Synthesis of Active-Mode Power-Gating Circuits

Jun Seomun, Insup Shin, and Youngsoo Shin, *Senior Member, IEEE*

***Abstract*—Active leakage is transient, which can be suppressed by design techniques such as dual-V_t. Active-mode power-gating (AMPG) [2] can further reduce active leakage by power-gating groups of gates that perform computations with results that are not loaded due to clock-gating. AMPG involves several challenges; the grouping of gates must take circuit timing into account, and current switches need to be sized to preserve power network integrity as well as circuit timing. We propose solutions to these problems in the content of the entire process of synthesizing AMPG circuits. The physical design of AMPG circuits is also difficult due to the large number of virtual ground rails that must be mutually isolated. We address these issues by integrating placement with power network synthesis. Experiments on several test circuits implemented in 45-nm technology demonstrate the effectiveness of AMPG in the circuits that we synthesized, in terms of power consumption, area, wirelength, and timing.**

***Index Terms*—Active leakage, active-mode power-gating, clock-gating, power-gating.**

I. Introduction

LEAKAGE current has been a focus of research for many years due to its growing contribution to the power consumption of circuits. Standby leakage occurs all the time when a circuit is in standby mode; it has received a lot of attention, and circuit design techniques such as power-gating and reverse body-biasing have been proposed to reduce it. Active leakage is transient and occurs when a circuit is actively computing. Its magnitude is much larger than that of standby leakage, roughly double in 45-nm technology (see Section II), and this is a full 30% of the total active-mode power consumption.

Despite its undoubted significance, active leakage has received less attention than standby leakage. The transient nature of active leakage makes estimation and minimization more difficult. Only design techniques such as dual-V_t and dual-gate-length are able to reduce it; these two techniques have respectively been reported to achieve 42% [3] and 30% [4] reductions in active leakage.

It has recently been suggested [2], [5] that active-mode power-gating (AMPG) can achieve further reductions in active leakage when combined with dual-V_t or standard power-gating. The basis of AMPG is the observation that, if some flip-flops are clock-gated (CG), a group of gates that generate their input data can be power-gated. An example of a circuit in which clock-gating is achieved by the inclusion of clock-gating logic and clock-gating cells is shown in Fig. 1(a). An AMPG version of the same circuit, which now contains n-channel metal–oxide–semiconductor field-effect transistor (nMOS) switches called footers, is shown in Fig. 1(b). Each footer is turned on when the clock is not gated (CG = 0) and turned off when the clock is gated (CG = 1). If a footer is turned off, the group of gates that are attached to it is cut off from the ground rail, reducing their active leakage.

A. Challenges in Designing AMPG Circuits

There are two main challenges in designing AMPG circuits. One is identifying groups of gates that can be power-gated together, and then sizing their footers: we will call this AMPG synthesis. The other is physical design, in particular placement and power network synthesis.

Intuitively, gates can be grouped together if they are only involved in the processing of data loaded by a set of flip-flops that are CG together [2], [6]. However, the grouping must allow a group that is woken up (CG = 0) to be ready to start a new computation before the next rising edge of the clock signal. Thus the time to discharge the virtual ground rail V_{ssv} shown in Fig. 1(b), as well as the gates in the group, should be considered in the grouping process. Once the gates have been grouped, it becomes necessary to determine the size of the footer attached to each group. This sizing process must take account of the discharge current flowing during wakeup and the circuit delay in active mode. Specifically, each footer should be small enough so that the instantaneous discharge current through it can be sustained by the ground rail; but also large enough so that the voltage drop across it can be kept acceptably small when the circuit is in active mode.

Each group of gates is associated with its own virtual ground rail V_{ssv}. If there are N groups, then placement of AMPG circuits has to care with N of these V_{ssv} rails, as well as the usual V_{ss} rail. The simplest approach is row-based placement, in which each row is exclusively occupied by the gates from the same group. However, a 30% increase in the length of wires has been reported [1], which makes the practicality of this approach questionable. We attack this problem by integrating placement and power network synthesis. We perform initial placement without considering the V_{ssv} rails. Then we create many local V_{ssv} networks while preserving the initial placement as far as possible. These V_{ssv} networks are then subject to further constrained placement, during

Manuscript received May 15, 2011; revised August 17, 2011; accepted September 28, 2011. Date of current version February 17, 2012. This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD, Basic Research Promotion Fund), under Grant KRF-2008-331-D00406. A preliminary version of this paper [1] was presented at the 47th Design Automation Conference, Anaheim, CA, June 13–18, 2010. This paper was recommended by Associate Editor D. Sylvester.

J. Seomun is with Samsung Electronics, Yongin, Gyeonggi-Do 449-711, Korea (e-mail: j.seomun@samsung.com).

I. Shin and Y. Shin are with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Korea (e-mail: isshin@dtlab.kaist.ac.kr; youngsoo@ee.kaist.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCAD.2011.2171963

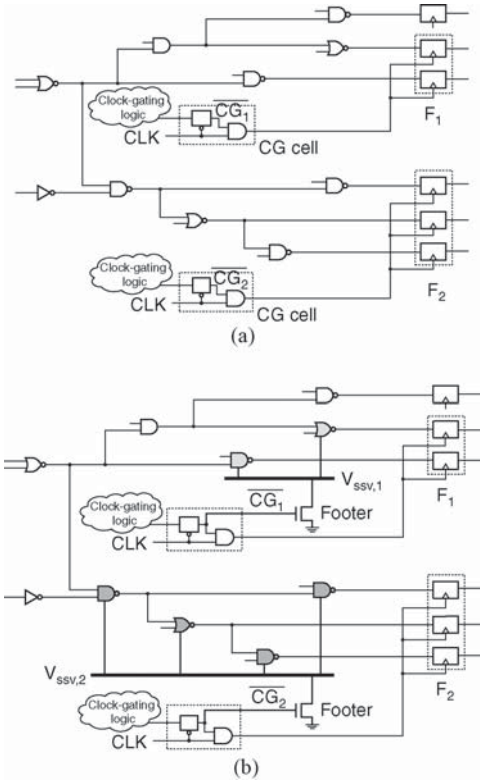

Fig. 1. (a) Clock-gating circuit. (b) AMPG circuit.

which the cells are forced to stay within their corresponding networks.

B. Main Contributions

The main contributions of this paper are:

1) an analysis of active leakage at both gate and circuit level in 45-nm technology (Section II), allowing a clearer understanding of the circumstances in which active leakage is significant;
2) definition of the AMPG synthesis problem and its solution (Section III), together with estimation of the maximum and average discharge current (ADC);
3) placement of AMPG circuits, including power network synthesis (Section IV);
4) experiments to assess AMPG circuits in terms of active leakage, active-mode power consumption, area, wirelength, and circuit timing (Section V), as well as to verify correct functionality.

II. Active Leakage

A typical operation of digital circuit has an active and a standby mode. The current drown while in active mode consists of the switching current, caused by charging and discharging of load capacitances, short-circuit current, and the active-leakage current. Standby mode does not involve any transistor switching (assuming that no switching occurs in clocks), and thus consists of the standby-leakage current alone.

Standby leakage is a steady-state current and can readily be measured by circuit-level simulation tools such as fast SPICE [7], which is indeed what we use. However, both switching current and active leakage are transient, and are also difficult to separate from each other. Subthreshold leakage, which is the main component of leakage current, is strongly dependent on temperature, but switching current is not. Thus, active-mode current converges to some asymptotic value as temperature decreases, as illustrated in Fig. 2. At a low temperature, such as $-25\,^{\circ}\mathrm{C}$, it is reasonable to approximate the active-mode by the switching current alone; the difference between this current and the active-mode current at some maximum operating temperature, such as 125°C, can be regarded as the maximum active leakage.

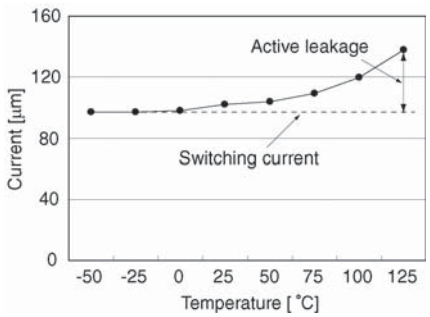

Fig. 2. Active-mode current of the ps2 circuit at different temperatures.

TABLE I
Standby and Active Leakage of a Two-Input NAND Gate in 45-nm Technology

Input (AB)	Standby Leakage (nA)	Active Leakage (nA)		
		1 ns	5 ns	10 ns
00	1.0	11.2	3.0	1.9
01	16.6	16.2	16.6	16.6
10	10.3	17.7	10.4	10.3
11	26.4	26.4	26.4	26.4

A. Gate-Level Analysis

The standby leakage of the two-input NAND gate shown in Fig. 3(a) for the different inputs is given in the second column of Table I. It is well known that this leakage is lowest when the input is 00, as Table I confirms. This is because of a positive voltage v_m which builds up between M_1 and M_2 and turns M_1 off strongly, due to a negative gate-to-source voltage; this voltage also raises the effective threshold voltage of M_1. This phenomenon as a whole is called the stacking effect [8], because the leakage shrinks as stacked metal–oxide–semiconductor transistors are turned off.

We now turn our attention to active leakage. When the input is maintained at 01, the internal node capacitance c_m is fully discharged. If the input is changed to 00 after 1 ns, as depicted in Fig. 3(b),[1] the small leakage current through M_1 starts to charge c_m. As v_m rises, the leakage through M_1 falls further due to the stacking effect. But this transition takes a long time, as shown in Fig. 3(b). The effect on leakage of a change of

[1]Input B makes a falling transition, thus v_m spontaneously drops below zero due to coupling between gate and drain terminals of M_2. As a result, V_{ds} of M_1 becomes larger than V_{dd}, causing a spike of instantaneous large current.

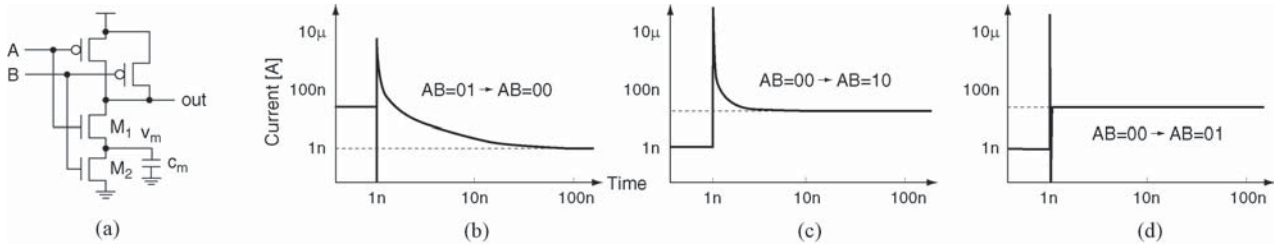

Fig. 3. (a) Two-input NAND gate. Active leakage for input transitions from (b) 01 to 00, (c) 00 to 10, and (d) 00 to 01.

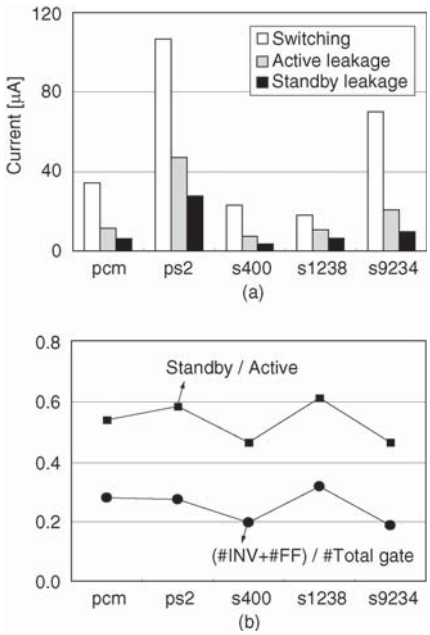

Fig. 4. (a) Switching current (white bars), active leakage (gray bars), and standby leakage (black bars) of some test circuits. (b) Ratio of standby to active leakage; the lower curve shows the proportion of unstacked gates.

input from 00 to 10 is shown in Fig. 3(c). The large turn-on current through M_1 initially charges c_m; however, as v_m rises, M_1 turns off, but then its leakage current takes over and continues to charge c_m, even though the leakage is gradually falling. If M_2 is turned on, for instance by the change of input from 00 to 01 shown in Fig. 3(d), the corresponding leakage transition is virtually spontaneous since c_m is quickly discharged.

The average active leakage over different periods after the change of input value is given in the last three columns of Table I. Each figure is also averaged over all the transitions that lead to the inputs shown in the first column; thus the first row covers transitions from 01 to 00, 10 to 00, and 11 to 00.

The standby and active leakage are about the same when a 1 is applied to input B (01 and 11 of Table I), which turn on M_2. The leakages for 10 and, especially, 00 are significantly different, in particular for the period immediately after the transition, implying a higher operating frequency.

B. Circuit-Level Analysis

The three components of the current drown by some test circuits, which are switching, active leakage, and standby leakage, are compared in Fig. 4(a). The clock period was arbitrarily set to 5 ns, 100 random vectors were applied, and the average currents are reported. The average standby leakage is 54% of the average active leakage. The variation in the leakage ratio between circuits, which is shown in Fig. 4(b), can be explained by the extent of the stacking effect in each circuit. When there are a lot of gates that exhibit the stacking effect in standby mode, we expect the difference between active and standby leakage to increase. To test this hypothesis, we counted the number of inverters and flip-flops, which are representative of the gates that are not stacked in each circuit. The proportion of these unstacked gates is also shown in Fig. 4(b). This graph supports our hypothesis. The contribution of active leakage to the total active-mode current, which is the sum of the switching current and the active leakage, is 28% on average.

Because of the way in which we extract the active leakage from the total active-mode current, which is as illustrated in Fig. 2, the proportion of active leakage decreases with temperature, as shown in Fig. 5(a). The ratio of standby to active leakage also declines, as Fig. 5(a) shows, suggesting that the importance of active leakage grows as the temperature drops. When this happens, the transient change in active leakage due to a transition (see Fig. 3) takes longer because of its reduced magnitude, which means that c_m is charged more slowly; this increases the difference between active and standby leakage.

As the clock frequency increases and the clock period decreases, the magnitude of the active leakage will increase while the standby leakage remains the same. This is evident from the decreasing ratio between the standby and active leakage shown in Fig. 5(b). The total switching current is independent of the clock period, as long as that period is sufficient to accommodate all the switching required. While the average switching current and the active leakage both increase as the clock period decreases, the average switching current increases more rapidly. Thus the active leakage comes to represent a lower proportion of the total active-mode current, as we see in Fig. 5(b).

III. Synthesis of AMPG Circuits

A. Problem Formulation

The synthesis of AMPG circuits can be formally defined as follows.

Problem 1: We are given a sequential circuit with the clock-gating signals $CG_1, CG_2, \ldots$ (see Fig. 1). Signal CG_i unblocks

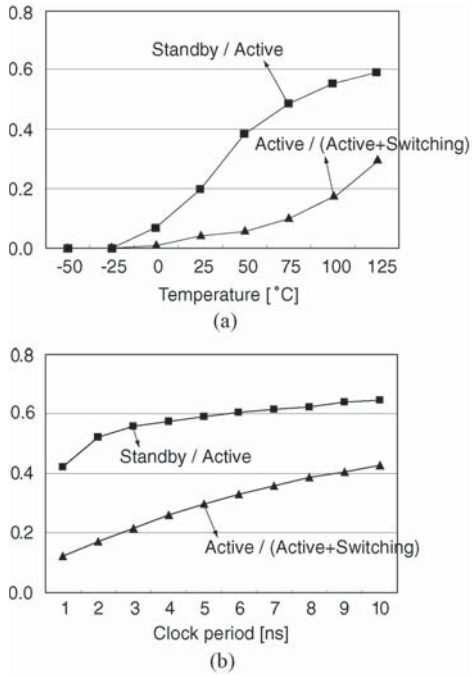

Fig. 5. Ratio of standby to active leakage, and the proportion of active leakage in circuit ps2. (a) With varying temperature. (b) With varying clock period.

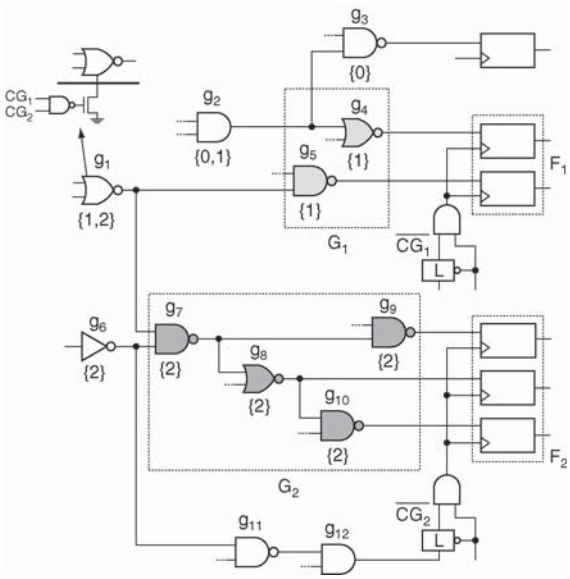

Fig. 6. Grouping gates under the functionality constraint.

($CG_i = 0$) or blocks ($CG_i = 1$) a clock toward a set of flip-flops F_i. The AMPG synthesis problem is to derive, for each CG_i, a group of gates G_i that are power-gated together and to determine the size of their footer.

We assume that the clock-gating logic, which generates the signals CG_i is already embedded in the circuit, which is either specified during register-transfer level design or automatically synthesized from a gate-level netlist [9]; we relied on the latter approach in the experiment reported in Section V. The synthesis problem is subject to three constraints, namely, functionality, timing, and current.

TABLE II
AGGREGATE RESULTS OF POWER-GATING UNDER THE FUNCTIONALITY CONSTRAINT

Circuit	Total No. of Gates	No. of Gates that Can be Power-Gated by				
		1 CG_i	2 CG_is	3 CG_is	4 CG_is	$\geq$ 5 CG_is
aes	5677	3098	153	6	1119	136
aquarius	16 344	2386	13	0	15	3
pcm	191	114	0	0	0	0
ps2	1603	733	24	0	0	0
ucore	10 128	3466	108	3	0	0
warp	21 935	4860	373	53	21	4
wbdma	3987	1478	97	37	9	2
b14	4297	887	119	1190	439	990
b17	16 715	3266	63	2	0	77
b18	22 101	7050	1010	395	69	280

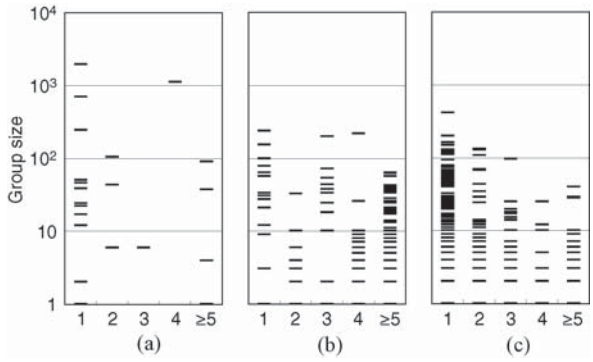

Fig. 7. Distribution of group size for different numbers of clock-gating signals. (a) aes. (b) b14. (c) b18.

1) *Functionality Constraint:* Identifying the gates that can be power-gated by CG_i is straightforward as regards functionality. In Fig. 6, g_4 and g_5 can be power-gated by CG_1 because they generate the inputs to the flip-flops in F_1; thus when $CG_1 = 1$, those flip-flops do not load their inputs, and so there is no need for g_4 and g_5 to function. The same applies to g_7, g_8, g_9, and g_{10}. Gate g_3 has to be active because its output goes to the topmost flip-flop, that is never gated; the same holds for g_2, even though it is also involved in processing the data loaded by the upper flip-flop in F_1. The gates g_6, g_{11}, and g_{12} are not power-gated because they are part of the clock-gating logic: these three gates determine whether CG_2 is 1 or 0.

An appropriate grouping can be obtained as follows. From each flip-flop that belongs to F_i, an index i (= $1, 2, \ldots$) is propagated toward the primary inputs, and an index 0 is propagated from the flip-flops that are not gated and from the primary outputs. The gates that have a single nonzero index i become a member of G_i. From each G_i, we then discard the gates that are involved in generating clock-gating signals; g_6 is an example of such a gate.

There are also some gates that are power-gated only when more than one clock-gating signal is active, such as g_1. If n is the number of clock-gating signals, then an n-input NAND gate together with a footer can be used to gate such a gate, as shown in Fig. 6. Table II shows the number of gates that can be power-gated by different numbers of clock-gating

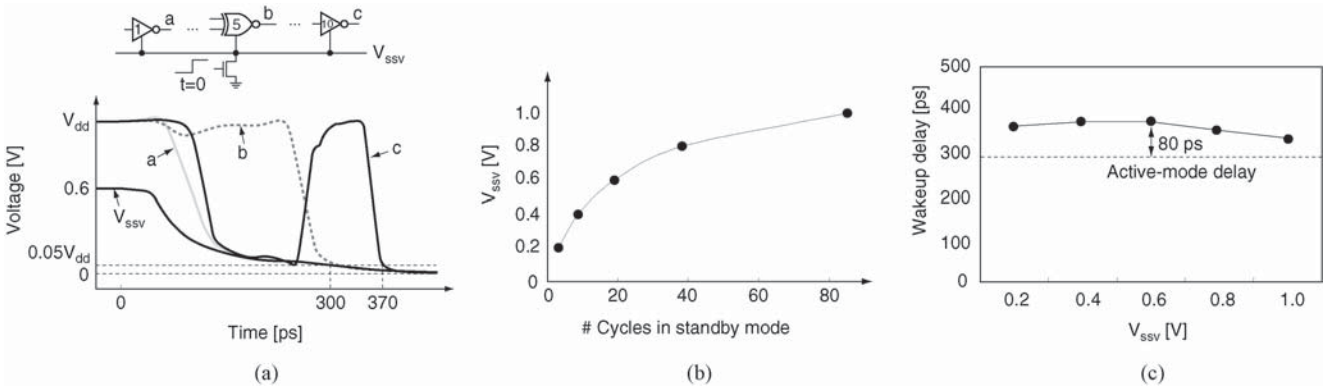

Fig. 8. (a) Transient waveforms of V_{ssv} and internal node voltages during the wakeup process for circuit c5315. (b) Standby-mode V_{ssv} versus the number of cycles remaining in standby mode. (c) Wakeup delay for different values of V_{ssv}.

signals, for the test circuits of Section V; it is apparent that the majority of gates can be power-gated by a single clock-gating signal. Circuits aes, b14, and b18 are exceptions; Fig. 7 shows the distribution of group size against different numbers of clock-gating signals controlling each group. We note that some groups are very small, especially in b18. Small groups may not be good candidates for power-gating, because of the overhead of a NAND gate and footer (see Fig. 6), as well as their disproportionate contribution to the cost of physical design (see Section IV).

This suggests that we need to find a minimum group size, below which power-gating is not cost-effective; this can only be determined empirically, as we will discuss in Section V-C. To simplify our presentation, we will focus on Problem 1 for the remainder of this paper unless otherwise stated; the generalization to multiple clock-gating signals is not difficult.

2) *Timing Constraint:* In Fig. 1, a latch within each clock-gating cell becomes nontransparent when CLK = 1, so that the value of CG_i remains constant, and is unaffected by any glitches from the clock-gating logic. If we set CG_i to 1, then the flip-flops in F_i become CG and the gates in G_i become power-gated. We assume for now that the clock-gating logic produces the wakeup signal before the falling edge of CLK, which has a duty-cycle of 0.5. CG_i becomes 0 at the falling edge of CLK, then two things have to happen before the next rising edge when normal operation is supposed to begin. $V_{ssv,i}$ must return to its nominal value, say 5% of V_{dd}, and all the gates in G_i must return to their original logic values, which are those that they had before the footer was turned off. We will call the delay involved in these operations the wakeup delay, and this must not exceed half a clock period; this constitutes the timing constraint. When CG_i becomes 0 after the falling edge, the timing constraint becomes correspondingly tighter.

It should be noted that the wakeup delay cannot be found by simply summing its two components. This is illustrated in Fig. 8(a) for c5315, one of the ISCAS benchmark circuits. We will assume that V_{ssv} has been at 0.6 V before time 0, when the footer is turned on; it returns to 5% of V_{dd} at 300 ps. Fig. 8(a) shows the voltage waveforms at three nodes, *a*, *b*, and *c*, on the timing-critical path. During standby mode, these voltages are all close to V_{dd}. Assuming that all three nodes were at logic 0 before entering standby mode, they make transitions (node *c* makes multiple transitions due to a glitch) back to 0 during the wakeup process. This is similar to active-mode switching except that it takes slightly more time because V_{ssv} also makes a transition. Node *c* has completed its transition after 370 ps, and the corresponding active-mode delay is 290 ps.

The standby-mode V_{ssv} is a function of the number of cycles spent in standby mode; this is illustrated in Fig. 8(b). We repeated the experiment that produced Fig. 8(a) while varying the initial V_{ssv} from 0.2 V to 1.0 V and measuring the time at which *c* completes its transition. The results, shown in Fig. 8(c), indicate that the difference between the wakeup delay and the active-mode delay is largely unaffected by V_{ssv}. The same was observed in experiments with other circuits, which suggests that 100 ps would be a conservative estimate of the difference between the delays for all the circuits that we tested, provided that V_{ssv} discharges fast enough, as it does in Fig. 8(a). Since we use a footer that is large enough to sink all the discharge currents encountered, this condition is always satisfied.

Once we have found G_i that meets the functionality constraint, we check its maximum delay by means of a static timing analysis (STA). If the delay plus the timing margin (this is the 100 ps that we just discussed) exceeds the timing constraint, then we drop a gate that leads a critical path [this might be gate 1 in Fig. 8(a)]; we repeat this process until the timing constraint is satisfied.

3) *Current Constraint:* Once we have found a group of gates G_i that meets the functionality and timing constraints, we need to determine the size of the footer that will be attached to it (see Fig. 1). This choice must take two factors into account [10]: the discharge current during the wakeup process, also called the rush current, and the active-mode circuit delay. These are conflicting requirements: the instantaneous rush current through the footer is more likely to be sustainable by the ground rail if the footer is small, since smaller footer can drain smaller amount of current (but at the cost of longer time to drain all charges); but there is a smaller active-mode voltage drop across a large footer, and this in turn reduces the circuit delay.

The power rails are typically designed to sustain the maximum discharge current (MDC). Therefore, after we have estimated the MDC of G_i, a task which we address in

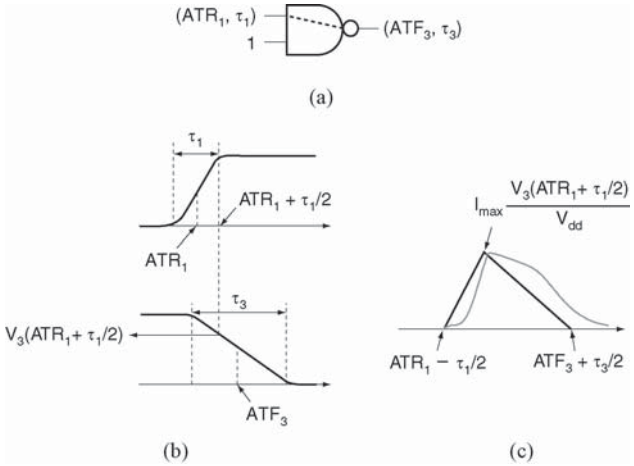

Fig. 9. Modeling discharge current. (a) Two-input NAND gate. (b) Input and output waveforms. (c) Discharge current model, shown with the waveform from SPICE.

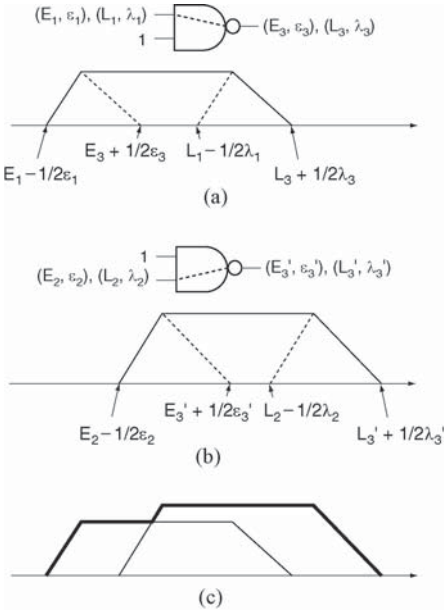

Fig. 10. Gate-level estimation of MDC: current profiles corresponding to a rising signal. (a) At the first input. (b) At the second input. (c) Final current profile.

Section III-B, we need to size the footer so that the rush current through it does not exceed this MDC.

To accommodate the active-mode circuit delay, we employ the widely used average-current method (ACM) [11], [12], [13]. In a practical design, we often require the voltage drop ΔV across a footer to be very small (say 1% of $V_{\rm dd}$) so that power-gating does not introduce any noticeable increase in the delay. In this situation, the ADC can be used in the footer sizing process, because ΔV is weakly dependent on the input pattern [11]. Notice that, in theory, MDC must be used for this sizing; however, pessimisms that are involved in typical MDC estimation may cause unnecessarily large footer size, which is why ACM is more popular. After the ADC of G_i has been determined, an activity which we also address in Section III-B, we calculate ΔV using the footer size determined from the rush current constraint. If $\Delta V > \epsilon V_{\rm dd}$ for some ϵ (say 1%), we drop gates from G_i until the new ΔV (which is smaller because it is derived from the smaller ADC) is less than $\epsilon V_{\rm dd}$. The details of this process will be explained in Section III-C.

B. Estimation of the MDC and ADC

Fast yet accurate estimation of the MDC and ADC is important to determine a footer size that satisfies the current constraint and then to refine G_i accordingly.

1) *Discharge Current Model:* An isosceles or right-angled triangle is a popular shape for the current model [14], [15], as either is simple to implement. However, we have introduced [1] a more accurate model with an adjustable shape, which is determined from timing information obtained by STA.

Let us consider two-input NAND gate shown in Fig. 9(a). The arrival time of the rising input signal ($\rm ATR_1$) and its transition time (τ_1) are obtained by STA; the other input is assumed to be at logic high. We want to estimate the discharge current at the output in this situation.

The arrival time of the falling output signal ($\rm ATF_3$) and its transition time (τ_3) can also be obtained through STA. The waveforms of the input and output signals are illustrated in Fig. 9(b). The output discharge current can be approximated by the triangle shown in Fig. 9(c). As soon as the input voltage exceeds the threshold voltage of nMOS devices, a discharge current starts to flow (initially together with a short-circuit current); this occurs at approximately the same time as the input transition starts, which is $\rm ATR_1 - \tau_1/2$. Discharge is complete at the end of the output transition, which occurs at $\rm ATF_3 + \tau_3/2$. We can make a reasonable approximation to reality by assuming that the discharge current is at its peak when the input transition completes (i.e., when $V_{\rm gs}$ is at its maximum), which occurs at $\rm ATR_1 + \tau_1/2$ [see Fig. 9(b)]. The value of the peak current is assumed to be proportional to the output voltage, as follows:

$$I_{\rm peak} = I_{\rm max} \frac{V_3(\rm ATR_1 + \tau_1/2)}{V_{\rm dd}} \tag{1}$$

where $I_{\rm max}$ is the MDC, which is unique to the particular NAND gate and is characterized *a priori*, and $V_3(t)$ is the value of V_3 at time t, which can be obtained from Fig. 9(b). In Fig. 9(c), we see the triangular current model created by this process superimposed on the SPICE waveform, which demonstrates its accuracy.

For a complex gate, such as AND gate, two discharge current models are set up: one when the output discharges (the internal inverter is responsible for discharge current); the other when the output charges (the internal NAND is responsible for discharge current). Either ATR or ATF is not available in the internal gates, thus the approximation is made. One of the models are chosen from the input pattern.

2) *Estimation of the MDC:* The time at which a particular gate discharges depends on the input pattern applied to the circuit. To estimate MDC without resorting to pattern information, we look at the time intervals during which each gate is able to discharge. Consider again the two-input NAND gate shown in Fig. 10(a). We will now specify the ATR of the

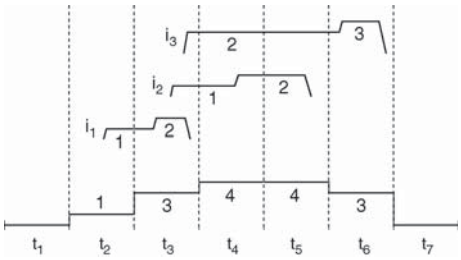

Fig. 11. Circuit-level estimation of MDC.

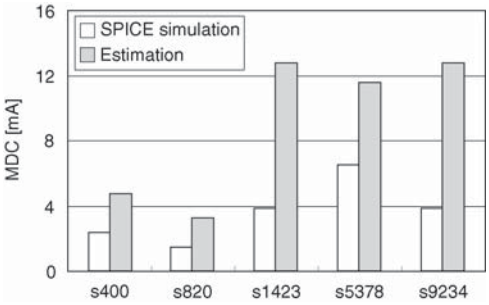

Fig. 12. Comparison of analytic estimates (gray bars) of the MDC with the results of SPICE simulations (white bars).

first input as an interval bounded by E_1 and L_1 which are the earliest and latest ATR, respectively. We can then denote the transition time corresponding to each of these limits by ϵ_1 and λ_1. All these quantities can be obtained by STA. By assuming that the second input is at logic high, we can obtain the interval that contains the ATF at the output, $[E_3, L_3]$, and the corresponding transition times. Two triangular current models can then be constructed by following the method in Section III-B1. The peaks of these triangles can then be connected in the way shown in Fig. 10(a), reflecting an assumption that intermediate peak currents can be obtained by linear interpolation. This process is then repeated for the rising signal at the second input, which yields another current profile, as shown in Fig. 10(b). Finally, we obtain an envelope for the two current profiles, as illustrated in Fig. 10(c), which represents the MDC of the NAND gate.

After we have obtained current profiles for all the gates, we can estimate the MDC of the whole circuit. This process is illustrated in Fig. 11. We will assume that there are three current profiles, i_1, i_2, and i_3. And we will divide clock period into a sequence of timeframes, $t_1, t_2, \ldots, t_7$. We also find sets of gates whose times of discharge overlap [15]; let us suppose that the gates corresponding to i_2 and i_3 belong to the same set. At t_2, the MDC will be approximately 1, which is the maximum value of i_1 in that timeframe. At t_3, the sum of i_2 and i_3 is 3, while i_1 is 2, and thus the MDC is set to 3. This process is repeated for the remaining timeframes.

Analytic MDC estimates for several test circuits are compared with results from SPICE simulations in Fig. 12. The simulations were run for 10 000 random input patterns to yield the MDC. We see that our analytic approach yields an MDC which is on average 2.5 times that produced by SPICE. This compares favorably with the factors of 2.7 [15] and 3.4 [16] produced by previous methods, even though these methods

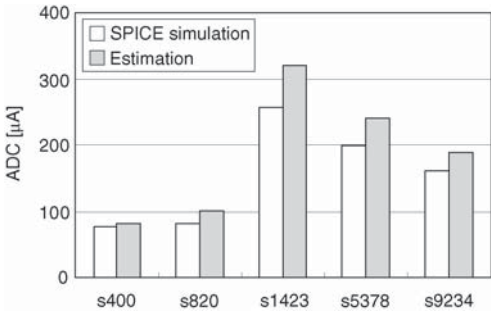

Fig. 13. ADCs of five circuits estimated analytically and compared with SPICE simulations.

Algorithm *AMPG_Synthesis*
Input: netlist of clock-gated circuit
Output: the same netlist, with G_is identified and connected to properly sized footers

L1 **for** each CG_i **do**
Functionality constraint:
L2 Identify a G_i that respects the functionality constraint
Timing constraint:
L3 **while** (delay of G_i + margin) > timing constraint **do**
L4 Remove a gate from G_i that leads a critical path
Current constraint:
L5 $I_{MDC} \leftarrow$ MDC of G_i
L6 Size footer s.t. (its current when $\Delta V = V_{dd}$) $< I_{MDC}$
L7 $I_{ADC} \leftarrow$ ADC of G_i
L8 **while** $\Delta V|_{I=I_{ADC}} > \varepsilon V_{dd}$ **do**
L9 Remove a gate j with max $I_{j,av}$ and leads G_i
L10 Update I_{ADC}

Fig. 14. AMPG synthesis algorithm.

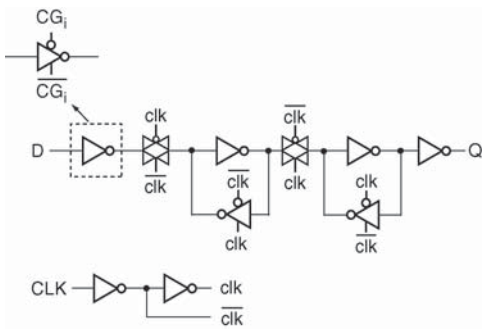

Fig. 15. Floating-prevention flip-flop.

were compared against gate-level simulations with random input patterns rather than SPICE simulations.

3) *Estimation of the ADC:* We use the same discharge current model to estimate the ADC. Consider a two-input NAND gate with the signal probabilities (p_1 and p_2 are the probabilities of each input being at logic high) and the transition probabilities (t_1 and t_2 are the probabilities of a rising or falling transition). These probabilities can be evaluated by propagating the signal probabilities of the primary inputs [17], usually specified by designers.

The output of this NAND gate can be represented by two discharge current models, constructed using the method described

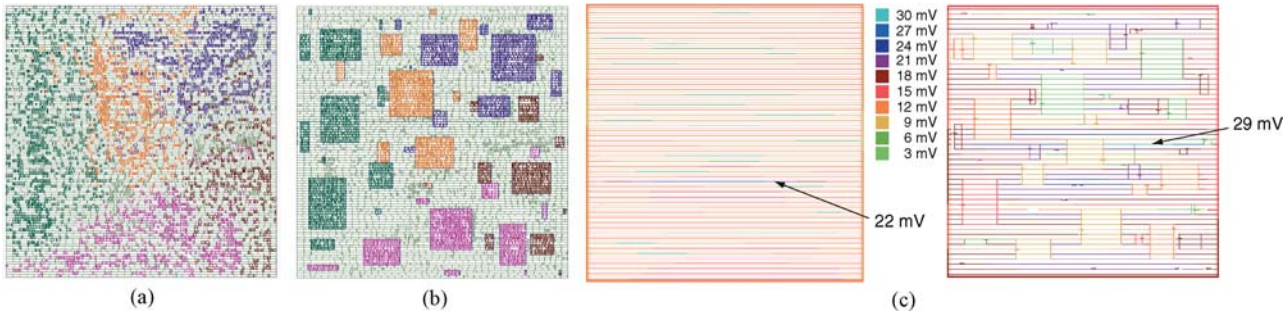

Fig. 16. (a) Initial placement of circuit s35932; different colors indicate the cells of different G_i. (b) Creating bounding boxes, constrained placement, power network synthesis, and footer insertion. (c) Comparison of the IR drop on the V_{ss} rail of (a) and on the V_{ssv} rails of (b).

in Section III-B1, with currents I_3 corresponding to input 1 and I_3' to input 2. Current I_3 flows when input 1 makes a rising transition, which has a probability of $t_1/2$, and input 2 is at logic high, which has a probability of p_2; the converse situation produces I_3'. The ADC over a clock period T_c can now be expressed as

$$I_{3,av} = \left(\frac{t_1}{2} p_2 \int I_j \, dt + \frac{t_2}{2} p_1 \int I_j' \, dt \right) / T_c. \tag{2}$$

The same process is repeated at all nodes of G_i and the sum of all these currents can be taken to represent the ADC.

We estimated the ADC of several test circuits; the results are shown in Fig. 13, where they are also compared with the results of SPICE simulations. We see that our analytic estimation of ADCs is 18% higher on average.

C. AMPG Synthesis Algorithm

Our synthesis algorithm is summarized in Fig. 14. The three constraints are checked one by one to obtain a group of gates G_i that can be power-gated by CG_i and to decide the size of footer that will be attached to G_i, which constitutes the output.

An initial group of gates that respects the functionality constraint is obtained in step L2. If the maximum delay of G_i returned by STA, added to a timing margin to accommodate the wakeup delay, is larger than the timing constraint (L3), then we remove a gate that leads a critical path (L4); this process is repeated (L3) until the timing constraint is satisfied. The MDC is then obtained (L5), and the footer is sized (L6) in such a way that the maximum current drained by a footer during the wakeup process does not exceed the MDC; the voltage across a footer (ΔV) is conservatively estimated to be V_{dd}, because that voltage produces the highest discharge current. We obtain the ADC (L7) and thus obtain ΔV using the footer size determined in step L6. If ΔV is larger than ϵV_{dd}, we drop a gate that leads G_i, which has the maximum ADC (L9).

Care needs to be taken with flip-flops that are CG. If the gates that fan in to such flip-flops are power-gated, then a large short-circuit current may flow through the flip-flops, since their inputs are floating; one particular consequence of this is that the voltage at the flip-flop inputs may rise very slowly with the rise of V_{ssv}. This can be resolved by modifying the standard design of flip-flop so that its input is decoupled when its clock is gated ($CG_i = 1$). This can be achieved using a tristate inverter instead of a standard inverter, as illustrated in Fig. 15. Our experiments suggest that the cost of this modification is a 17% increase in setup time and an 18% increase in layout area.

IV. Physical Design of AMPG Circuits

A group of gates G_i that are power-gated by a clock-gating signal CG_i has its own virtual ground rail $V_{ssv,i}$ (see Fig. 1). The gates that are not power-gated have V_{ss} as their usual ground rail. Therefore, if there are N groups of gates, there will be $N+1$ ground rails and one V_{dd} rail, which makes the placement of AMPG circuits a significant challenge.

The simplest approach is row-based placement [1], [6], which is also popular for the placement of dual-V_{dd} circuits [18]; in this scheme, each placement row is exclusively occupied by the gates of a single group. The quality of physical designs produced by this approach is somewhat discouraging; a 30% increase in wirelength [1] has been reported, compared to unconstrained standard placement, in situations where the latter is possible. Another approach is to design standard cells with a configuration which allows their ground connections to be supplied by signal routing [5] rather than standard V_{ss} rails. But this approach is hard to translate into practical designs because it reduces power network integrity and increases routing congestion.

Our approach can be explained using Fig. 16. An initial placement is performed with the AMPG circuitry but without footers, as shown in Fig. 16(a). This is a CG circuit, which is an input to the AMPG synthesis presented in Section III, but the G_is are identified and floating-prevention flip-flops are used wherever they are needed. The absence of footers means that there are no virtual ground rails yet. The cells that belong to the same G_i [shown in the same color in Fig. 16(a)] are grouped together, as far as is possible without altering the initial placement too much. A bounding box is created around each group for use by the constrained placement that follows. A power distribution network is then constructed around each bounding box to serve as the V_{ssv} rail, and the requisite numbers of footers are inserted. Fig. 16(b) shows the final layout.

A. Power Network Synthesis

Fig. 17(a) is a diagrammatic representation of the result of constrained placement, in which two bounding boxes are depicted. We will assume that a double-back layout pattern is being used. This consists of alternate rows of cells and

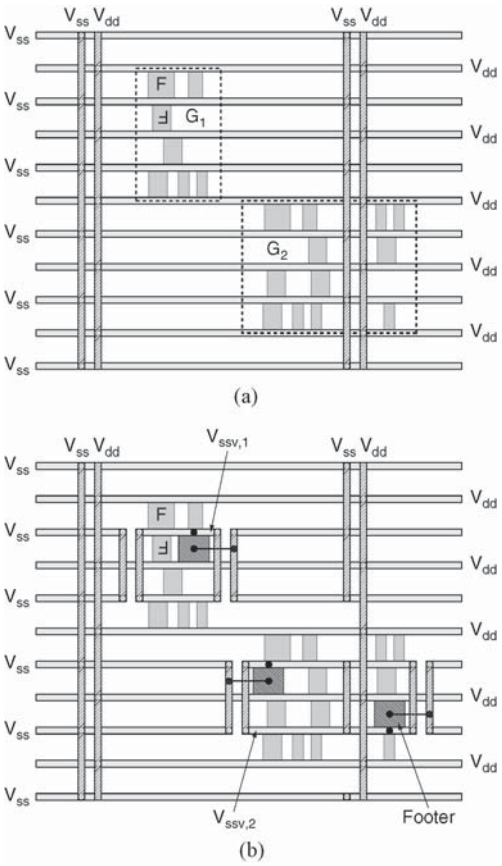

Fig. 17. (a) Constrained placement with bounding boxes. (b) Power network synthesis and footer insertion.

Algorithm *AMPG_Placement*

L1 Perform unconstrained initial placement
L2 **for** each group of power-gated gates G_i **do**
L3 Initialize a list $\mathcal{L} \leftarrow G_i$
L4 **while** $\mathcal{L}$ is not empty **do**
L5 $R_j \leftarrow \{c_k \in \mathcal{L} | \text{dist}(c_j, c_k) \leq \text{RoI}\}$, for each $c_j \in \mathcal{L}$
L6 $m \leftarrow \text{argmax}_j |R_j|$
L7 Create a bounding box for R_m
L8 $\mathcal{L} \leftarrow \mathcal{L} - R_m$
L9 Resolve overlaps between bounding boxes
L10 Perform constrained placement with bounding boxes
L11 Synthesize a power distribution network
L12 Place footers

Fig. 18. Placement algorithm.

of flipped cells; the horizontal V_{dd} and V_{ss} rails alternate as a result. The power distribution network is completed by the vertical rails which carry V_{dd} and V_{ss}, alternately, with a predetermined spacing.

Fig. 17(b) illustrates how the V_{ssv} rails are constructed. The V_{ss} rails are cut as they enter each of the bounding boxes shown in Fig. 17(a). Then the disjoint sections of the V_{ss} rails within each bounding box are connected together to become the V_{ssv} rails. Note that each V_{ssv} network in Fig. 17(b) is necessarily smaller than its corresponding bounding box in Fig. 17(a), because only V_{ssv} is partitioned in this way, whereas V_{dd} is shared across the whole layout.

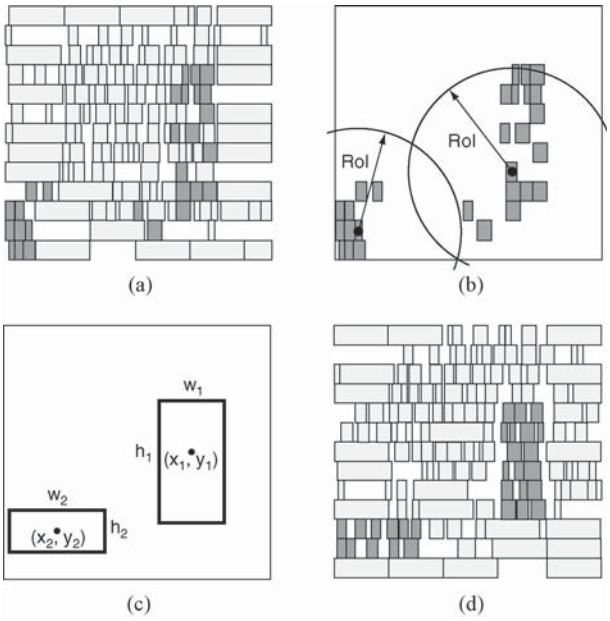

Fig. 19. (a) Initial placement. (b) Counting the number of cells within the region of interest (RoI). (c) Creating bounding boxes. (d) Constrained placement.

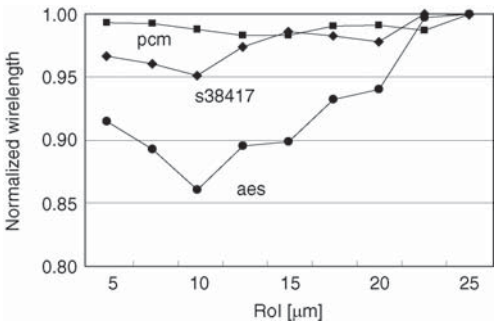

Fig. 20. Normalized wirelength after constrained placement and routing with varying RoI.

Placement is finalized by inserting footers, which can also be seen in Fig. 17(b). Each footer is connected to its adjacent V_{ss} as well as its local V_{ssv}. The number of footer cells is determined by dividing the footer size that respects the current constraint by the size of a single footer cell. When there is more than one bounding box for a single G_i [several bounding boxes of the same color can be seen in Fig. 16(b)], the footers are distributed in proportion to the size of each bounding box, which will be roughly proportional to the number of cells it contains. For more accurate footer sizing, MDC and ADC may be estimated in each bounding box again and footer size is adjusted accordingly.

We assessed this form of proposed power distribution network through an IR drop analysis of several circuits. Fig. 16(c) compares the drop on the V_{ss} rail after initial placement (a), which has a maximum value of 22 mV, with the drop on the V_{ssv} rails after final placement (b), which has a maximum of 29 mV. The difference in the IR drop is modest, if we consider the fact that the drop measured at the V_{ssv} rails includes the drop on the V_{ss} rail and the drop across the footers. This

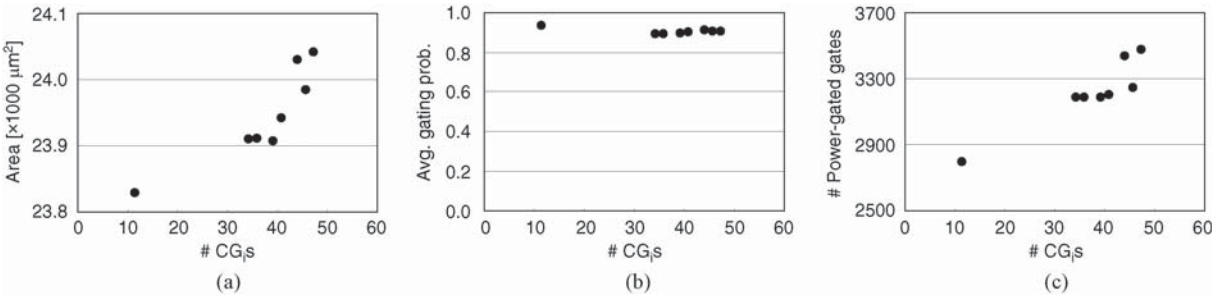

Fig. 21. (a) Circuit area. (b) Average gating probability. (c) Number of gates that are power-gated, versus the number of clock-gating signals in warp circuit.

analysis also showed an acceptably small increase in the IR drop on the V_{ssv} rails. Overall, these results suggest that this form of power distribution network will be satisfactory.

B. Placement

The placement algorithm is shown in Fig. 18, and Fig. 19 provides an illustrative example. An initial placement such as the one shown in Fig. 16(a) is performed in step L1 of the algorithm. A group of gates G_i is assembled (in no particular order) one by one (L2), and a set of bounding boxes is created (L3–L8). For each cell c_j that is in the list $\mathcal{L}$, we count the number of other cells that intersect with a circular region centered on c_j, with radius RoI (L5); Fig. 19(b) shows how these neighboring gates are counted for two example cells. The algorithm selects the cell c_m with the maximum number of neighboring gates (L6). Thus, we may expect c_m to lie at the center of the region that is most densely populated by the cells of G_i; and R_m is the set of cells within such region.

A bounding box is then created (L7), and all the cells in R_m will be forced to stay within that box during the constrained placement [19] process (L10). The area of this bounding box is set to the sum of the areas of all the cells in the R_m, divided by the design figure for area utilization (e.g., 70%). The aspect ratio of the bounding box (its width divided by its height) is set to the ratio between standard deviation of x and y-coordinates of the cells, and its center is located at the average value of these coordinates. Fig. 19(c) shows two bounding boxes created in this way. All the cells of R_m are removed from $\mathcal{L}$ (L8), and the process is then repeated for another bounding box.

Overlaps between bounding boxes can be eliminated by moving the offending boxes (L9), but overlaps turned out to be rare occurrences in our experiments. Final placement using bounding boxes is then performed (L10), as illustrated in Fig. 19(d). A power distribution network is synthesized (L11) as described in Section IV-A, and then at last the footers are inserted (L12).

The RoI is an important parameter, because it affects the quality of placement as well as routing. An RoI that is too small causes too many bounding boxes to be created in the placement region. This is undesirable because each box imposes an irreducible overhead on physical design in terms of its placement constraint, contribution to routing congestion, and the space required between the V_{ss} and V_{ssv} rails [see Fig. 17(b)]. Conversely, an RoI that is too large leads to the creation of a small number of cumbersomely large bounding boxes, which are likely to cause too much change to the initial placement. Fig. 20 shows what happens to the total wirelength after placement and routing as we vary the size of RoI. These results clearly suggest that some optimum value does indeed exist, and we chose 10 μm as the radius of the RoI in our experiments.

V. Experimental Results

We chose several sequential circuits from OpenCores [20] for these experiments. Our selections include parts of processor cores (aquarius and ucore), multimedia cores (aes and warp), and control circuits (pcm, ps2, and wbdma). Three more circuits were added from the ITC benchmark to make up the list of test circuits shown in Table III. Each circuit was synthesized to a gate-level netlist using industrial 1.1 V 45-nm technology. The clock-gating logic was included in the logic synthesis process [21], which was set up to ensure at least three flip-flops were gated by each clock-gating signal. Columns 2–4 of Table III list the numbers of combinational gates, flip-flops, and primary outputs, respectively.

A. Result of AMPG Synthesis

The AMPG_Synthesis algorithm presented in Section III was implemented in OpenAccess [22]. Columns 5 and 6 of Table III report the number of clock-gating signals and CG flip-flops (as a percentage) produced by the clock-gating synthesis. The last two columns, 7 and 8, give the average gating probability of the CG flip-flops and the number (and proportion) of the combinational gates that are power-gated during the synthesis. These two figures have a strong bearing on the saving in active leakage.

The AMPG_Synthesis algorithm receives a CG netlist as one of its input. This means that the way in which clock-gating has been performed to produce the netlist affects the quality of the final circuit. We varied the minimum number of flip-flops that are CG together, which is a control parameter of the clock-gating synthesis [21], and generated a series of netlists; each netlist is then submitted to AMPG_Synthesis to generate its AMPG version. As the number of clock-gating signals increases, more flip-flops tend to be CG, which allows more gates to be power-gated, as shown in Fig. 21(c); however, the average probability of gating does not change much, as shown in Fig. 21(b). This suggests that active leakage could

TABLE III
TEST CIRCUITS AND RESULTS OF AMPG SYNTHESIS

Circuit	No. of Gates	No. of F/Fs	No. of Primary Outputs	No. of CG_is	Gated F/Fs (%)	Average Gating Prob.	No. of Gates That Are Power-Gated
aes	5677	670	129	20	94	0.91	3775 (67%)
aquarius	16 344	1202	73	33	82	0.90	2921 (18%)
pcm	191	87	9	4	83	0.84	94 (49%)
ps2	1603	238	43	14	60	0.95	495 (31%)
ucore	10 128	1192	190	27	84	0.12	3212 (32%)
warp	21 935	1640	103	44	52	0.92	3438 (16%)
wbdma	3987	987	217	22	61	0.89	1409 (35%)
b14	4297	212	54	16	80	0.78	1348 (31%)
b17	16 715	1314	97	56	40	0.93	2340 (14%)
b18	22 101	1616	23	87	53	0.81	5836 (26%)

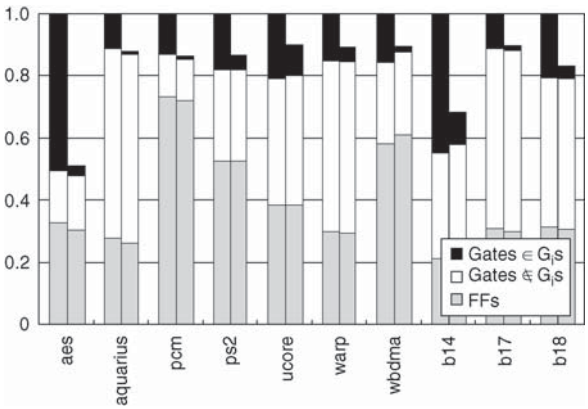

Fig. 22. Comparison of (normalized) active leakage between CG (left-hand bars) and AMPG circuits (right-hand bars).

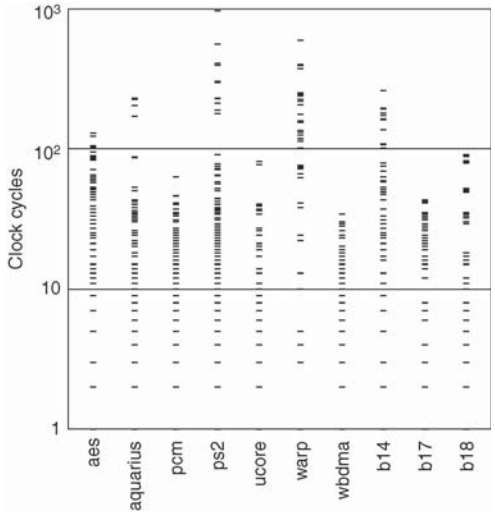

Fig. 23. Distribution of idle periods (when each CG_i is set to 1).

be reduced further if there were more clock-gating signals; but these come at a cost in circuit area, as shown in Fig. 21(a), because of the need for more clock-gating cells, footers, and floating-prevention flip-flops.

B. Power Consumption

The active leakage of CG and AMPG circuits, obtained by circuit simulation [7], is compared in Fig. 22. Three circuit components are identified: gates that are power-gated, other combinational gates, and flip-flops. The average overall saving in leakage is 18%, but there is wide variation from circuit to circuit, which is to be expected. Circuits aes and b14 benefit most, primarily because they have a large proportion of combinational gates; but they also have many gates that can be power-gated, and a high gating probability (see Table III).

In the original CG circuits, active leakage represents 45%, on average, of the total active-mode power consumption; this is larger than the 28% number reported in Section II-B because the switching current has now been reduced due to clock gating. The average active-mode power consumption of the AMPG versions of the circuits is 12% less than that of the CG circuits.

To reduce active leakage of a flip-flop, a part of it may be power-gated. This can be achieved by power-gating the master latch as well as the inverter and transmission gate (see Fig. 15), similar to retention flip-flop used in standard power-gating [10]. The modified flip-flop was tested in circuits aes, ps2, and b14. The additional saving in leakage turned out to be modest, about 3%; this comes at a cost of substantial increase of wirelength, about 19%, after physical design because flip-flops now have to stay within their corresponding bounding boxes during placement.

Power-gating comes at a cost of extra energy to turn off and on footers and to charge and discharge virtual ground rail. Thus, there exist minimum number of clock cycles during which power-gating is applied (or clock is gated), such that extra energy is outweighed by saving in leakage. An analytical model has been proposed [23] for this purpose, which we employ. With typical values being used for model parameters and for various gate groups we tested, the minimum value turned out to range from five to seven clock cycles. Fig. 23 illustrates a distribution of idle periods, when 1000 random vectors are applied to each circuit. The majority of the periods lie above ten clock cycles, which suggest that leakage is indeed saved in most of the periods.

C. Physical Design

The sum of the areas of all the cells of the original CG circuits and the AMPG circuits is given in columns 3 and 4

TABLE IV
COMPARISON OF AREA, TOTAL WIRELENGTH, AND POSTLAYOUT DELAY OF CG AND AMPG CIRCUITS

Circuit	# BBs	Area (μm^2)			Wirelength (mm)			Delay (ns)	
		CG	AMPG	Inc (%)	CG	AMPG	Inc (%)	CG	AMPG
aes	98	7525	7729	2	91	106	17	1.48	1.47
aquarius	231	17 504	17 874	2	173	192	11	0.82	0.85
pcm	9	490	501	2	2	2	11	0.26	0.27
ps2	16	2246	2294	2	15	16	7	0.71	0.72
ucore	100	12 409	12 678	3	137	142	3	1.21	1.26
warp	119	23 308	23 722	5	194	199	3	1.40	1.45
wbdma	67	7309	7451	3	86	94	10	0.91	0.90
b14	61	3983	4113	3	39	42	7	0.88	0.93
b17	92	18 534	19 511	2	171	193	13	0.45	0.45
b18	221	23 873	24 521	2	220	240	9	1.29	1.28
Average				3			9		

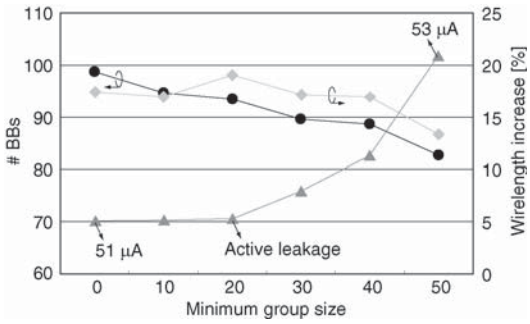

Fig. 24. Number of bounding boxes, wirelength increase (compared to the equivalent CG circuit), and active leakage of the aes circuit for different values of the minimum group size.

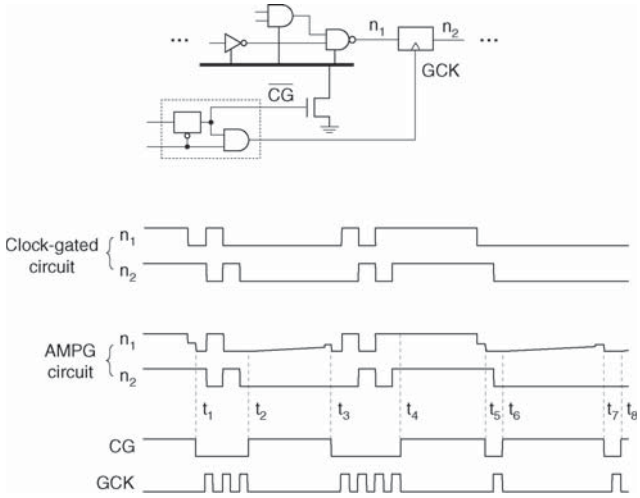

Fig. 25. Operation of CG and AMPG versions of the ps2 circuit; n_1 and n_2 are the sample nodes that we probed.

of Table IV, and column 5 indicates the percentage change. The area of the AMPG circuits is 3% larger on average, which we would expect because of the inclusion of footers and floating-prevention flip-flops. The CG circuits were placed using a commercial placement tool [19]; the AMPG circuits were placed using AMPG_Placement algorithm, which was implemented using Python and Tcl script to the same commercial placement tool. Utilization of the placement area was set to 70%. Routing was subsequently performed assuming six layers of metal.

The total wirelength is compared in columns 6–8 of Table IV. It is longer in AMPG circuits by 9% on average, which is substantially better than the 30% reported for row-based placement [1]. To assess the impact of the longer wires on circuit timing, postlayout STA was performed and the critical-path delay was compared. The delay in the AMPG circuits is 4% higher on average as reported in the last two columns, as well as the longer wires, this is caused by the increased setup time needed for the floating-prevention flip-flops.

The variation of increase in wirelength between circuits (column 8 of Table IV) can be understood if we compare the number of clock-gating signals (CG_is) in column 5 of Table III and the number of bounding boxes (# BBs) that were created, which is given in the second column of Table IV. For example, 98 bounding boxes were created for the aes circuit, even though there are only 20 clock-gating signals, increasing the wirelength by 17%; the corresponding figures for the ps2 circuit are 16 and 14, which cause 7% increase of wirelength. This suggests that reducing the number of bounding boxes favors a lower wirelength; but fewer bounding boxes reduces the number of gates that can be power-gated, thus allowing more leakage. We demonstrated this experimentally using the aes circuit, with the results shown in Fig. 24. We varied the minimum group size used by the AMPG_Synthesis routine, which does not power-gate groups with fewer gates than specified size. As the minimum group size is increased, fewer bounding boxes are created (left-hand y-axis), which stems the decrease in wirelength (right-hand y-axis), at the cost of more active leakage.

D. Verification

The operation of an AMPG circuit was verified by SPICE simulation. We compared CG version of the ps2 circuit (which is the original version of this circuit and serves as a reference) with the AMPG version. Part of this circuit is shown in Fig. 25, which indicates the two nodes that we probed. It can readily be seen that the flip-flop in the AMPG circuit correctly captures n_1 when the clock is not gated. A series of time intervals during which wakeup (CG = 0) and sleep (CG = 1) alternate

are denoted by $t_1, \ldots, t_8$. At t_1, node n_1 returns to 0, which is its value (see node n_1 of the CG circuit) during the discharging process; n_1 starts to float at t_2 once the footer is turned off, but n_1 does not float at t_4, even though the footer is turned off, because its original value in the CG circuit is logic high.

VI. Conclusion

We have performed a quantitative analysis of active leakage in 45-nm technology. We have seen that the extent of the stacking effect, temperature, and the clock period must be taken into account in assessing the importance of active leakage in total active-mode current and in comparing it with standby leakage. We have described active-mode power-gating, which is a circuit technique that can be used together with dual-V_t and standard power-gating (in which case the whole circuit is also power-gated during standby mode). We have itemized the principal issues that occur in the synthesis of AMPG circuits: the grouping of gates and footer sizing while preserving functionality, circuit timing, and the discharge current during wakeup and in active mode. We have suggested how integrated placement and power network synthesis can be used in the physical design of AMPG circuits.

As we have shown in experiments, the implementation of clock-gating affects the quality of the resulting AMPG circuits. The integration of clock-gating with AMPG synthesis is therefore a topic that deserves further investigation.

References

[1] J. Seomun, I. Shin, and Y. Shin, "Synthesis and implementation of active mode power gating circuits," in *Proc. Des. Automat. Conf.*, Jun. 2010, pp. 487–492.

[2] K. Usami and H. Yoshioka, "A scheme to reduce active leakage power by detecting state transitions," in *Proc. Int. Midwest Symp. Circuits Syst.*, Jul. 2004, pp. 493–496.

[3] Y. Shimazaki, R. Zlatanovici, and B. Nikolic, "A shared-well dual-supply-voltage 64-bit ALU," *IEEE J. Solid-State Circuits*, vol. 39, no. 3, pp. 494–500, Mar. 2004.

[4] P. Royannez, H. Mair, F. Dahan, M. Wagner, M. Streeter, L. Bouetel, J. Blasquez, H. Clasen, G. Semino, J. Dong, D. Scott, B. Pitts, C. Raibaut, and U. Ko, "90nm low leakage SoC design techniques for wireless applications," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2005, pp. 138–139.

[5] K. Usami and N. Ohkubo, "A design approach for fine-grained run-time power gating using locally extracted sleep signals," in *Proc. Int. Conf. Comput. Des.*, Oct. 2006, pp. 155–161.

[6] L. Bolzani, A. Calimera, A. Macii, E. Macii, and M. Poncino, "Enabling concurrent clock and power gating in an industrial design flow," in *Proc. Des., Automat. Test Eur. Conf. Exhib.*, Apr. 2009, pp. 334–339.

[7] *NanoSim User Guide*, Synopsys, Inc., Mountain View, CA, Dec. 2007.

[8] Y. Ye, S. Borkar, and V. De, "A new technique for standby leakage reduction in high-performance circuits," in *Proc. Symp. VLSI Circuits*, Jun. 1998, pp. 40–41.

[9] E. Arbel, C. Eisner, and O. Rokhlenko, "Resurrecting infeasible clock-gating functions," in *Proc. Des. Automat. Conf.*, Jul. 2009, pp. 160–165.

[10] Y. Shin, J. Seomun, K. Choi, and T. Sakurai, "Power gating: Circuits, design methodologies, and best practice for standard-cell VLSI designs," *ACM Trans. Des. Automat. Electron. Syst.*, vol. 15, no. 4, pp. 28:1–28:37, Sep. 2010.

[11] S. Mutoh, S. Shigematsu, Y. Gotoh, and S. Konaka, "Design method of MTCMOS power switch for low-voltage high-speed LSIs," in *Proc. Asia South Pacific Des. Automat. Conf.*, Jan. 1999, pp. 113–116.

[12] H.-S. Won, K.-S. Kim, K.-O. Jeong, K.-T. Park, K.-M. Choi, and J.-T. Kong, "An MTCMOS design methodology and its application to mobile computing," in *Proc. Int. Symp. Low Power Electron. Des.*, Aug. 2003, pp. 110–115.

[13] C. Hwang, P. Rong, and M. Pedram, "Sleep transistor distribution in row-based MTCMOS designs," in *Proc. Great Lakes Symp. VLSI*, Mar. 2007, pp. 235–240.

[14] H. Kriplani, F. Najm, and I. Hajj, "Maximum current estimation in CMOS circuits," in *Proc. Des. Automat. Conf.*, Jun. 1992, pp. 2–7.

[15] C. Hsieh, J. Lin, and S. Chang, "Vectorless estimation of maximum instantaneous current for sequential circuits," *IEEE Trans. Comput.-Aided Des.*, vol. 25, no. 11, pp. 2341–2352, Nov. 2006.

[16] C. Hsieh, J. Lin, and S. Chang, "Efficient transition-mode Boolean characteristic function with its application to maximum instantaneous current analysis," in *Proc. Int. Symp. Qual. Electron. Des.*, Mar. 2007, pp. 602–606.

[17] S. Ercolani, M. Favalli, M. Damiani, P. Olivo, and B. Ricco, "Estimate of signal probability in combinational logic networks," in *Proc. Eur. Test Conf.*, Apr. 1989, pp. 132–138.

[18] K. Usami, M. Igarashi, F. Minami, T. Ishikawa, M. Kanzawa, M. Ichida, and K. Nogami, "Automated low-power technique exploiting multiple supply voltages applied to a media processor," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 3, pp. 463–472, Mar. 1998.

[19] *IC Compiler Implementation User Guide*, Synopsys, Inc., Mountain View, CA, Mar. 2010.

[20] OpenCores. (2009) [Online]. Available: http://www.opencores.org

[21] *Design Compiler User Guide*, Synopsys, Inc., Mountain View, CA, Mar. 2007.

[22] OpenAccess. (2009) [Online]. Available: http://www.si2.org

[23] Z. Hu, A. Buyuktosunoglu, V. Srinivasan, V. Zyuban, H. Jacobson, and P. Bose, "Microarchitectural techniques for power gating of execution units," in *Proc. Int. Symp. Low Power Electron. Des.*, Aug. 2004, pp. 32–37.

Jun Seomun received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 2005, 2007, and 2011, respectively.

He is currently with Samsung Electronics, Yongin, Korea. His current research interests include computer-aided design for low-power and reliability-aware design.

Insup Shin received the B.S. and M.S. degrees in electrical engineering from Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 2007 and 2009, respectively, where he is currently pursuing the Ph.D. degree from the Department of Electrical Engineering.

His current research interests include computer-aided design for low-power design, high-level synthesis, and structured application-specific integrated circuits.

Youngsoo Shin (M'00–SM'05) received the B.S., M.S., and Ph.D. degrees in electronics engineering from Seoul National University, Seoul, Korea.

From 2000 to 2001, he was a Research Associate with the University of Tokyo, Tokyo, Japan. From 2001 to 2004, he was a Research Staff Member with the IBM T. J. Watson Research Center, Yorktown Heights, NY. He joined the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 2004, where he is currently an Associate Professor. His current research interests include computer-aided design with emphasis on low-power design and design tools, high-level synthesis, sequential synthesis, and programmable logic.

Dr. Shin received several awards, including the Best Paper Award at the 2005 International Symposium on Quality Electronic Design and the 2002 IP Excellence Award from Japan. He was a member of the Technical Program Committee and Organizing Committee of many technical conferences, including DAC, ICCAD, ISLPED, ASP-DAC, CASES, ISVLSI, and ISCAS. He is a member of the ACM SIGDA Low Power Technical Committee and an Associate Editor of ACM TODAES.