

## 커널 추출을 이용한 저전력설계

이귀상, 정미경

전남대학교 전산통계학과

Tel:(062)530-3425 , E-mail:gslee@chonnam.chonnam.ac.kr

### Low Power Design Using the Extraction of kernels

Guee-sang Lee, Mi-gyoung Jung

Department of Computer Science and Statistics, Chonnam National University

Tel:(062)530-3425 , E-mail:gslee@chonnam.chonnam.ac.kr

#### Abstract

In this paper, we propose a new method for power estimation in nodes of multi-level combinational circuits and describe its application to the extraction of common expressions for low power design. It is assumed that each node is implemented as a complex gate and the capacitance and the switching activity of the nodes are considered in the power estimation. Extracting common expressions which is accomplished mostly by the extraction of kernels, can be transformed to the problem of rectangle covering. We describe how the newly proposed estimation method can be applied to the rectangle covering problem and show the experimental results with comparisons to the results of SIS-1.2.

#### 1. 서론

오늘날 저전력화는 거의 모든 전자제품에 공통적으로 필요한 사항이 되어 있다. 정보화에 대한 인식과 활용이 보편화되면서 발생하는 개인 휴대용 컴퓨터를 비롯한, 이동 통신이나 개인 휴대통신의 폭발적인 수요를 고려할 때, 관련제품의 경쟁력을 결정하는 소형화, 경량화의 문제를 해결하기 위해서 저전력 설계방법은 반도체 설계 시 필수적인 고려사항으로 인식되고 있다.

이러한 현실적인 요구로 인하여 지난 수년간 저전력 반도체설계방법에 대한 연구가 매우 활발히 진행되었으나 어떤 단 하나의 방법이 전력소모의 문제점을 단

번에 해결해 주지는 못하였다. 저전력 설계의 효과를 얻기 위해서는 소자 및 회로 수준, layout 수준, 논리 수준 그리고 구조 및 알고리즘 수준의 모든 설계단계에서 전력소모가 고려되어야 하며 본 논문에서는 반도체 설계 계층(design hierarchy)의 여러 단계 중에서 논리합성단계에서의 저전력 설계 방법을 기술하며 특히 논리합성방법의 기초가 되는 불리안 함수에서의 공통표현추출에 대하여 연구한다.

논문에서는 공통표현 추출에 있어서 어떻게 전력소모를 비용함수로 적용하는지 분석하여 이를 합성도구로 구현한다. 이러한 공통 표현 추출에는 커널 추출이 주된 역할을 하며 구체적으로 SIS-1.2[1]에서의 논리합성방법을 개선하여 그 결과를 기존의 방법과 비교한다.

본 논문의 구성은 다음과 같다. 제2장에서는 전력소모 예측방법과 공통표현추출의 기본이론인 커널 추출, 사각형 커버링에 대하여 알아보고 3장에서는 저전력 설계를 위해 이를 어떻게 사각형 커버링에 적용하여 논리합성을 실행하는지를 설명하고 4장에서 이의 실험 결과를 보이며, 마지막으로 결론을 제시한다.

#### 2. 관련연구

##### 2.1 전력소모 예측 기법[2,3,4,5,6]

일반적으로 VLSI 회로들은 CMOS로 구현되고 있으며 이러한 CMOS 회로에서의 주된 전력소모는 회로 노드의 charge/discharge에 의해서다. 대부분의 최적화 방법들은 이에 의한 전력소모를 최소화하는 데 중점을 두고 있다. 따라서 대부분의 저전력화 방법들은 CMOS 회로에서의 스위칭 동작에 의한 전력 (switching activity power)을 최소화하기 위한 방법들

은 다루고 있으며 함수의 구현은 CMOS 회로로 이루어진다고 가정하며 이러한 가정 하에서 다음과 같은 식으로 평균전력소모를 나타낼 수 있다.

$$P_{avg} = \frac{1}{2} C_{load} \frac{V_{dd}^2}{T_C} E(\text{transitions})$$

여기서  $C_{load}$ 는 출력 로드의 커패시턴스이고  $V_{dd}$ 와  $T_C$ 는 각각 공급전압과 전체 클럭 사이클을 나타내고  $E(\text{transitions})$ 는 게이트출력의 스위칭 활동량의 기대치이다. 정확한 전력소모 예측은 이러한 각 요소들과 함께 적절한 지연 모델(delay model)을 선택하여야 하며 여기에는 zero delay model과 real delay model이 있다. 본 논문에서는 global BDD를 이용한 zero delay model의 전력소모 예측기법을 사용하며 필요에 따라 본 연구결과를 SIS-1.2의 simulation에 의한 real delay model로 확장할 수 있을 것이다.

### 2.2 커널의 추출과 사각형 커버링[7]

간단히 커널과 사각형 커버링을 정의하고 그 개념을 설명한다. 임의의 표현  $f$ 은 어떤 큐브로도 균등하게 그 표현을 나눌 수 없으면 "cube-free"라 한다.

예)  $f = ab + c$  cube-free  
 $f = ab + acf = abc$  not cube-free

[정의 1] 임의의 논리함수  $f$ 의 primary divisor  $D(f)$ 와 커널  $K(f)$ 는 다음과 같이 정의된다.

$$D(f) = \{f/c \mid c \text{ is a cube}\}$$

$$K(f) = \{g \mid g \in D(f) \text{ and } g \text{ is cube-free}\}$$

블리안 대수 연산에서 가장 열쇠가 되는 것은 divisor를 알아내는 것이다. 커널은 factoring, decomposition 과 extraction에 필요한 divisor의 집합을 나타내며 이는 하나 또는 다수의 공통 큐브를 구하는 것과 함께 이러한 공통 표현을 찾는 수단으로 사용된다. 이러한 공통표현추출은 사각형 커버링으로 변환할 수 있으며 사각형은 다음과 같이 정의된다.

[정의 2] 행렬  $B, B_{ij} = \{0, 1, *\}^2$ 의 사각형  $(R, C)$ 은 모든  $i \in R, j \in C$ 에 대하여  $B_{ij} \in \{1, *\}$ 인 행들의 부분집합  $R$ 과 열들의 부분집합  $C$ 로 이루어진다.

만일  $[B_{ij} = 1 \rightarrow i \in R^k, j \in C^k]$ 이면 사각형들의 집합  $\{(R^k, C^k)\}$ 은 행렬  $B$ 의 사각형 커버라 한다.  $B$ 의 각 1은 적어도 하나 이상의 사각형에 의하여 cover 되어야 한다.  $B$ 의 항 중에서 \*는 don't care 항을 말한다.

이러한 사각형 커버링의 개념을 사용하여 커널을 계산해 낼 수 있다. 예를 들어 함수  $f = abd' + acd + bcd$ 을 행렬로 표시하면 (큐브-리터럴 행렬) 각 행에는 각각의 큐브를, 각 열에는 각각의 리터럴(literal)을 두어

서 (표 1)와 같이 표시할 수 있다.

(표 1)에서 사각형  $\{R, C\} = \{(2, 3), (3, 4)\}$ 는 큐브  $cd$ 에 해당하며 co-rectangle  $\{R, C\} = \{(2, 3), (1, 2, 5)\}$ 은 커널  $a + b$ 을 나타낸다. 따라서 사각형은 co-kernel

	a	b	c	d	d'
abd'	1	1	0	0	1
acd	1	0	1	1	0
bcd	0	1	1	1	0

표 1 큐브 리터럴 행렬

에 해당하며 커널을 얻기 위해 필요한 큐브 divisor이다. 즉, 큐브 divisor는 열  $C$ 에 나타난 리터럴들의 집합  $cd$ 을 말한다. 다시 말하면 큐브-리터럴 행렬에서 두 개 이상의 행을 갖는 주 사각형(prime rectangle)은 함수  $f$ 의 co-kernel 큐브를 나타낸다. 그러므로 커널은 그러한 주 사각형의 co-rectangle에 해당한다고 할 수 있다. 여기서 level-0 커널은 다른 어떤 사각형도 포함하지 않는 열들의 집합을 갖는 주 사각형의 co-rectangle이다.

### 3. 저전력설계를 위한 공통 커널의 추출

본 논문에서는 전력소모 예측을 각 노드에서의 정전용량과 스위칭 활동량을 고려하여 수행한다. 실제로 SIS-1.2에서는 모든 전력소모는 각 노드의 정전용량에서의 charge/discharge에 의한 전력소모를 주로 계산하며 이는 각 노드의 정전용량과 스위칭 활동량을 고려하므로 이에 맞는 전력소모 예측함수를 설정하여야 한다.

따라서 본 논문의 초점은 커널을 추출하는데 사각형 커버링을 적용함에 있어서 이러한 정전용량과 스위칭 활동량을 어떻게 이용하는가 이다. 기본적으로 본 논문은 입력변수 load의 변화를 스위칭 활동량과 함께 적용하는 것이다. 임의의 함수  $F$ 의 커널 추출 시, 정전용량을 각 노드에서의 입력변수의 크기에 비례한다고 보고 이를 고려하여 전력소모식을 설정한다.

$$R \sum_{i=0}^M f(v_i)w(D)CL(v_i, D) +$$

$$(P-1) \sum_{i=0}^R \sum_{j=0}^M f(v_i)w(q_j)CL(v_i, q_j)$$

$$- R w(D)f(D) - \sum_{i=0}^M f(v_i)w(D)CL(v_i, D)$$

여기서  $w(K)$ 는 논리식  $K$ 가 속한 노드의 무게(weight)를 말하며 본 논문에서는 이 무게함수로 그 노드의 입력 수를 사용한다. 위의 식에서 첫째항의 경우  $D$ 의 표현이 소속된 노드는 새로운 노드를 적용하기 전의 원래의 식이고 네 번째 항의  $D$ 는 새로운 노드가 형성된 후의 이 노드의 표현을 말한다. 그리고 둘째 항에서의  $w(q_j)$ 도 큐브  $q_j$ 가 소속된 노드, 즉 원래의 식에서의 무게를 말한다. 세 번째 항은 새로운 노드의 출력변수를 원래의 식에서 입력변수로 사용할 때의 경우를 나타낸다. 그러므로 이 때의 무게도 원래

의 식의 무게를 말하며 이러한 무게에 따라 스위칭 활동량의 합계가 달라짐을 알 수 있다.

예를 들어 식(1)에서 식  $F$ 의 입력변수의 무게는 7이며 식  $G$ 의 무게는 4, 그리고 식  $X$ 의 무게는 2임을 알 수 있다. 물론 여기에서 새로운 노드를 대입한 후의 입력변수의 수는 그 전의 입력변수의 수와는 다르다. 그러므로 여기에서의 문제점은 새로운 노드를 대입 전에 새로운 변수를 사용한 노드의 입력 수를 예

$$F = af + bf + ag + cg + ade + bde + cde$$

$$G = af + bf + ace + bce$$

이 논리식들이 다음과 같이 변화될 수 있다.

$$F = deX + fX + ag + cg + cde$$

$$G = ceX + fX \quad \text{---식(1)}$$

$$X = a + b$$

측하기가 어렵다는 것이다. 이를 정확히 계산하기 위해서는 원래의 식을 새로운 노드(커널)로 나누어야 하며 이는 상당한 cost를 요구할 것으로 생각되며 본 논문에서는 근사적인 방법으로 새로운 노드를 대입하기전의 노드의 무게를 사용하며 이것이 변화될 때마다 바로 무게합수를 바꾸어 다음에 적용될 때는 새로운 무게값을 사용하도록 한다. 이를 사각형 커버링에 적용하기 위한 방법을 다음과 같이 기술한다.

임의의 사각형  $R=(R, C)$  을 고려해보자. 이 행렬의 각 열은 커널 큐브들을, 각 행은 이 커널에 대한 co-kernel 큐브들에 관련되어 있다. 이 사각형에서 위의 전력소모식을 적용하기 위하여 (그림 1)과 같은 리스트들을 사용한다. 기본적으로 각각의 커널, co-kernel 큐브들에 대하여 이 큐브들의 크기뿐 아니라 이 큐브들이 소속된 노드의 무게를 기록하며 또한 그 입력변수들의 스위칭 활동량을 기록한다.

rectangle value를 계산하기 위한 자료구조	
1. rectangle 의 각각의 co-kernel cubes (행렬의 행)에 대하여	
row_cost[]	- 리터럴 수
row_weight[]	- 큐브를 포함한 노드의 fanin 수
row_power[]	- switching activity
2. rectangle 의 각각의 kernel cubes (행렬의 열)에 대하여	
col_cost[]	- 리터럴 수
col_power[]	- switching activity

그림 1 rectangle value를 계산하기 위한 자료구조

(그림 2)는 임의의 사각형에 대해 적용되는 자료를 나타낸다.  $p(k)$ 는 논리식 또는 큐브  $k$ 에 대한 스위칭 활동량을 나타내며  $w(k)$ 는 논리표현  $k$ 가 포함된 노드의 무게를 나타낸다.

	$p(d_1)$	$p(d_2)$	...	$p(d_p)$
$p(q_1)$	$w(q_1)$			
$p(q_2)$	$w(q_2)$			
...	...			
$p(q_R)$	$w(q_R)$			

그림 2 임의의 사각형에 대해 적용되는 자료

	$p(d_1)$	$p(d_2)$	...	$p(d_p)$
$p(q_1)$	$w(q_1)p(q_1), w(q_1)p(d_1)$	$w(q_1)p(q_1), w(q_1)p(d_2)$	$w(q_1)p(q_1), w(q_1)p(d_p)$	
$p(q_2)$	$w(q_2)p(q_2), w(q_2)p(d_1)$	$w(q_2)p(q_2), w(q_2)p(d_2)$	$w(q_2)p(q_2), w(q_2)p(d_p)$	
...	...			
$p(q_R)$	$w(q_R)p(q_R), w(q_R)p(d_1)$	$w(q_R)p(q_R), w(q_R)p(d_2)$	$w(q_R)p(q_R), w(q_R)p(d_p)$	

그림 3

(그림 3)에서 보듯이  $(d_1 + \dots + d_p)(q_1 + \dots + q_R) = Dq_1 + \dots + Dq_R$ 와 같이 논리식이 변화하므로 없어지는 큐브들의 무게와 스위칭 활동량의 곱은 식(2)와 같이 나타낼 수 있다.

$$(P-1) \sum_{i=1}^R w(q_i)p(q_i) + W \sum_{i=1}^P p(d_i), \quad \text{---식(2)}$$

$$W = w(q_1) + \dots + w(q_R)$$

식(2)와 같이 변화하므로 그리고 새로 생겨나는 노드와 리터럴들의 스위칭 활동량과 무게의 곱은 새로 생겨나는 커널에 대한 노드와 이 노드의 입력에서 생기는 전력소모를 고려할 수 있다. 그러므로 이는 다음과 같이 나타낼 수 있다.

$$w(D) \sum_{i=1}^R p(d_i) + W p(D) \quad \text{---식(3)}$$

식(3)에서 첫째 항은 커널의 입력에 대한 전력소모로서 커널의 무게에 각 큐브의 스위칭 활동량을 곱한 것이며 둘째 항은 새로 도입된 노드의 리터럴이 각각의 co-kernel에 대해 적용되므로 새 노드인 커널 큐브들에 대해 커널의 무게를 곱한 것이다. 그러므로 다음과 같이 최종적인 전력소모식을 만들 수 있다.

$$(P-1) \sum_{i=1}^R w(q_i)p(q_i) + W \sum_{i=1}^P p(d_i)$$

$$- w(D) \sum_{i=1}^R p(d_i) - W p(D)$$

전력소모 감소를 위한 커널 추출은 다음과 같이 진행된다.

- 1) 모든 커널 set  $K$ 를 계산한다.
- 2)  $K$ 를 이용한 모든 커널 intersection  $D$ 를 계산한다.
- 3) 여기서 최대 전력감소를 가져올 부분 표현  $D_i \in D$ 을 선택한다.

#### 4. 실험 결과

본 논문의 실험을 위하여 위에 제시한 방법들을 SIS-1.2에 구현하여 그 결과를 비교하였다. 주로 본 논문에서 제안한 전력소모식을 SIS-1.2에 적용하고 이를 위해 사각형 커버링의 전력소모계산방법을 구현하였다. 실험에는 MCNC benchmark의 조합회로들을 이용하였으며 먼저 pla 형식으로 읽어서 이를 gkx 또는 gcx 등의 명령에 적용하였다. 실험결과의 전력소모는 SIS-1.2에서의 20MHz clock, Vdd=5V를 가정하고 마이크로 와트( $\mu W$ )로 계산하였다.

(그림 4)는 커널 추출을 위해 사용한 script를 보여

주며 여기에서 사용하는 다른 방법들은 gkx의 option으로 구현하여 서로 구분하였다.

```
read_pla a.pla
sim2 /* simplify SOP */
/* kernel 추출, 본 논문의 경우 -p option으로 구분함 */
gkx -b
ps /* 리터럴 수의 계산 */
pe -S /* power estimate */
```

그림 4 커널 추출을 위한 SIS-1.2 script

(표 2)는 커널 추출에 있어서의 본 논문의 구현결과를 보여준다. 전체적으로 리터럴수의 총합은 약 1% 정도 증가한 반면, 전력소모는 약 8% 정도 감소하였음을 보여준다. 그러나 연산시간은 아직까지는 매우 많음을 알 수 있다. 이것은 스위칭 활동량 계산을 위해 전체 회로의 BDD(Binary Decision Diagram)을 계산 때마다 매번 계산함으로써 인한 것이다. 본 논문에서는 이러한 BDD를 사용함에 있어서의 연산시간감소보다는 적절한 전력소모식의 제시와 저전력 설계를 위한 논리합성에 주안점을 두었으므로 상대적으로 연산시간감소에 주력하지 않았다. 저전력 설계에 있어서의 연산시간감소와 사용공간감소의 분야는 또 다른 연구분야로서 많은 연구가 진행되고 있으며 본 연구의 방법에 대한 연산시간은 추후 연구에 의해서 상당히 줄어들 수 있을 것으로 예상된다.

name	#literals			Power Consumption			Time(sec)	
	sis	*1	비율	sis	*1	비율	sis	*1
5xp1	155	157	1.01	775.2	767.2	0.98	1.07	5.77
apex6	1172	1181	1.00	5569.2	5151.6	0.92	14.66	245.63
apex7	363	372	1.02	1683.6	1578.9	0.93	10.88	42.63
b1	12	12	1.00	40.3	33.4	0.82	0.39	0.43
b9	163	164	1.00	721.9	697.2	0.96	1.46	7.54
bw	187	187	1.00	868.5	826.3	0.95	3.20	6.92
c8	189	190	1.00	926.2	827.7	0.89	0.86	6.54
cht	230	230	1.00	1234.7	937.2	0.75	1.06	6.83
cm138a	31	31	1.00	87.2	63.6	0.72	0.69	1.75
cm151a	30	30	1.00	141.6	132.8	0.93	0.47	0.95
cm152a	28	28	1.00	137.5	122.5	0.89	0.44	0.88
cm42a	34	34	1.00	110.7	80.7	0.72	0.68	1.74
cm82a	34	34	1.00	160.6	123.4	0.76	0.48	1.45
cm85a	65	65	1.00	279.3	272.1	0.97	0.66	3.77
con1	23	23	1.00	127.7	107.1	0.83	0.41	0.66
rd73	117	119	1.01	528.6	494.9	0.93	1.36	20.68
sao2	180	193	1.07	821.3	804.3	0.97	1.20	13.21
vg2	99	99	1.00	465.5	446.4	0.95	1.32	2.99
Sum	3112	3149	1.01	14679.6	13467.3	0.91	41.29	370.73

표 2 저전력설계를 위한 커널 추출에서의 리터럴과 전력소모 증감  
 sis : oroginal SIS-1.2, 전력소모 고려없음  
 \*1 : 본 논문에서 제안한 방법

Power consumption:  $\mu W$  assuming 20MHz clock and Vdd = 5V  
 리터럴수의 1%증가에 대비 전력소모 8% 감소함

### 5. 결론 및 연구방향

본 논문에서는 논리합성의 중요한 연산인 커널 추출에 있어서의 저전력 설계를 위하여 전력소모 예측함수의 설정, 공통표현추출방법과 전력소모 예측함수의 적용방법을 제시하고 이를 구현하여 그 실험결과를 알아 보았다. 일반적인 회로의 구성은 CMOS로 구성된다고 가정하였으며 따라서 전력소모는 출력노드의 정전용량과 스위칭 활동량에 의하여 좌우되고 이 두 가지의 요소가 적절히 반영되도록 무게합수를 이용하여 스위칭 활동량과 함께 전력소모식을 제시하였으며 이를 SIS-1.2에서의 커널 추출에 사용하는 사각형 커버링에 적용하는 방법을 제시하였다.

본 논문에서의 전력소모표현은 각 노드에서의 정전용량과 그 스위칭 활동량을 반영하되 노드의 표현 및 구현이 SOP(Sum of Products) 표현에 의한 complex gate로 이루어진다고 가정한다. 이러한 가정아래 정전용량은 노드의 입력 수에 비례한다고 가정하였으며 스위칭 활동량은 BDD를 구성하여 계산하는 방법을 사용하였다. 실험결과 비교적 적은 수(약 1%)의 리터럴 증가에 비하여 많은 전력소모 감소(약 8%)를 이루었으며 기존의 발표결과와의 비교에서도 더 나은 결과를 보였으며 본 논문결과를 토대로 ESOP(Exclusive Sum of Products)등 새로운 표현을 도입하면 더욱 발전될 수 있을 것으로 예상된다.

### 참고문헌

- [1] "SIS: A system for sequential circuit synthesis," Report M92/41, UC Berkeley, 1992.
- [2] K. Roy, S.C.Prasad, "Circuit Activity Based Logic Synthesis for Low Power Reliable Operations," IEEE Transactions of CAD, Dec. 1993, pp.503-513.
- [3] S. Iman, M. Pedram, "Multi-level Network Optimization for Low Power," ICCAD, 1994, pp. 372-377.
- [4] S. Iman, M. Pedram, "Logic Extraction and Factorization for Low Power," DAC, 1994, pp.248-253.
- [5] Rohfleisch, et. al, "Reducing Power Dissipation after Technology Mapping by Structural Transformations," DAC, 1996, pp.789-794.
- [6] S. Iman, M. Pedram, "POSE: Power Optimization and Synthesis Environment," DAC, 1996, pp.21-26.
- [7] R.K.Brayton, G.D. Hachtel, A. L. Sangiovanni-Vincentelli, "Multilevel Logic Synthesis," Proceedings of IEE, vol. 78, no. 2, 1990, pp.264-300.