Deep Submicron Technology: Opportunity or Dead End for Dynamic Circuit Techniques

Claas Cornelius[1] Frank Grassert[1] Siegmar Köppe[2] Dirk Timmermann[1]
[1]Institute of Applied Microelectronics and Computer Engineering, University of Rostock
[2]Advanced Systems and Circuits, Infineon Technologies AG
{claas.cornelius|dirk.timmermann}@uni-rostock.de; siegmar.koeppe@infineon.com

Abstract

Dynamic circuit techniques offer potential advantages over static CMOS, especially if more complex logic is to be implemented. Therefore, they are extensively used in high performance designs to speed up critical subsystems. However, the speed benefit is traded off for increased power consumption, area overhead, design effort, and reduced noise margins. The continuing process of technology scaling raises further concerns of reliability and limits the wide use of dynamic logic. This paper presents evaluations in terms of area, power dissipation, and propagation delay for several dynamic logic styles as well as for static CMOS in a 90 nm technology. The intention is to assess if dynamic circuit techniques are still an option to boost performance against the background of the issues of nanotechnology. Moreover, issues of reliability and signal integrity, gained from practical experience for different testbenches, and possible solutions are discussed. Finally, an automated design flow for dynamic logic, derived from a standard CMOS flow, is presented.

1. Introduction

Static Complementary Metal Oxide Semiconductor (CMOS) logic has been the engineer's first choice when designing complex integrated circuits. Automated design tools allow therewith the implementation of reliable, robust, and scalable systems with good cost-value ratio. Besides these properties, CMOS also features good performance in respect of propagation delay. However, dynamic circuit techniques outperform static CMOS noticeably at the price of increased power consumption and reduced noise margins [1-4]. Nonetheless, critical paths in subsystems such as arithmetic logic units, register files, and large multiplexers make extensive use of dynamic circuits to boost performance. A widespread representative of dynamic logic is the Domino logic [5] and its many derivatives. Intel Pentium 4, Sun UltraSPARC, and IBM PowerPC are commercial examples that employ Domino as well as AMULET, an asynchronous processor developed at the University of Manchester.

Reducing transistor dimensions has been an effective mechanism to cut down on propagation delay. But as the so called scaling continues, new effects show up and challenge engineers from production and design level [6]. Current technology approaches physical limits, as for instance the mean number of dopant atoms in the transistor's channel becomes less than 100. Alike, the gate's oxide thickness is already in the range of only a few layers of atoms. These and several more issues are the cause for parameter variations and result in diverse device behavior that has to be faced with new design methodology. To make the situation worse, scaling does not affect all system properties likewise. The shift of leakage currents as the dominant portion of total power consumption is one prominent example. The interconnects are another one that will be dominating the overall system performance. Furthermore, characteristics of integrated circuits are becoming of importance that earlier did not need to be considered so in-depth, like electromigration and crosstalk. With transistor and system behavior changing as well as steadily increasing costs for development and production to cope with these problems, it needs to be considered if the situation is an opportunity or a dead end for dynamic circuit techniques as an alternative to static CMOS.

On the one hand, the speed advantage of dynamic logic seems to be a possible opportunity that enables the use of older technology whereas the same system performance can be achieved as with standard CMOS in a newer one, though without the exponential increase of costs for production. But this requires design tools and automated design flows for dynamic logic to maintain the cost benefit during development and verification. For instance, clock distribution needs to be considered additionally, compared to static CMOS, and is the cause for area overhead and increased power consumption. On the other hand, concerns of reliability and the possible loss of the speed advantage might mark a dead end. Charge leakage, charge sharing, crosstalk, and power supply noise are problems that seriously endanger signal integrity of dynamic circuits. Srivastava et al. [4] identify several more issues and describe practical solutions that can not completely avoid but at least mitigate them. The second mentioned aspect, the speed advantage, might get lost because dynamic logic accomplishes its speed through reduced input capacitance. And this benefit might diminish when the wire capacitance dominates the overall load, making the input capacitance of relatively small importance. Both mentioned point of views have found their advocates. Anders et al. [3] predict that conventional Domino logic will not work past the 70 nm generation when the level of robustness shall be retained. In contrast thereto, Yang et al. [7] propose a modified Domino style to suppress subthreshold as well as gate leakage currents and present simulation results for a 32 bit adder in a 45 nm technology.

This paper examines, based on pre- and post-layout simulation results for a 90 nm technology, if dynamic logic is an alternative to static CMOS in respect of performance, power, and area. Moreover, aspects of signal integrity and requirements for an automated design flow are discussed. Section 2 introduces the examined dynamic circuit techniques and section 3 describes the simulation setup. The results are presented and discussed in section 4 before a final conclusion is drawn in section 5.

2. Dynamic Circuit Techniques

CMOS's performance suffers from the complementary structure and the massive use of pFETs with their decreased charge mobility, in contrast to nFETs. To achieve similar rise and fall times for pulling up and down the output, expanded gatewidths have to be used leading to high input capacitance and degraded performance. Dynamic circuit techniques avoid the large input capacitance by using a clocked pull-up transistor. Fig. 1 a) depicts a basic dynamic gate. The fundamental mode of operation can be divided into precharge and evaluation phase. The signal clk is low during precharge and the dynamic node X is charged high through transistor p1. When clk goes high evaluation starts and X can be discharged through the Pull Down Network (PDN) and transistor n1. If the PDN does not provide a path to ground, X will be floating, storing the high signal on the parasitic capacitance C_p. The setup denotes that only one transition from high to low can be made during evaluation and signal recovery due to charge loss of node X is not possible. Thus, the floating state makes the dynamic node X sensitive to noise and is the cause for possible malfunction. The generation and distribution of clock signals causes increased power consumption, area overhead, and design complexity. Hence, the use of dynamic gates pays especially off when complex functions are implemented so that the clock overhead becomes relatively smaller. When dynamic gates as in fig. 1 a) are cascaded, very exact timing is needed both for the gate delays and the clock scheme to avoid failure as well as increased power dissipation. However, such a successful setup can be achieved and results in very short propagation delays [8].

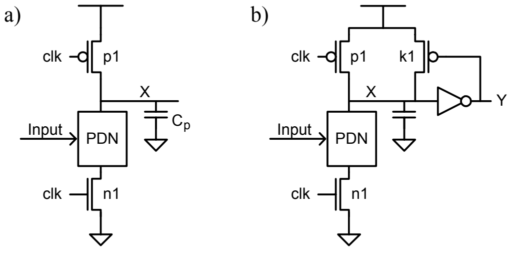

Fig. 1. a) Basic dynamic gate, b) Single-Rail Domino with a keeper k1 to compensate charge loss of dynamic node X

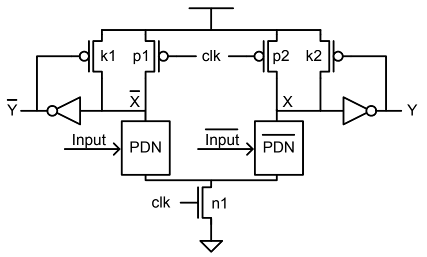

Fig. 2. DCVS-Domino provides differential outputs

In the following several dynamic logic styles are introduced which require input signals and their inverse but also provide differential outputs. These techniques are called dual-rail in contrast to static CMOS and conventional Domino which are so called single-rail techniques. The outputs of dual-rail gates are equal after precharge and become inversely when the evaluation is ready. The inverse output signals can be used to detect the evaluation's completion which is especially useful for asynchronous and self-timed circuits. The general disadvantage of differential structures is that one node is discharged every cycle independent of the input signals. Four of the presented circuit techniques belong to the Domino family which is the most widespread representative of dynamic logic. However, to present a fair and thorough comparison, three more techniques were chosen that exhibit interesting properties.

Single-Rail Domino (SR-Domino): A simple approach to assure correct operation in a chain of cascaded dynamic gates is to add an inverter to the output as shown in fig. 1 b). The inverter adds an additional gate delay but also drives the wire load as well as the input capacitance of the subsequent gate(s) and encapsulates the dynamic node X. This circuit technique is called Single-Rail Domino [5] and only non-inverting functions can be implemented which complicates the use in automated design flows. Another modification that is needed for practical use of Domino is the additional pFET k1 connected to the dynamic node X and the output Y resulting in an optional path to the supply voltage. This so called keeper can compensate charge loss when node X is floating. But the keeper is also the cause for short circuit currents and performance reduction because k1 is momentarily still turned on when the PDN starts to discharge the node X.

DCVS-Domino: SR-Domino's shortcoming of only non-inverting functions is overcome by the Domino derivate of the Differential Cascode Voltage Switch (DCVS) logic [9]. The gate structure can be understood as two complementary SR-Domino gates with a shared nFET clock transistor n1 (see fig. 2). Transistors of both PDNs of such a dual-rail gate can be used jointly to implement complex logic functions very efficiently. Chu et al. [10] present two techniques to construct such joint PDNs from Karnaugh maps and from tabular descriptions. For instance, XOR3 or the logic function a XOR (b + c • d) can be implemented with only 10 transistors in the PDNs.

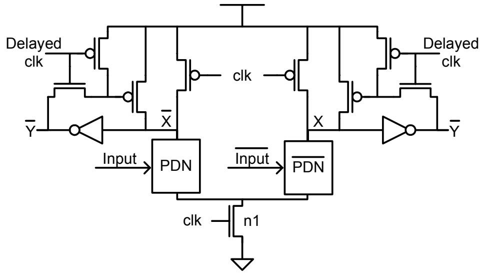

Fig. 3. High-Speed Domino with an extended keeper structure to avoid contention between the keepers and the PDNs

High-Speed Domino (HS-Domino): The introduced keeper causes short circuit currents and performance reduction as mentioned earlier. Allam et al. [11] propose High-Speed Domino to avoid the so called contention by turning the keeper on after the evaluation is ready. This is obtained by an extended keeper structure controlled by a delayed clock signal (see fig. 3). Indeed, the short circuit currents can be avoided but at the cost of increased area demand for the additional inverters as well as for routing and generating the delayed clock signal. Besides the extra clock signal also increases the power consumption.

Cross-Coupled Domino (XC-Domino): The fourth examined Domino derivative is Cross-Coupled Domino. The difference to DCVS-Domino is that the gates of the keepers are connected to the opposite dynamic node and not to the outputs of the corresponding inverters. Short circuit currents are avoided because the keepers are turned on after the evaluation is ready, in contrast to HS-Domino without any additional effort. An implementation of a NAND2 is shown in fig. 4. It needs to be mentioned that in the case of faulty timing, the dynamic nodes can possibly float when the evaluation has started (clk is high) and no valid input signals have arrived yet. All input signals will be low and the dynamic nodes are neither connected to ground nor to the supply voltage. This scenario also defines a minimum clock frequency to guarantee that the dynamic nodes are precharged again to the supply voltage and charge loss does not cause malfunction.

Cross-Coupled Differential (XC-Differential): The basic structure of Cross-Coupled Differential is derived from XC-Domino by omitting the inverters at the outputs (see fig. 4). The output X and its inverse are charged high during precharge leading to conducting transistors in the PDNs of the succeeding gate. This state causes signal loss, increased power dissipation, or even malfunction. But the problems can be faced by applying a clock scheme so that each gate in a chain is supplied with a clock signal that is delayed from its predecessor [8]. The so called clock delay has to be adjusted accurately and is very critical to achieve high-performance and to avoid increased power consumption due to overlapping precharge and evaluation phases of consecutive gates.

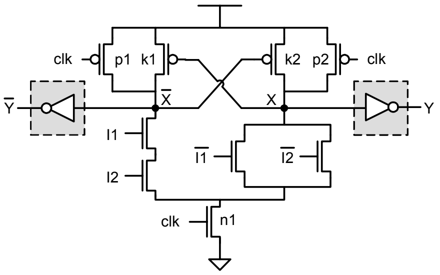

Fig. 4. XC-Domino uses inverters whereas XC-Differential avoids them but needs adjusted clock delay to work properly

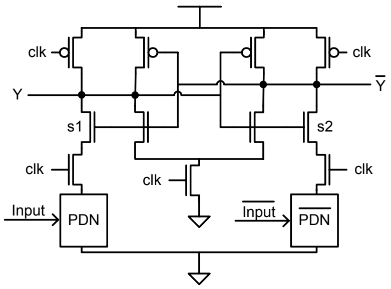

Fig. 5. Differential Current Switch Logic (DCSL)

Differential Current Switch Logic (DCSL): The DCSL circuit technique was proposed by Somasekhar et al. [12] and can be described as a sense amplifier. Fig. 5 presents such a gate with two cross-coupled inverters, complementary PDNs, and precharge transistors. What makes this logic different from the introduced ones is that the dynamic nodes are not fully discharged by the PDNs but by the inverters which makes the performance quite insensitive to stack depth. The outputs are charged high during precharge and both inverters try to discharge the dynamic nodes when the evaluation starts. One and only one of the inverters will be supported by a conducting PDN resulting in an even faster discharge. This tendency is amplified by the inverter so that the opposite inverter is finally turned off. In addition, the opposite PDN is also switched off by the transistor s1 or s2, respectively. DCSL offers the potential for very short propagation delays but the tendency for discharge can be corrupted by charge sharing, crosstalk, and even asymmetric wire loads at the complementary outputs. Moreover, the clock load is high due to the need of a delayed clock scheme and 5 clock transistors per gate.

Sympathetic Precharged Static Domino (SPSD): SPSD is also a sense amplifying logic and insensitive to stack depth just as DCSL. But in contrast to all other introduced techniques, the evaluation takes place when clk is low. Fig. 6 presents such a gate as proposed by Gayles et al. [13]. The precharge phase is actually a predischarge so that the outputs are discharged to ground. Both outputs are then charged during evaluation whereas one of the PDNs will be conducting, resulting in a temporary short circuit current from VDD to ground and a slow down of the charge process of the corresponding node. This tendency is amplified and finally one of the cross-coupled transistors p1 and p2 is turned off, which eliminates the short circuit current, and the outputs become differential. SPSD offers reduced clock load in comparison to DCSL but is also very sensitive to the timing of clock and data signals. Besides, the temporary short circuit current has negative influence on the power consumption.

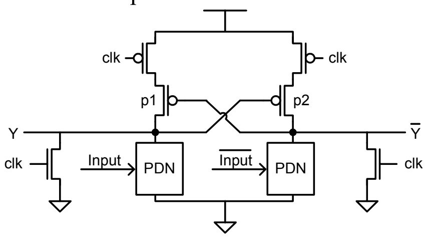

Fig. 6. Sympathetic Precharged Static Domino (SPSD) predischarges the outputs and evaluates when clk is low

3. Simulation Setup

Most circuit techniques can be accelerated in a wide range by simply applying larger gate widths to the transistors. But the applicability of the physical layout limits the gate widths to realistic values. For this reason we limited the maximum gate width to 25 times the minimum gate width for pFETs as well as nFETs and defined the maximum wire load to be equivalent to routing across 125 average cells of the corresponding circuit technique. To reproduce the worst-case scenario under the given constraints, we looked for a first testbench that should be easy to validate. As the delay of a single gate is not a good measure to compare circuit techniques we used a chain of six NAND4 and NOR4 gates connected alternately as inverters with the specified maximum wire load. In this case the output of each gate has to be discharged via the series of transistors representing the critical path (e.g. via I1 and I2 for the NAND2 in fig. 4). SR-Domino can not implement inverting functions so that we set up a comparable testbench of AND4 and OR4 gates for this circuit technique. To have another more realistic testbench with varying wire loads and fan-in, we designed a 4x4 bit Wallace tree multiplier. Both testbenches were surrounded by registers for the input and output signals to consider a setup in a pipeline structure or the interface to static logic, respectively.

All simulations were performed for the 90 nm technology from Infineon Technologies AG. The transistors were of type Low Threshold Voltage (LVT) to achieve high performance and to better observe the influence of noise as the cause for failure. Additional parasitic capacitances, e.g. intra-cell interconnects and overlapping diffusions were added to the schematics to map the behavior of the layout as accurate as possible. All circuit techniques were dimensioned similarly with balanced rise and fall times to gain a fair comparison. For instance, transistors accelerating the evaluation were sized to the defined upper limit, the clock transistors were sized with equal driver strength as the PDNs, and the keepers were initially of minimum size. The most promising circuit techniques were finally also implemented in the layout to perform post-layout simulations and to validate the results on a test chip. For instance, the layout of the worst-case scenario and the 4x4 bit Wallace tree multiplier in DCVS-Domino are depicted in fig. 7. The structures to the left and right are the registers. In addition, the regular structures on top and bottom of fig. 7 a) are the wire loads that are implemented by fingered and stacked metal layers to achieve a constant capacitance in contrast to the more area-effective gate capacitance.

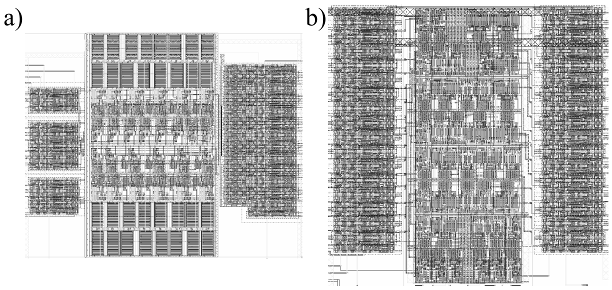

Fig. 7. Layout of a) worst-case scenario and b) Wallace tree mutiplier

4. Results

This section presents and discusses the results for the introduced circuit techniques and testbenches. The pre- and post-layout simulations did not differ much as accurate parasitic capacitance models were already included in the schematics. Thus, there is no need for a separate discussion. All data shown were obtained under the maximum achievable frequency and a fixed adjusted clock delay. The values for the area, given in table I, were derived from the accumulated cell sizes for the chain of NAND-NOR gates. But this testbench does not account for the possibility of differential gates to effectively implement complex functions or the fractional assignment of single-rail gates. The Wallace tree multiplier is an example better suited for differential circuits so that the area in this case was nearly the same as for static CMOS but values of only 85 % area demand have also been reported [2].

4.1. Performance and Power

Fig. 8 depicts the results of the NAND-NOR chain for the various examined circuit techniques and table I resumes the results in comparison to static CMOS. The diagram represents the Power-Delay-Product (PDP) over the reciprocal of the delay to be able to set the power consumption in relation to the delay. So a result far right and down represents a high-performance circuit technique consuming a disproportionably amount of power.

The first observation shows that all examined circuit techniques outperform static CMOS by a factor of 2 to 3 at the price of increased power consumption. The differential Domino techniques do not deviate much when the keepers are kept at minimum size. The delay as well as the PDP only varies less then 2.3 %. But when the keepers are enlarged to five times the minimum size, the techniques differ up to 12 % and 7.3 % for the delay and the PDP, respectively. Such an extension of the keepers could be necessary to guarantee correct operation and to mitigate signal integrity problems. Then, XC-Domino is the best choice showing similar performance as HS-Domino but with reduced power consumption. SR-Domino also offers similar performance but has to pay off, just as DCVS-Domino, for the short circuit currents caused by the

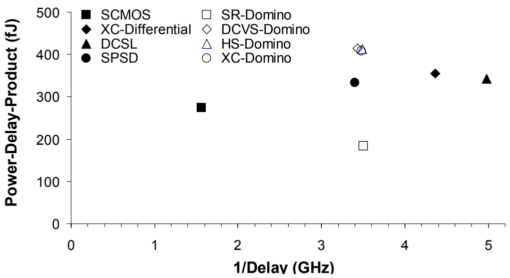

Fig. 8. Results for the worst-case scenario, measured for each corresponding maximum achievable frequency

TABLE I. COMPARISON OF VARIOUS DYNAMIC CIRCUIT TECHNIQUES IN A WORST-CASE SCENARIO VERSUS STATIC CMOS

Circuit technique	Area	Power dissipation	Delay	Power-Delay-Product	Area-Delay-Product
Static CMOS	1	1	1	1	1
SR-Domino	0.913	1.524	0.438	0.667	0.399
DCVS-Domino	1.826	3.378	0.446	1.508	0.815
HS-Domino	1.911	3.432	0.439	1.506	0.838
XC-Domino	1.826	3.385	0.440	1.490	0.803
XC-Differential	1.302	3.688	0.351	1.295	0.457
DCSL	1.598	4.061	0.307	1.249	0.491
SPSD	1.154	2.597	0.468	1.216	0.540

conventional keeper in terms of delay and power consumption. Nevertheless, the power advantage of SR-Domino in comparison to the differential Domino circuits is significant due to the single-rail structure and the reduced load. In the same range of performance is SPSD with the power dissipation in between single-rail and differential Domino techniques. This is originated from the number of transistors and correlates with the capacitive load. Unfortunately, the evaluation delay is mainly determined by two pFETs connected in series which suffer from reduced charge mobility in contrast to nFETs. The two best performing circuit techniques are XC-Differential and DCSL. The obvious explanation for the performance gain of XC-Differential is the missing inverters which reduces propagation delay as well as power consumption. Finally, DCSL outperforms all other presented techniques in terms of delay. The speed benefit comes from the sense amplifying nature, but unlike SPSD the evaluation delay is determined by two nFETs connected in series. DCSL's large number of transistors per gate causes the highest power dissipation (see table I) but this is disproportionably to the achieved delay so that the PDP is noticeably better compared to the differential Domino techniques.

Two possible optimizations can further be deployed. The first one is to omit Domino's nFET clock transistor n1 resulting in up to 15 % performance gain at the price of 12 % power increase in average systems. The difference comes from the reduced stack depth and short circuit currents during overlapping precharge and evaluation phases of consecutive gates. The second one is to omit DCSL's transistors s1 and s2. Thus, stack depth is shortened and propagation delay as well as power dissipation is reduced by 5 % and 10 %, respectively. However, the values for the power dissipation in table I do not consider the adjusted clock tree so that extra 18 to 27 % have to be added with Domino at the bottom and DCSL at the top of the given range. These figures are based on the gate's input capacitances and do not account for the general clock scheme, distribution, number of connected registers, and accurateness which is in part also needed for static CMOS. One more thing to keep in mind is that all values were derived for highly active systems. Power consumption of dynamic gates is dependent on the state of input signals and not on the switching probability as static CMOS. Therefore, dynamic power is consumed even though no new evaluation is performed. That is why additional action has to be taken to avoid unnecessary power loss during long idle phases. Clock gating or dynamic frequency scaling are possible solutions. The problem of long idle phases gets worse when leakage currents will in the future dominate the overall power consumption. As a rule of thumb it can be said that such systems will favor circuit techniques with smaller number of transistors or smaller active area, respectively.

4.2. Reliability and signal integrity

There is a wide field of issues endangering reliability and signal integrity of dynamic circuits but in part also static CMOS: leakage currents, clock skew, power supply noise, crosstalk, and charge sharing to name just a few. Clock skew does not affect Domino logic much because the data ripple independently through the design as long as the gates are in evaluation mode. But it can severely corrupt the functionality of timing critical circuits such as DCSL, SPSD, and XC-Differential where the time frame for arriving clock and data signals is rather narrow. Even unequal signal arrival times at the inputs or unbalanced loads at the differential outputs can cause failure. Additional buffers to balance different path delays and matched output loads might have to be inserted to solve these problems. Keepers were introduced to compensate charge loss of dynamic nodes. Admittedly, they are useful for countervailing small and slow charge loss due to for instance capacitive coupling via the miller capacitance or leakage currents. However, keepers are not strong enough to compensate fast signal loss. Charge sharing is such an example where the charge of a dynamic node is distributed onto various parasitic capacitances in the PDN resulting in signal degradation. Possible solutions are to precharge internal nodes of the PDN or to apply additional control logic that enables the conventional setup to precharge the internal nodes as well. Issues that can not be solved so easily are temporary low-resistive paths. Such paths can arise when crosstalk on input wires lifts the gate voltage above the threshold. Power supply noise can have the same effect. There are further issues like α particles, substrate charge injection, and more that can not be covered briefly. Srivastava et al. [4] give a comprehensive overview of those and other issues and point out several methods to increase robustness.

As it can be observed by the diverse publications related to this topic, it is difficult to define a distinct line where issues of reliability only limit or already prohibit the use of dynamic circuit techniques [3,4,7]. A multitude of ideas has been proposed which trade off reliability for increased area, power consumption, system delay, or design effort.

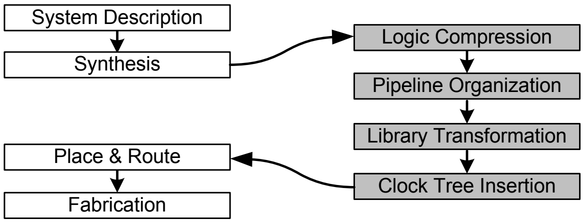

Fig. 9. Design flow for dynamic logic derived from a standard CMOS flow

However, there is no doubt that deep submicron technologies have significant influence on reliability and have put forth even more serious issues that need to be observed when designing integrated circuits. This applies to static CMOS but especially to the more susceptible dynamic circuit techniques.

4.3. Design Flow

Complex integrated systems can only be designed with an acceptable cost-value ratio when automated design tools are used. Unfortunately, commercial products focus on static CMOS and only sporadic tools support particular design steps for dynamic circuits. In the following, a design flow for dynamic logic is discussed which is derived from a standard CMOS flow [14]. Fig. 9 depicts such a possible design flow. A more detailed description with for instance iterative loops, simulations, and backannotation is left out for simplification. The entry point is a conventional system description that is synthesized onto a standard CMOS library. As the synthesis is optimized for static CMOS (e.g. small fan in is favored), logic compression is needed to substitute logic functions that are not supported by the dynamic target library and to exploit possible advantages. Afterwards, a pipeline organization is performed. This step is needed to balance different parallel data paths for timing critical circuit techniques. But aspects of pipelining with buffer and register insertion should also be handled. The library transformation maps the netlist onto the dynamic target library and adds connections for dual-rail structures and possible control signals (e.g. reset). Finally, a clock tree has to be inserted and connected. Timing critical circuits like DCSL also need matched output capacitances to avoid malfunction. Hence, additional iterations of place and route or manual engagement are needed.

The required algorithms are very complex and have to handle a multitude of degrees of freedom. Hence, it needs corporate endeavors to assemble existing partial solutions or to develop a completely new framework. Due to the facts that such a framework will probably only be unrestrictedly applicable for a single dynamic circuit technique and the extremely high development effort, it is questionable whether such a framework will be available at any time.

5. Conclusion

Dynamic circuit techniques are still a promising option to boost performance because they clearly outperform static CMOS, especially when implementing complex logic functions. However, the speed benefit comes at the price of increased power consumption and area overhead. Furthermore, reliability is severely endangered by more and more deep submicron effects so that the electrical integrity has to be costly monitored and verified during the design process. For these reasons and the lack of existing automated design tools, dynamic circuits will in the future not be an option for large integrated systems nor synthesized top-down approaches. Nevertheless, critical units in ultra high speed applications will require dynamic circuits to achieve the target performance. This can be accomplished in the future deep submicron era with full custom and assisted design flows when the related constraints are considered.

Acknowledgment

The authors gratefully acknowledge Winfried Kamp from Infineon Technologies AG for his technical assistance and helpful suggestions.

References

[1] K. Chu and D. Pulfrey, “A comparison of CMOS circuit techniques: Differential cascode voltage switch logic versus conventional logic,” IEEE Journal SSC, vol. 22, pp. 528-532, Aug 1987

[2] P. Ng, P. Balsara, and D. Steiss, “Performance of CMOS differential circuits,” IEEE Journal SSC, vol. 31, pp. 841-846, Jun 1996

[3] M. Anders, R. Krishnamurthy, R. Spotten, and K. Soumyanath, “Robustness of sub-70nm dynamic circuits: Analytical techniques and scaling trends,” Symp. VLSI Circuits, vol. 3, pp. 23-24, Jun 2001

[4] P. Srivastava, A. Pua, and L. Welch, ”Issues in the design of Domino logic circuits,” GLSVLSI, pp. 108 112, 1998

[5] R. Krambeck, C. Lee, and H. Law, “High-Speed compact circuits with CMOS,” IEEE Journal SSC, vol. 17, pp. 614 619, Jun 1982

[6] A. Allan et al., “2001 Technology Roadmap for Semiconductors,” in Computer, vol. 35, pp. 42 53, 2002

[7] G. Yang, Z. Wang, S. Kang, “Leakage-proof domino circuit design for deep sub-100nm technologies,” VLSID, pp. 222 227, Jan 2004

[8] G. Yee and C. Sechen, “Clock-delayed Domino for adder and combinational logic design,” ICCD, pp. 332-337, Oct 1996

[9] L. Heller, W. Griffin, J. Davis, and N. Thoma, “Cascode voltage switch logic: A differential CMOS logic family,” ISSCC, pp. 16-17, Feb 1984

[10] K. Chu and D. Pulfrey, “Design procedures for differential cascode voltage switch circuits," IEEE Journal SSC, vol. 21, pp. 1082–1087, Dec 1986

[11] M. Allam, M. Anis, and M. Elmasry, “High-speed dynamic logic styles for scaled-down CMOS and MTCMOS technologies,” ISLPED, pp. 155-160, 2000

[12] D. Somasekhar and K. Roy, “Differential current switch logic: A low power DCVS logic family,” IEEE Journal SCC, vol. 31, pp. 981 991, Jul 1996

[13] E. Gayles, K. Acken, R. Owens, and M. Irwin, “A clocked, static circuit technique for building efficient high frequency pipelines,” GLSVLSI, pp. 182 187, 1997

[14] S. Flügel et al., “A design flow for 12.8 GBit/s triple DES using dynamic logic and standard synthesis tools,” SNUG, 2001