Accurate logic-level power estimation

A. Bogliolo   B. Riccó
DEIS, Universitá di Bologna
Bologna, Italy

L. Benini   G. De Micheli
CIS, Stanford University
Stanford CA, USA

1 Introduction

In recent years, power dissipation has become a primary design constraint for complex VLSI systems. Designers need computer-aided analysis tools that accurately and rapidly estimate power dissipation.

At a level of abstraction higher than electrical simulation, logic-level simulation allows power estimation for very large blocks, often enabling full-chip simulation. As a consequence, for CMOS digital circuits logic-level simulation is often the preferred tool for validation and debugging. Mainly for these reasons, many attempts have been made to provide power estimation at the logic level. In the most simplified model, power is estimated observing the *switching activity* (toggle count) at the output of the basic logic blocks of the circuit (weighted with loading capacitance).

Power estimation based on switching activity has however limited accuracy, mainly because it does not consider phenomena such as signal transition times, spurious transitions (glitches), short circuit currents and gate internal capacitances, that may have a sizable impact on the total power dissipation. In order to overcome these difficulties, advanced logic simulation techniques have been proposed [1, 2]. These approaches are based on lookup-tables obtained with electrical simulation of the basic building blocks and have reported promising results, but they have two major limitations. First, they do not assume any model for the internal structure of the basic building blocks (*gates*). Second, they do not deal with multiple input transitions that are not perfectly aligned in time.

In this work we propose a more accurate model that overcomes the limitations above mentioned, while keeping computational efficiency competitive with traditional gate-level power estimation based on transition activity. Our technique exploits a BDD-based symbolic model for describing the charge and discharge of parasitic (and load) capacitance and the flow of short circuit current. Lookup tables are used only for modeling the timing behavior of the circuit (as it is commonly done in full-delay simulation), therefore power simulation only marginally increases memory usage. Our method is highly accurate also for single gate (*local*) power estimate, allowing the individuation of critical gates during design optimization.

We have implemented our techniques using VERILOG-XL as simulation platform, therefore maintaining full compatibility with design environments based on Verilog HDL. For our test library the accuracy on *single gate* power estimation is within 5% from SPICE under a wide range of fan-in and fan-out conditions. The accuracy on the average power dissipation for large benchmarks is similar. The performance degradation with respect to standard-delay-model Verilog simulation with toggle count is within a factor of 20. The performance loss is expected to decrease in future engineered versions of our tool.

2 Simulation technique

Gate-level power estimation based on output toggle count does not take into account charge sharing and the switching of gate internal capacitances. To overcome this limitation we first provide for each gate in the library the layout-extracted internal capacitances. We model the parasitics with constant capacitors to ground. The computation of the energy dissipated in load and internal gate capacitances is based on the observation that the increase in the total charge of capacitors with connection to the power supply (V_{dd}) is due to charge provided by the power supply itself. The total energy provided by V_{dd} during a time period $\Delta t = t_f - t_i$ is $E = V_{dd} \int_{t_i}^{t_f} i\, dt$. Since $i = dq/dt$, we have $E = V_{dd}\, \Delta q$, where Δq is the charge provided by V_{dd} during the transition, assuming that there is no transient DC path to ground. The energy provided by the power supply to charge circuit capacitances can therefore be computed with the following equation:

$$E_{ch} = V_{dd} \sum_{i \in C} \Delta q_i \qquad (1)$$

where C is a set of connected nodes with a connection to V_{dd}. For each gate, the computation of E_{ch} requires the knowledge of the charge status of the nodes at the beginning and at the end of the transition. Moreover, we need to dynamically determine which nodes belong to C. We can solve these problems if we keep track of the connection of each node to ground (V_{ss}), to V_{dd} and to each other node in the gate.

We efficiently compute these Boolean conditions using a set of BDDs called *connection matrix*. For each node in the gate a set of BDDs represents the input conditions enabling the connections with other nodes and the reference nodes (V_{dd} and V_{ss}). The input variables of the BDDs are the primary inputs of the gate. For each input pattern the evaluation of the connection status of each node is obtained with a simple BDD evaluation. The connection matrix is triangular and the BDD representation allows a consistent amount of sharing among the Boolean coefficient of the connection matrix. Clearly, our method takes into account charge sharing between nodes, because the charge status of each node is computed exploiting the complete knowledge of the connection matrix. When evaluating the internal charge status, the body effect is automatically taken into account by using suitable threshold values. The only source of error in our estimation is then the constant capacitor to ground model (floating capacitors are modeled as capacitors to ground).

In CMOS circuits, additional power dissipation can be caused by short circuit currents between V_{dd} and V_{ss}. Disregarding short circuit power (energy) dissipation may lead

to sizable errors for gates with highly loaded inputs and lightly loaded outputs [3]. The connection matrix can be used to detect conditions for which there is a transient open path between V_{dd} and V_{ss}. On every input pattern transition, we check if any node that was connected to V_{dd} in the old input configuration is connected to V_{ss} in the new one or viceversa. If this condition is verified, short circuit energy (E_{cc}) has to be taken into account. We estimate E_{cc} with the following formula:

$$E_{cc} = f_{cc}(k_1 S_1 + ... + k_n S_n + k_{n+1}(1/L)) \qquad (2)$$

where f_{cc} is a Boolean flag that is one only if there has been a transient connection between V_{dd} and V_{ss}, $S_1, ..., S_n$ are the input slopes (set to 0 if the corresponding input does not change) and L is the output load. The coefficients $k_1, ..., k_{n+1}$ are computed with min-square fitting of the E_{cc} values obtained by circuit simulation. Notice that we use table-based estimation only for E_{cc} with a number of parameters that is equal to $n + 1$ where n is the number of gate inputs.

The total energy provided by V_{dd} during an input transition is $E = E_{ch} + E_{cc}$. No precision is lost for aligned multiple input patterns. The overhead with respect to a simple model based on output transitions is represented by the connection matrix and the more involved formulas for the computation of E_{cc} and E_{ch}.

Although the model described above is accurate for perfectly aligned multiple input transitions, this is a situation that is not often encountered in practice. In the majority of cases, multiple input transitions are slightly misaligned, possibly by short times (compared to the propagation delay of the gate). In this case a model that computes the power dissipation observing input transitions may produce large errors, because it will consider a slightly misaligned multiple transition as a sequence of complete transitions.

Assume that a multiple input transition from input pattern a to c is not perfectly aligned. The misalignment causes the intermediate pattern b to appear at the input of the gate for a short period of time. Assume that $\delta_{b,c}$ is the delay between patterns b and c. We call Δ_b the time needed to reach 90% of the total charge transfer from V_{dd} to capacitances in the gate (caused by the transition $a \rightarrow b$). If $\delta_{b,c} \rightarrow 0$, pattern b disappears and the energy dissipated is $E_{a,c}$. On the other side, if $\delta_{b,c} >> \Delta_b$, we have two complete transitions and the total energy dissipation is $E_{a,b} + E_{b,c}$.

We approximate the intermediate cases using a linear interpolation between the two limit cases, namely:

$$E = (E_{a,b} + E_{b,c})\delta_{b,c}/\Delta_b + E_{a,c}(1 - \delta_{b,c}/\Delta_b) \qquad (3)$$

clearly this formula holds if $\delta_{b,c} < \Delta_b$. If this is not true we have $E = E_{a,b} + E_{b,c}$. The linear approximation is exact at the boundaries, but its accuracy strongly depends on the definition of Δ_b. In general, Δ_b is not equal to the delay used for event propagation. We determine it by means of electrical simulation during the library characterization phase, with methods similar to those used for timing characterization: for each input pattern a set of min-square fitting parameters is obtained. These parameters are used in conjunction with back-annotated fan-in and fan-out information for the computation of Δ_b and pattern dependent delays at simulation time.

Circuit		SPICE		PPP		err
name	gates	Power	CPU	Power	CPU	%
cmb	50	1.447	182	1.407	4	2.7
parity	75	2.411	255	2.346	5	2.7
pcle	64	2.150	226	2.128	4	1.0
pcler8	80	2.921	294	2.876	5	1.5
f51m	142	6.060	1463	5.849	21	3.5
comp	163	6.382	762	6.137	15	3.8
b9	121	4.363	533	4.174	10	4.3
frg1	124	4.888	588	4.735	11	3.1
x1	345	13.110	2028	12.518	32	4.5
alu2	359	16.710	2880	16.754	36	0.3

Table 1: Average power consumption and simulation time of benchmark circuits simulated by Spice and by PPP.

3 Experimental Results

We have implemented our simulator (PPP) using a VERILOG-XL platform, and tested it using a low-power CMOS [4] library (including complex gates and two-level cells). Each library cell has been characterized as described in Section 2, and then simulated for all possible test-pairs and for a wide range of fan-in and fan-out conditions to verify the accuracy of single-gate single-pattern power estimations. In the worst case (a three input NAND-gate) the average error from SPICE has been of 4% of the average estimated power, with standard deviation of 0.2%.

A second set of experiments has been performed by applying to each cell a sequence of randomly generated test vectors with the 50% of overlapping input transitions. In this case, the accuracy of the average power estimation has always been within 2% from SPICE, due to the effects of error compensations.

At last, we have simulated a set of circuits obtained by mapping several combinational benchmarks on our test library. Circuits have been simulated using randomly generated sequences of 100 input patterns with a clock period of 20ns. The experimental results obtained are reported in Table (1), where average power consumptions are expressed in mW, and CPU times in seconds. The worst case error with respect to SPICE is of 4.5%, while the performance is two orders of magnitude faster. The performance loss with respect to Verilog simulation with toggle count and simplified delay models is within a factor of 20.

4 Acknowledgements

References

[1] J.-Y. Lin et al., "A cell-based power estimation in CMOS combinational circuits," Proc. of the Int'l Conference on Computer-Aided Design, pp. 304–309, 1994.

[2] B. J. George et al., "Power analysis and characterization for semi-custom desing," in Proc. of the Int'l Workshop on Low-Power Design, pp. 215–218, 1994.

[3] H. Veendrick, "Short-Circuit dissipation of static CMOS circuits and its impact on the design of buffer circuits," IEEE Journal of Solid-State Circuits, vol. SC-19, no. 4, pp. 468–473, 1984.

[4] T. Burd, "Low-power CMOS library design methodology," M. S. Report UC Berkeley, UCB/ERLM94/89.