

# 저전력 저면적의 논리 회로 설계를 위한 효율적인 커널 기반 분할 알고리즘

정희원 황선영\*, 김형\*\*, 최익성\*\*\*, 정기조\*

## An Efficient Kernel-based Partitioning Algorithm for Low-power, Low-area Logic Circuit Design

Sun-Young Hwang\*, Hyoung Kim\*\*, Ick-Sung Choi\*\*\*, Ki-Jo Jung\* *Regular Members*

### 요 약

본 논문에서는 조합 논리 회로의 면적과 전력 소모를 낮추기 위한 효율적인 커널 기반의 분할 알고리즘을 제안한다. 제안한 알고리즘은 커널을 이용하여 회로를 분할함으로써 회로의 전력 소모를 줄이고 분할된 회로들의 중복되는 게이트를 최소화시켜 면적 overhead를 감소시킨다. MCNC 표준 테스트 회로에 대한 실험을 통하여 제안된 알고리즘이 면적과 전력 소모면에 있어서 기존의 precomputation 회로 구조에 바탕을 둔 알고리즘에 비해 전력 소모는 평균 43.6%, 면적은 평균 30.7% 향상된 결과를 보인다.

### ABSTRACT

This paper proposes an efficient kernel-based partitioning algorithm for reducing area and power dissipation in combinational circuit design. The proposed algorithm decreases the power consumption by partitioning a given circuit utilizing a kernel, and reduces the area overhead by minimizing duplicated gates in the partitioned subcircuits. Experimental results for the MCNC benchmarks show that the proposed algorithm is effective by generating circuits consuming 43.6% less power with 30.7% less area on the average, when compared to the previous algorithm based on precomputation circuit structure.

### 1. 서론

최근 통신 및 회로 설계 기술의 발달과 함께 휴대용 전자 시스템의 수요가 급증함에 따라 전지에 의해 동작하는 휴대용 시스템의 설계에 관한 관심이 높아지고 있다. 휴대용 시스템은 다른 응용 시스템과는 달리 무게와 크기의 제약에 따라 한정된 전지 용량으로 동작시켜야 하는 특징이 있으며, 시스템의 동작시간이 휴대용 제품의 성공을 좌우하는 중요한 사양으로 등장하여 회로의 전력 소모를 낮

추는 문제가 주요한 설계 사양으로 부각되고 있다. 저전력 회로의 설계는 휴대용 시스템 뿐만 아니라 고성능 회로의 동작속도를 제한하는 방대한 발열량을 감소시키기 위해 그 중요성이 점차 증가하고 있으며, 이를 해결하기 위하여 저전력 회로 설계에 대한 연구가 활발하게 진행되고 있다. CMOS 논리 회로의 전력 소모는 회로의 스위칭 활동(switching activity)에 의한 전력 소모가 대부분을 차지하므로 회로의 평균 스위칭 활동을 확률로 계산하여 CMOS 논리 회로의 평균 전력 소모를 구하며<sup>[1]</sup>

\* 서강대학교 전자공학과

\*\* 경민대학 소프트웨어개발과

\*\*\* 한국전자통신연구원 교환전송기술연구소

논문번호 : 00013-0113, 접수일자 : 2000년 1월 13일

CMOS 논리 게이트를 이용하여 설계한 회로의 전력 소모는 식(1)과 같이 주어진다<sup>[2]</sup>.

$$P_{avg} = \frac{1}{2} \cdot C_{load} \cdot V_{dd}^2 / T_c \cdot E(transitions) \quad (1)$$

여기서  $C_{load}$ 는 출력단의 capacitance,  $V_{dd}$ 는 공급 전압,  $T_c$ 는 클럭 주기를 나타내며  $E(transitions)$ 는 클럭 사이클당 평균 스위칭 활동을 의미한다. 전력 소모를 적게 하기 위한 최적화는 다양한 수준의 설계 계층(design hierarchy)에 적용될 수 있다. High-level 설계 단계에서 IC의 전력 소모는 구조적 변환(architectural transformation) 방법에 의해 감소될 수 있고<sup>[3]</sup>, circuit-level에서는 전력 소모를 줄이기 위해 게이트 크기 조정(gate resizing)방법이 제안되었으며, 이는 게이트 slack을 정확히 측정하여 게이트 크기를 줄여 전력 소모를 감소시킨다<sup>[4]</sup>. Logic-level에서는 전력 소모 최소화의 초점이 스위칭 활동이므로 이 점에 착안하여 조합 논리 회로의 전력 소모를 감소시키는 여러 방법이 제안되었다. 회로의 논리 함수를 대상으로 product 항의 수와 스위칭 활동을 최소화하기 위해 don't-care set을 이용하거나<sup>[5][6]</sup> 저전력 소모를 갖는 multi-level 회로를 발생하기 위하여 커널 추출 방법을 사용한다<sup>[7]</sup>. 커널 추출 방법은 주어진 논리 함수들의 커널을 각각 계산하고 계산된 커널을 중심으로 교집합을 구하여 common subexpression을 추출하는 다단 논리 최적화를 통해 회로의 노드에서 중복으로 발생할 수 있는 스위칭 활동을 줄인다. 회로의 전체 스위칭 활동을 최소화하기 위해 기술 분할(technology decomposition) 및 기술 매핑 방법이 제안되었으며<sup>[8]</sup>, 기술 분할에서는 회로를 primitive 게이트 중심으로 분할하여 분할된 회로의 내적인 노드의 스위칭 활동을 최소화한다. 기술 매핑에서는 NAND 분할 트리를 이용하는데, 2개의 tree-traversal 오퍼레이션인 postorder traversal과 preorder traversal을 사용하며 postorder traversal에서 트리의 루트에 도착 가능한 시간을 정하며, preorder traversal에서는 요구되는 시간 제약 조건하에서 전력 소모를 최소화하는 매핑 해를 결정한다. 논리 최적화된 회로에 이 기술 분할 방법을 적용할 경우 전력 소모를 감소시키는데 한계가 있다. 논리 회로에 precomputation 논리를 구현하기 위한 회로를 추가하여 전력 소모를 감소시키는 방법이 제시되었다<sup>[9]</sup>. Precomputation 논리를 이용하는 경우 이 논리를 구현하기 위해서는 논리 회로에서의 여러 입력 가운데 특정 입력들을

정하여야 하며 이 입력들을 선택하기 위해 입력 선택 알고리즘이 제안되었다<sup>[9]</sup>. 이 알고리즘은 주어진 회로의 여러 입력 가운데 알고리즘에 의해 선택된 입력 집합을 중심으로 예상되는 출력 값을 미리 계산하여 이 값에 따라 이어지는 클럭 주기에서는 내적인 스위칭 활동을 감소시켜 회로의 전력 소모를 줄이는 방법을 사용한다. 이 알고리즘의 단점으로는 주어진 본래 회로에 precomputation 논리를 구현하기 위한 회로가 추가됨으로 인해 회로 면적과 클럭 주기가 증가된다는 점이다. 본 논문에서는 회로의 공통되는 부분인 커널을 근거로 회로를 2개의 세부 회로로 분할한 후 커널에 따라 세부 회로가 선택되어 수행되는 구조로 회로를 변환하여 기존의 precomputation 논리 알고리즘의 면적이 증가되는 단점을 개선하고, 더 나아가 면적을 추가로 개선하기 위하여 분할된 회로에서 중복되는 부분을 추출하여 회로를 구현하였다. 2 장에서는 기존의 저전력 소모를 위한 알고리즘을 제시하고, 3 장에서는 제안한 알고리즘을 자세히 설명하고, 4 장에서는 실험 결과를 통하여 제안한 알고리즘의 효율성을 보이며 마지막으로 결론을 보인다.

## II. 기존의 저전력 소모 논리 회로 구조

Precomputation 논리에 바탕을 둔 저전력 소모 알고리즘은 게이트 레벨 기본 회로의 변형된 회로 구조에 근거한다. 그림 1은  $L_1$ 와  $C_1$  불력으로 구성된 원래의 회로에 대해 precomputation 논리 구조로 변환한 것으로 여기서  $g_1$ ,  $g_2$ 는 논리 함수로서 일종의 예측(predict) 함수이다<sup>[9]</sup>. 이 함수들은 출력 함수  $f$ 의 입력으로써 표시되며  $g_1 = 1$ 이면  $f = 1$ 이 되고  $g_2 = 1$ 이면  $f = 0$ 이 되는 조건을 만족시킨다. 클럭 주기  $t$  동안  $g_1$  또는  $g_2$ 가 1이 되면 래치  $L_1$ 의 load-enable(LE) 신호는 0이 되어 클럭 주기  $t+1$  동안 조합 논리 불력  $C_1$ 의 입력은 바뀌지 않게 된다. 같은 클럭 주기 동안에는 제시된 조건에 의하여  $g_1$ 과  $g_2$ 가 동시에 1이 될 수는 없으며,  $g_1 + g_2$ 의 입력 조건에 따라 불력  $C_1$ 의 입력이 스위칭 활동이 일어나지 않게 되어 불력  $C_1$ 의 전력 소모가 감소된다.

그림 1에서 제시한 회로의 불력  $C_1$ 의 입력 신호 집합  $I = \{i_1, \dots, i_n\}$ 에 대한 논리 함수  $f(i)$ 로 가정할 때 입력 변수  $i_j$ 에 대한 observability don't-care set (ODC)은  $ODC_j = f_{i_j} \cdot \overline{f_{\overline{i_j}}} + \overline{f_{i_j}} \cdot f_{\overline{i_j}}$  이 된다<sup>[9]</sup>.

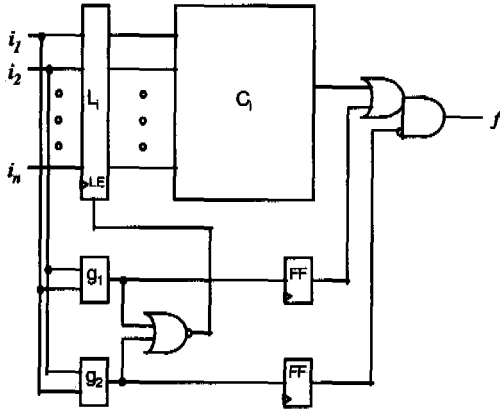


그림 1. Precomputation 논리를 이용한 회로

여기서  $f_{i_1}$  와  $f_{\bar{i}_1}$  는  $i_1$  에 대한  $f$  의 cofactor이고  $\bar{f}_{i_1}$  와  $\bar{f}_{\bar{i}_1}$  는  $\bar{f}$  의 cofactor이다. 주어진 입력 부분 집합이 ODC<sub>i</sub>에 있다면  $i_j$ 가 래치로 load되지 않게 할 수 있다. 입력  $i_m, i_{m+1}, \dots, i_N$ 을 래치에 load하지 않을 경우 함수  $g$ 는  $g = \prod_{j=m}^N ODC_j$ 로 표현된다<sup>[9]</sup>. 최적의 입력 변수를 선택하는 알고리즘과 출력 함수가 여러 개 존재할 경우 이 함수들 중에서 precomputation 논리를 적용할 출력 함수의 부분 집합을 선택하는 알고리즘은 이미 제안되었으며, precomputation 논리에 필요한 입력의 수는 전체 입력의 수보다 훨씬 적어야 좋은 접근 방법이 된다.

제어 변수(control variable)에 근거한 저전력 소모 알고리즘은 Shannon expansion을 이용한다. 모든 논리 함수는 Shannon expansion으로 표시할 수 있으며 입력 신호 집합이  $I = \{ i_1, \dots, i_n \}$ 인 함수  $f$ 에 대해 입력 변수  $i_1$ 에 대한 전개는  $f = i_1 f_{i_1} + \bar{i}_1 f_{\bar{i}_1}$ 이 된다. 여기서  $f_{i_1}, f_{\bar{i}_1}$ 는  $i_1$ 에 대한  $f$ 의 cofactor이다. 논리 함수  $f$ 에 대해 Shannon expansion을 수행하면 그림 2의 구조를 갖는 회로로 변환된다<sup>[9][10][11]</sup>. 입력 변수  $i_1$ 의 값에 따라 입력 래치  $L_1, L_2$  중 한 래치의 load-enable 신호를 세팅함으로써 cofactor  $f_{i_1}, f_{\bar{i}_1}$  중 한 cofactor는 disable되고 다른 한 cofactor는 계산되어 스위칭 활동은 감소한다. 입력  $i_1$ 은 해당 cofactor를 계산하는 멀티플렉서의 제어 신호로 작동한다. 선택된 입력 변수에 대해 Shannon expansion을 수행하여 세부 회로내의 게이트 수를 줄일 수 있다면 회로의 전력 소모를 보다 줄일 수 있다. 따라서 논리 함수가 주어졌을 때 제어 변수인 최적의 입력 변수를

선택하는 알고리즘이 주요 관건이 된다.

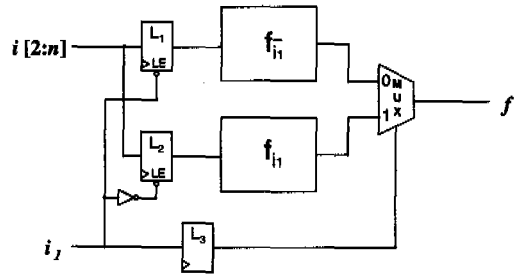


그림 2. Shannon expansion을 이용한 회로

### III. 회로 최적화를 위한 커널 기반 분할 알고리즘

기존의 precomputation 논리에 근거를 둔 저전력 소모 알고리즘을 적용하여 회로를 구현한 경우 precomputation 논리를 구현하기 위한 회로가 원래 회로에 추가되어야 하므로 경로 지연시간이 증가되어 non-critical 회로 섹션에 이 알고리즘을 적용하는 것이 바람직하며, precomputation 논리 회로가 전력 소모 면에서 효율적이기 위해서는 이 논리를 구현함으로써 발생하는 추가 회로로 인한 전력 소모 증가, 회로 면적 및 지연 시간의 증가 등이 최소의 비용을 발생하도록 추가 회로를 구현하여야 한다. 따라서 함수  $f$ 의 입력들 중에서 precomputation 논리를 구현할 수 있는 최소한의 입력 변수를 선택하는 것이 중요하면서도 어려운 작업이 된다. 제어 변수에 근거한 저전력 소모 알고리즘<sup>[11]</sup>은 그 근거가 되는 회로 구조가 기존의 알고리즘의 회로 구조에 비해 간단하고, disable되는 입력이 기존의 알고리즘과 달리 observability don't-care set에 있을 필요가 없게 되며 또한 함수  $f$ 의 여러 입력 신호 중에서 한 입력 변수를 선택하는 일이 기존의 알고리즘보다 용이하다. 그러나 이 알고리즘에 근거한 회로 구현시 제어 변수를 기준으로 Shannon expansion을 수행하여 양분화된 회로내에 제어 변수에 영향을 받지 않는 공통 회로가 여러 개 존재하게 되어 전력은 감소하나 회로 면적은 오히려 증가하는 경우가 발생한다. 본 논문에서 제안한 커널 기반 분할 알고리즘은 회로 최적화를 위한 커널을 선택하여 회로를 분할하고, 분할된 회로내에 공유되지 않는 공통 회로를 추출한다. 회로의 분할시 둘 혹은 그 이상의 함수에서 공유하는 공통 부분 회로를 효율적으로 찾기 위하여 커널을 사용한다<sup>[12]</sup>. 커널은 회로내에서 공유되는 공통되는 부분을 나타낼 수 있

는 가장 기본적인 단위이므로, 회로를 나누는 효율적인 divisor가 될 수 있어 저전력 소모 합성시 전력 과 면적의 개선이 가능하다<sup>[12]</sup>. 회로를 분할하는 공통 인수를 어떻게 선택하느냐에 따라 면적, 전력과 속도에 영향을 미치므로, 커널의 선택에 사용되는 비용 함수를 적절히 설계해야 한다. 제안한 설계 방법은 predictor 함수 계산을 위해 ODC가 필요하지 않으므로 합성에 적절한 ODC가 존재하지 않는 회로에도 합성할 수 있는 장점이 있다. 주어진 회로에 대해 최적의 커널을 선택하여 이 커널을 중심으로 회로를 분할하는 알고리즘에 근거한 회로 구조를 그림 3에 보인다.  $I_1$  과  $I_2$  는 각각 회로의 전체 입력 집합과 선택된 커널  $K$ 를 합성한 회로의 입력 집합을 나타낸다. 선택된 커널을 근거로 원래 회로의 논리 함수  $f$ 를 표현하면  $f = Kf_K + \bar{K}f_{\bar{K}}$ 으로 표현되며  $f_K$ 와  $f_{\bar{K}}$ 는 코커널(cokernel)으로서 선택된 커널  $K$ 에 대하여 회로를 분할하여 얻어진 세부 회로이며,  $f_C$ 는 분할된 회로들에서 추출하되 커널에 의해 영향을 받지 않는 공통 회로이다. 회로내 래치의 load-enable 신호는 커널을 합성한 회로( $K$ )의 출력 값에 따라 결정되며, 출력은 공통 회로인  $f_C$ , 그리고 두 가지 세부 회로  $f_K, f_{\bar{K}}$  중 커널에 따라 동작되는 세부 회로에 의해 결정되는데 이 세부 회로는 회로 출력 단에 멀티플렉서 혹은 transmission 게이트를 이용하여 선택된다. 출력 단의 멀티플렉서를 transmission 게이트를 사용하여 wired-OR 논리 구조로 대체하는 경우, 각 출력 당 하나의 transmission 게이트로 구현이 가능하여 멀티플렉서를 게이트로 구현한 경우보다 전력 소모량이 줄어드는 장점이 있다. 그림 3과 같이 합성된 회로에서 소모되는 전력을 최소화 하기 위해 최적의 커널을 선택하는 알고리즘이 중요한 변수가 된다.

그림 4에 주어진 회로에서 최적의 커널을 선택하는 알고리즘을 제시하였다. 먼저 주어진 회로  $F$ 에 대해 BDD를 생성하고, 이 BDD를 근거로 회로에서 구해 질 수 있는 모든 커널을 추출한 다음 각 커널을 중심으로 회로  $F$ 를 2개의  $f_K, f_{\bar{K}}$  회로로 분할한 후, 분할된 회로들을 대상으로 커널에 의해 영향을 받지 않는 공통 회로  $f_C$ 를 구한다. 결국 주어진 회로는 커널을 합성한 회로, 분할된 회로들 중 공통 회로가 제외된 세부 회로, 공통 회로로 구성된다. 따라서 회로의 전체 전력 소모는 커널을 합성한 회로의 전력 소모, 커널을 합성한 회로의 신호 확률

$p_s(i)$ 에 좌우되는 세부 회로  $f_K$ 에서의 전력 소모와 신호 확률  $1 - p_s(i)$ 에 좌우되는 세부 회로  $f_{\bar{K}}$ 에서의 전력 소모를 포함한 값으로 구해진다. 이 전력 소모를 가장 작게 가지는 커널을 최적의 커널로 선택하여 회로를 분할한다. 그림 5 (a)는 MCNC 벤치마크 회로 중 b1 회로 예제를 보인다. 이 회로는 sis1.2<sup>[13]</sup>

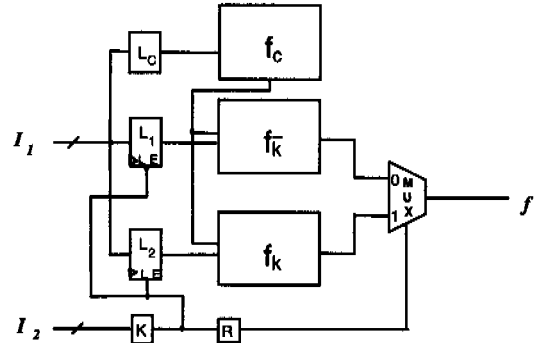


그림 3. 커널 기반 분할 알고리즘에 의해 생성된 회로

에서 script.rugged를 이용하여 최적화한 회로로 45  $\mu W$ 의 전력을 소모하며 면적을 나타내는 리터럴의 갯수는 15이다. 이 예제를 그림 1에 보인 pre-computation 구조로 합성할 경우 선택한 입력 집합이  $\{a, b\}$ 이고 예측 함수가  $g = ab + \bar{a}\bar{b}$  일 때 전력 소모가 40  $\mu W$  인 회로를 그림 5 (b)와 같이 얻을 수 있다. 예제 회로를 그림 3의 회로 구조에 근거한 Shannon expansion 알고리즘을 적용하여 회로를 합성한 경우 그림 5 (c)와 같은 회로를 얻을 수 있으며 입력  $c$ 를 선택했을 때 전력 소모가 23  $\mu W$ 로 감소한다. 그림 3의 회로 구조를 바탕으로 예제 회로에 대하여 제안한 커널 기반 분할 알고리즘을 사용하여 합성할 경우 커널  $(a+b)$ 를 선택할 때 최적의 결과를 얻을 수 있으며, 이 회로를 그림 5 (d)에 나타내었다. 이 때 소모되는 전력은 15  $\mu W$  이므로 전력 소모가 효율적으로 감소하였음을 확인할 수 있다. 이어 리터럴의 갯수를 비교할 경우 15 개의 리터럴을 갖는 본래 회로와 비교하면 그림 5 (b)의 precomputation 구조 회로가 19 개, 그림 5 (c)의 Shannon expansion 알고리즘에 근거한 회로가 18 개, 그림 5 (d)의 제안한 커널 기반 분할 알고리즘에 근거한 회로가 7 개이다. 따라서 제안한 커널 기반 분할 알고리즘이 다른 알고리즘에 비해 면적과 전력 소모면에서 가장 효율적임을 보인다.

```

Kernel-based_Partitioning( F = { f1, ..., fn } )
/* F : 출력 f1, ..., fn을 갖는 다중 출력 회로
Kernels : 추출된 커널들의 집합
P(i), P(fc), P(fK), P(fK̄) :
    각 분할된 회로의 전력 소모
ρs(i) : 커널 회로 출력 노드 i의 신호 확률 */
{
    주어진 회로 F에 대해 BDD를 생성;
    Min_Power = ∞;
    Kernels = BDD로부터 추출된 커널 집합;
    for each k ∈ Kernels {
        커널에 대해 회로 F를 분할하여 fK, fK̄ 및 fc 추출;
        Power(k) =
            P(i) + P(fc) + ρs(i) * P(fK)
            + (1 - ρs(i)) * P(fK̄) ;
        if (Power(k) < Min_Power)
            Min_Power = Power(k);
    }
    k = Min_Power 값을 갖는 커널;
    Kernel k를 이용하여 회로를 분할;
}
    
```

그림 4. 커널 기반 분할 알고리즘.

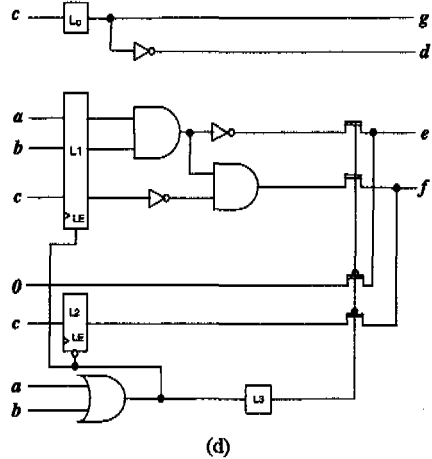
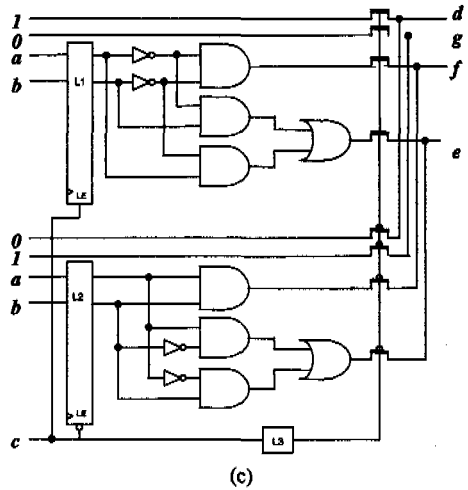
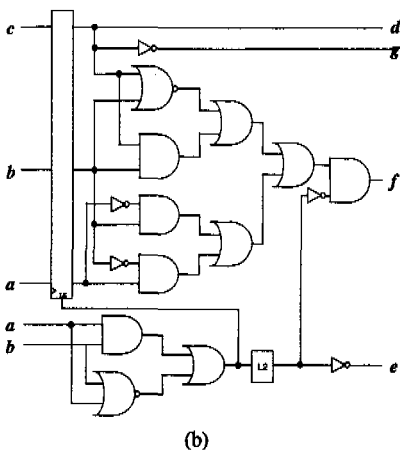
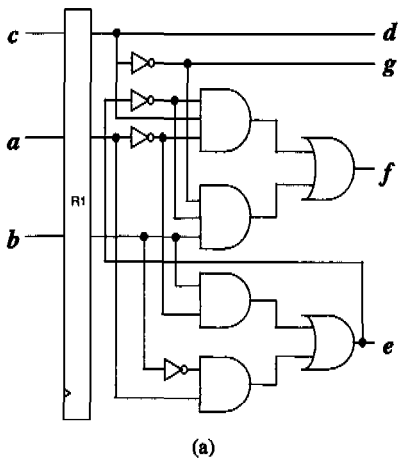


그림 5. 예제 회로의 저전력 설계를 위한 다양한 회로  
 (a) 본래의 조합 논리 회로  
 (b) Precomputation 논리를 이용한 회로  
 (c) Shannon expansion 알고리즘을 이용한 회로  
 (d) 커널 기반 분할 알고리즘을 이용한 회로

#### IV. 실험 결과

제안한 커널 기반 분할 알고리즘은 SUN UNIX 환경하에서 C 언어로 구현되었다. 제안한 커널 기반 분할 알고리즘은 기존의 sis 시스템의 면적 최적화 알고리즘<sup>[13]</sup>, precomputation 알고리즘<sup>[9]</sup> 및 Shannon expansion 알고리즘<sup>[11]</sup>과 결과를 비교하였다. 전력 소모 측정은 sis 시스템의 전력 소모 예측 기능을 이용하되 무지연 시간(zero-delay) 모델 가정 하에 20 MHz의 클럭 주파수, 5 V의 공급 전압을 사용하였다. 전력 소모 측정 시 무지연 시간 모델을 사용한 이유는 제안된 알고리즘이 기술 독립적인

레벨에서 노드의 스위칭 활동을 줄임으로써 전력 소모를 줄이는 알고리즘이므로 전이 활동 감소에 따른 전력 소모 감소 효과를 효율적으로 반영할 수 있기 때문이다. 실험에 사용한 MCNC 벤치마크 회로를 sis 1.2 script의 script.rugged를 사용하여 최적화하여 표 1에 제시했으며 표에서 '#ins' 열은 입력 개수, '#outs' 열은 출력 개수, 'Area(#lits)' 열은 리터털 수를 나타내는 면적을 나타낸다. Sis 시스템을 이용한 최적화 방법, precomputation 논리 방법, Shannon expansion 알고리즘 방법, 커널 기반 분할 알고리즘 방법에 의해 구해진 회로의 전력 소모는 표 2, 회로의 면적은 표 3에 제시하며 표 2에서 '△Power(%)' 열은 제안한 알고리즘이 sis를 이용한 방법, precomputation 논리 방법 및 Shannon expansion 알고리즘을 이용한 방법과 비교한 전력 소모의 차를 나타낸다. 표 3에서 '△Area(%)' 열은 면적의 차를 나타낸다. 전력 소모를 나타내는 실험 결과 표 2에서 cm162a 회로, 9symm1 회로는 precomputation 논리 시스템보다 제안한 커널 기반 분할 알고리즘을 적용할 경우 평균 73.7% 이상의 전력 소모 감소를 초래하고 sis 1.2를 이용한 회로에 비해서는 평균 53.9% 이상 전력 소모가 감소한다. 이 회로들의 경우 선택된 커널을 중심으로 회로를 분할할 때 분할된 회로의 크기가 많이 감소하여 전력 소모가 크게 줄어든다. 표 2의 실험 결과에 따르면 커널 기반 분할 알고리즘을 적용한 경우 precomputation 논리 시스템보다 평균적으로 43.6%, sis로 최적화한 회로보다는 평균 53.9% 정도 전력 소모 감소를 보이며 Shannon expansion 알고리즘을 적용한 회로에 비해서는 평균 3.8%의 전력 소모 감소를 보인다. 표 3의 실험 결과에 따르면 면적은 제안한 커널 기반 분할 알고리즘을 적용해 구현한 회로들이 precomputation 논리가 적용된 회로들보다 평균 30.7% 정도 감소하고 Shannon expansion 알고리즘을 적용한 회로들보다 평균적으로 41.1% 감소하며 sis 적용 회로들에 비해서는 평균 20.7% 정도 감소함을 보인다. Shannon expansion 알고리즘을 적용한 회로들에 비해 제안한 커널 기반 분할 알고리즘을 적용할 경우 면적이 더 감소하는 것은 제어 변수를 근거로 회로가 분할된 후 분할된 회로들이 제어 변수에 영향을 받지않는 게이트를 공통으로 포함하고 있기 때문이다. 표 2에서 전력 소모가 크게 감소한 회로인 cm162a 회로, 9symm1 회로가 면적면에서 44.3% ~ 49.3% 정도 감소됨으로 나타나, 결국 제안한 알고리즘을 적용하여 회로가

분할된 후 분할된 회로의 크기가 줄어 전력 소모 감소의 효과를 얻게 됨을 알 수 있다. 표 4에서는 제안한 커널 기반 분할 알고리즘을 적용하되 벤치마크 회로를 대상으로 sis 1.2의 mcnc 라이브러리를 이용하여 기술 매핑 후 지연 시간(general delay) 모델 환경하에서 전력 소모 및 면적을 측정한 결과를 나타낸다. 벤치마크 회로에 비해 전력 소모 면에서는 평균 55.7% 정도 감소하며 면적의 경우에는 평균 19.4% 정도 감소하여 성능이 향상됨을 보인다.

표 1. Sis에 의해 면적 최적화된 벤치마크 회로

Circuits	#ins	#outs	Power(μW)	Area(#lits)
b1	3	4	45	15
cc	21	20	182	79
cht	47	36	526	261
cm42a	4	10	111	34
cm82a	5	3	101	36
cm138a	6	8	87	31
cm162a	14	5	267	91
cmb	16	4	234	62
cu	14	11	234	78
duke2	22	29	1191	642
f51m	8	8	590	279
majority	5	1	64	18
misex2	25	18	363	128
pcl	19	9	692	109
pcler8	27	17	283	133
sao2	10	4	611	210
sct	19	13	321	114
x2	10	7	228	68
z4ml	7	4	208	64
9symm1	9	1	1058	299

## V. 결론

본 논문에서는 전력 소모 및 회로 면적을 감소시키는 논리 회로 합성을 위한 커널 기반 분할 알고리즘을 제안하였다. 제안한 커널 기반 분할 알고리즘은 모든 논리 회로에 적용할 수 있는 알고리즘으로 우선 주어진 논리 회로에 제안한 알고리즘을 적용하여 최적의 커널을 선택하고, 이 커널에 대해 회로를 분할하고 분할된 회로들에서 커널에 의해 영향을 받지 않는 공통 회로를 추출하는 방법으로 회로를 합성한다.

표 2. 전력 소모에 대한 비교

Circuits	Power( $\mu$ W)				$\Delta$ Power(%)		
	sis	Precomp. logic	Shannon expansion	Proposed	Proposed vs sis	Proposed vs Precomp. logic	Proposed vs Shannon expansion
b1	45	40	23	15	-66.7	-62.5	-34.8
cc	182	182	93	103	-43.4	-43.4	10.8
cht	526	415	306	442	-16.0	6.5	44.4
cm42a	111	108	51	59	-46.8	-45.4	15.7
cm82a	101	112	69	60	-40.6	-46.4	-13.0
cm138a	87	57	49	24	-72.4	-57.9	-51.0
cm162a	267	159	87	61	-77.2	-61.6	-29.9
cmb	234	174	156	62	-73.5	-64.4	-60.3
cu	234	150	108	84	-64.1	-44.0	-22.2
duke2	1191	1031	720	444	-62.7	-56.9	-38.3
f51m	590	468	348	279	-52.7	-40.4	-19.8
majority	64	39	28	34	-46.9	-12.8	21.8
misex2	363	433	191	215	-40.8	-50.3	12.6
pcl	692	402	91	122	-82.4	-69.7	34.1
pcler8	283	182	103	128	-54.8	-29.7	24.3
sao2	611	402	417	530	-13.3	31.8	27.1
sct	321	297	167	226	-29.5	-23.9	35.3
x2	228	232	125	106	-53.5	-54.3	-15.2
z4ml	208	214	102	84	-59.6	-60.7	-17.6
9symml	1058	1312	191	191	-81.9	-85.4	0.0
Average	370	320	171	163	-53.9	-43.6	-3.8

표 3. 면적에 대한 비교

Circuits	Area(#lits)				$\Delta$ Area(%)		
	sis	Precomp. logic	Shannon expansion	Proposed	Proposed vs sis	Proposed vs Precomp. logic	Proposed vs Shannon expansion
b1	15	19	18	7	-53.3	-63.2	-61.1
cc	79	123	82	83	5.1	-32.5	1.2
cht	261	262	412	233	-10.7	-11.1	-43.4
cm42a	34	46	48	30	-11.8	-34.8	-37.5
cm82a	36	38	53	30	-16.7	-21.1	-43.4
cm138a	31	51	56	27	-12.9	-47.1	-51.8
cm162a	91	106	69	59	-35.2	-44.3	-14.5
cmb	62	56	156	37	-40.3	-33.9	-76.3
cu	78	89	74	65	-16.7	-27.0	-12.2
duke2	642	704	789	559	-12.9	-20.6	-29.2
f51m	279	207	220	151	-45.9	-27.1	-31.4
majority	18	24	39	13	-27.8	-45.8	-66.7
misex2	128	225	219	115	-10.2	-48.9	-47.5
pcl	109	112	158	103	-5.5	-8.0	-34.4
pcler8	133	189	210	117	-12.0	-38.1	-44.3
sao2	210	242	337	209	-0.5	-13.6	-38.0
sct	114	122	195	115	0.9	-5.7	-41.0
x2	68	71	117	57	-16.2	-19.7	-51.3
z4ml	64	59	107	46	-28.1	-22	-41.8
9symml	299	211	184	107	-64.2	-49.3	-43.4
Average	138	148	177	108	-20.7	-30.7	-41.1

표 4. 기술 매핑 후의 전력 소모 및 면적 비교

Circuits	#ins/#outs	Power( $\mu$ W)			Area(#lits)		
		sis	Proposed	$\Delta$ Power (%)	sis	Proposed	$\Delta$ Area (%)
b1	3 / 4	63	15	-76.2	23	7	-69.6
cc	21 / 20	326	153	-53.1	125	136	8.8
cht	47 / 36	753	639	-15.1	320	324	1.3
cm42a	4 / 10	115	64	-44.3	39	37	-5.1
cm82a	5 / 3	111	57	-48.6	44	33	-25.0
cm138a	6 / 8	90	27	-70.0	36	32	-11.1
cm162a	14 / 5	279	69	-75.3	99	76	-23.2
cmb	16 / 4	244	62	-74.6	86	51	-40.7
cu	14 / 11	273	84	-69.2	103	82	-20.4
duke2	22 / 29	1330	441	-66.8	780	678	-13.1
f51m	8 / 8	488	320	-34.4	188	197	4.8
majority	5 / 1	56	36	-35.7	21	17	-19.0
misex2	25 / 18	399	211	-47.1	171	140	-18.1
pcl	19 / 9	319	135	-57.7	156	131	-16.0
pcler8	27 / 17	397	158	-60.2	196	172	-12.2
sao2	10 / 4	661	335	-49.3	273	277	1.5
sct	19 / 13	331	237	-28.4	138	137	-0.7
x2	10 / 7	239	111	-53.6	81	70	-13.6
z4ml	7 / 4	261	85	-67.4	97	50	-48.5
9symml	9 / 1	1127	159	-85.9	395	124	-68.6
Average		393	170	-55.7	169	139	-19.4

합성된 회로는 커널에 의해 영향을 받지 않는 공통 회로와 커널에 좌우되는 두 세부 회로로 분할되는데, 공통 회로는 항상 동작하나 회로 크기가 전반적으로 작고 나머지 세부 회로는 커널에 의해 한 세부 회로만 구동되고 다른 세부 회로는 disable되어 그만큼 세부 회로의 스위칭 활동은 감소하게 되므로 회로에서 발생하는 전력 소모는 감소하며 세부 회로에서 중복되는 게이트를 최소화시킴으로써 회로 면적은 감소한다. 실험 결과에서 커널 기반 분할 알고리즘을 적용해 합성된 회로들이 저전력 소모를 위한 precomputation 논리에 근거한 입력 선택 알고리즘과 비교할 때 전력 소모와 회로 면적면에서 효율적이며 Shannon expansion 알고리즘과 비교할 경우에는 면적면에서 효율적임을 보인다. 더 나아가 커널 기반 분할 알고리즘을 사용하여 설계 작업을 수행할 경우 빠른 시간 내에 적절한 전력 소모 및 회로 면적을 갖는 회로를 합성할 수 있어 최적의 회로를 구현하기 위한 설계 자동화의 효율을 높일 수 있다.

감사의 글

본 연구는 1998년도 한국학술진흥재단 연구비 지

원(과제번호: 1998-01-ED0846 )에 의하여 수행되었습니다.

참고 문헌

- [1] A. Ghosh, S. Devadas, K. Keutzer, and J. White, "Estimation of Average Switching Activity in Combinational and Sequential Circuits," in Proc. 29th DAC, pp. 253-259, June 1992.
- [2] R. Bahar and F. Somenzi, "Boolean Techniques for Low Power Driven Synthesis," in Proc. ICCAD, pp. 428-432, Nov. 1995.
- [3] A. Chadrakasan, M. Potkonjak, R. Mehra, J. Rabaey, and R. Brodersen. "Optimizing Power Using Transformations," IEEE Trans. on CAD, Vol. 14, No. 1, pp. 12-31, Jan. 1995.
- [4] R. Bahar, G. Hachtel, E. Macii, and F. Somenzi, "A Symbolic Method to Reduce Power Consumption of Circuits Containing False Paths," in Proc. ICCAD, pp. 368-371, Nov. 1994.
- [5] A. Shen, S. Devadas, and A. Ghosh,



"Probabilistic Construction and Manipulation of Free Boolean Diagram," in Proc. ICCAD, pp. 544-549, Nov. 1993.

[6] S. Iman and M. Pedram, "Multi-level Network Optimization for Low Power," in Proc. ICCAD, pp. 372-377, Nov. 1994.

[7] S. Prasad and K. Roy, "Circuit Activity Driven Multilevel Logic Optimization for Low Power Reliable Operation," in Proc. European Conf. on Design Automation," pp. 368-372, Feb. 1993.

[8] C.-Y. Tsui, M. Pedram, and A. Despain, "Technology Decomposition and Mapping Targeting Low Power Dissipation," in Proc. 30th DAC, pp. 68-73, June 1993.

[9] M. Alidina, J. Monteiro, S. Devadas, and A. Ghosh, "Precomputation-based Logic Optimization for Low Power," in Proc. ICCAD, pp. 74-81, Nov. 1994.

[10] G. De Micheli, *Synthesis and Optimization of Digital Circuits*, McGraw-Hill, 1994.

[11] H. Kim, I.S. Choi, and S.Y. Hwang, "Design of Heuristic Algorithms Based on Shannon Expansion for Low-power Logic Circuit Synthesis," IEE Proc.-Circuits Devices Syst., Vol. 144, No. 6, Dec. 1997.

[12] S. Iman and M. Pedram, "Logic Extraction and Factorization for Low Power," in Proc. 32nd DAC, pp. 248-253, June 1995.

[13] E. Sentovich, H. Savoj, R. Brayton, and A. Sangiovanni-Vincentelli, "SIS: A System for Sequential Circuit Synthesis," Memorandum No. UCB/ERL M92/41, Electronic Research Laboratory, University of California, Berkeley, May 1992.

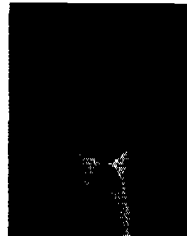
**황 선 영(Sun-Young Hwang)**



1976년 2월 : 서울대학교  
전자공학과 졸업.  
1978년 2월 : 한국과학원 전기  
및 전자공학과  
공학 석사 취득.  
1986년 10월 : 미국 Stanford 대학  
공학박사 학위 취득.

1976년~1981년 : 삼성 반도체 주식회사 연구원.  
1986년~1989년 : Stanford대학 Center for Integrated  
Systems 연구소 책임 연구원, Fairchild  
Semiconductor Palo Alto Reserch Center  
기술 자문.  
1989년~1992년 : 삼성전자(주) 반도체 기술자문.  
1989년 3월~현재 : 서강대학교 전자공학과 부교수.  
<주관심 분야> CAD시스템, Computer Architecture  
및 Systems Design, VLSI 설계등임

**김 형(Hyoung Kim)**



1979년 2월 : 서강대학교  
전자공학과 졸업  
1981년 2월 : 서강대학교  
전자공학과  
공학석사 취득  
1997년 8월 : 서강대학교  
전자공학과  
공학박사 학위 취득  
1983년 11월 : 럭키금성 그룹 기획조정실 근무  
1985년 1월 : 럭키엔지니어링 근무  
1987년 3월 : 금성소프트웨어 근무  
1992년 3월~현재 : 경민대학 소프트웨어개발과 부  
교수  
<주관심 분야> CAD 시스템, Logic Synthesis for  
Low Power, Computer Architecture 등임.

**최 익 성(Ick-Sung Choi)**



1992년 2월 : 서강대학교  
전자공학과 학사 취득.  
1994년 2월 : 서강대학교 대학원  
전자공학과  
석사학위 취득.  
1999년 2월 : 서강대학교 대학원  
전자공학과  
박사학위 취득.  
1999년 6월~현재 : 한국 전자통신연구원 교환전송  
기술연구소 선임연구원 재직 중  
<주관심 분야> CAD 시스템, Synthesis for Low  
Power, Computer architecture 및 VLSI  
Testability 등임.

정 기 조(Ki-Jo Jung)



1999년 2월: 서강대학교

전자공학과 졸업.

1999년 2월~현재: 동대학원에서

석사과정 재학중.

<주관심 분야> CAD 시스템, Logic Synthesis for  
Low Power, Computer architecture 등임.