

논문 2006-43SD-2-7

디지털 회로에서의 새로운 모델 기반 IP-Level 소모 전력 추정 기법

(New Model-based IP-Level Power Estimation Techniques for Digital Circuits)

이창희*, 신현철**, 김경호***

(Changhee Lee, Hyunchul Shin, and Kyungho Kim)

요약

반도체 공정기술의 발달로 인해 칩의 집적도가 향상되고 높은 성능의 SoC (System On a Chip)의 구현이 가능해졌다. 하지만 이로 인한 칩의 전력 소모량 증가는 칩 설계시의 중요 제한 요소가 되고 있다. 칩 설계의 하위 단계로 갈수록 설계의 수정은 시간과 금전적 비용을 기하급수적으로 증가시키기 때문에, 설계의 상위 단계에서부터 칩의 소모 전력을 미리 추정하는 기술은 필수적이다. 이에 본 연구에서는 효율적인 상위 레벨 소모 전력 추정을 위해 회로를 레벨화 하고, 일부 레벨의 스위칭을 기반으로 회로의 소모 전력을 look up 테이블을 이용하여 모델링하였다. 제안한 기술을 이용하여 ISCAS'85 벤치마크 회로에 대해 평균 소모 전력을 추정한 결과, 기존에 알려진 소모 전력 추정 기술^[1]에 비해 평균 추정 오차를 9.45%에서 3.84%로 크게 개선한 결과를 얻을 수 있었다.

Abstract

Owing to the development of semiconductor processing technology, high density complex circuits can be integrated in a System-on-Chip (SoC). However, increasing energy consumption becomes one of the most important limiting factors. Power estimation at the early stage of design is essential, since design changes at lower levels may significantly lengthen the design period and increase the cost. In this paper, logic level circuits are levelized and several levels are selected to build power model tables for efficient power estimation. The proposed techniques are applied to a set of ISCAS'85 benchmark circuits to illustrate their effectiveness. Experimental results show that significant improvement in estimation accuracy and slight improvement in efficiency are achieved when compared to those of a well-known existing method^[1]. The average estimation error has been reduced from 9.49% to 3.84%.

Keywords : Low-power, Power estimation, Power modeling.

I. 서론

반도체 공정 기술의 발달은 칩의 집적도를 향상시켜, 동일한 면적에 집적할 수 있는 게이트 수를 지속적으로

증가 시켰다. 이로써 다양한 기능을 한 칩 내에 집적시킨 SoC (System on Chip)의 구현이 가능해 졌지만, 칩의 개발 단계에서부터 출시되기까지의 시간(time-to-market)이 점점 짧아지고 있는 상황에서, 다양하고 복잡한 설계 및 제조 과정으로 인해 발생하는 많은 비용과 생산시간은 큰 문제점이 되고 있다. 때문에 반도체 설계의 상위 단계에서부터 향후 발생할 수 있는 문제점들을 미리 파악하고, 이를 바탕으로 다음 단계에서 발생할 수 있는 문제점들에 대한 조치를 미리 취한다면, 시간적 지연과 추가 비용 발생 문제를 어느 정도 방지할

* 학생회원, ** 평생회원 한양대학교 전자컴퓨터공학부 (School of Electrical and Computer Engineering Hanyang University)

*** 평생회원, 삼성전자 정보통신총괄 통신연구소 (Telecom R&D Center, Telecommunication Networks Samsung Electronics)

접수일자: 2005년10월10일 수정완료일: 2006년1월16일

수 있다. 또한, 시장을 선점하는 데에도 유리한 위치에 설 수 있다.

최근에 디지털 회로 설계 과정에서 발생하는 많은 문제점 중 하나가 완성된 칩의 전력 소모 문제이다^{[2][3]}. 배터리 기술의 발전 속도가 칩의 집적도 증가로 인한 전력 소모를 감당하지 못하고 있기 때문이다. 이러한 상황에서 현재의 설계 기술은 칩의 전력 소모량을 대부분 합성과정을 거친 게이트 수준에서나 추정하는 수준에 머물고 있다. 하지만 이를 좀 더 상위 단계에서 미리 추정할 수 있다면, 배터리 허용치를 넘는 소모 전력으로 인해 하위 설계 단계에서 발생하는 문제점들을 미리 파악하여 대처할 수 있을 것이다. 이는 칩의 생산에 소요되는 비용과 시간을 감소시키는 효과를 가져온다. 따라서 상위 단계에서 소모 전력을 추정하기 위한 많은 방법들이 연구되어 왔다.

회로에 인가되는 입력에 따라 simulation을 수행하여 소모 전력을 추정하면 비교적 정확한 값을 얻을 수 있다. 하지만 회로의 크기가 커지거나 인가되는 입력의 양이 많아지면 simulation 수행 시간이 길어진다는 단점이 있다. 이러한 단점을 보완하기 위해 Monte Carlo Simulation을 사용하여, 일정 기준에 이르면 모든 입력에 대해 simulation을 수행하지 않아도 simulation을 중단하는 방법이 있다^{[4][5]}. 이 경우에 확률적으로 계산하여 결정되는 simulation 중단 기준의 신뢰성이 중요하다.

인가되는 모든 입력에 따른 회로의 스위칭을 계산하지 않고, 입력 단 스위칭의 확률적 분포를 이용하여 소모 전력을 추정하는 방법도 있다.^[6-9] 이 방법들은 빠른 시간 안에 소모 전력을 계산할 수 있지만, 전체 회로의 스위칭을 입력 단 스위칭의 확률로써 계산하기 때문에 상당한 오차를 감수해야 하는 단점이 있다.

RTL에서의 Boolean function을 이용하여 회로의 소모 전력을 추정하는 방법도 있다. Boolean function에 따라 회로의 게이트 카운트를 계산하여 capacitance를 추정하고, 입력 단의 스위칭에 따라 회로의 스위칭을 추정함으로써 소모 전력을 계산한다.^[2] 이 경우 RTL에서 게이트 레벨로 합성되면 cell library에 따라 회로의 Boolean function이 달라진다는 문제점을 안고 있다.

다른 방법으로는 소모 전력을 매크로 모델식으로 모델링하고, 이 모델식을 이용하여 소모 전력을 추정하는 방법이 있다.^{[1],[10]} 이 방법은 간단한 모델식으로 소모 전력을 빠르게 추정할 수 있다는 장점이 있지만, 간단한 모델식 만으로는 소모 전력을 정확하게 모델링 할 수가 없어 소모 전력 추정 오차가 다소 커진다는 단점이 있

다. 또한 이 방법에서 정확도를 높이기 위해 2차식 혹은 3차식으로 매크로 모델식을 모델링한다면 계산량이 늘어나게 된다.

잘 설계된 IP(Intellectual-property)를 재사용하여 설계하는 것이 요즘 SoC 설계 추세이다. 제품의 수명이 점점 짧아지면서 IP 재사용으로 설계 시간을 줄이고, 완성될 칩의 신뢰성 또한 확보할 수 있다. 이러한 설계 추세를 반영하여 IP 단위로 소모 전력을 추정하기도 한다. 그 중 하나가 각 IP 단위로 circuit level에서 simulation을 하여 power library를 작성하고, 이를 참조하여 전체 회로의 소모 전력을 추정하는 방법이다.^{[3],[11],[12]} HyPE는 상위레벨에서 설계에 사용된 각 IP의 power library를 사용하여 전체 회로의 소모 전력을 추정하는 방법을 제안하고 있다^[3]. 적절한 크기의 library는 소모 전력 추정 수행 시간을 매우 빠르게 해주지만, library의 크기가 커지면 library 작성에 많은 시간이 소요될 뿐만 아니라, 이를 참조하여 소모 전력을 분석 하는데도 많은 시간이 걸린다. 하지만, 무조건 library의 크기를 줄이는 것은 소모 전력 추정 오차를 크게 만드는 결과를 가져온다. 때문에 이 방법에서는 효율적으로 library의 크기를 줄이는 것이 관건이다. 회로에 인가되는 입력에 따라 각 node에서 발생하는 Charging and Discharging Capacitance (CDC) Value를 이용하여 테이블을 작성하고 이를 참조하여 소모 전력을 추정하는 방법^[11]은 library의 크기를 줄이는 방법을 제안하고 있다. 이 방법에서는 Dynamic Grouping 기법을 사용하여 테이블의 크기를 줄인다. 하지만, CDC Value 계산을 위해 인가되는 입력에 따라 회로의 모든 node의 switching activity를 분석해야만 한다. 이를 위해 수행하는 logic simulation은 회로의 소모 전력 추정 수행 시간을 늘리는 결과를 낳는다. 회로의 크기가 커질수록 소모 전력 추정 시간은 직선적 이상으로 늘어나게 된다.

본 논문에서는 IP 별로 소모 전력을 추정하기 위한 매크로 모델식과 library를 모델링하고, 이를 기반으로 빠르고 정확하게 소모 전력을 추정하는 방법을 제안한다. 제안한 방법에서는 입력에 따른 스위칭에 따라 library를 참조하여 소모 전력을 추정한다. 이 과정에서 수행 시간을 줄이기 위해 회로를 각 게이트를 입력 단자로 부티의 거리에 따라 단계별로 레벨화하고, 일부 레벨에 대해서만 logic simulation을 수행하였다. 입력이 바뀔 때 따라 각 레벨의 node에서의 스위칭과 Hamming distance가 결정된다. 제안한 방법의 power library는 이러한 Hamming distance와 전체 회로의 소모 전력과의

관계를 look up 테이블을 이용하여 모델링하는 새로운 방법이다. 소모 전력 추정은 일부 레벨의 스위칭에 따른 Hamming distance 정보를 바탕으로 library를 참조하여 이루어진다.

II. 본 론

1. 소모 전력

조합 논리 회로에서 한 cycle 동안 소모되는 전력은 식 (1)과 같이 leakage current에 의해 소모되는 전력(static-power)과 입력 벡터의 변화에 의해 소모되는 전력(dynamic-power)의 합으로 나타낼 수 있다.

$$P = P_{static} + P_{dynamic} \quad (1)$$

일반적으로 static-power(P_{static})는 dynamic-power($P_{dynamic}$)에 비해 그 양이 매우 작았다. 하지만, 공정이 90nm 혹은 65nm로 발전 할수록 임계전압이 낮아지고 leakage current 또한 급격히 증가하고 있기 때문에 향후에는 칩 설계에 있어 중요 요소가 될 것이다. 본 논문에서는 현재 공정 기술에서 많은 부분을 차지하는 dynamic-power를 모델링하여 소모 전력을 추정하는데 초점을 맞춘다. 입력 벡터에 의해 소모되는 dynamic-power는 식 (2)와 같이 표현된다.

$$P_{dynamic} = \frac{1}{2} V_{dd}^2 \cdot \sum_{i=1}^n C_i \cdot f_i(x_1, x_2) \quad (2)$$

식 (2)에서 x_1 과 x_2 는 각각 이전 입력과 현재 입력을 나타낸다. 그러므로 $f_i(x_1, x_2)$ 은 이전 입력과 현재 입력에 의해 발생하는 node i 의 switching frequency를 의미한다. 또한 C_i 는 node i 의 capacitance를 나타낸다. 즉, 회로의 소모 전력은 node의 capacitance와 스위칭 발생 유무에 의해 좌우됨을 의미한다. 회로의 소모 전력을 모델링하기 위해서는 식 (2)에 나타난 바와 같이, 이전 입력과 현재 입력을 모두 고려해야 한다. n 개의 입력을 갖는 회로의 경우에 고려해야 할 경우의 수는 2^{2n} 개가 된다. 입력의 수가 증가할수록, 고려해야 할 경우의 수는 지수적으로 증가하기 때문에, 소모 전력을 모델링하기 위해 모든 입력 벡터를 고려하는 것은 현실적으로 불가능한 일이다. 따라서 입력 벡터에 따른

스위칭 정보를 최소화해서 소모 전력을 모델링 하는 새로운 기법이 필요하다. 하지만 이 경우에 최소한의 스위칭 정보를 가지고 얼마나 효율적으로 소모 전력을 모델링하느냐가 관건이다.

2. Level Selection

게이트 레벨 시뮬레이션은 회로 레벨에서의 시뮬레이션보다 훨씬 그 속도가 빠르다. 하지만, 크기가 매우 큰 회로에 대해서는 여전히 많은 시간을 필요로 한다. 조합 논리 회로는 그림 1과 같이 입력으로부터 거리에 따라 레벨화 시킬 수 있으며, 입력의 스위칭에 따라 내부 node의 값이 변화한다. 이때, 각 레벨 별로 스위칭한 비트의 수를 Hamming distance로 정의한다.

각 레벨 별 Hamming distance의 값에 따라 소모 전력을 모델링하는 것은 회로 전체 node의 모든 스위칭 경우의 수를 이용하는 것보다 계산 량이 줄어들게 된다. 다음 식 (3)은 그림 1 회로 전체 node의 모든 스위칭 경우의 수를 나타내고 있다. 이 경우 인풋 node를 제외한 모든 node의 개수가 21이고, 총 6개의 레벨로 나누어져 있다.

$$N_{node-switching} = 2^{21} = 2097152 \quad (3)$$

Hamming distance를 고려할 경우 5개의 node를 가지고 있는 레벨 1의 Hamming distance의 가능한 값은 0부터 5까지 6가지이다. 그러므로 그림 1 회로의 전체 Hamming distance 경우의 수는 식 (4)와 같아진다.

$$N_{HD} = 6 \times 8 \times 5 \times 2 \times 3 \times 3 = 4320 \quad (4)$$

각 레벨 별 Hamming distance의 변화에 따라 소모 전력을 모델링하면 모델링 과정에서 고려해야 할 경우의 수가 상당히 줄어든다. 하지만 큰 회로의 경우에는

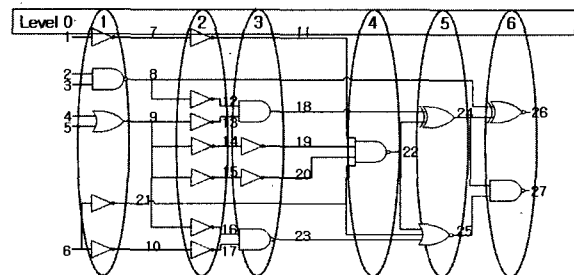


그림 1. 회로의 레벨화
Fig. 1. Circuit Levelization.

여전히 그 경우의 수가 많은 것이 사실이다. 만일, 회로의 모든 레벨에 대해 logic simulation을 수행하지 않고, 일부만을 선택하여 logic simulation을 수행한 후 이를 바탕으로 소모 전력을 추정할 수 있다면, 훨씬 빠르게 소모 전력을 추정할 수 있다. 다음 표 1에서는 회로의 전체 node에 대해 logic simulation을 수행했을 때와 일부 레벨을 선택하고 선택된 레벨에 속한 node에 대해서만 logic simulation을 수행했을 때의 CPU time을 비교하였다. 본 실험에서는 3000개의 input vector를 사용하였다.

표 2는 전체 node의 스위칭 정보를 이용하여 소모 전력을 추정할 때의 결과와 선택된 레벨의 node의 스위칭 정보만을 이용하여 소모 전력을 추정했을 때의 소모 전력 추정 결과를 나타낸다. 표 1과 2를 보면, 일부 레벨을 선택하여 logic simulation을 수행하는 것은 평균 20.36%의 소모 전력 추정 수행 시간 이득을 얻으면서도 그 정확도는 1.99%밖에 감소하지 않음을 알 수 있다.

제한한 방법에서는 전체 레벨에 대해 logic simulation을 수행한 후, 스위칭 빈도수가 가장 높은 3개의 레벨을 선택하여 소모 전력을 모델링하고, 이들 레벨에서만 logic simulation을 수행하여 소모 전력을 추정하였다.

표 1. Logic Simulation CPU time 비교
Table 1. Logic Simulation CPU times comparison.

Circuit	#Gate	Logic Simulation CPU Time		속도 향상 (%)
		전체 node (Sec)	선택된 level에 속한 node (Sec)	
C432	160	1.22	1.03	15.53
C2670	1193	7.58	5.72	24.58
C3540	1669	15.57	14.61	6.15
C7552	3512	21.65	14.03	35.19
평균 속도 향상 비율				20.36

표 2. Power estimation 오차 비교
Table 2. The power estimation error comparison.

Circuit	#Gate	Power Estimation 오차		정확도 (%)
		전체 node (%)	선택된 level에 속한 node (%)	
C432	160	1.69	0.23	+1.46
C2670	1193	3.16	6.29	-3.13
C3540	1669	3.96	5.14	-1.18
C7552	3512	1.73	3.90	-2.17
평균 정확도 감소				-1.99

레벨 선택 기준은 회로에 입력 벡터가 인가되었을 때, 내부 node의 스위칭 분포가 특정 레벨에 집중되는 경향을 보이는 결과에 기초하였다. 그림 2는 ISCAS'85 C432 benchmark circuit에 random 입력을 인가했을 때 각 레벨 별로 스위칭 정도를 나타내고 있다.

C432 benchmark circuit의 경우 입력 node를 제외하고 총 17개의 레벨로 나누어져 있으며, 입력 벡터에 따른 내부 회로의 스위칭 분포가 특정 레벨(레벨 1, 레벨 2, 레벨 5, 레벨 9)에 집중되어 있다. 특히 스위칭 빈도수가 가장 높은 상위 3개 레벨(레벨 1, 레벨 2, 레벨 9)의 스위칭 분포는 회로 전체 스위칭 분포의 48%를 차지하고 있다. 회로의 소모 전력이 node의 스위칭에 따라 결정된다는 사실에 비추어 볼 때, 스위칭 빈도수가 가장 높은 상위 3개 레벨의 스위칭 정보만을 이용하여 소모 전력을 모델링하여도 충분히 좋은 결과의 소모 전력 추정 값을 얻을 수 있음을 의미한다. 표 3에는 여러 가지 회로에 대해 스위칭 빈도수가 높은 상위 3개 레벨의 스위칭 빈도수가 전체 회로의 스위칭 빈도수에서 차지하는 비율을 나타내었다. 이 실험에서 예제 회로는 ISCAS'85 benchmark circuit을 사용하였다.

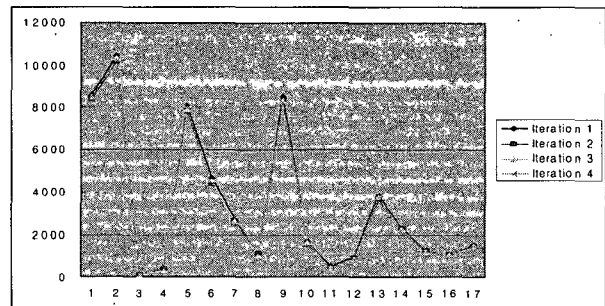


그림 2. C432 회로의 17 레벨에 따른 스위칭 분포
Fig. 2. The switching distribution each level of C432 benchmark circuit, on 17 levels.

표 3. 선택된 3개 레벨의 스위칭 빈도
Table 3. The switching frequency of selected three levels.

Circuit	# I	# O	# L	# G	비율
C432	36	7	17	160	48 %
C499	41	32	11	202	71 %
C880	60	26	24	383	30 %
C1196	32	32	24	529	51 %
C1238	32	32	22	508	50 %
C1908	33	25	40	880	32 %
평균 스위칭 비율					47 %

기존에 제안된 소모 전력 모델링 방법 중에도 회로의 3개 레벨만을 선택하여 소모 전력을 모델링하는 방법이 있다^[1]. 하지만, 이 방법은 본 논문에서 제안하는 방법과 달리 Spice-simulation과 같은 circuit level에서의 real-delay simulation을 통해 얻어지는 소모 전력 값과 계산식을 이용하여 추정한 소모 전력 값과의 오차를 최소화하는 3개 레벨을 선택하였다. 이 방법은 레벨 선택 과정에서 real-delay simulation을 수행하기 때문에 레벨 선택을 위한 시간이 오래 걸린다는 단점이 있다. 또한, 스위칭 빈도 분포 및 real-delay simulation 결과와의 오차를 모두 고려하기 때문에 레벨 선택 과정의 계산량 또한 크게 많아진다. 반면, 본 논문에서 제안한 레벨 선택 방법은 random 입력 logic simulation 만으로 간단히 3개 레벨을 선택할 수 있다. 본 방법은 효율적이면서, 다음에 기술하는 lookup table을 이용하면 정확도를 크게 높일 수 있다.

3. Look-up Table을 이용한 Power Modeling

회로 각 레벨의 Hamming distance는 인풋 벡터 쌍 (x_1, x_2) 에 의해 결정되며, 회로의 소모 전력은 이러한 Hamming distance에 따라 추정할 수 있다^[1]. 식 (5)와 같이 간단히 1차식으로 소모 전력을 모델링 할 수 있다.

$$Y = c_0(h) + c_1(h)H_{L1} + c_2(h)H_{L2} + c_3(h)H_{L3} \quad (5)$$

식 (5)에서 Y 은 모델식을 이용해 추정하는 평균 소모 전력이다. 또한 $c_0(h)$, $c_1(h)$, $c_2(h)$, $c_3(h)$ 은 모델링하는 과정에서 결정되는 계수이며, H_{L1} , H_{L2} , H_{L3} 는 세 레벨(L_1 , L_2 , L_3)에서의 Hamming distance이다.

각 레벨의 Hamming distance가 증가하면 회로의 실제 소모 전력 또한 증가한다. 그러나 이 둘의 관계는 반드시 직선적이지는 않다. 그림 3은 C880 benchmark circuit에서의 Hamming distance에 따른 소모 전력 변화를 보여준다. 식 (5)와 같은 소모 전력 모델식은 그림 3과 같은 non-linear한 특성을 제대로 반영하지 못해 소모 전력 추정 오차가 크게 나타난다. 때문에 본 논문에서는 non-linear한 특성을 보이는 Hamming distance와 소모 전력과의 관계를 테이블로 제작하고, 선택된 레벨의 Hamming distance에 따라 테이블을 참조하여 소모 전력을 추정하는 새로운 방법을 사용하였다. 또한 데이

터를 참조하기 위해 식 (5)와 같은 소모 전력 모델식을 식 (6)과 같이 변형하였다.

$$Y = c_0(h) + c_1(h)P_{L1}(H_{L1}) + c_2(h)P_{L2}(H_{L2}) + c_3(h)P_{L3}(H_{L3}) \quad (6)$$

식 (6)에서 $P_{L1}(H_{L1})$, $P_{L2}(H_{L2})$, $P_{L3}(H_{L3})$ 은 각각 선택된 3개 레벨의 Hamming distance 값(H_{L1} , H_{L2} , H_{L3})에 따라 테이블에서 참조한 소모 전력 값을 의미한다.

표 4는 본 논문에서 제안한 테이블의 구조를 보여준다. 이 테이블은 simulation의 각 cycle마다 선택된 3개 레벨에서의 Hamming distance에 따른 전체 회로의 소모 전력을 나타낸다. 우리는 테이블을 작성하기 위해 3,000개의 random vector simulation 데이터를 사용하였다. 때문에 이 테이블은 선택된 레벨의 Hamming distance가 가질 수 있는 모든 경우를 반영하지 못한다.

이 문제를 해결하기 위해서는 모든 경우를 고려할 수 있을 때까지 random vector simulation을 반복 수행하면 된다. 하지만 회로의 실제 소모 전력 값을 얻기 위해 real-delay simulation 또한 반복적으로 수행해야 한다.

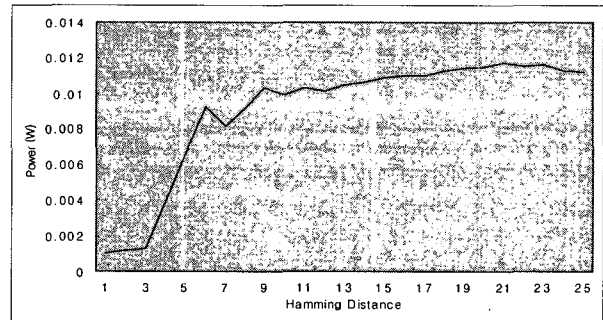


그림 3. Hamming distance와 소모 전력과의 관계
Fig. 3. The relationship between Hamming distance and power consumption.

표 4. 제안한 소모 전력 테이블
Table 4. The suggested power consumption table.

선택된 레벨 i 의 Hamming distance (H_{Li})	부분 추정 소모 전력 ($P_{Li}(H_{Li})$)
0	$P_{Li}(0)$
1	$P_{Li}(1)$
2	$P_{Li}(2)$
:	:
H_{Limax}	$P_{Li}(max)$

때문에 소모 전력 모델링에 많은 시간이 소요된다. 그래서 우리는 이 문제를 평균 소모 전력을 모델링 하는 과정에서 해결하기 보다는 평균 소모 전력을 추정하는 과정에서 해결하였다. 다음 세 가지 경우는 표 5와 같은 소모 전력 테이블을 참조하여 평균 소모 전력을 추정할 때, 테이블에서 해당 레벨의 Hamming distance 에 해당하는 부분 추정 소모 전력 값이 없을 때의 문제 해결 방안을 설명한 것이다.

가. 테이블의 HD 범위보다 작은 경우

이 경우에는 테이블에 존재하는 최하위 두 개의 소모 전력 차를 이용하여 Hamming distance 범위보다 작은 경우의 소모 전력을 계산한다. 표 2에서 Hamming distance가 0일 때와 1일 때의 경우는 다음 식 (7), (8) 을 각각 이용한다.

$$P_{Li}(0) = P_{Li}(2) - (P_{Li}(3) - P_{Li}(2)) \times 2 \quad (7)$$

$$P_{Li}(1) = P_{Li}(2) - (P_{Li}(3) - P_{Li}(2)) \times 1 \quad (8)$$

나. 테이블의 HD 범위보다 큰 경우

이 경우에는 가.의 경우와 같은 방식으로 계산하되 테이블에 존재하는 최상위 두 개의 소모전력 차를 이용하여 계산한다. 표 2에서 Hamming distance가 HDmax 일 때의 경우는 다음 식 (9)를 이용한다.

$$P_{Limax} = P_{Li}(max - 1) + (P_{Li}(max - 1) - P_{Li}(max - 2)) \times 1 \quad (9)$$

다. 테이블의 HD 범위 내에 있지만 누락된 경우

이 경우에는 소모 전력값이 누락된 항목에서 가장 가까운 앞뒤 소모 전력 값을 이용하여 계산한다. 표 5에서 Hamming distance가 4일 때와 5일 때의 경우 다음 식 (10), (11)을 각각 이용한다.

$$P_{Li}(4) = P_{Li}(3) + \left(\frac{P_{Li}(6) - P_{Li}(3)}{3} \right) \times 1 \quad (10)$$

$$P_{Li}(5) = P_{Li}(3) + \left(\frac{P_{Li}(6) - P_{Li}(3)}{3} \right) \times 2 \quad (11)$$

위 세 가지 방법은 random vector simulation을 통해 테이블을 생성하는 과정에서 고려되지 못한 Hamming

표 5. 테이블에 누락된 소모 전력 값
Table 5. The missed value at the table entry.

Hamming distance	부분 추정 소모 전력
0	non
1	non
2	$P_{Li}(2)$
3	$P_{Li}(3)$
4	non
5	non
6	$P_{Li}(6)$
:	:
$H_{Limax-2}$	$P_{Li}(max-2)$
$H_{Limax-1}$	$P_{Li}(max-1)$
H_{Limax}	non

distance에 따른 소모 전력을 linear한 식을 이용하여 interpolation/extrapolation하는 방법이다. 이 방법은 테이블을 작성하는 것이 non-linear한 소모 전력의 특성을 감안하기 위함이라는 목적에 위배된다. 하지만, 모든 경우의 수에 대해 테이블을 작성하기 위해서 감수해야할 real-delay simulation의 오버헤드가 매우 크다는 사실에 비추어 볼 때, 제안하는 방법은 충분히 타당성을 지니고 있다.

4. Recursive Least Square Algorithm

회로의 평균 소모 전력을 구하기 위해서 3절에서 식 (6)과 같은 모델식을 제안하였다. 이와 같은 식을 모델링하기 위해서는 실제 real-delay simulation과의 오차를 최소가 되게 하는 계수 $c_0(h)$, $c_1(h)$, $c_2(h)$, $c_3(h)$ 를 결정하는 것이 중요하다. 우리는 최적화된 계수들을 찾기 위해 Recursive Least Square (RLS) Algorithm^{[13],[14],[15]}을 사용하였다. RLS Algorithm은 실제 실험적인 수치의 통계량을 이용하여 추정 모델식을 만드는 유용한 Algorithm이다. RLS Algorithm을 사용하면 real-delay simulation을 수행하여 얻은 기준 평균 소모 전력 Y 와 모델식을 계산하여 얻어진 추정 평균 소모 전력 Y' 와의 오차 e 가 최소가 되도록 계수를 결정할 수 있다. 이 Algorithm은 오차 e 의 제곱 항이 최소가 되는 계수를 찾기 위해 식 (12)에서부터 출발한다.

$$\sum e^2 = \sum (Y - Y')^2 \quad (12)$$

위 식에서 Y' 는 식 (6)과 같으므로 식 (12)는 식 (13)

과 같이 다시 쓸 수 있다.

$$\sum e^2 = \sum [Y - (c_0(h) + c_1(h)P_{L1}(H_{L1}) + c_2(h)P_{L2}(H_{L2}) + c_3(h)P_{L3}(H_{L3}))]^2 \quad (13)$$

식 (13)을 계수 $c_0(h)$, $c_1(h)$, $c_2(h)$, $c_3(h)$ 에 대하여 각각 편미분하면 다음 식(14), (15), (16), (17)과 같은 4개의 식을 얻을 수 있다.

$$\sum 2(Y - c_0(h) - c_1(h)P_{L1}(H_{L1}) - c_2(h)P_{L2}(H_{L2}) - c_3(h)P_{L3}(H_{L3}))(-1) = 0 \quad (14)$$

$$\sum 2(Y - c_0(h) - c_1(h)P_{L1}(H_{L1}) - c_2(h)P_{L2}(H_{L2}) - c_3(h)P_{L3}(H_{L3}))(-P_{L1}(H_{L1})) = 0 \quad (15)$$

$$\sum 2(Y - c_0(h) - c_1(h)P_{L1}(H_{L1}) - c_2(h)P_{L2}(H_{L2}) - c_3(h)P_{L3}(H_{L3}))(-P_{L2}(H_{L2})) = 0 \quad (16)$$

$$\sum 2(Y - c_0(h) - c_1(h)P_{L1}(H_{L1}) - c_2(h)P_{L2}(H_{L2}) - c_3(h)P_{L3}(H_{L3}))(-P_{L3}(H_{L3})) = 0 \quad (17)$$

식 (14)를 정리하면 다음 식 (18)과 같다.

$$c_0(h) = \bar{Y} - c_1(h)\overline{P_{L1}(H_{L1})} - c_2(h)\overline{P_{L2}(H_{L2})} - c_3(h)\overline{P_{L3}(H_{L3})} \quad (18)$$

식 (18)에서 \bar{Y} , $\overline{P_{L1}(H_{L1})}$, $\overline{P_{L2}(H_{L2})}$, $\overline{P_{L3}(H_{L3})}$ 은 각 실험 결과로부터 얻은 Y , $P_{L1}(H_{L1})$, $P_{L2}(H_{L2})$, $P_{L3}(H_{L3})$ 값들의 평균이다. 식 (18)을 식(15), (16), (17)에 대입하여 정리하면 식 (19)와 같아진다.

$$\begin{aligned} \left(\begin{array}{c} \sum P_{L1}(H_{L1}) \cdot Y - \bar{Y} \sum P_{L1}(H_{L1}) \\ \sum P_{L2}(H_{L2}) \cdot Y - \bar{Y} \sum P_{L2}(H_{L2}) \\ \sum P_{L3}(H_{L3}) \cdot Y - \bar{Y} \sum P_{L3}(H_{L3}) \end{array} \right) &= c_1(h) \left(\begin{array}{c} \sum (P_{L1}(H_{L1}))^2 - \overline{P_{L1}(H_{L1})} \sum P_{L1}(H_{L1}) \\ \sum P_{L1}(H_{L1}) \cdot P_{L2}(H_{L2}) - \overline{P_{L1}(H_{L1})} \sum P_{L2}(H_{L2}) \\ \sum P_{L1}(H_{L1}) \cdot P_{L3}(H_{L3}) - \overline{P_{L1}(H_{L1})} \sum P_{L3}(H_{L3}) \end{array} \right) \\ &+ c_2(h) \left(\begin{array}{c} \sum P_{L2}(H_{L2}) \cdot P_{L1}(H_{L1}) - \overline{P_{L2}(H_{L2})} \sum P_{L1}(H_{L1}) \\ \sum (P_{L2}(H_{L2}))^2 - \overline{P_{L2}(H_{L2})} \sum P_{L2}(H_{L2}) \\ \sum P_{L2}(H_{L2}) \cdot P_{L3}(H_{L3}) - \overline{P_{L2}(H_{L2})} \sum P_{L3}(H_{L3}) \end{array} \right) \\ &+ c_3(h) \left(\begin{array}{c} \sum P_{L3}(H_{L3}) \cdot P_{L1}(H_{L1}) - \overline{P_{L3}(H_{L3})} \sum P_{L1}(H_{L1}) \\ \sum P_{L3}(H_{L3}) \cdot P_{L2}(H_{L2}) - \overline{P_{L3}(H_{L3})} \sum P_{L2}(H_{L2}) \\ \sum (P_{L3}(H_{L3}))^2 - \overline{P_{L3}(H_{L3})} \sum P_{L3}(H_{L3}) \end{array} \right) \quad (19) \end{aligned}$$

식 (19)의 Y 항에 real-delay simulation 을 통해 얻

은 소모 전력 값을 대입하고, P_{L1} , P_{L2} , P_{L3} 항에 선택된 레벨의 Hamming distance에 따라 테이블에서 찾은 값을 대입하면 real-delay simulation에서 얻은 기준 소모 전력 값 Y 와 모델식을 이용하여 계산하는 추정 소모 전력 값 Y' 와의 오차 e 가 최소가 되는 계수 $c_0(h)$, $c_1(h)$, $c_2(h)$, $c_3(h)$ 를 찾을 수 있다.

III. 실험

우리는 회로를 레벨화 시키고, 3,000개의 random vector simulation을 통해 스위칭 빈도수가 가장 높은 3개 레벨을 선택하였다. 그 후, 선택된 3개 레벨에 대해 logic simulation과 real-delay simulation을 수행하여 얻은 Hamming distance 정보와 전체 회로의 소모 전력을 테이블로 모델링하였으며, RLS Algorithm을 이용하여 모델식을 만들었다. 이와 같은 power modeling process를 요약하면 다음과 같다.

- i) 회로의 레벨화
- ii) Random vector logic simulation
- iii) Switching activity 순으로 상위 3개 레벨을 선택
- iv) Random vector real-delay simulation
- v) Power table 작성
- vi) Recursive Least Square(RLS) Algorithm을 사용하여 power equation 모델링
- vii) 완성된 power 모델을 데이터 베이스에 추가

위와 같은 과정으로 만들어진 테이블과 모델식을 이용하여 ISCAS'85 benchmark circuit에 새로운 1,000개의 input vector가 인가되었을 때 소모 전력 추정 실험을 하였다. 본 실험은 Ultra Sparc 450MHz dual CPU machine에서 이루어졌으며, SunOS 5.8을 사용하였다.

표 6은 실험 결과를 보여준다. # I, # O, # L, # G는 각각 input, output, level, gate의 개수를 의미한다. 표 6에서 HSPICE simulation 결과와 본 논문에서 제안한 power 모델을 이용한 결과는 평균 3.84%의 오차를 보인다. 이는 표 7에 나타낸 바와 같이 기존 방법^[1]의 결과에 비해 상당히 개선된 것이다. 또한 표 8에서 보듯이 소모 전력 추정 시간이 HSPICE simulation 시간에 비해 현저하게 줄어들었다.

표 6. 소모 전력 추정 결과

Table 6. The results of the power estimation.

Circuit	# I	# O	# L	# G	HSPICE (mW)	Power Model (mW)	오차 (%)
C432	36	7	17	160	5.43	5.42	0.23
C499	41	32	11	202	10.87	10.87	0.21
C880	60	26	24	383	9.50	10.05	5.80
C1196	32	32	24	529	8.42	8.26	1.90
C1238	32	32	22	508	8.52	7.92	7.05
C1355	41	32	24	546	11.68	11.48	1.66
C1908	33	25	40	880	23.78	23.16	2.36
C2670	233	140	32	1193	42.63	45.31	6.29
C3540	50	22	47	1669	44.57	46.86	5.14
C5315	178	123	49	2307	68.32	63.53	7.01
C6288	32	32	124	2416	60.12	62.87	4.57
C7552	207	108	43	3512	79.31	76.22	3.90
평균 오차							3.84

표 7. 기존 방법과 제안한 방법의 오차 비교

Table 7. The error comparison of the previous work and suggested method.

Circuit	기존 방법 ^[1] 의 오차 (%)	제안한 방법의 오차 (%)	개선도 (%)
C432	15.68	0.23	15.45
C499	4.43	0.21	4.22
C880	13.47	5.80	7.67
C1196	n. a.	1.90	n. a.
C1238	n. a.	7.05	n. a.
C1355	8.92	1.66	7.26
C1908	12.46	2.36	9.78
C2670	17.72	6.29	11.43
C3540	19.23	5.14	14.09
C5315	9.65	7.01	2.64
C6288	17.06	4.57	12.49
C7552	14.64	3.90	10.74
평균	13.33	3.84	9.49

(n. a. : not available)

IV. 결론

본 논문에서는 테이블 look-up 방법에 기초한 새로운 소모 전력 모델링 및 추정 방법을 제안하였다. 그 결과 트랜지스터 레벨에서 소모 전력량을 계산하는 HSPICE simulation에 비해 3410~22259배 빠르게 소모 전력량을 계산할 수 있었다. 뿐만 아니라 HSPICE simulation 결

표 8. 제안하는 방법의 소모 전력 추정 수행 시간

Table 8. The power estimation CPU time of the suggested method.

Circuit	HSPICE simulation CPU time (sec)	제안한 방법의 CPU time (sec)
C432	6398	1.5
C499	11549	1.2
C880	11516	1.1
C1196	30719	1.4
C1238	31163	1.4
C1355	7631	4.6
C1908	12848	8.2
C2670	25232	7.4
C3540	33597	7.3
C5315	45221	9.6
C6288	46877	7.9
C7552	79661	9.1

과와 0.2~7.1 %의 오차만을 보이는 상당히 좋은 결과를 얻을 수 있었다. 이는 기존 논문^[1]에서의 추정 오차 4.4~15.7 %를 0.21~7.05 %로 크게 개선한 결과이다.

감사의 글

※ 본 연구에 사용한 HSPICE 툴은 IDEC에 의해 지원되었음.

참고 문헌

- [1] S. Gupta and F.N. Najm, "Energy and peak-current per-cycle estimation at RTL", *IEEE Transaction on Very Large Scale Integration (VLSI) Systems*, vol. 11, no. 11, pp.525-537, Aug. 2003.
- [2] K.M. Buyuksahin and F.N. Najm, "Early power estimation for VLSI circuits", *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 7, pp.1076-1088, July 2005.
- [3] X. Liu and M.C. Papaefthymiou, "HyPE: Hybrid power estimation for IP-based Systems-on-Chip", *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 7, pp.1089-1103, July 2005.
- [4] M. Xakellis and F. Najm, "Statistical estimation of the switching activity in digital circuits", *in*

31st ACM/IEEE Design Automation Conference, San Diego, pp.728-733, 1994.

[5] R. Burch, F. Najm, P. Yang, and T. Trick, "McPOWER : A monte carlo approach to power estimation," *IEEE/ACM International Conference Computer-Aided Design, Santa Clara, CA*, pp. 90-97, Nov. 1992.

[6] F. Najm, "Transition density: a new measure of activity in digital circuits", *IEEE Transaction on Computer-Aided Design*, vol. 12, no. 2, pp.310-323, Feb. 1993.

[7] R. Brodersen, A. Chandrakasan, and S. Sheng, "Technologies for personal communications", *1991 Symposium VLSI Circuits, Tokyo, Japan*, pp.5-9, 1991.

[8] F. Najm, R. Burch, P. Yang, and I. Hajj, "CREST-A current estimator for CMOS circuits", *IEEE International Conference Computer-Aided Design, Santa Clara, CA*, pp.204-207, Nov. 1988.

[9] M. Cirit, "Estimation dynamic power consumption of CMOS circuits", *IEEE international conference Computer-Aided Design*, pp.534-537, Nov. 1987.

[10] A. Bogliolo, R. Corgnati, E. Macii and M. Poncino, "Parameterized RTL power models for soft macros", *IEEE Transaction on Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 6, pp.880-887, Dec. 2001.

[11] C.-Y. Hsu, C.-N. J. Liu and J.-Y. Jou, "An efficient IP-level power model for complex digital circuits", *Design Automation Conference, 2003. Proceedings of the ASP-DAC 2003. Asia and South Pacific, 21-24*, pp.610-613 Jan. 2003.

[12] R. Ferreira, A. Trullemans, J. Costa, and J. Monteiro, "Probabilistic bottom-up RTL power estimation", *Quality Electronic Design, 2000. ISQED 2000. Proceedings. IEEE 2000 First International Symposium on*, pp.439-446, Mar. 2000.

[13] 이우리, 오광우, "회귀분석 입문 및 응용", 탐진, 340-353쪽, 2003년

[14] L. Ljung, "System Identification: Theory for the User." Englewood Cliffs, NJ: Prentice-Hall, 1987

[15] S. Neter and W. Wasserman, "Applied Linear Statistical, Models.", Homewood, IL: Richard D. Irwin, 1974.

저 자 소 개



이 창 희(학생회원)
2004년 한양대학교 전자컴퓨터 공학부 학사 졸업.
2004 ~ 현재 한양대학교 전자전기 제어계측공학과 석사과정.

<주관심분야: Power Estimation, 저전력 설계>



신 현 철(평생회원)
1978년 서울대학교 전자공학과 학사 졸업.
1980년 한국과학기술원 전기 및 전자공학 석사 졸업.
1983~1987년 U.C Berkeley Ph.D
1983~1987년 Fulbright scholarship

1987~1989년 MTS, AT&T Bell Lab's, Murray Hill N.J., USA
1989~현재 한양대학교 전자컴퓨터공학부 교수
1997~현재 IDEC 한양대학교 지역센터 센터장
<주관심분야: CAD&VLSI, 통신용반도체 설계, 저전력 설계>



김 경 호(평생회원)
1984년 연세대학교 전자공학과 학사 졸업
1987년 한국과학기술원 전기 및 전자공학 석사 졸업.
1991년 한국과학기술원 전기 및 전자공학 박사 졸업.

1983~현재 삼성전자 정보통신총괄 통신연구소 상무

1997년 Marquis's Who's Who 인명록 등재
1998년 IBC 인명록 등재
2003년 자랑스런 삼성인상 수상(모뎀개발)
<주관심분야: 저전력 설계, 통신모뎀, IT SoC>