instance, let us analyze the TGFF for $C_L = 16C_{inv,min}$ and $\alpha_{sw} = 0.1$. By considering $T_{CK}/FO4 = 10$ and $T_{CK}/FO4 = 80$ conditions, the optimal sizing of the circuit to minimize the product ED changes. Tab. V.II reports the optimum design variables w_k, and shows that a smaller size of the circuit is required when the leakage contribution increases (high $T_{CK}/FO4$), for a targeted FOM.

TABLE V.II: TGFF SIZING FOR $T_{CK}/FO4 = 10$ AND $T_{CK}/FO4 = 80$

$(\alpha_{sw} = 0.1)$ $(C_L = 16C_{inv,min})$	*Optimum sizing* $T_{CK}/FO4 = 10$	*Optimum sizing* $T_{CK}/FO4 = 80$
w_1	3	2
w_2	3	2
w_3	3	2
w_4	5	3
$\tau_{DQ,min}$ $[FO4]$	5.214	5.835
E_{TRAN} $[E_{min}]$	28.353	25.623
E_{STAT} $[E_{min}]$ $(T_{CK}/FO4 = 10)$	0.816	0.611
E_{STAT} $[E_{min}]$ $(T_{CK}/FO4 = 80)$	6.534	4.886
ED $[E_{min}]$ $(T_{CK}/FO4 = 10)$	152.072	153.083
ED $[E_{min} * FO4]$ $(T_{CK}/FO4 = 80)$	181.883	178.166

In general, CSEs exhibit an extremely high dynamic energy consumption compared to combinational logic gates, as clock is the signal with the highest transition rate within a chip. However, with technology scaling, the growing impact of leakage on transistor sizing can be significant even in active mode, according to Tab. V.II. In particular, the impact of leakage is certainly stronger for circuits that do not adopt precharge and whose topologies do not lead to frequent transitions on the internal nodes. The topologies exhibiting significant sizing changes when the parameter $T_{CK}/FO4$ is varied from 10 to 80, are reported in Tab. V.III, where:

- $\Delta W_{10\to80,max}$ is the maximum relative variation in the size of a single transistor.
- $\overline{\Delta W_{10\to80}}$ is the average relative variation by including all the CSE transistors.
- $\sigma_{\Delta W_{10\to80}}$ is the standard deviation of the relative variation by including all the FF transistors.

The three above quantities $\Delta W_{10\to80,max}$, $\overline{\Delta W_{10\to80}}$ and $\sigma_{\Delta W_{10\to80}}$ are evaluated by considering all the optimum E^iD^j designs for $C_L = 16C_{inv,min}$ and $\alpha_{sw} = 0.1$. From Tab. V.III, it is apparent that MS CSEs exhibit significant changes in the transistor sizes because of leakage, since their

TABLE V.III: OPTIMUM SIZING VARIATION FOR $T_{CK}/FO4 = 10$ AND $T_{CK}/FO4 = 80$

	$\Delta W_{10\to80,max}$	$\overline{\Delta W_{10\to80}}$	$\sigma_{\Delta W_{10\to80}}$
TGFF	66.7 %	12.5 %	20.3 %
WPMS	40.0 %	5.0 %	13.4 %
GMSL	66.7 %	10.1 %	20.8 %
DET-TGLM	40.0 %	5.7 %	12.3 %

dynamic consumption is rather low. On the other hand, transistors sizes in Pulsed and Differential CSEs are negligibly impacted by leakage, as energy is always dominated by the transient energy contribution.

5.7.2 Leakage impact in standby mode and tradeoff with delay

In standby mode, leakage currents represent the only source of dissipation. Especially when circuits stay idle for very long periods, standby leakage may be even more important than the active mode energy.

In the following we refer to the average leakage current $I_{Leak,avg}$. In Tab. V.IV, $I_{Leak,avg}$ is reported for the various CSEs and under three typical optimum sizings, i.e. minimum ED^3, ED and E^3D (the optimization is carried out in the reference case). Absolute and normalized (to the best circuit, highlighted in gray) values are reported.

TABLE V.IV: AVERAGE LEAKAGE (NORMALIZED TO THE MINIMUM IS REPORTED IN BRACKETS) UNDER VARIOUS OPTIMUM SIZING AND AVERAGE (AMONG THE FOMS) RATIO BETWEEN AVERAGE AND MINIMUM LEAKAGE

$C_L = 16C_{inv,min}$ $\alpha_{sw} = 0.25$ $T_{CK}/FO4 = 40$	*Min* ED^3 $I_{Leak,avg}$ $[I_{Leak,min}]$	*Min* ED $I_{Leak,avg}$ $[I_{Leak,min}]$	*Min* E^3D $I_{Leak,avg}$ $[I_{Leak,min}]$	*Average* $\left[\frac{I_{leak,avg}}{I_{Leak,minimum}}\right]$
TGFF	26.996 (1.65x)	9.390 (1.00x)	7.525 (1.03x)	1.219
WPMS	18.325 (1.12x)	12.575 (1.34x)	9.317 (1.28x)	1.208
GMSL	42.694 (2.60x)	28.383 (3.02x)	24.860 (3.41x)	2.070
DTLA	29.273 (1.78x)	26.113 (2.78x)	22.507 (3.08x)	85.428
HLFF	16.409 (1.00x)	9.739 (1.04x)	7.299 (1.00x)	1.323
SDFF	24.293 (1.48x)	17.778 (1.89x)	13.265 (1.82x)	1.329
USDFF	22.308 (1.36x)	17.397 (1.85x)	11.997 (1.64x)	1.293
IPPFF	21.592 (1.32x)	13.560 (1.44x)	11.387 (1.56x)	1.153
CPFF	20.329 (1.24x)	13.635 (1.45x)	8.863 (1.21x)	1.238
SEPFF	24.325 (1.48x)	15.277 (1.63x)	10.909 (1.49x)	1.158
TGPL	32.165 (1.96x)	19.951 (2.12x)	11.876 (1.63x)	1.293
MSAFF	19.815 (1.21x)	13.990 (1.49x)	9.079 (1.24x)	1.079
STFF	33.947 (2.07x)	19.170 (2.04x)	14.850 (2.03x)	1.113
CCFF	35.366 (2.16x)	24.423 (2.60x)	14.972 (2.05x)	1.172
VSWFF	36.495 (2.22x)	25.545 (2.72x)	16.600 (2.27x)	1.231
DET-TGLM	28.181 (1.72x)	15.765 (1.68x)	9.739 (1.33x)	1.315
DET-SPGFF	32.540 (1.98x)	15.632 (1.66x)	11.563 (1.58x)	1.165
DET-SPL	31.976 (1.95x)	17.810 (1.90x)	13.665 (1.87x)	1.207
DET-CDFF	26.109 (1.59x)	15.913 (1.69x)	15.356 (2.10x)	1.183

From Tab. V.IV, the circuits with the greatest leakage are the clock-gated and the Differential ones (except MSAFF), due to the high number of transistors and layout complexity (which leads to oversized transistors for a given speed target). On the other hand, all Pulsed CSEs (both EP/IP and SET-DET) except HLFF show a moderate leakage. This is explained by considering that such CSEs extensively employ stacked transistors and hence leakage is somewhat reduced [NC06].

DET CSEs (except DET-TGLM) do not show higher leakage currents than their SET counterparts, because they need to employ only a slightly more complex PG (DET-SPL and DET-CDFF) or because they can again exploit the leakage reduction due to stacking (DET-SPGFF). Instead, DET-TGLM is significantly worse than SET MS CSEs (TGFF and WPMS) since the increased complexity due to the duplication of some stages is not compensated by the DET property (as opposite to transient consumption in active mode).

As previously mentioned, MSAFF tends to be downsized compared to other Differential CSEs because of the simpler layout, and also because the branching due to the cross-coupling in the first stage would prevent significant speed improvements if the size were strongly increased.

Finally WPMS and in particular TGFF and HLFF have the minimum leakage. Indeed, MS CSEs cannot exploit stacking but have very simple structures and typically small transistors sizes. HLFF is the simplest among IP CSEs and extensively employs stacking.

When circuits operate in standby mode, the $E-D$ tradeoff must be reinterpreted as leakage-delay tradeoff, although the optimum CSEs sizings so far discussed, where the active mode energy is considered, is sill referred. In Fig. 5.17, such tradeoff is depicted. For practical delay ranges, SEPFF, TGPL, HLFF and TGFF show the best compromise.

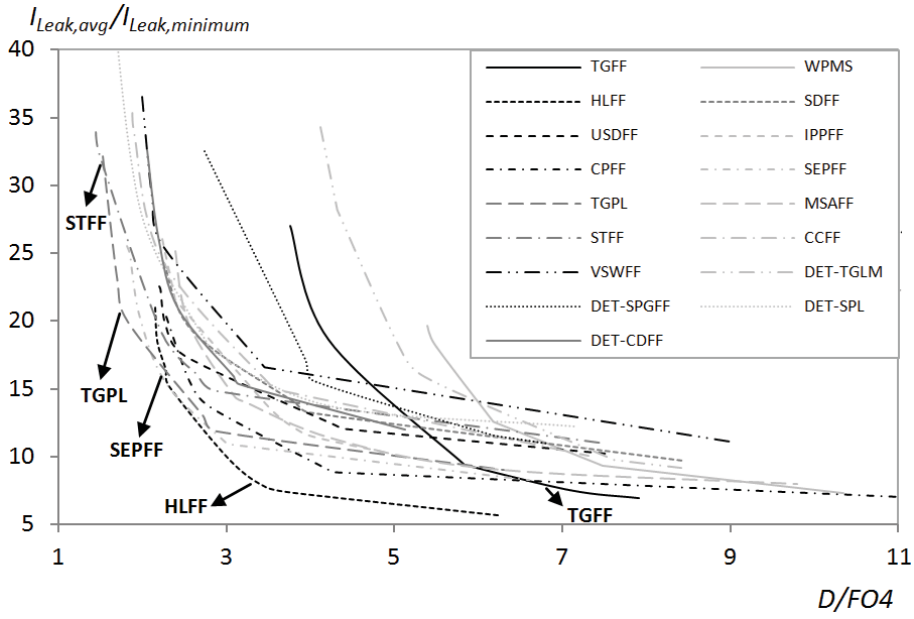


Fig. 5.17. Leakage-delay tradeoff. Optimization in active mode: $C_L = 16C_{inv,min}$, $\alpha_{sw} = 0.25$, $T_{CK}/FO4 = 40$.

5.7.3 Effectiveness of leakage reduction techniques

Leakage can be reduced by resorting to techniques like the Input State Assignment (ISA) [AFP04] or the Reverse Body Biasing (RBB) [RMM03].

The effectiveness of ISA can be simply analyzed by evaluating the proportion between the average leakage current and the minimum one, considering all possible values of input (data and clock) and output. Tab. V.IV reports the average ratio between $I_{Leak,avg}$ and the minimum leakage current $I_{Leak,minimum}$. Such average ratio is extrapolated by considering all optimum sizings for the FOMs in (5.19) and all the C_L, α_{sw} and $T_{CK}/FO4$ conditions.

Except for GMSL and DTLA, the energy savings achievable in standby mode through ISA are moderate, i.e. in the range $8-33\%$. Clock-gated CSEs behave differently, as the output of one of the gate in the PG of DTLA remains floating for $CK=1$, leading to an enormous leakage. It is worth noting that the results in the previous subsections were derived by neglecting the leakage contributions for $CK=1$, i.e. assuming an intentional (and necessary) $CK=0$ driving in standby mode for DTLA.

Effectiveness of RBB can be evaluated through parameter

$$S_{Leak}^{V_{BB}} = \left(\frac{\partial\left(log_{10} I_{I_{Leak,avg}}\right)}{\partial V_{BB}} \right)^{-1} \tag{5.20}$$

which is a figure of merit that was recently introduced to evaluate the ability to reduce leakage with small reverse body voltages, where V_{BB} is the body bias voltage. In particular, $S_{Leak}^{V_{BB}}$ represents the reverse body voltage that must be applied to reduce leakage by an order of magnitude. Starting from $V_{BB}=0$ ($V_{BB}=VDD$) for NMOS (PMOS), the body bias voltage is decreased (increased) by up to $0.4\,VDD=0.4\,V$ for NMOS (PMOS) transistors. The average RBB slope $S_{Leak}^{V_{BB}}$ (again considering all the aforementioned sizings) is reported in Fig. 5.18. From this figure, no appreciable differences arise among the CSE topologies.

Interestingly, by evaluating $S_{Leak}^{V_{BB}}$ for a single NMOS and a single PMOS transistor (both with $|V_{SG}|=0$ and $|V_{DS}|=V_{DD}$), one finds $S_{Leak}^{V_{BB}}=2.5$ and $S_{Leak}^{V_{BB}}=1.7$ respectively. The average of such values, 2.1, is very close to the $S_{Leak}^{V_{BB}}$ values found for complex circuits like the analyzed CSEs. This means that the leakage sensitivity of CSEs to body biasing is approximately that of a single transistor, which agrees with the previous intuition.

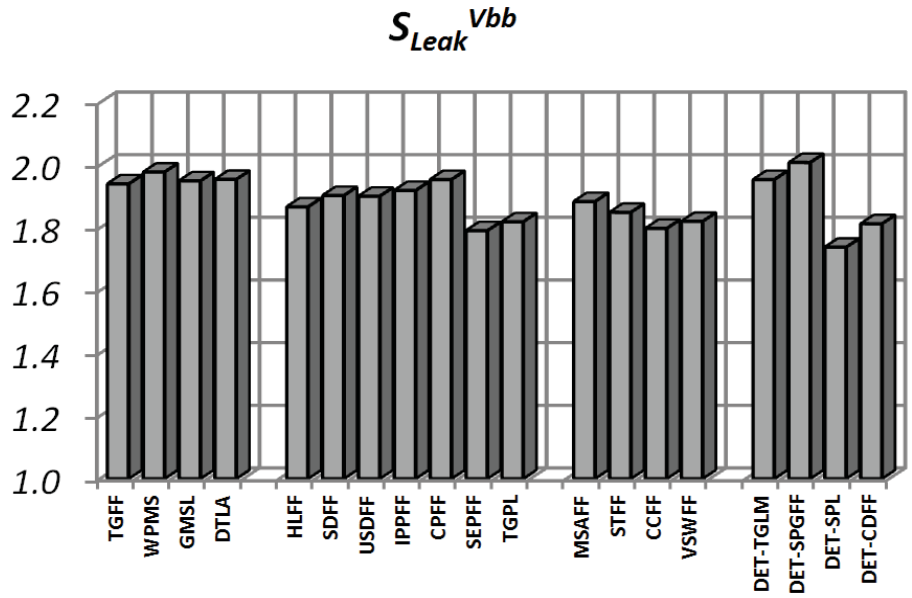


Fig. 5.18. Average RBB slope.

5.8 Silicon Area

5.8.1 Comparison of CSEs area

The silicon area occupied by CSEs can be accurately estimated by using the same procedure that is used to evaluate the interconnects length, as discussed in Paragraph 4.7. Hence, this procedure permits for the first time to extensively compare CSEs in terms of silicon area (previous works did not analyze this aspect).

Tab. V.V reports the area of the various CSEs under three typical optimum sizings, i.e. minimum ED^3, ED and E^3D (the optimization is carried out in the reference case). Absolute and normalized (with respect to the best circuit, highlighted in gray) values are reported.

Area is mostly dictated by the topological complexity. By inspection of Tab. V.V, one can draw the following main conclusions, which roughly hold for all the considered sizings:

- DTLA and the conditional Differential CSEs (CCFF and VSWFF) have the greatest area (for minimum ED-E^3D, the major complexity of DTLA is the dominant factor);
- TGFF, HLFF and MSAFF have the smallest areas. Indeed, as explained when dealing with leakage, MSAFF requires a very low area (despite its Differential nature) thanks to its regularity, while TGFF and HLFF have the simplest structures among the considered MS and Pulsed CSEs.

TABLE V.V: ABSOLUTE AREA UNDER VARIOUS OPTIMUM SIZING (AREA NORMALIZED TO THE MINIMUM IS REPORTED IN BRACKETS) IN THE FIRST THREE COLUMNS

LAYOUT EFFICIENCY UNDER VARIOUS OPTIMUM SIZING IN THE LAST EIGHT COLUMNS

$C_L = 16C_{min}$ $\alpha_{sw} = 0.25$ $T_{CK}/FO4 = 40$	***Min*** ED^3 ***Area*** $[\chi^2]$	***Min*** ED ***Area*** $[\chi^2]$	***Min*** E^3D ***Area*** $[\chi^2]$	$Transistors\ count/Area\ [\chi^2]$ %							
				ED^5	ED^4	ED^3	ED^2	ED	E^2D	E^3D	E_0
TGFF	628.8 (1.00x)	420.8 (1.00x)	420.8 (1.00x)	3.18	3.18	3.18	3.97	4.75	4.75	4.75	4.75
WPMS	729.6 (1.16x)	646.4 (1.54x)	625.6 (1.49x)	2.93	2.93	3.02	3.40	3.40	3.52	3.52	3.52
GMSL	797.6 (1.27x)	714.4 (1.70x)	714.4 (1.70x)	2.65	2.65	3.89	3.89	4.34	4.34	4.34	4.34
DTLA	1212.0 (1.93x)	1170.4 (2.78x)	1108.0 (2.63x)	3.88	3.88	3.88	4.02	4.02	4.24	4.24	4.24
HLFF	681.6 (1.08x)	462.4 (1.10x)	462.4 (1.10x)	2.62	2.69	2.93	2.93	4.33	4.33	4.33	4.53
SDFF	869.6 (1.38x)	703.2 (1.67x)	588.0 (1.40x)	2.87	2.87	2.87	3.18	3.56	4.11	4.25	4.29
USDFF	983.2 (1.56x)	816.8 (1.94x)	644.8 (1.53x)	2.85	2.85	2.85	3.18	3.43	4.34	4.34	4.34
IPPFF	816.8 (1.30x)	624.0 (1.48x)	603.2 (1.43x)	3.00	3.00	3.31	3.31	4.33	4.48	4.48	4.48
CPFF	912.0 (1.45x)	704.0 (1.67x)	541.6 (1.29x)	2.52	2.52	2.52	3.00	3.27	4.25	4.25	4.25
SEPFF	946.4 (1.51x)	759.2 (1.80x)	644.0 (1.53x)	2.69	2.75	2.75	3.16	3.42	4.04	4.04	4.04
TGPL	780.8 (1.24x)	635.2 (1.51x)	552.0 (1.31x)	3.07	3.07	3.07	3.66	3.78	4.19	4.35	4.35
MSAFF	691.2 (1.10x)	504.0 (1.20x)	504.0 (1.20x)	3.27	3.27	3.76	5.16	5.16	5.16	5.16	5.16
STFF	1202.4 (1.91x)	765.6 (1.82x)	724.0 (1.72x)	2.66	2.66	2.66	2.81	4.18	4.42	4.42	4.55
CCFF	1397.6 (2.22x)	1106.4 (2.63x)	804.0 (1.91x)	2.50	2.50	2.50	2.50	3.16	4.35	4.35	4.35
VSWFF	1397.6 (2.22x)	1106.4 (2.63x)	804.0 (1.91x)	2.36	2.36	2.36	2.73	2.98	4.10	4.10	4.10
DET-TGLM	761.6 (1.21x)	616.0 (1.46x)	595.2 (1.41x)	3.08	3.41	3.41	4.08	4.22	4.37	4.37	4.37
DET-SPGFF	1015.2 (1.61x)	650.4 (1.55x)	650.4 (1.55x)	3.15	3.15	3.15	4.77	4.92	4.92	4.92	4.92
DET-SPL	744.8 (1.18x)	599.2 (1.42x)	536.8 (1.28x)	2.66	2.66	2.95	3.43	3.67	3.67	4.10	4.10
DET-CDFF	925.6 (1.47x)	700.8 (1.67x)	700.8 (1.67x)	2.56	2.56	3.03	3.32	4.00	4.00	4.00	4.12
			Statistics	ED^5	ED^4	ED^3	ED^2	ED	E^2D	E^3D	E_0
			μ	2.87	2.89	3.06	3.50	3.94	4.29	4.33	4.36
			σ	0.36	0.37	0.44	0.68	0.61	0.38	0.35	0.35
			σ/μ	0.12	0.13	0.14	0.20	0.16	0.09	0.08	0.08

As concerns EP CSEs, the values in Tab. V.V are somewhat pessimistic. Indeed, when sharing the PG among an increasing number of latches, the area increase of the PG is very low. Thus, the actual area evaluation is affected by the number of latches sharing the same PG.

5.8.2 Area-delay tradeoff

The area-delay tradeoff is illustrated for the reference case in Fig. 5.19. From this figure, the area-delay tradeoff closely resembles the energy-delay tradeoff discussed in Paragraphs 5.5-5.6. The reason is that (differently from the leakage-delay tradeoff) the overall energy dissipation is strongly related with the area and the size of the circuits. The main differences with the composite EEC in Fig. 5.15 are the very good tradeoff offered by the HLFF in the delay range $[3-5]\ FO4$ and the better features of TGFF with respect to DET-TGLM in the low-energy (i.e. high delay) region.

The area degradation versus sizing (i.e., when optimizing FOMs where more emphasis is given to the speed) is also analyzed. The results in Fig. 5.20 (where MS, Pulsed and Differential CSEs are depicted with continuous, dashed and dotted lines, respectively) refer to the usual optimization conditions and are normalized with respect to the minimum area for each CSE (obviously achieved when optimizing E_0). Note that the area is a

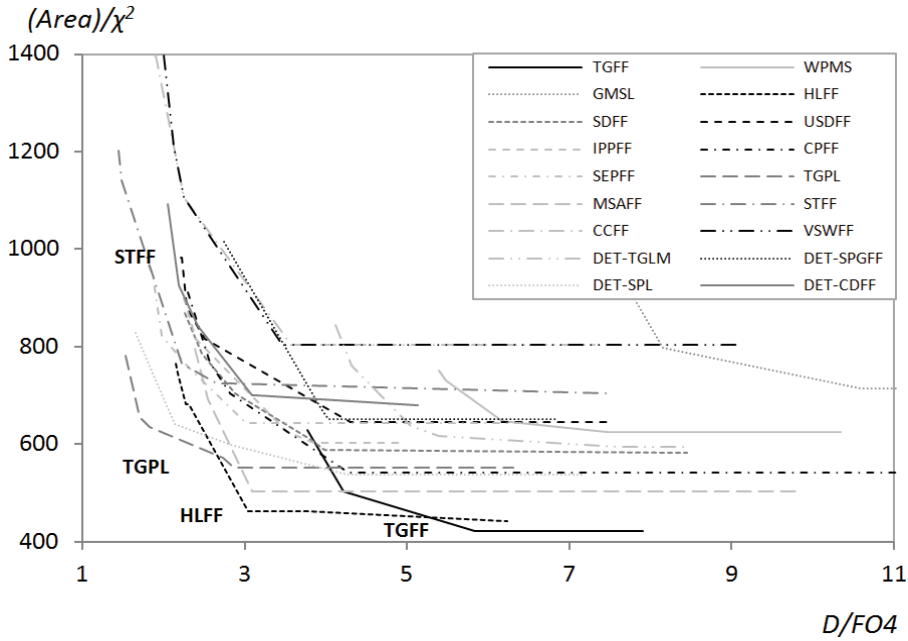


Fig. 5.19. Area-delay tradeoff. Optimization in active mode: $C_L = 16C_{inv,min}$, $\alpha_{sw} = 0.25$, $T_{CK}/FO4 = 40$.

parameter that does not change continuously, given the use of folded layout technique when dealing with large transistors.

Differential CSEs see the highest relative increase in their area (up to 1.8X) when they are progressively increased for smaller delays. Indeed, their complex layouts and the high branching effects due to local wires' parasitics and additional gates (not lying in the $D-Q$ paths) require a significant transistor oversizing of their critical stages. Pulsed CSEs (both IP and EP) show an intermediate behavior, with area increases up to $1.4-1.7$X. MS CSEs exhibit the smallest relative increase, up to $1.1-1.5$X.

5.8.3 Area related properties

In order to express how the topology is amenable for efficient physical design, in Tab. V.V the layout-efficiency, which is defined as the ratio of transistors count and the CSE area normalized to χ^2, is evaluated (it represents the number of transistors in a square with side χ). Due to the large number of different sizings and conditions, the mean value μ, the standard deviation σ and the variability σ/μ of the layout efficiency for each CSE topology are evaluated.

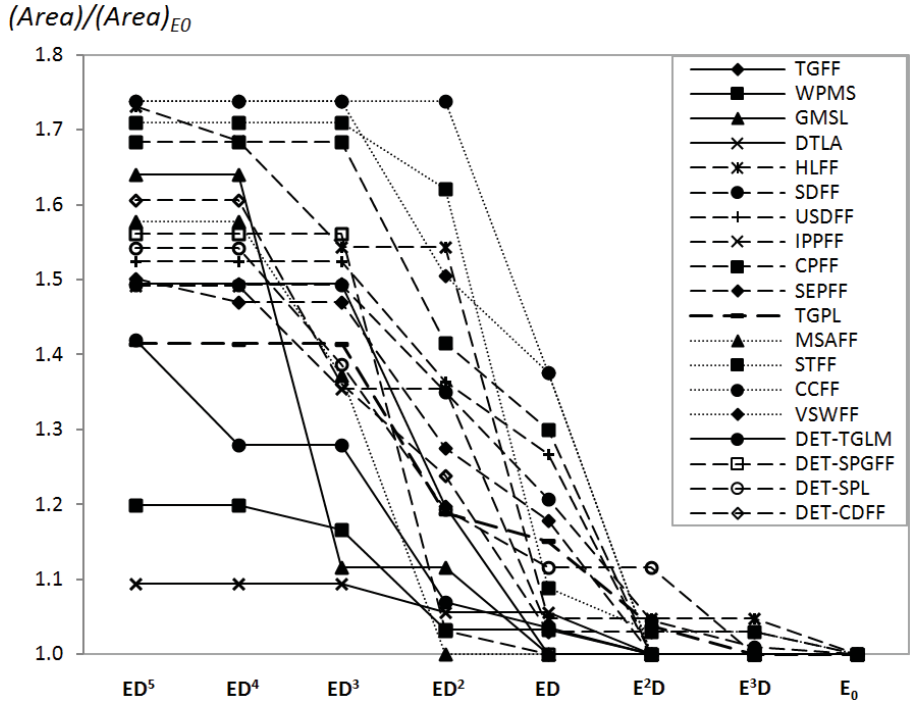


Fig. 5.20. Area degradation (normalization to E_0 sizing). Optimization in active mode: $C_L = 16C_{inv,min}$, $\alpha_{sw} = 0.25$, $T_{CK}/FO4 = 40$.

From Tab. V.V, as expected the layout efficiency decreases in high-speed designs (i.e., minimum ED^5, ED^4 and ED^3) since transistors (those in $D-Q$ paths) must be larger, compared to low-energy designs. Moreover, the layout efficiency tends to be almost the same for all CSEs when referring to the low-energy sizings. This is quantitatively confirmed by the variability σ/μ which is very small (less than 10%). On the other hand, bigger differences are found in high-speed designs (σ/μ is up to 20%).

It is also interesting to analyze the relationship between area and leakage for the various CSEs. By considering all the C_L, α_{sw} and $T_{CK}/FO4$ conditions and all the sizings, (i.e., on the whole, 216 different sizings), the correlation coefficient between the average leakage current $I_{Leak,avg}$ (minimum leakage current $I_{Leak,minimum}$) and area was evaluated. This correlation coefficient turns out to be very close to unity (always larger than 0.95 for any CSE), which means that area is always proportional to leakage for any specific CSE topology.

To infer if the area-leakage correlation can be assumed as a general property independently of the specific CSE topology, the correlation coefficients relative to the 19 CSEs altogether is also analyzed. The latter turns out to be 0.71 (0.80) for $I_{Leak,avg}$ ($I_{Leak,minimum}$), which is still rather close to unity. Hence, area can be still considered to be almost linearly related to leakage despite of the CSE topology. In other words, the silicon area of CSEs can be inferred immediately from the analysis of leakage (i.e., it does not require a separate analysis). Quantitatively, by again considering the 19 CSEs altogether, the linear proportionality coefficient between $I_{Leak,avg}$ ($I_{Leak,minimum}$) and area results to 30.8 $nA/\mu m^2$ (24.0 $nA/\mu m^2$), which is a very useful information to roughly estimate leakage once the area is given (and vice versa).

5.9 Clock Load

5.9.1 Clock load comparison and tradeoff with delay

The clock load C_{CK} of a CSE is defined as the capacitance seen from the CSE clock terminal, and is an important feature since it is closely related with the design of the clock network, which is responsible for a large fraction (up to $30-50\%$ [NO05]) of the whole energy budget in high-performance microprocessors [GBP98], [BB98], [ACP10-1]. Indeed, the higher the clock load, the larger the clock buffers that locally distribute the clock signal to CSEs throughout the various clock domains [WH04], [ACP10-1]. Therefore, in addition to the evaluation of the energy spent to charge/discharge the clock input capacitance (see Paragraph 5.2.3), the clock load is a further figure of merit since its value is inherently related to the consumption of the clock network.

TABLE V.VI: CLOCK LOAD UNDER VARIOUS OPTIMUM SIZING (CLOCK LOAD NORMALIZED TO THE MINIMUM IS REPORTED IN BRACKETS) AND AVERAGE (AMONG THE FOMS) PERCENTAGE CONTRIBUTION OF CLOCK WIRES

$C_L = 16C_{inv,min}$ $\alpha_{sw} = 0.25$ $T_{CK}/FO4 = 40$	*Min* ED^3 C_{CK} $[C_{min}]$	*Min* ED C_{CK} $[C_{min}]$	*Min* E^3D C_{CK} $[C_{min}]$	*Average* $\left[\frac{C_{CK,par}}{C_{CK}}\right]$ %
TGFF	25.060 (10.01x)	11.094 (4.43x)	10.427 (4.16x)	53.39
WPMS	16.996 (6.79x)	13.919 (5.56x)	12.346 (4.93x)	67.98
GMSL	6.583 (2.63x)	4.917 (1.96x)	4.917 (1.96x)	26.14
DTLA	2.504 (1.00x)	2.504 (1.00x)	2.504 (1.00x)	60.06
HLFF	14.547 (5.81x)	6.660 (2.66x)	5.994 (2.39x)	45.53
SDFF	15.430 (6.16x)	8.901 (3.55x)	6.178 (2.47x)	34.90
USDFF	18.237 (7.28x)	13.853 (5.53x)	8.469 (3.38x)	48.17
IPPFF	10.125 (4.04x)	4.683 (1.87x)	4.683 (1.87x)	47.02
CPFF	20.563 (8.21x)	12.674 (5.06x)	7.593 (3.03x)	41.17
SEPFF	5.841 (2.33x)	4.508 (1.80x)	3.841 (1.53x)	53.76
TGPL	9.045 (3.61x)	5.841 (2.33x)	4.508 (1.80x)	43.83
MSAFF	7.240 (2.89x)	4.129 (1.65x)	3.129 (1.25x)	45.46
STFF	10.381 (4.15x)	5.057 (2.02x)	4.391 (1.75x)	31.98
CCFF	16.801 (6.71x)	10.912 (4.36x)	5.624 (2.25x)	36.94
VSWFF	16.355 (6.53x)	10.798 (4.31x)	5.178 (2.07x)	42.21
DET-TGLM	33.411 (13.34x)	19.258 (7.69x)	15.258 (6.09x)	49.91
DET-SPGFF	23.486 (9.38x)	8.113 (3.24x)	4.446 (1.78x)	38.51
DET-SPL	16.532 (6.60x)	9.148 (3.65x)	7.183 (2.87x)	22.97
DET-CDFF	10.815 (4.32x)	5.516 (2.20x)	5.516 (2.20x)	31.34

In Tab. V.VI the clock load of the various CSEs under three typical optimum sizings, i.e. minimum ED^3, ED and E^3D, is reported (the optimization is carried out in the reference case). Absolute and normalized (with respect to the best circuit, highlighted in gray) values are reported.

CSEs have obviously a decreasing clock load when going towards low-energy designs, except DTLA, which has the minimum and constant clock load for all sizings (its PG sees a small internal load and hence is always minimum-sized).

MS CSEs exhibit the highest clock load in almost all conditions (the loads seen by the true and complementary versions of clock signals are added), since, independently from their sizing, they have a quite high number of clocked transistor and clock interconnects.

DET-SPGFF shows wide clock load variations when sized for high-speed or low-energy. Indeed, as previously mentioned, clocked precharge transistors lie in two of its four $D-Q$ paths and hence need to be strongly oversized. This is not the case when sizing for low-energy.

EP CSEs exhibit a very small clock load thanks to the "decoupling" effect accomplished through the use of a PG. Since the PG dissipation is already fully accounted for, EP CSEs reveal another significant advantage, given that they do not bring a great load to clock distribution network. DET-SPL is slightly worse because of the features of its PG, which does not

guarantee a full decoupling. Also the clock gating logic (GMSL) and the NOR gates in the STFF separate the external clock from the internal nodes and hence such CSEs have a low clock load.

The tradeoff between the clock load and the delay can be understood from Fig. 5.21, which reports the clock load increase with respect to the minimum-energy sizing when CSEs are progressively sized for high speed (continuous, dashed and dotted lines for MS, Pulsed and Differential CSEs respectively). From Fig. 5.21, EP CSEs (except DET-SPL) show clock load increments up to 2.5X compared to the minimum energy sizing, which are relatively low compared to other classes (because of the presence of the PG, as explained above). IP CSEs (except DET-SPGFF), MS, MSAFF and STFF show clock load increments up to 3.5X. Conditional Differential CSEs (CCFF and VSWFF) reach nearly 4X clock load increase. For the previously mentioned reasons, DET-SPGFF exhibits the greatest increase (up to 5.5X).

5.9.2 Impact of layout parasitics on the clock load

In the analysis reported above, the clock load includes the contribution of layout parasitics. Here, the fraction of clock load due to these parasitics is evaluated in detail. To this aim, all the C_L, α_{sw} and $T_{CK}/FO4$ conditions and all the minimum E^iD^j designs (i.e., 216 different sizings) are considered. In Tab. V.VI, the average percentage ratio between the clock load fraction $C_{CK,par}$ due to layout parasitics and the total clock load C_{CK} is reported.

From Tab. V.VI, the layout parasitics are a sizeable fraction of the overall clock load, which typically is in the $40-60\%$ range (i.e., the layout parasitics can even account for most of the clock load in a CSE). This confirms that layout parasitics must be necessarily taken into account to fairly compare CSEs, although they were neglected in previous papers.

Globally, MS CSEs (except GMSL) have the most complex clock (and complementary clock) routing paths, showing an impact of clock wires higher than 50%. An exception to this trend is represented by DET-SPL and DET-CDFF, where the clock terminal is not decoupled from some internal transistors (because of turned on transmission gates) and hence the clock wires have a minor impact on C_{CK}.

5.9.3 Joint CSEs and clock distribution energy dissipation

According to the traditional approach adopted in the literature, up to this point the CSEs comparison has been carried out by considering the dissipation related to the only CSEs. However, as previously mentioned, the dissipation of clock buffers in the clock domains is directly connected with the clock load. Therefore, one has to further investigate this aspect and in case revise the so far reported results, by carrying out an analysis that include the clock network contribution in the overall energy breakdown.

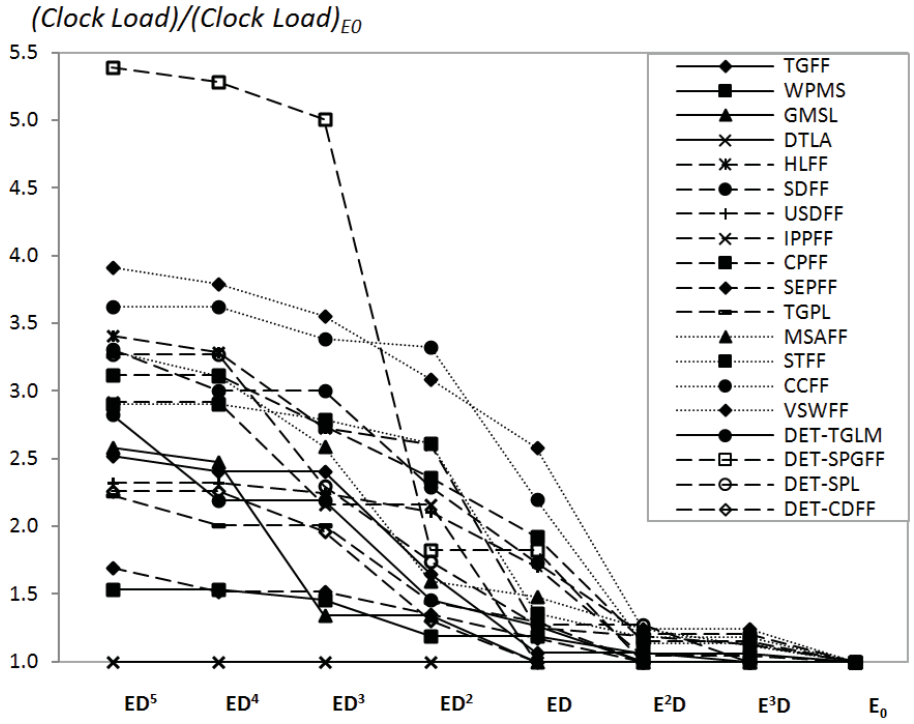


Fig. 5.21. Clock load degradation (normalization to E_0 sizing). Optimization in active mode: $C_L = 16C_{inv,min}$, $\alpha_{sw} = 0.25$, $T_{CK}/FO4 = 40$.

Traditionally, when designing clock networks, a steep clock waveform, typically featuring a $FO3$ clock slope, is ensured. However, in [ACP10-1] it is shown that a proper clock slope optimization with a FOX value ($X > 3$), allows to reduce the overall clocking energy (i.e., clock buffers and CSEs), at the cost of a negligible speed and local skew/jitter degradation (see Chapter 6).

As shown in [ACP10-1], the energy of a tapered clock buffer driving a clock load equal to C_L and featuring an FOX slope is

$$E_{buf} = \left(2\frac{1-X^N}{1-X} - 1\right) jC_{inv,min}{V_{DD}}^2 + T_{CK}V_{DD}jI_{Leak,min}\frac{1-X^N}{1-X} \qquad (5.21)$$

where both dynamic and static energy are taken into account, $jC_{inv,min}$ is the capacitance of the first buffer stage (in the following $j = 16$) and $N = log_X\left(C_L/jC_{inv,min}\right)$ is the number of buffer stages.

Differently from [ACP10-1], given that the clock wires distributing the clock signal throughout the domain produce an increment of C_L that is

independent of CSEs features, they are neglected in the following analysis. Hence, C_L is simply equal to MC_{CK}, being M the number of CSEs within the clock domain (in the following $M = 128$).

In Tab. V.VII the energy E_{buf} (with respect to the energy due to the only CSEs that is ME_{CSE}) due to clock buffers driving the CSEs clock load is reported in the case of $FO3$ clock slope and for the optimum value FOX_{opt} (i.e., X_{opt} is the optimum tapering factor leading to the minimum clocking energy in the clock domain). The values of X_{opt} are also reported in Tab. V.VII (see [ACP10-1] for a detailed discussion on how these values are

TABLE V.VII: PERCENTAGE ENERGY INCREMENT DUE TO A CLOCK TAPERED BUFFER DRIVING $M = 128$ CSES (NORMALIZED TO CSES ENERGY, ME_{CSE})

	FO3 Clock Slope			***FOX_{opt} Clock Slope***					
	Min ED^3 E_{buf} [%ME_{CSE}]	***Min ED E_{buf} [%ME_{CSE}]***	***Min E^3D E_{buf} [%ME_{CSE}]***	***Min ED^3 E_{buf} [%ME_{CSE}]***	***Min ED E_{buf} [%ME_{CSE}]***	***Min E^3D E_{buf} [%ME_{CSE}]***	***Min ED^3 X_{opt}***	***Min ED X_{opt}***	***Min E^3D X_{opt}***
TGFF	73.66	68.47	69.82	30.0	16.7	16.5	4.5	6	6
WPMS	58.49	61.57	62.56	29.1	30.2	29.7	4	4	4
GMSL	16.68	16.07	17.23	10.0	7.8	8.3	4	4.5	4.5
DTLA	4.59	4.91	5.14	2.8	2.5	2.2	4	4.5	5
HLFF	28.21	18.83	18.96	20.9	11.1	9.3	3.5	4	4.5
SDFF	21.88	18.29	15.92	16.3	10.4	9.3	3.5	4	4
USDFF	26.91	25.63	22.26	20.0	19.1	13.0	3.5	3.5	4
IPPFF	18.84	14.85	16.09	13.8	8.0	7.3	3.5	4	4.5
CPFF	38.75	34.68	28.29	28.3	19.8	13.0	3.5	4	4.5
SEPFF	9.02	10.05	10.78	4.7	4.5	4.7	4.5	5	5
TGPL	14.29	13.65	13.94	8.8	6.9	6.0	4	4.5	5
MSAFF	11.13	10.08	10.58	6.8	5.1	4.4	4	4.5	5
STFF	10.34	9.61	9.93	6.5	5.0	4.4	4	4.5	5
CCFF	21.88	18.87	13.30	13.2	11.6	6.7	4	4	4.5
VSWFF	22.01	18.45	11.89	13.2	11.3	6.0	4	4	4.5
DET-TGLM	113.91	106.51	109.36	37.2	27.9	28.1	5	5.5	5.5
DET-SPGFF	34.50	26.77	19.67	17.2	10.0	6.7	4.5	5.5	6
DET-SPL	24.92	24.73	23.41	11.0	10.7	8.7	5	5	5.5
DET-CDFF	18.31	16.28	16.40	7.3	5.5	5.6	5.5	6	6

derived given the CSEs features) and the considered sizings are minimum ED^3, ED and E^3D (in the reference case).

By inspection of Tab. V.VII, it is apparent that, except for the MS CSEs, when considering a steep clock waveform (i.e., $FO3$), the energy increment E_{buf} is typically $10-30\%$ of the CSEs energy, nearly regardless of CSEs sizing (actually it slightly diminishes for low-energy design), and EP CSEs are again rewarded.

Anyhow, the previously reported rankings change in the low-energy region because of the behavior of MS CSEs, which, due to their basic low energy and their high clock load, see a very high energy increment (up to 100% for the DET-TGLM) due to E_{buf}.

In the case the optimum clock slope is used to minimize the overall clocking energy, it can be easily seen that the energy increments become much more similar for all the CSEs topologies, and they become equal to $30-35\%$ for MS CSEs. Moreover, the energy increments significantly diminish for low-energy sizings. Hence, by also considering that CSEs speed is not practically degraded by assuming a smoother slope up to $FO5-FO6$ [ACP10-1], the previously reported rankings in the energy-delay space do not change significantly when adopting the optimum clock slope. Nevertheless, one should emphasize that, although they remain the most suitable circuits for very low-energy applications, the inclusion of E_{buf} worsen the performances of MS CSEs (e.g., for minimum ED).

5.10 Summary

In this chapter, results relative to an exhaustive comparison [ACP11-1], [ACP11-2] of a large number of CSEs (19 topologies belonging to four different classes) in nanometer (65-nm) CMOS technology have been reported, differently from the other most relevant analyses in the literature that have so far adopted technologies up to 0.13 µm. The comparison has been performed in the whole energy-delay-area design space. The impact of layout parasitics has been included in the transistor-level design phase. The contribution of leakage has been considered in both standby and active mode, weighting it according to the logic depth in the active case. Wide loading and switching activity conditions have been explored, and other properties (e.g., the clock load) have been analyzed in detail.

Through the adoption of a novel and general framework for CSEs design (see Chapter 4), analysis and comparison, several results have been derived. As opposite to previous papers, figures of merit that designers are familiar with have been considered to gain an insight into the considered tradeoffs in a wide range of applications. Analysis showed that the results are different from previous papers because, here, the layout parasitics have been

explicitly included from the beginning and a much wider range of topologies has been considered [ACP11-1], [ACP11-2].

According to the presented results, the fastest topology is the STFF, the best low-energy CSEs are the DET-TGLM and TGFF, whereas the most energy-efficient throughout a wide region of the energy-delay design space is the TGPL. Moreover, the best topologies within each of the main CSE classes have been identified as well. As a further contribution, the resulting trends have been justified through detailed circuit analyses.

For the first time, the layout efficiency of CSEs has been analyzed. In particular, HLFF, MSAFF and TGFF exhibit a very efficient area-delay tradeoff. Moreover, it has been shown that area is almost proportional to leakage regardless of the CSE topology and the transistor sizing. Hence the leakage of CSEs can be inferred immediately from the analysis of silicon area and vice versa (i.e., a separate analysis is not required).

The differences between the leakage-delay and the more general energy-delay tradeoff have been pointed out. It has also been shown that leakage has a significant impact on the optimum transistor sizing, especially for MS CSEs.

The clock load seen from the clock terminal of a CSE and the related dissipation of the clock distribution network, has also been analyzed. Results showed that the clock load is severely impacted by layout parasitics, and that EP CSEs have a small clock load thanks to the decoupling effect brought by the PG. It is also shown that, by including the impact of local clock distribution buffers, whose dissipation is directly related with CSEs clock load, the rankings of CSEs in the E-D space do not change significantly, unless for the MS class that is somewhat penalized.

As a general remark, simpler basic structures are rewarded in nanometer technologies because of the strong impact of layout parasitics. In particular, EP topologies, and specifically the TGPL, have been recognized as the most efficient CSE topologies in a very wide range of applications from many points of view. Indeed, the presence of the PG ensures the possibility to achieve time-borrowing by properly adjusting the transparency window, a greater speed with respect to IP CSEs because of the reduced transistors stacking and a small clock load. Area and leakage are moderate, whereas the energy consumption, which entirely includes the PG contribution, is obviously somewhat high. Hence, the previous considerations are further reinforced by considering that the energy and area contribution of PG can be further reduced if it is shared among different latches.

Chapter 6

ENERGY-EFFICIENT CLOCK SLOPE DESIGN AT THE CLOCK DOMAIN LEVEL

In this chapter, the influence of the clock slope on the speed of various classes of clocked storage elements (CSEs) and on the overall energy dissipation of both CSEs and clock domain buffers is analyzed. The analysis shows that an optimum clock slope exists, which minimizes the energy spent in a clock domain [ACP10-1]. Results show that the clock slope requirement can be relaxed with respect to traditional assumptions, leading up to 30 ÷ 40% energy savings and at a very small speed performance penalty [ACP10-1]. The effectiveness of the clock slope optimization is discussed in detail for the existing classes of CSEs. The impact of such an optimization in terms of additive skew and jitter contributions is discussed, together to the analysis of the impact of technology scaling [ACP10-1].

6.1 Basic Considerations on the Role of the Clock Slope

In the design of the clock network, the most important requirements are the maximum skew and jitter, along with the clock slope (i.e., the rise/fall time of clock edges) [RDS94], [GBP98], [BB98].

In general, the clock slope impacts the speed performances of CSEs, but also their energy consumption and their robustness [PCB01]. Indeed, by adopting a smooth clock slope (i.e., a long rise/fall time clock waveform), the CSEs energy consumption increases due to the stronger effect of the short-circuit currents and the current contentions between pull-up and pull-down networks [GLP97], [PS99]. Robustness problems are related to the risk of data-races (e.g., because of the contemporaneous transparency of Master and Slave latches) that can lead to the corruption of the information

stored in CSEs [H00]. The paper that most closely approaches the clock slope related issues is [LS94], but it is mainly focused on showing the robustness problems coming from "very" smooth clock edges.

Moreover, when dealing with the overall clock distribution network and in order to satisfy the skew/jitter requirement, the clock slope has to be properly set [VM95], [MCM97].

As a general rule of thumb, in high-speed designs the clock waveform is conservatively steep (i.e., short rise/fall time clock waveform) and the slope is preliminarily chosen regardless of the number of CSEs and their topology. Commonly adopted values of the clock slope range from $FO2$ to $FO3$ [O03], [OSM03], [GNO07], [MHA07], being FOX the slope of the output waveform of an inverter loaded by X inverters with the same size. In [LS94] it is stated that, in order to guarantee significant safety margins, the clock slope should be set to $FO2 \div FO4$. But, it is also stated that, with low supply voltages (approaching the value $V_{TH,n}+|V_{TH,p}|$), the clock slope related robustness issues become much less problematic. Indeed, basing on a 65-nm technology where the ratio V_{TH}/V_{DD} is significantly high, one does not encounter any malfunctioning problem in the wide slope range [$FO2 \div FO6$.

Anyhow, there are no previous works dealing with the clock slope optimization from the point of view of the joint energy dissipation of clock buffers and CSEs at the clock domain level, given that an inherent energy tradeoff arises among the two contributions [ACP10-1].

6.2 Setup to Simulate CSEs Under a Varying Clock Slope

Fig. 6.1 shows the simulation setup used to test a generic CSE under different clock slope conditions. The local clock, CLK, is generated by the clock buffer (i.e., an inverter gate) that is symmetrically sized, i.e. with the NMOS (PMOS) channel width equal to $w_{clk}W_{min}$ ($2w_{clk}W_{min}$), being w_{clk} the NMOS width normalized to the minimum size W_{min}.

In the scheme in Fig. 6.1, the required clock slope is tuned by setting the buffer size w_{clk} to obtain a clock slope equal to $FO2$, which is surely the minimum value that is adopted in real designs. Smoother clock slopes, FOX (with $X > 2$), are obtained by inserting and tuning the capacitance C_X, which slows down the signal CLK. The considered clock slope range is $FO2$ to $FO6$ with a step of 0.5, which is a wide range compared to typical values [O03], [OSM03], [GNO07], [MHA07]. The strategy to estimate CSEs energy and delay and to account for interconnects parasitics have been already discussed in Chapters 4-5.

The slope of the clock buffer output waveform (i.e., rise/fall time) can be represented by using the LE model, as shown in (6.1)

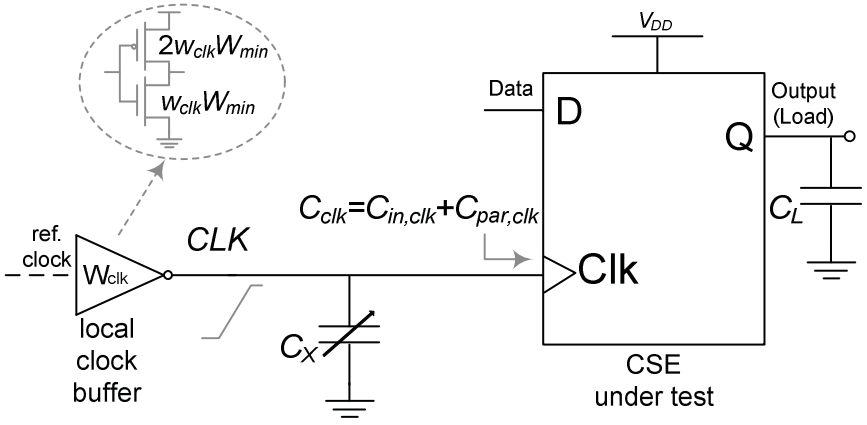


Fig. 6.1. Setup used to simulate CSEs under various clock slope values.

$$d = ghb + p = h + 1 \tag{6.1}$$

where $g = b = p = 1$ for an inverter. Accordingly, a given slope FOX, is achieved by simply setting $h = X$. From Fig. 6.1, h results to

$$h = \frac{C_{in,clk}+C_{par,clk}+C_X}{C_{inv}} \tag{6.2}$$

where it was considered that the external load of the clock buffer is the sum of the input (gate) capacitance $C_{in,clk}$ seen from the CSE clock terminal, the parasitic capacitance $C_{par,clk}$ due to local (internal to CSE) clock wires and the capacitance C_X (inserted to tune the clock slope as discussed above). From Fig. 6.1, $C_{inv} = w_{clk}C_{inv,min}$. On the other hand, $C_{in,clk}$ is set by the overall width of transistors within the CSE that are connected to the clock input terminal, namely $\sum_i w_{i,clk}$, where $w_{i,clk}$ is the normalized width of the generic i-th transistor within the CSE contributing to $C_{in,clk}$. Hence, $C_{in,clk}$ can be rewritten as the product of the input capacitance of a minimum-sized transistor (i.e., $C_{inv,min}/3$) and $\sum_i w_{i,clk}$, whereas h becomes

$$C_{in,clk} = \frac{C_{inv,min}}{3}\sum_i w_{i,clk} \tag{6.3}$$

$$h = \frac{\sum_i w_{i,clk}+3\frac{C_{par,clk}+C_X}{C_{inv,min}}}{3w_{clk}} \tag{6.4}$$

From (6.4), the buffer size w_{clk} ensuring a $FO2$ clock slope is obtained by setting $C_X = 0$ and $h = 2$ and solving for w_{clk}.

Higher values FOX of the clock slope (with $X > 2$) are then obtained by setting $h = X$ in (6.4) and solving for C_X, which yields

$$C_X = XC_{inv,min}w_{clk} - \frac{C_{inv,min}}{3}\sum_i w_{i,clk} - C_{par,clk} \quad (6.5)$$

6.3 CSEs Timing and Energy Versus Clock Slope

In this paragraph, a rather large number of CSE topologies is considered to emphasize that results are very general. In particular, Master-Slave (MS), Implicitly-Explicitly Pulsed (IP/EP), Differential and DET classes are considered as in Chapter 5 and the performances of the following topologies are analyzed under various clock slopes: TGFF, WPMS, GMSL, HLFF, USDFF, CPFF, SEPFF, TGPL, MSAFF, STFF, CCFF, DET-TGLM, DET-SPGFF, DET-SPL, DET-CDFF (see Chapter 5).

The considered CSEs were designed to minimize the energy-delay product $ED = E_{CSE}\tau_{DQ,min}$ (see Chapter 5) with a typical clock slope of $FO3$. The resulting $\tau_{DQ,min}$, $\tau_{CQ,min}$ and E_{CSE}, normalized to the case of $FO2$ clock slope, are plotted versus the clock slope FOX in Figs. 6.2-6.4. To improve the readability of these figures, solid lines, dashed lines and dotted lines are used for the SET-DET MS CSEs, SET-DET IP-EP CSEs and Differential CSEs, respectively [ACP10-1], [ACP09-2], [ACP09-3], [ACP09-4], [ACP10-6].

6.3.1 Impact of clock slope on $\tau_{DQ,min}$

From inspection of Fig. 6.2, the delay $\tau_{DQ,min}$ always increases for smoother clock slopes, as occurs in any CMOS logic style. However, this delay increase is rather modest and is always lower than 5.5%, except for the TGFF and DET-TGLM. Moreover, $\tau_{DQ,min}$ of Pulsed CSEs and in particular of the EP ones (SEPFF, TGPL, DET-SPL, DET-CDFF) exhibits very small increase even at $X = 6$ (in the order of $1 \div 2\%$). This is easily explained by considering that the transitions defining the optimum delay $\tau_{DQ,min}$ occur during the transparency window of these topologies [SO99] (i.e., after the clock transition), hence they are not affected by the clock waveform. This very small sensitivity of $\tau_{DQ,min}$ to the clock slope is a significant advantage in that it permits to relax the clock slope requirement while keeping essentially the same speed performance. This interesting property of Pulsed CSEs adds to the well-known soft-clock-edge property that makes $\tau_{DQ,min}$ insensitive to clock skew [OSM03], which again makes the design of clock network less critical. Interestingly, Differential CSEs (MSAFF, STFF, CCFF) show a similar behavior, as their operation resembles that of Pulsed single-ended topologies.

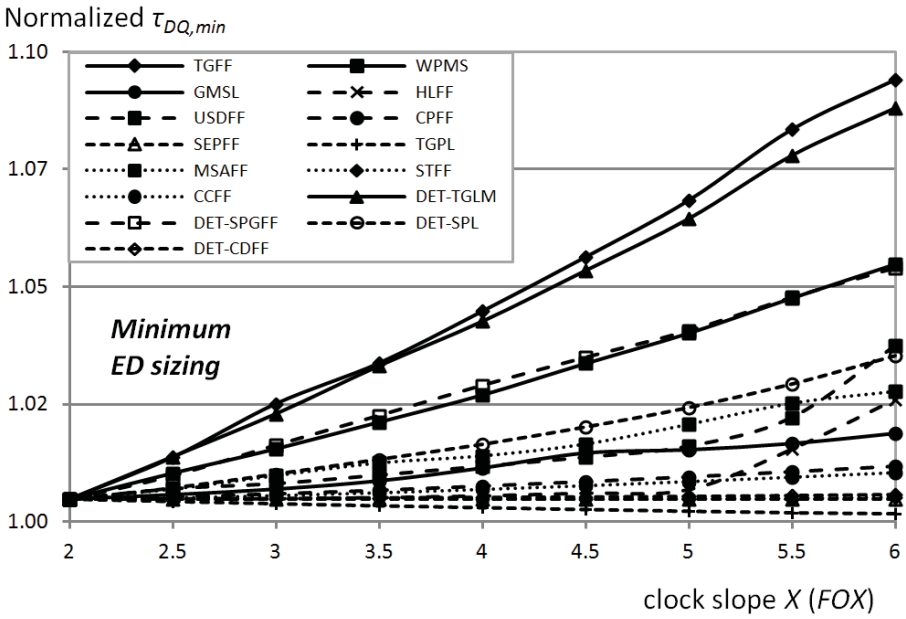


Fig. 6.2. Normalized (to $FO2$ case) $\tau_{DQ,min}$ vs. clock slope.

In other topologies, the above discussed decoupling effect is not present, since there is no transparency window. In particular, MS CSEs (TGFF, WPMS and DET-TGLM) are more sensitive to the clock slope compared to other topologies, due to the fact that the transmission gates enabling the input signal tend to turn-on slowly when the clock slope is smoother. Observe that the DET-SPGFF shows a sensitivity similar to that of WPMS, because its critical path exhibits the clock transition as the critical one. Instead, the GMSL, which is clock gated and MS at the same time, does not suffer from a significant speed performance degradation, since the clock gating circuitry tends to reduce the impact of the clock slope. It is interesting to observe that, in some IP topologies (USDFF and HLFF), $\tau_{DQ,min}$ has a rapidly increasing sensitivity for clock smoother than a $FO5$ one. This is because these CSEs have different critical paths that are nominally designed to have similar delays. However, for smoother clock slopes, the faster path (which may have a different sensitivity to clock slope) can become the slower one.

6.3.2 Impact of clock slope on $\tau_{CQ,min}$, t_{setup} and t_{hold}

The resulting $\tau_{CQ,min}$ is plotted in Fig. 6.3 versus the clock slope. Again, MS circuits suffer from the greatest degradation, together with some IP CSEs (HLFF, USDFF). In particular, the increase of $\tau_{CQ,min}$ is confined in

the range $3 \div 9\%$ ($6 \div 17\%$) at a clock slope $FO4$ ($FO6$). As expected, the degradation of $\tau_{CQ,min}$ due to a smoother clock slope is greater than the case of $\tau_{DQ,min}$ but it has the same order of magnitude. Hence, such degradation is sufficiently small and does not turn fast paths (where $\tau_{CQ,min}$ is the CSE speed parameter to be considered) into critical paths.

For completeness, t_{setup} and t_{hold} were also evaluated versus X.

Results show that t_{setup} has a low sensitivity to clock slope. For MS topologies (TGFF, WPMS, GMSL and DET-TGLM), t_{setup} is actually the Master delay and is nearly unaffected by the clock slope since the clock has already settled when the data input is applied to the Master. In particular, for MS topologies, the variations of t_{setup} in the $X = 4$, $X = 5$ and $X = 6$ slope cases with respect to the $X = 2$ one are equal to $(-0.15 \div 0.09)FO4$, $(-0.12 \div 0.13)FO4$ and $(-0.20 \div 0.25)FO4$, respectively.

Pulsed and Differential CSEs, which basically have a negative t_{setup}, experience even more negative t_{setup} values when increasing the clock slope. This can be easily understood by observing that, when the clock slope is smoother, the transparency window closes later in paths with the soft-clock-edge property (because the falling edge of the pulsed clock is less steep). Hence the data can be further delayed while still keeping a correct operation. For Pulsed and Differential CSEs, the variations of t_{setup} in the cases with $X = 4$, $X = 5$ and $X = 6$ with respect to the $X = 2$ one are equal to $(-0.02 \div -0.28)FO4$, $(-0.09 \div -0.44)FO4$ and $(-0.06 \div -0.52)FO4$, respectively, i.e. they are slightly negative.

Whereas the decrease of t_{setup} with X can be even advantageous in terms of possible time-borrowing, the dependence of t_{hold} on X needs a special attention since higher values of t_{hold} are critical for the system operation because of the possibility of races in fast-paths [OSM03], [RCN03].

MS CSEs (TGFF, WPMS and DET-TGLM) have a negative t_{hold} and, hence, are not critical from this point of view. Moreover, their t_{hold} further decreases with X and its variations in the $X = 4$, $X = 5$ and $X = 6$ slope cases with respect to the $X = 2$ one are $(-0.04 \div -0.20)FO4$, $(-0.01 \div -0.34)FO4$ and $(-0.04 \div -0.48)FO4$, respectively.

Regarding GMSL, it has a positive t_{hold} (because of the gating logic) and, hence, must be analyzed together with Pulsed and Differential CSEs, all of which exhibit a positive t_{hold}. For all these CSEs, the variations of t_{hold} in the $X = 4$, $X = 5$ and $X = 6$ slope cases with respect to the $X = 2$ one are equal to $(0.05 \div 0.21)FO4$, $(0.06 \div 0.27)FO4$ and $(0.08 \div 0.35)FO4$, respectively. Therefore, even in the extreme $FO6$ slope case, the t_{hold} increase is no more than one third of the $FO4$ delay, i.e. nearly 6ps.

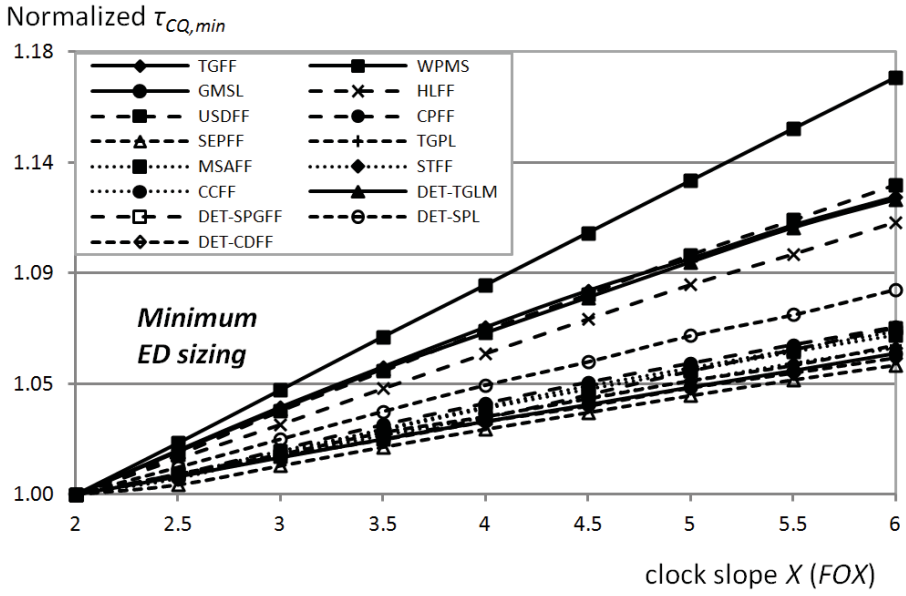


Fig. 6.3. Normalized (to $FO2$ case) $\tau_{CQ,min}$ vs. clock slope.

Nevertheless, the correct parameter that evaluates the immunity against races in fast paths is the well-known race immunity $R = \tau_{CQ} - t_{hold}$ [OSM03]. When R is positive, there is no race risk, whereas, when R is negative, it determines the minimum delay of logic to be inserted in fast paths. Hence, the dependence of R on X has also been analyzed. For all the considered CSEs except CPFF, TGPL, DET-SPL and DET-CDFF, parameter R increases with respect to the $X = 2$ case, thus leading to an even greater immunity towards races. The variations of R in the $X = 4$, $X = 5$ and $X = 6$ slope cases with respect to the $X = 2$ one are equal to $(0.01 \div 0.50)FO4$, $(0.02 \div 0.80)FO4$ and $(0.02 \div 1.09)FO4$, respectively. For the remaining four CSEs, the variations of R in the $X = 4$, $X = 5$ and $X = 6$ slope cases with respect to the $X = 2$ one are equal to $(-0.04 \div 0.03)FO4$, $(-0.06 \div 0.07)FO4$ and $(-0.10 \div 0.10)FO4$, respectively. Therefore, it is apparent that the race immunity loss is negligible (it is no greater than $0.1FO4 \approx 1.8$ps).

6.3.3 Impact of clock slope on E_{CSE} and operation robustness

The average energy per clock cycle E_{CSE} is plotted in Fig. 6.4 versus the clock slope for all CSEs. From this figure, E_{CSE} significantly increases as smoothing the clock slope, because of the growing impact of the short-circuit energy consumption. Again, MS circuits exhibit the worst degradation, as E_{CSE} increase up to more than 200% at clock slope $FO6$. All

the other analyzed CSEs show increment no greater than 70% at $FO6$ and between $15 \div 60\%$ at $X = 4 \div 5$.

From Fig. 6.4, it is clear that energy consumption almost linearly increases as smoothing the clock slope for all the considered topologies. Hence, it is possible to extrapolate a linear relationship

$$E_{FF} = aX + b \tag{6.6}$$

where X defines the clock slope FOX. The resulting parameters a and b in (6.6) are reported in Tab. VI.I for all CSEs, whose transistor sizing strategy is chosen to minimize the energy-delay products ED, ED^3 and E^3D.

Finally it is interesting to analyze any possible loss of robustness of the CSEs when the clock slope is smoothed [LS94]. Results showed that none of the considered topologies suffers from any malfunctioning problem in the range $[FO2 \div FO6]$. This means that the range $[FO2 \div FO6]$ permits a correct operation for all the considered topologies.

6.4 Energy of Local Clock Buffers Versus Clock Slope

Let us consider a generic clock domain consisting of a local clock buffer driving a number M of CSEs. As usual, the clock buffer is a tapered buffer

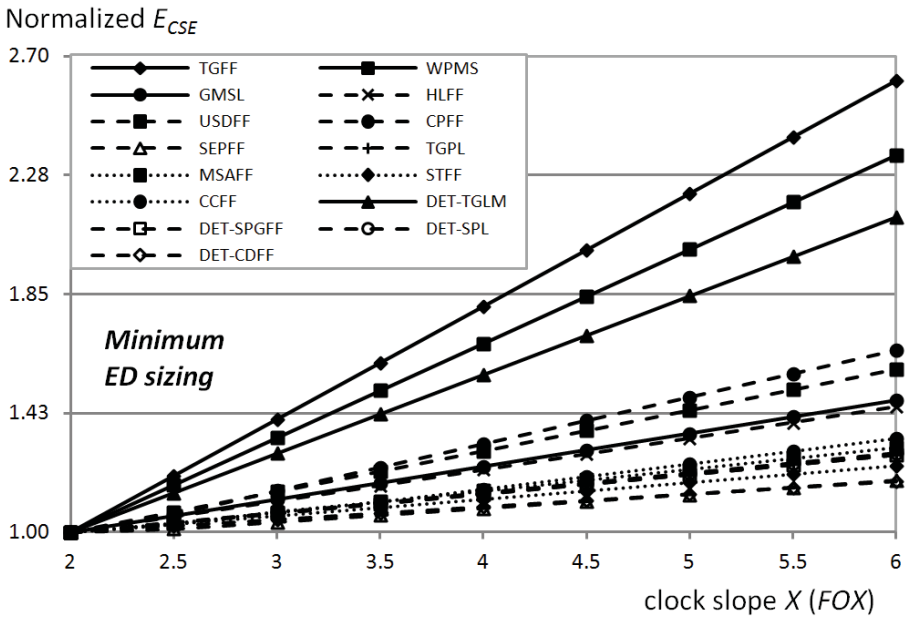


Fig. 6.4. Normalized (to $FO2$ case) E_{CSE} vs. clock slope.

TABLE VI.I: PARAMETERS a AND b (SIZINGS FOR MINIMUM ED, ED^3, E^3D)

	a (fJ)			b (fJ)		
Design	ED	ED^3	E^3D	ED	ED^3	E^3D
TGFF	1.361	3.788	1.358	0.643	2.427	0.459
WPMS	3.303	4.075	3.233	3.185	4.350	2.805
GMSL	1.250	1.650	1.240	8.000	10.400	7.460
HLFF	1.675	3.350	1.275	11.750	15.300	10.450
USDFF	3.300	4.125	2.075	16.000	19.650	11.550
CPFF	2.425	4.075	1.675	9.950	13.650	7.450
SEPFF	0.800	1.300	0.825	15.800	22.900	12.250
TGPL	1.275	2.075	0.800	15.550	23.150	12.000
MSAFF	1.225	2.050	0.825	13.350	21.700	9.600
STFF	1.225	2.000	0.800	18.650	36.500	15.600
CCFF	1.950	3.225	1.250	19.100	24.350	14.200
DET-TGLM	2.065	3.225	1.648	3.210	5.150	2.415
DET-SPGFF	0.825	2.675	0.393	10.350	22.950	7.945
DET-SPL	1.050	1.700	0.850	12.900	23.400	10.700
DET-CDFF	0.650	1.025	0.625	12.500	21.650	12.650

made up of N inverter stages, as shown in Fig. 6.5 [RCN03]. Let $C_{in\_buf,1}$ be the first stage's input capacitance, $C_{in\_buf,1} = jC_{inv,min}$, which is preliminarily assigned during the global clock network design. Since the clock buffer must provide a specified clock slope of FOX, each inverter must have an electrical effort $h = X$, i.e. the size of each inverter is greater than the previous one by a factor X. Hence, the input capacitance of the i-th inverter within the buffer, $C_{in\_buf,i}$, is

$$C_{in\_buf,i} = X^{i-1}C_{in\_buf,1} = X^{i-1}jC_{inv,min} \quad (i = 2 \dots N) \tag{6.7}$$

which is also the load driven by the $(i-1)$-th inverter, and the last inverter drives a load capacitance C_L given by [RCN03]

$$C_L = X^N C_{in\_buf,1} = X^N jC_{inv,min} \tag{6.8}$$

In general, C_L consists of the contribution $C_{M,CSE}$ associated with the input capacitances of the M driven CSEs and the contribution $C_{par,domain}$ due to the interconnections that distribute the clock signal throughout the clock domain

$$C_L = C_{M,FF} + C_{par,domain} \tag{6.9}$$

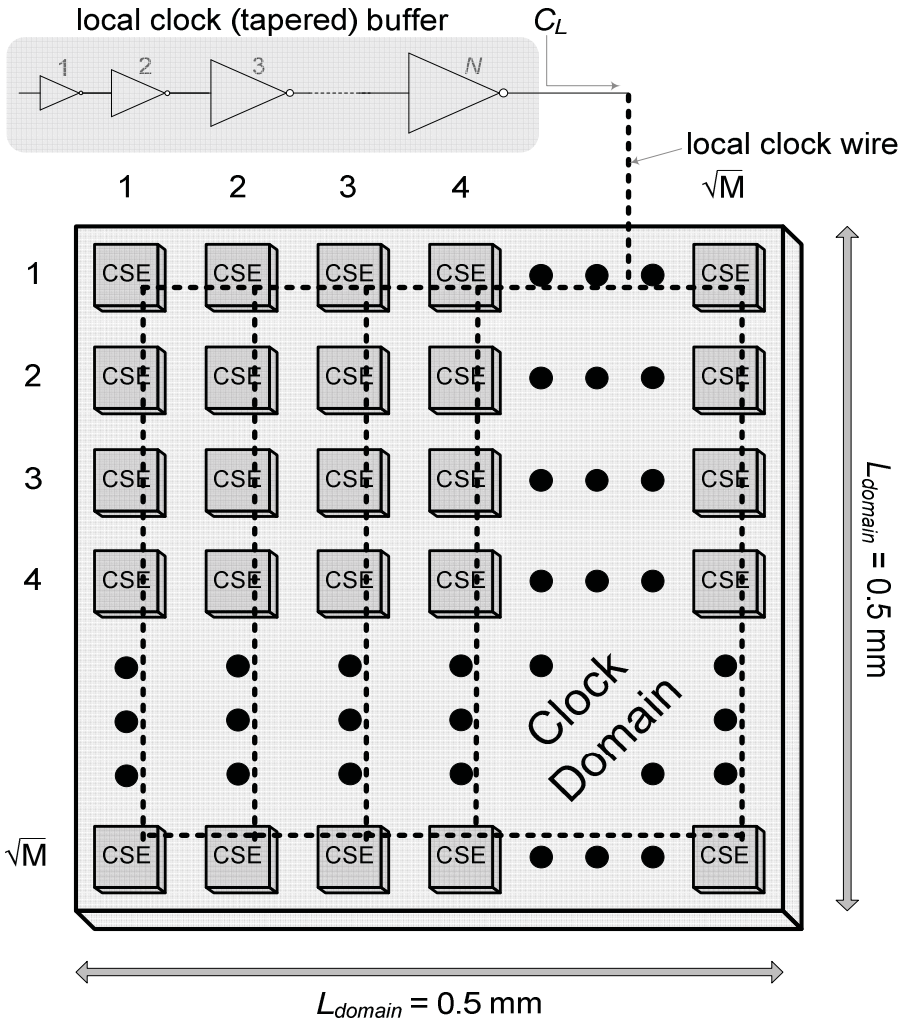


Fig. 6.5. Adopted scheme for the clock signal distribution in a clock domain.

The CSE input capacitance accounts for the transistor capacitance in (6.3) and the interconnect capacitance $C_{par,clk}$ connected to CLK within each CSE, hence one gets

$$C_{M,FF} = M\left(C_{par,clk} + \frac{C_{inv,min}}{3}\sum_i w_{i,clk}\right) \tag{6.10}$$

On the other hand, $C_{par,domain}$ depends on the placement of CSEs during synthesis. In the following, let us consider the realistic case of a square clock domain whose side length is $L_{domain} = 0.5\text{mm}$ [WH04], and assume that

CSEs are uniformly distributed, as in Fig. 6.5. Moreover, let us assume that the wires distributing the clock signal among the CSEs have the shape indicated by the dashed lines and are realized in Metal5 [WH04]. Since there are $\sqrt{M}$ CSEs for each side of the square clock domain, the parasitic lumped capacitance results

$$C_{par,domain} = c_{M5}\left(\sqrt{M} + 2\right)L_{domain} \tag{6.11}$$

being c_{M5} the capacitance for unit length of the Metal5 layer (equal to $100\ \mathrm{aF/\mu m}$ for the considered $65\mathrm{nm}$ CMOS technology) and the addendum 2 takes into account the horizontal edges in Fig. 6.6. It is worth noting that these assumptions are reasonable and do not lead to any loss of generality in the following analysis.

The energy consumption of the tapered buffers consists of the dynamic, static and short-circuit contributions [RCN03]. The dynamic energy can be evaluated by considering that in each period the i-th inverter dissipates an energy $C_{out,i}{V_{DD}}^2$ ($C_{out,i}$ is the overall capacitance at the output node of the i-th inverter). Thus, the dynamic energy of the tapered buffer is

$$E_{dyn,buf} = {V_{DD}}^2 \sum_{i=1}^{N-1} C_{out,i} \tag{6.12}$$

where $C_{out,i}$ includes the contribution of the extrinsic load capacitance $C_{in_{buf},i+1}$ in (6.7) and of the output parasitic capacitance $C_{par,i}$ of the i-th inverter itself. Considering that $C_{par,i} \approx C_{in\_buf,i}$ in real technologies [SSH98] and substituting the above expression of $C_{in\_buf,1}$, the dynamic energy consumption results to

$$\begin{aligned} E_{dyn,buf} &= \left[jC_{inv,min} + \left(\sum_{i=2}^{N} 2jC_{inv,min}X^{i-1}\right)\right]{V_{DD}}^2 = \\ &= \left(2jC_{inv,min}\frac{1-X^N}{1-X} - jC_{inv,min}\right){V_{DD}}^2 \end{aligned} \tag{6.13}$$

The tapered buffer static dissipation is due to the sum of the leakage currents of its N inverters, i.e.

$$I_{leak,buf} = jI_{Leak,min}\sum_{i=1}^{N} X^{i-1} = jI_{Leak,min}\frac{1-X^N}{1-X} \tag{6.14}$$

and hence the tapered buffer static energy consumption results to

$$E_{stat,buf} = T_{CK}V_{DD}I_{leak,buf} \tag{6.15}$$

Thus, the total energy dissipation of the tapered buffer is[1]

$$E_{buf} \approx E_{dyn,buf} + E_{stat,buf} \tag{6.16}$$

Relationship (6.16) was validated with extensive simulations under different conditions, with X ranging from 2 to 6 and for different numbers of inverters N. The results found are summarized in Tab. VI.II, where the percentage error is shown to be always lower than 14%, and typically lower than 8%. It is apparent that the error increases with X, since the buffers short-circuit currents are neglected. However, such increase is small (only a 7% factor from $FO2$ to $FO6$) and, hence, the assumption does not affect the results on the clock slope optimization carried out in the following.

TABLE VI.II: PREDICTED AND SIMULATED ENERGY OF A TAPERED BUFFER

X , N ($j = 16$)	Range of N	Average percentage error [%]
$X = 2$	$N = 1 \div 10$	-5.75
$X = 3$	$N = 1 \div 6$	-7.28
$X = 4$	$N = 1 \div 5$	-9.38
$X = 5$	$N = 1 \div 4$	-11.18
$X = 6$	$N = 1 \div 4$	-13.26

6.5 Design Considerations and Optimum Clock Slope

In Paragraphs 6.3 and 6.4 it has been shown that the CSE (local clock buffer) energy within a clock domain tends to increase (decrease) when smoothing the clock slope. Hence, the overall energy spent in clocking a clock domain (i.e., their sum) can be minimized by properly choosing the clock slope. This optimization is discussed in the following.

6.5.1 Analytical evaluation of the optimum clock slope

The energy consumption of the considered clock domain network can be computed from (6.6) and (6.16)

$$E_{TOT} = E_{dyn,buf} + E_{stat,buf} + ME_{FF} \tag{6.17}$$

where, since from (6.8)

$$N = log_X\left(\frac{C_L}{jC_{inv,min}}\right) \tag{6.18}$$

[1] (6.16) is multiplied by 2 for MS CSEs (driven by a pair of complementary clocks).

one can rewrite (6.13) and (6.15) as

$$E_{dyn,buf} = \left(2\frac{C_L - jC_{inv,min}}{X-1} - jC_{inv,min}\right){V_{DD}}^2 \tag{6.19}$$

$$E_{stat,buf} = T_{CK}V_{DD}jI_{Leak,min}\frac{\frac{C_L}{jC_{inv,min}}-1}{X-1} \tag{6.20}$$

By setting to zero the derivative of (6.17) with respect to the clock slope X and using (6.19)-(6.20), the optimum clock slope X_{opt} results to

$$X_{opt} = 1 + \sqrt{\frac{\left(2{V_{DD}}^2 + \frac{T_{CK}V_{DD}I_{Leak,min}}{C_{inv,min}}\right)\left(C_L - jC_{inv,min}\right)}{Ma}} \tag{6.21}$$

Relationship (6.21) can be further simplified considering that $C_L \gg jC_{inv,min}$ and neglecting the buffer leakage contribution with respect to its dynamic energy. It is worth noting that this is correct due to the very high buffer dynamic energy, since the clock node has the highest possible switching activity (i.e., 1), as was verified for the 65-nm adopted technology. Under these assumptions, substituting (6.9)-(6.11), (6.21) is simplified into

$$X_{opt} \approx 1 + V_{DD}\sqrt{2\frac{C_{par.clk} + \frac{C_{inv,min}}{3}\sum_i w_{i,clk} + \frac{\sqrt{M}+2}{M}c_{M5}L_{domain}}{a}} \tag{6.22}$$

Just as example, Fig. 6.6 depicts the simulated ME_{CSE} and E_{buf} of DET-TGLM sized for minimum ED, for $M = 128$. By inspection of Fig. 6.6, the optimum clock slope X_{opt} leading to the minimum total energy dissipation is 5. This value is close to the analytical value 5.19 given by (6.22).

6.5.2 Dependencies and typical optimum clock slope X_{opt}

To find the typical range of the optimum clock slope, (6.22) has been computed for the clock domain network with M equal to 64, 128, 256, and 512 (widely covering typical values of M), and considering all the CSE topologies listed at the beginning of Paragraph 6.3.

The CSEs were designed in the considered 65-nm CMOS technology by sizing the transistors to minimize ED, ED^3 and E^3D, and assuming a typical buffer input capacitance of $16C_{inv,min}$. Figs. 6.7a-b report the values of $\frac{C_{inv}}{3}\sum_i w_{i,clk}$ and $C_{par,clk}$ obtained from transistors optimization and layout.

The resulting optimum values of the clock slope are summarized in Figs. 6.8a-d. From these figures, the optimum clock slope can significantly depart

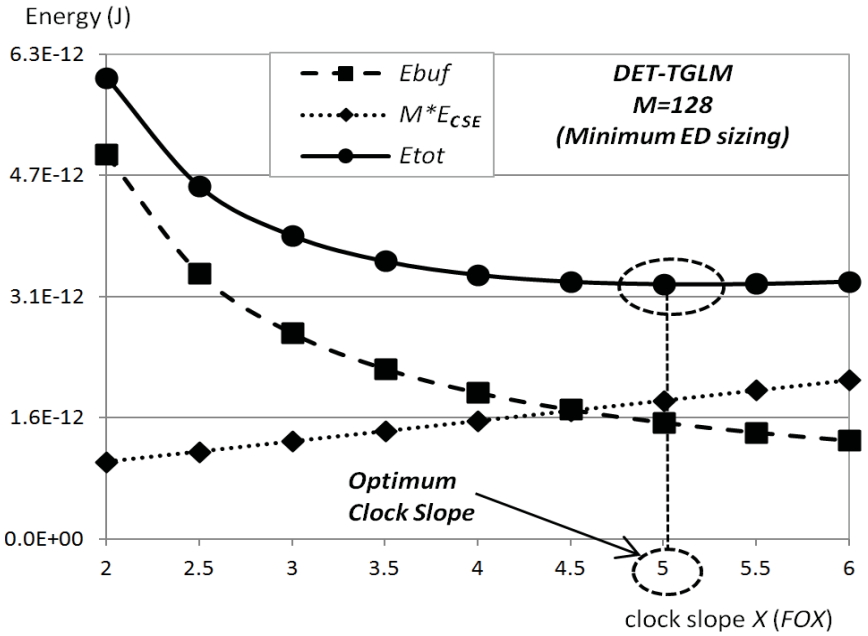


Fig. 6.6. ME_{CSE}, E_{buf} and E_{TOT} vs. clock slope (DET-TGLM, $M = 128$).

from the traditional high-speed design approach (i.e., an $FO2 \div FO3$ clock slope). Indeed, X_{opt} always reduces for increasing values of M, but it is always higher than 3, and can be as high as 5 or 6 (i.e., a clock slope of $FO5$ or $FO6$ is optimum).

The reduction of X_{opt} due to the increase in the number M of CSEs predicted by (6.22) can be intuitively explained as in the following. For a given clock slope, when M is increased, the energy dissipated by CSEs tends to dominate over the buffer energy. Hence, when M increases, the overall clocking energy can be more strongly reduced by reducing the CSE energy, and hence making the clock slope steep according to (6.6).

In other words, X_{opt} decreases, as expected. The same reasoning can be followed to justify why X_{opt} increases when increasing the clock input capacitances within the CSE (factor $C_{par.clk} + \frac{C_{inv,min}}{3}\sum_i w_{i,clk}$ in (6.22)) or the capacitance of the interconnections distributing the clock signal among CSEs (factor $c_{M5}\left(\sqrt{M}+2\right)L_{domain}$ in (22)).

Finally, let us evaluate the impact of the transistor sizing. By inspection of Figs. 6.8a-d, the optimum clock slope is approximately the same even when considering strongly different sizings. More specifically, greater transistor sizes (e.g., CSEs sized for minimum ED^3) in CSEs lead to a modest increase in the optimum X_{opt}.

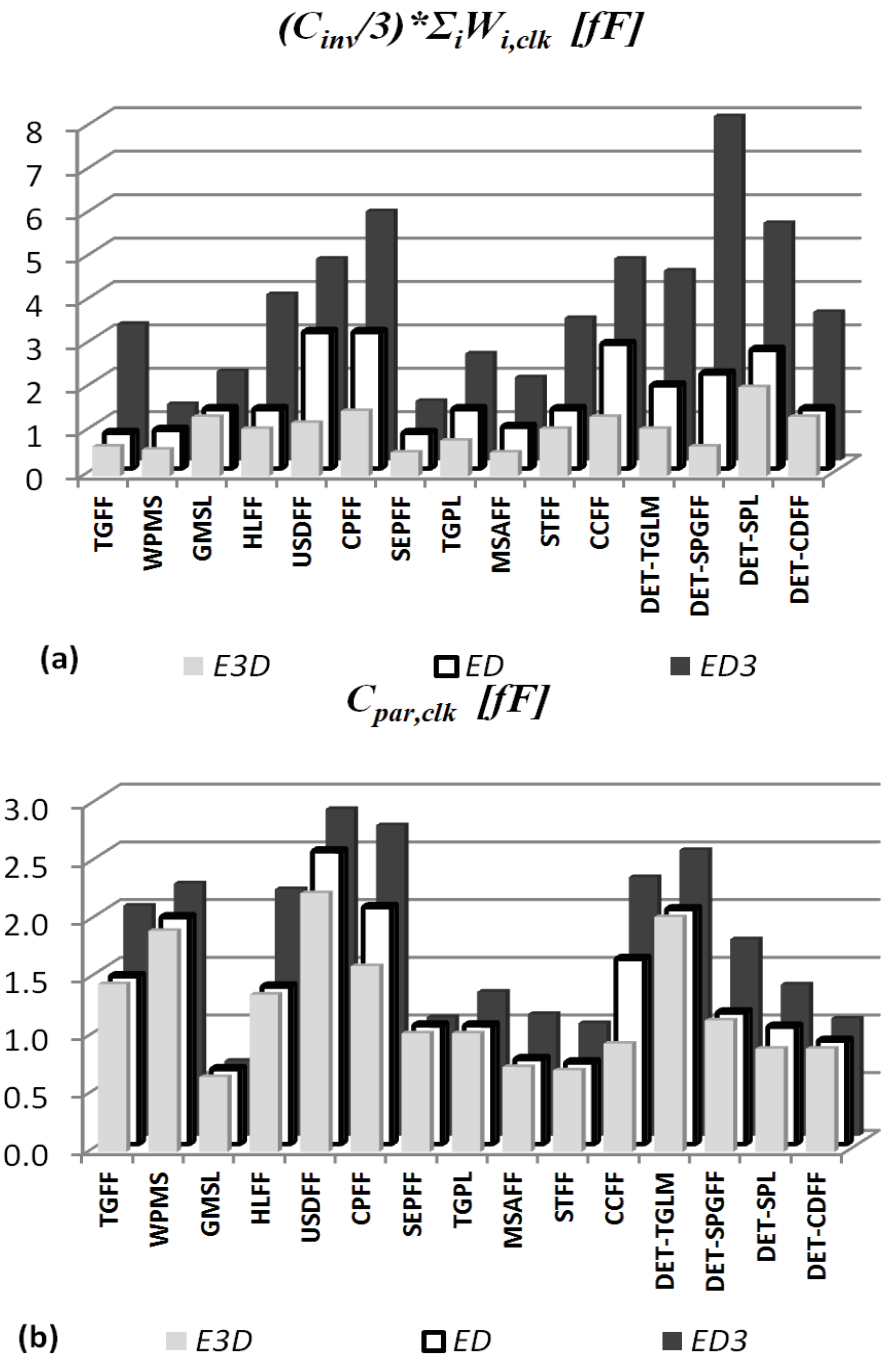


Fig. 6.7. Gate, $(C_{inv}/3)\sum_i w_{i,clk}$ (a), and local interconnections, $C_{par,clk}$ (b), clock capacitances for the analyzed CSEs.

6.5.3 Effectiveness of clock slope optimization and CSEs comparison

To comparatively evaluate the considered CSE topologies, let us consider the results in Figs. 6.8a-d relative to the minimum-ED sizing. Interestingly, within each CSE class, the optimum clock slope tends to be roughly the same for all topologies. In particular, the MS class (i.e., TGFF, WPMS, GMSL) shows an optimum $X_{opt} \in [4.5 \div 6]$ ($X_{opt} \in [3.5 \div 4.5]$) for $M \in [64 \div 128]$ ($M \in [256 \div 512]$).

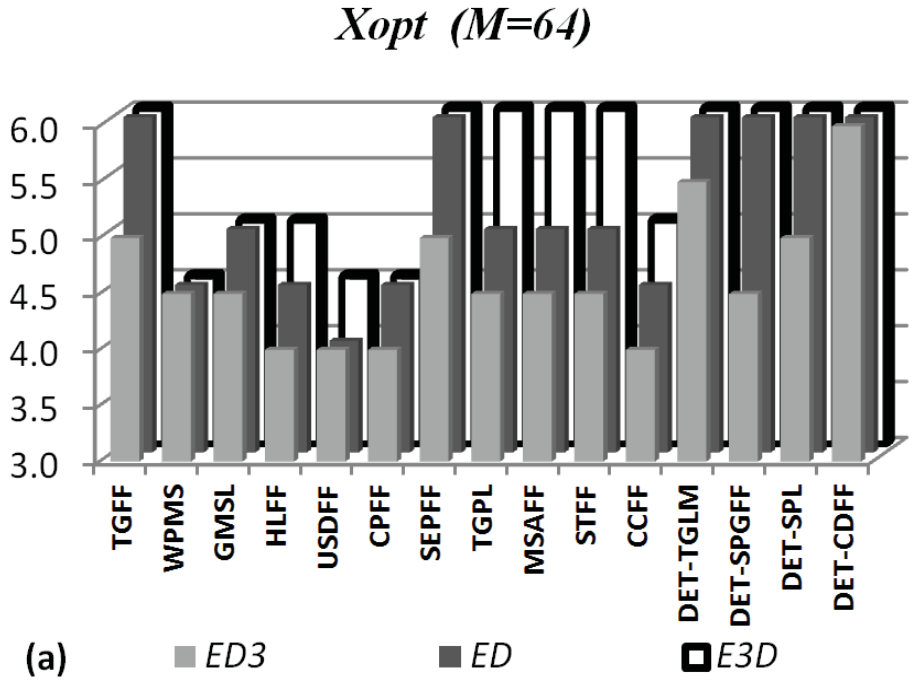
Similar results are obtained for the EP CSEs (i.e., SEPFF, TGPL) and the Differential topologies (MSAFF, STFF, CCFF). Instead, the IP CSEs (HLFF, USDFF, CPFF) have a somewhat steeper optimum clock slope (i.e. closer to the usually adopted values). On the other hand, DET CSEs exhibit a smoother optimum clock slope (typically, $X_{opt} \in [5.5 \div 6]$ ($X_{opt} \in [4.5 \div 5]$) for $M \in [64 \div 128]$ ($M \in [256 \div 512]$). By resuming, the optimum clock slope of MS, EP and Differential CSEs is significantly smoother than usual values, and even more in the case of DET CSEs. This means that, except for IP CSEs, the overall clocking energy can be reduced by properly relaxing the clock slope requirement.

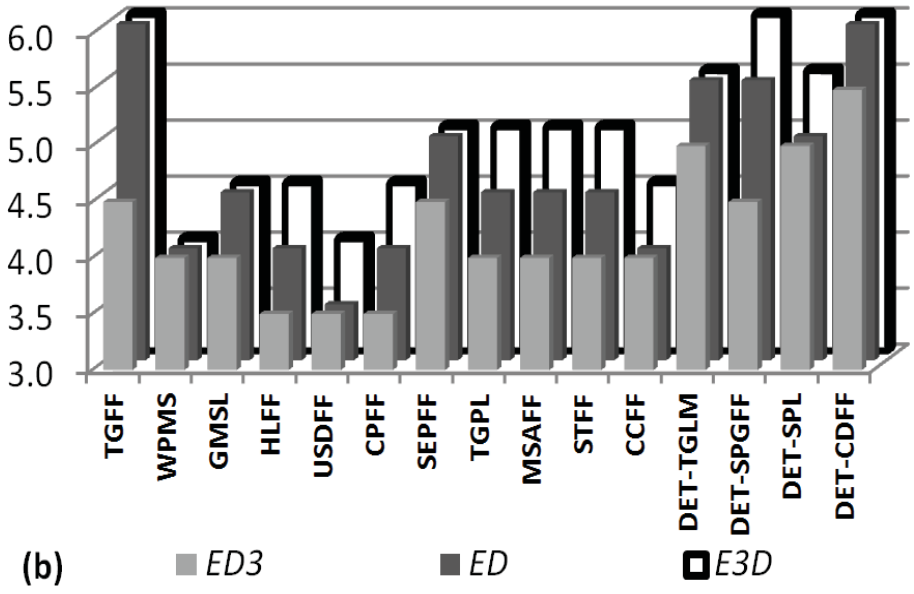
To evaluate the energy savings obtained by optimizing the clock slope, the overall clocking energy of a clock domain with 64, 128, 256 and 512 CSEs is plotted versus the clock slope in Figs. 6.9a-d, respectively. In these plots, the energy is normalized to the minimum value obtained for the optimum clock slope X_{opt}, with transistors sized to minimize ED.

From Figs. 6.9a-d, the adoption of the optimum clock slope permits an energy saving in the range $12 \div 235\%$ ($2 \div 40\%$), compared to the case with clock slope $FO2$ ($FO3$), when considering MS, DET and EP CSEs. On the other hand, lower energy savings are achieved in IP and Differential CSEs, which are in the range $12 \div 41\%$ ($1 \div 9\%$) compared to the case with clock slope $FO2$ ($FO3$). This confirms that the clock slope optimization is worthwhile from the point of view of energy efficiency, at the cost of a minor speed performance degradation as was discussed in Paragraph 6.3.

In regard to the energy sensitivity to the clock slope, in general, from Figs. 6.9a-d, it is apparent that it is high in the low-X region (which is far from the optimum X_{opt}), while there is a low sensitivity in the high-X for all topologies. In particular, DET CSEs have the highest sensitivity in the low-X region: this is explained by considering that the contribution of short-circuit energy contribution occurs once per cycle, as opposite to SET CSEs that experience two clock transitions per cycle (i.e., a doubled contribution due to short circuit current).

On the other hand, in the intermediate region for $X \approx X_{opt}$ a different behavior can be observed in different classes. Indeed, the Pulsed and Differential CSEs tend to be much less sensitive to the clock slope, compared to the other topologies: for example, a clock slope change from


Xopt (M=64)
6.0
5.5
5.0
4.5
4.0
3.5
3.0
TGFF
WPMS
GMSL
HLFF
USDFF
CPFF
SEPFF
TGPL
MSAFF
STFF
CCFF
DET-TGLM
DET-SPGFF
DET-SPL
DET-CDFF
(a)
ED3
ED
E3D


Xopt (M=128)
6.0
5.5
5.0
4.5
4.0
3.5
3.0
TGFF
WPMS
GMSL
HLFF
USDFF
CPFF
SEPFF
TGPL
MSAFF
STFF
CCFF
DET-TGLM
DET-SPGFF
DET-SPL
DET-CDFF
(b)
ED3
ED
E3D

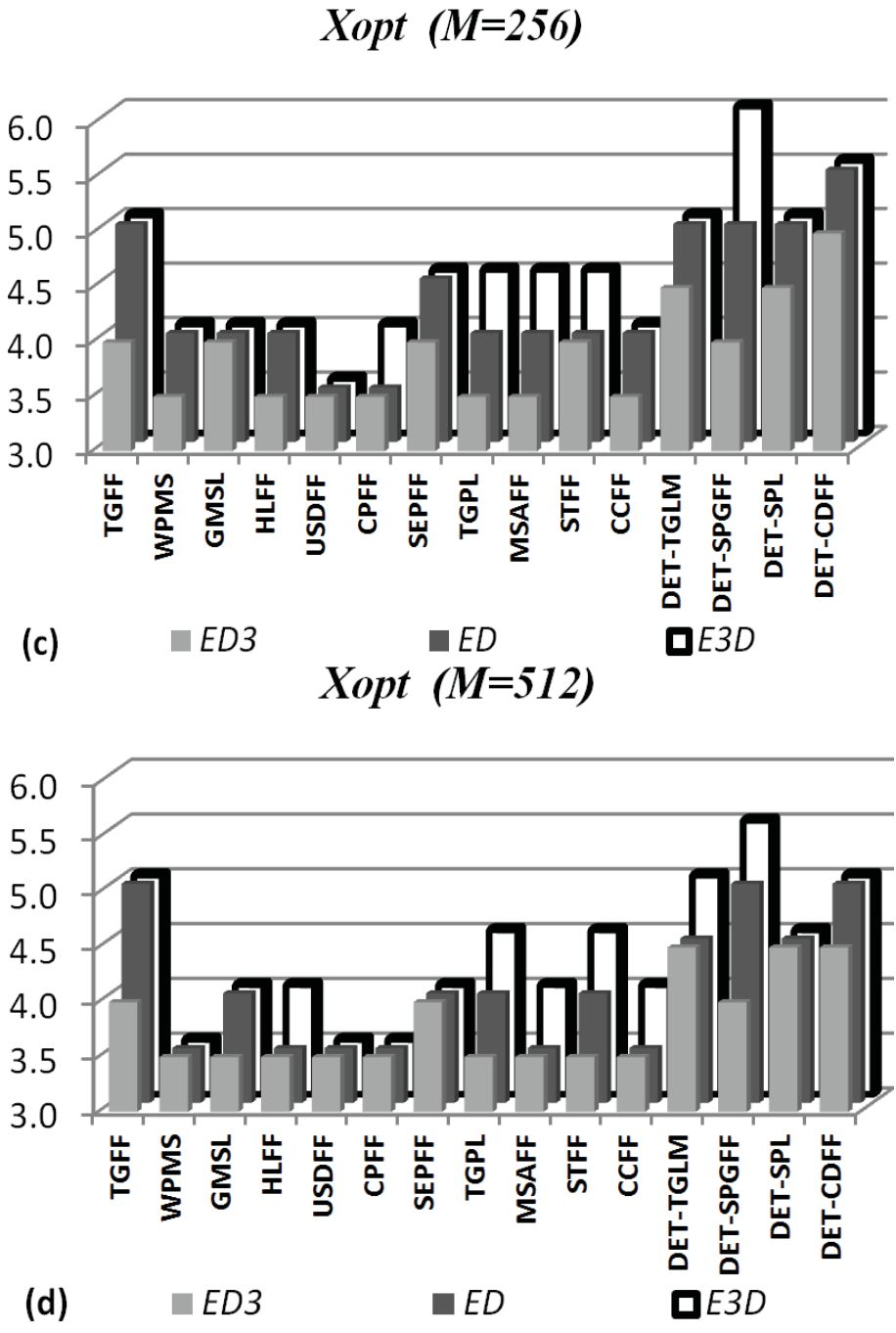
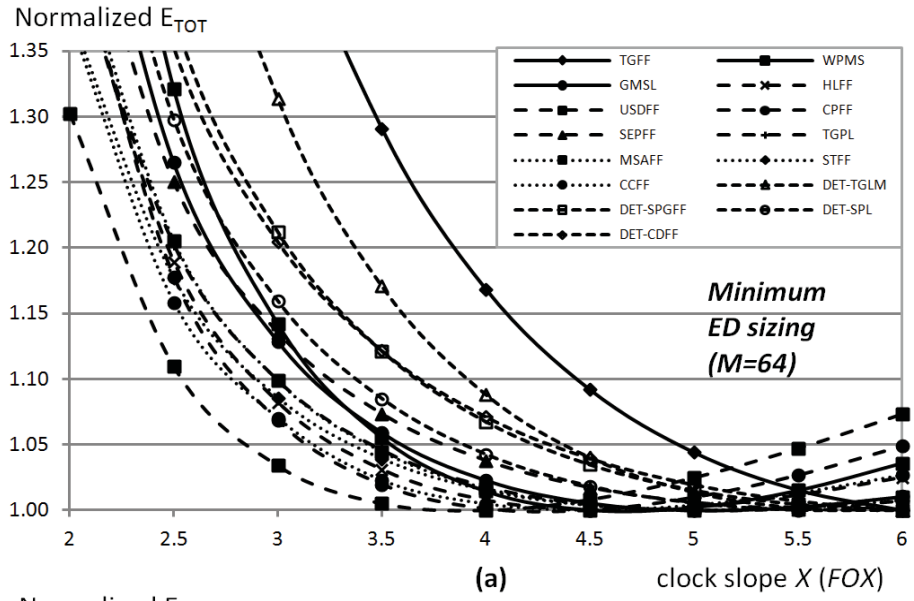
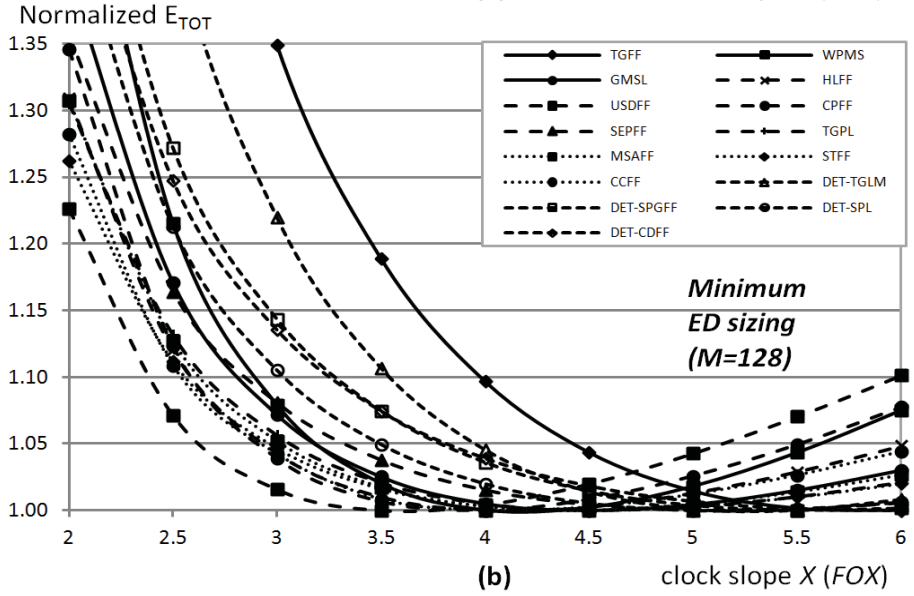


Fig. 6.8. Optimum clock slope X_{opt} under various M-values (64, 128, 256 and 512) (a-d) and sizings (minimum ED^3, ED and E^3D).

the optimum X_{opt} to $FO6$ determines only a 2.5% energy increase under $M = 512$. This means that Pulsed and Differential CSEs have essentially the same energy even in the presence of a significant clock slope degradation compared to the optimum.





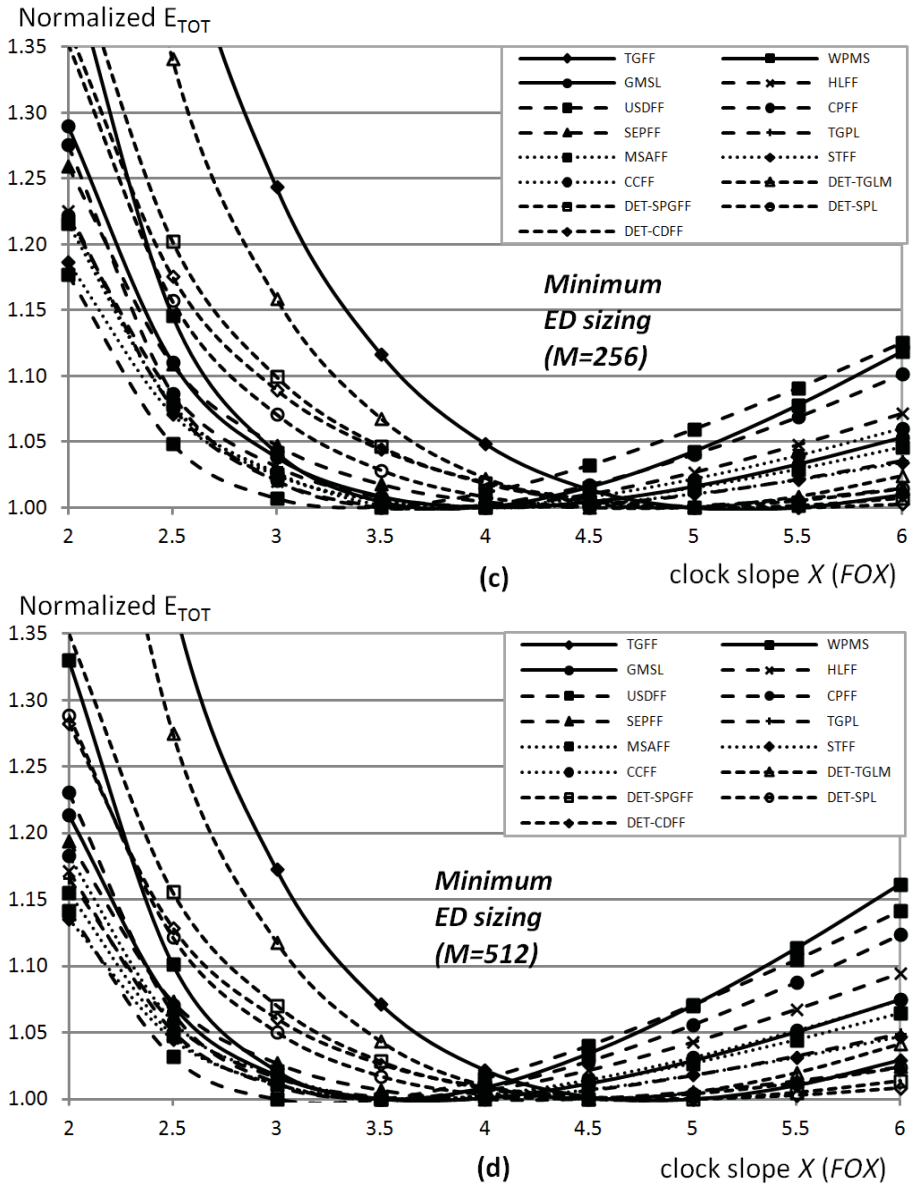


Fig. 6.9. Normalized E_{TOT} for all CSEs with $M = 64$ (a), $M = 128$ (b), $M = 256$ (c), $M = 512$ (d).

Hence, by recalling the considerations in Paragraph 6.3, energy and speed performance of Pulsed and Differential CSEs are almost independent of clock slope for values around optimum. Again this allows for relaxing the slope specification with no speed/energy penalty.

6.6 Impact of Clock Slope on Skew, Jitter and Variability

6.6.1 Additive skew and jitter due to a smoother clock slope

Traditionally, a smooth clock slope is not employed to avoid a potential increase in skew and jitter. Thus, these issues need to be addressed when the clock slope is changed from typical $FO2$ values up to $FO6$.

Since the investigation refers to the clock distribution network at the clock domain level, one has to evaluate the impact of the clock slope on the skew/jitter sources acting only on the local buffer when X changes within the range $[2 \div 6]$ [HN01]. In particular, jitter is mainly due to the supply voltage variations in the buffer driving the local clock, as well as to its capacitive coupling with other switching signals [RCN03], whereas clock skew is determined by the intradie process variations within the buffer [WH04].

The setup for jitter/skew analysis is described below:

- supply voltage noise is analyzed by referring to $\pm 5\,\%\ V_{DD}$ variations, and it is evaluated as semi difference (to refer to one-sided supply variations), $\Delta_{V_{DD}}$, between the buffer delay at $V_{DD} = 0.95$ and $V_{DD} = 1.05$;
- capacitive crosstalk is analyzed by considering a coupling capacitance $C_C = 20\text{fF}$ at the buffer output (equivalent to the coupling between two Metal5 wires with spacing double than minimum and running side by side for $400\mu\text{m}$). Crosstalk noise is measured as the buffer delay increment, Δ_{C_C}, when C_C is inserted to couple with an aggressor buffer (sized to drive the same load) switching exactly in the opposite direction of the buffer under analysis. Note that, in order to magnify the coupling effect, the aggressor is always $FO2$-sized;
- process variations are quantified through the standard deviation σ_D of the difference between the delays of two identical buffers subject to intradie variations (Montecarlo simulations with 2,000 samples are performed).

The above parameters $\Delta_{V_{DD}}$ and Δ_{C_C} (σ_D) are jitter (skew) contributions that add to those associated with the higher levels in the clock network hierarchy. In the following, these parameters are evaluated for buffers with $X > 2$ with respect to the reference case of a buffer with $X = 2$. Various loading conditions are explored, i.e. both the $X > 2$ buffers and the reference $X = 2$ buffer are loaded with $M = 64 \ldots 512$ CSEs, assuming three different values of the CSE capacitance seen from the clock terminal (i.e., 9fF, 5fF and 2fF).

The resulting $\Delta_{V_{DD}}$ is plotted in the scattering plot in Fig. 6.10, where the horizontal axis refers to the reference buffer sized for $X = 2$, whereas the vertical axis refers to the value of the considered buffers with $X > 2$ driving the same load. It is apparent that almost all the values lie below the bisector,

i.e., with respect to the $X = 2$ case, $\Delta_{V_{DD}}$ tends to diminish when $X > 2$. As concerns these results:

- on one hand, longer rise/fall times ($X > 2$) imply an increased sensitivity of delay to supply variations [AP06];
- on the other hand, by increasing X, the stages number N and the buffer delay decrease and hence, the absolute delay variations due to supply noise (which are a percentage of the nominal delay) decrease too.

The latter effect overcomes the former one and hence the impact of supply noise on jitter is magnified when the buffer delay increases (as expected, [HN01]). Therefore, buffers sized for $X > 2$ typically have a lower supply-related jitter contribution ($\Delta_{V_{DD}}$) compared to the case with $X = 2$.

The jitter contribution Δ_{C_C} is due to the delay variation observed when the buffer output is coupled with another buffer performing the opposite transition. The resulting Δ_{C_C} is plotted in the scattering plot in Fig. 6.11, where the horizontal axis refers to the reference buffer sized for $X = 2$, whereas the vertical axis refers to the value of the considered buffers with $X > 2$ driving the same load. From Fig. 6.11, almost all values lie above the bisector, hence buffers sized for $X > 2$ suffer from a greater jitter contribution Δ_{C_C}, compared to the case $X = 2$. This is easily explained by considering that buffers sized for $X > 2$ have a weaker final stage, hence they are more sensitive to the coupling with the aggressor. Nevertheless, it should be noted that the increase in Δ_{C_C} in Fig. 6.11 is typically lower than 1ps ($< 0.05\,FO4$), which is a rather small contribution. As a result, the degradation of the jitter contribution Δ_{C_C} due to the adoption of greater values of X is negligible in practical cases.

Regarding the skew contribution due to intradie process variations in the local buffer, the resulting σ_D is plotted in the scattering plot in Fig. 6.12, where the horizontal axis refers to the reference buffer sized for $X = 2$, whereas the vertical axis refers to the value of the considered buffers with $X > 2$ driving the same load. It was found that, even considering $3\sigma_D$ variations, the delay deviations of the buffers with $X > 2$ are always greater than those of the reference $X = 2$ buffer, but at most by $0.04\,FO4$. This means that buffers with $X > 2$ suffer from an increased delay variability due to process variations, which is easily explained by considering that higher values of X lead to a lower numbers of stages and smaller transistor sizes, both of which determine a higher variability [ONB08], [APP10]. Nevertheless, the increase in the skew contribution σ_D is again extremely small (it is a very small fraction of $FO4$), and hence it is negligible in practical applications.

Summarizing, the above results let us infer that the additive skew/jitter contributions in buffers sized for $X > 2$ are essentially the same as the case $X = 2$ (as in the case of Δ_{C_C} and σ_D), or even better (as in the case of $\Delta_{V_{DD}}$).

Similar results are found when comparing with the $X = 3$ case and hence are not reported for the sake of brevity. This confirms that the adoption of values of X greater than $2 \div 3$ is a feasible option at the clock-domain level, since no appreciable jitter/skew degradation is observed.

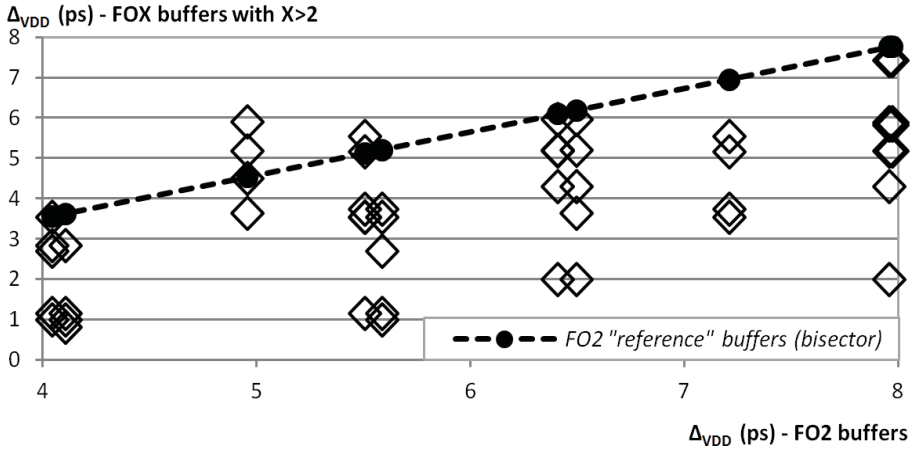


Fig. 6.10. $\Delta_{V_{DD}}$ of the reference buffers (with $X = 2$) and of the buffers with $X > 2$ driving the same load.

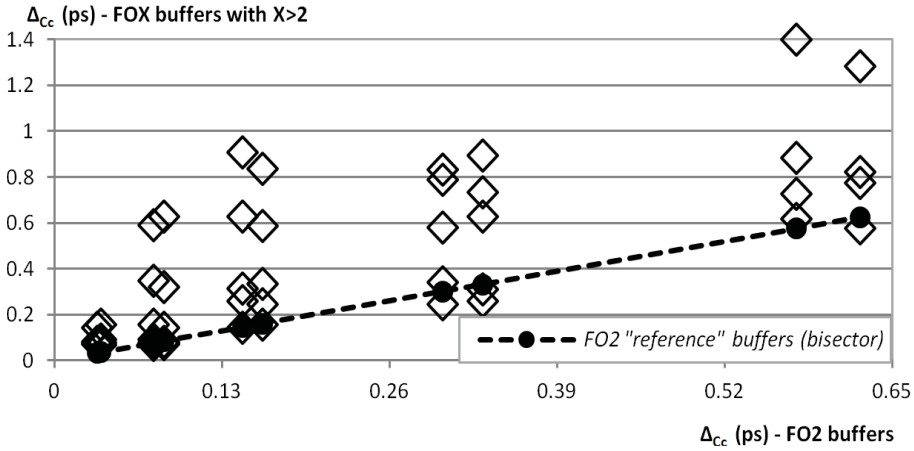


Fig. 6.11. Δ_{C_C} of the reference buffers (with $X = 2$) and of the buffers with $X > 2$ driving the same load.

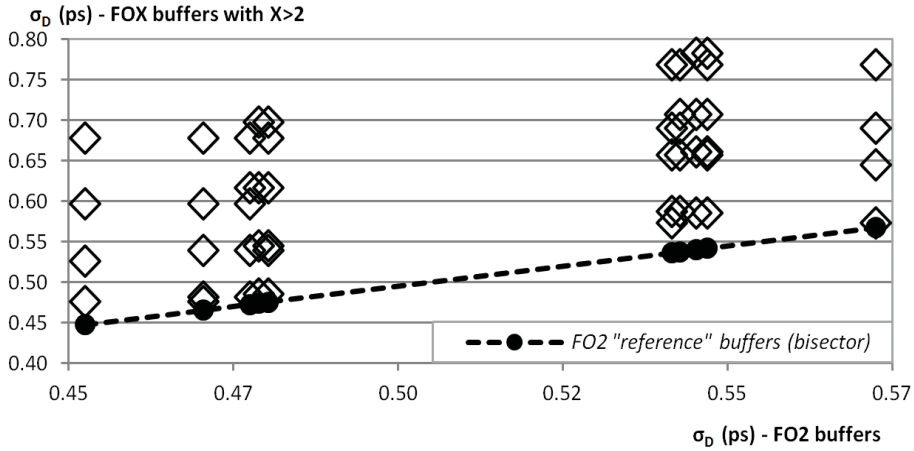


Fig. 6.12. σ_D of the reference buffers (with $X = 2$) and of the buffers with $X > 2$ driving the same load.

6.6.2 The impact of clock slope on CSEs delay variability

Interdie and intradie variations were considered in Monte Carlo simulations with $N = 2{,}000$ samples, in order to evaluate the dependence of CSEs delay variability on X. The variability σ/μ of both $\tau_{DQ,min}$ and $\tau_{CQ,min}$ delays is reported versus the clock slope X in Figs. 6.13 and 6.14, respectively. In these figures, all possible inputs transitions are considered, namely two for SET CSEs (data rising and falling) and four for DET CSEs (the two different clock transitions are also considered).

By inspection of Figs. 6.13 and 6.14, no particular trend of delay variability versus circuital topology emerges. Moreover, whereas for the $\tau_{CQ,min}$ an increase of the variability with X is observed, in the case of $\tau_{DQ,min}$ the behavior is much more irregular, as expected from the considerations in Paragraph 6.3. More specifically, the variations of $(\sigma/\mu)_{\tau_{DQ,min}}$ in the cases $X = 4$, $X = 5$ and $X = 6$ deviate from the case $X = 2$ by $(-5 \div 9)\%$, $(-5 \div 8)\%$ and $(-4 \div 12)\%$, respectively. The maximum increments of $(\sigma/\mu)_{\tau_{CQ,min}}$ in the $X = 4$, $X = 5$ and $X = 6$ slope cases with respect to the $X = 2$ one are equal to 5%, 8% and 11%, respectively. Hence, the quantitative impact of clock slope on delay variability is rather small compared to the clock cycle.

Moreover, the maximum increments of $\sigma_{\tau_{DQ,min}}$ in the $X = 4$, $X = 5$ and $X = 6$ slope cases with respect to the $X = 2$ one are equal to $0.040\,FO4$, $0.045\,FO4$ and $0.053\,FO4$, respectively. The maximum increments of $\sigma_{\tau_{CQ,min}}$ in the $X = 4$, $X = 5$ and $X = 6$ slope cases with respect to the $X = 2$ one are equal to $0.019\,FO4$, $0.029\,FO4$ and $0.042\,FO4$, respectively.

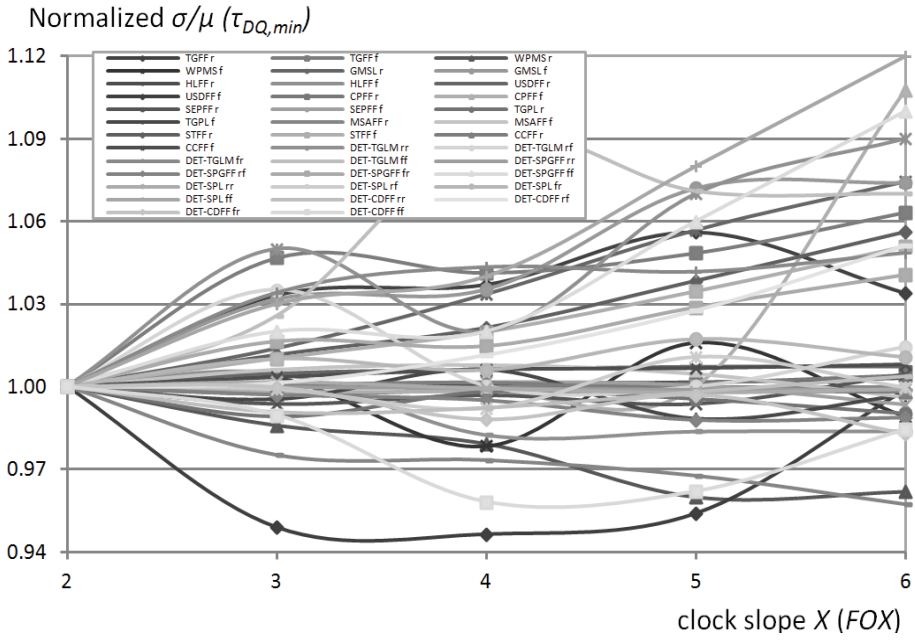


Fig. 6.13. Normalized $(\sigma/\mu)_{\tau_{DQ,min}}$ vs. clock slope.

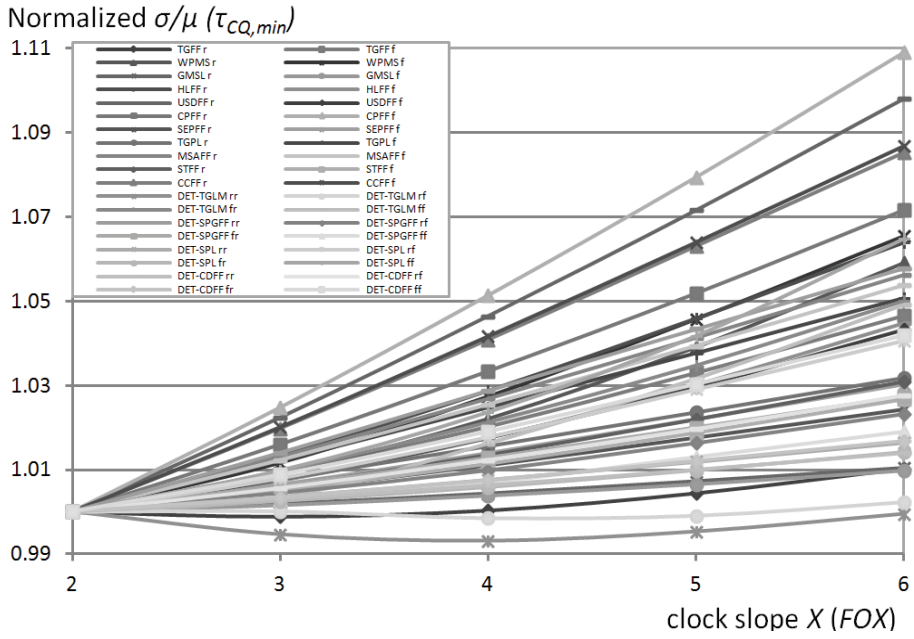


Fig. 6.14. Normalized $(\sigma/\mu)_{\tau_{CQ,min}}$ vs. clock slope.

Therefore, even considering 3σ variations with respect to the nominal CSE delay value, no more than $0.15\ FO4$ increment (i.e., less than 3ps) is observed in the extreme $FO6$ slope case.

6.7 The Impact of Technology Scaling

This paragraph deals with the technology scaling and its impact on the previous results. In Tab. VI.III the scaling factors on the parameters of interest for each new technology generation are reported. The fundamental scaling factors are highlighted in gray, while the other scaling factors can be derived from them. The scaling factors summarized in Tab. VI.III are taken from [KS95], [HMH01], [CSH03], [WMC05], [ITRS]. Moreover, an equal unchanged area for the clock domain is assumed, thus implying an increasing number of CSEs for each new generation.

It is worth noting that, from the herein adopted 65-nm technology node and below, the scaling factor on the clock frequency comes back to the value 1.4x, which is related only to technology scaling. One also needs to distinguish the scaling factors on the local wires' parasitic capacitances and on those related with the clock domain distribution.

By using all the aforementioned considerations, one can readjust all the parameters of interest, both for the CSEs and for the tapered buffer. In particular, one can directly refer to the value of X_{opt} expressed by formula (6.21). Let us also define s as an integer that counts the future technology generations with respect to the 65-nm technology node (referred to as $s = 0$; for example, 45-nm node has $s = 1$). Once the assumed scaling factors have been applied, one get (6.23), where the approximation holds since the capacitive loading contributions brought by each CSE (gate input and local/global interconnections parasitic capacitances), have the same order of magnitude and their scaling factors are very similar and close to the value 0.68x.

Relationship (6.23) allows to infer that, almost independently of the CSE parameters and depending on the proportion between the dynamic and static energy consumptions in the tapered buffer, the optimum clock slope moves towards smoother values with technology scaling or substantially remains the same. For example, by assuming $T_{CK} = 40\ FO4$ and a buffer input capacitance equal to $16C_{inv,min}$, a number of CSEs $M = 128$ in the 65-nm case and assuming that the transistor sizes in CSEs scale according to the minimum feature size, the results reported in Tab. VI.IV are found (the minimum ED design is considered).

In conclusion, the increase of X_{opt} with the technology scaling is mainly due to the increasing impact of leakage energy, which is independent on the clock slope and leads to a slighter relative increment of E_{CSE} with X.

TABLE VI.III: ADOPTED SCALING FACTORS TO MODEL NEXT TECHNOLOGY GENERATIONS

Parameter of interest	***Scaling Factor***	
Clock Domain Area (mm^2)	$S_{CD,A}$	1.00x
Dimension (nm)	S_D [43]	0.70x
Supply Voltage (V)	S_V [44]	0.85x
Leakage Current per µm Width (nA/µm)	$S_{I_{L/\mu m}}$ [45]	4.00x
Frequency (GHz)	$S_{F_{CK}}$ [46]	1.40x
C per Unit Length–Metal Layers (aF/µm)	$S_{C_{M/\mu m}}$ [47]	0.95x
Number of Flip-Flops in the Clock Domain	$S_{N_{FF}} = {S_D}^{-2}$	2.04x
Leakage Current (nA)	$S_{I_L} = S_D * S_{I_{L/\mu m}}$	2.80x
C – Gate and Junction (aF)	$S_C = S_D$	0.70x
C - Local (Flip-Flop) Interconnects (aF)	$S_{C_{I,FF}} = S_D * S_{C_{M/\mu m}}$	0.66x
C - Global (Domain) Interconnects (aF)	$S_{C_{I,CD}} = S_{CD,A} * S_{C_{M/\mu m}}$	0.95x
Dynamic Energy - Gates (fJ)	$S_{ED_G} = S_C * {S_V}^2$	0.51x
Dynamic Energy – Flip-Flop Interconnects (fJ)	$S_{ED_{I,FF}} = S_{C_{I,FF}} * {S_V}^2$	0.48x
Dynamic Energy – Domain Interconnects (fJ)	$S_{ED_{I,CD}} = S_{C_{I,CD}} * {S_V}^2$	0.69x
Static Energy (fJ)	$S_{ES} = S_{I_L} * S_V * {S_{F_{CK}}}^{-1}$	1.70x

TABLE VI.IV: OPTIMUM CLOCK SLOPE AT VARIOUS TECHNOLOGY GENERATIONS

X_{opt} - Minimum ED sizing $T_{CK}/FO4 = 40$, $j = 16$, $M = 128$ at 65-nm				
	$s = 0$ (65-nm)	**$s = 1$ (45-nm)**	**$s = 2$ (32-nm)**	**$s = 3$ (22-nm)**
TGFF	5.84	5.84	6.17	7.38
WPMS	4.22	4.22	4.44	5.25
GMSL	4.51	4.51	4.76	5.66
HLFF	4.18	4.18	4.41	5.21
USDFF	3.64	3.65	3.86	4.56
CPFF	4.01	4.03	4.27	5.07
SEPFF	5.34	5.34	5.63	6.71
TGPL	4.57	4.57	4.82	5.73
MSAFF	4.47	4.47	4.70	5.57
STFF	4.56	4.56	4.82	5.72
CCFF	4.24	4.26	4.51	5.37
DET-TGLM	5.34	5.34	5.67	6.79
DET-SPGFF	5.69	5.70	6.06	7.28
DET-SPL	5.25	5.27	5.61	6.73
DET-CDFF	5.95	5.95	6.31	7.56

$$X_{opt} = 1 + \sqrt{\frac{\left((0.85^{2s})2V_{DD}{}^2 + \frac{T_{cK}}{(1.4^s)}\frac{(0.85^s)V_{DD}(2.80^s)I_{leak,inv}}{(0.70^s)C_{inv,min}}\right)\left((0.66^s)C_{par.clk} + \frac{(0.70^s)C_{inv,min}}{3}\sum_i w_{i,clk} + (0.95^s)c_{M5}\frac{\sqrt{(2.04^s)M}+2}{(2.04^s)M}L_{domain} - \frac{j(0.70^s)C_{inv,min}}{(2.04^s)M}\right)}{(0.51^s)a}}$$

$$\approx 1 + \sqrt{(1.33^s)\frac{\left((0.72^s)2V_{DD}{}^2 + (2.43^s)\frac{T_{cK}V_{DD}I_{leak,inv}}{C_{inv,min}}\right)\left(C_{par.clk} + \frac{C_{inv,min}}{3}\sum_i w_{i,clk} + c_{M5}\frac{\sqrt{M}+2}{M}L_{domain} - \frac{jC_{inv,min}}{M}\right)}{a}} \quad (6.23)$$

Chapter 7

NOVEL ULTRA-FAST AND ENERGY-EFFICIENT PULSED LATCH TOPOLOGIES

In this chapter, two novel pulsed latch clocked storage element (CSE) topologies are discussed [CAP12]. The operation of these circuits, featured by a very fast push-pull second stage and a conditional data-dependent pulse generator, is first presented. A chip prototype in 65nm ST-CMOS065 technology has been developed and the integrated test circuits to measure the energy and delay performances of the proposed CSEs are discussed. Finally, results extracted through on-silicon measurements are reported. In their minimum ED^3 (ED) product sizing the novel CSEs achieve $0.7FO4$ ($0.8FO4$) minimum data-to-output delay and consume 70fJ (40fJ) transient energy per clock cycle at 25% data switching activity. Compared to state of the art energy-efficient Transmission-Gate Pulsed Latch, they exhibit 2.3X (1.3X) lower ED^3 (ED) product, thereby proving to be the fastest and most energy-efficient (in the high-speed design region) CSE topologies ever proposed [CAP12].

7.1 State of the Art and Preliminary Considerations

From the thorough investigation carried out in Chapter 5 [ACP11-1], [ACP11-2], the Transmission-Gate Pulsed Latch (TGPL) [NH02], [NCF02] clearly proves to be the most energy-efficient CSE in a very wide design frame ranging from high-speed designs (minimizing E^iD^j products with $j > i$) to minimum ED product designs, while transmission-gate based Master-Slave (MS) CSEs like Transmission-Gate Flip-Flop (TGFF) [GGD94], [MNB01] or Adaptive Coupling Flip-Flop (ACFF) [TFH11] are

the most energy-efficient in the low-energy region. Overall, TGPL has the lowest $\tau_{DQ,min}$, together with Skew-Tolerant Flip-Flop (STFF) [NOW03], which, however, exhibits a much worse energy-efficiency.

Here, two novel CSEs are introduced, namely the Conditional Push-Pull Pulsed Latch (CP^{3}L) and its version with a Shareable (CSP^{3}L) Pulse Generator (PG) [CAP12]. The adoption of a fast push-pull second stage, which requires the employment of a conditional data acquisition technique in the PG, enable 50% to 100% delay improvements compared to TGPL and absolute $\tau_{DQ,min}$ up to $0.7FO4$. In despite of a certain energy dissipation increment, CP^{3}L and CSP^{3}L exhibit a superior energy-efficiency compared to TGPL, both in terms of minimum ED^3 and ED products.

7.2 Operation of the Novel CP^{3}L and CSP^{3}L

The schematics and the operation of the proposed CSEs and TGPL are shown in Fig. 7.1.By using TGPL topology as a reference, in CP^{3}L the data-to-output path is broken into two parallel ones capturing data rising and falling transitions, respectively. The output inverter in TGPL is replaced by a push-pull stage, the first inverter plus the transmission-gate are replaced by two half latches and the gated keeper is moved to the output.

Since the push-pull stage is more prone to current contention than a simple inverter, only one among the internal nodes S_{neg} and R (equal to $'1'$ and $'0'$ in steady state) has to be enabled in each clock cycle depending on the previously stored Q value thanks to the usage of a conditional PG. Indeed, the pulsed signals CP_r (rising) and CP_f (falling) are alternately enabled by employing Pseudo-NOR/NAND that are gated by a delayed version of the output, Q_D. If $Q_D = '1'$ ($'0'$), CP^{3}L can change its state if $D = '0'$ ($D = '1'$) or not if $D = '1'$ ($D = '0'$). In any case, the Pseudo-NOR (Pseudo-NAND) does not change its output CP_r (CP_f) and hence also the internal node S_{neg} (R) does not make any transition. On the contrary, the Pseudo-NAND (Pseudo-NOR) is enabled and generates a pulse on CP_f (CP_r). Anyway, if D remains equal to $'1'$ ($'0'$), also the node R (S_{neg}) does not change its state.

It is worth noting that Q_D has to be sufficiently delayed in order to avoid the undesired enabling of the Pseudo-NOR (Pseudo-NAND) when the previously stored output is equal to $'1'$ ($'0'$). Indeed, if this happened, an undesired transition on CP_r (CP_f) would occur, i.e. there would be more energy dissipation. However, even in such case, this would not affect the correct operation, given that the input D has anyway to remain stable up to the end of the transparency window to avoid an hold time violation. Such a discussion on the delay from Q to Q_D is indeed closely related with the hold

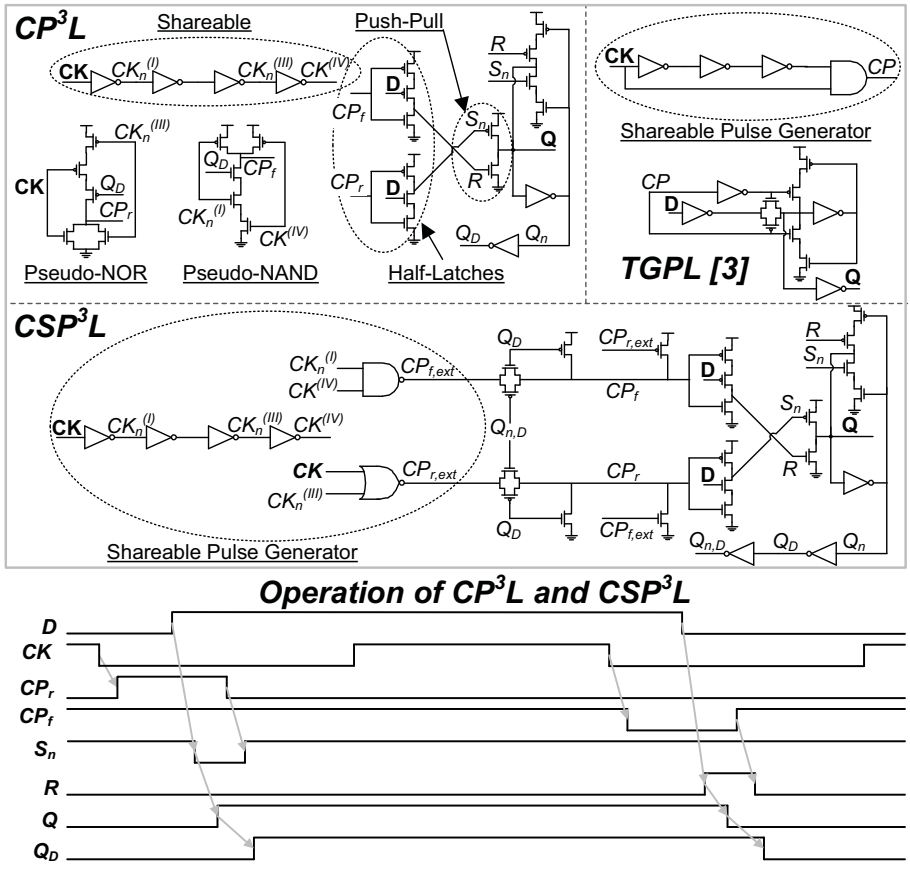


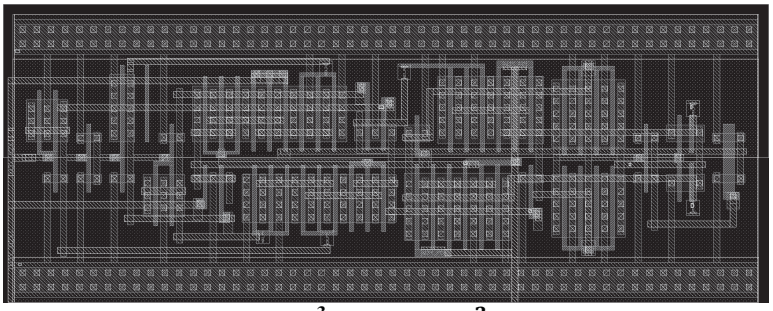
Fig. 7.1. Schematics and operation of CP^{3}L, CSP^{3}L and TGPL.

time characteristics of the novel CSEs and is later resumed in Paragraph 7.4.

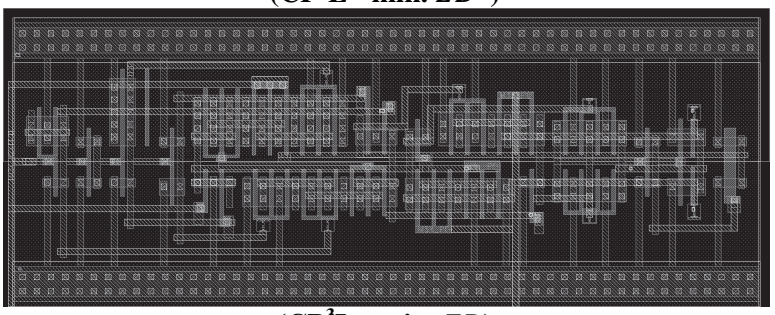
As concerns setup characteristics, as in TGPL, the inverters chain in the PG defines a freely adjustable transparency window allowing clock skew absorption and time borrowing [NH02], [NCF02], [OSM03], [GNO07].

CP^{3}L does not allow the sharing of the entire PG (only the four inverters can be shared) given that Pseudo-NOR/NAND are driven by Q_D which is different for each latch. CSP^{3}L solves this issue by employing standard NAND/NOR gates and by fully integrating the conditional logic in the latch. In such a case the whole PG (i.e., including NAND/NOR) can be shared among several CSEs. Two transmission gates and few small keepers have to be added at the two pulsed nodes to achieve the same operation as before (also an inverted version of Q_D is needed). Therefore, the advantage of the full PG sharing comes at the cost of a slightly increased latch complexity.

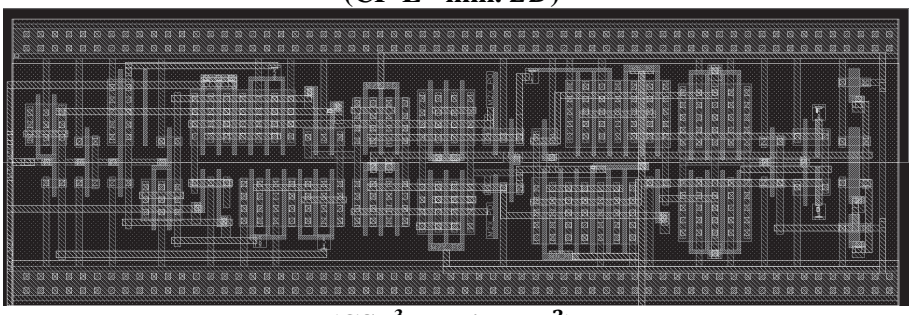
The layouts of the integrated sizings (minimum ED^3 and ED, see next paragraph) for CP^{3}L, CSP^{3}L and TGPL are shown in Fig. 7.2.



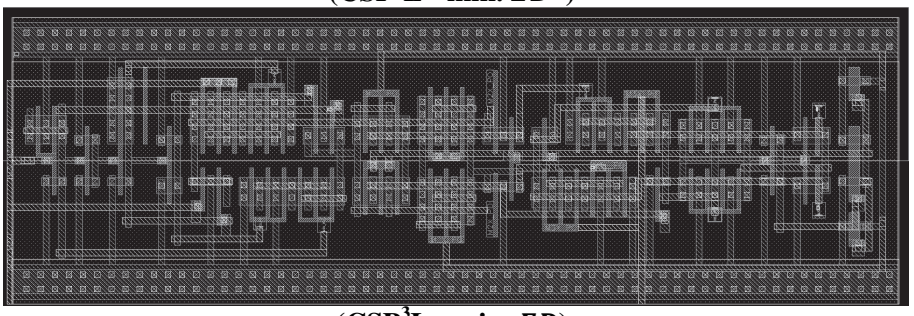
(CP^{3}L - min. ED^3)



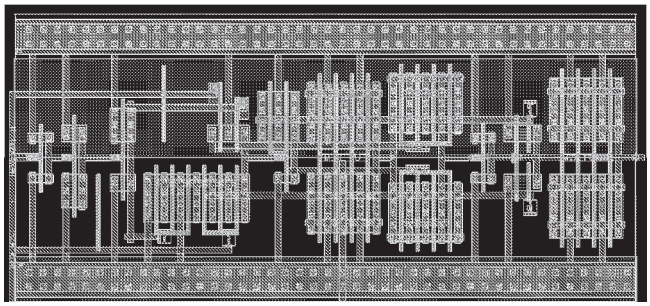
(CP^{3}L - min. ED)



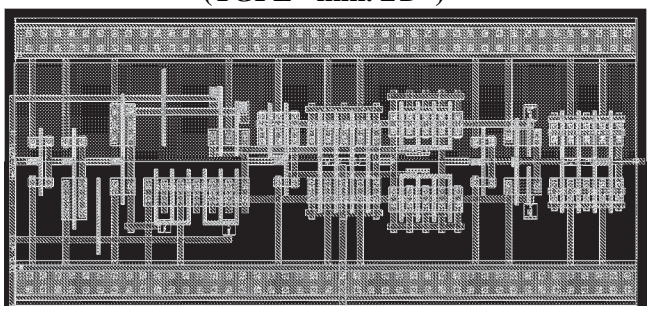
(CSP^{3}L - min. ED^3)



(CSP^{3}L - min. ED)



(TGPL - min. ED^3)



(TGPL - min. ED)

Fig. 7.2. Layouts of the integrated CSEs (min. ED^3 and ED sizings).

7.3 Test Chip and Circuits for Delay-Energy Measurement

Given the expected high performances of the proposed CSEs, a test circuit has been fabricated in 65-nm STCMOS065 technology (GP option at $VDD = 1\text{V}$). This activity has been carried out in collaboration with the Berkeley Wireless Research Center (BWRC), UC Berkeley. The test circuit is included within a chip containing other circuits to demonstrate the feasibility of some novel ultra-low power techniques, which however are out of the scope of this work and hence are not here discussed. The chip micrograph and the corresponding layout are shown in Figs. 7.3-7.4, while the micrograph of the test circuit to characterize the novel CSEs and its layout are shown in Figs. 7.5-7.6. As shown in Fig. 7.6, the test circuit is basically split in two blocks, one for delay (and in general timing characteristics) measurement and the other for (transient) energy measurement. Apart from CSEs arrays and some control logic in both blocks, it is worth highlighting that an overall 2.35nF on-chip capacitance (made up of MIM and MOS capacitors) has been used to filter resistive and inductive voltage drops due to interconnections and bonding parasitics.

Given that, as previously discussed, TGPL constitutes an essential reference for comparison and in order to achieve meaningful and fair results

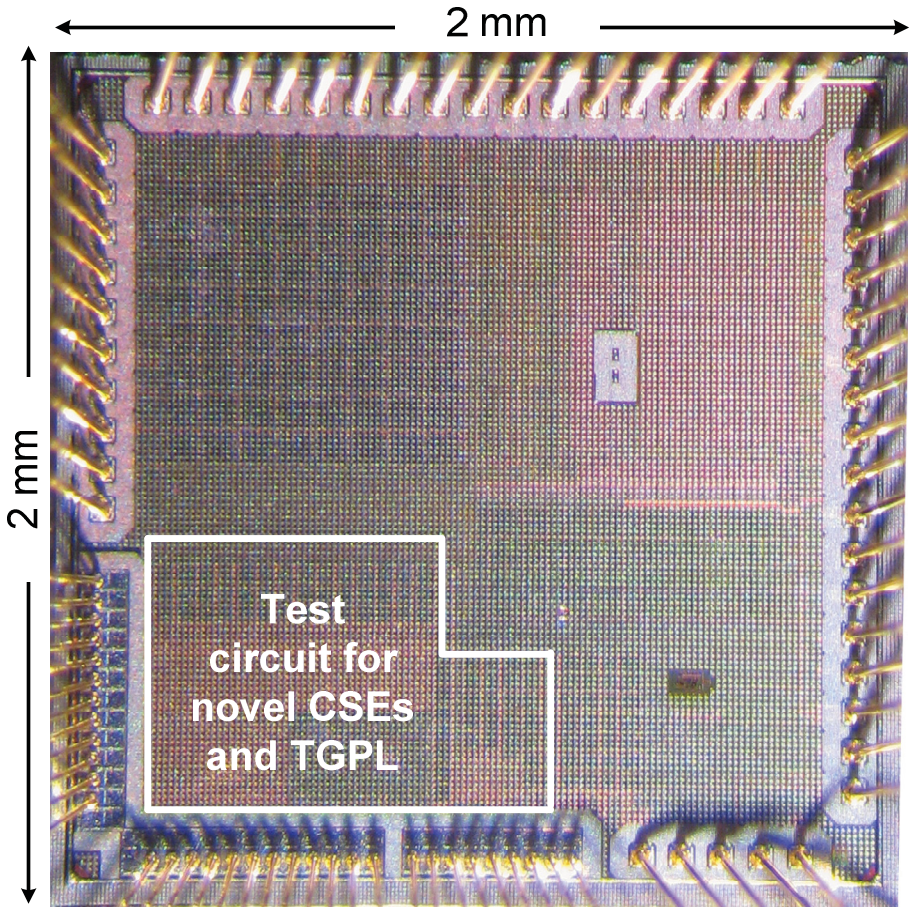


Fig. 7.3. Micrograph of the chip in 65-nm STCMOS065 technology.

both the novel CSEs (CP^3L and CSP^3L) and TGPL itself have been integrated in their minimum ED^3 and ED sizings. Different loads are chosen for the minimum ED^3 (64 minimum symmetrical inverters) and the minimum ED (16 minimum symmetrical inverters, implying the need of a larger sizing to achieve high-speed) designs.

Fig. 7.7 shows a block diagram of delay and energy measurement blocks, as well as the area occupation of the three CSE topologies in their two different optimum sizings.

The delay measurement setup is basically the same proposed in [NWO04], i.e. clock-to-data, τ_{CD}, clock-to-output, τ_{CQ}, and τ_{DQ} delays of each device under test (DUT) are measured as time differences by using an additional capturing Master-Slave (MS) CSE, which is clocked by a signal $CKMS$ obtained by making a pulse propagate through a programmable

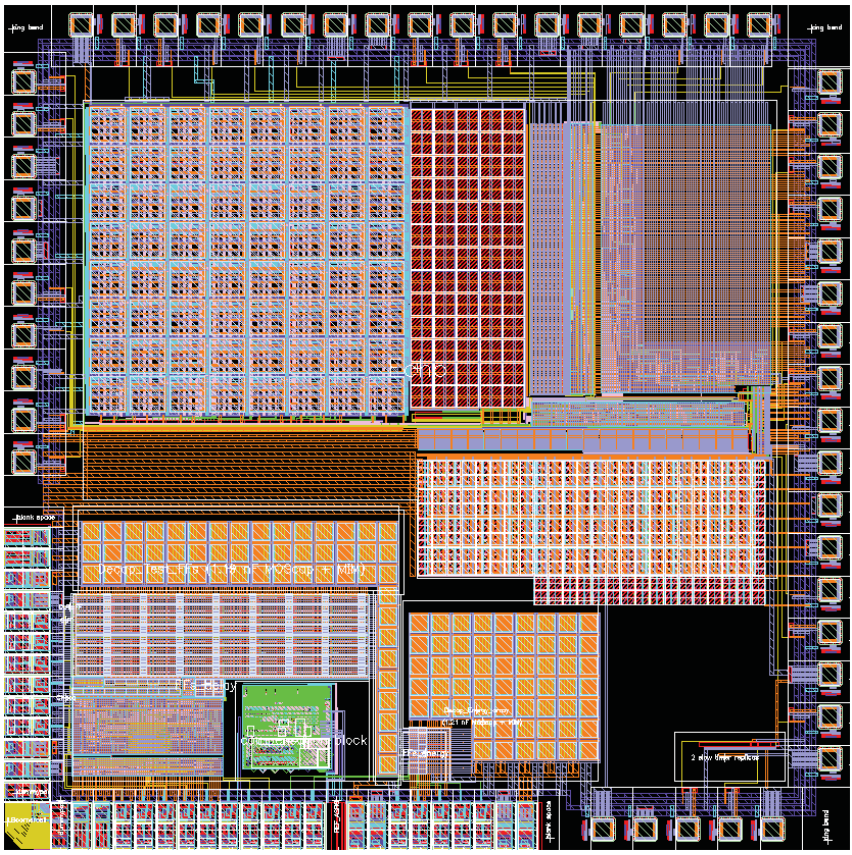


Fig. 7.4. Layout of the chip in 65-nm STCMOS065 technology.

Delay Generator (DG) circuit, as for D and CK inputs provided to the DUT. More specifically, by focusing on the i-th test unit picture in Fig. 7.7, there are four signals, i.e. D, CK and Q of the DUT and $CKMS$ (clock) of the capturing MS CSE. For each timing measurement, these signals are generated my making a pulse, generated from a Delayed (to allow settings to be captured) Pulse Generator (DPG), traverse the DG, thus being delayed in three different ways to originate D, CK and $CKMS$ signals that, after buffering and conditioning, are provided to the i-th test unit. For instance, assume that one wants to measure τ_{CD}:

a) First CK is selected as data input of the capturing MS CSE through the $3:1$ multiplexer (the 2 bits control signal is $Sel - DUT_{<1:0>}$). By sweeping (through the DG) the delay between CK and $CKMS$, at a certain point the correct CK value is not captured by MS CSE anymore. This event is identified by a certain $CKMS$ delay, t_{CK}, which is known (see following details on DG characterization).

b) Then, D is selected as data input for the MS CSE. Analogously, the missing D capture is identified by another $CKMS$ delay, t_D.
c) The difference between t_D and t_{CK} is the measured τ_{CD}

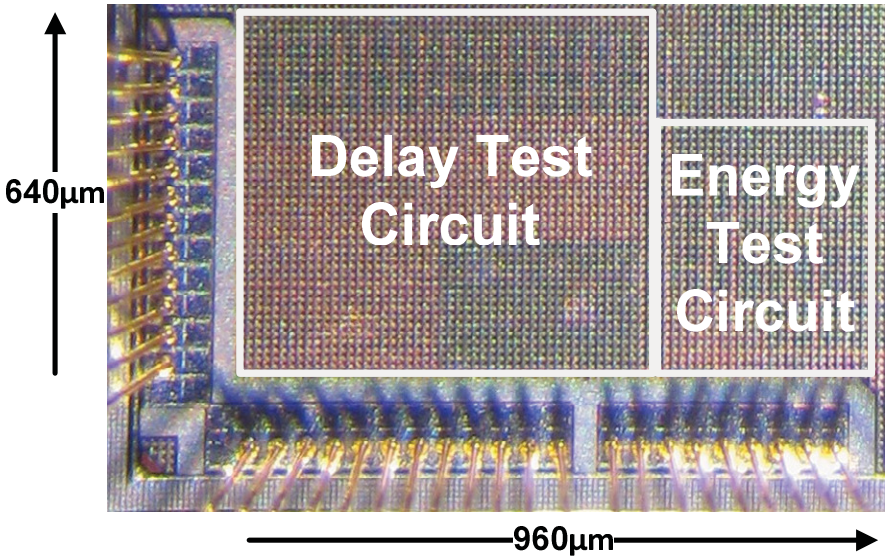


Fig. 7.5. Micrograph of the test circuit for novel CSEs and TGPL.

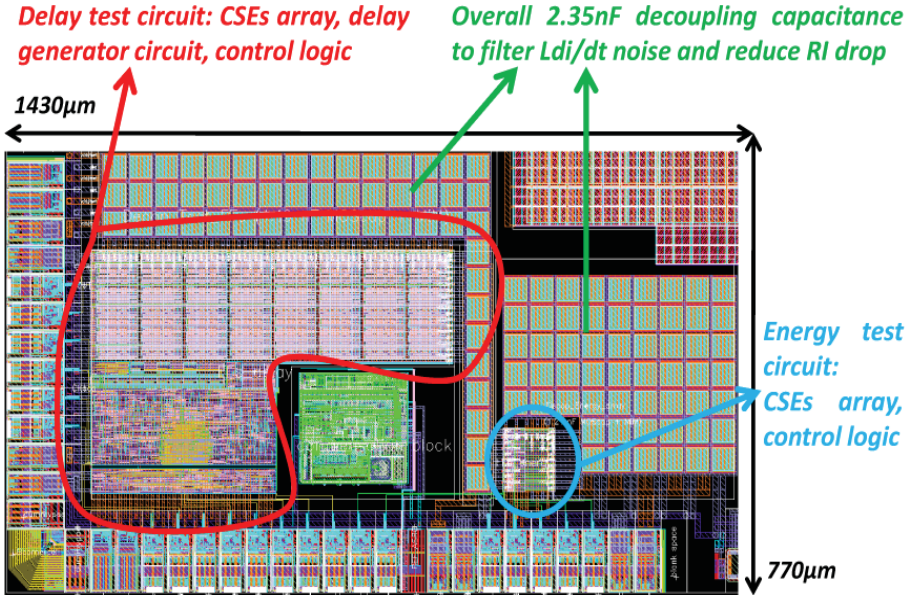


Fig. 7.6. Layout of the test circuit for novel CSEs and TGPL.

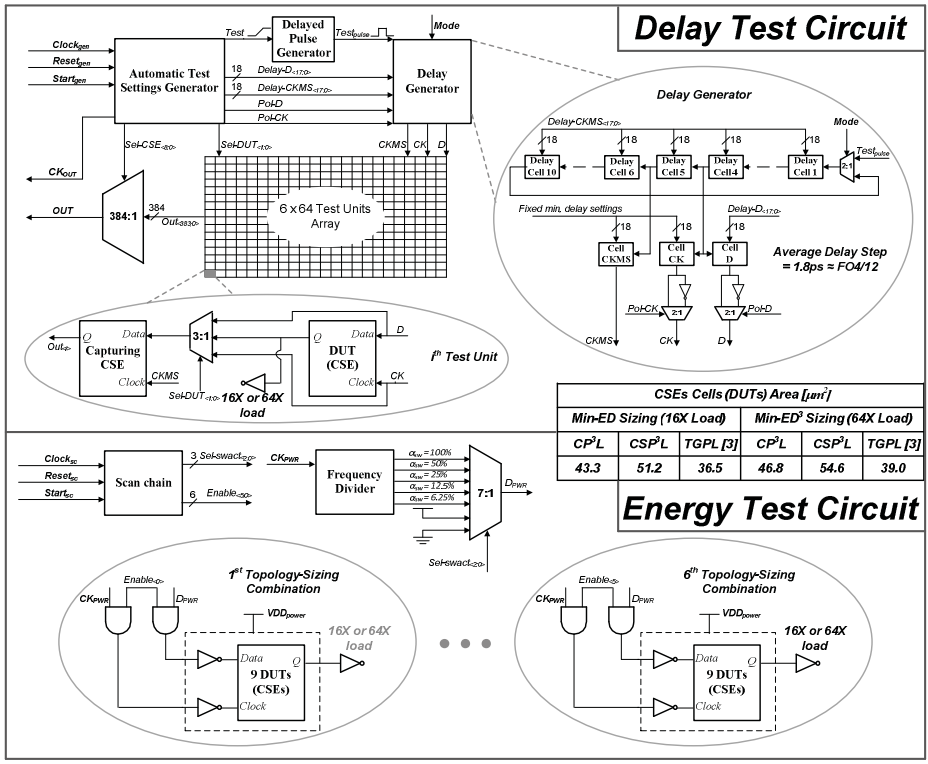


Fig. 7.7. Block diagram of delay and energy measurement circuits.

An analogous procedure is followed to measure τ_{CQ} and τ_{DQ} (by differently setting $Sel-DUT_{<1:0>}$) and this is obviously done under different τ_{CD} values in order to explore the setup and hold timing characteristic of the DUT. The delays between CK/D and $CKMS$ and between CK and D are set through two 18 bits control signals, $Delay-CKMS<17:0>$ and $Delay-D<17:0>$, allowing a coarse and fine tuning of the delays themselves. The absolute delays from the DG to the DUT are carefully regulated in order to explore the significant parts of the τ_{CQ} vs. τ_{CD} and τ_{DQ} vs. τ_{CD} curves. Finally, two additional control signals, $Pol-CK$ and $Pol-D$, are used to select a rising or falling CK (given that CP^{3}L and CSP^{3}L are negative edge triggered, while TGPL is positive edge-triggered) and D transition.

The strength of this approach [NWO04] is that each delay measurement is carried out locally as the difference of two times (among t_{CK}, t_D and t_Q) by employing an additional dummy signal $CKMS$. This allows to compensate for the unavoidable skew that occurs when distributing CK and D (and hence also the arising Q) signals.

The delay measurement resolution is set by the minimum delay step that can be achieved through the DG. This is measured by setting the DG in a ring oscillator fashion through $MODE$ external signal and measuring the (divided by 512 through a frequency divider) oscillation frequency. It is found that the minimum delay step is equal to 1.8ps (nearly $FO4/12$), which is quite smaller than the value achieved in [NWO04] (5ps at 0.11μm technology) even by accounting for technology scaling.

Moreover, since the delays measurement is carried out locally through the capturing MS CSE coupled to each DUT, differently from [NWO04] where a DG was used for each DUT, a single DG has been here employed to test an array of 384 CSEs, i.e. 64 DUTs for each topology (CP^3L/CSP^3L/TGPL) - sizing (min. ED/ED^3) combination. This is obviously done to extract also variability results. A final 384:1 multiplexer (control signal $Sel-CSE<8:0>$) is used to select the output of one among the 384 test units.

All the control signals so far discussed are automatically internally generated through a quite complex finite state machine. Overall, by considering that for each reference time measurement (i.e., t_{CK}, t_D or t_Q) four clock cycles are needed (given that one has to correctly initialize the state of the DUT according to whether setup or hold characteristics are explored), and the sweep of all the control signals in the various possible combinations, a total 268697600 number of clock cycles is needed to fully explore the timing characteristics of all the DUTs within a chip and a total ≈ 64M significant output bits are sent at the output.

Also the energy measurement setup draws inspiration from [NWO04] and allows to extract the transient energy (i.e. dynamic plus short-circuit contributions) per clock cycle under different data input switching activity, α_{sw}, values. A frequency divider is used to sweep α_{sw} in the $[0-100]\%$ range and, at each time, only one topology/sizing combination among the six available ones is selected (9 DUTs for each to reach an easily observable power dissipation), while the other five are disabled (gated clock and data). A scan chain is used to provide the above settings and the leakage is compensated at the beginning by measuring the power dissipation when all the DUTs are gated. The results on the leakage currents of the novel CSEs and TGPL that are reported in the following have been extracted through simulations and not through measurements as done for the transient energy.

7.4 Setup and Hold Timing Characteristics

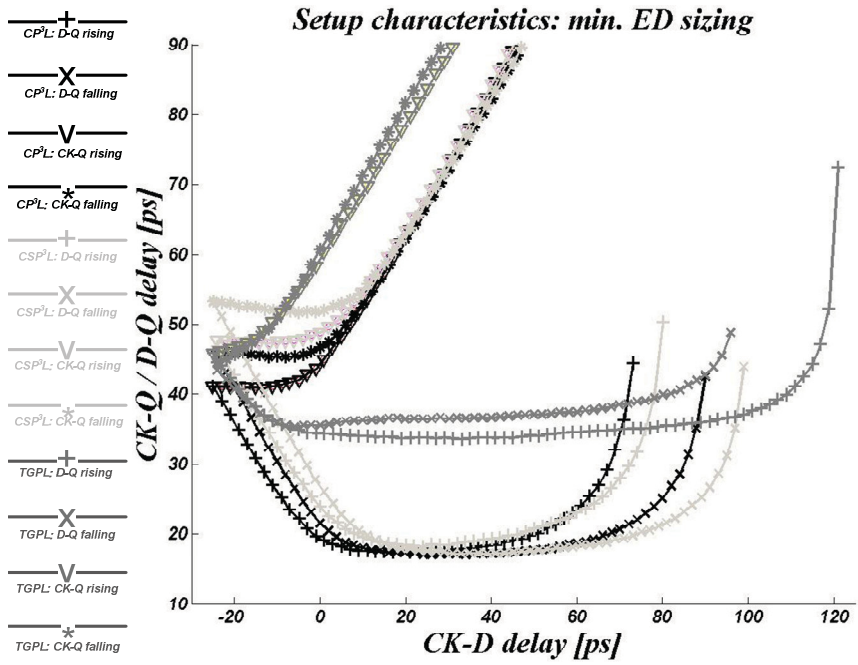
Fig. 7.8 shows typical measured setup and hold characteristics of the proposed CSEs and TGPL. CP^3L and CSP^3L achieve a $\tau_{DQ,min}$ close to 15ps ($0.7FO4$) and 17.5ps ($0.8FO4$) in their minimum ED (16X load) and ED^3

(64X load) sizings, respectively, while TGPL has a $\tau_{DQ,min}$ 50% (100%) larger in the min. ED^3 (ED) case.

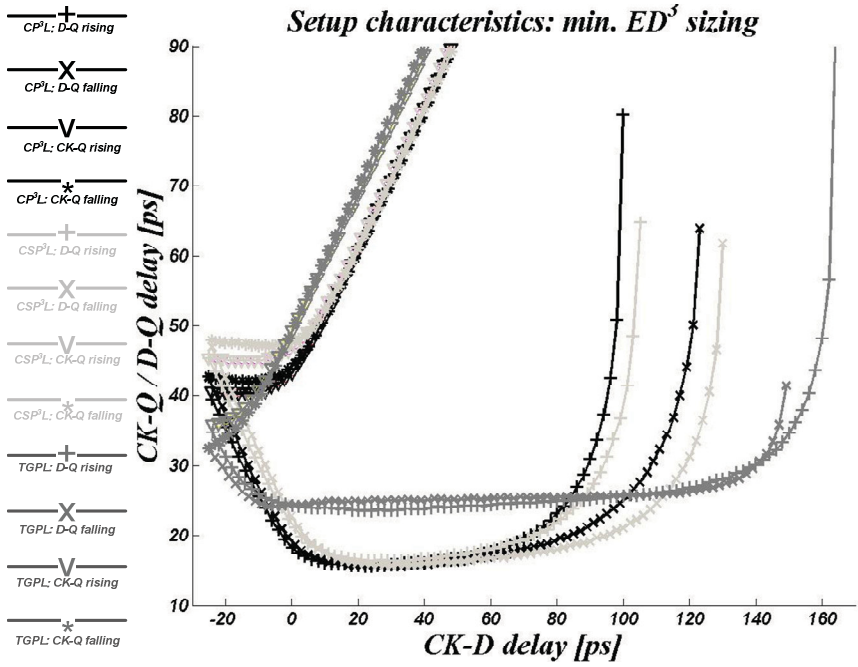
Note that TGPL is a very fast topology thanks to the small logical effort, branching effort and parasitic delay of stages in data-to-output path and the small layout impact leading to lower interconnect parasitics (see Chapter 5 and [ACP11-2]). The $\tau_{DQ,min}$ of the proposed CSEs is further lowered mainly given that S_{neg} and R nodes have roughly half the load than TGPL. Hence the Half Latches are very fast and push-pull stage size can be increased without degrading the energy-efficiency. Also the branching due to fixed size gates is reduced and, from a delay perspective, the load due to interconnect parasitics is smaller.

Although the PGs of all CSEs were designed to achieve a nearly $4FO4$ pulse width duration, it is apparent that the flat τ_{DQ} region is more pronounced in TGPL, while the proposed topologies exhibit a flat τ_{DQ} region slightly smaller (nearly $2.5FO4$) because of the more complex pulse generation mechanism.

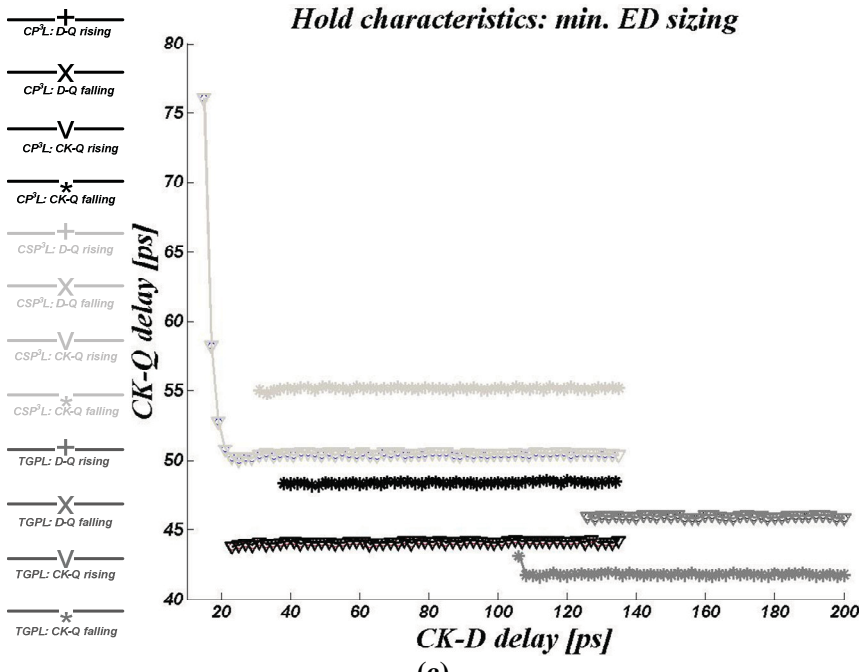
By inspection of setup and hold time characteristics, the proposed CSEs do not seem to suffer from the normal setup-hold times tradeoff, since when the previously stored Q value is equal to $'1'$ ($'0'$), only a data falling (rising) transition can be captured in the successive clock cycle. This is true because Q_D is enough delayed so that even if a new $'1'$ ($'0'$) value is captured at the



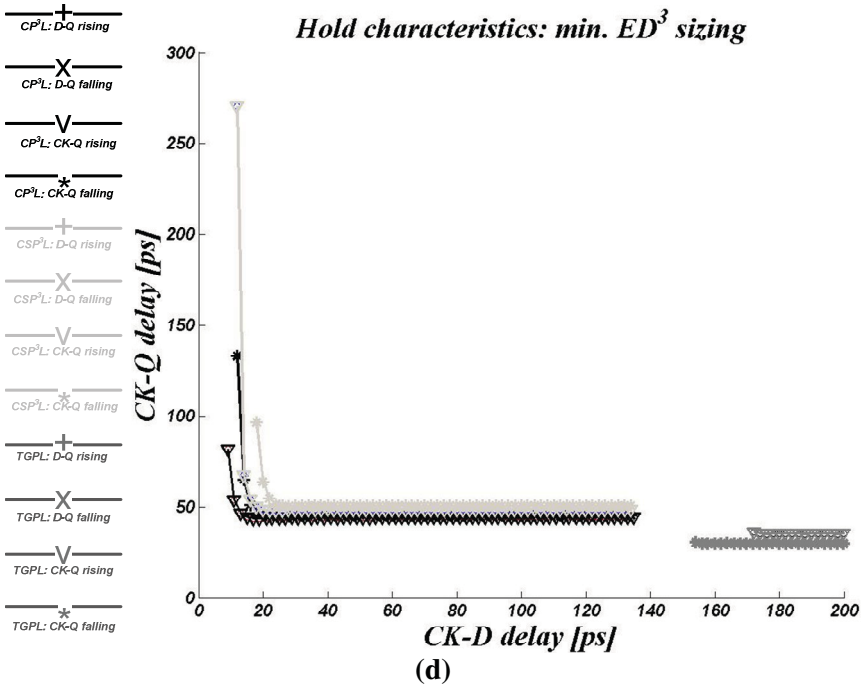
(a)


Setup characteristics: min. ED³ sizing
CP²L: D-Q rising
CP²L: D-Q falling
CP²L: CK-Q rising
CP²L: CK-Q falling
CSP²L: D-Q rising
CSP²L: D-Q falling
CSP²L: CK-Q rising
CSP²L: CK-Q falling
TGPL: D-Q rising
TGPL: D-Q falling
TGPL: CK-Q rising
TGPL: CK-Q falling
CK-Q / D-Q delay [ps]
CK-D delay [ps]
90
80
70
60
50
40
30
20
10
-20
0
20
40
60
80
100
120
140
160

(b)


Hold characteristics: min. ED sizing
CP²L: D-Q rising
CP²L: D-Q falling
CP²L: CK-Q rising
CP²L: CK-Q falling
CSP²L: D-Q rising
CSP²L: D-Q falling
CSP²L: CK-Q rising
CSP²L: CK-Q falling
TGPL: D-Q rising
TGPL: D-Q falling
TGPL: CK-Q rising
TGPL: CK-Q falling
CK-Q delay [ps]
CK-D delay [ps]
80
75
70
65
60
55
50
45
40
20
40
60
80
100
120
140
160
180
200

(c)



(d)

$T_{SETUP,rise/fall}$ = *CK-D* : $d(CK\text{-}Q)/d(CK\text{-}D)$ = -1 (minimum *D-Q*)
$T_{HOLD,rise/fall}$ = *CK-D* : $d(CK\text{-}Q)/d(CK\text{-}D)$ = -1 (CSE has to change its stored value)
$T_{HOLD,D=Q=0/1}$ = *CK-D* : CSE should maintain stored value but does not

Fig. 7.8. Setup (a-b) and Hold (c-d) timing characteristics of CP^{3}L, CSP^{3}L and TGPL.

beginning of the transparency window (as happens in hold characteristics definition), the Pseudo-NAND (Pseudo-NOR) gate is not enabled since the transparency window itself closes before the enabling $Q_D = '1'$ ($Q_D = '0'$) value arrives. This means that the hold time evaluated in the τ_{CQ} vs. τ_{CD} curves, called $T_{HOLD,rise/fall}$ in Fig. 7.8, is not above the setup time, $T_{SETUP,rise/fall}$, as normally happens in other CSEs like TGPL. Instead it is well below and is limited only by the delay from CK to S_{neg}/R occurring at the beginning of the transparency window (i.e., D has to remain stable only for the minimum time allowing its initial capture).

However, one must note that the hold time has to be defined also in the case where the previously stored value has to be maintained. In such case, the setup-hold times tradeoff is basically restored also for CP^{3}L and CSP^{3}L, given that the hold time becomes the last point in the setup characteristic leading to finite τ_{DQ} and τ_{CQ} delays, called $T_{HOLD,D=Q=0/1}$ in Fig. 7.8.

7.5 Dissipation and Energy/Leakage-Delay Tradeoffs

Fig. 7.9 shows the measured transient energy dissipated per clock cycle, E_{TRAN}, versus data switching activity. CP^{3}L and CSP^{3}L show an energy consumption which is 40% to 60% higher than TGPL, given their higher complexity. Nevertheless, the adoption of the conditional technique allows to save some energy by avoiding unnecessary internal transitions when the data input remains equal to its previous value.

The energy-delay tradeoff in the 25% data switching activity condition is depicted in Fig. 7.10. The energy-efficiency of CP^{3}L and CSP^{3}L is higher than TGPL in the tested conditions. In particular, CP^{3}L has 1.3X (2.3X) better ED (ED^3) product than TGPL. Given that the trends can be inferred from the single energy-efficient points (see Chapter 5), these results let us infer that the novel CSEs outperform TGPL in the whole high-speed region of the energy-delay space and up to the minimum ED point. Also the leakage-delay tradeoff has been analyzed as shown in Fig. 7.11. CP^{3}L has 2.7X (5.4X) better $I_{Leak}D$ ($I_{Leak}D^3$) product, being I_{Leak} the average leakage current of the CSE (estimated by means of simulations).

Note that the measured $FO4 = 22$ps, $E_{min} = 0.3$fJ and $I_{Leak,min} = 9.3$nA values have been used for results normalization.

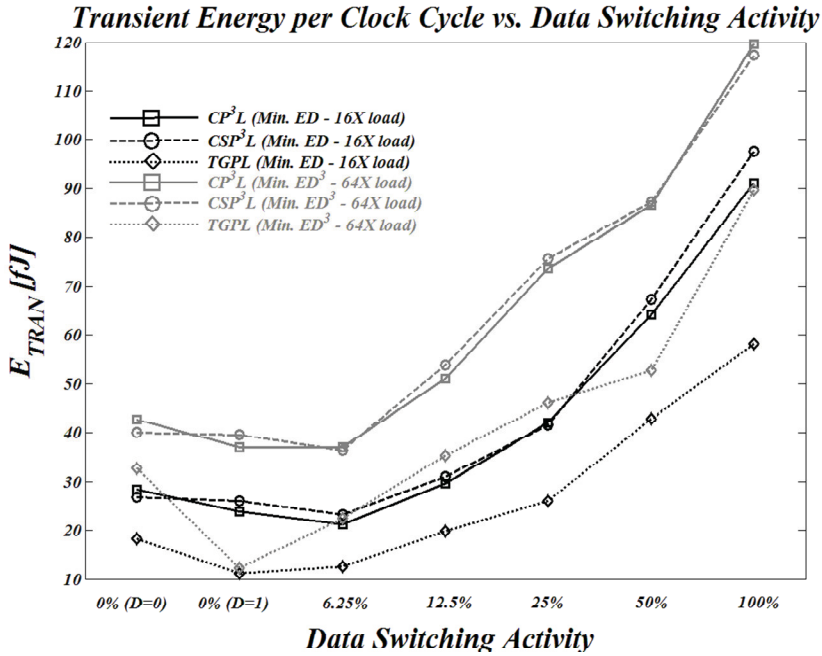


Fig. 7.9. Transient energy vs. switching activity.

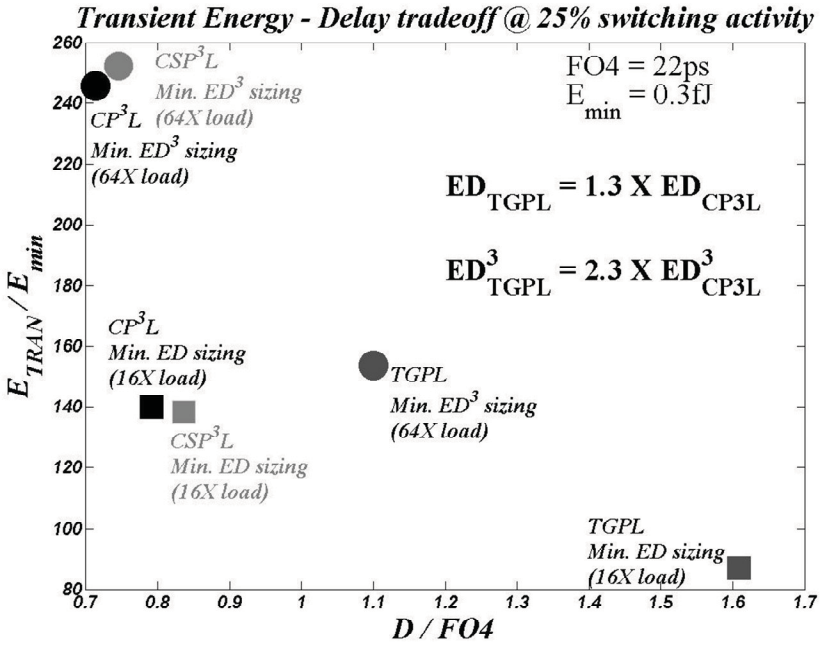


Fig. 7.10. Energy-delay tradeoff.

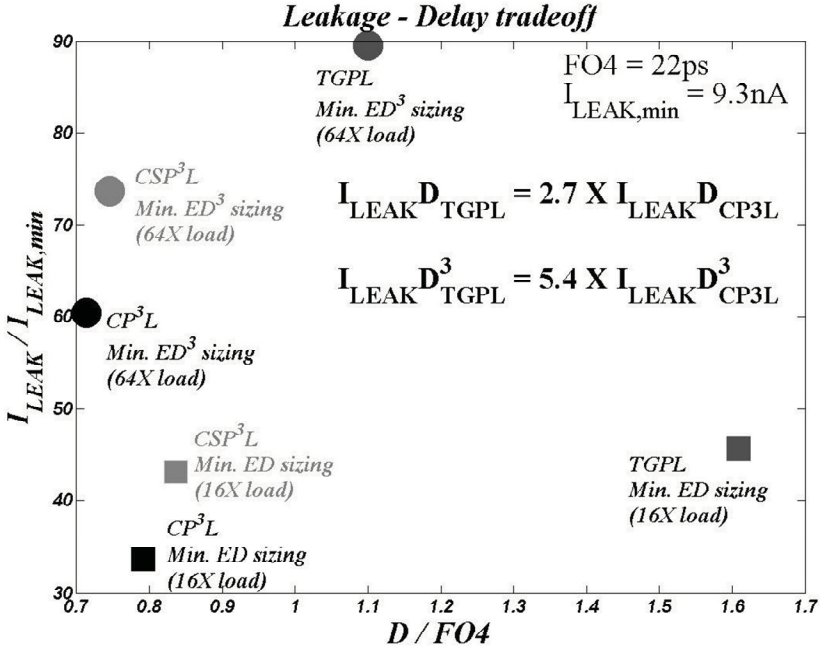
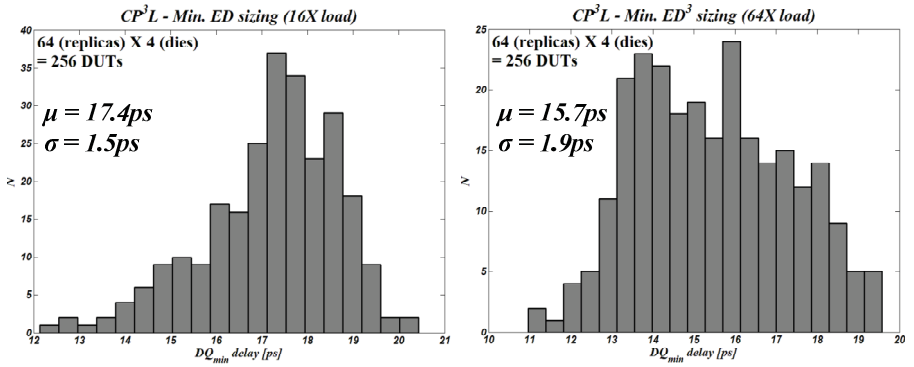


Fig. 7.11. Leakage-delay tradeoff.

7.6 Variability of Timing Parameters and Leakage

The mean, standard deviation and variability of the various timing parameters and simulated leakage are shown in Fig. 7.12. Timing parameters variability is estimated through measurements on 256 DUTs belonging to four different dies, while leakage variability is estimated through 2000 runs Monte Carlo simulations.

By inspection of results, percentage variability is similar for CP^3L, CSP^3L and TGPL for all the considered parameters, except for $\tau_{DQ,min}$ and $\tau_{CQ,min}$ delays for which a slightly augmented value is observed in the proposed topologies. Nevertheless, even considering the impact of process variations, the proposed CSEs are still largely faster than TGPL.



Min. ED sizing (16X load)										
Parameter	Unit	μ			σ			σ/μ%		
		CP^3L	CSP^3L	TGPL	CP^3L	CSP^3L	TGPL	CP^3L	CSP^3L	TGPL
$D\text{-}Q_{min,rise}$	[ps]	17.4	18.4	33.7	1.5	1.6	2.2	8.4	8.8	6.6
$D\text{-}Q_{min,fall}$	[ps]	17.1	17.3	35.4	1.4	1.5	2.5	8.1	8.8	7.0
$T_{SETUP,rise}$	[ps]	-20.1	-24.1	-35.3	1.4	1.6	2.8	7.0	6.7	7.9
$T_{SETUP,fall}$	[ps]	-33.0	-35.1	5.0	2.4	2.4	0.4	7.3	6.7	7.9
$T_{HOLD,rise}$	[ps]	23.0	20.2	124.9	1.8	1.7	8.9	8.0	8.6	7.1
$T_{HOLD,fall}$	[ps]	38.1	30.9	103.0	3.0	2.8	6.2	7.9	8.9	6.0
$T_{HOLD,D=Q=0}$	[ps]	73.9	80.0	121.9	8.8	5.4	22.9	11.9	6.8	18.8
$T_{HOLD,D=Q=1}$	[ps]	90.5	99.3	96.4	8.7	7.4	6.9	9.6	7.5	7.2
$CK\text{-}Q_{min,rise}$	[ps]	44.2	50.5	45.9	2.7	3.0	2.4	6.1	6.0	5.3
$CK\text{-}Q_{min,fall}$	[ps]	48.5	55.2	41.8	2.8	3.2	2.8	5.9	5.8	6.8
Leakage	[nA]	313.6	561.7	401.6	136.7	247.7	175.9	43.6	44.1	43.8
Min. ED3 sizing (64X load)										
Parameter	Unit	μ			σ			σ/μ%		
		CP^3L	CSP^3L	TGPL	CP^3L	CSP^3L	TGPL	CP^3L	CSP^3L	TGPL
$D\text{-}Q_{min,rise}$	[ps]	15.7	16.4	23.7	1.9	2.0	1.8	11.8	12.3	7.5
$D\text{-}Q_{min,fall}$	[ps]	15.5	15.8	24.2	1.8	1.9	1.9	11.3	11.9	7.9
$T_{SETUP,rise}$	[ps]	-22.9	-28.8	-28.4	1.6	1.9	1.9	7.0	6.7	6.8
$T_{SETUP,fall}$	[ps]	-28.9	-30.7	4.2	2.0	2.0	0.3	7.0	6.5	7.6
$T_{HOLD,rise}$	[ps]	14.2	17.8	171.1	1.6	2.2	13.7	11.0	12.5	8.0
$T_{HOLD,fall}$	[ps]	17.6	23.4	152.6	2.1	2.8	11.6	11.8	12.0	7.6
$T_{HOLD,D=Q=0}$	[ps]	100.0	105.4	164.3	7.4	8.1	12.7	7.4	7.7	7.7
$T_{HOLD,D=Q=1}$	[ps]	123.6	130.1	149.1	8.0	9.0	11.8	6.5	6.9	7.9
$CK\text{-}Q_{min,rise}$	[ps]	44.5	48.7	35.4	2.5	2.6	0.6	5.7	5.4	1.8
$CK\text{-}Q_{min,fall}$	[ps]	45.4	50.7	29.9	2.1	1.8	0.8	4.6	3.5	2.5
Leakage	[nA]	685.7	424.6	832.5	304.5	184.7	368.0	44.4	43.5	44.2

Fig. 7.12. Variability of timing parameters and leakage.

7.7 Performances Summary and Comparison

Tab. VII.I summarizes the main figures of merit of the integrated CSEs and includes results for TGFF and STFF extracted through simulations and measurements for ACFF reported in [TFH11]. Simulation results are significant, given that they match measurements typically within 10% in the case of the integrated CSEs, while the comparison with ACFF is even pessimistic given that results in [TFH11] are relative to a 40nm technology.

These topologies have not been integrated given the different applicative target (TGFF/ACFF, see Chapter 5) or the expected worse energy-efficiency (STFF, see Chapter 5) but represent valuable references for comparison because of their small ED product (TGFF/ACFF) and $\tau_{DQ,min}$ (STFF). Other

TABLE VII.I: NOVEL CSES COMPARISON WITH TGPL, TGFF, ACFF, STFF

		Measured 65nm VDD = 1V			*Simulated (***) 65nm VDD = 1V*		*Measured 40nm VDD = 1.1V*
Parameter	***Unit***	***CP²L Min. ED 16X load***	***CSP²L Min. ED 16X load***	***TGPL Min. ED 16X load***	***TGFF [1] Min. ED 16X load***	***STFF [5] Min. ED 16X load***	***ACFF [4] (****)***
D-Q_{min} (*)	[ps]	**17.3 (1.0X)**	**17.9 (1.0X)**	34.6 (2.0X)	106.6 (6.2X)	40.6 (2.3X)	264.0 (15.3X)
T_{SETUP} (*)	[ps]	-26.6 (**)	-29.6 (**)	-15.2 (**)	43.1 (**)	-40.9 (**)	197.0 (**)
CK-Q_{min} (*)	[ps]	46.4 (1.1X)	52.9 (1.2X)	43.9 (1.0X)	61.1 (1.4X)	60.2 (1.4X)	67.0 (1.5X)
T_{HOLD} (*)	[ps]	82.2 (**)	89.7 (**)	114.0 (**)	-20.0 (**)	130.4 (**)	-73.0 (**)
E_{DYN}@50%	[fJ]	64.0 (3.2X)	67.4 (3.4X)	42.8 (2.2X)	19.7 (1.0X)	63.6 (3.2X)	n.a.
E_{DYN}@25%	[fJ]	42.0 (4.3X)	41.5 (4.3X)	26.1 (2.7X)	9.7 (1.0X)	41.2 (4.2X)	n.a.
E_{DYN}@10%	[fJ]	26.1 (13.1X)	28.0 (14.0X)	17.1 (8.6X)	5.8 (2.9X)	29.3 (14.7X)	2.0 (1.0X)
Leakage	[nA]	313.6 (2.4X)	401.6 (3.1X)	424.6 (3.2X)	130.9 (1.0X)	408.4 (3.1X)	n.a.
Clock Load	[fF]	2.8 (2.5X)	2.8 (2.5X)	1.1 (1.0X)	3.5 (3.2X)	1.4 (1.3X)	n.a.
E*D@ 50%	[ps*fJ]	**1107.2 (1.0X)**	1206.5 (1.1X)	1480.9 (1.3X)	2100.0 (1.9X)	2582.2 (2.3X)	n.a.
E*D@ 25%	[ps*fJ]	**726.6 (1.0X)**	**742.9 (1.0X)**	903.1 (1.2X)	1034.0 (1.4X)	1672.7 (2.3X)	n.a.
E*D@ 10%	[ps*fJ]	**451.5 (1.0X)**	501.2 (1.1X)	591.7 (1.3X)	618.3 (1.4X)	1189.6 (2.6X)	528.0 (1.2X)
Parameter	***Unit***	***CP²L Min. ED³ 64X load***	***CSP²L Min. ED³ 64X load***	***TGPL Min. ED³ 64X load***	***TGFF Min. ED³ 64X load***	***STFF Min. ED³ 64X load***	
D-Q_{min} (*)	[ps]	**15.6 (1.0X)**	**16.1 (1.0X)**	24.0 (1.5X)	83.4 (5.3X)	23.8 (1.5X)	
T_{SETUP} (*)	[ps]	-25.9 (**)	-29.8 (**)	-12.1 (**)	30.1 (**)	-25.0 (**)	
CK-Q_{min} (*)	[ps]	45.0 (1.4X)	49.7 (1.5X)	32.7 (1.0X)	51.8 (1.6X)	63.5 (1.9X)	
T_{HOLD} (*)	[ps]	111.8 (**)	117.8 (**)	161.9 (**)	-18.9 (**)	126.3 (**)	
E_{DYN}@50%	[fJ]	86.5 (3.3X)	87.2 (3.3X)	52.8 (2.0X)	26.6 (1.0X)	69.5 (2.6X)	
E_{DYN}@25%	[fJ]	73.7 (5.0X)	75.7 (5.1X)	46.1 (3.1X)	14.8 (1.0X)	58.6 (4.0X)	
E_{DYN}@10%	[fJ]	44.6 (4.3X)	45.2 (4.4X)	30.7 (3.0X)	10.3 (1.0X)	50.0 (4.9X)	
Leakage	[nA]	561.7 (2.3X)	685.7 (2.8X)	832.5 (3.4X)	242.2 (1.0X)	609.6 (2.5X)	
Clock Load	[fF]	3.3 (3.0X)	3.3 (3.0X)	1.1 (1.0X)	4.4 (4.0X)	1.4 (1.3X)	
E*D^3@50%	10^3*[ps³*fJ]	**328.4 (1.0X)**	363.9 (1.1X)	729.9 (2.2X)	15430 (47X)	936.9 (2.9X)	
E*D^3@25%	10^3*[ps³*fJ]	**279.8 (1.0X)**	315.9 (1.1X)	637.3 (2.3X)	8585.4 (31X)	790.0 (2.8X)	
E*D^3@10%	10^3*[ps³*fJ]	**169.3 (1.0X)**	188.6 (1.1X)	424.4 (2.5X)	5975.0 (35X)	674.1 (4.0X)	

(*) Average between rise/fall cases

(**) Since they can assume positive and negative values, Setup and Hold times are not normalized

(***) Simulation results match measurements within 10% in the case of the integrated FFs

(****) The comparison with ACFF [4] is pessimistic because of technology scaling

CSEs have not been nor integrated nor included in the simulations for this comparison given that they are all outperformed by TGPL in terms of speed and energy-efficiency (see Chapter 5 and [ACP11-2]).

The improvements introduced with the proposed topologies are highlighted in the table, where it is shown that TGPL and STFF have $\tau_{DQ,min}$ always more than 50% higher than CP^{3}L and CSP^{3}L. As concerns energy-efficiency, the proposed CSEs have largely the best ED^3 product (always more than 120% reduction) and outperform TGPL, TGFF and ACFF in terms of ED product (always more than 20% reduction), i.e. even at the boundary between high-speed and low-energy design regions.

Overall, CP^{3}L and CSP^{3}L prove to be the fastest voltage mode CSEs so far proposed and the most energy-efficient in the whole high-speed energy-delay space region where E^iD^j products with $j \geq i$ are optimized.

CONCLUSION

Clocked storage elements (CSEs) are among the most important elements in the design of digital circuits and, in particular, of microprocessors. CSEs separate the various stages by which a pipeline is made up, maintain the present logic state and assure the transition to the successive logic state at the right instant. Basically, they allow to synchronize the entire flow of data.

With the aim of obtaining a conspicuous performance increment at every new technology node, the dimensional scaling is supported by the reduction of the number of logic levels inside each pipeline stage. As a consequence, the fraction of the clock period that is occupied by delay of CSEs proportionally grows. On the other hand, due to their typically high switching activity, the circuits related to the clock generation, distribution and those that synchronize the system (CSEs), contribute about for the $30-50\%$ of the whole energy consumption.

Moreover, the issue of energy consumption can be no more faced separately from that of speed performances, since, in practice, unavoidable constraints on the power-budgets subsist. Hence, energy dissipation is as fundamental as speed in the design of high-performance microprocessors.

For the above reasons, the need for optimal CSEs design, comparison and selection strategies arises.

The research activity carried out by the candidate in pursuit of the Ph.D. degree has mainly focused on the development of novel (1) transistor-level and (2) clock-domain micro-architectural level energy-efficient design methodologies for CSEs, on the (3) comparison of state of the art CSEs by accounting for effects that were previously neglected and on the (4) definition of novel very high-speed CSE topologies. The main results that have been achieved are summarized in the following.

1) A general and complete transistor-level CSE design methodology has been proposed basing on suitable energy-delay metrics with a clear physical meaning, which in turn has permitted to gain an insight into the design issues [ACP10-2], [ACP12-1]. So far, typical approaches had been based on extensive simulations, which are computationally inefficient and force the designer to arbitrarily discard potentially good solutions [ACP09-1]. Instead, the proposed approach exploits the properties of the energy-efficient curve in order to search only among the best and promising design points [ACP10-3]. The methodology starts from a preliminary identification of independent and dependent design variables and, unlike the previously strategies, it also includes the CSE input capacitance as a design parameter in order to fully explore the energy-efficiency potentials [ACP12-2].

Furthermore, an optimum design approach, based on a modification of the Logical Effort method, has been introduced to improve the delay and dissipation of high-speed Transmission-Gate based Master-Slave Flip-Flops [CPP11], [CPP12].

2) As concerns micro-architectural level design of clock networks, an optimum clock slope sizing methodology has been proposed from the point of view of the joint energy dissipation of clock buffers and CSEs at the clock domain level, given that an inherent energy tradeoff arises among the two contributions [ACP10-1]. The effect of clock slope variation on timing and energy features of CSEs and clock buffers has been analyzed for a wide range of topologies and then the optimum clock slope has been evaluated [ACP09-2], [ACP09-3], [ACP09-4], [ACP10-6]. The analysis shows that the optimum value can significantly depart from typical fast $FO2 - FO3$ assumptions. The impact of such an optimization on local skew/jitter sources, on the CSEs tolerance to process variations and the influence of technology scaling have also been considered.

3) A thorough investigation on previously proposed CSE topologies has been carried out in order to select those exhibiting the best energy-delay tradeoff when including the influence of several parameters (load, switching activity, logic depth) [ACP11-1], [ACP11-2]. The results found are significantly different with respect to previous comparisons in the literature because of the broad spectrum of circuits investigated and of the inclusion of effects arising in nanometer technologies (the analysis has been performed in a 65-nm CMOS technology), such as the impact of parasitics due to local interconnections [ACP10-4], [ACP10-5], [ACP11-3]. For the first time, the impact of leakage on CSEs energy in standby and active mode has been discussed, and the influence on the CSEs design has been highlighted. The tradeoff between leakage, area, clock load and delay has been analyzed. Several additional CSEs features have also been considered, like the load on

the clock network, the layout efficiency and the leakage-area interdependence.

4) Starting from the result that Pulsed Latch circuits exhibit the highest energy-efficiency in the high-speed region of the energy-delay space, two novel Pulsed Latch topologies, called Conditional (Shareable) Push-Pull Pulsed Latch (CP^3L and CSP^3L), have been proposed [CAP12]. Thanks to the extremely fast inner latch and to the usage of a conditional Pulse Generator, these circuits achieve by far better performances than state of the art Transmission-Gate Pulsed Latch (TGPL) in terms of minimum delay and energy-delay products. A 65-nm test chip has been fabricated in collaboration with the Berkeley Wireless Research Center to measure speed (delay resolution $< 2\text{ps}$) and dissipation of the proposed circuits and to test performances variability through several replicas. Measurements have confirmed that CP^3L and CSP^3L are the fastest CSE topologies ever proposed ($0.7FO4$ delay at 64X load) and exhibit the lowest ED and ED^3 products, thereby representing the most energy-efficient solution in the whole high-speed region of the energy-delay design space [CAP12].

REFERENCES

[ACP09-1] M. Alioto, E. Consoli, G. Palumbo, "Metrics and design considerations on the energy-delay tradeoff of digital circuits," in Proc. of *IEEE International Symposium on Circuits and Systems*, pp. 3150-3153, May 2009.

[ACP09-2] M. Alioto, E. Consoli, G. Palumbo, "Impact of Clock Slope on Energy/Delay of Pulsed Flip-Flops and Optimum Clock Domain Design," in Proc. of *European Conference on Circuit Theory and Design*, pp. 61-64, August 2009.

[ACP09-3] M. Alioto, E. Consoli, G. Palumbo, "Dependence of Differential Flip-Flops Performance on Clock Slope and Relaxation of Clock Network Design," in Proc. of *International Conference on Microelectronics*, pp. 110-113, December 2009.

[ACP09-4] M. Alioto, E. Consoli, G. Palumbo, "Optimum Clock Slope for Flip-Flops within a Clock Domain: Analysis and a Case Study," in Proc. of *International Conference on Electronics, Circuits and Systems*, pp. 275-278, December 2009.

[ACP10-1] M. Alioto, E. Consoli, G. Palumbo, "Flip-Flop Energy/Performance versus Clock Slope and Impact on the Clock Network Design," *IEEE Transactions on Circuits and Systems – Part I*, vol. 57, no. 6, pp. 1273-1286, June 2010.

[ACP10-2] M. Alioto, E. Consoli, G. Palumbo, "General Strategies to Design Nanometer Flip-Flops in the Energy-Delay Space," *IEEE Transactions on Circuits and Systems – Part I*, vol. 57, no. 7, pp. 1583-1596, July 2010.

[ACP10-3] M. Alioto, E. Consoli, G. Palumbo, “Nanometer Flip-Flops Design in the E-D Space," in Proc. of *International Conference on Microelectronics*, pp. 132-135, December 2010.

[ACP10-4] M. Alioto, E. Consoli, G. Palumbo, “Physical Design Aware Comparison of Flip-Flops for High-Speed Energy-Efficient VLSI Circuits," in Proc. of *International Workshop on Power And Timing Modeling Optimization and Simulation*, pp. 62-72, September 2010.

[ACP10-5] M. Alioto, E. Consoli, G. Palumbo, “Physical Design Aware Selection of Energy-Efficient and Low-Energy Nanometer Flip-Flops," in Proc. of *International Conference on Microelectronics*, pp. 60-63, December 2010.

[ACP10-6] M. Alioto, E. Consoli, G. Palumbo, “Clock Distribution in Clock Domains with Dual-Edge Triggered Flip-Flops to Improve Energy-Efficiency,” in Proc. of *IEEE International Symposium on Circuits and Systems*, pp. 321-324, June 2010.

[ACP11-1] M. Alioto, E. Consoli, G. Palumbo, “Analysis and Comparison in the Energy-Delay-Area Domain of Nanometer CMOS Flip-Flops: Part I – Methodologies and Design Strategies,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 5, pp. 725-736, May 2011.

[ACP11-2] M. Alioto, E. Consoli, G. Palumbo, “Analysis and Comparison in the Energy-Delay-Area Domain of Nanometer CMOS Flip-Flops: Part II – Results and Figures of Merit,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 5, pp. 737-750, May 2011.

[ACP11-3] M. Alioto, E. Consoli, G. Palumbo, “DET FF Topologies: A Detailed Investigation in the Energy-Delay-Area Domain,” in Proc. of *IEEE International Symposium on Circuits and Systems* , pp. 563-566, May 2011.

[ACP12-1] M. Alioto, E. Consoli, G. Palumbo, “From Energy-Delay Metrics to Constraints on the Design of Digital Circuits,” in print on *International Journal of Circuit Theory and Applications*.

[ACP12-2] M. Alioto, E. Consoli, G. Palumbo, “Design in the Energy-Delay Space,” in *Advanced Circuits for Emerging Technologies*, *Part I - Digital Design and Power Management*, 1st Chapter, *Wiley*, 2012.

[AFP04] A. Abdollahi, F. Fallah, M. Pedram, “Leakage Current Reduction in CMOS VLSI Circuits by Input Vector Control,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 2, pp. 140-154, February 2004.

[AP06] M. Alioto, G. Palumbo, “Impact of Supply Voltage Variations on Full Adder Delay: Analysis and Comparison,” *IEEE Transactions on Very*

Large Scale Integration (VLSI) Systems, vol. 14, no. 12, pp. 1322-1335, December 2006.

[APP10] M. Alioto, G. Palumbo, M. Pennisi, "Understanding the Effect of Process Variations on the Delay of Static and Domino Logic," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 18, no. 5, pp. 697-710, May 2010.

[AS90] M. Afghahi, C. Svensson, "A Unified Single-Phase Clocking Scheme for VLSI Systems," *IEEE Journal of Solid-State Circuits*, vol. 25, no. 1, pp. 225-233, February 1990.

[B99] S. Borkar, "Design challenges of technology scaling," *IEEE Micro*, vol. 19, no. 4, pp. 23–29, July/August 1999.

[BB98] D. Bailey, B. Benschneider, "Clocking Design and Analysis for a 600-MHz Alpha Microprocessor," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 11, pp. 1627-1633, November 1998.

[BKM05] S. Boyd, S. Kim, S. Mohan, "Geometric Programming and its Applications to EDA Problems," in Proc. of *Design, Automation & Test in Europe Conference*, Tutorial paper, June 2005.

[BKP05] S. Boyd, S. Kim, D. Patil, M. Horowitz, "Digital Circuit Optimization via Geometric Programming," *Operations Research*, vol. 53, no.6, pp. 899-932, November 2005.

[BPTM] *Berkeley Predictive Technology Models (BPTM)*, 2008. [Online]: http://www.eas.asu.edu/~ptm/.

[BV03] S. Boyd, L. Vandenberghe, "Convex Optimization," *Cambridge University Press*, 2003.

[CAP12] E. Consoli, M. Alioto, G. Palumbo, J. Rabaey, "Conditional Push-Pull Pulsed Latches with 726fJ•ps Energy-Delay Product in 65nm CMOS," in print on Proc. of *IEEE International Solid-State Circuits Conference*, February 2012.

[CBF01] A. Chandrakasan, W. Bowhill, F. Fox, "Design of High-Performance Microprocessor Circuits," *IEEE Press*, 2001.

[CEM99] A. Conn, I. Elfadel, W. Molzen, P. O'Brien, P. Strenski, C. Visweswariah, C. Whan, "Gradient-Based Optimization of Custom Circuits Using a Static Timing Formulation," in Proc. of *Design Automation Conference*, pp. 452-459, June 1999.

[CH99] Y. Cheng, C. Hu, "MOSFET Modeling and BSIM3 User's Guide," *Springer*, 1999.

[CK02] D. Chinnery, K. Keutzer, "Closing the Gap between ASIC and Custom: Tools and Techniques for High-Performance ASIC Design," *Kluwer Academic Publishers*, 2002.

[CPP11] E. Consoli, G. Palumbo, M. Pennisi, "TG Master-Slave FFs: High-Speed Optimization," in Proc. of *IEEE International Symposium on Circuits and Systems*, pp. 554-557, May 2011.

[CPP12] E. Consoli, G. Palumbo, M. Pennisi, "Reconsidering High-Speed Design Criteria for Transmission-Gate Based Master-Slave Flip Flops," in print on *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*.

[CSB92] A. Chandrakasan, S. Sheng, R. Brodersen, "Low-power CMOS digital design," *IEEE Journal of Solid-State Circuits*, vol. 27, no. 4, pp. 473–484, April 1992.

[CSH03] B. Chatterjee, M. Sachdev, S. Hsu, R. Krishnamurthy, S. Borkar, "Effectiveness and Scaling Trends of Leakage Control Techniques for Sub-130 nm CMOS Technologies," in Proc. of *IEEE International Symposium on Low Power Electronics and Design*, pp. 122-127, August 2003.

[DA99] J. Daga, D. Auvergne, "A Comprehensice Delay Macro Modeling for Submicrometer CMOS Logics," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 1, pp. 42-55, January 1999.

[DGR74] R. Dennard, F. Gaensslen, V. Rideout, Bassous, A. LeBlanc, "Design of ion-implanted MOSFETs with very small physical dimensions," *IEEE Journal of Solid-State Circuits*, vol. 9, no. 5, pp. 256–268, October 1974.

[DML95] S. Dutta, S. Mahant Shetty, S. Lusky, "A Comprehensive Delay Model for CMOS Inverters," *IEEE Journal of Solid-State Circuits*, vol. 30, no. 8, pp. 864-871, August 1995.

[DZO06] H. Dao, B. Zeydel, V. Oklobdzija, "Energy optimization of pipelined digital systems using circuit sizing and supply scaling," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 2, pp. 122-134, February 2006.

[E48] E. Elmore, "The Transient Response of Damped Linear Networks with Particular Regard to Wideband Amplifiers," *Journal of Applied Physics*, pp. 55-63, January 1948.

[FD85] J. Fishburn, A. Dunlop, "TILOS: A Posynomial Programming Approach to Transistor Sizing," in Proc. of *IEEE International Conference on CAD*, pp. 326-328, November 1985.

[GBP98] P. Gronowski, W. Bowhill, R. Preston, M. Gowan, R. Allmon, "High-Performance Microprocessor Design," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 5, pp. 676-686, May 1998.

[GCM91] B. Gieseke, R. Conrad, J. Montanaro, D. Dobberpuhl, "Push-Pull Cascode Logic," *U.S. patent*, no. 5,023,480, June 1991.

[GEH93] A. Gago, R. Escano, J. Hidalgo, "Reduced Implementation of D-Type DET Flip-Flops," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 3, pp. 400-402, March 1993.

[GGD94] 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, J. Kahle, "A 2.2 W, 80 MHz superscalar RISC microprocessor," *IEEE Journal of Solid-State Circuits*, vol. 29, no. 12, pp. 1440–1452, December 1994.

[GGH97] R. Gonzalez, B. Gordon, M. Horowitz, "Supply and threshold voltage scaling for low power CMOS," *IEEE Journal of Solid-State Circuits*, vol. 32, no. 8, pp. 1210-1216, August 1997.

[GLP97] S. Ganguly, D. Lehther, S. Pullela, "Clock Distribution Methodology for PowerPC™ Microprocessors," *Journal of VLSI Signal Processing Systems*, vol. 16, no. 2/3, pp. 181-189, June/July 1997.

[GNO07] C. Giacomotto, N. Nedovic, V. Oklobdzija, "The Effect of the System Specification on the Optimal Selection of Clocked Storage Elements," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 6, pp. 1392-1404, June 2007.

[H84] M. Horowitz, "Timing Models for MOS Circuits," *Ph.D. Dissertation*, Stanford University, 1984.

[H00] D. Harris, "Skew-Tolerant Circuit Design," *Morgan Kaufmann*, 2000.

[HA01] S. Heo, K. Asanovic, "Load-Sensitive Flip-Flop Characterization," in Proc. of *IEEE Computer Society Workshop on VLSI*, pp. 87-92, April 2001.

[HAA00] R. Heald, K. Aingaran, C. Amir, M. Ang, M. Boland, P. Dixit, G. Gouldsberry, D. Greenley, J. Grinberg, J. Hart, T. Horel, J. Hsu, J. Kaku, C. Kim, S. Kim, F. Klass, H. Kwan, G. Lauterbach, R. Lo, H. McIntyre, A. Mehta, D. Murata, S. Nguyen, Y. Pai, S. Patel, K. Shin, K. Tam, S. Vishwanthaiah, J. Wu, G. Yee, E. You, "A Third Generation SPARC V9 64-b Microprocessor," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 11, pp. 1526-1538, November 2000.

[HHW97] D. Harris, R. Ho, G. Wei, M. Horowitz, "The Fanout-of-4 Inverter Delay Metric," *unpublished manuscript*, 1997. [Online]: http://odin.ac.hmc.edu/~harris/research/FO4.pdf.

[HJ87] N. Hedenstierna, K. Jeppson, "CMOS Circuit Speed and Buffer Optimization," *IEEE Transactions on Computer-Aided Design*, vol. 6, no. 2, pp. 270-281, March 1987.

[HJF02] M. Hrishikesh, N. Jouppi, K. Farkas, D. Burger, S. Keckler, P. Shivakumar, "The Optimal Logic Depth per Pipeline Stage is 6 to 8 FO4 Inverter Delays," in Proc. of *Annual International Symposium on Computer Architecture*, pp. 14-24, May 2002.

[HJR09] M. Hwang, S. Jung, K. Roy, "Slope Interconnect Effort: Gate-Interconnect Interdependent Delay Modeling for Early CMOS Circuit Simulation," *IEEE Transactions on Circuits and Systems – I: Regular Papers*, vol. 56, no. 7, pp. 1428-1441, July 2009.

[HKA07] S. Heo, R. Krashinsky, K. Asanovic, "Activity-Sensitive Flip-Flop and Latch Selection for Reduced Energy," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 15, no. 9, pp. 1060-1064, September 2007.

[HMA05] S. Hsu, S. Mathew, M. Anders, B. Bloechel, R. Krishnamurthy, S. Borkar, "A 110GOPS/W 16b Multiplier and Reconfigurable PLA Loop in 90nm CMOS," in Proc. of *IEEE International Solid-State Circuit Conference*, vol. 1, pp. 376-377, February 2005.

[HMH01] R. Ho, K. Mai, M. A. Horowitz, "The Future of Wires," *Proc. of IEEE*, vol. 89, no. 4, pp. 490-504, April 2001.

[HN01] D. Harris, S. Naffziger, "Statistical Clock Skew Modeling with Data Delay Variations," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 6, pp. 888-898, December 2001.

[HTA98] M. Hamada, M. Takahashi, H. Arakida, A. Chiba, T. Terazawa, T. Ishikawa, M. Kanazawa, M. Igarashi, K. Usami, T. Kuroda, "A Top-Down Low Power Design Technique Using Clustered Voltage Scaling with Variable Supply-Voltage Scheme," in Proc. of *IEEE Custom Integrated Circuits Conference*, pp. 495-498, May 1998.

[HWA94] R. Hossain, L. Wronski, A. Albicki, "Low Power Design Using Double Edge Triggered Flip-Flops," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 2, no. 2, pp. 261-265, June 1994.

[ISN04] F. Ishihara, F. Sheikh, B. Nikolic, "Level-Conversion for Dual-Supply Systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 2, pp. 185-195, February 2004.

[ITRS] International Technology Roadmap for Semiconductors. [Online]: http://public.itrs.net/.

[JB08] S. Joshi, S. Boyd, "An Efficient Method for Large-Scale Sizing," *IEEE Transactions on Circuits and Systems – Part I*, vol. 55, no. 9, pp. 2760-2773, October 2008.

[JKK02] S. Jung, K. Kim, S. Kang, "Noise Constrained Transistor Sizing and Power Optimization for Dual V_t Domino Logic," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 10, no. 5, pp. 532-541, October 2002.

[K96] S. Kozu et al., "A 100MHz, 0.4W RISC Processor with 200MHz Multiply-Adder, Using Pulse-Register Technique," in Proc. of *IEEE International Solid-State Circuit Conference*, pp. 140-141, February 1996.

[KAD99] F. Klass, C. Amir, A. Das, K, Aingaran, C. Truong, R. Wang, A. Mehta, R. Heald, G. Yee, "A New Family of Semidynamic and Dynamic Flip-Flops with Embedded Logic for High-Performance Processors," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 5, pp. 712-716, May 1999.

[KB95] U. Ko, P. Balsara, "High Performance, Energy Efficient Master-Slave Flip-Flop Circuits," in Proc. of *IEEE Symposium on Low Power Electronics*, pp. 16-17, October 1995.

[KB00] U. Ko, P. Balsara, "High-Performance Energy-Efficient D-Flip-Flop Circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 8, no. 1, pp. 94-98, February 2000.

[KC01] J. Kao, A. Chandrakasan, "MTCMOS Sequential Circuits," in Proc. of *European Solid-State Circuits Conference*, pp. 317-320, September 2001.

[KKJ01] B. Kong, S. Kim, Y. Jun, "Conditional-Capture Flip-Flop for Statistical Power Reduction," *IEEE Journal of Solid-State Circuits*, vol. 36, no. 8, pp. 1263-1271, August 2001.

[KLL82] R. Krambeck, C. Lee, H. Law, "High-Speed Compact Circuits with CMOS," *IEEE Journal of Solid-State Circuits*, vol. 17, no. 3, pp. 614-619, June 1982.

[KS95] T. Kuroda, T. Sakurai, "Overview of Low-Power ULSI Circuit Techniques," *IEICE Transactions on Electronics*, vol. E78-C, no. 4, pp. 334-344, April 1995.

[LE90] S. Lu, M. Ercegovac, "A Novel CMOS Implementation of Double-Edge-Triggered Flip-Flops," *IEEE Journal of Solid-State Circuits*, vol. 25, no. 4, pp. 1008-1010, August 1990.

[LEW06] B. Lasbouygues, S. Engels, R. Wilson, P. Maurine, N. Azémard, D. Auvergne, "Logical Effort Model Extension to Propagation Delay Representation," *IEEE Transactions on Computer-Aided Design*, vol. 25, no. 9, pp. 1677-1684, September 2006.

[LS93] D. Liu, C. Svensson, "Trading Speed for Low Power by Choice of Supply and Threshold Voltage," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 1, pp. 10-17, January 1993.

[LS94] P. Larsson, C. Svensson, "Impact of Clock Slope on True Single Phase Clocked (TSPC) CMOS Circuits," *IEEE Journal of Solid-State Circuits*, vol. 29, no. 6, pp. 723-726, June 1994.

[LS96] R. Llopis, M. Sachdev, "Low Power, Testable Dual Edge Triggered Flip-Flops," in Proc. of *IEEE International Symposium on Low Power Electronics and Design*, pp. 341-345, August 1996.

[M01] A. Martin, "Towards an energy complexity of computation," *Information Processing Letters*, vol. 77, no. 2-4, February 2001.

[MB90] W. Madden, W. Bowhill, "High Input Impedance, Strobed CMOS Differential Sense Amplifier," *U.S. patent*, no. 4,910,713, March 1990.

[MC79] C. Mead, L. Conway, "Introduction to VLSI Systems," *Addison Wesley*, 1979.

[MCM97] [12] A. Mehta, Y. Chen, N. Menezes, D. Wong, L. Pileggi, "Clustering and Load Balancing for Buffered Clock Tree Synthesis," in Proc. of *IEEE International Conference on Computer Design*, pp. 217-223, October 1997.

[MDG97] J. Monteiro, S. Devadas, A. Ghosh, K. Keutzer, J. White, "Estimation of Average Switching Activity in Combinational Logic Circuits Using Symbolic Simulation," *IEEE Transactions of Computer-Aided Design of Integrated Circuits and Systems*, vol. 16, no.1, pp. 121-127, January 1997.

[MFG10] A. Morgenshtein, E. Friedman, R. Ginosar, A. Kolodny, "Unified Logical Effort - A Method for Delay Evaluation and Minimization in Logic Paths with RC Interconnect," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 18, no. 5, pp. 689-696, May 2010.

[MHA07] B. Mesgarzadeh, M. Hansson, A. Alvandpour, "Jitter Characteristic in Charge Recovery Resonant Clock Distribution," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 7, pp. 1618-1625, July 2007.

[MNB01] D. Markovic, B. Nikolic, R. Brodersen, "Analysis and design of Low-Energy Flip-Flops," in Proc. of *IEEE International Symposium on Low Power Electronics and Design*, pp. 52-55, August 2001.

[MSN04] D. Markovic, V. Stojanovic, B. Nikolic, M. Horowitz, R. Brodersen, "Methods for true energy-performance optimization," *IEEE Journal of Solid-State Circuits*, vol. 39, no.8, pp. 1282-1293, August 2004.

[MTD03] D. Markovic, J. Tschanz, V. De, "Transmission-Gate Based Flip-Flop," *U.S. patent*, no. 6,642,765, November 2003.

[MWA96] J. Montanaro, R. Witek, K. Anne, A. Black, E. Cooper, D. Dobberpuhl, P. Donahue, J. Eno, W. Hoeppner, D. Kruckemyer, T. Lee, P. Lin, L. Madden, D. Murray, M. Pearce, S. Santhanam, K. Snyder, R. Stehpany, S. Thierauf, "A 160-MHz, 32-b, 0.5-W CMOS RISC Microprocessor," *IEEE Journal of Solid-State Circuits*, vol. 31, no. 11, pp. 1703-1714, November 1996.

[N03] N. Nedovic, "Clocked Storage Elements for High-Performance Applications," *Ph.D. Dissertation*, University of California, Davis, 2003.

[N08] B. Nikolic, "Design in the power-limited scaling regime," *IEEE Transactions on Electron Devices*, vol. 55, no. 1, pp. 71-83, January 2008.

[NAO01] N. Nedovic, M. Aleksic, V. Oklobdzija, "Conditional Techniques for Low Power Consumption Flip-Flops," in Proc. of *IEEE International Conference on Electronics, Circuits and Systems*, vol. 2, pp. 803-806, February/May 2001.

[NAO02] N. Nedovic, M. Aleksic, V. Oklobdzija, "Conditional Pre-Charge Techniques for Power-Efficient Dual-Edge Clocking," in Proc. of *IEEE International Symposium on Low Power Electronics and Design*, pp. 56-59, August 2002.

[NC06] S. Narendra, A. Chandrakasan, "Leakage in Nanometer CMOS Technologies," *Springer*, 2006.

[NCF02] S. Naffziger, G. Colon-Bonet, T. Fischer, R. Riedlinger, T. Sullivan, T. Grutkowski, "The Implementation of the Itanium 2 Microprocessor," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, pp. 1448-1460, November 2002.

[NH02] S. Naffziger, G. Hammond, "The Implementation of the Next-Generation 64b ltaniumTM Microprocessor, " in Proc. of *IEEE International Solid-State Circuits Conference*, pp. 276-504, February 2002.

[NO98] M. Nogawa, Y. Ohtomo, "A Data-Transition Look-Ahead DFF Circuit for Statistical Reduction in Power Consumption," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 5, pp. 702-706, May 1998.

[NO00] N. Nedovic, V. Oklobdzija, "Dynamic Flip-Flop with Improved Power," in Proc. of *International Conference on Computer Design*, pp. 323-326, September 2000.

[NO05] N. Nedovic, V. Oklobdzija, "Dual-Edge Triggered Storage Elements and Clocking Strategy for Low-Power Systems," *IEEE Transactions on Very*

Large Scale Integration (VLSI) Systems, vol. 13, no. 5, pp. 577-590, May 2005.

[NOW03] N. Nedovic, V. Oklobdzija, W. Walker, "A Clock Skew Absorbing Flip-Flop," in Proc. of *IEEE International Solid-State Circuits Conference*, pp. 342-344, February 2003.

[NSO00] B. Nikolic, V. Stojanovic, V. Oklobdzija, W. Jia, J. Chiu, M. Leung, "Improved Sense-Amplifier-Based Flip-Flop: Design and Measurements," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 6, pp. 876-884, June 2000.

[NWO02] N. Nedovic, W. Walker, V. Oklobdzija, M. Aleksic, "A Low Power Simmetrically Pulsed Dual Edge-Triggered Flip-Flop," in Proc. of *IEEE European Solid-State Circuits Conference*, pp. 399-402, September 2002.

[NWO04] N. Nedovic, W. Walker, V. Oklobdzija, "A Test Circuit for Measurement of Clocked Storage Element Characteristics," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 8, pp. 1294-1304, August 2004.

[O02] V. Oklobdzija, "Clocking in Multi-GHz Environment," in Proc. of *International Conference on Microelectronics*, vol. 2, pp. 561-568, May 2002.

[O03] V. Oklobdzija, "Clocking and Clocked Storage Elements in a Multi-GigaHertz Environment," *IBM Journal of Research and Development*, vol. 47, no. 5/6, pp. 567-583, September/November 2003.

[OK06] V. Oklobdzija, R. K. Krishnamurthy, "High-Performance Energy-Efficient Microprocessor Design," *Springer*, 2006.

[ONB08] M. Orshansky, S. Nassif, D. Boning, "Design for Manufacturability and Statistical Design," *Springer*, 2008.

[OS01] V. Oklobdzija, V. Stojanovic, "Flip-Flop," *U.S. patent*, no. 6,232,810, May 2001.

[OSM03] V. Oklobdzija, V. Stojanovic, D. Markovic, N. Nedovic, "Digital System Clocking: High-Performance and Low-Power Aspects," *Wiley-IEEE Press*, 2003.

[OZD05] V. Oklobdzija, B. Zeydel, H. Dao, S. Mathew, R. Krishnamurthy, "Comparison of High-Performance VLSI Adders in the Energy-Delay Space," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no.6, pp. 754-758, June 2005.

[PBS96] H. Partovi, R. Burd, U. Salim, F. Weber. L. DiGregorio, D. Draper, "Flow-Through Latch and Edge-Triggered Flip-Flop Hybrid Elements," in

Proc. of *IEEE International Solid-State Circuit Conference*, pp. 138-139, February 1996.

[PCB01] H. Partovi, A. Chandrakasan, W. Bowhill, F. Fox, "Clocked Storage Elements," in *Design of High-Performance Microprocessor Circuits*, *IEEE Press*, pp. 207-234, 2001.

[PK] D. Patil, S. Kim, "Stanford Circuit Optimization Tool (SCOT) User Guide,"[Online]: www.stanford.edu/class/ee371/tools/SCOT_UserGuide.pdf

[PM02] P. Penzes, A. Martin, "Energy-Delay Efficiency of VLSI Computations," in Proc. of *ACM Great Lake Symposium on VLSI*, pp. 104-111, April 2002.

[PS99] J. Pangjun, S. Sapatnekar, "Clock Distribution Using Multiple Voltages," in Proc. of *IEEE International Symposium on Low Power Electronics and Design*, pp. 145-150, August 1999.

[R09] J. Rabaey, "Low Power Design Essentials," *Springer*, 2009.

[RCN03] J. Rabaey, A. Chandrakasan, B. Nikolic, "Digital Integrated Circuits: A Design Perspective – 2nd Edition," *Prentice Hall*, 2003.

[RDS94] P. Ramanathan, A. Dupont, K. Shin, "Clock Distribution in General VLSI Circuits," *IEEE Transactions on Circuits and Systems – Part I*, vol. 41, no. 5, pp. 395-404, May 1994.

[RMM03] K. Roy, S. Mukhopadhyay, H. Mahmoodi-Meimand, "Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits," *Proceedings of the IEEE*, vol. 91, no. 2, pp. 305-327, February 2003.

[SK05] S. Shin, B. Kong, "Variable Sampling Window Flip-Flops for Low-Power High-Speed VLSI," in Proc. of *IEE Circuits, Devices and Systems*, vol. 152, no. 3, pp. 266-271, June 2005.

[SN90] T. Sakurai, R. Newton, "Alpha-Power Law MOSFET Model and its Applications to CMOS Inverter Delay and Other Formulas," *IEEE Journal of Solid-State Circuits*, vol. 25, no. 2, pp. 584-594, April 1990.

[SN91] T. Sakurai, R. Newton, "Delay Analysis of Series-Connected MOSFET Circuits," *IEEE Journal of Solid-State Circuits*, vol. 26, no. 2, pp. 122-131, February 1991.

[SNC99] A. Strollo, E. Napoli, C. Cimino, "Low Power Double Edge-Triggered Flip-Flop Using One Latch," *Electronics Letters*, vol. 35 , no. 3, pp. 187-188, February 1999.

[SO99] V. Stojanovic, V. Oklobdzija, "Comparative Analysis of Master-Slave Latches and Flip-Flops for High-Performance and Low-Power

Systems," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 4, pp. 536-548, April 1999.

[SOA73] Y. Suzuki, K. Odagawa, T. Abe, "Clocked CMOS Calculator Circuitry," *IEEE Journal of Solid-State Circuits*, vol. 8, no. 6, pp. 462-469, December 1973.

[SS02] C. Saint, J. Saint, "IC Mask Design," *McGraw-Hill*, 2002.

[SSH98] I. Sutherland, B. Sproull, D. Harris, "Logical Effort: Designing Fast CMOS Circuits," *Morgan Kaufmann Publishers*, 1998.

[T02] Y. Taur, "CMOS design near the limit of scaling," *IBM Journal of Research and Development*, vol. 46, no. 2-3, pp. 213-222, March 2002.

[T03] Y. Tsividis, "Operation and Modeling of the MOS Transistor – 2nd edition," *Oxford University Press*, 2003.

[TFH11] C. Teh, T. Fujita, H. Hara, M. Hamada, "A 77% Energy-Saving 22-Transistor Single-Phase-Clocking D-Flip-Flop with Adaptive-Coupling Configuration in 40nm CMOS," in Proc. of *IEEE International Solid-State Circuits Conference*, pp. 338-340, February 2011.

[TKM88] K. Toh, P. Ko, G. Meyer, "An Engineering Model for Short-Channel MOS Devices," *IEEE Journal of Solid-State Circuits*, vol. 23, no.4, pp.950-958, August 1988.

[TN09] Y. Taur, T. Ning, "Fundamentals of Modern VLSI Devices – 2nd edition," *Cambridge University Press*, 2009.

[TNC01] J. Tschanz, S. Narendra, Z. Chen, S. Borkar, M. Sachdev, V. De, "Comparative Delay and Energy of Single Edge-Triggered and Dual Edge-Triggered Pulsed Flip-Flops for High-Performance Microprocessors," in Proc. of *IEEE International Symposium on Low Power Electronics and Design*, pp. 147-152, August 2001.

[U81] S. Unger, "Double-Edge-Triggered Flip-Flops," *IEEE Transactions on Computers*, vol. 30, no. 6, pp. 447-451, June 1981.

[VM95] A. Vittal, M. Marek-Sadowska, "Power Optimal Buffered Clock Tree Design," in Proc. of *Design Automation Conference*, pp. 497-502, June 1995.

[WH04] N. Weste, D. Harris, "CMOS VLSI Design: A Circuits and Systems Perspective – 3rd edition," *Addison Wesley*, 2004.

[WM09] C. Wang, D. Markovic, "Delay Estimation and Sizing of CMOS Logic Using Logical Effort with Slope Correction," *IEEE Transactions on Circuits and Systems – II: Express Briefs*, vol. 56, no.8, pp. 634-638, August 2009.

[WMC05] B. Wong, A. Mittal, Y. Cao, G. Starr, "Nano-CMOS Circuit and Physical Design," *Wiley Interscience*, 2005.

[Y74] L. Yau, "A Simple Theory to Predict the Threshold Voltage of Short Channel IGFETs," *Solid-State Electronics*, vol. 17, no. 10, pp. 1059-1063, October 1974.

[YS89] J. Yuan, C. Svensson, "High-Speed CMOS Circuit Technique," *IEEE Journal of Solid-State Circuits*, vol. 24, no. 1, pp. 62-70, February 1989.

[YS96] J. Yuan, C. Svensson, "New TSPC Latches and Flipflops Minimizing Delay and Power," in Proc. of *IEEE Symposium on VLSI Circuits*, pp. 160-161, June 1996.

[Z03] V. Zyuban, "Optimization of Scannable Latches for Low Energy," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 11, no. 5, pp. 778-788, October 2003.

[Z06] R. Zlatanovici, "Power-Performance Optimization for Digital Circuits," *Ph.D. Dissertation,* University of California, Berkeley, 2006.

[ZDB02] P. Zhao, T. Darwish, M. Bayoumi, "Low Power and High Speed Explicit-Pulsed Flip-Flops," in Proc. of *IEEE Midwest Symposium on Circuits and Systems*, pp. 477-480, August 2002.

[ZDB04] P. Zhao, T. Darwish, M. Bayoumi, "High-Performance and Low-Power Conditional Discharge Flip-Flop," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 5, pp. 477-484, May 2004.

[ZS02] V. Zyuban, P. Strenski, "Unified methodology for resolving power-performance tradeoffs at the microarchitectural and circuit levels," in Proc. of *IEEE International Symposium on Low Power Electronics and Design*, pp. 166-171, August 2002.

[ZS03] V. Zyuban, P. Strenski, "Balancing Hardware Intensity in Microprocessor Pipelines," *IBM Journal of Research and Development*, vol. 47, no. 5-6, pp. 585-598, September 2003.