저 전력 아키텍처 설계를 위한 새로운 자원할당 알고리즘

신무경, 인치호*
sh0909@venus.semyung.ac.kr , ich410@venus.semyung.ac.kr*
세명대 컴퓨터과학과*

A New Resource Allocation Algorithm for Low Power Architecture

Mu-kyoung Shin, Chi-ho Lin*
sh0909@venus.semyung.ac.kr , ich410@venus.semyung.ac.kr*
*Dept. of Computer Science, Semyung University.

Abstract

This paper proposed resource allocation algorithm for the minimum power consumption of functional unit in high level synthesis process as like DSP which is circuit to give many functional unit.

In this paper, the proposed method though high level simulation find switching activity in circuit each functional unit exchange for binary sequence length and value bit are logic one value. To used the switching activity find the allocation with minimal power consumption, the proposed method visits all control steps one by one and determines the allocation with minimal power consumption at each control step.

Ⅰ. 서론

1)이전의 상위 수준 합성에 관련된 대부분의 연구는 VLSI의 면적과 성능을 최적화하는 방법에 대하여 집중적으로 연구 되어왔다[1][2]. 최근 들어, 디지털 시스템의 성능과 복잡도가 증가함에 따라 전력이 중요한 요소가 되기 때문에, 자원 할당 과정에서도 전력을 줄이기 위한 방법들이 제안되고 있다[3][4].

본 논문에서는 DSP 분야의 회로나 필터와 같은 회로를 대상으로 연산자가 소모하는 전력을 최소화 하고자 한다. DSP와 같은 분야에서는 산술 연산을 수행하는 연산자의 수가 많고 곱셈기와 같은 연산자가 소모하는 전력이 훨씬 크다. 따라서 회로전체의 전력 소모를 줄이기 위해 연산자가 소모하는 전력을 우선적으로 최소화하는 것은 큰 의미를 가진다. 특히 DSP와 같은 분야에서는 연산자에 비해 상대적으로 전력 소모가 작은 레지스터와 연결을 모두 고려할 경우 이들의 전력 소모를 줄여서 얻는 이익보다 복잡도의 증가로 인한 손실이 크므로 연산자를 대상으로 전력 소모를 줄이는 것은 전체 회로의 전력 소모와 수행 시간을 고려했을 때 바람직한 선택이다.

Ⅱ. CMOS 회로에서의 전력소모

CMOS 회로에서의 전력 소모는 동적 전력 소모, 단락 전류 전력 소모, 그리고 누설 전류에 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 다음

1) 본 연구는 반도체 설계 교육센터(IDEC)의 지원에 의한 것입니다.

과 같이 표현될 수 있다.

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

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 사이클 마다 출력변화의 평균수이다[5]. 회로 안의 노드에 대해서 C_L, V_{DD}^2, 그리고 f_{CLK}는 대게 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.

Ⅲ. 신호 확률과 변화 확률

교환동작은 단위 클럭 내에서 신호의 스위칭이 발생할 확률이다. 그리고 그것은 입력값의 신호 확률에 의해 계산되어진다[6]. 예를 들어 어떤 이진 순차입력을 x라고 할 때 순차입력의 길이를 length(L)이라 하고, 순차입력 중 '1'인 비트를 S라 하면 신호 확률은 S/L이 된다. 그리고 P(x->1) = S/L 로 표시한다. 또한 순차입력 중 '0'인 비트를 G라 할 때 G = length - S 가 되고, 신호 확률은 G/L이 되며, P(x->0) = G/L로 표시된다. 여기서 P(x->1) + P(x->0) = 1 이 된다.

하나의 신호가 클럭에 동조하여 입력된다고 할 때, 입력신호의 변화상태를 표1 과 같이 4가지 상태로 표현할 수 있다. 표1에서 A, B는 신호가 '0'에서 '1'로 '1'에서 '0'으로 변화된 상태를 나타낸다.

표 1 네 개의 동작 신호

x	P(x -> x')	동작
0	0 -> 0	'0'상태유지
A	0 -> 1	0->1변화
B	1 -> 0	1->0변화
1	1 -> 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.

$$P(x->0) = P(x^0) + P(x^B) \quad (식\ 2)$$
$$P(x->1) = P(x^1) + P(x^A) \quad (식\ 3)$$
$$P(x^0) + P(x^A) + P(x^B) + P(x^1) = 1 \quad (식\ 4)$$
$$P(x^A) = P(x^B) \quad (식\ 5)$$

위 식5에서 올라가는 것과 내려가는 것의 변화는 같다. 따라서 $P_{switching}$ 는 다음과 같이 나타낼 수 있다.

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

두 개의 클럭 사이클 안에 신호값이 서로 독립되었다고 가정하면

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

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.

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

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

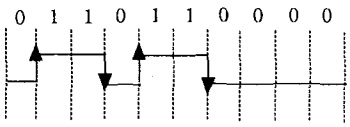


그림.1 순차입력 신호 (L=10, S=4)

위 그림1은 1개의 순차입력 예를 보여준다. 여기에서 첫 번째와 마지막 비트는 같은 값을 가지며, 올라가는 것 과 내려가는 것의 변화 수는 같다. 식 5에 의해 $P(x^A)=P(x^B)=0.2$ 가되고, 식 8에 의해 $P_{switching}$= 0.48의 값을 가지게 된다.

Ⅳ. 새로운 자원할당 알고리즘

본 논문에서 제안하는 자원할당 알고리즘은 각 제어 단계마다 할당할 수 있는 모든 경우에 대해서 순차입력

신호의 길이와 '1'인 비트수를 구한 후 입력신호의 교환동작을 구한다. 이렇게 구해진 교환동작값을 이용하여 식1에 의해 전력을 측정하게 된다. 그리고 각 제어 단계에서 전력 소모가 최소가 되는 경우를 찾아서 할당하므로 연산자 가 소모하는 전력을 줄일 수 있다.

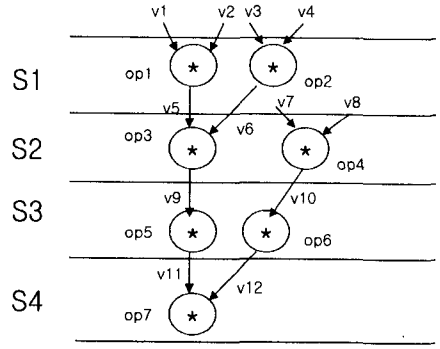
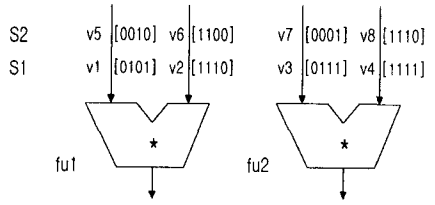


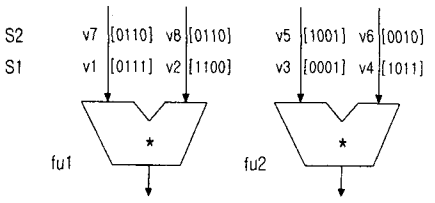
그림.2 스케줄링 된 DFG

그림2는 회로 중 곱셈기의 DFG만을 나타낸 것으로 4단계로 이미 스케줄링되어 있다고 가정한다. 그림2에서 주어진 DFG는 자원 할당과정을 거쳐 DFG내의 연산들을 그림 3에 있는 2개의 곱셈기에 할당할 것이다. 그림2에서 제어 단계 1과 제어 단계 2사이에서 할당할 수 있는 경우의 수는 다음 두 가지 경우가 있다.

첫 번째 경우 : fu1(op1, op3), fu2(op2, op4)
두 번째 경우 : fu1(op1, op4), fu2(op2,, op3)



첫 번째 경우 1



두 번째 경우 2

그림 3. 2가지 경우 모두 fu1과 fu2에 입력되는 비트

먼저 첫 번째 경우에서와 같이 fu1과 fu2에 입력되는 순차입력들에 대해서 입력비트의 길이와 '1'인 비트의 수를 구하여 식8에 의해 교환동작을 구한 후 식1에 의해 평균전력값을 구한다. 그런 다음 제어 단계를 다음단계로 증가 후 교환동작을 구하여 평균전력값을 누적하게 된다. 이렇게 구해진 평균전력값을 첫 번째 경우와 두 번째 경우를 비교하여 전력소모가 가장 적은 경우를 선택하여 연산자를 할당한다. 만약 연산이 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 길이 무한히 클 경우에는 L로 간주한다. 마지막 제어 단계에서 연산자를 할당할 때는 다른 제어 단계와는 다른 방법으로 신호동작을 구해야 한다. 마지막 제어 단계와 첫 번째 제어 단계사이의 스위칭도 고려하여 신호동작을 구해야 한다. 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 using
                    equation (1);
             Sum_power=Sum_power+
                    Fu_power;
          until empty
          if ( construction j ==
               minimum power)
          storage construction j
                  and Sum_power;
          increment j;
     until empty
     increment i;
  until empty
```

V. 실험 및 고찰

본 논문에서 제안하는 자원할당 알고리즘의 수행시간은 제어 단계수에 비례하고 하나의 제어 단계에서 할당 가능한 경우의 수에 비례한다. 본 논문에서는 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. 성능평가를 위해 volterra필터와 wavelet필터, 그리고 하나의 예로 테스트회로에 대해서 실험을 하였다. 회로에 대한 스케줄링은 ILP based 스케줄링 방법을 사용하였다.

표2. 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
Example	5.98	5.128	8.5

DFG내의 곱셈기에 대해 수행시간을 비교해 보면 Volterra필터와 Wavelet필터에서 많은 속도 향상이 있다.

표3. 곱셈기 DFG에 대한 수행 시간 비교

단위 : sec

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

VI. 결론 및 연구과제

본 논문에서 제안하는 방법은 제어 단계를 한 단계씩 증가시키면서 입력값의 길이와 비트가 '1'인 값을 구하여 교환동작을 구한 후 전력값을 계산하게 되고, 이 전력값을 이용하여 전력 소모를 최소화할 수 있는 경우를 선택하여 할당하게 된다. 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 MUX 할당 등의 과정이 있다. 앞으로 이러한 과정에서도 본 논문의 알고리즘이 적용되어 전력소모를 줄일 수 있는 연구를 할 필요가 있다.

참고 문헌

[1] D. Gajski and N. Dutt, High-level Synthesis: Introduction to Chip and System Design. Kluwer Academic Publishers, 1992.

[2] G. D. Micheli, Synthesis and Optimization of Digital Circuits. New York: Mc-Graw Hill, Inc., 1994.

[3] Jui-Ming Chang, Massound Pedram, "Register allocation and binding for low power," Proc. of 32nd Design Automatic Conference, pp29-35

[4] Raghunathan, A., and Jha, N. K., "An ILP formula tion for low power based on minimizing switched capacitance during data-path allocation, " Proc. of the IEEE International Symposium on Circuits and Systems.

[5] M. Pedram, "Power minimization in IC Desig n: Principles and applications," Transactions of ACM, vol.1,no. 1, pp.1-58, March, 1996.

[6] F. N. Najm, " A survey of power estimation techn iques in VLSI circuits," IEEE Transactions on VLSI Systems, vol.2 , no.4, pp.446-455, 1995