Logic Transformation for Low Power Synthesis*

Ki-Wook Kim†, Ting Ting Hwang‡, C. L. Liu‡ and Sung-Mo Kang†

† *Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, IL 61801*

‡ *Department of Computer Science, Tsing Hua University, Hsin-Chu, Taiwan 30043*

Abstract

In this paper, we present a new approach to the problem of local logic transformation for reduction of power dissipation in logic circuits. Based on the finite-state input transition (FIT) power dissipation model, we introduce a cost function which accounts for the effects of input capacitance, input slew rate, internal parasitic capacitance of logic gates, interconnect capacitance, as well as switching power. Our approach provides an efficient way of estimating the global effect of local logic transformations in logic circuits. In our approach, the FIT model for the transitive fanout cells of a locally transformed subcircuit can be reused to measure the global power dissipation by varying the input probabilities of the transitive fanout cells. Local logic transformation is carried out based on compatible sets of permissible functions (CSPF). Experimental results show that local logic transformation based on CSPF using our cost function can reduce power consumption by about 36% on average without increase in the worst-case circuit delay.

1 Introduction

At the logic synthesis level, logic transformation has been proved to be an effective technique to reduce power consumption by restructuring a mapped circuit by way of permissible signal substitution or perturbation. Logic transformation techniques based on the logic perturbation method [1] are used to re-synthesize a combinational circuit [9]. In [7], a structural transformation technique is proposed based on *compatible sets of permissible functions* (CSPF) [5]. Recently, in [2], more comprehensive permissible functions, referred to as *set of pairs of functions to be distinguished* (SPFD) [10], are introduced to obtain better FPGA implementations with lower power dissipation.

In this paper we present a new approach to the problem of logic transformation for the reduction of power dissipation. The two key ideas in our approach are :

(1) signal lines for logic transformation are identified as a group, not one-by-one sequentially.

(2) a new power dissipation model is used to compute power consumption effectively so that a large number of logic transformation candidates can be examined.

2 Local Logic Transformation

Traditionally, logic transformation in a logic circuit is carried out by identifying signals with high switching activities and re-synthesizing the signals one at a time [2, 7, 9]. Such a sequential, greedy approach is very fast. However, the final solution depends on the order in which the signals are re-synthesized, and yet very little is known on how to determine an optimal ordering of these signals.

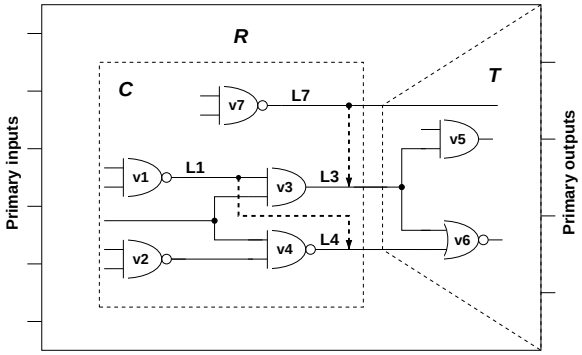

Figure 1: Local logic transformation example

Take the circuit in Figure 1 as an example. Suppose we can

(1) replace signal $L3$ by signal $L7$ and

(2) replace signal $L4$ by signal $L1$.

A sequential approach will replace the two signal lines one at a time. If (1) is carried out first, then because gates $v3$ and $v1$ will be deleted and signal $L1$ will no longer exist, (2) cannot be carried out. On the other hand, if (2) is carried out first, then (1) can be carried out subsequently. This example illustrates that a sequential approach will not be able to take into account the inter-dependency of signal lines.

On the other hand, for a non-sequential approach, when there are M modifiable signal lines and N candidate signals on average for each modifiable signal line, there will be N^M possible transformations. To consider all such transformations will be prohibitive in terms of computation time. To overcome such an exponential explosion in computational complexity, we propose a logic transformation scheme as follows:

(1) A subcircuit C of a logic circuit is identified.

(2) Structural transformations of the subcircuit C are carried out.

(3) For each structural transformation of C, compute the power consumption of the restructured circuit.

(4) Select the best structural transformation of C.

*Supported in part by National Science Foundation (NSF) under grant 1-5-31333 NSF MIP 96 12184

A significant advantage of such an approach is that we can try to identify a small subcircuit C the behavior of which is less dependent upon the signals outside of C. In this way, the number of possible structural transformations of C is relatively small and can be examined exhaustively. The details on how to identify a subcircuit C will be presented in Section 3.

To compute the power consumption of each restructured circuit in (3) could be extremely time-consuming because we need to recompute the power dissipation in the whole circuit. Note, however, that a structural transformation of C changes only the signal probabilities of the transitive fanout signals of C.

Let us partition the circuit into three parts as shown in Figure 1: a subcircuit C, the subcircuit T which contains all the transitive fanout cells of the cells in C, and the remainder of the circuit, R. Suppose a structural transformation is carried out to transform subcircuit C into C'. Clearly, the power consumed in subcircuit C' is different from that consumed in subcircuit C. The power consumed in subcircuit T will also be different because of changes in the input signals (or signal probabilities) to subcircuit T although the structure of T remains unchanged.

Therefore, the power gain ΔE corresponding to a transformation is given by

$$\Delta E = \Delta E(C) + \Delta E(T) \tag{1}$$

where $\Delta E(C)$ corresponds to the change in power consumption in subcircuit C (from C to C') which is referred to as *local power gain*, and $\Delta E(T)$ corresponds to the change in power consumption in subcircuit T which is referred to as *global power gain*. The global power gain is known to dominate the local power gain in many cases.

Clearly, the computational cost to recompute the power dissipation for every candidate transformation will be expensive [7], when a logic simulation is used to estimate the power consumed by a whole circuit. A power computation model which can take advantage of the structure invariant of the subcircuit T is required. We propose a new cost function based on the FIT power model which can overcome such computational difficulty while providing very accurate estimations on power consumption, as we shall see in Section 4.

3 Identification of a Subcircuit C

The identification of a subcircuit C is based on the notion of *direct* and *indirect dependency* of signals. Let L be a signal line. If L' is one of the candidate signals that can be used to replace L, then we say that signal line L is *directly dependent* on signal line L'. In Figure 1, $L3$ is directly dependent on $L7$.

To define the notion of indirect dependency, we introduce first the notion of *dominating fanin lines*. Let $\mathcal{L}$ be a set of signal lines. The dominating fanin lines of $\mathcal{L}$, denoted by $FI(\mathcal{L})$, are signal lines such that any transitive fanout path between a signal line in $FI(\mathcal{L})$ and a primary output line will include a signal line in $\mathcal{L}$. In Figure 1, $L1$ is in $FI(\{L3\})$.

Then, we say a signal line L is *indirectly dependent* on L' if L is directly dependent on any of the signal lines in $FI(\{L'\})$. In Figure 1, $L1$ is in $FI(\{L3\})$ and $L4$ is directly dependent on $L1$, so $L4$ is indirectly dependent on $L3$.

A subcircuit C can be defined conveniently by two sets F and G where F is the set of output signals (of the subcircuits) and G is the set of all the signals in the subcircuit. A set of subcircuit C's can be identified as follows:

Step 1: Set i = 1. Select a modifiable signal line L that is not included in any subcircuit C. Let $F_i = \{L\}$.

Step 2: If $F_i = null$, then stop.

Step 3: Let $G_i = F_i \cup FI(F_i)$.

Step 4: For all modifiable lines L_k, if L_k is directly or indirectly dependent on a line in G_i, then $F_i = F_i \cup \{L_k\}$ and $G_i = G_i \cup \{L_k\} \cup FI(F_i)$.

Step 5: Increase i by 1. Go to **Step 1**.

As a result, one or more subcircuits defined by F_i and G_i are identified. Each subcircuit C can be transformed independently. So, for a disjoint set of K subcircuit C's, there will be $N^{\frac{M}{K}} \cdot K$ possible transformations, assuming that modifiable lines are equally distributed among the subcircuits C's.

4 Power Model

We propose a finite-state input transition power model which enables us to determine the total power consumption in the circuit in three steps : cell characterization, subcircuit characterization, and input statistics application. One of the main features of the FIT model is the separation of input statistics application step from logic simulation step.

4.1 FIT Model for a Cell : Cell Characterization

Most of the power dissipated in CMOS logic is due to the dynamic power consumption P which consists of switching power and short circuit power [4]

$$P = C_L V_{dd}^2 f + k V_{dd} \tau \tag{2}$$

where C_L corresponds to the load capacitance, V_{dd} and f correspond to the supply voltage and the clock frequency, and τ corresponds to the input slew rate. The load capacitance consists of three components, namely

$$C_L = C_{int} + C_{wire} + C_{ext} \tag{3}$$

where C_{int} denotes the drain capacitance and the overlapping capacitance internal to the driver, C_{wire} denotes interconnection parasitic capacitance, and C_{ext} denotes the gate capacitances of the fanout nodes.

We characterized each cell by measuring the power consumption due to C_{int} and C_{ext} by using SPICE for varying input slew rates in order to consider the power consumption due to short circuit current. To account for the contribution of interconnection capacitances, interconnection capacitances C_{wire}, weighted by the number of fanout nodes, are provided as a parameter of power consumption of the whole circuit. Here, as in the case of the AUTOROUTE option in the conventional place and route tool, C_{wire} is assumed to

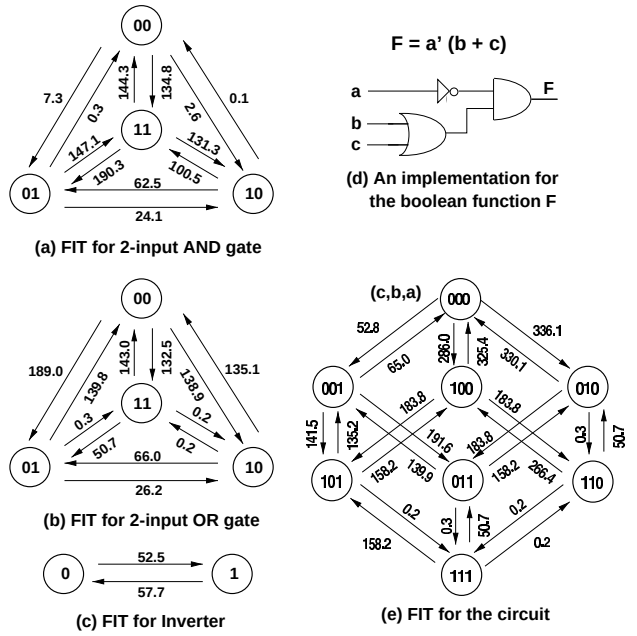

Figure 2: FIT model example

have a unit value per fanout, since the exact value for C_{wire} will not be available until placement and routing have been performed.

4.2 FIT Model for a Circuit : Subcircuit Characterization

The FIT model of a circuit contains complete information on power consumption for every input transition in the circuit. The FIT model for a circuit is constructed based on the FIT models for the individual logic cells in the library.

Figure 2(a), (b) and (c) shows the FIT model for a 2-input AND cell, a 2-input OR cell, and an inverter, respectively. The states in the finite-state transition diagram correspond to the input vectors for the cell. The weight, in μW, for each transition accounts for the power dissipation when an input transition takes place. For the boolean function $F = a'(b+c)$, and the implementation shown in Figure 2(d), the FIT model for the circuit can be constructed as in Figure 2(e) by logic simulation. (For brevity, we show only some of transitions in the FIT model.)

Note that the FIT model for a circuit takes into account delays in individual cells and consequently glitch activities in the circuit. In other words, the FIT model enables us to estimate the average power dissipation of combinational circuits with realistic delays.

4.3 Power Computation : Input Statistics Application

Once the FIT model for a circuit is built, we can easily compute the average power dissipation E in the circuit by applying the input transition probabilities

$$E = \sum_{i,j \in \Gamma} E_{ij} \cdot P_{ij} \tag{4}$$

where Γ is the set of input states, E_{ij} corresponds to power dissipation resulting from a transition from state i to state j, and P_{ij} corresponds to the transition probability from state i to state j.

4.4 Local and Global Power Gain

We are now ready to show how the power gain ΔE in Equation (1) can be computed for our power model.

The local power gain arises from reconfiguring a subcircuit C into C'. Meanwhile, the input transition probabilities for a subcircuit C remain unchanged. Therefore, the local power gain is computed as

$$\Delta E(C) = \sum_{i,j \in \Gamma} [E_{ij}(C) - E_{ij}(C')] \cdot P_{ij} \tag{5}$$

where $E_{ij}(C)$ and $E_{ij}(C')$ denote the power consumption resulting from a transition from state i to state j in subcircuits C and C', respectively. $E_{ij}(C)$ and $E_{ij}(C')$ are obtained by constructing the FIT models for C and C'. The computational cost to build the FIT model for every candidate of local logic transformation is not high since C is usually composed of a small number of cells.

On the other hand, the structure of the transitive fanout cells remains unchanged. Meanwhile, the transition probabilities will vary depending on new subcircuit C'. Therefore, the global power gain is computed as

$$\Delta E(T) = \sum_{i,j \in \Gamma} E_{ij}(T) \cdot (P_{ij} - P'_{ij}) \tag{6}$$

where P'_{ij} denotes the transitional probability observed at the input signal lines of subcircuit T when the subcircuit C is replaced by C'. The global power gains for all candidates for local logic transformation can be obtained by applying the corresponding P'_{ij} to the FIT model of subcircuit T. The key part is to note that the FIT model for a subcircuit T can be *reused* to compute the global power gain once the FIT model has been constructed. Consequently, we do not need to carry out a simulation for the subcircuit T for each candidate of the local logic transformation for C.

5 Local Logic Transformation Based on Permissible Function

The transformation methods we use to restructure the network include gate substitution, inverter insertion, and combined gate insertion based on permissible function [5]. The procedure for local logic transformation is as follows.

Step 1: Select a subcircuit C. Let F denote the set of output signals in subcircuit C.

1. Calculate the CSPF for all the signal lines in C.
2. Construct a FIT model for the transitive fanout cells of F.
3. For all candidate transformations for C, construct a FIT model for each transformation and compute the corresponding local power gain.
4. Select the best candidate.

Step 2: If there is no power reduction for all candidates, then stop.

Step 3: Go to **Step 1**.

In this procedure, in order to compute the power consumed by the subcircuits C and T, we need the exact transition probabilities corresponding to the spatio-temporal correlations on all signal lines in the logic network. However, even under a zero delay model, correct computation of the transition probability for any internal node is proven to be very time consuming. So, we compute the transition probability of a signal line based on both signal probability and switching activity of the signal line.

6 Experimental Results

FIT-based gain functions were implemented. Results for power minimization on CMOS combinational circuits using CSPF-based logic transformation were obtained.

First, each circuit in the benchmark set was optimized for area using SIS, and then mapped using the power driven technology mapper POSE [3], which implemented the low power mapping technique presented in [8].

Table 1 shows the results of power reduction for the MCNC benchmark circuits using logic transformation technique based on CSPF. The input transition probabilities are assumed to be given. Columns 2, 3 and 4 give the power, area and delay of the technology mapped circuits when they are optimized for minimum power using the POSE tool. The power, in mW, consumed by the circuit is measured by the FIT power estimator described in Section 4.

Columns 5, 6 and 7 correspond to the power, area and delay of the transformed circuits without delay constraints, whose values are normalized with respect to columns 2, 3 and 4, respectively. The average power consumption decreased by 38% on average. Gate elimination during the transformation contributes to a 15% area reduction on average.

Columns 8, 9 and 10 correspond to the power, area and delay of the transformed circuits with delay constraints, whose values are normalized with respect to columns 2, 3 and 4, respectively. The circuits consume 36% less power than the circuits mapped with the low power option. With the worst-case delay reduced by 5% on average, the transformed circuits require 15% less area than the circuits before transformation.

The column "TP Ratio" denotes the ratio of the *technology power* of transformed circuits with delay constraints to that of initially mapped circuits [6]. The larger the technology power, the more the on-chip functional events per unit time. The transformation results in a 22% technology power increase on average.

Figure 3 shows the relationship between power reduction and delay increase in the circuit `cc` of which delay increase is maximal among the benchmark circuits. Even for the circuit `cc`, the power reduction is up to 22% while the worst-case delay constraint is observed.

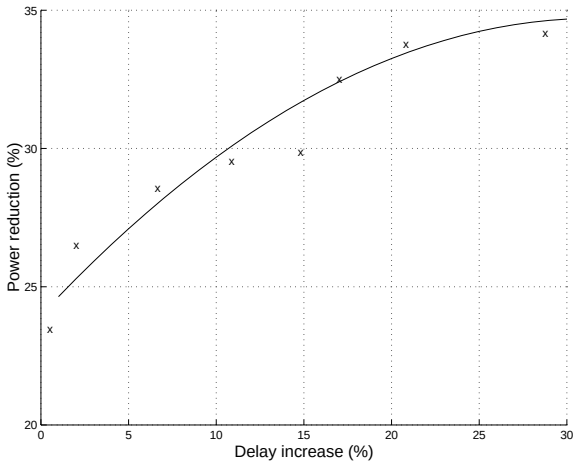

Figure 3: Power reduction vs. delay increase in the circuit "`cc`"

7 Conclusions

We presented an accurate and efficient cost function to account for local and global power gain based on the FIT power dissipation model. Power estimation based on the FIT power model involves three steps: construction of the FIT model for a cell, construction of the FIT model for the circuit, and application of input statistics to the FIT model.

The FIT model enables us to consider a group of candidate transformations at the same time. Simultaneous probing to select the best set of transformations enables us to account for the correlation between transformations, and thus provides better result than a greedy approach. Based on the dependency of signal lines, we presented a method to decide how cells are grouped to form a subcircuit C.

We implemented and used our gain function in local logic transformations based on CSPF. The CSPF-based local logic transformation produced 38% power reduction while the worst-case delay of the circuit decreased by 4% on average. With delay constraint, the power consumed by the circuit decreased by 36% on average.

References

[1] S. C. Chang, M. Marek-Sadowska, and K. T. Cheng. Perturb and simplify: multilevel boolean network optimizer. *IEEE Transactions on Computer-Aided Design*, 15(12):1494–1504, Nov. 1996.

[2] J. Hwang, F. Chun, and T. T. Hwang. A re-engineering approach to low power design using SPFD. In *Proc. of the ACM/IEEE Design Automation Conference*, June 1998.

[3] S. Iman and M. Pedram. POSE: Power optimization and synthesis environment. In *Proc. of the ACM/IEEE Design Automation Conference*, pages 21–26, June 1996.

[4] S. M. Kang and Y. Leblebici. *CMOS Digital Integrated Circuits: Analysis and Design*. McGraw-Hill, 2nd ed., 1998.

Circuit	Before transformation			After transformation w/o delay constraint			After transformation w/ delay constraint			TP Ratio
	Power	Area	Delay	Power	Area	Delay	Power	Area	Delay	
z4ml	4.35	115	16.6	0.37	0.53	0.93	0.37	0.53	0.93	2.03
cc	4.87	147	13.3	0.66	0.83	1.28	0.78	0.92	0.97	1.12
f51m	10.12	312	20.1	0.54	0.71	1.06	0.65	0.81	0.97	1.27
c8	13.06	413	17.2	0.77	0.82	0.99	0.77	0.82	0.99	1.23
apex7	14.84	586	29.7	0.66	0.87	0.91	0.66	0.87	0.91	1.26
cht	15.02	514	17.9	0.71	0.99	1.00	0.71	0.99	1.00	1.01
C432	15.28	543	59.7	0.65	0.85	0.99	0.65	0.85	0.99	1.19
alu2	28.21	862	70.9	0.52	0.84	0.81	0.52	0.84	0.81	1.47
x4	29.97	1064	27.5	0.71	0.93	1.00	0.71	0.93	1.00	1.08
C880	31.59	954	47.7	0.68	0.99	0.98	0.68	0.99	0.98	1.03
alu4	42.13	1617	80.5	0.71	0.86	0.90	0.71	0.86	0.90	1.08
C6288	166.23	4653	182.1	0.62	0.91	1.04	0.67	0.87	0.99	1.16
C7552	241.71	4237	106.4	0.64	0.97	0.97	0.64	0.97	0.97	1.06
Avg. ratio	1.00	1.00	1.00	0.62	0.85	0.96	0.64	0.85	0.95	1.22

Table 1: Power reduction by local logic transformation based on CSPF

[5] S. Muroga, Y. Kambayashi, H. Lai, and J. Culliney. The transduction method - design of logic networks based on permissible functions. *IEEE Transactions on Computers*, 38(10):1404–1424, Oct. 1989.

[6] B. T. Murphy. VLSI - directions and impact. In *Proc. of European Solid-State Circuits Conference*, Southampton, England, Sept. 1979.

[7] B. Rohfleisch, A. Kolbl, and B. Wurth. Reducing power dissipation after technology mapping by structural transformations. In *Proc. of the ACM/IEEE Design Automation Conference*, pages 789–794, June 1996.

[8] C. Y. Tsui, M. Pedram, and A. M. Despain. Technology decomposition and mapping targeting low power dissipation. In *Proc. of the ACM/IEEE Design Automation Conference*, pages 68–73, June 1993.

[9] Q. Wang and S. B. K. Vrudhula. Multi-level logic optimization for low power using local logic transformations. In *Proc. of the IEEE/ACM International Conference on Computer Aided Design*, pages 270–277, San Jose, CA, Nov. 1996.

[10] S. Yamashita, H. Sawada, and A. Nagoya. A new method to express functional permissibilities for LUT based FPGAs and its applications. In *Proc. of the IEEE/ACM International Conference on Computer Aided Design*, Nov. 1996.