

- 3) On the average, the error of BELT compared to VG4 is 5.2%; while on the average, the VG4 delay is almost a factor of three higher than that predicted by ELT.
- 4) Compared to SPICE-based VG4, the error of BELT is 9% on the average.

These experiments illustrate that our estimation technique is sufficiently accurate for design planning, while ignoring the blockages is prohibitively costly.

Finally, we note how efficient the estimation technique is. The total runtime in seconds for running the 13 test cases mentioned above was 0.24, 23.0, and 29.0 for BELT, VG1, and VG4, respectively. In other words, BELT is about 100 times faster than running an actual buffer-insertion algorithm. For medium sized nets such as n786 and n869, the runtime of VG4 plus the SPICE simulation is over 10 000 times slower than the runtime of BELT.

## VI. CONCLUSION

We presented closed-form formulas for estimating the achievable buffered delay when buffering restrictions exist in the layout. We demonstrated that adding blockages to the layout could cause significant error in estimation techniques that ignore the blockage terrain. We also showed that our technique is a lower bound, has an error of less than one percent for two-pin nets, and has only a few percent error for multisink nets.

## REFERENCES

- [1] C. J. Alpert and A. Devgan, "Wire segmenting for improved buffer insertion," in *Proc. ACM/IEEE Design Automation Conf. (DAC)*, Anaheim, CA, 1997, pp. 588–593.
- [2] C. J. Alpert, G. Gandham, M. Hrkic, J. Hu, A. B. Kahng, J. Lillis, B. Liu, S. T. Quay, S. S. Sapatnekar, and A. J. Sullivan, "Buffered Steiner trees for difficult instances," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 21, no. 1, pp. 3–14, Jan. 2002.
- [3] C. C. N. Chu and D. F. Wong, "Closed form solution to simultaneous buffer insertion/sizing and wire sizing," in *Proc. ACM/IEEE Int. Symp. Physical Design*, Napa, CA, 1997, pp. 192–197.
- [4] J. Cong, L. He, C.-K. Koh, and P. H. Madden, "Performance optimization of VLSI interconnect layout," *Integr. VLSI J.*, vol. 21, no. 1/2, pp. 1–94, Nov. 1996.
- [5] J. Cong, T. Kong, and D. Z. Pan, "Buffer block planning for interconnect-driven floorplanning," in *Proc. IEEE/ACM Conf. Computer-Aided Design*, San Jose, CA, 1999, pp. 358–363.
- [6] J. Cong and D. Z. Pan, "Interconnect performance estimation models for design planning," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 20, no. 6, pp. 739–752, Jun. 2001.
- [7] J. Cong and X. Yuan, "Routing tree construction under fixed buffer locations," in *Proc. ACM/IEEE Design Automation Conf. (DAC)*, Los Angeles, CA, 2000, pp. 379–384.
- [8] S. Dhar and M. A. Franklin, "Optimum buffer circuits for driving long uniform lines," *IEEE J. Solid-State Circuits*, vol. 26, no. 1, pp. 33–38, Jan. 1991.
- [9] W. C. Elmore, "The transient response of damped linear network with particular regard to wideband amplifiers," *J. Appl. Phys.*, vol. 19, no. 1, pp. 55–63, Jan. 1948.
- [10] M. Hrkic and J. Lillis, "Buffer tree synthesis with consideration of temporal locality, sink polarity requirements, solution cost and blockages," in *Proc. ACM/IEEE Int. Symp. Physical Design (ISPD)*, Del Mar, CA, 2002, pp. 98–103.
- [11] J. Hu, C. J. Alpert, S. T. Quay, and G. Gandham, "Buffer insertion with adaptive blockage avoidance," *IEEE Trans. Comput.-Aided Des.*, vol. 22, no. 4, pp. 492–498, Apr. 2003.
- [12] R. H. J. M. Otten, "Global wires harmful?" in *Proc. ACM/IEEE Int. Symp. Physical Design*, Monterey, CA, 1998, pp. 104–109.
- [13] P. Saxena, N. Menezes, P. Cocchini, and D. A. Kirkpatrick, "Repeater scaling and its impact on CAD," *IEEE Trans. Comput.-Aided Des.*, vol. 23, no. 4, pp. 451–463, Apr. 2004.
- [14] X. Tang, R. Tian, H. Xiang, and D. F. Wong, "A new algorithm for routing tree construction with buffer insertion and wire sizing under obstacle constraints," in *Proc. IEEE/ACM Int. Conf. Computer-Aided Design*, San Jose, CA, 2001, pp. 49–56.
- [15] L. P. P. van Ginneken, "Buffer placement in distributed RC-Tree network for minimal Elmore delay," in *Proc. Int. Symp. Circuits and Systems*, New Orleans, LA, 1990, pp. 865–868.

## Clock-Skew-Optimization Methodology for Substrate-Noise Reduction With Supply-Current Folding

Mustafa Badaroglu, Kris Tiri, Geert Van der Plas, Piet Wambacq, Ingrid Verbauwhede, Stéphane Donnay, Georges G. E. Gielen, and Hugo J. De Man

**Abstract**—In a synchronous clock distribution network with negligible skews, digital circuits switch simultaneously on the clock edge; therefore, they generate a lot of substrate noise due to the resulting sharp peaks on the supply current. A solution is to split a large design in different clock regions and introduce intentional clock skews between them, while taking the timing constraints into account. In this paper, the authors present a complete design flow to optimize the clock tree for less substrate-noise generation in large digital systems. It proposes a technique to assign combinatorial cells and flip-flops to the clock regions. It also takes into account the impact of unintentional clock skew such as jitter on the computed skews in order to assure a robust design. During the optimization, it uses compressed supply-current profiles to improve the CPU time. Experimental results show more than a factor-of-2 reduction in substrate-noise generation from large digital circuits of which the skews are optimized.

**Index Terms**—Clock skew, deep submicrometer, ground bounce, low-noise digital design, mixed analog–digital ICs, power modeling and estimation, substrate noise.

## I. INTRODUCTION

Substrate noise is a major obstacle for mixed-signal integration, where it is mainly caused by switching-induced ground bounce in the digital domain [1]. In large digital circuits, high peaks on the supply current create power-supply noise in the supply network. The part of this noise on the  $V_{SS}$  rail is the so-called ground bounce. Decreasing the peak and the slope of the supply current caused by the switchings of digital circuits will reduce the ground-bounce generation and, therefore, the substrate noise. Such decrease can be obtained by introducing different skews to a clock network driving a synchronous digital system, creating different clock regions (Fig. 1).

The use of intentional clock skew for noise reduction has been previously reported for reducing the peak current [2] and the ground bounce [3], [4]. In prior approaches, the skew of every individual flip-flop was optimized (fine-grained skewing) without considering the

Manuscript received July 1, 2004; revised December 10, 2004 and February 18, 2005. This paper was recommended by Associate Editor D. Blaauw.

M. Badaroglu, G. Van der Plas, P. Wambacq, and S. Donnay are with the Interuniversity Microelectronics Center (IMEC), B-3001 Leuven, Belgium (e-mail: badar@imec.be).

K. Tiri was with University of California, Los Angeles (UCLA), Los Angeles, CA 90095 USA, and is now with NVIDIA, Santa Clara, CA 95050 USA.

I. Verbauwhede is with Departement Elektrotechniek (ESAT), K.U. Leuven, B-3001 Leuven, Belgium, and also with UCLA, Los Angeles, CA 90095 USA.

G. G. E. Gielen is with ESAT, K.U. Leuven, B-3001 Leuven, Belgium.

H. J. De Man is with IMEC, B-3001 Leuven, Belgium, and also with ESAT, K.U. Leuven, B-3001 Leuven, Belgium.

Digital Object Identifier 10.1109/TCAD.2005.855952

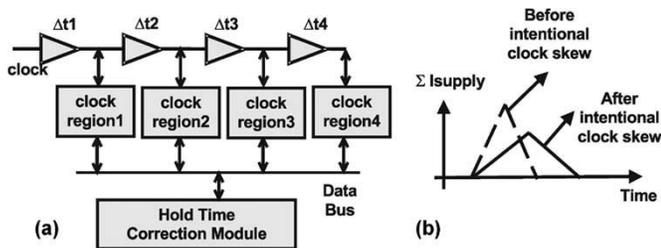


Fig. 1. (a) Implementation of a clock tree with different skews and (b) construction of the supply-current waveform under different skews.

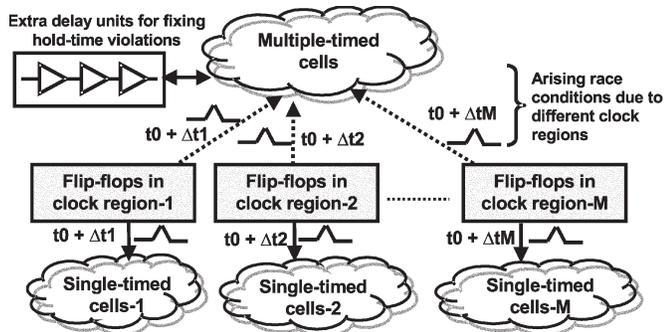


Fig. 2. Distribution of single-timed and multitimed cells across several clock regions.

communication power penalty between many clock regions due to increasing number of switchings and due to extra delay buffers for fixing the hold-time violations (Fig. 2). The number of switchings can be higher if the timing difference between clock regions is substantially large, i.e., large skew values are needed for reducing the ground bounce. In this case, multiple-timed cells toggle more than necessary, therefore, cause an increase in power. Power penalty can also be severe in designs that require large skew values when we implement the skew of a particular flip-flop by means of delay units. Fine-grained skewing may also require many iterations during the skew-optimization process since the fidelity of the results cannot be guaranteed if the switching behavior of the circuit changes as a result of the new skew values. Limiting the number of clock regions significantly reduces CPU and memory requirements during the skew-optimization process, since, for a large digital system, the required number of clock regions is substantially less than the total number of the flip-flops. For all these reasons, we need a clustering algorithm that groups the flip-flops into clock regions before the optimization. For each clock region, a small amount of skew spread can still be allowed in order to minimize the number of clock buffers driving the flip-flops [5]. This could be an extension of our approach. Up to now, none of the previous skew-optimization approaches [2], [3] gives a minimum (required) value for the number of clock regions, which is set by the relation between the major resonance frequency of the circuit and the rise/fall time of the supply current. In this paper, we consider this effect to design the optimal clock network [6]. In fact, we will show that after a required number of clock regions, no further reduction of the ground bounce is achieved. We also use representative supply-current profiles to optimize the clock skews since the use of the total transient data is not feasible for the clock-skew optimization [6], [7].

The cost function used for the optimization of the clock skews also yields more accurate results as compared to previous work, which uses some mathematical functions based on a triangular approximation of the supply current [2], [6]. For synchronous systems, the spectrum of the substrate noise has peaks at multiples of the digital clock frequency. These peaks are shown to be the dominant components of

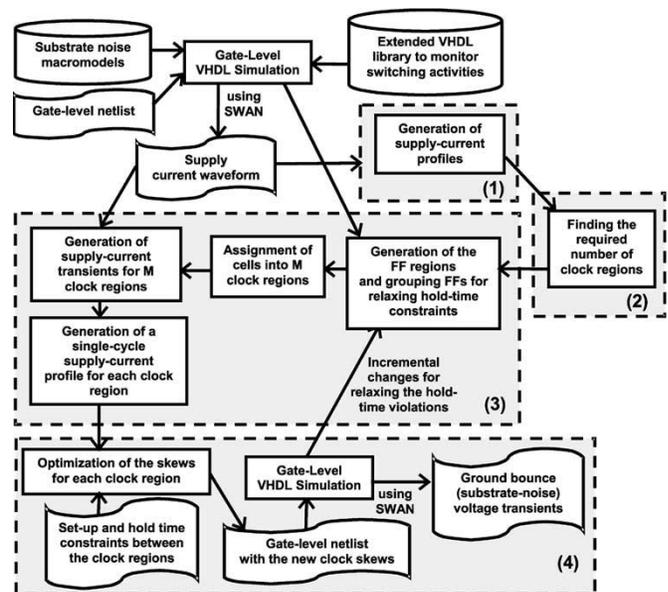


Fig. 3. Clock-skew-optimization methodology [the numbers 1)–4) refer to the different steps explained in the text].

the total supply current [7], [8]. As a result, the optimization should be performed at each clock harmonic on the constraint space formed by the skews. This gives more optimal results. Additional constraints such as performance/race reliabilities of the clock are also introduced in the optimization in order to have a clock tree tolerant to process variations, which have also been addressed [9].

Two techniques to reduce digital substrate-noise generation have been demonstrated in [7]: clock frequency modulation and intentional clock skews. The experimental validation of the use of intentional clock skews in order to compare its effectiveness with the use of other low-noise digital-design techniques such as decoupling has been presented in [8]. With respect to frequency modulation, in [7] we focus on the frequency-spectrum analysis of the supply current in an effort to find the optimum settings for the modulating waveform. In this paper, we particularly describe the clock-skew-optimization methodology where we highlight the following of our contributions: folding the supply current, determining the required number of clock regions, clustering the clock regions, and optimizing the clock skews.

## II. CLOCK-SKEW OPTIMIZATION FOR REDUCTION OF SUBSTRATE NOISE

The clock-skew-optimization methodology consists of four major steps: 1) supply-current folding (Section II-A); 2) finding a minimum (required) number of clock regions (Section II-B); 3) assignment of the digital cells to the clock regions (Section II-C); and 4) clock-skew optimization (Section II-D). The methodology flow is given in Fig. 3 with an indication of the main steps listed above. The different steps are now described in detail.

### A. Supply-Current Folding

For run-time efficiency of the clock-skew-optimization methodology, it is not possible to use the complete transient data of the supply current over a long time period. In this section, we present an algorithm to generate the representative supply-current profile(s) for each of the  $M$  current waveforms, where  $M$  is the number of clock regions. Each clock cycle is discretized into  $N$  time intervals. For synchronous CMOS circuits, each cycle of the supply current in the time domain can be approximated by a triangular waveform (Fig. 4) where  $I_p$ ,  $t_r$ ,

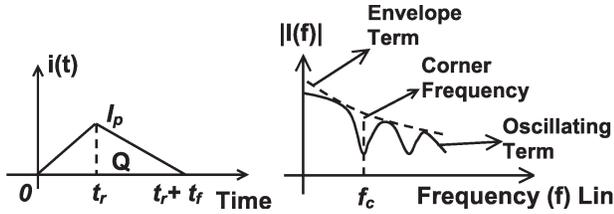


Fig. 4. Supply current in the time and frequency domains.

and  $t_f$  are the peak value, the rise time, and the fall time, respectively. The presented algorithm assures a certain maximum error bound on the parameters  $I_p$ ,  $t_r$ , and  $t_f$  of the system supply current constructed by using the supply-current profile(s) with respect to each cycle of the total supply current of each clock region. These representative supply-current profiles will then be used to optimize the clock skews of the  $M$  clock regions.

We define  $i[n, m, r]$  as the actual value of the supply current at the time point  $n(1, \dots, N)$ , the clock region  $m(1, \dots, M)$ , and the clock cycle  $r(1, \dots, R)$ . For each clock region, the union of the  $i[n, m, r]$  values is compressed into a set of supply-current profiles ( $i_p[n, m]$ ), each having a single clock-cycle representation.  $p$  is indexing into the set of supply-current profiles ( $IP(m)$ ) in the  $m$ th clock region.  $IP(m)$  is given by

$$IP(m) = \{ \{i_1[n, m], k_1\}, \{i_2[n, m], k_2\}, \dots, \{i_p[n, m], k_p\}, \dots \} \quad \text{for } n = 1 \dots N \quad (1)$$

where  $k_p$  is the number of clock cycles used for the construction of each supply-current profile  $i_p[n, m]$ . The number of supply-current profiles in each clock region is constrained to be the same and it is dependent on the allowed error on the parameters of the total supply current. Fig. 5 shows the folding procedure to generate the supply-current profiles.

We compute the parameters  $I_p(r)$ ,  $t_r(r)$ , and  $t_f(r)$ , which are the peak value, the rise time, and the fall time, respectively, at the clock cycle  $r(1, \dots, R)$  using the system supply current. We define  $\nu_p$ ,  $\nu_r$ , and  $\nu_f$  as the maximum allowed percentage variations of  $I_p(r)$ ,  $t_r(r)$ , and  $t_f(r)$ , respectively, of the actual supply current relative to the system supply current constructed by using the supply-current profile(s). Using these error bounds, we try to find the supply-current profile set that best matches the actual system supply current at the clock cycle  $r$ . If no such profile set exists, then an additional supply-current profile for every clock region is generated using the actual supply current at the clock cycle  $r$ . The user-specified error percentages  $\nu_p$ ,  $\nu_r$ , and  $\nu_f$  are chosen in a way to achieve a desired value for the ratio of the rms value of the spectral peaks of the supply current to the rms value of the continuous-spectrum floor (due to cycle-to-cycle variations) of the supply current. The ratio between these rms values is represented by  $\eta$  and it is given in [7] as

$$\eta^2 = \frac{1}{\frac{\nu_p^2}{3} + \frac{\nu_r^2}{3} + \frac{\nu_f^2 \cdot \nu_{rf}^2}{9}} \quad (2)$$

where  $\nu_{rf}$  is defined as the maximum percentage variation of the pulsewidth of the supply current, which is given by  $\nu_{rf} = (\nu_r \cdot t_{rp} + \nu_f \cdot t_{fp}) / (t_{rp} + t_{fp})$ . The parameters  $\nu_p$ ,  $\nu_r$ , and  $\nu_f$  are the maximum allowed percentage variations defined earlier. The parameter  $t_{r(f)p}$  is the rise(fall) time of the  $p$ th supply current profile. The compression efficiency decreases when tightening  $\nu_p$ ,  $\nu_r$ , and  $\nu_f$ .

Using the above procedure for each clock region, a set ( $IP(m)$ ) of supply-current profiles ( $i_p[n, m]$ ) is generated. From this set, we select the dominating supply-current profile ( $i_m[n]$ ) of the clock region  $m$

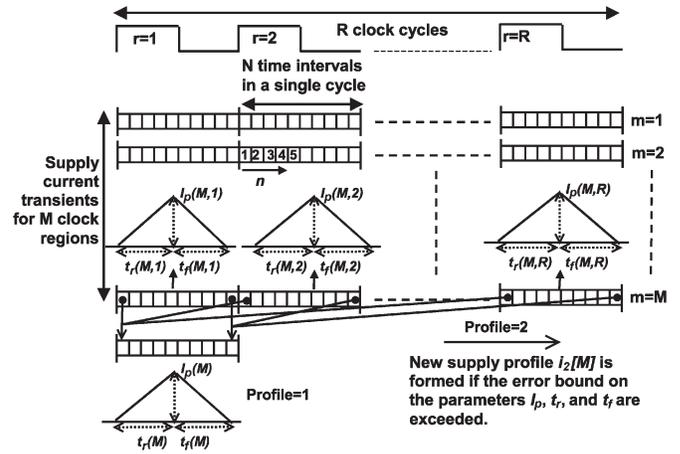


Fig. 5. Folding algorithm for forming the supply-current profiles in each clock region.

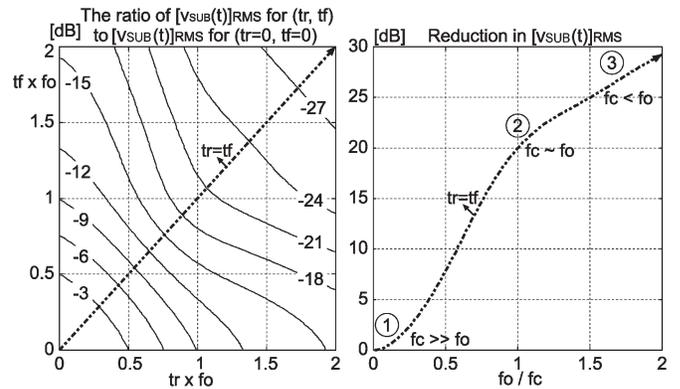


Fig. 6. Influence of the supply-current rise/fall time on the substrate-noise voltage simulated in SPICE for a 25 K-gates circuit in a 0.35- $\mu\text{m}$  3.3-V CMOS process with a supply-line inductance and resistance of 0.1 nH and 10 m $\Omega$ , respectively.

as the one representing the largest number of clock cycles. In some cases, the profiles other than the dominating profile can represent a significant portion of the cycles. This can possibly be caused due to different operating modes or due to the intrinsic periodicity of the circuit. In the case of different operating modes, the substrate-noise generation in each particular operating mode should be minimized using the profiles computed for this operating mode. In the case when the cycle-to-cycle variations have a known periodicity, we use a combination of the clock cycles to cover one period of the slowest clock.

#### B. Finding a Minimum Number of Clock Regions

Below a certain number of clock regions, the substrate-noise reduction by means of intentional clock skews may not be significant. In this section, we will find the required (minimum) number of clock regions in order to have a significant reduction in the substrate-noise generation.

Fig. 6 illustrates the influence of  $t_r$  and  $t_f$  on the rms value of the substrate-noise voltage ( $[v_{SUB}(t)]_{RMS}$ ) due to a triangular-shaped supply current simulated in simulation program with integrated circuits emphasis (SPICE) for a 25 K-gates circuit in a 0.35- $\mu\text{m}$  3.3-V CMOS process. For each of the power/ground rails, the circuit has a supply-line inductance ( $L_{p(g)}$ ) of 0.1 nH and a supply-line resistance ( $R_{p(g)}$ ) of 10 m $\Omega$ . The circuit has an extracted resonance frequency ( $f_o$ ) of 530 MHz and a damping factor of 0.19. The substrate-noise voltage has been normalized to its maximum value that occurs at  $t_r = t_f = 0$

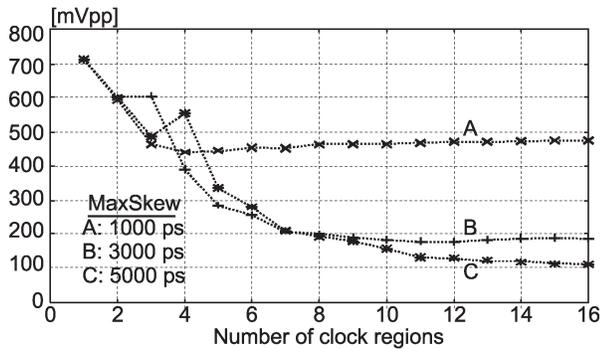


Fig. 7. PP value (simulated in SPICE) of the ground-bounce voltage generated from a circuit versus the granularity level (number of clock regions) with the maximum skew value as a parameter. The circuit is implemented in 0.18- $\mu\text{m}$  CMOS and contains 1000 flip-flops, where each flip-flop output is fed back to its data input through an inverter.

(where the supply current is a Dirac impulse). We define the corner frequency ( $f_c$ ) of the supply current as the bandwidth of the supply-current spectrum (see Fig. 4). Three regions are recognized (Fig. 6). 1) In the case when  $t_r$  and  $t_f$  are small,  $f_c$  is much larger than  $f_o$ . In this region, the power spectral density (PSD) of the supply current in the vicinity of  $f_o$  does not change significantly by modifying  $t_r$  and/or  $t_f$ . 2) By increasing  $t_r$  ( $t_f$ ),  $f_c$  approaches  $f_o$  and the iso-reduction lines become closely spaced. The rate of reduction increases significantly because the main lobe of the supply current is shifting well below  $f_o$ . 3) Decreasing  $f_c$  even more makes the current waveform a band-limited signal. In the limit, the main lobe becomes infinitely small, i.e., only dc level remains. Lowering  $f_c$  below  $f_o$  will therefore bring a significant reduction in the substrate-noise generation. This can be done by increasing the number of clock regions with skewed clocks. Without looking at the timing implications, the required minimum number of clock regions  $M$  is found by the ratio  $f_c/f_o$ . Choosing  $M$  or more clock regions spreads the supply current uniformly over a time period of  $\max(t_r, t_f) + t_r + t_f$ . This then sets the new corner frequency of the supply current at a frequency lower than the resonance frequency.  $M$  is given by

$$M = \frac{1}{f_o \cdot \min(t_r, t_f)}. \quad (3)$$

The actual rise/fall time ( $t_r, t_f$ ) is computed from a triangular approximation of the dominating supply-current profile extracted using the folding algorithm of Section II-A. After this step, the clock skews must be computed in order to have the desired corner frequency using  $M$  clock regions.

A fine-grained skewing (setting the number of clock regions as the total number of flip-flops) does not necessarily bring more ground noise reduction. For large circuits or for circuits having large supply inductances, the circuit resonance frequency decreases. For reducing the ground bounce, we need to increase the maximum skew value, therefore, it is necessary to increase the granularity level (number of clock regions) as well, but to a limit defined by (3). We will now illustrate the saturation of the granularity level for a circuit in 0.18- $\mu\text{m}$  CMOS and with 1000 flip-flops, where each flip-flop output is fed back to its data input through an inverter. With supply-line parasitics of  $L_{p(g)} = 1$  nH and  $R_{p(g)} = 0.2$   $\Omega$ , the circuit has a resonance at 395 MHz. To suppress the ground bounce, the corner frequency of the supply current should be set well below the resonance frequency. Fig. 7 shows the peak-to-peak (PP) value (simulated in SPICE) of the ground-bounce voltage generated from this circuit versus the number of clock regions with the maximum skew value as a parameter. The figure clearly shows that the amount of ground bounce saturates after

a certain number of clock regions. On the other hand, we need to trade the maximum skew value with the circuit timing (performance). Also, the reduction of the ground bounce slows down with increasing values of the maximum skew value when the corner frequency is well below the resonance frequency.

### C. Assignment of the Digital Cells to the Clock Regions

After computing the number of clock regions, it is now necessary to decide which digital cell belongs to which clock region. Such assignment should balance the supply-current contribution of each clock region for a significant reduction in the substrate-noise generation.

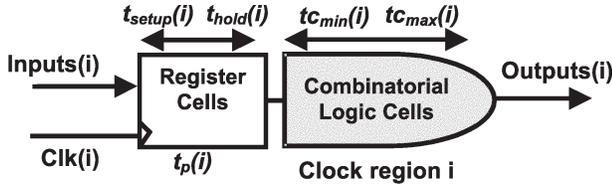
The assignment of  $N_{ff}$  flip-flops into  $M$  clock regions is done in two steps.

- 1) Assignment of the combinatorial cells to the flip-flop  $ff(i)$  of  $N_{ff}$  flip-flops. This flip-flop is called the driving cell [we also treat the input/output port(s) as the driving cell(s)]. The set of cells affected by the timing of the driving cell is called  $FF(i)$ .
- 2) Partitioning of the  $N_{ff}$  flip-flops into  $M$  clock regions, where typically  $N_{ff} \gg M$ . A cell is strictly an element of  $FF(i)$  when this cell belongs only to  $FF(i)$ . These cells are called the “single-timed cells.” Other cells, which are timed by more than one driving flip-flop, are called the “multiple-timed cells.” For multiple-timed cells, we assign a probability of switching due to each of the driving flip-flops. The probabilities are computed using the results from a very-high-speed integrated-circuit (VHSIC) hardware-description language (VHDL)-based switching activity simulator [1].

After the cell assignment, the sets  $FF(i)$  are grouped into appropriate clock regions such that their contributions into the total supply current are balanced as well as the contribution of multiple-timed cells to the supply current of different clock regions is minimized. It is vital to reduce the multiple-timed cells across different clock regions as much as possible to reduce possible glitches that cause an increase in power and signal-integrity problems. If multiple-timed cells across different clock regions still exist, the supply-current contribution of these cells is derived using the probabilities determined during the cell assignment. The assignment procedure also minimizes the hold-time constraints and data communication across different clock regions in order to relax timing constraints across different clock regions. This will also reduce the power overhead brought by extra buffers used for correcting the timing. Simulations show that the alternative approach of maximally spreading the multiple-timed cells over clock regions brings a large power penalty due to glitches.

The optimum assignment achieved by reducing the shared set of cells is not necessarily an optimum that targets at relaxing the placement/routing constraints during floorplanning. Merging these goals in the assignment problem is an interesting topic for further investigation. On the other hand, the proposed scheme of grouping the cells close to the driving flip-flops is also a major objective for a generic clock distribution network [10].

The assignment algorithm guarantees that, for each clock region, the sum of the ground-bounce contributions over all clock cycles is balanced. The equal balancing principle can be violated in some clock cycles due to temporal variations across cycles. In general, simultaneously switching flip-flops and clock buffers form a significant portion of the noise. In this case, the temporal variations across clock cycles do not severely violate the equal balancing principle. The temporal variability is a fundamental problem also for fine-grained skewing since the flip-flops that are adjacently skewed can sometimes be noisier than the other flip-flops in the circuit. On the other hand, the amount of



- $t_{setup}(i)$  : Setup time of the registers
- $t_{hold}(i)$  : Hold time of the registers
- $t_p(i)$  : Propagation delay of the registers
- $t_{c_{min}}(i)$  : Minimum delay of the combinatorial cells after the registers
- $t_{c_{max}}(i)$  : Maximum delay of the combinatorial cells after the registers

Fig. 8. Timing values for the  $i$ th clock region.

skew between these adjacent flip-flops does not necessarily guarantee a reduction of the ground bounce, as was described in Section II-B.

#### D. Clock-Skew Optimization

**Timing Constraints in Multiple Clock Regions:** The skews have to be constrained with the timing-constraint parameters defined in Fig. 8. Excessive negative skew may create a race condition, known as hold-time violation. This is prevented by keeping the clock skew  $\Delta t(i, j) > -tp_{min}(i, j)$ , where  $\Delta t(i, j)$  is the skew between the clock regions  $i$  and  $j$  ( $\Delta t(i, j) = \Delta t(i) - \Delta t(j)$ ) and  $tp_{min}(i, j)$  is the minimum propagation delay of the data path between two registers. On the other hand, excessive positive clock skew may limit the clock frequency, known as setup-time violation. This is prevented by  $\Delta t(i, j) < T_{clk} - tp_{max}(i, j)$ , where  $tp_{max}(i, j)$  is the maximum propagation delay of the data path between two clock regions. The communication between the clock regions  $i$  and  $j$ , therefore, has to satisfy the following:

- 1) no setup-time violation

$$\Delta t(i, j) < (T_{clk} - tp_{max}(i, j))$$

$$\text{where } tp_{max}(i, j) = t_p(i) + t_{setup}(j) + \delta - t_{c_{max}}(i) \quad (4)$$

- 2) no hold-time violation

$$\Delta t(i, j) > -tp_{min}(i, j)$$

$$\text{where } tp_{min}(i, j) = t_p(i) - t_{hold}(j) + t_{c_{min}}(i) - \delta \quad (5)$$

where  $\delta$  is the unintentional skew coming from the clock interconnect and the clock jitter. For each technology,  $\delta$  can be reduced by careful layouting and differential clocking techniques [10]. The noise-reduction factor can have a high sensitivity to this unintentional clock skew. To analyze this, a skew radius is constructed around the optimum skew bundle. We define the following indicator that shows the max/average values of the reduction factor due to unintentional clock skew within the skew radius  $\delta$ :

$$SF_{MAX(AVG)}(\delta) = \left( \sqrt{\frac{f_{cost}(\Delta t_{opt} \pm r)}{f_{cost}(0)}} \right)_{MAX(AVG)},$$

$$\forall r \in [0, \delta] \quad (6)$$

where  $\Delta t_{opt}$  and  $\delta$  are the optimum skew set and the unintentional skew on the optimized skew, respectively.  $f_{cost}(0)$  is the value of the cost function before the optimization. These timing constraints between the clock regions are computed using the analysis results of the PrimeTime tool [11].

The reliability of nonzero skews to the process variations and/or to (un)intentional skew has been introduced in [9]. The performance/race

reliability (PR/RR) of a circuit is the minimum of the distance from the actual skew to the upper/lower bound of the permissible skew range for all clock regions. These reliability figures are defined as

$$PR = \min (T_{clk} - tp_{max}(i, j) - \Delta t(i, j))$$

$$\forall i, j \in [1, M] \times [1, M]$$

$$RR = \min (\Delta t(i, j) + tp_{min}(i, j))$$

$$\forall i, j \in [1, M] \times [1, M]. \quad (7)$$

An ideal clock skew tolerant to process variations should be at the center of this permissible range. The performance and race reliabilities are used as constraints in the skew optimization, which are represented by  $PR_{target}$  and  $RR_{target}$ , respectively.

**Derivation of the Cost Function:** The optimization procedure is to find the best skew bundle  $\{\Delta t(1), \dots, \Delta t(m), \dots, \Delta t(M)\}$  that minimizes the cost function using the supply-current profiles shifted with the skews, where  $\Delta t(m)$  is defined as the skew value of clock region  $m$ . We arbitrarily set one of the skews to 0 such that the minimum skew after the optimization is then aligned to the clock edge. Therefore, we optimize  $M - 1$  skew variables.

The optimization tries to minimize the spectral energy of the substrate-noise voltage, which is derived from the product of the supply-current spectrum with its transfer function to the substrate-noise voltage. This transfer function can easily be derived using the chip-level substrate model introduced in [1]. The total supply current can be written as the sum of the supply-current spectra of the different clock regions. The frequency spectrum of a periodic waveform, where each period consists of the dominating supply-current profile, becomes the sample peaks of the envelope at the clock harmonics in the frequency domain. These peaks have been shown in [7] to be the dominant components of the total supply current. Therefore, the spectrum of the dominating supply-current profile at each clock region is evaluated at each clock harmonic  $p$  after a phase shift of  $e^{-j(2\pi/N)p \cdot R \cdot \Delta t(m)}$  to account for the delay  $\Delta t(m)$ . Consequently, the optimum skew bundle  $\{\Delta t(1), \Delta t(2), \dots, \Delta t(M)\}$  is found by solving the following optimization problem:

$$\min_{\Delta t} f_{cost}(\Delta t(1), \dots, \Delta t(M))$$

$$= \sum_{p=0}^{\frac{(N-1)}{2}} \left( |Z_{sb, mean}[p]|^2 \left| \sum_{m=1}^M I_m[p] \cdot e^{-j(\frac{2\pi}{N})p \cdot \Delta t(m)} \right|^2 \right) \quad (8)$$

where  $Z_{sb, mean}[p] = Z_{sb}[pR]$  is the supply-current transfer function to the substrate-noise voltage at the  $p$ th harmonic in the frequency domain, where  $R$  is the number of data points between two consecutive clock harmonics.  $I_m[p]$  is the discrete Fourier transform (DFT) value of the dominating supply-current profile of the  $m$ th clock region at the  $p$ th harmonic.

The constraint space (CS) of the optimization problem as a result of the timing constraints is given by

$$\text{subject to } \Delta t(1) = 0$$

$$-tp_{min}(i, j) + RR_{target} < \Delta t(i, j) < T_{cycle} - tp_{max}(i, j) + PR_{target}$$

$$\forall i, j \in [1, M] \times [1, M]. \quad (9)$$

All possible values of the cost function  $f_{cost}$  are computed exhaustively using (8) and using the constraint space given by (9). For a given bound  $\delta$  of the unintentional skew, the quality of the noise reduction is also checked. The optimum skew bundle  $\Delta t_{opt}$  is decided after

choosing a skew bundle giving the minimum worst penalty in the skew radius  $\delta$ . We first find the best ten skew bundles without considering the unintentional clock skew. Finally, from these bundles, we choose the best skew bundle  $\Delta t_{\text{opt}}$  by choosing the skew bundle that gives the minimum worst penalty in the skew radius  $\delta$ .

The optimum skews are implemented using a clock delay line (Fig. 1). For a better reduction in the substrate-noise voltage, we sometimes need to increase the search space for the optimum by allowing some hold-time violations. In this case, a timing correction module, which consists of delay buffers on the data path of different clock regions, can be used in order to correct the timing violations. On the other hand, the constraints for the setup-time case are not relaxed in the optimization. Therefore, the skew of each clock region should have a timing budget such that no setup violation exists. Here, the designer trades off the maximum operating frequency with the signal integrity. This tradeoff becomes much easier in advanced technologies where the intrinsic transistor switching speed in many cases exceeds the application requirements significantly.

The overall computational complexity of the optimization methodology as well as the supply-current folding methodology has a first-order dependence on the number of data points of the transient simulation. This comes as a result of using the compressed supply-current waveforms.

Top-level routing of each clock net to the corresponding clock region and data communication between different clock regions are the only incremental changes to be done in the layout if timing constraints are not met. On the other hand, this uncertainty as a result of routing is reflected as a random parameter ( $\delta$ ) in the timing constraints. As a last check, the fidelity of the timing results should be checked with a hierarchical timing analysis performed on the boundaries of each clock region.

The overhead in area and power for the implementation of clock skews is caused by extra circuits such as a clock delay line and (a limited amount of) data-path buffers for fixing the timing. The theoretical bound on the power overhead  $\Delta P$  (similarly, for the area overhead  $\Delta A$ ) is given by

$$\Delta P = \frac{\max(|\Delta t(i, j)|)}{T_{\text{ds}}} P_{\text{ds}} + \sum_{i=1}^M \sum_{\substack{j=1 \\ j \neq i}}^M \left( \frac{f(i, j) \cdot w(i, j) \cdot P_{\text{dh}}}{T_{\text{dh}}} + w(i, j) \cdot P_r(i, j) \right)$$

where  $f(i, j) = -\Delta t(i, j)$  for  $\Delta t(i, j) < 0$ , else 0. (10)

The first term in the sum is the power overhead in the clock delay line, where  $T_{\text{ds}}$  and  $P_{\text{ds}}$  are the delay and the power, respectively, of the unit delay buffer for implementing the clock delay line. The second term in the sum is the power penalty due to communication across different clock regions.  $T_{\text{dh}}$  and  $P_{\text{dh}}$  are the delay and the power, respectively, of the unit delay buffer for fixing the hold-time violations between different clock regions. The term  $w(i, j)$  is the number of nets leaving from clock region  $i$  to clock region  $j$  and  $P_r(i, j)$  is the power penalty due to long interconnects across different clock regions. Since the cells are clustered in a way to reduce the shared set of cells across different clock regions, the second term is also minimized.

### III. NUMERICAL RESULTS

We have tested the impact of intentional clock skews on ground-bounce generation (therefore, substrate noise) in a large telecom circuit consisting of 40 K gates and in ITC'99 benchmark circuits [12]. The circuits are implemented in a 0.18- $\mu\text{m}$  1.8-V CMOS process. Since the

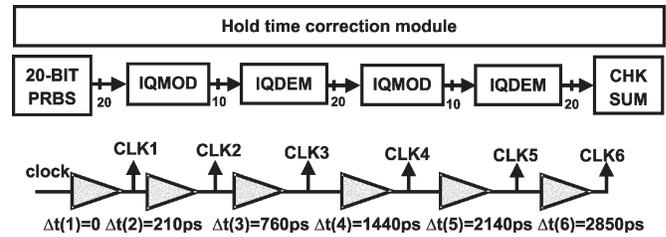


Fig. 9. Block diagram of the 40 K-gates telecom circuit together with the parameters of the clock generator circuit.

circuits are too complex to simulate with SPICE, we use the substrate waveform analysis (SWAN) tool [1] to simulate the supply current and the ground-bounce voltage. The accuracy of the SWAN tool has been verified with measurements such as from a 220 K-gates mixed-signal telecom circuit [1].

The 40 K-gates telecom circuit is composed of a 20-bit maximum-length-sequence pseudorandom bit sequence (PRBS) circuit driving two cascaded sets of IQ modulator and demodulator chains (Fig. 9). The supply-current transfer function to the ground node (therefore, substrate node) has a resonance frequency of 475 MHz. The design has a clock period of 20 ns and supply-line parasitics of  $L_{\text{p}(g)} = 200$  pH and  $R_{\text{p}(g)} = 0.2 \Omega$ . The design has 913 flip-flops, 8265 combinatorial gates, and 19 407 nets. Since the spectrum bandwidth (corner frequency) of the supply current is 1.8 GHz, choosing six clock regions will guarantee that the resulting corner frequency is lower than the resonance frequency, while smoothing the supply current in the time domain (3). The assignment of the combinatorial cells and the flip-flops to the clock regions has been performed in around 19 min of CPU time. In addition to balancing the supply current across clock regions, the assignment procedure also minimizes the hold-time constraints and data communication across different clock regions in order to relax timing constraints across different clock regions.

The dominating supply-current profile set has been computed using the actual supply-current data from a total transient simulation of 248 clock cycles within 16 min of CPU time using SWAN. For this computation we use an error bound of 20% on the peak value, the rise time, and the fall time ( $\nu_p = \nu_r = \nu_f = 0.2$ ). With these error bounds, the ratio between the rms values of the discrete spectral peaks and the continuous spectrum of the substrate-noise voltage becomes 15.71 dB (2). Using these parameters, a total of 13 supply-current profiles is generated for each of the six clock regions. The dominating supply-current profile set represents 31.7% of all clock cycles.

In accordance with (8), the skews have been selected to minimize the frequency content of the clock harmonics of the total supply current constructed by the dominating supply-current profile set in the frequency band from dc to 4 GHz within 1 min of CPU time. With these new skew values, we then simulate the ground-bounce voltage for 248 clock cycles within 20 min of CPU time. The total supply current before and after the clock skews is given in Fig. 10. The transient simulation of 248 clock cycles before/after the clock skews generates 2 096 686/2 231 179 input switchings. This shows that the clock-region assignment does not change the switching behavior of the circuit significantly, which is a change of 6.4%. In order to correct the hold-time violations between the clock regions, we have added a timing-correction module that consists of delay buffers on the data path between different clock regions (Fig. 9). This procedure resulted in a total of 134.3-ns hold-time correction by means of delay buffers over 219 nets. These delay buffers are designed with long-channel devices for minimizing the power/area overhead for a large delay. No setup-time violations occur with the new skew values since the maximum operating frequency of the design was 130 MHz.

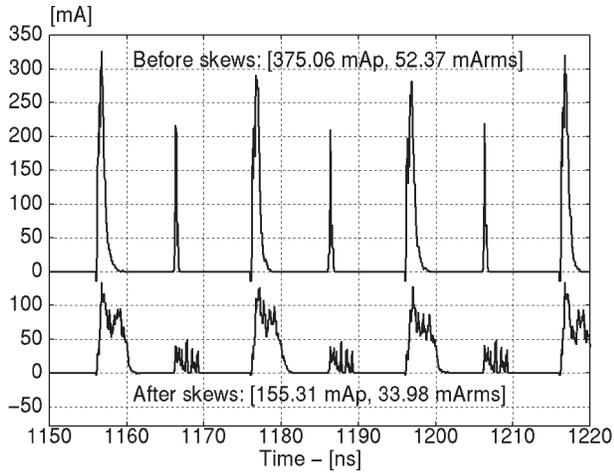


Fig. 10. Supply-current transients before/after intentional clock skews for the 40 K-gates telecom circuit with  $F_{\text{clk}} = 50$  MHz. (mA p: the peak value, mA rms: the cycle rms value of the supply-current transients simulated over 248 clock cycles).

The simulated peak/rms/dc values of the supply current are 351.1 mA, 52.37 mA rms, and 15.232 mA dc, respectively, before the skews, and 155.29 mA, 33.98 mA rms, and 16.517 mA dc, respectively, after the skews. After the skews, the amount of energy increases by 8.1% as a result of the increase in glitches. The ground-bounce reduction is around 9.5 dB at the circuit resonance in the frequency domain [Fig. 11(a) and (b)] and factors of  $2.85\times$  and  $2.62\times$  in the PP and rms values, respectively, in the time domain [Fig. 11(c)]. In the ground-bounce spectrum, local minima are visible around 250 MHz and 1.1 GHz after decreasing the spectrum bandwidth of the supply current by means of intentional clock skews. This is due to the reduction of the spectral power of the supply current close to the resonance frequency of the transfer function  $Z_{\text{sb}}(j\omega)$ , in line with (8). Around 250 MHz, the reduction of the spectral peak is around 14.5 dB. The skew sensitivity figures (explained in Section II-D) show that under worst case unintentional clock skew of 200 ps, the ground-bounce reduction factors are only 11% lower. This demonstrates the robustness of the approach.

The intentional clock skew is also tested on the ground-bounce (therefore, substrate noise) reduction figures of ITC'99 benchmark circuits [12] (see details in Table I). We use the SWAN tool to simulate the supply current and the ground-bounce voltage using random test vectors as input. Each of the circuits has supply-line parasitics of  $L_{\text{p}(g)} = 0.4$  nH and  $R_{\text{p}(g)} = 0.1$   $\Omega$ . For the construction of the dominating supply-current profile set, we use the following error bounds: ( $\nu_{\text{p}} = \nu_{\text{r}} = \nu_{\text{f}} = 0.2$ ). Table I shows the parameters of the circuits. Table II shows both the values of the supply current (sixth column) and the ground-bounce generation (seventh column) before/after the skews. The table also lists the number of supply-current profiles, the representation percentage of the dominating supply-current profile, the number of switchings, and the total hold-time penalty before/after the skews. It is concluded that the reduction is around 45%/56% in average for the PP/rms values of the ground-bounce voltage with a 4.4% increase in average for power. The reduction efficiency decreases in those circuits when the number of multiple-timed instances across different clock regions becomes very large. For these reasons, the reduction efficiency decreases in the circuits b20 and b21. For b22, we even have an increase in the ground bounce despite the fact that the peak current has been reduced considerably after the skews.

The reduction becomes significant when the cell assignment balances the supply-current contribution of each clock region for all clock

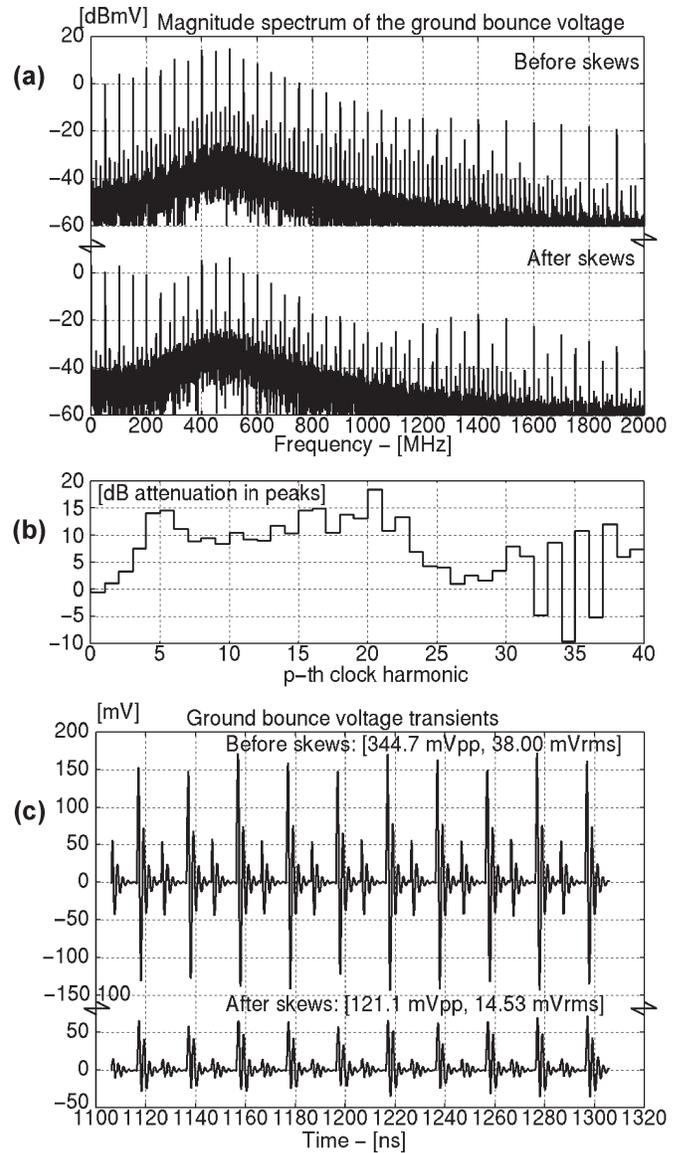


Fig. 11. Ground-bounce (a) spectrum, (b) reduction in decibels in the spectral peaks, and (c) transients before/after intentional clock skews for the 40 K-gates telecom circuit with  $F_{\text{clk}} = 50$  MHz.

TABLE I  
FEATURES OF THE ITC'99 BENCHMARK CIRCUITS  
( $C_{\text{CIR}}$  IS THE CIRCUIT CAPACITANCE)

Circuit Name	#Cells/ Flip-flops	Area in gates	$C_{\text{cir}}$ [pF]	$f_0$ [MHz]	Clk Cycles/ Freqn. [MHz]
B05	546/34	954	8.14	1972	248/50
B11	333/31	703	5.86	2324	248/50
B12	882/121	1976	15.67	1421	248/50
B13	239/53	644	5.17	2475	248/50
B14	3361/245	7717	67.99	682	248/50
B15	5873/449	11321	92.19	586	248/50
B17	16629/1415	33926	274.31	340	248/50
B18	44903/3276	90752	762.51	204	248/50
B20	6840/490	15833	138.24	479	248/50
B21	6632/490	15396	136.29	482	248/50
B22	10203/703	23362	205.92	392	248/50

cycles, and when there are no complications such as the increase in the amount of glitches due to different clock regions. Here, we would like to illustrate the case when timing and glitches are not

TABLE II  
GROUND-BOUNCE REDUCTION AND POWER/HOLD-TIME PENALTY FIGURES OF THE INTENTIONAL CLOCK SKEWS ON THE ITC'99 BENCHMARK CIRCUITS. *M*: NUMBER OF CLOCK REGIONS. *PM*: NUMBER OF PROFILES. *R%*: REPRESENTATION PERCENTAGE OF THE DOMINATING SUPPLY-CURRENT PROFILE SET. *IS*: NUMBER OF INPUT SWITCHINGS. *HT*: HOLD-TIME PENALTY. *HT-FF*: NUMBER OF FLIP-FLOPS CAUSING HOLD-TIME VIOLATIONS.  $I_S(T)$ : SUPPLY CURRENT.  $V_{SS}(T)$ : GROUND-BOUNCE VOLTAGE. *PP*: PEAK-TO-PEAK VALUE

Circuit Name	M	PM-R%	IS		HT [ns]	HT-FF	$[i_s(t)]_{P/RMS/DC}$			RunTime [min]
			Init-Final				[mA] Init-Final	$[V_{SS}(t)]_{PP/RMS}$	Init-Final	
B05	2	7	32934	0	12.67 / 0.961 / 0.139	138.2 / 12.62	1.37			
		73.57%	32907	0	9.340 / 0.748 / 0.139	91.78 / 5.812				
B11	2	8	33156	0	11.40 / 0.881 / 0.134	127.8 / 12.92	1.34			
		38.21%	33483	0	7.800 / 0.700 / 0.134	70.86 / 5.497				
B12	3	2	68354	0	43.66 / 3.268 / 0.378	233.3 / 30.48	2.03			
		50.0%	68354	0	22.33 / 2.013 / 0.378	99.43 / 12.95				
B13	2	5	45238	0	19.18 / 1.457 / 0.204	246.7 / 22.26	1.42			
		45.93%	45246	0	11.94 / 1.074 / 0.204	97.39 / 5.217				
B14	4	10	852400	0.9	92.69 / 11.08 / 3.548	328.5 / 30.83	28.78			
		25.20%	863210	6	83.21 / 9.879 / 3.497	256.2 / 26.41				
B15	5	3	478993	2.2	161.6 / 12.12 / 1.837	187.7 / 30.40	11.20			
		95.12%	598029	3	92.61 / 8.066 / 2.058	128.8 / 16.24				
B17	10	2	849548	7.3	507.8 / 37.53 / 4.435	163.8 / 27.13	35.69			
		99.59%	925437	11	127.0 / 14.60 / 4.540	60.50 / 7.897				
B18	12	2	2314012	4.6	1177 / 86.78 / 10.90	175.2 / 38.81	139.19			
		99.59%	2505265	3	202.4 / 28.80 / 11.20	56.63 / 10.33				
B20	8	11	1126175	3.6	177.1 / 16.00 / 4.201	173.1 / 21.60	24.49			
		32.92%	1286988	21	119.3 / 12.06 / 4.605	182.1 / 17.00				
B21	8	10	1070339	3.1	178.0 / 16.02 / 4.196	208.6 / 22.90	23.73			
		22.35%	1182747	22	118.8 / 11.54 / 4.708	166.9 / 15.32				
B22	10	8	1814431	19.9	254.6 / 24.68 / 7.131	203.5 / 23.04	40.77			
		29.26%	2019526	31	140.5 / 19.04 / 8.145	210.7 / 24.46				

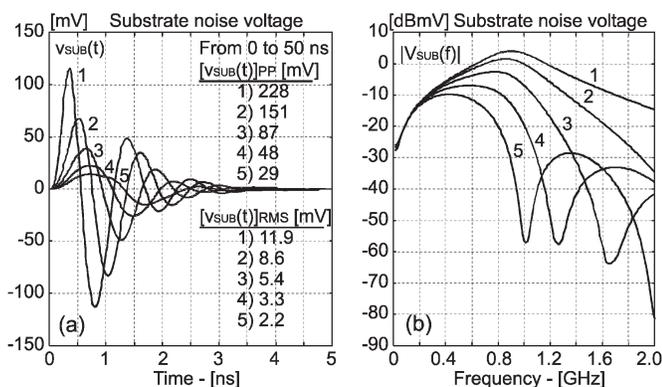


Fig. 12. Simulated (SWAN) substrate-noise (a) transients and (b) the corresponding spectra after different shapings of the supply current in a circuit having a resonance frequency of 950 MHz.

taken into account. Fig. 12 shows the transients and the corresponding spectra of a hypothetical circuit, having a resonance at 950 MHz, when the amplitude and slope of the supply current are changed by means of intentional clock skews. The supply current has been shaped from (50 mA, 200 ps) to (10 mA, 1000 ps) with equal rise and fall times. As a result of flattening the supply current, we achieve a significant reduction in the generated substrate-noise voltage:  $5.4\times$  in its PP and  $7.9\times$  in its rms value. Adding decoupling capacitors is useful for substrate-noise reduction. However, the value of the corner frequency should be guaranteed to be well below the new resonance frequency (after adding decoupling capacitors), which is now at a lower frequency than without addition of decoupling capacitors.

The use of intentional clock skews has been tested in an experimental mixed-signal chip fabricated in a  $0.35\text{-}\mu\text{m}$  3.3-V CMOS process on an epitaxial (EPI)-type substrate [8]. The chip includes a reference design (REF) and its two low-noise implementations (LN1 and LN2)

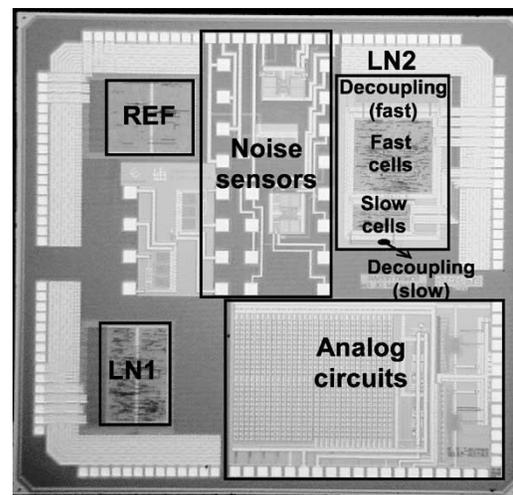


Fig. 13. Microphotograph of the test chip for the evaluation of the low-noise digital design techniques. The chip consists of a reference design (REF) and two low-noise designs (LN1 and LN2) with the same functionality. It also contains two substrate-noise sensors and also some analog circuits.

(see the chip photo in Fig. 13). The measurements show that a factor-of-2 reduction in substrate-noise generation is achieved with only a 3% area and a 4% power increase in the first low-noise design (LN1 in Fig. 13) with optimized clock skews to shape the supply current [8]. The overhead in area and power is caused by small-size circuits such as the clock delay line and some data-path buffers for fixing the timing. More than a factor-of-two reduction in substrate noise has been obtained for the second low-noise design (LN2 in Fig. 13), which has a separate substrate bias, dual supply, and on-chip decoupling, however, with a 70% increase in area but with 5% less power at the same operating frequency.

## IV. CONCLUSION

In this paper, we have introduced an efficient clock-skew-optimization methodology for reducing the ground bounce (therefore, substrate noise) in large digital circuits. The methodology splits the digital system in different clock regions and optimizes the clock skews between them to reduce the substrate-noise generation. The required number of clock regions is computed based on the elimination of the major resonance frequency determined by the on-chip circuit capacitance and the supply-line parasitics. The run-time of the optimization is improved by using supply-current profiles that can be used as periodic pulses for the representation of the total supply current. Additional constraints such as performance and race reliabilities have been introduced into the optimization in order to have a resulting clock tree tolerant to process variations.

Experimental results show around a factor-of-2 reduction in the rms value of the generated ground-bounce voltage. The effectiveness of the methodology has also been verified with measurements on a fabricated mixed-signal chip. The supply-current shaping by the use of intentional clock skews has been shown to be very effective for reducing the substrate-noise generation if timing constraints allow shaping.

## REFERENCES

- [1] M. Badaroglu, S. Donnay, H. De Man, Y. Zinzus, G. Gielen, T. Fonden, and S. Signell, "Modeling and experimental verification of substrate noise generation in a 220-KGates WLAN system-on-chip with multiple supplies," *IEEE J. Solid-State Circuits*, vol. 38, no. 7, pp. 1250–1260, Jul. 2003.
- [2] P. Vuillod, L. Benini, A. Bogliolo, and G. De Micheli, "Clock-skew optimization for peak current reduction," in *Proc. IEEE Int. Symp. Low Power Electronics and Design*, Monterey, CA, Aug. 1996, pp. 265–270.
- [3] A. Vittal, H. Lia, F. Brewer, and M. Marek-Sadowska, "Clock skew optimization for ground bounce control," in *Proc. ACM/IEEE Int. Conf. Computer-Aided Design*, San Jose, CA, Nov. 1996, pp. 395–399.
- [4] W.-C. D. Lam, C.-K. Koh, and C.-W. A. Tsao, "Power supply noise suppression via clock skew scheduling," in *Proc. Int. Symp. Quality Electronic Design*, San Jose, CA, Mar. 2002, pp. 355–360.
- [5] R. Chaturvedi and J. Hu, "Buffered clock tree for high quality IC design," in *Proc. Int. Symp. Quality Electronic Design*, San Jose, CA, Mar. 2004, pp. 381–386.
- [6] M. Badaroglu, K. Tiri, S. Donnay, P. Wambacq, I. Verbauwhede, G. Gielen, and H. De Man, "Clock tree optimization in synchronous CMOS circuits for substrate noise reduction using folding of supply currents," in *Proc. ACM/IEEE Design Automation Conf.*, New Orleans, LA, Jun. 2002, pp. 399–404.
- [7] M. Badaroglu, P. Wambacq, G. Van der Plas, S. Donnay, G. Gielen, and H. De Man, "Digital ground bounce reduction by supply current shaping and clock frequency modulation," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 24, no. 1, pp. 65–76, Jan. 2005.
- [8] M. Badaroglu, M. van Heijningen, V. Gravot, J. Compieg, S. Donnay, G. Gielen, and H. De Man, "Methodology and experimental verification for substrate noise reduction in CMOS mixed-signal ICs with synchronous digital circuits," *IEEE J. Solid-State Circuits*, vol. 37, no. 11, pp. 1383–1395, Nov. 2002.
- [9] J. L. Neves and E. G. Friedman, "Design methodology for synthesizing clock distribution networks exploiting nonzero localized clock skew," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 4, no. 2, pp. 286–291, Jun. 1996.
- [10] S. Rusu, J. Stinson, S. Tam, J. Leung, H. Muljono, and B. Cherkauer, "A 1.5-GHz 130-nm Itanium 2 processor with 6-MB on-die L3 cache," *IEEE J. Solid-State Circuits*, vol. 38, no. 11, pp. 1887–1895, Nov. 2003.
- [11] *PrimeTime Tool From Synopsys, Inc. for Timing Closure of ICs*. [Online]. Available: <http://www.synopsys.com/products/products.html>
- [12] *ITC'99 Benchmark Circuits Developed in the CAD Group of Politecnico di Torino*. [Online]. Available: <http://www.cad.polito.it/tools/itc99.html>

## A High-Performance Data Path for Synthesizing DSP Kernels

Michalis D. Galanis, George Theodoridis, Spyros Tragoudas, and Costas E. Goutis

**Abstract**—A high-performance data path to implement digital signal processing (DSP) kernels is introduced in this paper. The data path is realized by a flexible computational component (FCC), which is a pure combinational circuit and it can implement any  $2 \times 2$  template (cluster) of primitive resources. Thus, the data path's performance benefits from the intracomponent chaining of operations. Due to the flexible structure of the FCC, the data path is implemented by a small number of such components. This allows for direct connections among FCCs and for exploiting intercomponent chaining, which further improves performance. Due to the universality and flexibility of the FCC, simple and efficient algorithms perform scheduling and binding of the data flow graph (DFG). DSP benchmarks synthesized with the FCC data path method show significant performance improvements when compared with template-based data path designs. Detailed results on execution time, FCC utilization, and area are presented.

**Index Terms**—Binding, chaining, high-performance data path, scheduling, template units.

## I. INTRODUCTION

Digital signal processing (DSP) and multimedia applications usually spend most of their time executing a small number of code segments with well-defined characteristics, called kernels. To accelerate the execution of such kernels, various high-performance data paths have been proposed [1]–[11]. Research activities in high-level synthesis (HLS) [2]–[4] and application-specific instruction processors (ASIPs) [5]–[11] have proven that the use of complex resources instead of primitive ones [like a single arithmetic logic unit (ALU)] improves the performance of the data path. In these works, complex operations (instructions in the ASIP domain) are used at the behavioral level instead of groups of primitive resources. At the architectural level, the complex operations are implemented by optimal custom-designed hardware units, called templates or clusters. A template may be a specialized hardware unit or a group of chained units. Special hardware is usually used for operations that commonly appear (e.g., multiply–add). Chaining is the removal of the intermediate registers between the primitive units, improving the total delay of the combined units. The templates can be either obtained by an existing library [2]–[5] or can be extracted by the kernel's data flow graph (DFG) [6]–[11]. Although performance is improved with the usage of templates, the area of the produced data path is larger than the area of the corresponding data path implemented by primitive resources [2]–[11]. However, the area–time product, which indicates the efficiency of the synthesized data path [2], is decreased over a data path implemented by primitive resources.

Manuscript received April 7, 2004; revised November 8, 2004 and May 23, 2005. The work of M. D. Galanis was supported by the A. S. Onassis Public Benefit Foundation. This paper was recommended by Associate Editor A. Raghunathan.

M. D. Galanis and C. E. Goutis are with the Very Large Scale Integration (VLSI) Design Laboratory, Electrical and Computer Engineering Department, University of Patras, Rio 26500, Greece (e-mail: mgalanis@ee.upatras.gr; goutis@ee.upatras.gr).

G. Theodoridis is with the Physics Department, Aristotle University, Thessalonica 54124, Greece (e-mail: theodor@physics.auth.gr).

S. Tragoudas is with the Electrical and Computer Engineering Department, Southern Illinois University, Carbondale, IL 62901 USA (e-mail: spyros@engr.siu.edu).

Digital Object Identifier 10.1109/TCAD.2005.855965