VLSI DESIGN
2001, Vol. 12, No. 2, pp. 221–231
Reprints available directly from the publisher
Photocopying permitted by license only

Backward Propagated Capacitance Model for Register Transfer Level Power Estimation

JUNG YUN CHOI[a,*], YOUNG HWAN KIM[a,†] and KYOUNG-ROK CHO[b,‡]

[a]*Department of Electronic and Electrical Engineering, Pohang University of Science and Technology, Pohang, Kyungbuk, 790-784, Republic of Korea;* [b]*School of Electrical and Electronics Engineering, Chungbuk National University, Chungbuk, 361-763, Republic of Korea*

(Received 20 June 2000; In final form 3 August 2000)

We present a new approach to the power modeling of functional modules, referred to as the backward propagated capacitance model, for estimating the power consumption of VLSI systems that are described at the register transfer level (RTL). To construct the proposed model, we investigate the effect of the module's internal capacitance on power consumption at the gate level. Then, we store the effect in a library in terms of the equivalent input capacitance of the module. The equivalent input capacitance is used to compute the module's power without the lower level elaboration during the power analysis of the RTL system. In the experiment using benchmark functional modules, the proposed model showed the absolute modeling error of 1.39% on average. For the benchmark RTL systems, the proposed model exhibited the absolute error of 3.04% in power estimation on average. If signal characteristics deviate from the modeling condition, the modeling error may increase. Experimental results show that the modeling accuracy can be improved greatly by using a simple compensation method.

Keywords: Power estimation; RTL design; Power model

1. INTRODUCTION

Today's VLSI system consumes a large amount of electrical power, as exemplified by the recent high-performance microprocessors that consume tens of watts [1–3]. As the system speed becomes faster and the system function gets more complex, power consumption tends to become larger. Thus, low power became one of the important design issues for current VLSI systems [4]. Power minimization is stressed at all levels of design hierarchy, and we need to minimize power consumption whenever possible [5–9]. However, as the system architecture can be easily explored, a design can be optimized more effectively at high levels such as the register transfer level (RTL). Thus, it is very

*e-mail: jychoi@cafri.postech.ac.kr
†Corresponding author. Tel.: +82-54-279-2227, Fax: +82-54-279-5933, e-mail: youngk@postech.ac.kr
‡e-mail: krcho@cbucc.chungbuk.ac.kr

meaningful to develop accurate and efficient RTL power estimation techniques that can be used early in the design process.

For power estimation, RTL systems are divided into combinational and control logic circuits. The proposed approach is applicable for a variety of combinational logic circuits. In case of combinational logic circuits, the major effort has been put to develop accurate and efficient techniques to estimate the power consumption of functional modules for various input signal characteristics [10]. Among them, early techniques are based on the concept of the *gate equivalent*, which is an average number of reference gates required to implement a specific functional module [11, 12]. The power consumption of the functional module is estimated to be the product of the gate equivalent and the average power consumption of the reference gate. This approach is very efficient. However, Glaser *et al.*, use the data of the single reference gate and do not take account of the module types [11]. In addition, they use fixed activity factors for input signals, independent of their patterns. Thus, the method suffers from large analysis error. In an improved method [12], Svensson *et al.*, adopt the customized estimation techniques for the functional modules of different types. However, the method still uses fixed activity factors. Compared with this, recent approaches construct a power model for each type of functional modules [13]; *i.e.*, they analyze the functional modules through simulation, and store the information on power consumption in a library. Then, they refer to the library when analyzing RTL systems designed using the functional modules. Among them, the *power factor approximation* method [14] uses the input signal of the uniform white noise (UWN) for simulation. Thus, it exhibits the estimation error of up to 80% in comparison with gate level estimation results [15]. In the *DBT* power model [16, 17], Landman *et al.*, divide the input data into two regions, most significant bit (MSB) and least significant bit (LSB) regions. Then, for the simulation to extract model parameters, they apply the input signals with strong temporal dependency to the MSB region, while applying the UWN to the LSB region. With this simple method, they succeeded in improving the modeling accuracy greatly. The *DBT* power model has 10–15% error compared with the results of the switch level simulator, *IRSIM-CAP* [17]. However, the *DBT* power model is more suitable for DSP applications than other applications, and users need to provide the power models of the functional modules in analytic expressions that depend on the module functionality. This is a big burden on users. In [18–20], Nemani *et al.*, proposed to estimate the power consumption using only the functional description of the system such as Boolean equations. The advantage is that they do not require building a library in advance. But, these approaches may lead to inaccurate analysis results, because they need to make some assumptions and approximations to find the average activity and the area of the functional module before implementation. Nemani *et al.*, expressed that their approach has 33.7% error compared with gate level estimation results.

In this paper, we present a new RTL power model, referred to as *BPCM* (Backward Propagated Capacitance Model), which represents the power behavior of the functional module as the equivalent input capacitance for various input signal characteristics. The proposed model is constructed as follows: first, we expand the given RTL module into the gate level. Then, we visit the internal nodes of the given RTL module backward, beginning from the output. During the visit, we move the capacitance at each internal node in the direction of the input, while maintaining the power consumption same. During the process, we use the internal signal information that we can obtain through the gate level simulation with the full delay model. This is referred to as the *backward propagation* of the internal capacitance. When reaching the input nodes, we can represent all internal capacitance as the equivalent input capacitance of the module, which is stored in a library for future use. Since the proposed model is constructed through the investigation on the effect

of the internal capacitance on power, it can characterize the module's power consumption more accurately than existing library-based models [14, 16, 17, 21] in the wide range of input signal characteristics. In addition, the model parameters of the multi-input module are extracted for each input node, one by one and independent of other inputs. Thus, the module power can be characterized through simple procedures, and can be stored in a one-dimensional table using a small amount of memory. The good trade-off ability between complexity and accuracy makes the proposed model attractive to use for the power analysis and optimization at the high level of abstraction.

This paper is organized as follows. In Section 2, we describe the principle and embodiment of the proposed power model, *BPCM*. In Section 3, we present the application of *BPCM* to power analysis and experimental results, and, in Section 4, we conclude the paper.

2. THE PROPOSED POWER MODEL: BACKWARD PROPAGATED CAPACITANCE MODEL

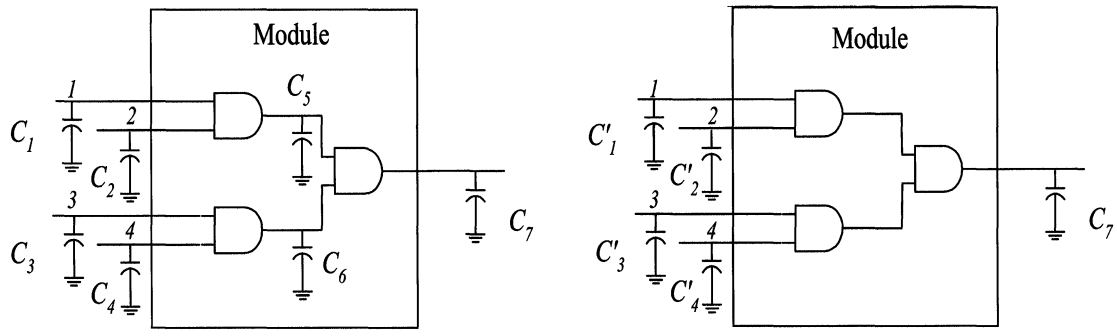
2.1. Principle of the Proposed Model

Consider an RTL module whose gate level structure is shown in Figure 1(a). In the circuit, C_1 to C_4 are the input capacitance, C_5 and C_6 are the internal capacitance, and C_7 is the output capacitance of the module.

The principle of the proposed model, *BPCM*, is to represent the internal capacitance, C_5 and C_6, as the input capacitance of the module, which is equivalent in power consumption, as shown in Figure 1(b). In the figure, C_5 and C_6, has been removed. Instead, C_i was replaced by C_i' that represents the sum of C_i and the equivalent input capacitance, ΔC_i, coming from C_5 and C_6. Note that, unlike the internal capacitance, the output capacitance C_7 remains untouched, as the output node appears in the RTL description of the system. With the equivalent input capacitance, the power consumption of the module is computed as follows:

$$PW = 0.5 \times V^2 \times f \times \left\{ \sum_{i=1}^{n} (C_i' \times S_i) + \sum_{o=1}^{m} (C_o \times S_o) \right\} \quad (1)$$

where V is the power supply voltage and f is the clock frequency synchronizing the RTL system. And, n and m are the number of the inputs and outputs of the module, S_i is the *switching activity* at the input i, and S_o and C_o are the switching activity and node capacitance at the output o, respectively. Here, the switching activity is the



(a) An RTL module example **(b) Application of *BPCM* to the RTL module of (a)**

FIGURE 1 The principle of *BPCM*.

average number of $0 \rightarrow 1$ or $1 \rightarrow 0$ transitions that a logic signal makes per clock period [22].

2.2. Derivation of the Proposed Model

The amount of power that a functional module consumes depends on not only the switching activity but also the signal probability at input nodes, where the *signal probability* is the probability that the logic state of the signal will be '1'. Thus, the equivalent input capacitance at input i, ΔC_i, coming from the internal capacitance is given as follows:

$$\Delta C_i = f(P_1, P_2, \ldots, P_n, S_1, S_2, \ldots, S_n), \quad i = 1, \ldots, n \qquad (2)$$

where P_i and S_i are the signal probability and switching activity at input node i of the functional module, respectively. From (2), ideally, we need to characterize the power behavior of the functional module for various combinations of signal probabilities and switching activities, preparing for all the cases that can happen during the actual operation. However, in that case, the modeling equations or the look-up table will be $(2 \times n)$ dimensional, and extensive simulation will be required to obtain the necessary data on the functional module. Thus, the method will be obviously too complex to be practical. Considering this, most approaches [14, 20, 21] assume that the input signals are UWN. However, these input vectors are certainly incapable of representing all important input conditions that the functional module would see during actual operation.

To derive the proposed approach, we assume that there are no glitches and no temporal correlation in the input signals of the functional module. Under the assumption, the following relationship holds [22]:

$$S_i = 2P_i(1 - P_i) \qquad (3)$$

where P_i and S_i are the signal probability and the switching activity at input node i. Substituting (3) into (2) yields

$$\Delta C_i = f(P_1, P_2, \ldots, P_n), \quad i = 1, \ldots, n \qquad (4)$$

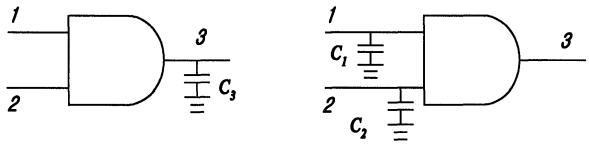
Thus, the power characterization becomes n dimensional. The compensation for the cases that do not satisfy (3) will be described in Section 2.5. Although the above method reduces the modeling dimension from $2n$ to n, (4) is still too complex for practical use. Therefore, when extracting ΔC_i, we set $P_j = 0.5, j \neq i$, to reduce the modeling complexity further. Then, ΔC_i is represented in the following one-dimensional form:

$$\Delta C_i = f(P_i), \quad i = 1, \ldots, n \quad \text{and} \quad P_j = 0.5,\ j \neq i \qquad (5)$$

From (5), we use the following input vectors for the simulation to obtain the information for modeling; for input i, we use the input vectors with various signal probabilities between 0 and 1, and for other inputs, we apply the input vectors with the signal probability of 0.5.

2.3. Backward Propagation of the Capacitance

The basic operation to obtain the equivalent input capacitance of the functional module from internal capacitance is the *backward propagation* of the capacitance, which converts the output capacitance of a gate to its equivalent input capacitance. This is illustrated in Figure 2, where both circuits show the same gate, but before and after propagating C_3 backward. Notice that, after the backward propagation, the capacitance C_3 was removed. Instead, the equivalent capacitance was added at the gate input, shown as C_1 and C_2.



(a) Before the backward propagation (b) After the backward propagation

FIGURE 2 The backward propagation of capacitance, C_3.

To maintain the power consumption of the both circuits same, we have to conserve the total switched-capacitance after the backward propagation. Let S_i represent the switching activity of the signal at node i. Then, for the circuit of Figure 2(a), the switched-capacitance is given as follows:

$$SWC(a) = S_3 \times C_3 \tag{6}$$

Assume that there are no glitches in the input, the input transition probability is evenly distributed in time, and the gate has the zero delay. Then, for the 2-input AND gate, the following relationship holds:

$$S_3 = S_1 \times P_2 + S_2 \times P_1 \tag{7}$$

where P_i is the signal probability at node i. Thus, the switched-capacitance is given as follows:

$$SWC(a) = (S_1 \times P_2 \times C_3) + (S_2 \times P_1 \times C_3) \tag{8}$$

On the other hand, the total switched-capacitance of the circuit of Figure 2(b) is given as follows:

$$SWC(b) = S_1 \times C_1 + S_2 \times C_2 \tag{9}$$

Thus, the comparison of (8) and (9) leads to the following equivalent input capacitance:

$$C_1 = P_2 \times C_3 \tag{10a}$$

$$C_2 = P_1 \times C_3 \tag{10b}$$

The above equations show that the equivalent input capacitance depends on the input signal probabilities. In addition, the relationship of the signal characteristics, shown in (7), becomes more complex as the gate has more input nodes, and it is affected by the gate functionality. As this complicates the backward propagation of the capacitance, we approximate that all input signal probabilities are same to improve practicality. Then, from (10a) and (10b), we obtain $C_1 = C_2$. Let $M = C_1 = C_2$. The total switched-capacitance of the circuit of Figure 2(b) is given as follows:

$$SWC(b) = S_1 \times M + S_2 \times M = (S_1 + S_2) \times M \tag{11}$$

From (6) and (11),

$$M = \frac{S_3}{S_1 + S_2} \times C_3 \tag{12}$$

By extending the above idea, we model that all inputs of the n-input general gate have the same equivalent capacitance after the backward propagation of the output capacitance. Thus, the conservation of the switched-capacitance leads to the following formula:

$$M = \frac{S_o}{\sum_i S_i} \times C_o \tag{13}$$

where M is the equivalent capacitance at each input node, S_i is the switching activity at the ith input of the gate, and S_o and C_o are the switching activity and node capacitance at the gate output.

2.4. Construction of the Proposed Model

For a given functional module, we construct the proposed model by obtaining the equivalent input capacitance at each input node, one by one. For the modeling, we need the switching activities at the internal nodes, and they are obtained through the gate level simulation of the functional module with the full delay model. For the simulation, we use the input vectors, generated as described in the end of Section 2.2.

To obtain the equivalent capacitance at the input node of the functional module, we need to visit the internal nodes backward from output nodes systematically, and we need to propagate the internal capacitance backward until we reach the input nodes. For this, we define the levels of the internal nodes of the functional module as follows, and visit them in the decreasing order of levels,

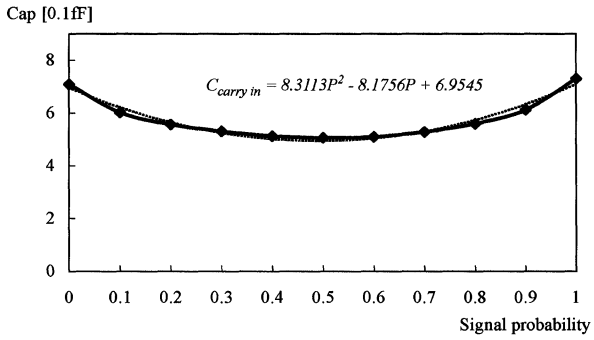


FIGURE 3 A *BPCM* polynomial example.

where *level* (*i*) is the level of node *i*:

$$level\,(i) = \text{Max}\,\{level\,(j) + 1\}, \quad \forall j \tag{14}$$

In (14), *j* is the fan-in nodes of *i*, and the level of all input nodes of the module is defined as 1.

After obtaining the equivalent capacitance values at each input node for various signal probabilities, we store them in a library in the form of the following polynomial after regression [23]:

$$C_i = a_n P^n + a_{n-1} P^{n-1} + \cdots + a_1 P + a_0 \tag{15}$$

where P is the signal probability at the input node and a_j is the polynomial coefficient. In (15), n is determined considering the accuracy and efficiency required. An example is shown in Figure 3, where the thick line with lozenges is the equivalent input capacitance at "*carry in*" of a 1-bit full adder. In this example, we obtained the equivalent input capacitance at "*carry in*" for the signal probability from 0 to 1 with the step of 0.1, and, then, we represented the capacitance using a second-order polynomial. This *BPCM* polynomial is stored in a library, and is used to retrieve the equivalent input capacitance at "*carry in*" of the 1-bit full adder, when analyzing the power consumption of the VLSI system.

2.5. Effect of the Switching Activity on Power Consumption in *BPCM*

During the derivation of the proposed model, *BPCM*, we assumed that there are no glitches and no temporal correlation in the input signals of the functional module, *i.e.*, the following relationship holds:

$$S_i = 2P_i(1 - P_i) \tag{16}$$

where P_i and S_i are the signal probability and switching activity at input node *i*. Let ΔC_i denote the equivalent input capacitance of the functional module, coming from the internal capacitance. Then, the power consumption by the internal capacitance at input *i* is given as follows:

$$\begin{aligned} PW_i &= 0.5 \times V^2 \times f \times \{\Delta C_i \times S_i\} \\ &= V^2 \times f \times \Delta C_i \times \{P_i \times (1 - P_i)\} \end{aligned} \tag{17}$$

However, in general, the assumption of (16) does not hold in VLSI systems, and this may result in inaccurate power estimation. The power dependency on switching activity is illustrated in Figure 4, which compares the power consumed by the internal capacitance of a 4-bit × 4-bit multiplier for various signal probabilities, while changing the switching activity. The data was obtained through the gate level analysis with the full delay model for gates. In the figure, large bullets represent the points where the relationship of (16) holds. From the figure, it is shown that the power consumption increases almost linearly with the switching activity, and the slopes are similar regardless of the value of the signal probability. This is a common tendency in most functional modules, while they exhibit different slopes. Thus, we compensate the power consumption by ΔC_i at input *i* for switching activity S_j, as follows:

$$PW_i(S_j) = \alpha(S_j - S_i) + PW_i \tag{18}$$

where α is the slope of the power variation with the switching activity, and PW_i is the power consumed at input *i* for the signal of the switching activity, S_i, given in (17). Since different signal probabilities exhibit similar values of α, we use the value of α at the signal probability of 0.5 to compute PW_i (S_j) in (18). After obtaining the

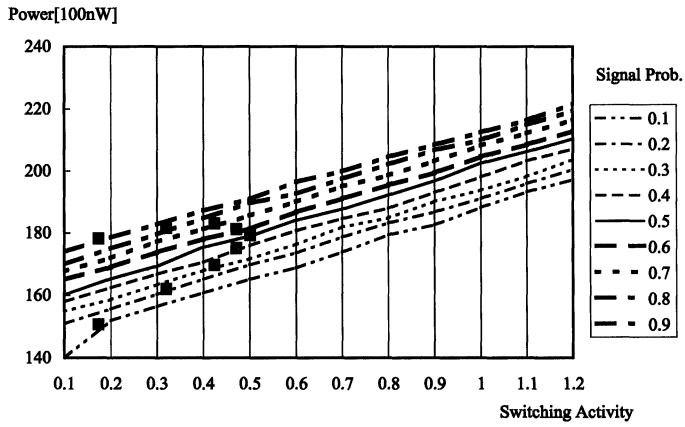


FIGURE 4 The effect of switching activity on power consumption.

values of α for all input nodes of the functional module, we store them in a library along with the *BPCM* polynomials.

3. APPLICATION TO POWER ANALYSIS AND EXPERIMENTAL RESULTS

To analyze the power consumption of the RTL system, we first simulate the given system using the user-specified input vectors, and obtain the signal information at the input of all functional modules in the system. Next, using the signal information, we obtain the power consumption of each module in the system, by referring to the *BPCM* library. Then, the sum of the power consumption of all functional modules becomes the system's power consumption. This process is illustrated in Figure 5.

We evaluated the accuracy of *BPCM* using *DesignPower*, a gate-level power analysis program from *Synopsys*. As test circuits, we used benchmark functional modules, and RTL systems designed using the functional modules. To calculate the power of the test circuits using *DesignPower*, we used the signal information in the circuits, which we obtained through the gate-level simulation with the full delay gate model. As the simulation input, we used an input vector of 10,000 sequences with 10 nSec data period.

Table I shows the modeling error of *BPCM* for various functional modules, found in the *Synopsys DesignWare* library and *ISCAS*'85 combinational

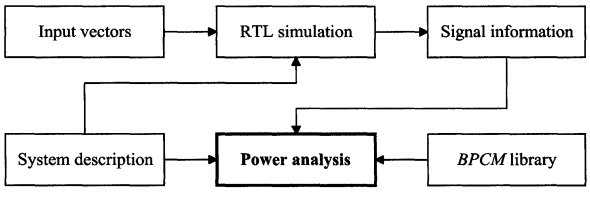


FIGURE 5 The process of power analysis using the proposed model, *BPCM*.

TABLE I Modeling error of *BPCM* for functional modules (UWN input)

Module	Number of inputs	Number of cells	Error
C17	5	6	+1.5%
3-bit decoder	3	10	+0.69%
4-bit barrel shifter	6	10	−0.77%
1-bit full adder	3	11	+1.2%
2-bit × 2-bit multiplier	5	17	−1.84%
4-bit ripple carry adder	9	20	−0.67%
C432	36	160	+1.86%
16-bit carry look-ahead adder	33	285	+1.18%
32-bit absolute-value circuit	32	322	−1.51%
16-bit subtractor	33	394	+1.05%
C3540	50	1,788	+1.58%
32-bit × 32-bit multiplier	65	8,571	−2.87%
Average absolute error			1.39%

logic benchmark circuits. As the input, we used UWN. In the table, *Error* is defined as follows:

$$Error[\%] = \frac{|Power(DesignPower) - Power(BPCM)|}{Power(DesignPower)} \times 100 \tag{19}$$

The table shows that the proposed model, *BPCM*, exhibits the absolute modeling error less than 1.4% on average, and about 1.6% error for C3540 that is as large as to contain 1,788 gate cells.

Table II shows the analysis error of *BPCM* for the RTL systems, designed using the functional modules that were presented in Table I. The table shows that *BPCM* exhibits the analysis error for RTL systems, which is similar to the modeling error for the functional modules. But, in this case, the error increased a little bit because we obtained the necessary signal information of the system through the RTL simulation that uses the zero delay model for the functional modules, while *DesignPower* used the full delay model for logic gates.

The *circuit for four rules of arithmetic*, in Table II, contains 120 functional modules. Figure 5 compares the power estimates for each functional module by *BPCM* and *DesignPower*. In the figure, the X and Y axes represent the power estimates by *DesignPower* and *BPCM*, and each dot corresponds to a functional module. If the power estimates by *BPCM* are exactly same as those by *DesignPower*, a 45° straight dotted line corner appears from the left bottom to the right top. From Figure 5, it is observed that *BPCM* provides accurate power estimates for all functional modules inside the circuit consistently.

TABLE II Analysis error of *BPCM* for RTL systems (UWN input)

System	Number of functional modules	Number of cells	Error
ALU	14	2,062	−2.98%
MIN/MAX circuit	18	1,120	+2.32%
Priority encoder	22	470	−3.26%
Duplex adder/subtractor	24	2,306	+2.4%
Circuit for generalized sum of products	58	12,580	+2.58%
Circuit for four rules of arithmetic	120	33,668	−4.71%
Average absolute error			3.04%

We characterize the power consumed at each input node of the given functional module, one by one. And, for the characterization, we fix all input signal probabilities at 0.5 except the input under characterization. However, the input signals may deviate from this condition, and Figure 7 illustrates the modeling error of *BPCM* for this case. For the experiment, we chose a 4-bit CSA (Carry Select Adder) and a 16-bit SUB (Subtractor) as test circuits, and changed the signal probabilities of all input signals together between 0.1 and 0.9. As we expected, the error is smallest when the signal probabilities are 0.5, and it increases as the signal probabilities deviate from 0.5. As the worst case, the error becomes as large as 8% when all input signal probabilities are 0.1 or 0.9. However, this seldom happens during the actual operation, and it has been reported that most of the input bits except the sign bit are very close to UWN in nature [24].

In Figure 8, we show the modeling error of *BPCM* for various signal probabilities of the sign bit, while fixing the other signal probabilities at 0.5. From the figure, it is observed that the

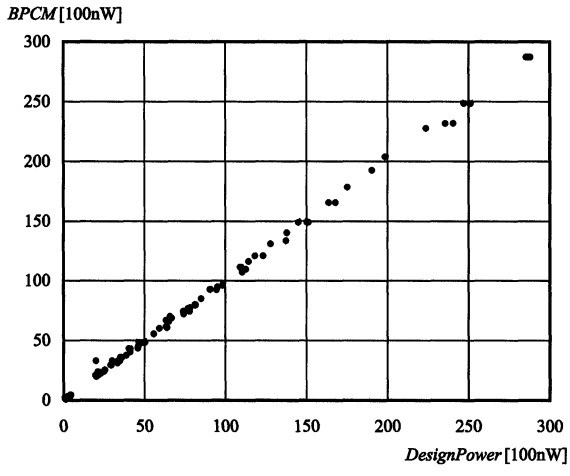


FIGURE 6 Power estimates comparison on the functional modules inside the *Circuit for four rules of arithmetic*.

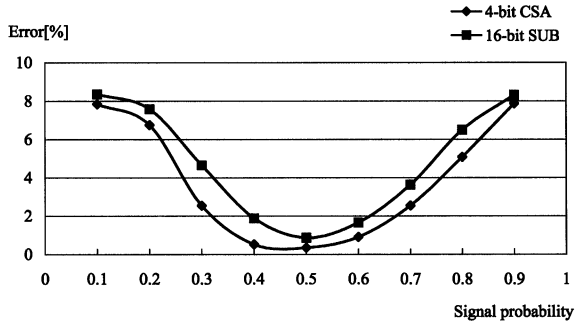


FIGURE 7 Modeling error for 4-bit CSA and 16-bit SUB when varying the signal probabilities of all input bits together.

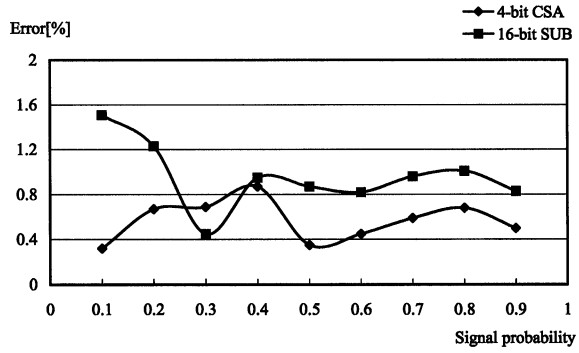


FIGURE 8 Modeling error for 4-bit CSA and 16-bit SUB when varying the signal probability of the sign bit.

modeling error of *BPCM* is less than 1.6% in the wide range of the signal probability.

In deriving *BPCM*, we assumed that the relationship of (3) holds between the signal probability and the switching activity of input signals. However, some input signals may not satisfy the relationship, and in this case, we compensate the power consumption for the switching activity using the formula of (18). Figure 9 compares the modeling error of *BPCM* before and after the compensation. For the experiment, we changed the switching activities of all input signals together from 0.1 to 1.2, while fixing their signal probabilities at 0.5. In the figure, the dotted lines and solid lines represent the modeling error of *BPCM* before and after the compensation, respectively. Similarly, Figure 10 shows the improvement of modeling accuracy by

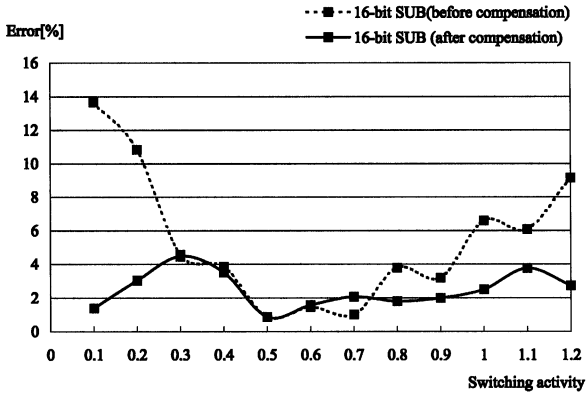


FIGURE 9 Modeling error before and after compensating for switching activity when all input signals do not satisfy the relationship of (3).

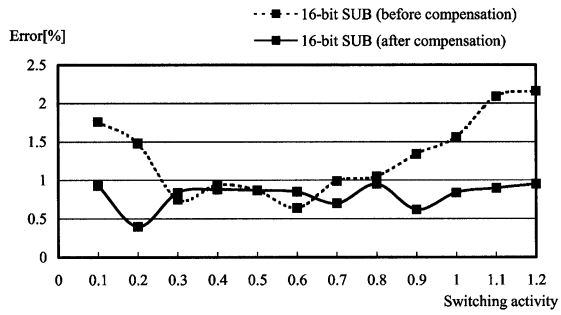


FIGURE 10 Modeling error before and after compensating for switching activity when the sign bit does not satisfy the relationship of (3).

the compensation when only the sign bit does not satisfy the relationship of (3). In both cases, it is observed that the compensation of (18) reduces the modeling error of *BPCM* greatly.

4. CONCLUSION

In this paper, we presented an accurate power model, called *BPCM*, that can be used to analyze the power consumption of the RTL system without low level elaboration. The proposed model represents the power consumption of the internal capacitance of the functional module as its equivalent input capacitance. The advantage of the proposed model is that it can characterize the

power consumption of functional modules for the wide range of input signal characteristics accurately. In addition, as the model parameters are extracted for each input node individually, independent of other inputs, the power library can be constructed through simple characterization procedures, and the model parameters are stored in a one-dimensional table using a small amount of memory. Experimental results show that *BPCM* has the absolute modeling error of 1.39% for the benchmark functional modules on average, when compared with the gate-level power estimator, *DesignPower*. For the benchmark RTL systems, *BPCM* exhibited the absolute analysis error of 3.04% on average. For the derivation of *BPCM*, we assumed that there are no glitches and no temporal correlation in the input signals of the functional module. However, the input signal may not satisfy these conditions. Experimental results showed that the compensation using (18) improves the modeling accuracy of *BPCM* greatly. From the experiment, it is concluded that the proposed model, *BPCM*, can be used to estimate the power consumption of the RTL systems accurately.

Acknowledgment

This work was supported by the Ministry of Education of Korea under the BK21 program and the Samsung Electronics Company, Republic of Korea.

References

[1] Bowhill, W. *et al.*, "A 300 MHz 64b quad-issue CMOS RISC microprocessor", *ISSCC'95 Digest of Technical Papers*, pp. 182–183, February, 1995.

[2] Gowan, M., Biro, L. and Jackson, D., "Power Considerations in the Design of the Alpha 21264 Microprocessor", *Proceedings of the 35th Design Automation Conference*, pp. 726–731, June, 1998.

[3] Tiwari, V., Singh, D., Rajgopal, S., Mehta, G., Patel, R. and Baez, F., "Reducing Power in High-performance Microprocessors", *Proceedings of the 35th Design Automation Conference*, pp. 732–737, June, 1998.

[4] Chandrakasan, A., Sheng, S. and Brodersen, R., "Low-Power CMOS Digital Design", *IEEE Journal of Solid-State Circuits*, **27**(4), 473–484, June, 1992.

[5] Bellaouar, A. and Elmasry, M. I., *Low-power Digital VLSI Design Circuits and Systems*, Kluwer Academic Publishers, 1995.

[6] Rabaey, J. and Pedram, M., *Low Power Design Methodologies*, Kluwer Academic Publishers, 1996.

[7] Devadas, S. and Malik, S., "A Survey of Optimization Targeting Low Power VLSI Circuits", *Proceedings of the 32nd Design Automation Conference*, pp. 242–247, June, 1995.

[8] Monteiro, J., Devadas, S., Ashar, P. and Mauskar, A., "Scheduling Techniques to Enable Power Management", *Proceedings of the 33rd Design Automation Conference*, pp. 349–352, June, 1996.

[9] Borah, M., Owens, R. and Irwin, M., "Transistor Sizing for Low Power CMOS Circuits", *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems*, **15**(6), 665–671, June, 1996.

[10] Macii, E., Pedram, M. and Somenzi, F., "High-level Power Modeling, Estimation, and Optimization", *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems*, **17**(11), 1061–1079, November, 1998.

[11] Glaser, K. M., Kirsch, K. and Neusinger, K., "Estimating Essential Design Characteristics to Support Project Planning for ASIC Design Management", *IEEE International Conference on Computer-aided Design*, pp. 148–151, November, 1991.

[12] Svensson, C. and Liu, D., "A Power Estimation Tool and Prospects of Power Savings in CMOS VLSI Chips", *International Workshop on Low-power Design*, pp. 171–176, April, 1994.

[13] Hsieh, C.-T., Wu, Q., Ding, C.-S. and Pedram, M., "Statistical Sampling and Regression Estimation in Power Macro-Modeling", *IEEE International Conference on Computer-aided Design*, pp. 583–588, November, 1996.

[14] Powell, S. R. and Chau, P. M. (1990). "Estimating Power Dissipation of VLSI signal processing chips: the PFA technique", *VLSI Signal Processing IV*, pp. 250–259.

[15] Gupta, S. and Najm, F., "Power Macromodeling for High Level Power Estimation", *Proceedings of the 34th Design Automation Conference*, pp. 365–370, June, 1997.

[16] Landman, P. and Rabaey, J., "Architectural Power Analysis: The Dual Bit Type Method", *IEEE Transactions on VLSI systems*, **3**, 173–187, June, 1995.

[17] Landman, P. and Rabaey, J., "Black-box Capacitance Models for Architectural Power Analysis", *Proceedings of the International Workshop on Low Power Design*, pp. 165–170, April, 1994.

[18] Nemani, M. and Najm, F., "Toward a High-Level Power Estimation Capability", *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems*, **15**(6), 588–598, June, 1996.

[19] Nemani, M. and Najm, F., "High-level Area Prediction for Power Estimation", *Proceedings of IEEE Custom Integrated Circuits Conference*, pp. 475–478, May, 1997.

[20] Nemani, M. and Najm, F., "High-Level Area and Power Estimation for VLSI Circuits", *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems*, **18**(6), 697–713, June, 1999.

[21] Katkoori, S. and Vemuri, R., "Architectural Power Estimation Based on Behavioral Level Profiling", *VLSI Design: An International Journal of Custom-chip Design, Simulation, and Testing*, **7**(3), 255–270, August, 1998.

[22] Pedram, M., "Power Minimization in IC Design: Principles and Applications", *ACM Transactions on Design Automation of Electronic Systems*, **1**(1), 3–56, January, 1996.

[23] Press, W., Teukolsky, S., Vetterling, W. and Flannery, B., *Numerical Recipes in C*, Cambridge University Press, 1992.

[24] Landman, P. and Rabaey, J., "Activity-Sensitive Architectural Power Analysis", *IEEE on Computer-aided Design of Integrated Circuits and Systems*, **15**(6), 571–587, June, 1996.

Authors' Biographies

Jung Yun Choi received the B.E. degree in electronics in 1997 from the Kyungpook National University, Rep. of Korea, and M.S. degree in electronic and electrical engineering in 1999 from the Pohang University of Science and Technology, Rep. of Korea. Currently, he is working toward the Ph.D. degree in electronic and electrical engineering at the Pohang University of Science and Technology, Rep. of Korea. His research interests include all aspects of the computer-aided design of integrated circuits, especially, high-level power estimation and optimization.

Young Hwan Kim received the B.E. degree in electronics in 1977 from the Kyungpook National University, Rep. of Korea, and the M.S. and Ph.D. degrees in electrical engineering in 1985 and 1988 from the University of California, Berkeley, USA. From 1977 to 1982, he was with the Agency for Defense Development, Rep. of Korea, where he was involved in various military engineering projects. From 1983 to 1988, he worked as a post graduate researcher at the Electronic Research Lab., University of California, Berkeley, USA. He also has served as editor of the Journal of the Institute of Electronics Engineers of Korea from 1991 to 1996. He is currently an associate professor of Electronic and Electrical Engineering, Pohang University of Science and Technology, Rep. of Korea. His research interest includes the design of special-purpose high-speed subsystem and all aspects of the computer-aided design of integrated circuits.

Kyoung-Rok Cho received the B.E. degree in Electrical Engineering in 1977 from the dept. of Electronics Engineering at Kyungpook National University, Rep. of Korea, and earned his MS. and Ph.D. degree in dept. of Electronics Engineering from the University of Tokyo, Tokyo, in 1989 and 1992, respectively. He is currently an associate professor in School of Electrical and Electronics Engineering of the Chungbuk National University, Rep. of Korea. His research interests are in the field of low-power and high speed circuits, self-timed circuits and microprocessor architectures. From Jan. 1999 to Jan. 2000, he was a visiting professor at the Oregon State University, Oregon. From 1979 to 1986, he was with TV research center of Gold Star Company in Rep. of Korea. He is a member of Institute of Electrical and Electronics Engineers (IEEE) and Korea Institute Telecommunication Electronics (KITE).