

# Simulation Based Architectural Power Estimation for PLA-Based Controllers

Srinivas Katkoori and Ranga Vemuri  
ECE&CS Department  
University of Cincinnati  
Cincinnati, OHIO, OH 45221-0030  
{skatkoor ,ranga}@ece.uc.edu

## Abstract

We present an architectural power simulation technique for PLA-based controllers. The contributions of this work are (1) a simple but efficient power characterization of PLAs; and (2) a strategy for developing a simulatable power model from the input description. Node Switching Capacitance (NSC) of a sub-component (such as AND plane) in a PLA is the average capacitance switched by a node in the sub-component, when the node undergoes a power consuming transition ( $0 \rightarrow 1$ ). Power characterization involves extracting NSC equations for different sub-components as a function of input size, output size and number of terms. Prototype PLAs whose are employed to derive NSC equations for a given technology. The input description is modified for power simulation by adding NSC equations with dependent variables instanced to the controller's parameters. For a given input sequence, the modified VHDL description is simulated to estimate the total power consumption. Experimental Results are obtained with average estimation error of 10.48 % with a minimum error of 0.19% and maximum error of 21.90%.

## 1 Introduction

Many high-level power estimation techniques have been developed recently [1, 2, 3]. This paper proposes a register-transfer level (RTL) power simulation technique for PLA based controllers. It is based on a simple PLA power characterization procedure, which can be extended to other macro-cell implementations of controllers.

Automated synthesis of controllers has become a common practice. A typical controller synthesis tool accepts controller description and attempts to optimize by performing optimizations such as merging equivalent states and removing unreachable states. The reduced controller is then realized in a target implementation style such as using PLAs and ROMs. We present a power estimation technique based on simulation of the behavioral specification of the controller.

A typical CMOS PLA structure consists of several types of macro-cells stitched together by a tile placement procedure. Our power estimation technique consists of two steps: (1) *Power characterization* of the target PLA implementation style by deriving equations, known as NSC equations. These equations compute the average switching capacitance of a node in four PLA sub-components as a function of input size, output size and number of terms. Prototype PLAs of varying complexity are used in the characterization step. (2) *Power model* is developed by modifying the input description by adding the NSC equations. It is simulated using a given input sequence and total power consumed is estimated. In this work, we

demonstrate the technique for VHDL descriptions.

Experimental results for a number of controllers are obtained with an average estimation error of 10.48% with a maximum error of 21.90% and a minimum error of 0.19%.

## 2 Previous Work

Most proposed controller power estimation techniques operate at the gate and layout levels of abstraction [4]. Landman and Rabaey proposed an architectural power analysis technique using an Activity-Based Control (ABC) model of finite state controllers. It is implemented in Stochastic Power Analysis (SPA) tool. This approach depends on the identification of two types of parameters:

(1) *Target-independent* parameters capture the effect of the controller structure on power consumption irrespective of its implementation style. For any controller, *complexity* (for eg. input-size, output-size) and *activity* denote two classes of target-independent parameters.

(2) *Target-dependent* parameters capture the effect of implementation style on power consumption. Capacitance model and capacitance coefficients are two such parameters. The capacitance model for an implementation style describes how the average capacitance switched during a single clock cycle scales with input-size, output-size, number of terms. A set of capacitance coefficients are derived which characterize the technology. Random prototype controllers in the target style are generated and simulated using input streams of varying signal probabilities to extract the coefficients.

We do not require explicit capacitance model as the structure of the controller is already provided in the input description. We define NSC of a subcomponent as the average capacitance switched by a node (for eg. a physical transistor in AND plane) when a power consuming transition takes place. The key difference between NSC values and capacitive coefficients is that a NSC value is associated with a node in a sub-component, while a capacitive coefficient is associated with a sub-component. Thus our characterization procedure is more detailed. NSC values are extracted using a characterization procedure and characterize a target technology completely. The main differences between the characterization procedures are: (1) Landman and Rabaey uses "random" prototype controllers with varying complexity, whereas we use prototype PLAs designed in a non-random fashion; (2) Landman simulates the prototype controllers with inputs of varying signal probabilities, while in our work, we need to simulate any prototype PLA with only a specific input vector sequence without keeping *activity* in mind. The activity factor is taken care of during the simulation.

In SPA, the number of min-terms is estimated without taking into account the encoding style, as this information is not available. This could give rise to serious

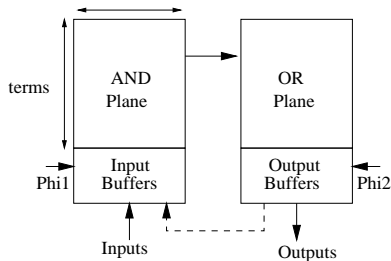


Figure 1: PLA Block Diagram

over/under estimate of the number of min-terms (as it is heavily dependent on the state encoding style), We take the state encoding style into account in the estimation of min-terms. Logic optimization is carried out assuming the user-specified state-encoding style and the resulting number of distinct terms is taken as the estimate of the number of terms.

In SPA, the ABC activity statistics of a controller are gathered through functional simulation. While SPA performs functional simulation *before* estimation, we estimate *during* functional simulation.

### 3 Power Characterization of PLAs

A PLA provides regular structure for implementing combinational and sequential logic functions. The basis for a PLA is sum of products form of representation of binary expressions.

A PLA has the following parameters : set of input variables ( $\mathcal{I}$ ), set of output variables ( $\mathcal{O}$ ), and set of product terms ( $\mathcal{T}$ ). We refer  $\mathcal{I}, \mathcal{O}, \mathcal{T}$  as the *PLA parameters*. Let the 3-tuple  $(\mathbf{I}, \mathbf{O}, \mathbf{T})$  represent a PLA with parameter values,  $|\mathcal{I}| = \mathbf{I}$ ,  $|\mathcal{O}| = \mathbf{O}$ , and  $|\mathcal{T}| = \mathbf{T}$ .

Figure 1 shows the block diagram of a typical PLA implemented as AND-OR structure. We can identify four sub-components namely (1) AND plane; (2) OR plane; (3) Input Buffers; and (4) Output Buffers. The *dimensions* of the four sub-components are: AND plane ( $\mathcal{I} \times \mathcal{T}$ ), OR plane ( $\mathcal{O} \times \mathcal{T}$ ), Input Buffers ( $\mathcal{I}$ ), and Output Buffers ( $\mathcal{O}$ ).

For a PLA implementation, the clocking scheme determines how a set of inputs are processed. In this work, we assume a two phase non-overlapping clocking scheme for dynamic precharged PLAs.

**Characterization Procedure** Let *Node Switching Capacitance* (NSC) of a sub-component in a PLA be the average capacitance switched by a node in the sub-component, when the node undergoes a power consuming transition (from logic 0 to logic 1).

The average switching capacitance of a node scales up with the dimensions of the sub-component. For example, in the AND plane, the length of vertical poly lines increases with the number of product terms. Thus the capacitance switched due to a  $0 \rightarrow 1$  event on an input line increases with increasing dimensions of the sub-component. *Power Characterization* extracts this scaling effect in the form of NSC equations.

The characterization procedure has the following three steps:

#### Step I : Generation of prototype PLAs

In this step, prototype PLAs with varying sizes of  $\mathcal{I}$ ,  $\mathcal{O}$  and  $\mathcal{T}$  are generated, whose program table is known *a priori*. A  $(\mathcal{I}, \mathcal{O}, \mathcal{T})$  PLA is synthesized using a set of boolean equations satisfying the following conditions :

- (1) the number of output variables is  $|\mathcal{O}|$
- (2) the number of input variables is  $|\mathcal{I}|$ , and
- (3) the number of terms in its PLA implementation is  $|\mathcal{T}|$ .

To ensure  $|\mathcal{T}|$  terms, we generate a set of simple irredundant boolean equations and synthesize the prototype PLA.

The parameters  $\mathcal{I}$ ,  $\mathcal{O}$  and  $\mathcal{T}$  are varied for the desired ranges, and logic equations are generated satisfying the above properties. The prototype PLAs so synthesized are combinational in nature. As we are interested in the average capacitance switched by a node, a combinational circuit gives us better control in switching a specific node in the PLA.

#### Step II : Extraction of NSC values in a prototype PLA

For a given prototype PLA, an input vector sequence is applied to switch nodes in the PLA. Capacitance switched by each node(s) is observed, and NSC for all sub-components is computed. We used IRSIM-CAP, a modified version of IRSIM switch level simulator for improved capacitance measurements.

For any PLA the length of the input sequence is reasonably small. The PLA characterization procedures in the previously proposed controller power estimation techniques [1, 6] involve simulation with a long stream of inputs until the consumed power converges. Compared to these approaches, the proposed approach is less time-consuming.

#### Step III : Derivation of NSC equations in PLA sub-components :

For varying PLA parameters, the raw NSC data obtained in the previous step is used to derive a set of NSC equations, by using linear and non-linear curve fitting techniques such as least-squares regression. The procedure yields the following set of NSC equations:

$$\begin{aligned} C_{phi1} &= C_1 * |\mathcal{I}| \\ C_{phi2} &= C_2 * |\mathcal{O}| \\ C_{ib} &= C_3 \\ C_{ob} &= C_4 \\ C_{in} &= C_5 * |\mathcal{I}| + C_6 * |\mathcal{T}| \\ C_{out} &= C_7 * |\mathcal{O}| + C_8 * |\mathcal{T}| \end{aligned}$$

where  $C_1, C_2, \dots, C_8$  are technology dependent constants.

### 4 Power Estimation of a Controller by Simulation

The power estimator accepts a behavioral VHDL description of the controller and technology NSC equations as an input and goes through the following tasks: (a) estimates target PLA parameters; (2) develops a power model; and (3) estimates power consumption. The PLA that would be synthesized from an input controller description, is referred to as “target PLA” of the controller. Firstly, the target PLA’s parameters are estimated. Using the estimated PLA parameters, an RTL power model of the controller is developed by modifying the input description. Power accumulation code is introduced to model power dissipation in different portions of the target PLA. The power model is simulated using user-specified input vectors to obtain the estimate of power consumption for the set of input vectors.

The RTL power model is generated automatically from the VHDL behavioral specification of the controller. Conditional statements are used to model a state in the controller’s state transition diagram. On the rising edge of first clock phase, the outputs are evaluated and on the rising edge of second clock phase, the inputs are evaluated and the next state of the controller is determined. Variations in functional modeling are allowed as long as the description adheres to the same clocking scheme used by the target PLA.

Power Model of the input controller is obtained by adding *power accumulating code* to the functional code.

We identify the statements in the functional code which correspond to “node”s in different sub-components of the target PLA. Whenever a power consuming event (i.e  $0 \rightarrow 1$  transition) occurs on a node, then the NSC of that node is accumulated. In order to compute NSC of different nodes, we need to know the target PLA’s parameters.

**Estimation of target PLA parameters :** In the port declaration, the ports of type in(out) are the primary inputs (outputs) of the controller, and correspond to the parameter  $\mathcal{I}$  ( $\mathcal{O}$ ) of the target PLA. The estimation of  $\mathcal{T}$ , is difficult as it depends on the state encoding scheme used (such as one-hot encoding, random encoding) during the logic optimization of the output and next state bit functions. To our knowledge, there exists no estimator, which, for a given set of logic equations, can estimate the number of terms resulting in the set of minimized equations.

A set of boolean equations is derived from the VHDL input and is fed to a logic optimizer such as espresso or misII [5]. The number of distinct terms in the set of minimized equations is taken as the PLA parameter,  $\mathcal{T}$ .

**Estimation of NSC values of target PLA :**

For a given target PLA, let  $C_{phi1}$ ,  $C_{phi2}$ ,  $C_{in}$ ,  $C_{ib}$ ,  $C_{out}$ , and  $C_{ob}$  be the node switching capacitances of phi1, phi2, input node, input buffer node, output node, and output buffer node, respectively, for a given set of target PLA parameters.

**Power Modeling :**

The total power consumed to process an input combination is the sum of power consumed by clock lines and power consumed in AND plane, OR plane, input and output buffers.

**Clock lines** By definition,  $C_{phi1}$  is the average capacitance switched in the target PLA, whenever there is a  $0 \rightarrow 1$  event on phi1 node. This can be modeled in the VHDL description by adding the following fragment of VHDL code:

```
-- phi1 clock power consumption
IF(phi1 = '1' AND NOT phi1'STABLE)
  THEN
    total_power := total_power + C_phi1;
  ENDIF;
```

The conditional expression checks for a power consuming transition and if it evaluates to true, then the variable “total\_power” accumulates the NSC of phi1 node, due to the current event on the node. Similar code fragment can be added for the phi2 node.

**AND plane and input buffers** The modeling of power consumption due to node activity in AND plane and OR plane is more involved. Consider an input column in a PLA with six product terms as shown in Figure 2. The input variable  $x$  appears in two product terms and its complement appears in three product terms. The value of  $x$  latched by the  $phi2$  clock. Whenever there is an  $0 \rightarrow 1$  event on the input line, it results in a power consuming event at the output of the inverter  $inv2$  as well as the two nodes on uncomplemented line. Thus, if we know the capacitances at the output of the inverter as well as the node capacitances, we can compute the power consumption due to a  $0 \rightarrow 1$  event on  $x$  line. We already have the NSCs of these nodes computed. Thus the total switching capacitance is  $(2 * C_{in} + C_{ib})$ . The following code fragment is added to the input description of the controller.

```
-- AND plane and input buffers
IF(phi2 = '1' AND NOT phi2'STABLE)
```

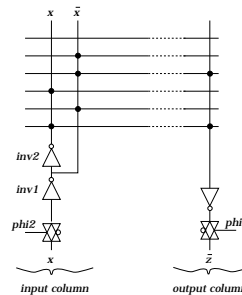


Figure 2: Input and Output Columns in a PLA with six product rows

```
THEN
  total_power := total_power + 2 * C_in + C_ib ;
ENDIF;
```

A  $1 \rightarrow 0$  event on  $x$  implies  $1 \rightarrow 0$  events at the output of  $inv1$  and three nodes on complemented line. Code fragment similar to the above code is added with  $(3 * C_{in} + C_{ib})$  as the switched capacitance.

For any input variable, we need to estimate the number of complemented and uncomplemented nodes appearing in the input column. This information is obtained from the minimized equations obtained after the logic minimization.

**OR plane and output buffers** Similar to an input column, an output column has nodes in the OR plane, with each node selecting a term. The terms so selected are combined to form an output line. In order to estimate the power consumed on an output line we need to know the number of nodes. This is again obtained from the optimized equations.  $C_{out}$  the NSC value of an output node is already computed.

**Precharging** In each clock cycle, a node in the OR plane is conditionally precharged depending on whether it has been discharged in the previous cycle when the inputs are evaluated. It is possible to model this conditional discharging of each node by adding one power modeling statement per node. Although this results in accurate modeling of power consumption, it slows down the simulation of the power model. From the minimized logic equations, we can obtain the total number of precharge nodes (transistors) appearing in the OR plane. In order to keep the model simpler, we assume that the number of precharge transistors are evenly distributed per output line, and half of them are precharged in every clock cycle. Thus, if the total number of precharge transistors is  $N_p$ , then the total capacitance switched due to precharging in a clock cycle, is given by  $0.5 * N_p * C_{precharge}$ .

## 5 Results

Power estimation results for the controllers of the following six designs are presented : (1) Newton-Raphson Divider; (2) Speech Recognition System; (3) 16-bit Microprocessor ; (4) Traffic Light Controller (TLC); (5) Decompression Chip ; and (6) Find Chip.

Experimental results for examples 1-3 are presented by Landman in [1]. For these examples, the estimation results obtained are compared with those reported in the above work. In [1], the power estimates are for  $1.2\mu$  technology and a supply voltage of 1.2V. While in this work, the estimates are for  $2\mu$  technology and a supply voltage of 5V. Examples 4-6 are ASIC designs synthesized by using a high level synthesis system known as Profile Driven Synthesis System [2], developed at University of Cincinnati. The synthesized controller is a hierarchical structural description consisting of local controllers. There are five

types of local controllers : (1) Top-level (Global) module; (2) Process module; (3) Loop module; (4) Subprogram module; and (5) Wait module.

TABLE I  
Pertinent Data of Example Controllers

Design	I	O	S	Lines in	Terms in
				power Model	target PLA
NR Divider	3	26	19	338	26
SR System	3	32	100	1781	95
Microprocessor	4	21	13	512	27
TLC_Global_FSM	5	11	14	396	20
TLC_Process_FSM	12	32	35	781	76
TLC_Wait_FSM	6	24	11	580	52
Decompress_Global_FSM	4	11	9	404	20
Decompress_Process_FSM	5	10	8	345	13
Decompress_Wait_FSM	10	10	46	345	13
Find_Global_FSM	4	16	15	504	21
Find_Process_FSM	6	36	13	336	25
Find_Loop_FSM	3	24	14	441	18
Find_Wait_FSM	3	26	21	568	27

Table I shows the relevant data of the controllers. TLC and Decompress have three local controllers, while Find has four local controllers. The PLA implementation is performed using random state encoding for each controller.

TABLE II  
Comparison of ABC Technique and NSC based technique

Design	Average % Estimation Error	
	ABC	NSC based
	Technique	Technique
NR Divider	6.0	4.8
SR System	12.6	10.6
Microprocessor	2.5	1.9

Table II compares the estimation results presented in [1] with our estimation results. As the target technology and the supply voltage parameters differ in both works, we could not compare the exact power values. Therefore, we are comparing the power estimation errors.

TABLE III  
Estimated and Actual Power Values for Three Examples

Design	Est.	Actual	%
	Power mW	Power mW	Error
NR Divider	4.02	3.83	4.9
SR System	3.88	4.34	10.6
Microprocessor	1.33	1.35	1.9
Average Error			5.7

Table III shows the comparison of actual and estimated power values for  $2\mu$  technology, supply voltage of 5V and at a frequency of 2.5 MHz. The actual power numbers are obtained using IRSIM-CAP on the switch level models extracted from the design's layout implementation. For a given number of vectors say  $N_v$ , let the total capacitance switched be  $C_{total}$ . Then the average power of the design is  $(0.5 * C_{total} * V_{supply}^2 * f) / (N_v)$ .

Table IV shows the comparison of actual and estimated power values for the local controllers of TLC, Decompression Chip and Find Chip. The average power is computed

by the formula presented above. The total number of vectors applied for each example, i.e.,  $N_v=500$ . The average estimation is 11.90% with a minimum error of 0.19% and maximum error of 21.90% .

TABLE IV  
Comparison of Estimated and Actual Power Values

Design	Leaf Controller Label	Est.	Actual	% Error
		Power ( $\mu$ W)	Power ( $\mu$ W)	
TLC	Global_FSM	525	526	0.19
	Process_FSM	2,900	3,334	13.00
	Wait_FSM	885	1,055	16.11
Decompress	Global_FSM	400	426	5.98
	Process_FSM	253	207	21.90
	Wait_FSM	220	247	10.93
Find	Global_FSM	423	372	13.84
	Process_FSM	928	823	12.71
	Loop_FSM	439	425	3.47
	Wait_FSM	108	89	20.94
Average Error				11.90

## 6 Conclusions

In this work, we presented a new power estimation technique for PLA-based controllers. It is based on a simple but efficient power characterization of PLAs. The regular structure of PLA implementation is exploited to obtain node switching capacitance equations for a given technology.

## Acknowledgments

We thank Paul Landman with Texas Instruments and Jan Rabaey with University of California, Berkeley, for providing us with the three controller examples presented in this work.

This work is done at the University of Cincinnati and is supported in part by the Solid State Electronics Directorate of the Wright Laboratory of the US Air Force under contract number F33615-91-C-1811 and by the Advanced Research Projects Agency HPCC program monitored by the Federal Bureau of Investigation under contract no. J-FBI-93-116.

## References

- [1] Paul E. Landman and Jan M. Rabaey, "Activity-Sensitive Architectural Power Analysis for the Control Path", *1995 International Symposium on Low Power Design*, pp 93-98, April 1995.
- [2] Nand Kumar, Srinivas Katkoori, Leo Rader and Ranga Vemuri, "Profile-Driven Behavioral Synthesis for Low Power VLSI Systems", *IEEE Design & Test of Computers*, pp. 70-84, Fall 1995.
- [3] Srinivas Katkoori and Ranga Vemuri, "Architectural Power Estimation Based on Behavioral Profiling", To appear in the *Special Issue on Low Power Design, Journal on VLSI DESIGN*, 1996.
- [4] F.N.Najm, "A Survey of Power Estimation Techniques in VLSI circuits", *IEEE Transactions VLSI Systems*, vol. 2, no. 4, pp. 446-455, January 1995.
- [5] R. Brayton, G. Hachtel, C. McMullen, and A. Sangiovanni-Vincentelli, *Logic Minimization Algorithms for VLSI Synthesis*, Kluwer Academic Publishers, 1984.
- [6] Srinivas Katkoori, Nand Kumar and Ranga Vemuri, "High Level Profiling Based Low Power Synthesis Technique", *Proceedings of International Conference on Computer Design 1995*, Austin, October 2-5, pp. 759-765, 1995.