A Methodology for High Level Power Estimation and Exploration

Vamsi Krishna and N. Ranganathan
Center for Microelectronics Research
Department of Computer Science and Engineering
University of South Florida
Tampa, FL 33620
{krishna, ranganat}@csee.usf.edu

Abstract

Effective power reduction can be achieved at higher levels of design abstraction. A number of such techniques have been proposed for power optimization in the literature. These techniques use RT level templates which characterize the area, delay and power of the design. The templates are based on some knowledge of the logic block such as the number of nodes, levels and their interconnections. Methods which model the power consumption of a logic block whose internal details are not known are desirable to explore trade-offs early on in the design cycle. Recently, lower bounds for switching activity at the gate level based on decision theory have been proposed by the authors. This has been extended to derive the average switching activity of a module based solely on its functionality. The experimental results on ISCAS '85 benchmark circuits indicate that the approach gives reasonably accurate estimates at low computational cost. In this paper, we use the RT level estimates for power exploration at the behavioral level for various high level synthesis benchmarks. The experimental results show that appropriate design decisions can be taken at the high level to reduce the cost of redesigning which would be incurred if committed to a particular circuit structure.

Keywords: *High Level Designs, Power Estimation, Low Power Designs, Switching Activity*

1 Introduction

Low power designs can be realized at various levels of the design cycle. Therefore, accurate estimation of power consumption at each level is necessary for the synthesis of these designs. The increasing demand for personal computing devices and wireless communication equipments with real time applications has resulted in the need for designing circuits with low power consumption. Until recently, designers were mainly concerned with area and throughput as the important design parameters. The addition of power as a third parameter to the design search space has consequently lead to the exploration of the tradeoffs between area, delay and power.

At higher levels of abstraction, the designers have a wide choice for implementing a particular circuit. Optimizations performed at this level have shown to be very effective in power reduction. Accurate power estimation techniques are necessary to guide the optimization procedures. Behavioral level techniques depend on some RT level template which characterizes the area, delay and power. To address the issue of power characterization, two approaches [1] have been developed: *bottom-up and top-down methods.* In the bottom-up approach, the switching activity is accurately characterized by means of profiling the behavioral description or by direct simulation. The bottom-up approaches are library based and depend on the power consumption characterization of the cells in the library set. Also, designers use libraries from different sources and hence a library independent approach for power analysis is very much desirable.

In an effort to facilitate a library independent behavioral power estimation method, the top-down approach was proposed. In the top-down methods, no information is available about the module structure. Hence the main issue is estimation of the average circuit switching activity and its complexity. Two methods have been proposed in [2, 3] which relate the switching activity to input-output entropy and the functionality. Recently, a decision theoretic approach for characterizing the switching activity at the gate and RT level was proposed in [4]. In this paper, we use the RT level estimates to guide the power optimization at the behavioral level. Specifically, we show that the decision theoretic estimates serve as good performance measures during behavioral transformations for low power designs.

0-8186-8409-7/98 $10.00

The paper is organized as follows. Section 2 describes the related work. The RT level switching activity modeling is briefly shown in Section 3. The use of the RT estimates in power exploration is described in Section 4. Section 5 shows the experimental results on various high level synthesis benchmarks. Conclusions and future work are given in Section 6.

2 Related Work

Many attempts have been made for power estimation and optimization at the higher levels of design abstraction. An efficient technique for architectural power estimation was proposed in [5]. They develop capacitance models for different modules by differentiating the MSBs and LSBs switching activities. The LSBs perform as uncorrelated noise and the MSBs are strongly correlated. The power model takes into account the LSB and MSB capacitance values. The switching capacitance values for various input statistics are stored in tables and are invoked in the form of look-up tables. This gives very accurate results but the characterization process is very tedious and cell library dependent.

The work in [6, 7] propose heuristics and ILP techniques for performing scheduling, module allocation and clock selection. They use a simulation based method to consider the inter and intra iteration effects on the switched capacitance. In [8], profiling is used to obtain the average switching activity of datapath, controllers and interconnects. Typical statistics used for analysis are the number of operations, number of accesses of a given operation and I/O operations. Analytical equations are then built using these statistics and used in behavioral synthesis for low power. A similar approach is described in [9] where a predetermined value of the average switched capacitance obtained from simulation is assumed.

A cycle accurate power macro-model is developed in [10]. This is important since it gives the power consumed in every cycle. The macro-models are generated by taking into account the structure and functionality of the circuit and hence are very accurate. They also propose a method of generating macro-models by assuming an initial equation with many variables and then pruning them by simulating with different input sequences. This is less accurate but is very fast and requires no user intervention. Another technique to generate a power macro-model for RTL circuits is presented in [11]. The approach accounts for glitches present in the circuit. But the number of variables in the macro-model is high. The work in [1] proposes a look-up table based macro-model. The look-up table consists of the average input signal probability and the average input and output transition densities. From the look-up table, power can be estimated for any input statistics.

High level transformations can be applied to optimize a design for area, speed and power. The approach described in [12] applies various transformations to reduce power in data dominated circuits. Concurrency is introduced by means of parallelism and pipelining and this improved throughput is used reduce the supply voltage. The overall power decreases due to the quadratic effect of power on the supply voltage. Other transformations such as loop interchange and input reordering are used in [13] to reduce the switching activity in functional units. Precharacterized cells guide the transformations in [12, 13].

In contrast to the above bottom-up approaches, three top-down methods for power analysis have been proposed which do not depend on the internal details of a circuit. In [2, 3], entropy and informational energy have been used as measures of switching activity. It is shown that the switching activity is upper bounded by a function of entropy. This method is useful in computing the average switching activity of a RT level module. A method to obtain the entropy for dataflow graphs has also been described in [3]. A simplified model for the RT level structure has been used which characterizes it in terms of NAND gates. Although the entropy based method in [3] describes application to dataflow graphs, no experimental results are given. In [2], the average switching activity is estimated in terms of the input and the output entropy and the number of inputs and outputs. This information is obtained from a high level description of the module. The work in [4] proposes a decision theoretic method to compute lower bounds for the switching activity at the gate level. The switching activity is modeled as the decision error of an abstract two class problem. The proposed lower bounds are then used for switching activity characterization at the RT level. The RT level estimates are based on the input-output behavior and the functionality of a module. Such high level, library independent estimation techniques are desirable for redesigning at an early stage.

In this paper, we use the RT level estimates proposed in [4] for power exploration at the behavioral level for various high level synthesis benchmarks. The experimental results show that appropriate design decisions can be taken at the high level to reduce the cost of redesigning which would be incurred if committed to a particular circuit structure.

3 High Level Switching Activity Characterization

In this section, we briefly describe the average switching activity characterization at the RT level using decision theoretic arguments. The details of the method can be found in [4].

3.1 Decision Theory

Decision theory is concerned with guessing the unknown nature of an observation, for instance, deciding on the median of random observations from a given set. If the observation is $X = x$, then its unknown nature, $Y = y$, is called a *class*. A *classifier* is a function $g(x) : \mathcal{R}^d \rightarrow \{0, \cdots, Q-1\}$ that maps the d-dimensional input space into one of the Q classes. A classification error occurs on X if $g(X) \neq Y$, and the *probability of error* for g is given by

$$L(g) = P(g(X) \neq Y) \tag{1}$$

For a two class problem, if $\eta(X) = P(Y = 1|X)$ is the posterior (conditional) probability of class-1 given the observation X, the Bayes classifier, g^*, is defined as

$$g^*(X) = \begin{cases} 1, & if\ \eta(X) > \frac{1}{2} \\ 0, & otherwise \end{cases} \tag{2}$$

The Bayes error L^* can be shown to be [14]

$$L^* = L(g^*) = \mathbf{E}(min(\eta(X), 1 - \eta(X))) \tag{3}$$

where $\mathbf{E}$ denotes the expected value. Another classifier is based on the k nearest neighbor ($k - NN$) rule. For the $1 - NN$ rule, the asymptotic classification error is given by [15]:

$$L_{NN} = L(g_n) = \mathbf{E}(2\eta(X)(1 - \eta(X)) \tag{4}$$

Figure 1 shows the correspondence between Markov chains and decision theory. In [4] it is shown that the switching activity is a classification error lower bounded by L^*. Further, it is improved to obtain a tighter bound L_1, which is based on the 1-nearest neighbor classification error. L^* and L_1 are functions of the signal and conditional probabilities of a line.

3.2 RT Level Switching Activity Characterization

The lower bounds L^* and L_1 are used for obtaining the average activity of a RTL module. Such a high level power exploration capability is essential to explore design trade-offs at an early stage of the design cycle. The module is considered as a black box and its average switching activity is determined based on the information available about the inputs and the outputs only. Let $\mathcal{L}^*_{avg}$ denote the average L^* per node and $\mathcal{L}_{1avg}$ denote the average L_1 per node in a circuit. It is shown in [4] that

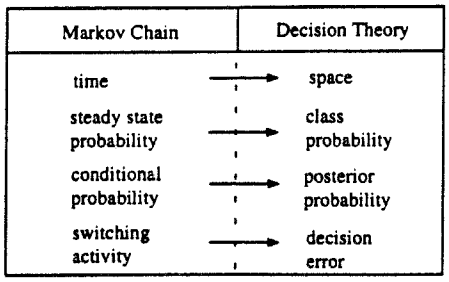

Figure 1: Mapping of Markov chain to decision theory

$$\mathcal{L}^*_{avg} \approx \frac{2}{(n+m)} \left[\frac{4\mathcal{L}^*_{out} + \mathcal{L}^*_{in}}{5} \right] \tag{5}$$

where, $\mathcal{L}^*_{in}$ is the sum of the L^*s of the input nodes and $\mathcal{L}^*_{out}$ is the sum of the L^*s of the output nodes. Similarly,

$$\mathcal{L}_{1avg} \approx \frac{2}{(n+m)} \left[\frac{4\mathcal{L}_{1out} + \mathcal{L}_{1in}}{5} \right] \tag{6}$$

Clearly, the estimates of the switching activity at the RT level using the above equations depend only on the number of inputs and outputs and the L^*, L_1 at the input and output. $\mathcal{L}^*_{in}$ and $\mathcal{L}^*_{out}$ can be obtained by a quick functional simulation. Hence, this approach gives a fast estimate of the average switching activity at the RT level. Experimental results for the RT level switching activity estimates for ISCAS '85 circuits are presented in [4]. The technique is simple and produces accurate estimates.

The accuracy of the estimates depend mainly on the tightness of the measures used to characterize the switching activity. Figure 2, part (i) shows the graphical comparison of the two lower bounds with the actual switching activity, S, and the entropic upper bound given in [2, 3], namely $\frac{1}{2}H(x)$. We define the two errors to be $E_1 = |\frac{1}{2}H(x) - S|$ and $E_2 = |L_1 - S|$. Figure 2, part (ii) shows that E_2 is much smaller than E_1 for certain values of the signal probability p. For the other

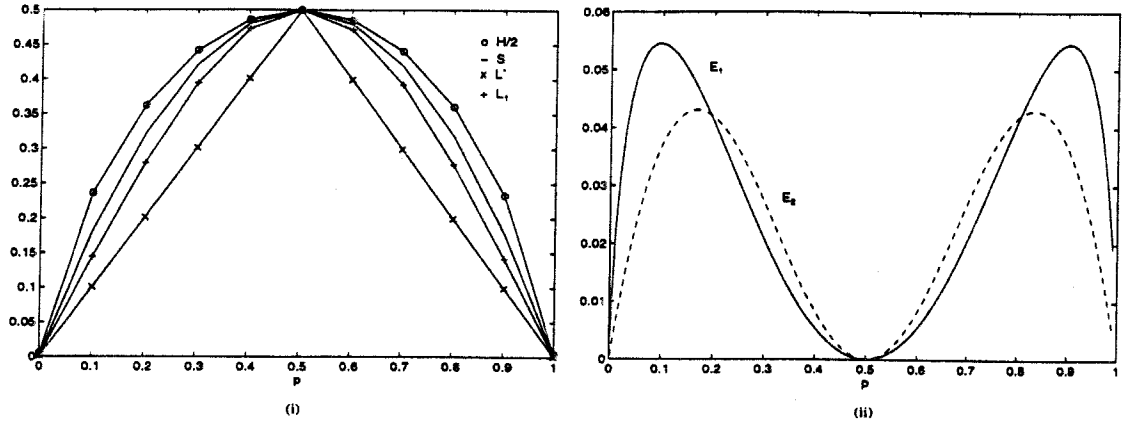

Figure 2: (i) Theoretical comparison (ii) Error plot

values of p, the difference between E_1 and E_2 is very small. This suggests that L_1 can be used as a better estimate of S as compared to $\frac{1}{2}H(x)$ depending on the probability distribution and the application at the RT level of abstraction.

4 Design Space Exploration

In the previous section, the average switching activity of a RT level module was derived. For a power estimate, the average switching activity should be multiplied by a measure of the total module capacitance. This figure is called the *switched capacitance* which contributes to the power dissipation. Given a module with n internal lines, the total switched capacitance is given by

$$SC_{total} = \sum_n C_n.S_n \approx S_{avg} \sum_n C_n$$

where, C_n is the capacitance of line n with its switching activity S_n and the summation is performed over all the lines in the module and S_{avg} is the average switching activity per node. Using the results from the previous section, the estimated switched capacitance can be written as

$$SC_{est} \approx \mathcal{L}_{1avg}. \sum_n C_n = \mathcal{L}_{1avg}.C_{total} \qquad (7)$$

where C_{total} is the estimated total capacitance of the module.

Several works have been reported which predict the total capacitance value given only the high level description of a circuit [12, 16]. We use the model developed in [12] for obtaining the power estimate. For example, the capacitance of a N bit comparator is $181N$. Hence, for a 8 bit comparator, the total capacitance is given by $C_{total} = 181 * 8 = 1448fF$. Performing functional simulation and using Equation 6, the average switching activity, $\mathcal{L}_{1avg}$ is 0.237. Hence the average switched capacitance is 343.18 fF.

We explain the power exploration methodology with the help of a high level synthesis benchmark circuit, the IIR Filter. Different transformations can be applied to get different implementations of the same design to consider trade-offs among area, delay and power. We use the HYPER system [17] to apply the various transformations at the behavioral level. The transformations used are i) Algebraic law ii) Dead code elimination iii) Common sub-expression elimination and iv) Constant multiplication. The first three techniques are commonly used in software compilers and are grouped together and denoted by $I1$. The fourth transformation converts multiplications into shift-add operations and hence is a very useful transformation in DSP applications. This is denoted as $I2$. We apply both the transformations $I1$ and $I2$ to the IIR filter circuit. The number of control steps is set to 16. The schedules obtained after $I1$ and $I2$ are shown in Figure 3. The schedule shows the various operations in each control step. For example, in Figure 3, part(i), in control step 2, a multiplier and an adder are used. The values of $\mathcal{L}_{1avg}$ for each functional unit are obtained by behavioral simulation. The estimated average switched capacitance per cycle using Equation 7 is $146.80pF$ after applying $I1$ and that after $I2$ is $17.3pF$. This suggests that the schedule (ii) is better as far as power consumption is concerned. To verify

this, the circuit was synthesized using HYPER and the power consumption was estimated using the tool SPA [5]. As mentioned in the related work section, SPA gives very accurate results. It was observed that the switched capacitance per cycle for the functional units of implementation (i) is $171.58pF$ and that of implementation (ii) is $20.6pF$. This indicates that the proposed estimation technique gives accurate results and allows quick design space exploration without having to consider the internal details of the final implementation.

5 Experimental Results

In this section, we present the experimental results for various benchmark circuits, namely, the FIR filter, the WAVELET filter and the LATTICE filter. Different transformations using HYPER are applied on the original flowgraphs and are classified into two groups denoted by $F(j), L(j)$ and $W(j)$ for $j = 1, 2$ for the FIR, lattice and wavelet circuits respectively. The transformations are chosen in a random order to give different schedules for each circuit. The estimated switched capacitance per cycle after applying the transformations on all the circuits is shown in Table 1. From the table, we observe the following. The transformations $F1, L1$ and $W1$ cause less capacitance to be switched and hence result in reduced power consumption. The actual values obtained from SPA are also summarized in Table 1. As seen from Table 1, a similar conclusion can be drawn. Hence, the proposed top down power exploration methodology gives good insight into various design alternatives for trade-offs for power. This facilitates suitable design choices before committing to a particular implementation. The proposed approach estimates the power consumed by the functional units only. Typically, a circuit consists of interconnects and control circuit apart from the execution units. Similar high level characterization procedures should be developed for the interconnects and the controller to get accurate power consumption estimates at low cost. The technique proposed in this paper is suited for data path intensive architectures where the execution units consume more power as compared to the interconnects and the control logic.

6 Conclusions and Future Work

This paper presents a methodology for power estimation and exploration at the higher levels of design abstraction. Such a capability is essential for redesign at an early stage of the design process. The power characterization process is cell library independent. The experimental results indicate that the proposed approach is suitable for data path intensive architectures and allows quick design space exploration at low cost. Future work includes developing similar characterization methods for the interconnects and the control logic.

References

[1] S. Gupta and F. Najm, "Power Macromodeling for high level power estimation," *Proc. of DAC*, 1997.

[2] F. Najm, "Towards a high level power estimation capability," *Proc, of Int. Symposium on low power design*, pp. 87-92, 1995.

[3] D. Marculescu, R. Marculescu, and M. Pedram, "Information theoretic measures for power analysis," *IEEE Transaction on CAD*, vol 15, no 6, pp. 599-610, 1996.

[4] V. Krishna, R. Chandramouli and N. Ranganathan, " Computation of lower bounds for switching activity based on decision theory," *Revised and resubmitted to IEEE Trans. on VLSI*

[5] P. Landman and J. M. Rabaey, "Architectural power analysis: The dual-bit type method," *IEEE Transaction on VLSI*, vol 3, no 6, pp. 173-187, 1995.

[6] A. Raghunathan and N. K. Jha, "Behavioral synthesis for low power," *Proc of ICCD*, pp. 318-322, 1994.

[7] A. Raghunathan and N. K. Jha, "An ILP formulation for low power based on minimizing switched capacitance during data path allocation," *Proc of ISCAS*, May, 1994.

[8] N. Kumar, S. Katkoori, L. Rader and R. Vemuri, "Profile driven behavioral synthesis for low power VLSI systems," *IEEE Design and Test of Computers*, vol 12, no 3, pp. 70-84, 1995.

[9] J. Zhu, P. Agrawal and D. D. Gajski, "RT Level power analysis," *Proc of ASP-DAC*, pp. 517-522, 1997.

[10] Q. Wu, C. S. Ding, C. T. Hsieh and M. Pedram, "Statistical design of macro-models for RT level power evaluation," *Proc of ASP-DAC*, pp. 523-528, 1997.

[11] A. Raghunathan, S. Dey and N. K. Jha, "Register transfer level estimation techniques for switching activity and power consumption," *Proc of ICCAD*, pp. 158-165, 1996.

[12] A. Chandrakasan, M. Potkonjak, J. Rabaey and R. Brodersen, "Optimizing power using transformation," *IEEE Trans on CAD*, vol 14, no 1, pp. 12-31, 1995.

[13] E. Musoll and J. Cortadella, "High level synthesis techniques for reducing the activity of functional units ," *Proc of Int. Symp. on Low Power Design*, pp. 99-104, 1995.

[14] Berger J.O, *Statistical decision theory and Bayesian analysis*,Springer-Verlag, NY, 1985.

[15] Cover T. and Hart P, "Nearest neighbor pattern classification", *IEEE Transactions on Information Theory*, Vol. 13, pp. 21-27, Jan 1967.

[16] M. Nemani and F. Najm, "High level power estimation and the area complexity of boolean functions", *Proc of Int. Symp. on Low Power Design*, pp. 12-14, 1996.

[17] J. Rabaey, C. Chu, P. Hoang and M. Potkonjak, "Fast prototyping of data path intensive architectures", *IEEE design and Test*, Vol. 8, pp. 40-51, Jan 1991.

Circuit	Transformations	$SC_{est}(pF)$	$SC_{SPA}(pF)$
FIR	F1	31.2	36.7
	F2	261.3	298.8
lattice	L1	128.2	160.34
	L2	228.2	201.22
wavelet	W1	23.4	31.6
	W2	88.8	109.07
IIR	I1	146.8	171.58
	I2	17.3	20.6

Table 1: Switched capacitance values for functional units

Functional Units Used

a-Multipliers : 1
b-Adders : 1
c-Subtractors: 1

control step	a	b	c
1	x		
2	x	x	
3	x	x	
4	x		x
5	x	x	
6	x		
7	x	x	
8		x	
9	x		x
10	x	x	
11	x	x	
12		x	
13			x
14		x	
15			x
16		x	

(i)

Functional Units Used: Shifters: 2, Adders:2, Subtractor:1

a-Shifter 1 d-Adder 2
b-Shifter 2 e-Subractor
c-Adder 1

control step	a	b	c	d	e
1	x	x			
2	x	x	x		
3	x	x	x		x
4	x	x	x	x	
5	x		x		x
6		x	x		x
7	x		x	x	x
8	x		x		
9				x	x
10		x			x
11				x	x
12					x
13		x			x
14				x	x
15					x
16				x	

(ii)

Figure 3: Schedule after transformations, (i) I1 and (ii) I2