VLSI Design, 2002 Vol. 14 (2), pp. 219–227

Taylor & Francis
Taylor & Francis Group

An Empirical Algorithm for Power Analysis in Deep Submicron Electronic Designs

MAY HUANG[a,*], RAYMOND KWOK[b] and SHU-PARK CHAN[b]

[a]*Virtual Silicon Technology, Inc., 1200 Crossman Ave., Sunnyvale, CA 94089, USA;* [b]*International Technological University, 1650 Warburton Ave., Santa Clara, CA 95050, USA*

(Received 10 September 1999; Revised 4 February 2000)

An empirical algorithm applied to logic level power analysis in deep submicron VLSI designs is introduced in the paper. The method explores a static analysis strategy using unit functions to represent signal transitions. It can be extended to the use of a Register Transfer Level (RTL) power analysis after RTL codes are translated to Boolean equations. A new method for representing state-dependent power models is also introduced in the paper to reduce the complexity of power modeling and to improve the performance of power analysis. The modeling method supports not only the empirical power analysis, but also general simulation-based power analysis methods.

Keywords: Analysis; Power consumption model; Power estimation; CMOS ASIC design; State-dependent power representation; Characterization

INTRODUCTION

Power analysis becomes a very important consideration in deep submicron electronic designs [1]. Along with the rapidly advancing semiconductor technology, the complexity and performance of IC designs are both increased significantly. The increasing integration density and high performance make the power dissipation momentous. Low power design techniques and power analysis are now applied at all phases of a power sensitive design in deep submicron electronics. Power management is mainly affected by the architecture of a design, i.e. 30–50% influence of design improvement at Register Transfer Level (RTL) whereas 10–20% at transistor level [2]. Being able to quickly estimate average power dissipation early in a design cycle can assist designers in improving their design structure promptly to achieve their design goals. The most important considerations of power management are computation efficiency, accuracy, and high-level analysis ability.

Two prevailing technologies are used for logic level power analysis: probabilistic analysis and simulation based analysis [1]. In the probabilistic approach, statistical properties of primary inputs are propagated through the netlist to obtain the switching activity of all nodes [1]. Power consumption is proportional to switching activities. A simulation-based analysis includes pattern-dependent approaches such as exhaustive simulation or applicable pattern simulation, and pattern-independent approaches such as statistical simulation [3,4]. A simulation-pattern-based power analysis is used to calculate the capacitive power at an interconnection, then the internal power of related cells provided by a library is added to obtain the total average power at the net. The summation of the power calculated for all nets is the total average power of a design. If the power calculation is steered by signal-switching events, the power consumption at a net, a block, or a design can be reported as a curve of power over time.

The method proposed in this paper is an exploration of a pattern-dependent static-power analysis. It represents a signal switching event with a unit function, calculates propagated event activities, and estimates the power consumption based on the signal events.

POWER MODELING

The accuracy of logic-level power analysis highly depends on the accuracy of the power models of library cells, and

*Corresponding author. Tel.: +1-408-548-2746. Fax: +1-408-548-2750. E-mail: mayh@virtual-silicon.com

ISSN 1065-514X print/ISSN 1563-5171 online © 2002 Taylor & Francis Ltd
DOI: 10.1080/10655140290010123

those power models are generated with power characterization.

It is well known that the power dissipation in a CMOS circuit is a function of input slope, input state and output load. An accurate power model should be able to represent all of these influences properly. Excellent literature is available for power modeling and analysis methods in CMOS designs [1,3–10]. A prior estimation method sacrifices accuracy for efficiency by considering the capacitive power dissipation of interconnections only [5]. An appealing method introduced in Eisenmann and Kohl's paper [6] decomposes a multi-stage ASIC cell into a combination of several single-stage cells, then computes the capacitive power at each interconnection and adds to it the short-circuit power to obtain the total average power of the stage. It makes the previously invisible internal capacitive power visible and countable thus improving the accuracy. The drawback is that it requires an extra library to represent decomposed cells. Power modeling by Sarin and NcNelly [7] is an extension of the ISM (Input–Slew Model) proposed by Misheloff [8] for timing modeling. It divides input slew rate into fast and slow regions defined by the Critical Input Ramp (CIR) line, and assumes that the power dissipation is independent of the input slew rate in fast mode and increases for increasing input slew rate in slow mode. An STGPE graph by Lin *et al.* [9] for modeling the power consumption of a state transition, and a BDD-based symbolic model by Bogliolo *et al.* [10] for describing the charge and discharge of parasitic capacitances and the flow of short-circuit current, are also interesting approaches.

A power Look-Up-Table (LUT) is the prevailing method used to represent the internal power of a cell. The power model on LUTs is a piecewise linear approximation with indexes of input slope and output load. The number of tables is decided according to the representation of input states. The determination of table index is based on the following consideration. The capacitive power and short-circuit power are influenced by output loads, and the maximum load is determined by timing tolerance, and the minimum load is determined by the minimum input capacitance of cells. The short-circuit power is influenced by input slopes, and the maximum slope is determined by timing tolerance, and the minimum slope excluding zero is determined by manufacturing technology. One or more points can be chosen in between based on error analysis during characterizations.

Input-state Dependent Power Model

A state-dependent power model is desirable in deep submicron electronic designs. There is plenty of research exploring the optimization [12] and the selection of input vectors [13], as well as a proper method to represent state-dependent power models [9]. The challenge is to effectively simplify the modeling complexity while reserving the precision.

Considering each input as one of the four possible states: rising, falling, "0" and "1", and representing these states as a set, the equation is $s = \{r, f, 0, 1\}$. If allowing only one input change at a time, the so called single-event limitation, the permutation of input states, N_p, of a n-input cell is given as

$$N_p = n2^n. \tag{1}$$

If the limitation is removed, the permutation of input states of a n-input cell becomes [14]

$$4^n. \tag{2}$$

Equations (1) and (2) show the complexity of the state-dependent power model of a cell.

General characterization tools provide state-dependent power models with the single-event limitation. With the impact of the input slope and the output load, the power model of a cell becomes considerably complicated. A Simplified Power Equation (SPE) is proposed and described below to reduce complexity of state-dependent power models.

The SPE for Input State

For a n-input cell, the SPE requires acquisitions

$$2n \tag{3}$$

for state-dependent power representation [11]. For input pin i of a cell, two power data acquisitions are

$$P_T\{u(t_i)\}, \quad P_{NT}\{u(t_i)\} \tag{4}$$

or

$$P_{NT}\{u_\alpha(t_D)_{CK=0}\}, \quad P_{NT}\{u_\beta(t_D)_{CK=1}\} \tag{5}$$

for data pin D of a D-Flip–Flop.

Function $u(t_i)$ is a unit function, where $t_i = t - \delta_i$, $\delta_i > 0$ represents a rising event happening at pin i at time δ_i, or $-t_i = -t + \delta_i$, $\delta_i > 0$ represents a falling event. Let ν be a circuit voltage, then $\nu u(t_i)$ represents a signal switching at pin i. Function $P_T\{u(t_i)\}$ presents the power consumption when the switch at input pin i causes the output switching, and it takes the average of rising and falling events, i.e. $P_T\{u(t_i)\} = (1/2)(P\{u(t_i)\} + P\{u(-t_i)\})$. Function $P_{NT}\{u(t_i)\}$ represents the power consumption when the switch at input pin i does not cause the output switching. It takes the average value of power consumption with all those corresponding input states.

A two-input NAND is used as an example to describe how the SPE is applied for a power characterization. Based on Eq. (1), a two-input NAND has eight single-event input states which are: $0r$, $r0$, $1r$, $r1$, $0f$, $f0$, $1f$, $f1$. To characterize $P_T\{u(t_A)\}$, states r_1 and f_1 are counted and averaged. Similarly, $P_{NT}\{u(t_A)\}$ covers states $r0$ and $f0$, $P_T\{u(t_B)\}$ covers $1r$ and $1f$ and $P_{NT}\{u(t_B)\}$ covers $0r$ and $0f$.

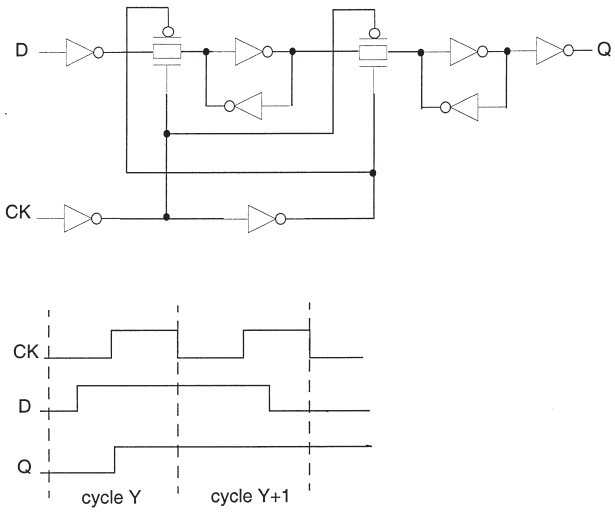

FIGURE 1 Schematic of a D-Flip–Flop.

A general rule is defined below for further reduction:

Rule 1: Combine the states if the difference is tolerable as defined by design technology.

According to Rule 1, $P_{\mathrm{NT}}\{u(t_{\mathrm{A}})\}$ and $P_{\mathrm{NT}}\{u(t_{\mathrm{A}})\}$ in the above example might be combined and averaged to reduce the complexity.

Using a D-Flip–Flop as another example, the set of input states is $S_{\mathrm{DFF}}(D,\mathrm{CK}) = \{0r, r0, 1r, r1, 0f, f0, 1f, f1\}$. Note that a toggle event at pin D does not cause the output switching directly, but the power consumptions are obviously different for a clock staying high or low as shown in Fig. 1. This is the reason that states are considered separately by Eq. (5) SPE power characterization for a DFF has $P_{\mathrm{NT}}\{u_{\alpha}(t_D)_{\mathrm{CK}=0}\}$ covering $r0$ and $f0$, $P_{\mathrm{NT}}\{u_{\beta}(t_D)_{\mathrm{CK}=1}\}$ covering $r1$ and $f1$, $P_{\mathrm{T}}\{u(t_{\mathrm{CK}})\}$ covering $0r$ and $1r$, and $P_{\mathrm{NT}}\{u(t_{\mathrm{CK}})\}$ covering $0f$ and $1f$.

Table I shows the accuracy of the SPE model compared to the complete state-dependent power model. Three cells are used to represent single-stage (NAND), multi-stage (AND) and sequential (DFFRS) cells. Cell DFFRS is a D-Flip–Flop with reset and set. Exhaustive functional test patterns are used, and average power are calculated for the comparison.

Table II illustrates the characterization results based on SPICE stimulation in 0.18 micron, with input slope 0.15 ns and output load 0.5 pf. Table III shows the relative errors by using SPE. Note that a relative error can be as large as 15%, but because the energy value is very small, its influence on the overall power calculation is too small to be significant.

SPE power models can be used to support either a simulation-based power analysis method introduced above, or an equation-based power analysis method proposed in next section.

TABLE I Accuracy analysis of complete power model over SPE

Cell	Cell leakage power (pW)	Cell internal power (nW)	Net switching power (nW)	Total dynamic power, P (nW)	Error % [(Pspe − Pcom)/Pcom]100%
Two-input NAND					
Complete model	92.5820	19.3672	9.2299	28.5971	
SPE model	92.5820	19.4658	9.2299	28.6957	0.345
Two-input AND					
Complete model	95.5700	23.7597	6.6845	30.4443	
SPE model	95.5700	24.2308	6.6845	30.9153	1.547
DFFRS					
Complete model	286.2550	71.5178	13.1972	84.7150	
SPE model	286.2550	74.5362	13.1972	87.7334	3.563

AN EQUATION BASED POWER ANALYSIS METHOD

An empirical algorithm was developed using unit functions and the state-dependent power models generated

TABLE II State-dependent power analysis of an AOI cell in SPICE

Switches	States (AB)	Energy, E (average) (pJ)
$A1$ 01/10 → Z01/10 when $A2 = 1$, $B1 = 0$, $B2 = 0$	r100/f100	0.1200
$A1$ 01/10 → Z01/10 when $A2 = 1$, $B1 = 1$, $B2 = 0$	r110/f110	0.1175
$A1$ 01/10 → Z01/10 when $A2 = 1$, $B1 = 0$, $B2 = 1$	r101/f101	0.1140
$A2$ 01/10 → Z01/10 when $A1 = 1$, $B1 = 0$, $B2 = 0$	1r00/1f00	0.1220
$A2$ 01/10 → Z01/10 when $A1 = 1$, $B1 = 1$, $B2 = 0$	1r10/1f10	0.1200
$A2$ 01/10 → Z01/10 when $A1 = 1$, $B1 = 0$, $B2 = 1$	1r01/1f01	0.1170
$B1$ 01/10 → Z01/10 when $B2 = 1$, $A1 = 0$, $A2 = 0$	00r1/00f1	0.1065
$B1$ 01/10 → Z01/10 when $B2 = 1$, $A1 = 1$, $A2 = 0$	10r1/10f1	0.1085
$B1$ 01/10 → Z01/10 when $B2 = 1$, $A1 = 0$, $A2 = 1$	01r1/01f1	0.1055
$B2$ 01/10 → Z01/10 when $B1 = 1$, $A1 = 0$, $A2 = 0$	001r/001f	0.1090
$B2$ 01/10 → Z01/10 when $B1 = 1$, $A1 = 1$, $A2 = 0$	101r/101f	0.1110
$B2$ 01/10 → Z01/10 when $B1 = 1$, $A1 = 0$, $A2 = 1$	011r/011f	0.1080
$A1$-01 when $Z = 1$, $A2 = 0$, $B1 = 0$, $B2 = 0$	r000	0.015
$A1$-01/10 when $Z = 1$, $A2 = 0$, $B1 = 1$, $B2 = 0$	r010/f010	0.015
$A1$-01/10 when $Z = 1$, $A2 = 0$, $B1 = 0$, $B2 = 1$	r001/f001	0.015
$A1$-01 when $Z = 0$, $A2 = 0$, $B1 = 1$, $B2 = 1$	r011	0.014
$A1$-10 when $Z = 0$, $A2 = 1$, $B1 = 1$, $B2 = 1$	f111	0.013
$A2$-01/10 when $Z = 1$, $A1 = 0$, $B1 = 0$, $B2 = 0$	0r00/0f00	0.015
$A2$-01 when $Z = 0$, $A1 = 1$, $B1 = 1$, $B2 = 1$	1r11	0.012
$A2$-10 when $Z = 1$, $A1 = 0$, $B1 = 1$, $B2 = 0$	0f10	0.014
$A2$-10 when $Z = 1$, $A1 = 0$, $B1 = 0$, $B2 = 1$	0f01	0.014
$A2$-01 when $Z = 0$, $A1 = 0$, $B1 = 1$, $B2 = 1$	0f11	0.014
$B1$-01/10 when $Z = 1$, $A1 = 1$, $A2 = 0$, $B2 = 0$	10r0/10f0	0.015
$B1$-01/10 when $Z = 1$, $A1 = 0$, $A2 = 1$, $B2 = 0$	01r0/01f0	0.015
$B2$-01/10 when $Z = 1$, $A1 = 1$, $A2 = 0$, $B1 = 0$	100r/100f	0.014
$B2$-10 when $Z = 1$, $A1 = 0$, $A2 = 1$, $B1 = 0$	010f	0.014

by the SPE. The fundamental of the power analysis is the power dissipated on a cell for each set of input vectors. If a set of vectors is applied based on a clock cycle, power analysis becomes a cycle based method.

Representing an input event as a unit function, a series of events at a pin can be represented with an equation of unit functions. In order to process a combinatorial circuit, empirical formulas upon a pie method are developed to appraise the dependency of the output to the input states. The formula with respect to AND, OR, and XOR logic are shown below.

A Logical AND, $f = u_1 \cdot u_2$

The possible states can be illustrated by a pie shown in Fig. 2, and the empirical formulas are given below.

(1). Falling before rising

$$f = u_1(-t + t_1).u_2(t - t_2) = 0, \quad t_2 > t_1, \quad t_1, t_2 > 0. \tag{6.1}$$

(2). Rising before falling

$$f = u_1(t - t_1).u_2(-t + t_2) = \begin{cases} 0, & t < t_1 \\ 1, & t_1 \le t < t_2, \quad t_2 > t_1, \quad t_1, t_2 > 0 \\ 0, & t \ge t_2 \end{cases} \tag{6.2}$$

(3). Rising and falling simultaneously

$$f = u_1(t - t_1).u_2(-t + t_1) = 0, \quad t_1 > 0. \tag{6.3}$$

(4). Both rising or both falling simultaneously

$$f = u_1(t - t_1).u_2(t - t_1) = u(t - t_1), \quad t_1 > 0,$$

$$\text{or } f = u_1(-t + t_1).u_2(-t + t_1) = u(-t + t_1), \quad t_1 > 0,$$

TABLE III Error analysis after reduction of an AOI cell

Model	States ($A1A2B1B2$)	Energy, E' (average) (pJ)	Error = $[(E' - E)/E]100\%$
$E_T\{u(t_{A1})\}$	r100/f100/r110/f110/r101/f101	0.1172	−2.33, −0.25, 2.8
$E_T\{u(t_{A2})\}$	1r00/1f00/1r10/1f10/1r01/1f01	0.1197	−1.89, −0.25, 2.31
$E_T\{u(t_{B1})\}$	00r1/00f1/10r1/10f1/01r1/01f1	0.1068	0.28, −1.57, 1.23
$E_T\{u(t_{B2})\}$	001r/001f/101r/101f/011r/011f	0.1093	0.28, −1.53, 1.20
$E_{NT}\{u(t_{A1})\}$	r000/r010/f010/r001/f001/r011/f111	0.0144	−4, 2.86, 10.77
$E_{NT}\{u(t_{A2})\}$	0r00/0f00/1r11/0f10/0f01/0f11	0.0138	−8, 15, −1.43
$E_{NT}\{u(t_{B1})\}$	10r0/10f0/01r0/01f0	0.015	0
$E_{NT}\{u(t_{B2})\}$	100r/100f/010f	0.014	0

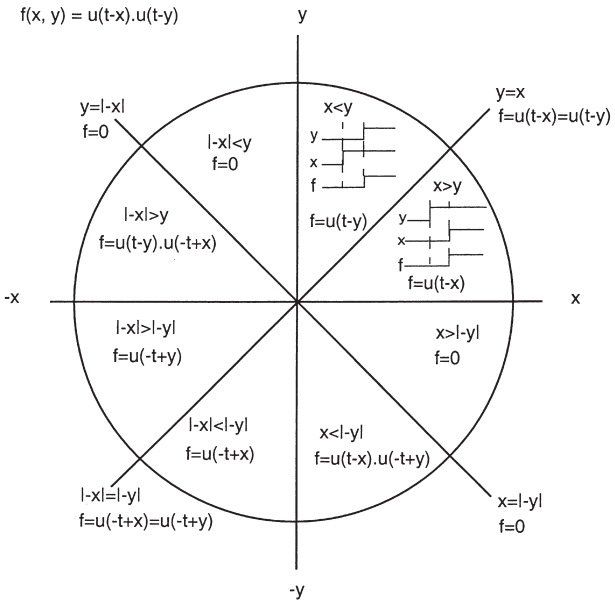

FIGURE 2 A state pie of a two-input AND.

(5). Rising in succession

$$f = u_1(t - t_1).u_2(t - t_2) = u(t - t_2), \quad t_2 > t_1, \quad t_1, t_2 > 0. \tag{6.5}$$

(6). Falling in succession

$$f = u_1(-t + t_1).u_2(-t + t_2) = u(-t + t_1), \quad t_2 > t_1, \quad t_1, t_2 > 0. \tag{6.6}$$

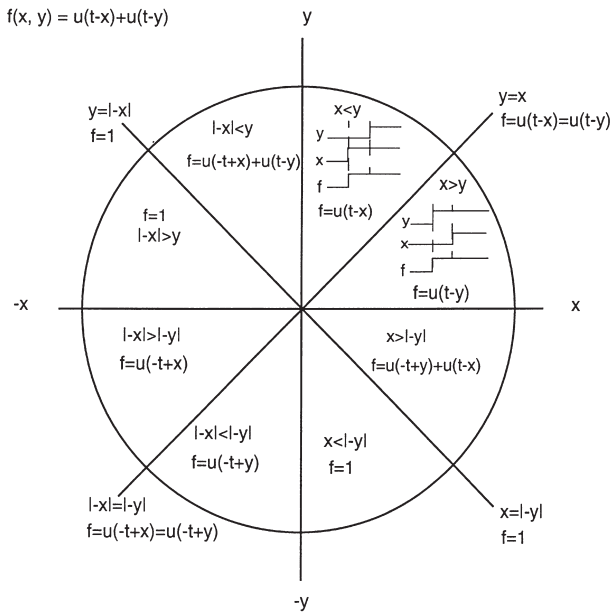

FIGURE 3 A state pie of a two-input OR.

A Logical OR, $f = u_1 + u_2$

The state pie is given in Fig. 3, and the empirical formulas are:

(1). Falling before rising

$$f = u_1(-t + t_1) + u_2(t - t_2) = \begin{cases} 1, & t < t_1 \\ 0, & t_1 \le t < t_2, \quad t_2 > t_1, \quad t_1, t_2 > 0 \\ 1, & t \ge t_2 \end{cases} \tag{7.1}$$

(2). Rising before falling

$$f = u_1(t - t_1) + u_2(-t + t_2) = 1, \quad t_2 > t_1, \quad t_1, t_2 > 0. \tag{7.2}$$

(3). Rising and falling simultaneously

$$f = u_1(t - t_1) + u_2(-t + t_1) = 1, \quad t_1 > 0. \tag{7.3}$$

(4). Both rising or both falling simultaneously

$$f = u_1(t - t_1) + u_2(t - t_1) = u(t - t_1) \quad t_1 > 0,$$

or $f = u_1(-t + t_1) + u_2(-t + t_1) = u(-t + t_1), \ t_1 > 0.$

(5). Rising in the succession

$$f = u_1(t - t_1) + u_2(t - t_2) = u(t - t_1), \quad t_2 > t_1, \quad t_1, t_2 > 0. \tag{7.5}$$

(6). Falling in the succession

$$f = u_1(-t + t_1) + u_2(-t + t_2) = u(-t + t_2), \quad t_2 > t_1, \quad t_1, t_2 > 0.$$

A Logical XOR, $f = u_1 \oplus u_2$

The state pie is given in Fig. 4, and the empirical formulas are:

(1). Both rising or both falling

$$f = u_1[k(t - t_1)] \oplus u_2[k(t - t_2)] = \begin{cases} 0, & t < t_1 \\ 1, & t_1 \le t < t_2, \quad t_1, t_2 > 0 \\ 0, & t \ge t_2 \end{cases} \tag{8.1}$$

where $k = 1$ if both input signals are rising, or $k = -1$ if both failing.

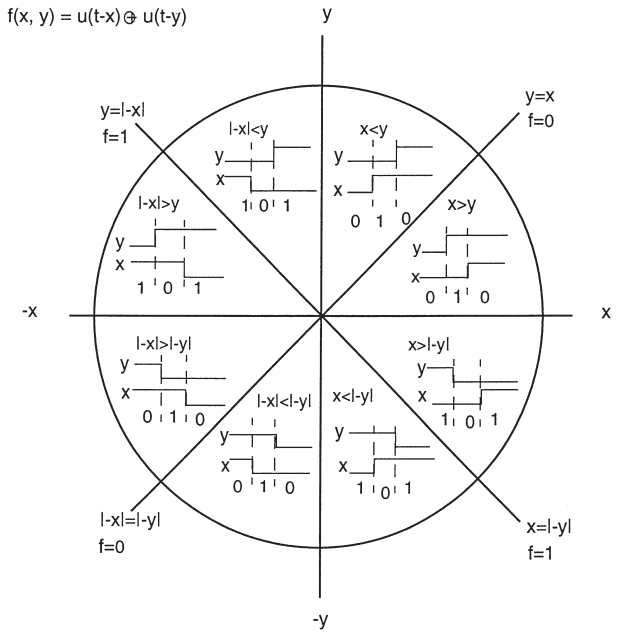

FIGURE 4 A state pie of a two-input XOR.

(2). One rising and one failing

$$f = u_1(t - t_1) \oplus u_2(-t + t_2)$$

$$= \begin{cases} 1, & t < t_1 \\ 0, & t_1 \le t < t_2, \quad t_1, t_2 > 0 \\ 1, & t \ge t_2 \end{cases} \tag{8.2}$$

(3). Rising and falling simultaneously

$$f = u_1(t - t_1) + u_2(-t + t_1) = 1, \quad t_1 > 0. \tag{8.3}$$

(4). Both rising or both falling simultaneously

$$f = u_1(t - t_1) + u_2(t - t_1) = 0. \tag{8.4}$$

The above empirical formulas represent two-input logical AND, OR and XOR operations. To process a cell with more than two inputs, the first two signals are considered in the order that the events occurred, then the result is used with the third signal and so on till all the inputs have been processed.

Applying the formulas with the logical function of a cell, the output events can be derived. The cells shown in Figs. 5 and 6 are two examples to illustrate the applications.

The input events shown in Fig. 5 establish the case of rising events in succession, so that the output will act as $f = u_B(t - t_B).u_A(t - t_A) = u(t - t_A)$, where $t_A > t_B$. The power dissipated in this cycle will be $P = P_T\{u(t_A)\} + P_{NT}\{u(t_B)\}$. An AOI cell in Fig. 6 has two sets of input events occurred in cycle Y and cycle $Y + 1$ as shown in Fig. 7. Considering the events at $A1$ and $A2$ during cycle Y, they represent the case of rising before falling, so that function $A1{\cdot}A2$ has two switches corresponding to $u_{A1}(t - t_1)$ and $u_{A2}(-t + t_3)$ according to Eq. (6.2). The events occurred at $B1$ and $B2$ during cycle Y illustrate the case of falling before rising, and the result of $B1{\cdot}B2$ is 0 according to Eq. (6.1). The output $z = \overline{A1A2 + B1B2}$ is thus affected by the result of $A1{\cdot}A2$, and its two switching are caused by the events which happened at pin $A1$ and pin $A2$. Items $A1{\cdot}A2$ and $B1{\cdot}B2$ are processed prior to the OR function so that the overlap of input signals can be managed as shown in cycle $Y + 1$ in Fig. 7.

The power consumptions during the two cycles are calculated as:

$$P_Y = P_T\{u(t_{A1})\} + P_T\{u(t_{A2})\} + P_{NT}\{u(t_{B1})\} + P_{NT}\{u(t_{B2})\}$$

$$P_{Y+1} = P_{NT}\{u(t_{A1})\} + P_T\{u(t_{A2})\} + P_T\{u(t_{B1})\} + P_{NT}\{u(t_{B2})\}$$

The input events are counted sequentially in a clock cycle, and power consumptions are accumulated according to the influences of the input events.

For a sequential cell, the empirical formulas are developed below.

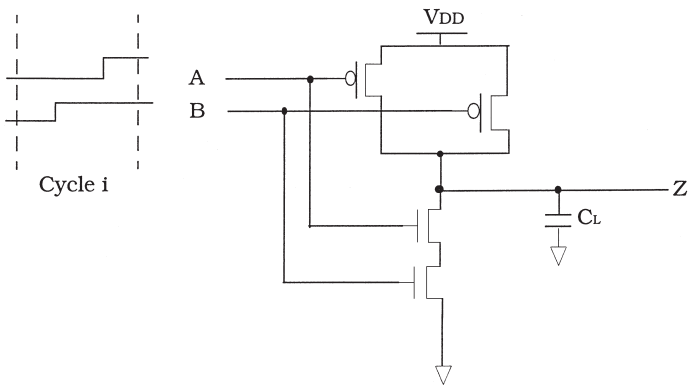

FIGURE 5 A two-input NAND.

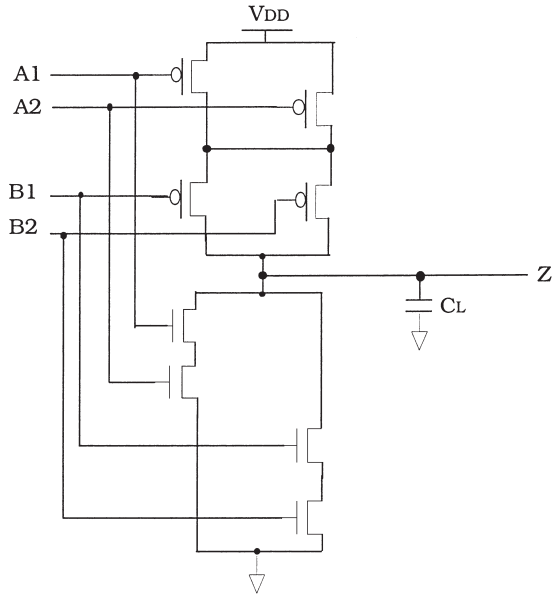

FIGURE 6 A four-input AND–OR–INVERTER (AOI).

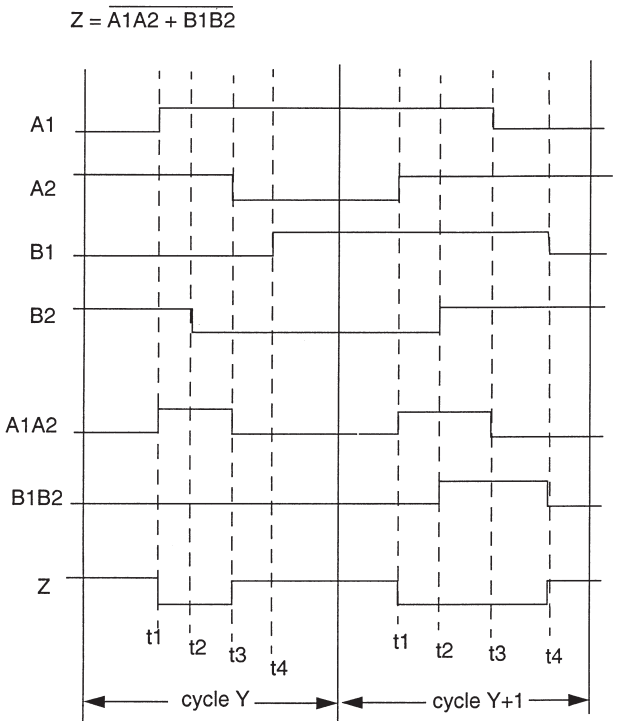

FIGURE 7 A waveform of a AOI.

AD-Flip–Flop With Inputs CK (clock) and D (data)

(1). Assuming only one even occurred at data pin D in a clock cycle:

Output

$$Q = u_{\mathrm{CK}}(k(t - t_{\mathrm{CK}})), \quad t_{\mathrm{CK}} > t_D, \quad t_{\mathrm{CK}}, t_D > 0, \quad (9.1)$$

where $k = 1$ if D is rising, or $k = -1$ if D is falling.

(2). Assuming m events occurred at data pin D before clock rising:

(a). if $m = 2n$, $n = 1, 2, 3, \ldots$, then

$$Q = Q_{-}, \quad t_{\mathrm{CK}} > t_{D_m}, \quad t_{\mathrm{CK}}, t_{D_m} > 0, \quad (9.2)$$

where $Q\_$ represents the output value in the previous clock cycle.

(b). if $m = 2n + 1$, $n = 1, 2, 3, \ldots$, then

$$Q = u_{\mathrm{CK}}(k_m(t - t_{\mathrm{CK}})), \quad t_{\mathrm{CK}} > t_{D_m}, \quad t_{\mathrm{CK}}, t_{D_m} > 0, \quad (9.3)$$

where $k_m = 1$ if the last event at D is rising, or $k_m = -1$ if the last event at D is falling.

For instance, considering the events which happened at two clock cycles Y and $Y + 1$ of a DFF shown in Fig. 1, the corresponding power consumption is calculated as:

$$P = P_{\mathrm{NT}}\{u_\alpha(t_D)\} + P_{\mathrm{T}}\{u(t_{CK})\} + P_{\mathrm{NT}}\{u(t_{\mathrm{CK}})\} + P_{\mathrm{NT}}\{u_\beta(t_D)\}.$$

A latch with inputs EN (enable, active high) and *D* (data)

(1). EN = 1

$$Q = u_D[k(t - t_D)], \quad t_D > 0, \quad (10.1)$$

where $k = 1$ if D is rising, or $k = -1$ if D is falling.

(2). EN = 0

$$Q = Q_{,} \quad (10.2)$$

which implies no changes at Q, or no equations between Q and D.

(3).

$$Q = u_{\mathrm{EN}}[D(t - t_{\mathrm{EN}})], \quad t_{\mathrm{EN}} > 0, \quad (10.3)$$

The above empirical formulas are used to predict the output activities corresponding to the input events of cells, so that the signal switching activities applied at primary inputs can propagate through a gate-level netlist. Note that the power analysis method is an equation-based method, and no delay effect is modeled. The glitch power caused by signal delays is excluded.

Applying a set of input vectors in a clock cycle, a cycle-based power calculation method is implemented as follows.

For each input event of a cell, the consumed energy of the cell is counted by associating the value preserved in the power model of the cell in a library. For a cell with i inputs, the power consumed in a clock cycle can be calculated as:

$$P = \sum_i \left((E_{\mathrm{T}}\{u(t_i)\}n1 + E_{\mathrm{NT}}\{u(t_i)\}n2). \frac{1}{\tau} \right)$$

where $u(t)_i$ represents an even that occurs at pin i, τ is the

TABLE IV Deviation of power analysis at gate-level from SPICE results

Cell names	Size (gates) (k)	Power in spice Psp (uW)	Power in gate level Pgate (uW)	Error% (Pgate-Psp)/Psp
12-bit Counter	0.35	1264.989	1247.65	−1.37
Alarm-Clock	2.3	2097.777	2058.31	−1.88
8-bit ALU	1.24	1683.6006	1627.66	−3.32

time of the clock cycle, $n1$ or $n2$ is the number of switching activities which cause or not cause the output change. The definitions of $E_T\{u(t_i)\}$ and $E_{NT}\{u(t_i)\}$ are similar to $P_T\{u(t_i)\}$ and $P_{NT}\{u(t_i)\}$ except that they represent energy consumption.

For m cells, the power consumed in a clock cycle can be calculated as:

$$P = \sum_m \left(\sum_i \left((Em_T\{u(t_i)\}n1 + Em_{NT}\{u(t_i)\}n2).\frac{1}{\tau} \right) \right)$$

The power of m cells consumed in k clock cycles can be calculated as:

$$P = \sum_k \left(\sum_m \left(\sum_i \left((Emk_T\{u(t_i)\}n1 + Emk_{NT}\{u(t_i)\}n2).\frac{1}{\tau} \right) \right) \right)$$

It is an average power consumption of a gate-level netlist assuming that the lumped capacitance at a net is known within the given netlist. A lumped capacitance consists of the output capacitance of a driving node, the input capacitance of driven nodes and wire capacitance in the interconnection. Power Look-Up Tables have been prepared by power characterization with respect to input slope and output load, and attached to associated pins. For instance, when an even happened at input A causes a switching activity at the output, the corresponding power consumption is $P_T\{u(t_A)\}$, or $E_T\{u(t_A)\}1/\tau$. The summation of power consumed on a cell in a clock cycle, or m cells in a clock cycle, or m-cell in k-clock cycle is calculated in turn to obtain the total average power consumption of the design with respect to applied test vectors.

Table IV shows the application of the method with SPE power models. Three gate-level circuits are tested. One is a 12-bit synchronous counter with 122 ASIC cells in 30 different types. The second one is an alarm-clock circuit with a scan chain, containing 507 ASIC cells in 65 types. The third one is an 8-by-8 Arithmetic-Logic Unit (ALU) with 411 ASIC cells in 37 types. The result is an average power dissipation of each test circuit calculated by the empirical power analysis prototype. The deviation from SPICE calculation is also listed in the table as a comparison for accuracy. The results in the table establish that the proposed empirical algorithm with SPE models is efficient and accurate for gate-level power analysis.

A defect of this method is the deficiency of a unified formula for various types of cells. However, the pie method is used beneficially for developing empirical formulas of combinatorial cells.

An attractive property of the method is that it can be extended to the used of RTL power analysis with quick synthesis techniques. Generally, an RTL analysis is implemented using either quick synthesis or an RTL library approach [15,16]. The quick-synthesis method translates an RTL description into Boolean equations, and then maps them into a gate netlist without or with minor optimization. The RTL-library method interprets the RTL codes into Boolean equation, then maps them into functional blocks, so called macro cells, provided in an RTL library. Those macro cells in an RTL library have prior characterization with timing and power models as those in a standard cell library. The empirical power-analysis method introduced in the paper can directly be extended for use in RTL power analysis with quick synthesis approach. To extend it to a library based RTL analysis, the unit equation needs be derived for each macro cell according to its logic function.

CONCLUSIONS

An empirical algorithm used for static power analysis is proposed in the paper. It uses equations of unit functions to represent and predict event activities in a given circuit to achieve a static analysis. Recent trends leading to the replacement of gate-level simulations by formal verification and static timing analysis makes this static power analysis approach more attractive. The method also provides the possibility of extending its use to an RTL power analysis method, a new power modeling method, SPE, is introduced. The SPE can be used for either prevalent simulation-based power analysis or this equation-based power analysis method, to reduce the complexity of power modeling dependencies with the preservation of accuracy.

Acknowledgements

The authors would like to acknowledge Dr Michael Kohl of Synopsis, Inc., and Dr Raj Kumar of Hewlett Packard, for their valuable technical assistance. The authors are grateful to Mr Michael Grossman of Virtual Silicon Technology, Inc., Dr Robert Payne and Dr Paul Wiscombe of VLSI Technology, Inc., and Mr Jack Thomas of Hitachi Semiconductor America, for their continuous support of this project.

References

[1] Nobel, W. and Mermet, J. (1997) Low Power Design in Deep Submicron Electronics (Kluwer Academic Publishers, Dordrecht).
[2] DTG Low Power Training, Version 1.0, Synopsys, Inc., 1998.
[3] Najm, F.N. (1994) "A survey of power estimation techniques in VLSI circuits", *IEEE Trans. VLSI Syst.* **2**(4), 446–454.
[4] Burch, R., Najm, F.N., Yang, P. and Trick, T.N. (1995) "A Monte Carlo approach for power estimation", *IEEE Trans. VLSI Syst.* **1**(1), 405–415.
[5] Liu, D. and Svensson, C. (1994) "Power consumption estimation in CMOS VLSI chip", *IEEE Trans. Solid-State Circuits* **29**(6), 663–670.
[6] Eisenmann, W. and Kohl, M. (1992). "Power calculation for CMOS gate array", *Proceedings of IEEE ASIC Conference, Rochester, NY*.
[7] Sarin, H.K. and McNelly, A. (1995). "A power modeling and characterization method for logic simulation", *Proc. of IEEE custom Integrated Circuit Conference*, pp. 363–366.
[8] Misheloff, M.N. (1992). "Improved modeling and characterization system for logic simulation", *Prof. of IEEE ASIC Intl. Conference*.
[9] Lin, J.Y., Shen, W.Z. and Jou, J.Y. (1996). "A power modeling and characterization method for the CMOS standard cell library", *Proc. of IEEE Int'l Conf. On Computer Aided Design*, pp. 400–404.
[10] Bogliolo, A., Benini, L. and Ricco, B. (1996). "Power estimation of cell-based CMOS circuit", *Proc. of 33rd. Design Automation Conference*.
[11] Huang, M., Power analysis in CMOS ASIC designs PhD Thesis, Department of Electrical Engineering, International Technological University).
[12] Marculescu, R., Marculescu, D. and Pedram, M. (1997). "Hierarchical sequence compaction for power estimation", *Proc. of 34rd Design Automation Conference*, pp. 570–575.
[13] Schneider, P. and Krishnamoorthy, S. (1996). "Effects of correlations on accuracy of power analysis—an experimental study", *IEEE International Symposium on Low Power Electronics and Design*, Monterey.
[14] Cameron, P.J. (1994) Combinatorics: Topics, Techniques, Algorithms (Press Syndicate of University of Cambridge), p 32.
[15] Macii, E., Pedram, M. and Somenzi, F. (1997). "High-level power modeling, estimation, and optimization", *Proc. of 34rd. Design Automation Conference*, pp. 504–511.
[16] Gupta, S. and Najm, F.N. (1997). "Power macromodeling for high level power estimation", *Proc. of 34rd. Design Automation Conference*, pp. 365–370.

Authors' Biographies

May Huang was a senior and a staff design engineer of VLSI Technology Inc. and Hitachi Semiconductor America, in charge of projects including the methodology of power modelling and analysis, RTL sign-off, static timing analysis, power management for a microprocessor design with low power and high frequency in deep submicron VLSI, etc. She participated as a balloter on VITAL, Verilog, and Analog Extensions of VHDL to IEEE standard. Dr Huang is currently a principal methodology engineer of Virtual Silicon Technology, Inc.

Raymond S. Kwok was a guest scientist at the Los Alamos National Laboratory, a designer and a scientist in several other companies. Dr Kwok is currently the Vice Chair of the Electrical Engineering Department at International Technological University. His current research interests are device modelling, applied electromagnetism, and communication systems and devices. Dr Kwok is a Senior Member of the IEEE.

Shu-Park Chan has devoted over 40 years to education doing teaching and research among four universities and colleges: VMI, UI, Santa Clara University, serving as Chair of EECS Dept., Dean of Engineering school and endowed Chair Professor, and International Technological University (ITU) which he founded in 1994. Prof. Chan is a Fellow of IEEE. He was appointed by President George Bush to serve on the J. William Fulbright Foreign Scholarship Board for the period of 1991–1993.