

# 상위 레벨 합성을 위한 저 전력 스케줄링 및 자원할당 알고리즘

신 무 경<sup>†</sup> · 인 치 호<sup>††</sup>

## 요 약

본 논문은 상위 레벨 합성 과정에서 DSP와 같은 회로를 대상으로 전력소모를 최소화 하는 스케줄링 및 자원할당 알고리즘을 제안한다. 본 논문에서는 스케줄링 시의 저 전력 설계는 리스트 스케줄링 방법을 이용한다. 그리고 자원공유를 통하여 자원할당 시 입력을 재 사용할 수 있는 가능성을 증가시킨다. 스케줄링 후 자원할당 방법은 두 입력 사이의 평균 해밍 거리와 교환동작을 계산한 결과값을 고려하여 전력 함수를 이용한다. 먼저 두 연산자 사이의 평균 해밍 거리를 계산한 후 입력 값에 대한 교환동작을 구하며, 입력 값의 비트 패턴을 이용하여 전력 값을 구한다. 자원 할당 과정은 제어 단계를 한 단계 씩 증가시키면서 각 제어 단계에서 할당 될 수 있는 모든 경우들에 대하여 평균 해밍 거리가 가장 작고 전력 함수에 의한 전력이 가장 작게 소비되는 연산자를 할당한다. 기존 방법과 비교했을 때 그 수행속도는 사용하는 연산자의 개수와 최대 제어 단계에 따라서 빨라진다. 그리고 소모하는 전력이 6% 에서 8%까지 감소효과가 있었다.

## A Low Power Resource Allocation and Scheduling Algorithm for High Level Synthesis

Mu-kyoung Shin<sup>†</sup> · Chi-ho Lin<sup>††</sup>

### ABSTRACT

This paper proposes a low power resource allocation and scheduling algorithm that minimized power consumption such as DSP circuit in high-level synthesis process. In this paper, we have used list-scheduling method for low power design in scheduling step. Also, it increase possibility to reuse input through resource sharing when assign resource. After scheduling, the resources allocation uses the power function in consideration of the result of calculating average hamming distances and switching activity between two input. First, it obtain switching activity about input value after calculate average hamming distances between two operator and find power value make use of bit pattern of the input value. Resource allocation process assign operator to minimize average hamming distance and power dissipation on all occasions which is allocated at each control step according to increase control step. As comparing the existed method, the execution time becomes fast according to number of operator and be most numerous control step. And in case of power that consume, there is decrease effect from 6% to 8% to be small.

**키워드 :** 상위레벨합성(high-level synthesis), 기능장치(functional unit), 스케줄링(scheduling), 자원할당(allocation), 저 전력(low power), 교환동작(switching activity), 해밍거리(hamming distance)

### 1. 서 론

이전의 상위 수준 합성(high level synthesis)에 관련된 대부분의 연구는 VLSI의 면적과 성능을 최적화하는 방법에 대하여 집중적으로 연구되어 왔다. 최근에는 휴

대폰, 전자 수첩, 노트북 등과 같은 휴대용 기기의 사용이 증가함에 따라 저 전력 설계의 요구가 증가하게 되었다. 따라서 저 전력 소모는 설계 시 중요한 문제가 되고 있다.

이전에는 전력 소모를 줄이기 위해서 게이트 수준이나 레지스터 전송 수준에서 연구가 많이 되었다. 하지만 반도체 집적 기술의 발달로 하나의 칩 속에 집적될 수 있는 트랜지스터 수가 증가하였기 때문에 그 칩 속에 많은 기능의

† 준 회원 : 세명대학교 대학원 전자정보학과  
 †† 정 회원 : 세명대학교 컴퓨터과학과 교수  
 논문접수 : 2001년 2월 20일, 심사완료 : 2001년 7월 10일

집적이 가능하게 되었다. 이에 최근 들어 회로를 설계하는 방식이 VHDL (VHSIC Hardware Description Language) 이나 Verilog와 같은 상위 수준 언어를 사용하여 동작적 수준(behavioral level)에서 회로를 설계하는 경우가 많아 졌다. 동작적 수준에서 회로를 설계할 경우 상위 수준 합성이 라는 과정을 거쳐야 한다. 상위 수준의 합성은 회로의 동작 적 기술(behavioral description)로부터 레지스터 전송 수준(register transfer level)으로의 합성을 의미한다[1-3]. 상위 수준 합성 과정은 크게 스케줄링 과정과 자원 할당 과정, 제어기 합성 과정으로 나눌 수 있다. 스케줄링은 연산자를 제어 구간에 할당하는 것이고, 할당과정은 연산자들과 변수 등을 기능장치(functional unit), 레지스터, 멀티플렉서나 버스와 같은 연결 요소 등에 할당하는 것이다. 스케줄링의 목 적은 사용 가능한 제한된 하드웨어를 이용하여 수행 시간 을 최소화하는 것이다. 자원 할당은 사용되는 기능장치의 개수를 줄이는 것이다. 또한 자원 할당 과정에서 소비전력 을 줄이기 위한 방법들이 제안되고 있다[4-5]. 대부분의 상 위 수준 합성과정은 NP(Non-deterministic Polynomial)-hard 문제로 알려져 있다.

저 전력 회로구현은 여러 설계 수준의 범위를 포함해야 하나 특히 회로 설계 초기 단계의 결정은 다음 단계에 큰 영향을 미치므로 상위 수준에서의 최적화는 매우 중요하다. 이전에 제안된 방법으로는 데이터 경로의 병렬화, 파이프라이 닝 또는 다중 전압(multiple voltage)을 이용하여 공급전압 (supply voltage)을 감소시키거나, 피연산자를 공유하는 시 블링 연산(sibling operation)을 같은 기능장치에 할당하는 스케줄링(scheduling)과 바인딩(binding) 방법을 제시하였 다[9, 10]. 소모되는 전력은 공급전압에 비례하므로 공급전 압을 줄이는 것은 매우 효과적이다. 하지만 전력 소모를 줄 이기 위해서 회로의 공급전압을 임의적으로 낮출 수는 없다 [11]. 전력 소모를 줄이기 위해서 공급전압을 낮추게 되면 회 로의 지연시간(delay)이 증가하게 되어 처리량(throughput) 이 감소하게 된다. 따라서 처리량을 유지하기 위해서 병렬화, 파이프라이닝 또는 다중 전압을 이용하여야 한다[6-8].

본 논문에서는 상위 수준 합성과정에서 DSP(Digital Signal Processing) 분야의 회로나 필터와 같은 회로를 대상으로 기능장치가 소모하는 전력을 최소화하는 스케줄링 및 자원 할당 알고리즘을 제안한다. DSP와 같은 분야에서는 산술 연산을 수행하는 기능장치의 수가 많고 곱셈기와 같은 기 능장치가 소모하는 전력이 크므로 회로전체의 전력 소모를 줄이기 위해 기능장치가 소모하는 전력을 우선적으로 최 소화하는 것은 큰 의미를 가진다.

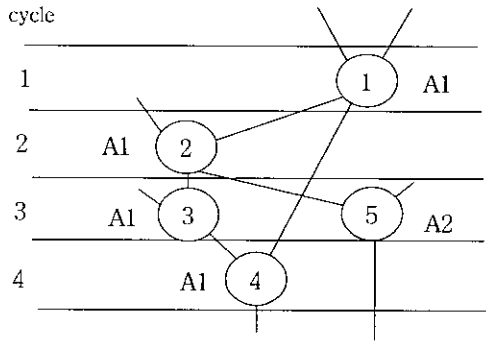
본 논문의 구성은 다음과 같다. 2장에서는 저 전력을 고 려한 스케줄링에 대해서 살펴본다. 3장은 전력 소모 모델

및 신호확률(signal probability)에 의한 교환동작(switching activity) 계산에 대해서 알아본다. 4장에서는 평균 해밍 거리 (average hamming distance)와 전력함수를 이용한 자원할 당 알고리즘을 보여준다. 5장에서는 실험 및 결과를 보이고 6장에서는 결론을 맺는다.

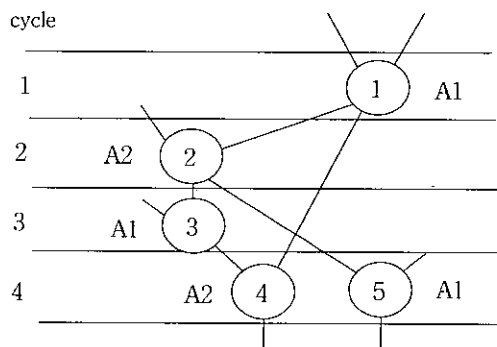
## 2. 저 전력을 고려한 스케줄링

동작적 수준에서 설계된 회로는 컴파일 과정을 거쳐 중 간형태(intermediate format)의 데이터로 저장이 된다. 중 간 형태의 데이터는 일반적으로 CDFG(Control-Data Flow Graph) 형태로 표현된다. CDFG는 회로내의 연산들의 연 결 관계를 표현한 DFG(Data Flow Graph), 그리고 이와 제어신호를 주고받는 CFG(Control Flow Graph)로 구성되 어 있다. 컴파일 과정에서 CDFG라는 중간 형태로 변환된 회로는 설계자가 설정한 속도나 면적, 전력과 같은 제한 조건을 만족시키도록 연산들을 특정한 제어 단계에 할당 하는 스케줄링 과정이 필요하다. 하나의 연산을 특정 제어 단계에 할당함으로써 하나의 제어 단계에서 동작하는 연 산수와 그것을 수행하는 기능장치의 수가 결정된다. 스케 줄링 알고리즘은 회로에 주어지는 제한 조건에 따라서 크 게 기본 스케줄링 알고리즘과 DSP와 같이 처리량이 일정 한 회로를 목적으로 하는 시간 제한 스케줄링 알고리즘과 자원 제한 스케줄링 알고리즘 등이 있다. 본 논문에서는 회로내의 연산들의 연결관계를 표현한 DFG형태를 가지고 자원 제한 스케줄링 알고리즘인 리스트 스케줄링을 이용 한다.

저 전력을 위한 스케줄링 목적은 기능장치에 대한 입력 변수의 재사용 가능성을 최대로 하는 것이다. 이것은 두 개 의 연산자를 같은 기능장치에 연속적으로 수행시킴으로써 기능장치의 한 개의 입력을 고정시켜 스위칭 변화가 일어 나지 않도록 스케줄링 한다. (그림 1)은 두 가지의 스케줄 링 방법을 보여준다. 먼저 아래 (그림 1a)는 저 전력을 고 려하지 않은 리스트 스케줄링을 보여주고, (그림 1b)는 저 전력을 고려한 리스트 스케줄링이다. 아래 (그림 1)에서 1 의 연산의 결과가 2와 4의 입력으로 들어가고 2의 연산의 결과가 3과 5의 입력으로 들어간다. (그림 1b)에서 5를 제 어구간 4에 스케줄링 함으로 연산자 3과 5를 같은 기능장 치로 공유할 수 있다. 그러므로 2개의 adder인 A1과 A2의 한쪽의 입력을 고정시켜 스위칭 동작이 일어나지 않도록 하여 저 전력을 고려 할 수 있다. 아래 (그림 1)은 두 개의 기능장치에 연산자를 할당 할 수 있는 자원제약 조건 하에 스케줄링이 수행되었다.



(a) 저 전력을 고려하지 않은 리스트 스케줄링



(b) 저 전력을 고려한 리스트 스케줄링

(그림 1) 리스트 스케줄링

본 논문에서는 먼저 리스트 스케줄링을 수행한 DFG를 이용하여 연산자 공유 가능한 그룹을 형성한다. 그리고 그 그룹 안의 모든 연산자는 같은 기능장치에 실행 될 수 있다. 저 전력을 고려한 리스트 스케줄링은 공유 그룹의 연산자만큼 스케줄링을 수행한다. 연산자의 하나의 출력에 대해서 입력으로 받는 연산자는 다른 기능장치에 할당 가능하도록 스케줄링하고, 각각 다른 제어구간에 스케줄링 한다.

### 3. 전력 소모 모델 및 신호확률에 의한 교환동작 계산

일반적으로 CMOS 회로에서의 전력 소모는 동적 전력 소모(dynamic power dissipation), 단락 전류 전력 소모(short-circuit power dissipation), 그리고 누설 전류(leakage power dissipation)에 의한 전력 소모로 구성된다. 전형적으로 단락 전류 전력 소모에 의한 전력 소모는 전체 회로에서의 전력 소모 중 약 10% 이하를, 그리고 누설 전류에 의한 전력 소모는 약 5% 이하를 차지하기 때문에, 보통 상위 수준 합성에서는 동적 전력 소모만을 고려한다. CMOS 회로에서의 평균전력은 다음과 같이 표현될 수 있다[12].

$$P = \frac{1}{2} C_L V_{DD}^2 f_{CLK} P_{switching} \quad (1)$$

여기서  $C_L$  은 유효 정전용량,  $V_{DD}^2$  공급전압 그리고  $f_{CLK}$  는 클럭 주파수이고,  $P_{switching}$  는 교환동작의 평균값을 나타낸다. 여기서 교환동작의 평균값은  $1/f_{CLK}$  클럭 사이클마다 출력변화의 평균수이다[13]. 회로 안의 노드에 대해서  $C_L, V_{DD}^2$ , 그리고  $f_{CLK}$  는 일반적으로 주어진다. 그러나  $P_{switching}$  는 입력패턴과 회로구조 안에서 결정해야한다. 위의 식에서 알 수 있듯이, 유효정전용량과 공급전압, 그리고 클럭 주파수가 주어진다면 교환동작의 평균값을 최소화시키면 CMOS 회로에서의 평균전력은 최소화되어 진다.

교환동작은 단위 클럭 내에서 신호의 스위칭이 발생할 확률이다. 그리고 그것은 입력 값의 신호 확률에 의해 계산되어진다[14]. 예를 들어 어떤 이진 순차입력을  $x$ 라고 할 때 순차입력의 길이를  $length(L)$ 이라 하고, 순차입력 중 '1'인 비트를  $S$ 라 하면 신호 확률은  $S/L$ 이 된다. 그리고  $P(x \rightarrow 1) = S/L$ 로 표시한다. 또한 순차입력 중 '0'인 비트를  $G$ 라 할 때  $G = length - S$ 가 되고, 신호 확률은  $G/L$ 이 되며,  $P(x \rightarrow 0) = G/L$ 로 표시된다. 여기서  $P(x \rightarrow 1) + P(x \rightarrow 0) = 1$ 이 된다.

하나의 신호가 클럭에 동조하여 입력된다고 할 때, 입력 신호의 변화상태를 <표 1>과 같이 4가지 상태로 표현할 수 있다. <표 1>에서 A, B는 신호가 '0'에서 '1'로 '1'에서 '0'으로 변화된 상태를 나타낸다.

<표 1> 네 개의 동작 신호

x	$P(x \rightarrow x')$	동 작
0	$0 \rightarrow 0$	'0' 상태유지
A	$0 \rightarrow 1$	$0 \rightarrow 1$ 변화
B	$1 \rightarrow 0$	$1 \rightarrow 0$ 변화
1	$1 \rightarrow 1$	'1' 상태유지

이전 입력 신호 값과 현재 입력 신호 값에 존재확률을  $P(x^0), P(x^A), P(x^B), P(x^1)$ 로 나타낸다. 동작 확률의 관계에 따라 신호 확률을 나타내면 아래 식과 같다.

$$P(x \rightarrow 0) = P(x^0) + P(x^B) \quad (2)$$

$$P(x \rightarrow 1) = P(x^1) + P(x^A) \quad (3)$$

$$P(x^0) + P(x^A) + P(x^B) + P(x^1) = 1 \quad (4)$$

$$P(x^A) = P(x^B) \quad (5)$$

위 식 (5)에서 올라가는 것(rising)과 내려가는 것(falling)의 변화는 같다. 따라서  $P_{switching}$  는 다음과 같이 나타낼 수 있다.

$$P_{switching} = P(x^A) + P(x^B) = 2P(x^A) = 2P(x^B) \quad (6)$$

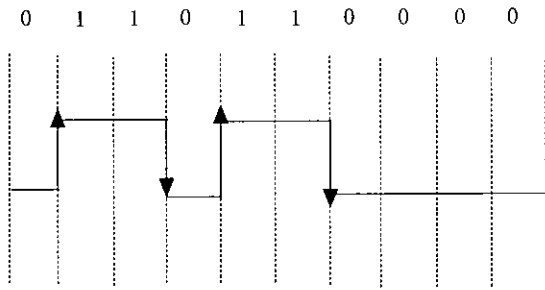
두 개의 클럭 사이클 안에 신호 값이 서로 독립되었다고 가정하면

$$P_{switching} = P(x^A) + P(x^B) = 2P(x \rightarrow 1) * P(x \rightarrow 0) \quad (7)$$

이와 같이 나타낼 수 있다. 그리고 순차입력 비트의 길이를 L이라 하고 순차입력 중 '1'인 비트를 S라 하면 S에 대해 교환동작을 다시 나타낼 수 있다.

$$P_{switching} = 2 \left( \frac{S}{L} \right) \cdot \left( \frac{L-S}{L} \right) \quad (8)$$

예를 들어 10비트의 입력이 순차적으로 입력된다고 간주하며 4개의 '1'과 6개의 '0'을 가진다고 한다면 신호 확률은  $P(x \rightarrow 0) = 0.6$ 이고  $P(x \rightarrow 1) = 0.4$ 가 될 것이다.



(그림 2) 순차입력 신호 (L=10, S=4)

위 (그림 2)는 1개의 순차입력 예를 보여준다. 여기에서 첫 번째와 마지막 비트는 같은 값을 가지며, 올라가는 것과 내려가는 것의 변화 수는 같다. 식 (5)에 의해  $P(x^A) = P(x^B) = 0.2$ 가 되고, 식 (8)에 의해  $P_{switching} = 0.48$ 의 값을 가지게 된다.

#### 4. 평균 해밍 거리와 전력함수를 이용한 자원할당 알고리즘

본 논문에서 제안하는 자원할당 알고리즘은 각 제어 단계마다 할당할 수 있는 모든 경우에 대해서 순차입력신호의 길이와 '1'인 비트수를 구한 후 입력신호의 교환동작을 구한다. 이렇게 구해진 교환동작 값을 이용하여 식 (1)에 의해 전력을 측정하게 된다. 기능장치의 전력 소모는 동작 기술에서의 변수사이의 평균 해밍 거리를 고려하여 줄일 수 있다. 기능장치의 하나의 입력 단에 대하여 연속적인 입력의 변화가 적을수록 전력 소모량은 감소한다. 따라서 연속적인 두 입력간의 평균 해밍 거리를 구하여 식 (1)에 의

한 전력 값과 평균 해밍 거리가 최소인 연산자를 각 제어 단계에서 할당하므로 기능장치가 소모하는 전력을 줄일 수 있다. 또한 입력사이에 발생하는 상관 관계까지 고려하므로 좀 더 정확한 전력 정보를 얻을 수 있다.

해밍거리는 두 변수의 입력 값인 2진수를 XOR를 취해서 구해진다. 즉, 해밍거리는 두 개의 2진수 사이의 비트의 차이를 말한다. 만약 연속적인 두 입력간의 평균 해밍 거리가 0이 되면 전력소모도 0이 된다. 연속적인 입력 변수사이의 평균 해밍 거리는 다음과 같은 식으로 구해질 수 있다[9].

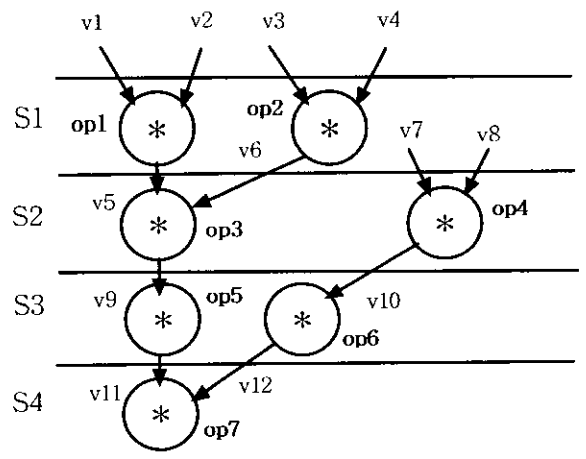
$$H(x) = ADH(x) = \lim_{n \rightarrow \infty} \frac{\sum_{i=1}^n H(x_i, x_{i-1})}{n} \quad (9)$$

여기서  $H(x_i, x_{i-1})$ 는  $x_i$ 와  $x_{i-1}$ 사이의 해밍 거리이며  $x_i$ 는 제어구간 i에서의 입력 x의 값이다. 입력 y에 대한 평균 해밍 거리  $H(y)$ 도 비슷하게 표현된다. 위 식 (9)에 대한 해밍거리가 최소가 되는 값이 전력 소모가 가장 작다는 것을 알 수 있다.

(그림 3)은 회로 중 곱셈기의 DFG만을 나타낸 것으로 4단계로 이미 저 전력을 고려한 스케줄링이 되어 있다고 가정한다.

(그림 3)에서 주어진 DFG는 자원 할당과정을 거쳐 DFG 내의 연산들을 (그림 4)에 있는 2개의 곱셈기에 할당할 것이다. (그림 3)에서 제어 단계 1과 제어 단계 2사이에서 할당 가능한 경우의 수는 다음 두 가지 경우가 있다.

먼저 첫 번째 경우에서와 같이 fu1과 fu2에 공유 가능한 연산자에 대하여 입력되는 순차입력들에 대해서 입력비트의 길이와 '1'인 비트의 수를 구하여 식 (8)에 의해 교환동작을 계산하게 된다. 그런 다음 식 (1)에 의해 전력 값을 구

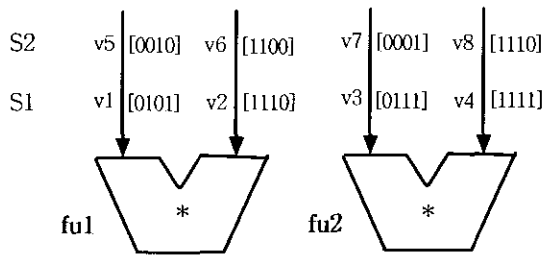


(그림 3) 스케줄링 된 DFG

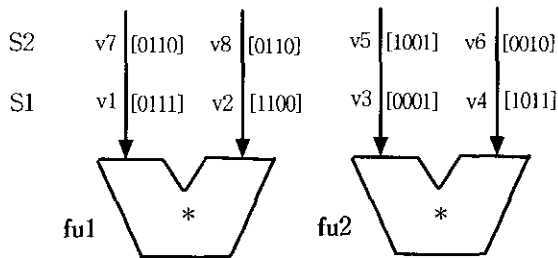
하고 식 (9)에 의해 이전 제어단계의 입력 값과 현재 제어 단계의 입력값 사이의 해밍 거리를 계산한다. 이렇게 구해진 전력 값과 해밍거리를 합하여 저장하게 된다. 해밍거리를 더하는 이유는 입력사이에 발생하는 상관 관계까지 고려하므로 좀 더 정확한 전력 정보를 얻을 수 있기 때문이다. 그런 다음 할당 가능한 두 번째의 경우에서도 위와 같은 방법으로 전력 값과 해밍거리를 구하게 된다. 이렇게 구해진 누적 값을 첫 번째 경우와 두 번째 경우를 비교하여 전력소모가 가장 적은 경우를 선택하여 연산자를 할당한다. 만약 연산자가 할당되지 않은 제어 단계가 존재한다면 이러한 제어 단계에서는 실제적으로 연산자가 전력을 소모하지 않기 때문에 교환동작을 구할 때는 신호의 개수로 (L-1)를 사용한다. 그러나 만약 신호의 길이가 무한히 클 경우에는 L로 간주한다.

첫 번째 경우 : fu1(op 1, op 3), fu2 (op 2, op 4)

두 번째 경우 : fu1(op 1, op 4), fu2 (op 2, op 3)



첫 번째 경우 1



두 번째 경우 2

(그림 4) 2가지 경우 모두 fu1과 fu2에 입력되는 비트

마지막 제어 단계에서 연산자를 할당할 때는 다른 제어 단계와는 다른 방법으로 교환동작을 구해야 한다. 마지막 제어 단계와 첫 번째 제어 단계사이의 스위칭도 고려하여 교환동작을 구해야 한다. 예를 들어 위 (그림 4)에서 제어 단계 4와 제어단계 1사이의 스위칭도 고려하여야 한다. 즉, 연속적인 입력 값의 순서가 제어단계 4와 다음 제어 단계인 1의 입력순서로 기능장치의 입력 단으로 들어오게 된다.

위와 같은 방법에 의해 제어단계 1과 제어단계 2사이의 할당 가능한 연산자들 중 연속적인 입력변수에 대한 교환 동작과 전력 소모가 최소가 되는 연산자를 기능장치에 할당하게 된다. 그런 다음 제어단계를 한 단계증가 시켜 위에서 할당된 연산자와 증가된 제어단계에서 할당 가능한 연산자 사이의 할당 가능한 경우의 수를 다시 구하게 된다. 이렇게 제어단계를 계속 증가시키며 할당 가능한 모든 경우에 대해 교환동작과 전력소모가 최소가 되는 자원을 기능장치에 할당하게 된다. 아래 <표 2>는 위의 (그림 4)의 두 경우의 공유 가능한 연산자와 할당된 연산자를 표로 나타내었다.

본 논문에서 제안하는 자원할당 방법에 대한 수행시간은 스케줄링 된 DFG의 제어단계 수와 입력신호의 개수, 그리고 기능장치의 개수의 곱이 된다.

본 논문에서는 저 전력을 고려한 스케줄링 과정을 거친 후의 DFG를 가지고 자원할당을 함으로 기능장치의 개수는 고정되어 있기 때문에, 수행시간 계산에서 입력 신호의 개수와 제어단계 수에 대해서만 고려한다. 그러므로 기존의 자원할당 방법보다 수행 시간이 빨라질 수 있다. 예를 들어 회로의 내부에 2개의 곱셈기를 가지고 있다고 가정하면 각 제어 단계에서 할당 가능한 경우의 수는 4가 된다. 만약 모든 제어 단계에서 곱셈기를 사용한다고 가정할 때, 전역 기술(global technique)방법을 사용하면 최대 제어 단계가 9라면  $4^9 = 65536$  가지의 경우의 수가 발생한다. 그러나 본 논문에서 제안하는 방법을 이용할 경우 최대  $4 * 8 = 32$  가지의 경우에 대해서만 고려를 하면 되므로 기존의 방법과 비교할 때 빠른 수행 시간을 보인다.

<표 2> 기능장치에 대한 연산자 할당

	기능장치	공유 가능한 연산자	입력변수	할당된 연산자
경우 1	Fu1	op1, op2	v1-v5, v2-v6	경우1의 선택 (op1, op3) (op2, op4)
	Fu2	op2, op4	v3-v7, v4-v8	
경우 2	Fu1	op1, op4	v1-v7, v2-v8	경우1의 선택 (op1, op3) (op2, op4)
	Fu2	op2, op3	v3-v5, v4-v6	

본 논문에서 제안하는 알고리즘은 다음과 같다.

```

consider scheduled DFG is given
loop control step i
    allocation possible case construction;
    loop construction j
        Sum_power = 0;
    
```

```

loop Fu k
    Fu_power = 0;
    L = length;
    S = bit logic 1;
    P_switching = Calculate using
                    equation(8);
    Fu_power = Calculate equation (1);
    ADH = Calculate using equation (9);
    Fu_ADH = Fu_power + ADH;
    Sum_power = Sum_power + Fu_ADH;
until empty
if ( construction j == minimum power)
storage construction j and Sum_power;
increment j;
until empty
increment i;
until empty
    
```

(그림 5) 제안하는 알고리즘

### 5. 실험 및 결과

본 논문에서 제안하는 자원할당 알고리즘의 수행시간은 제어 단계 수에 비례하고 하나의 제어 단계에서 할당 가능한 경우의 수에 비례한다. 본 논문에서는 저 전력을 고려한 스케줄링 과정을 거친 후의 DFG를 가지고 자원할당을 함으로 연산자의 개수는 고정되어 있기 때문에, 수행시간 계산에서 입력신호의 개수와 제어단계 수에 대해서만 고려한다. 그러므로 본 논문에서 제안하는 알고리즘은 그래프를 이용한 알고리즘과 비교할 때 시간 복잡도가 높지 않다. 또한 입력에 대한 상관 관계까지 고려하여 좀 더 정확한 전력측정을 할 수 있다.

본 논문이 제안하는 자원할당 방법에 대한 실험은 Ultra-sparc 5에서 수행하였다. 그리고 성능평가를 위해 volterra 필터와 wavelet 필터, 그리고 하나의 예로 테스트회로에 대해 덧셈기와 곱셈기의 개수에 제한을 두어 실험을 하였다. 스케줄링 문제는 NP-hard 문제이기 때문에 실험적인 방법 (heuristic method)이 많이 연구되어 왔다. 본 논문에서는 여러 가지 실험적인 방법 중에서 계산의 복잡도가 낮고, 구현하기 쉬운 리스트 스케줄링 알고리즘을 이용하고 있다. 그래서 회로에 대한 스케줄링은 저 전력을 고려한 리스트 스케줄링 방법을 사용하였다.

본 논문에서는 무작위 초기 자원 지정(random initial assignment)을 한 후 반복적으로 개선(iterative improvement)시키는 방법을 사용하는 HYPER[15] 합성 시스템의 자원 할당과정과 비교 실험하였다. HYPER 합성시스템의 스케

줄링 과정은 자원 이용률(resource utilization)을 극대화 하는 개선된 리스트 스케줄링 알고리즘을 사용한다. 그리고 자원 지정 과정과 스케줄링 과정을 여러 번 반복하여 자원 이용률을 높인다. 그러나 본 논문에서는 기능장치에 대한 입력 변수의 재사용 가능성을 최대로 하고, 두 개의 연산자를 같은 기능장치에 연속적으로 수행시킴으로써 기능장치의 한 개의 입력을 고정시켜 스위칭 변화가 일어나지 않도록 스케줄링 하여 전력 소모를 감소시킨다. 또한 자원 할당과정은 제어단계를 한 단계 씩 증가시키면서 한번만 수행하므로 수행속도가 빨라진다. 기존의 자원할당 과정[15]에서는 스케줄링처럼 연산의 수행 순서를 변경하여 전력 소모를 줄이고자 하였으나, 이러한 방법으로는 최적의 전력 소모 감소를 줄일 수 없다. 따라서 본 논문에서는 각각의 기능장치에 입력되는 신호의 특성을 이용하여 데이터 상관관계를 고려하여 기능장치가 소모하는 전력을 최소화한다.

본 논문이 제안하는 자원할당 방법에서 한번에 할당하는 제어단계의 개수를 한 개로 하여 전력 소모측정을 한 결과 volterra필터와 wavelet필터에서는 각각 6.78%와 8.25%의 전력 감소율을 보이며, IIR필터는 6.21%의 감소율을 보인다. 그리고 실험에 사용한 테스트회로에서는 8.5%의 전력 감소율을 보이고 있다.

<표 3> 전력 소모 비교

(단위 : mW)

	기존의 자원 할당방법	제안한 자원 할당방법	전력감소 (%)
Volterra	11.211	10.45	6.78
Wavelet	11.976	10.987	8.25
IIR	11.050	10.980	6.21
Example	5.98	5.128	8.5

아래 <표 4>는 DFG내의 곱셈기에 대해 자원할당을 수행하는데 걸리는 시간을 나타낸다. DFG내의 곱셈기에 대해 수행시간을 비교해 보면 volterra필터와 wavelet필터에서 많은 속도 향상이 있다. 본 논문에서 제안한 자원할당 방법은 각 제어 단계에서 할당 가능한 경우의 수만 고려하여 수행하게 된다. 그러나 기존의 자원할당 방법, 특히 HYPER 합성시스템은 전역 기술방법을 사용하므로 최대 제어단계 모두에 대한 할당 가능한 경우의 수를 고려하므로 많은 경우의 수가 생기게 되어 수행속도가 본 논문에서 제안한 방법보다 길어지게 된다. 따라서 본 논문에서 제안하는 자원할당 방법을 이용할 경우 연산자의 수가 많아지거나 제어단계가 길어질수록 전역 기술방법을 사용할 경우와 비교할 때 많은 수행 시간의 차이를 보인다.

〈표 4〉 곱셈기 DFG에 대한 수행 시간 비교

(단위 : sec)

	기존의 자원 할당방법	제안한 자원 할당방법
Volterra	180606.96	15.32
Wavelet	93843.75	14.48
ILR	87394.64	17.23
Example	61.71	5.657

### 6. 결 론

본 논문에서는 최근에 많이 연구되고 있는 상위 수준 합성 과정 중, 자원할당과정에서 기능장치의 전력 소모를 최소화 할 수 있는 할당 방법을 제안하였다. 스케줄링 과정에서는 자원 제약 하에 가능한 한쪽의 입력을 고정시켜 자원 할당 시 전력 소모를 줄일 수 있도록 스위칭 동작을 최소화하는 스케줄링을 수행하였다.

본 논문에서 제안하는 자원할당 방법은 제어 단계를 한 단계씩 증가시키면서 입력 값의 길이와 비트가 '1'인 값을 구하여 교환동작을 구한 후 전력 값을 계산하게 된다. 그리고 회로전체의 전력소모를 줄이기 위해 입력사이의 해밍 거리를 계산하여 위에서 구해진 전력 값과의 합을 이용하여 전력 소모를 최소화할 수 있는 경우를 선택하여 각각의 제어구간에 자원을 할당하게 된다. 입력 사이의 해밍 거리를 이용하게 됨으로서 좀 더 정확한 전력 정보를 얻을 수 있다.

본 논문에서 제안하는 방법을 이용하여 자원할당을 할 경우 기존 방법과 비교했을 때 그 수행속도는 사용하는 연산자의 개수와 최다 제어 단계에 따라서 빨라 질 수 있다. 그리고 소모하는 전력의 경우, 작게는 6%에서 8%까지 감소효과가 있다.

본 논문은 상위 레벨 합성과정 중 자원할당 과정에서 소모하는 전력을 최소화하는 것이 목적이다. 하지만 상위 레벨 합성과정에는 레지스터할당, 버스, 그리고 MUX할당 등의 과정이 있다. 앞으로 이러한 과정에서도 본 논문의 알고리즘이 적용되어 전력소모를 줄일 수 있는 연구를 할 필요가 있다.

### 참 고 문 헌

[1] M. C. McFarland, A. C. Parker, R. Camposano, "The High Level Synthesis of Digital Systems," Proceedings of the IEEE, Vol.78, No.2, Feb, 1990.  
 [2] D. Gajski and N. Dutt, "High-level Synthesis : Introduction to Chip and System Design," Kluwer Academic Pub-

fishers, 1992.  
 [3] G. D. Micheli, "Synthesis and Optimization of Digital Circuits," New York : Mc-Graw Hill, Inc., 1994.  
 [4] Jui-Ming Chang, Massoud Pedram, "Low Power Register Allocation and Binding Power," Proc. of 32nd Design Automatic Conference, pp.29-35, 1995.  
 [5] Raghunathan, A., and Jha, N. K., "An ILP Formulation for Low Power Based on Minimizing Switched Capacitance during Data-path Allocation," Proc. of the IEEE International Symposium on Circuits and Systems, pp. 1069-1073, 1995.  
 [6] A. P. Chandrakasan, M. Potkonjak, R. Mehra, J. Rabaey, and R. W. Brodersen, "Optimizing power using transformation," IEEE Tr. on CAD/ICAS, pp.12-31, Jan. 1995.  
 [7] W. T. Shiue, C. Chakrabarti, "Low-Power Scheduling with Resources Operating at Multiple Voltages," IEEE Transactions on Circuits and Systems, Vol.47, No.6, pp.536-543, 2000.  
 [8] Y.-R. Lin, C.-T. Hwang, and A. C.-h. Wu, "Scheduling techniques for variable voltage Low Power Design," ACM Trans. Design Automat. Electro. Syst., Vol.2, No.2, pp. 81-97, April 1997.  
 [9] E. Musoll and J. Cortadella, "Scheduling and Resource Binding for Low Power," in Proceedings of International Symposium on System Synthesis, pp.104-109, Apr. 1995.  
 [10] Y. Fang and A. Albicki, "Joint Scheduling and allocation for Low Power," Int'l Symposium on Circuits & Systems, pp. 556-559, May. 1996.  
 [11] Massoud Pedram, "Power Minimization in IC design : principles and application," ACM Transactions on Design Automation of Electronic Systems, pp.3-56, Vol.1, No.1, Jan. 1996.  
 [12] A. Raghunathan, Niraj K. Jha, S. Dey, "High-Level Power Analysis and Optimization," Kluwer Academic Publishers, pp.17-25, 1998.  
 [13] M. Pedram, "Power Minimization in IC Design : Principles and applications," Transactions of ACM, Vol.1, No.1, pp. 1-58, March, 1996.  
 [14] F. N. Najm, "A Survey of Power Estimation Techniques in VLSI Circuits," IEEE Transactions on VLSI Systems, Vol.2, No.4, pp.446-455, 1995.  
 [15] A. P. Chandrakasan, M. Potkonjak, R. Mehra, J. Rabaey, and R. W. Brodersen, "Optimizing power using transformations," IEEE Trans Computer-Aided Design, Vol. 14, pp.12-51, Jan, 1995.



### 신 무 경

e-mail : smk0909@hanmail.net

2000년 세명대학교 컴퓨터과학과 졸업  
(이학사)

2001년~현재 세명대학교 전산정보학과  
석사과정

관심분야 : Low Power, CAD & VLSI  
design, High-level synthesis



### 인 치 호

e-mail : ich410@venus.semyung.ac.kr

1985년 한양대학교 전자공학과 공학사

1987년 한양대학교 대학원 공학석사

1996년 한양대학교 대학원 공학박사

1992년~현재 세명대학교 컴퓨터과학과  
부교수

관심분야 : VLSI CAD, ASIC 설계, CAD 알고리즘, RTOS 및  
내장형 시스템