Dynamic Power Estimation Using the Probabilistic Contribution Measure(PCM)

Hoon-Sang Jin, Myung-Soo Jang, Jin-Suk Song, Jin-Yong Lee, Taek-Soo Kim, and Jeong-Taek Kong
CAE, Semiconductor R&D Center, Samsung Electronics Co., Ltd.
San #24 Nongseo-Ri, Kiheung-Eup Yongin-City, Kyungki-Do, 449-900 Korea.
Email: {hsjin,mister,arcadian,mustbe,tkim0,jkong}@samsung.co.kr

Abstract

In this paper, we present CubicPower which is a dynamic power estimator based on Verilog/VHDL simulators. We propose the power characterization model and the probablistic contribution measure(PCM) algorithm to calculate the actual power consumption of cell instances with given switching information. In addition to PCM, the state dependency and non-switching activity of gates are taken into account for more accurate power estimation. Experimental results of CubicPower show less than 10% error compared with the results of PowerMill simulation and the measured values of the IMS test equipment. Due to the PCM algorithm, CubicPower is more accurate than the leading commercial dynamic power estimator at the gate level and is 2-3 orders of magnitude faster than PowerMill.

1 Introduction

Most of the researches about power estimation are focused on the estimation of switching information[1][2]. However, how to calculate the power consumption of each cell instance with pre-characterized power consumption is still questionable. The accuracy of the estimation result is largely dependent on the power characterization methodology and the algorithm to calculate the power consumption of instances.

To calculate the power consumption of cell instances, the power characterization values of each component is fed into the dynamic power estimator together with the switching information. However, conventional approaches of dynamic power estimation lack the accuracy due to the followings.

- **Inaccuracy in the power characterization of multiple output components :** If a component has m outputs $(F_1, F_2, \cdots, F_{m-1}, F_m)$ as shown in Fig. 1(a), the power characterization table for each input transition should be $m+1$ dimensions which have m axes for the output loads $(C_1, C_2, \cdots, C_{m-1}, C_m)$ and 1 axis for the input slope, since the power consumption during the signal transition of an output is dependent on the load of the rest of the output pins as well as the input slope. However, generating the $m+1$ dimensional table for each primitive is impractical (if not impossible). As a result, when multiple output primitives are used in a circuit which is true for most cases in reality, there are certain amount of source of errors inherent in the power characterization model. Another problem occurs when there are multiple output transitions caused by a single input transition.

ISLPED99, San Diego, CA, USA
©1999 ACM 1-58113-133-X/99/0008..$5.00

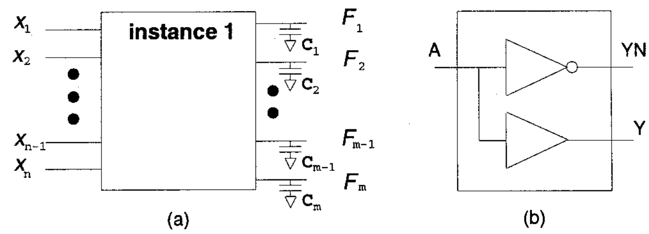

Figure 1: An example of multi-output components.

We will call this the transition state overlap problem. In Fig. 1(b), the outputs Y and YN change simultaneously when the input A is switching. If we calculate the power consumption for each output transition without considering the transition state overlap problem, the power estimation results will be too pessimistic.

- **Signal path selection problem :** Exact estimation of power consumption requires the information of input patterns that causes output transitions, but the absence of this information is a source of errors during dynamic power estimation. Actually, we can get this information through the logic simulation, but in the case of large circuits, generating a list of all events during logic simulation is impractical because it can be enormous in size and significantly degrade the speed of simulation.

- **Non-output switching power consumption:** It is difficult to drive a quantitative value for the input switching activity which does not generate output transitions from the given input and output switching information of instances. Thus, we cannot get accurate power consumption of the non-output switching case, even if we characterize the power value of non-output switching.

In this paper, we propose a new characterization methodology and the probabilistic contribution measure(PCM) algorithm which improve the accuracy of the dynamic power estimation. The proposed techniques have been implemented in (a software program)CubicPower. The discrepancy between the results of CubicPower and the measured values is within 10% whereas that of a commercial tool is over 20%.

2 Power Characterization Model

The power estimation model of CubicPower consists of switching power(E_{swt}) and non-switching power($E_{non-swt}$). The switching power is the power dissipated by a gate when its output is switching, and the non-switching power is the power dissipated by a gate only when its input is switching. E_{swt} is characterized for the rise/fall energy for each input-output arc. In order to characterize the switching energy, we measure the current at the source voltage node(VDD). Thus, only E_{swt} of the rising transition includes the energy charging the output load.

To manage the multiple output case more effectively, we separate the energy charged in the output load C_m and the energy consumed internally by subtracting $C_m VDD^2$ from the rising transition energy.

We also consider the state dependency during the power characterization. The state dependency means that the characterization result of the arc from the input i to the output j is dependent on the state of other inputs. To consider the state dependency during the estimation process, all states causing different characterization results are classified using a function of input values before the output transition occurs.

We next define the switching state and non-switching state which correspond to characterized energy $E_{swt_k^{ij}}$ and $E_{non-swt_k^i}$ to estimate the total power consumption.

Definition 1 Swiching State
For the j-th output transition caused by the i-th input transition, we define the switching state S_k^{ij} based on the state of other inputs. S_k^{ij} is represented as a Boolean function, $\frac{\partial F_j}{\partial x_i} = \bigcup_{k=1}^{s_{ij}} S_k^{ij}$, where s_{ij} is the number of switching states. And $E_{swt_k^{ij}}$ is defined as the characterized energy of S_k^{ij}.

Definition 2 Non-Swiching State
For a case when no output transition occurs from the i-th input transition, we define the non-switching state S_k^i based on the state of other inputs. S_k^i is represented as a Boolean function, $(\frac{\partial F_j}{\partial x_i})' = \bigcup_{k=1}^{s_i} S_k^i$, where s_i is the number of non-switching state. And $E_{non-swt_k^i}$ is defined as a characterized energy of S_k^i.

3 Power Calculation Algorithm

We propose the probabilistic contribution measure(PCM) algorithm to resolve the problem mentioned in Section 1. The PCM algorithm not only removes the logic simulation overhead which may be required to identify the event propagation path but also increases the accuracy of the estimation.

In dynamic power estimation, the charged power in the output load($Power_{load}$), switching power($Power_{swt}$), non-switching power($Power_{non-swt}$) are calculated for each instance. The PCM algorithm for each instance is given by (1)-(7).

$$Power_{inst} = Power_{load} + Power_{swt} + Power_{non-swt} \quad (1)$$

The following notations are used in equations.

- n : the number of inputs
- m : the number of outputs
- Sen_{ij} : sensitizing probability from input i to output j
- $P(f)$: signal probability of function f
- $n_i(T)$: net switching count of i during interval T
- C_j : load capacitance of j-th output

First, $E_{swt_k^{ij}}$ and $E_{non-swt_k^i}$ are calculated by the output loading capacitance and input slew. Then, the power consumption of instances is calculated with the given transition density D_i [3].

$$Power_{load} = \sum_{j=1}^{m} C_j VDD^2 \frac{D_j}{2} \quad (2)$$

Next we define $P_{norm_k^{ij}}$ as in (3). The nominator represents the probability that the output j is switched by the transition of input i excluding the switching activity of previous outputs from 1 to $j-1$. The denominator represents the probability for the sum of states of input i that causes outputs transition. Thus, the meaning of $P_{norm_k^{ij}}$ can be stated as the contribution factor of the kth state of of jth output transition on power, excluding the transition overlap using the switching state function. Then, $P_{norm_k^{ij}} E_{swt_k^{ij}}$ represents the internal energy consumed by state k with input i and output j node transition. For example, if S_k^{ij} is already included in the previous output $P_{norm_k^{ij}}$ calculation phase, S_k^{ij} is a subset of $\bigcup_{j'=1}^{j-1} S_k^{ij'}$, thus $P\left(S_k^{ij} - \bigcup_{j'=1}^{j-1} S_k^{ij'}\right)$ is zero. Therefore, $P_{norm_k^{ij}}$ is zero which means a switching state S_k^{ij} does not contribute to the power consumption any more. Through the concept of PCM, we can solve the **transition state overlap problem.**

$$P_{norm_k^{ij}} = \frac{P\left(S_k^{ij} - \bigcup_{j'=1}^{j-1} S_k^{ij'}\right)}{P\left(\bigcup_{j'=1}^{m} S_k^{ij'}\right)} \quad (3)$$

Then PCM_{ij} from input i to output j is defined as in (4). The summation term represents the effective power consumption of all states with input i and output j transition.

$$PCM_{ij} = \left\{\sum_{k=1}^{s^{ij}} P_{norm_k^{ij}} E_{swt_k^{ij}}\right\} \quad (4)$$

Finally, we obtain $Power_{swt}$ as shown in (5). The term Sen_{ij} is used for extracting the portion of output j switching affected by input i among the total transition density D_j. This sensitizing ratio is used to account for the **path selection problem** mentioned previously in Section 1.

$$Power_{swt} = \sum_{i=1}^{n} \sum_{j=1}^{m} PCM_{ij} D_j \frac{Sen_{ij}}{\sum_{i'=1}^{n} Sen_{i'j}} \quad (5)$$

We must classify the quantity of input switching into the quantity affects to the switching power and non-switching power during the power calculation, because the switching and non-switching states are mutually exclusive as defined in Section 2. D_i^{oeff} represents the transition density for an i-th input contributing to the switching power as shown in (6). Thus, we can calculate the **non-switching power** of input i accurately.

$$D_i^{oeff} = \left(\sum_{j=1}^{m} D_j \left(\sum_{k=1}^{s^{ij}} P_{norm_k}\right) \frac{Sen_{ij}}{\sum_{i'=1}^{n} Sen_{i'j}}\right) \quad (6)$$

$$Power_{non-swt} = \sum_{i=1}^{n} \left(\sum_{k=1}^{s^i} \frac{P(S_k^i)}{P\left(\bigcup_{k'=1}^{s^i} S_{k'}^i\right)} E_{non-swt_k^i}\right) \left(D_i - D_i^{oeff}\right) \quad (7)$$

Finally we have solved the problems which are defined as transition state overlap, path selection and non-switching problem with the proposed PCM algorithm.

4 Experimental Results

To demonstrate the accuracy of the proposed algorithm, we compare the result of CubicPower with those of the IMS tester[4] and PowerMill[5]. Table 1 shows the number of gate count for these circuits. The switching activity and signal probability are obtained using Verilog simulation through the customized programmable logic interface(PLI)[6] for CubicPower.

Table 1: Sample Design Specification.

design name	gate count	operating frq.
sample1	1574	28Mhz
sample2	16139	10Mhz
sample3	238925	33Mhz
sample4	245998	33Mhz

Table 2: PowerMill vs. CubicPower.

Name	current(mA):run time(sec)		speed &
	CubicPower	PowerMill	accuracy
sample1	2.82 : 49	2.97 : 7320	183X 5.0%
sample2	6.80 : 180	7.29 : 19800	110X 6.7%
sample3	150.07 : 7200	NA : NA	- -
sample4	177.40 : 8100	NA : NA	- -

Table 2 shows the run time comparison for 4 example circuits. These examples were run on UltraSparc 1 with 256MB of memory. PowerMill could not complete the simulation for circuits larger than 200K gates. For smaller examples, CubicPower is about two orders of magnitude faster than PowerMill while the accuracy is within 7% of PowerMill. The run time of CubicPower in this table includes the run time of CubicPower as well as Verilog simulation.

We also compared the result of CubicPower with measured values of the IMS test equipment and results of a leading commercial dynamic power estimator, as shown in Table 3. The average power values consumed by *sample*3 and *sample*4 circuit are obtained using the IMS tester for various input vectors. The length of these input vectors range from 40k to 80k patterns. We run CubicPower and ToolA which is commonly used commercial dynamic power estimator, for same circuits using the same input vectors. Table 3 shows the % error of CubicPower and a commonly used leading industry dynamic power estimator compared to the measured values using the IMS tester. The average error of CubicPower is 6% while that of the commercial tool is 15%. We did the same experiment on the circuit *sample*4 as shown in Table 4. In this case, CubicPower is also more accurate than the commercial tool.

Table 3: sample3 Comparison Table.

Test Vector	IMS (mA)	CubicPower (mA)	ToolA (mA)	CubicPower (err %)	ToolA (err %)
vector1	147	150.7	172.8	2.5	17.6
vector2	140	152.4	163.4	8.9	16.7
vector3	149	155.3	164.2	4.2	10.2
vector4	150	159.4	181.8	6.3	21.2
vector5	159	165.2	172.9	3.9	8.7
vector6	147	162.2	171.2	10.3	16.5
vector7	160	165.4	175.4	3.4	9.6
Average				5.6	14.6

Table 4: sample4 Comparison Table.

Test Vector	IMS (mA)	CubicPower (mA)	ToolA (mA)	CubicPower (err %)	ToolA (err %)
vector1	175	177.4	197.5	1.4	12.9
vector2	190	189.4	193.4	-0.3	1.8
vector3	150	155.7	165.6	3.8	10.4
vector4	160	173.9	183.6	8.7	14.7
vector5	170	170.8	182.3	0.5	7.2
vector6	165	169.4	175.6	2.7	6.4
vector7	145	150.1	157.1	3.5	8.3
Average				2.9	8.8

5 Conclusions

We described the new power estimation algorithm used in CubicPower. For more accurate dynamic power estimation, we proposed a probabilistic method called probabilistic contribution measure(PCM) which overcomes the limitation of conventional method, such as the transition state overlap problem, path selection problem and non-switching power calculation. Experimental results of CubicPower show less than 10% error compared to the measured values of the IMS test equipment, and also show 7% error with the results of PowerMill simulation. Due to the PCM algorithm, CubicPower is more accurate than the leading commercial dynamic power estimator at the gate level and is 2-3 orders of magnitude faster than PowerMill.

CubicPower is a component of CubicWare[7] which is composed of the floorplanner, delay calculator and power estimator. The result of CubicPower can be interfaced with the IR drop analysis called RailAdvisor[8].

References

[1] A. Ghosh, S. Devadas, K. Keuzer, and J. White, "Estimation of Average Switching Activity in Combinational and Sequential Circuits." *Proc. Design Automation Conf.*, pp. 253–259, June 1992.

[2] D. Marculescu, R. Marculescu and M. Pedram, "Switching Activity Analysis Considering Spatiotemporal Correlations," *Proc. IEEE Int. Conf. CAD*, pp. 294–299, Nov. 1994.

[3] F. N. Najm. "Transition Density: A New Measure of Activity in Digital Circuits." *IEEE Trans. Computer-Aided Design*, Vol.10,No. 4, pp. 310–323, Feb. 1993.

[4] Integrated Measurement System Inc. *IMS Manual*, 1998.

[5] SYNOPSYS Inc. *PowerMill User Manual*, 1998.

[6] Cadence Design Systems Inc. *Programming Language Interface Guide*, June 1998.

[7] M.-S. Jang, H.-S. Jin, B.-H. Lee, J.-Y. Lee, T.-S. Kim, J.-T. Kong and S.-J. Song, "A Hierarchical Design System for Deep Submicron ASIC." *to be appeared in Proc. IEEE Int. ASIC/SOC Conference*, Sep. 1999.

[8] D.-S. Cho, K.-H. Lee, G.-J. Jang, T.-S. Kim and J.-T. Kong, "Efficient Modeling Techniques for IR Drop Analysis in ASIC Designs." *to be appeared in Proc. IEEE Int. ASIC/SOC Conference*, Sep. 1999.