

論文98-35C-1-1

게이트 사이징 과 감작 경로를 이용한 클럭 주기 최적화 기법

(Clock Period Optimization by Gate Sizing and Path Sensitization)

金 周 虎 *

(Juho Kim)

要 約

출력이 래치 되고 입력 벡터가 연속적으로 입력에 들어가는 회로에서, 임계 경로의 지연을 감소시키기 위한 게이트 크기변환에 의한 방법만으로는 성능이 향상 되지 않을 수도 있다. 조합 회로에서 클럭 주기는 최장 경로의 지연 뿐만 아니라 최단 경로의 지연에 의해서도 결정 되기 때문에, 클럭 주기는 최장 경로의 지연을 감소시키거나 최단 경로의 지연을 증가 시킴으로써 최적화 될 수 있다. 회로의 원하는 클럭 주기를 얻기 위해서, 최장 감작 경로 또는 최단 위상 경로에 있는 게이트들을 사이징 할 수 있다. 그러나 최단 위상 경로에 있는 게이트의 크기를 변화할 경우 최장 감작 경로의 지연을 변화 시킬 수 있으므로 감작 경로를 고려한 게이트 선택은 어려운 문제이다. 새로운 클럭 주기 최적화를 위한 게이트의 선택 방법이 이 논문에 제안 되었다. 새로운 게이트 선택 알고리즘은 최단 위상 경로 상에 있는 게이트의 크기변환을 하는 동안 최장 감작 경로의 지연이 증가하는 것을 막고, 최장 감작 경로에 있는 게이트의 크기변환을 하는 동안 최단 위상 경로의 지연이 감소 하는 것을 막을 수 있다. 결과적으로 각각의 게이트 변환 단계는 클럭 주기가 증가하지 않도록 보장하게 된다. 제안된 알고리즘은 ISCAS85 벤치 마크 회로로 테스트 되었고, 실험 결과로부터 클럭 주기는 제안된 게이트 선택 방법에 의해 효율적으로 최적화 됨을 알 수 있다.

Abstract

In the circuit model that outputs are latched and input vectors are successively applied at inputs, the gate resizing approach to reduce the delay of the critical path may not improve the performance. Since the clock period is determined by delays of both long and short paths in combinational circuits, the performance (clock period) can be optimized by decreasing the delay of the longest path, or increasing the delay of the shortest path. In order to achieve the desired clock period of a circuit, gates lying in sensitizable long and short paths can be selected for resizing. However, the gate selection in path sensitization approach is a difficult problem due to the fact that resizing a gate in shortest path may change the longest sensitizable path and vice versa. For feasible settings of the clock period, new algorithms and corresponding gate selection methods for resizing are proposed in this paper. Our new gate selection methods prevent the delay of the longest path from increasing while resizing a gate in the shortest path and prevent the delay of the shortest path from decreasing while resizing a gate in the longest sensitizable path. As a result, each resizing step is guaranteed not to increase the clock period. Our algorithms are tested on ISCAS85 benchmark circuits and experimental results show that the clock period can be optimized efficiently with our gate selection methods.

* 正會員, 西江大學校 電子計算學科

로 수행되었음

(Dept. of Computer Science, Sogang University)

接受日: 1997年8月13日, 수정완료일: 1998年1月3日

※ 본 연구는 1997년 서강대학교 교내연구비 지원으

I. 서 론

일반적으로 조합 회로의 성능은 최장 위상 경로의 지연에 의해서 결정되는 것으로 알려져 있다. 최근에, 감작 경로의 지연을 계산하는 몇 가지 알고리즘^[1,2,3,4]이 제안 되었다. 타이밍 분석에서, 최대 지연은 주어진 시간 제한을 초과 하는지를 조사하기 위해 사용된다. 최장 경로의 지연이 주어진 시간 제한을 넘을 경우, 타이밍 최적화가 회로의 지연을 감소시키기 위해서 사용 될 수 있다. 타이밍 최적화의 한 방법이 게이트의 크기 변환이다.

출력이 래치 되고 입력 벡터가 입력에 연속적으로 가해지는 회로에서, 성능은 기본 입력 벡터에 의해서 출력 래치에 나타나는 연속적인 신호 사이의 지연 차이인 클럭 주기에 의해 나타내어진다. 입력과 출력은 같은 클럭에 의해 트리거 되면, 클럭 주기는 최장 위상 경로와 최단 위상 경로의 지연에 의해 결정된다. 논문 [6,7]에 의하면 최장 위상 경로의 지연을 τ_{\max} 로 가정하고 최단 위상 경로의 지연을 τ_{\min} 으로 가정하면 클럭 주기 C_p 의 범위는 공식 $C_p \geq \tau_{\max} - \tau_{\min} + t_c$ (t_c 는 래치의 셋업 시간과 홀드 시간의 합)로 제한된다. 최장 감작 경로를 이용한 새로운 클럭 주기 설정 방법이 [5]에 제시되었다. 만일 최장 감작 경로의 지연을 τ^*_{\max} 로 가정하면 새로운 클럭 주기 C_p^* 의 범위는 $C_p^* \geq \tau^*_{\max} - \tau_{\min} + t_c$ 가 된다. 최장 감작 경로의 지연 τ^*_{\max} 는 최장 위상 경로의 지연 τ_{\max} 보다 작기 때문에 [5]에서 얻어진 클럭 주기 C_p^* 가 C_p 보다 작게 된다.

회로의 성능을 최적화하기 위해서, 클럭 주기 C_p^* 는 게이트 크기 변환에 의해 줄어 들 수 있다. 그러나 감작 경로를 이용한 크기 변환을 하게 될 경우에 게이트를 선택하는 것은 어려운 문제이다. 최장 감작 경로의 지연 τ^*_{\max} 을 갖는 경로를 P 라고 가정하면, 클럭 주기 C_p^* 를 감소시키기 위해 τ_{\min} 을 증가 시키는 방법이 선택될 때 최단 위상 경로에는 속하지만 경로 P 에는 속하지 않는 게이트들이 더 느리게 변환될 수 있다. 한 느리게 변환된 게이트가 다른 긴 경로 Q 에 속할 때 사이징의 결과로 Q 의 지연이 경로 P 의 지연 보다 더 커질 수 있다. 사이징한 후 회로의 최장 감작 경로는 새로운 경로 Q 가 될 수 있고 결과적으로, 클

럭 주기는 전혀 줄어들지 않을 수 있다.

이 논문에는 성능 최적화를 위해 크기변환 될 게이트를 선택하는 새로운 방법이 제안되었다. 임계 경로와 최단 불안정 감작 경로(shorestest destabilizing path)를 알아내기 위한 방법이 사용되었고 사이징의 매 반복마다 클럭 주기를 증가 시키지 않도록 보장하기 위한 효율적인 게이트 선택 알고리즘이 제시되었다. 최장 감작 경로를 구하는 방법^[4]에 의해 최장 감작 경로와 최단 불안정 감작 경로를 결정하는 과정이 II장에 나타나 있다. 클럭 주기를 결정하는 방법이 III장에 논의 되었고, IV장에는 제시된 게이트 선택 방법을 포함해서 클럭 주기를 최적화하는 구체적인 알고리즘이 제안되었다. V장에서는 실험 결과를 VI장에서는 결론으로 끝을 맺는다.

II. 최장 및 최단 경로의 탐색

먼저 이 논문에 자주 나오는 기본적인 용어들을 몇 가지 서술하겠다. 조합 회로의 경로 $P = \{f_0, G_0, f_1, G_1, \dots, G_{m-1}, f_m\}$ 는 게이트와 간선들로 표현된다. 간선 f_i 는 게이트 G_{i-1} 과 게이트 G_i 를 연결하는 P 의 주입력 (on-input)이고, 간선 f_i 는 게이트 G_i 에는 연결되어 있지만 게이트 G_{i-1} 에는 연결 되어 있지 않은 경우로 부입력 (side-input)이라고 정의 된다. 제어 값 (Controlling value)은 G_i 의 다른 입력에 관계없이 간선 f_{i+1} 의 로직이 결정되는 f_i 의 로직이다. 예를 들어, AND 게이트의 경우 입력이 0일 경우 다른 입력 값에 관계없이 출력은 0이 되므로 AND 게이트의 제어 값은 0이다. 마찬가지로 하나의 입력이 1이면 다른 입력에 관계없이 출력은 1이 되므로 OR 게이트의 제어 값은 1이다. 비제어 값 (Non-controlling value)은 제어 값의 보수로서 정의 될 수 있다. 입력에서부터 출력까지 전이(transition)를 전파(propagation) 시킬 수 있는 기본 입력 벡터가 적어도 한 개 존재할 경우 그 경로를 감작 경로라고 한다. 어떠한 입력 벡터에 의해서도 감작되지 않는 경로는 차오 경로 (false path)라고 한다. 마지막으로 임계 경로는 회로에서 최장 감작 경로로 정의 된다.

1. 최장 그리고 최단 위상 경로

위상 경로는 회로의 기능을 고려하지 않고 게이트와

간선을 연결한 경로로 정의 된다. 최장 위상 경로는 최초 경로를 포함할 수 있으며 최장 위상 경로의 지연은 회로 지연의 상한 값을 가지게 된다. 마찬가지로 기능을 고려하지 않은 최단 경로의 지연은 회로 지연의 하한 값을 가지게 된다. 먼저, 테스트 될 회로는 그래프로 변형되고 회로의 그래프 표현에서 두 개의 가장 노드를 생성한다. 소스 노드 (s)는 입력에 연결되고 싱크 노드 (t)는 출력에 연결된다. (s)와 (t) 노드를 추가한 후에 그래프에서의 경로는 (s) 노드에서 (t) 노드로의 경로가 된다.

그래프에서 최장 위상 경로와 최단 위상 경로를 계산하는 과정은 (s) 노드에서 (t) 노드까지 경로를 탐색하면서 (s) 노드에서 각 노드까지의 최대 지연과 최소 지연을 기록한다. 각 노드에서 최대 지연과 최소 지연을 기록한 후, 최장 위상 경로와 최단 위상 경로를 찾기 위해 그래프를 (t)에서 (s)로 탐색 한다. 이 방법의 시간 복잡도는 $O(V + E)$ 가 된다.

2. 최장 감작 경로

최장 감작 경로인 임계 경로(critical path)를 찾기 위해서 가장 큰 지연을 갖는 경로부터 검색되어 큐 QL에 저장 된다. K 개의 진 경로를 베스트-퍼스트 검색 (Best-First Search [8])을 이용하여 찾고 큐에 저장된 각 경로의 감작여부를 검사한다. 타이밍 분석에서는 측정된 지연이 회로의 실제 지연보다 크거나 같으면 옳은 분석으로 알려져 있다. 지금까지 증명된 옳은 감작 경로 검색표준^[4] 중의 하나로 루스표준 (loose criterion)^[5] 있고 다음과 같이 정의 될 수 있다.

정의 1 (Loose Path Sensitization Criterion) 만약 경로의 주입력이 게이트 입력 중 가장 빨리 도착한 제어 입력 이거나 모든 부입력이 비제어 입력일 때 비제어 입력이 되게 하는 하나 이상의 기본 입력 벡터들이 존재할 경우의 경로를 루스 감작 경로라고 한다. 테스트 될 경로에 있는 모든 게이트를 입력에서부터 출력으로 loose path sensitization criterion을 적용하여 감작 경로의 여부를 판단한다. 최장 위상 경로는 최초 경로일 수도 있기 때문에 최장 감작 경로의 지연은 항상 최장 위상 경로의 지연보다 작거나 같고, 결론적으로 회로 지연의 상한 값이 줄어 들게 된다.

3. 최단 불안정 감작 경로

최장 감작 경로와는 대조적으로 최단 불안정 감작

경로는 출력이 안정 상태에서 불안정 상태가 되는 최단 시간으로 정의 된다. [4]에 있는 정의에 따라 불안정 감작 경로를 정의 하면 다음과 같다.

정의 2 (불안정 감작 경로) 만약 경로의 주입력이 게이트 입력 중 가장 늦게 도착한 제어 입력 이거나 모든 부입력이 비제어 입력일 때 가장 빨리 도착한 비제어 입력이 되게 하는 하나 이상의 주입력 벡터들이 존재할 경우의 경로를 불안정 감작 경로라고 한다.

최단 불안정 감작 경로를 찾기 위해서, 최단 위상 경로의 집합을 찾아서 각각 큐 QS에 저장 된다. 최단 경로의 집합은 베스트-퍼스트 검색을 사용해서 구할 수 있다. 최대를 최소로 바꾸는 BFS의 간단한 변형에 의해 K개의 경로 집합을 찾아서 QS에 저장 된다. 불안정 감작 경로를 구별하기 위해서, 최단 경로부터 시작해서 큐 QS에 있는 경로가 최장 감작 경로의 탐색과 유사한 방법에 의해 테스트 된다. 위의 알고리즘은 출력 상태가 안정에서 불안정하게 되는 최단 시간을 결정하기 때문에, 간선과 게이트 지연의 최소 값이 사용되었다. 최단 경로의 지연은 회로의 지연보다 큰 값이 된다. 그러나, 최단 불안정 감작 경로의 지연은 항상 최단 위상 경로의 지연보다 항상 같거나 크다.

III. 클럭 주기 설정

조합 회로의 클럭 주기는 $C_p \geq \tau_{\max} - \tau_{\min} + t_c$ (where $t_c = t_s + t_h$) 셋업 타임 t_s , 홀드 타임 t_h , 최대 지연 τ_{\max} , 그리고 최소 지연 τ_{\min} 에 의해서 결정 된다. 최장 그리고 최단 위상 경로가 τ_{\max} 와 τ_{\min} 을 계산하기 위해 사용될 때, 클럭 주기는 $C_p \geq D(L) - D(S) + t_c$ 로 하한 값이 결정되며, 여기서 L은 최장 위상 경로를 S는 최단 위상 경로를 나타낸다. II장에 있는 *Loose Path Sensitization Criterion*을 사용하면, 회로의 최대 전파 지연인 τ_{\max} 는 $D(L^*)$ 라 할 수 있으며, 여기서 L^* 은 최장 감작 경로를 나타낸다. 마찬가지로 τ_{\min} 은 $D(S^*)$ 라 할 수 있으며, S^* 은 최단 불안정 감작 경로를 나타낸다. 회로의 클럭 주기는 세 개의 식으로 나타낼 수 있으며 [5]에 이 식들이 증명되어 있다.

$$C_p(LS) \geq D(L) - D(S) + t_c$$

$$C_p(L^*S) \geq D(L^*) - D(S) + t_c$$

$$C_p(LS^*) \geq D(L) - D(S^*) + t_c$$

최대 및 최단 위상 경로의 지연에 의해 결정되는 클럭 주기 C_p 는 $C_p(LS)$ 로 표기된다. 유사하게 최장 감작 경로의 지연과 최단 위상 경로의 지연에 의한 클럭 주기는 $C_p(L^*S)$ 로 나타내어 진다. 정의에 의하여 $D(L) \geq D(L^*)$ 이고 $D(S) \leq D(S^*)$ 이므로 $C_p(LS)$ 가 항상 최대값을 가지게 된다. $C_p(L^*S^*)$ 에 의한 클럭 주기가 분명 최소값을 나타내지만, 클럭 주기 $C_p(L^*S^*)$ 는 잘못된 설정임이 [5]에 증명되어 있다.

최대 위상 경로의 지연과 최장 감작 경로의 지연의 차이가 최단 불안정 감작 경로의 지연과 최단 위상 경로의 차이 보다 크다면 $(D(L) - D(L^*)) > D(S^*) - D(S)$, $C_p(L^*S)$ 에 의한 클럭 주기가 $C_p(LS^*)$ 에 의한 클럭 주기보다 짧게 된다. $D(L) = D(L^*)$ 일 경우는 $C_p(LS^*)$ 이 더 유용하다. 그러나 $D(L) - D(L^*) = D(S^*) - D(S)$ 일 경우는 $C_p(L^*S)$ 와 $C_p(LS^*)$ 의 클럭 주기가 서로 같게 된다.

IV. 케이트 선택 문제

조합 회로로부터 얻어진 클럭 주기는 요구되는 성능을 만족하지 못 할 경우도 있다. 회로의 성능 향상을 위해 클럭 주기를 최적화 하고자 하는 경우에는 버퍼 삽입이나 케이트 사이징 기법이 사용 될 수 있다. 이 장에서는 원하는 클럭 주기를 얻기 위한 새로운 케이트 사이징 방법이 제시 되었다.

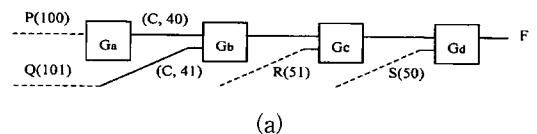
Full-custom layout에서 케이트는 어떠한 크기로 도 사이징 될 수 있다. 그러나 회로가 standard cell의 라이브러리에 의해 설계된다면, 케이트는 일정한 단위에 의해서만 사이징 될 수 있다. 이 논문에서는 단순화를 위해 일정한 단위에 의한 케이트 사이징이 가정 되었고, 케이트의 지연은 면적(크기)에 반비례 한다고 가정하였다. 클럭 주기는 $C_p = \tau_{\max} - \tau_{\min} + t_c$ 의 식으로 표현 되므로 케이트의 크기 변화를 통해 τ_{\max} 를 감소 시키거나 τ_{\min} 을 증가 시킴으로써 요구되는 클럭 주기가 얻어 질 수 있다. 사이징을 위한 새로운 케이트 선택 방법이 클럭 주기의 세 가지 설정에 따라 이 장에 나타낼 것이다. 클럭 주기 $C_p(LS)$ 는 최장 위상 경로와 최단 위상 경로에 의해 결정되므로

케이트 선택은 비교적 단순하다. 그러므로 최장 감작 경로 탐색을 사용한 클럭 주기 $C_p(L^*S)$ 와 최단 불안정 감작 경로를 이용한 클럭 주기 $C_p(L^*S)$ 를 최적화하기 위한 새로운 케이트 선택 방법을 기술 하겠다. 이 논문 전체를 통해서 얻고자 하는 목표 클럭 주기는 C_p^* 로 표시 하며, 최장 위상 경로에 있는 케이트의 집합을 $G(L)$ 로 최단 위상 경로에 있는 케이트의 집합을 $G(S)$ 로 표시하였다. 마찬가지로 최장 감작 경로에 있는 케이트의 집합은 $G(L^*)$ 로 최단 불안정 감작 경로에 있는 케이트의 집합은 $G(S^*)$ 로 표시하였다.

1. $C_p(L^*S)$ 을 최적화하기 위한 케이트 선택

공식 $C_p(L^*S) = D(L^*) - D(L) + t_c$ 에 의해서 정해진 클럭 주기는 $D(L^*) \leq D(L)$ 이므로 $C_p(LS)$ 보다 작거나 같다. $C_p(L^*S)$ 을 최적화하기 위해서 $D(L^*)$ 이 감소하는 방법이 적용 될 때 최장 감작 경로에 있는 케이트들이 $D(S)$ 를 줄이지 않고 사이징 될 수 있다. 그러므로 변환 될 케이트들은 $G(L^*\bar{S})$ 에 속해야 하며 여기서, $G(L^*\bar{S})$ 는 $G(L^*\bar{S}) = G(L^*) - (G(L^*) \cap G(S))$ 가 된다.

그림 1(a)에서, 최장 감작 경로는 loose path sensitization criterion에 의해서 $D(L^*) = 100$ 을 가지는 $L^* = P$ 로 가정 한다. 최단 위상 경로가 50의 지연을 갖는 S 라고 가정하면 예제 회로의 클럭 주기는 $C_p(L^*S) = 100 - 50 + 0 = 50(t_c = 0)$ 이 된다. 요구되는 클럭 주기가 $C_p^* = 40$ 일 때 최장 감작 경로의 지연은 $G(L^*\bar{S})$ 에 있는 케이트들을 사이징 함으로서 줄여들 수 있다. 케이트 G_a 와 G_c 는 최단 경로에 있기 때문에 그림 1(a)에 있는 예에서 좋은 결과를 얻을 수 없다. 따라서 케이트 G_a 또는 G_b 를 사이징 해야 한다. G_b 의 주입력이 제어 값일 경우, 경로 P 가 감작 되기 위해서는 주입력 신호의 도착 시간이 제일 빨라야 한다. 그러므로 케이트 G_a 또는 G_b 의 지연이 줄어 든다 해도 최장 감작 경로를 변화 시키지는 않는다.



(a)

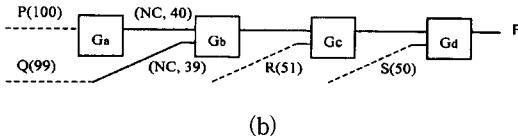


그림 1. ($\Delta=10$)만큼 $D(L^*)$ 를 감소 시킴으로써 $C_p(L^*S)$ 을 최적화 하는 예

(a) $L^* = P$ 이고 G_b 의 주입력이 제어 값 일 때: 게이트 선택은 G_a 또는 G_b (b) $L^* = P$ 이고 G_b 의 주입력이 비제어 값 일 때: 게이트 선택은 G_b

Fig. 1. An example of optimizing $C_p(L^*S)$ by decreasing $D(L^*)$ by ($\Delta=10$). (a) $L^* = P$ and primary input to G_b is controlling value (b) $L^* = P$ and primary input to G_b is non-controlling value

반면에 게이트의 주입력이 비제어 값일 경우는 좀 다르다. Loose path sensitization criterion에 의해 주입력이 비제어 값일 때 부입력도 비제어 값 이어야 한다. 그림 1(b)에서 $L^* = P$ 이고 게이트 G_a 의 전파지연이 $\Delta=10$ 만큼 감소하도록 더 큰 단위로 사이징 될 때 G_b 의 주입력은 비제어 값을 유지 한 채 $40 - \Delta = 30$ 로 도착 시간이 줄어 든다. Loose path sensitization에서 게이트 사이징 후에도 경로 P 는 여전히 감작 경로로 남아 있게 된다. 그러나, 이 회로의 최장 감작 경로는 99의 지연을 갖는 경로 Q 가 된다. 그러므로 G_a 를 사이징 한 후 클럭 주기는 $C_p(L^*S) = 99 - 50 = 49$ 가 된다. 이 예제에서 최장 감작 경로와 두 번째로 긴 감작 경로는 같은 게이트를 공유하므로 게이트 G_b 를 선택하는 것이 더 좋은 결과를 얻을 수 있다. 이러한 관점에서 게이트 선택 방법이 다음에 기술 되어 있다.

$G(L^*\bar{S})$ 에 있는 각각의 게이트 G_i 를 지나는 최단 위상 경로의 지연을 먼저 계산한다. 게이트 G_i 를 지나는 최단 위상 경로의 지연을 계산하기 위해서 소스 노드 (s)와 싱크 노드 (t)를 먼저 생성하고 각각 입력과 출력에 연결한다. (s)에서 G_i 까지의 최단 위상 경로의 지연과 G_i 에서 (t)까지의 최단 위상 경로의 지연을 더해서 전체 최소 지연을 구할 수 있다. (s)에서 G_i 까지의 최단 위상 경로의 지연이 $\min_delay_from_source(G_i)$ 로 정의 되고 G_i 에서 (t)까지의 최단 위상 경로의 지연이 $\min_delay_to_sink(G_i)$ 로

정의 될 때 G_i 를 지나는 최단 위상 경로의 지연은 다음 공식에 의해 계산 된다.

$\min_delay_from_source(G_i) \leftarrow$

$$\min_{k \in pred(G_i)} \{\min_delay_from_source(G_k) + d(G_k)\}$$

$\min_delay_to_sink(G_i) \leftarrow$

$$\min_{k \in succ(G_i)} \{\min_delay_to_sink(G_k) + d(G_k)\}$$

$$\min_delay(G_i) = \min_delay_from_source(G_i) + \min_delay_to_sink(G_i)$$

여기서 $pred(G_i)$ 은 G_i 의 모든 선행자의 집합이고, $succ(G_i)$ 은 G_i 의 모든 후속자의 집합이다. $\min_delay_from_source(G_i)$ 을 계산하기 위해 게이트는 G_i 에서 (s)까지 역으로 탐색 될 수 있다. 마찬가지로 G_i 에서 (t)까지의 최단 경로의 지연은 G_i 에서 (t)로 순방향 탐색에 의해 계산 될 수 있다. G_i 를 지나는 최장 위상 경로의 지연은 약간의 변경만으로 계산 될 수 있다.

$$\max_delay(G_i) = \max_delay_from_source(G_i) + \max_delay_to_sink(G_i)$$

마지막으로 $D(L^*)$ 를 감소시키기 위한 게이트 선택 방법이 다음에 나타내었다.

$C_p(L^*S)$ 에서 $D(L^*)$ 을 감소하기 위한 게이트 선택 방법

$\min_{G_i} \{ \max_{si \in side_inputs_of_G_i} \{ \max_{ay(si)} \} - D(L^*) \}$ 을 가진 게이트 G_i 는 Δ 만큼의 지연을 사이징 할 후보가 된다. 여기서 $G_i \in a$ set of gates $C_p(L^* \bar{S})$ 인 게이트 G_i 는 $\min_delay(G_i) \geq D(S) + \Delta$ 을 만족 한다.

정리 1 $D(L^*)$ 을 감소시키기 위해 게이트 선택 알고리즘에 의해 선택된 게이트를 사이징 한 후 새로운 클럭 주기 $C_p(L^*S)$ 는 이전의 클럭 주기 $C_p(L^*S)$ 보다 작거나 같다.

증명) 정리는 $D(L^*_{new}) \leq D(L^*_{old})$ 과 $D(S_{new}) \geq D(S_{old})$ 을 보임으로써 증명 될 수 있다. 여기서, L^*_{old} 과 L^*_{new} 은 각각 사이징 전 후의 최장 감작 경로이고

S_{old} 와 S_{new} 도 사이징 전 후의 최단 위상 경로이다.

사이징 전에는 $G_i \in G(L^* \bar{S})$ 인 게이트 G_i 에 있어 이므로 $\min_delay(G_i) \geq D(S) + \Delta$ 을 증명하는 것은 쉽다. Δ 만큼 G_i 의 지연을 줄여서 새로운 최단 경로인 S_{new} 가 되도록 G_i 가 최단 경로에 있다고 가정하자. 여기서 $D(S_{new}) = \min_delay(G_i) - \Delta$ 가 된다. 위의 식에서 양변에서 Δ 를 뺀다면 $\min_delay(G_i) - \Delta \geq D(S_{new})$ 를 얻을 수 있다.

$D(L^*_{new}) \leq D(L^*_{old})$ 를 증명하기 위해 경로 $P = \{\dots, G_i, f_{i+1}, G_{i+1}, \dots\}$ 는 최장 감작 경로이며 게이트 G_i 는 $D(P)$ 를 감소시키기 위해 크게 변환한다고 가정하자. 또한 사이징 후에 $D(L^*_{new}) > D(L^*_{old})$ 가 되며 최장 감작 경로 L^*_{new} 가 되는 경로 $Q = \{\dots, s_{i+1}, G_{i+1}, \dots\}$ 가 있다고 가정한다. 게이트 G_i 의 출력인 f_{i+1} 가 후속 게이트 G_{i+1} 로의 제어 입력이라면 경로 P 는 loose path sensitization criterion에 의해 사이징 후에 L^* 가 된다. 그러므로 f_{i+1} 와 s_{i+1} 둘 다 G_{i+1} 로의 비제어 입력이어야만 경로 Q 가 $D(L_{new}) = D(Q)$ 의 지연을 가지는 새로운 최장 감작 경로인 L_{new} 가 된다. 그러나 f_{i+1} 와 s_{i+1} 이 게이트 G_{i+1} 에 대한 비제어 입력이므로 사이징 전에 경로 P 의 지연은 경로 $Q(D(P) > D(Q))$ 의 지연보다 크거나 같다. 그러므로 $D(L^*_{new}) \leq D(L^*_{old})$ 임이 증명 된다. Q.E.D. 이제 $D(S)$ 를 증가 시켜서 $C_p(L^* S)$ 를 최적화하는 경우를 생각해 보기로 하자. 사이징 할 게이트는 $G(L^* \bar{S})$ 의 집합에 속해야 한다. 여기서 $G(L^* \bar{S}) = G(S) - (G(S) \cap G(L^*))$ 로 정의 된다. 게이트 선택을 위한 휴리스틱 방법이 다음에 서술 되었다.

$C_p(L^* S)$ 에서 $D(S)$ 를 증가 시키기 위한 게이트 선택 방법

$\min_{G_i} \{ \min_{si \in side_inputs_of_G_i} \{ \min_delay(si) \} \} - D(S)$ 을 가진 게이트 G_i 는 Δ 만큼의 지연을 사이징 할 후보가 된다. 여기서 $G_i \in a \text{ set of gates } C_p(L^* \bar{S})$ 인 게이트 G_i 는 $\max_delay(G_i) \leq D(L^*) - \Delta$ 을 만족한다.

정리 2 $D(S)$ 을 증가 시키기 위한 게이트 선택 방법에 의해 선택된 게이트를 사이징 한 후에 최장 감작 경로의 지연은 변화하지 않는다.

(증명) 게이트 선택 방법에 의해 $D(L^*_{new}) = D(L^*_{old})$ 와 $D(S_{new}) \geq D(S_{old})$ 을 보임으로써 증명할 수 있다. 게이트 $C_p(L^* \bar{S})$ 의 집합은 사이징 후에 새로운 최단 위상 경로의 지연이 이전의 최단 위상 경로 보다 크거나 같음을 보이는 것은 쉬운 일이다. $D(L^*_{new}) = D(L^*_{old})$ 조건은 수학적 귀납법에 의해 증명 될 수 있다.

그래프에 있는 노드 V 와 간선 E 에서 $\min_delay(G_i)$ 또는 $\max_delay(G_i)$ 의 시간 복잡도는 $O(V + E)$ 이 된다. 그러므로 위의 게이트 선택 방법에 의한 시간 복잡도는 $O(m(V + E))$ 이 되고 여기서 m 은 최장 감작 경로에 있는 게이트의 수이다.

2. $C_p(L S^*)$ 를 최적화하기 위한 게이트 선택

$C_p(L S^*) = D(L) - D(S^*) + t_c$ 에 의해 결정되는 클럭 주기는 $D(L)$ 를 감소 시키거나 $D(S^*)$ 를 증가 시킴으로써 최적화 될 수 있다. L 에 있는 게이트가 $D(S^*)$ 를 바꾸지 않고 $D(L)$ 를 감소시키기 위해서 사이징 될 때 선택될 게이트는 게이트 $G(L \bar{S}^*)$ 의 집합에 속하게 된다. 여기서 $G(L \bar{S}^*) = G(L) - (G(L) \cap G(S^*))$ 이다. 게이트 선택을 위한 휴리스틱 방법은 다음과 같다.

$C_p(L S^*)$ 에서 $D(L)$ 를 감소하기 위한 게이트 선택 방법

$\min_{G_i} \{ \max_{si \in side_inputs_of_G_i} \{ \max_delay(si) \} - D(L^*) \}$ 을 가진 게이트 G_i 는 Δ 만큼의 지연을 사이징 할 후보가 된다. 여기서 $G_i \in$ 게이트 집합 $G(L \bar{S}^*)$ 인 게이트 G_i 는 $\min_delay(G_i) \geq D(S^*) + \Delta$ 을 만족한다.

정리 3 $D(L)$ 을 감소시키기 위한 게이트 선택 방법에 의해 사이징한 후 새로운 클럭 주기 $C_p(L S^*)$ 는 이전의 클럭 주기 $C_p(L S^*)$ 보다 작거나 같다.

(증명) 정리는 $D(L_{new}) \leq D(L_{old})$ 이고 $D(S^*_{new}) \geq D(S^*_{old})$ 임을 보임으로써 증명 될 수 있으며 여기서는 증명을 생략한다.

$C_p(L S^*)$ 에 의한 클럭 주기는 $D(S^*)$ 를 증가 시킴으로써 최적화 될 수 있다. 그때 $G(L \bar{S}^*)$ 에 있는 게이트는 최단 불안정 감작 경로의 지연을 증가하기 위

해서 더 작은 단위로 변환될 수 있다. 여기서 $G(\bar{L}S^*) = G(S^*) - (G(L) \cap G(S^*))$ 이다. 게이트 선택의 휴리스틱 방법이 아래에 제시 되었다.

$C_p(LS^*)$ 에서 $D(S^*)$ 를 증가시키기 위한 게이트 선택 방법

$\min_{G_i} \{\min_{si \in \text{side_inputs_of } G_i} \{\min_delay(si)\} - D(L^*)\}$ 을 가진 게이트는 만큼의 지연을 사이징 할 후보가 된다. 여기서 게이트 집합 인 게이트는을 만족한다.

정리 4 $D(S^*)$ 를 증가시키기 위해 게이트 선택 방법에 의해 선택된 게이트를 사이징한 후 새로운 클럭 주기 $C_p(LS^*)$ 는 이전의 클럭 주기 $C_p(LS^*)$ 보다 작거나 같다.

증명) 정리는 $D(L_{new}) \leq D(L_{old})$ 이고 $D(S^*_{new}) \geq D(S^*_{old})$ 임을 보임으로써 증명 될 수 있으며 여기서는 증명을 생략한다.

3. 알고리즘

본 논문에서는 클럭 주기 $C_p(LS)$, $C_p(LS^*)$, 그리고 의 세 가지 설정을 이용한 게이트 선택 방법이 제안되었다. 긴 경로에 있는 게이트가 경로의 지연을 감소시키기 위해서 더 큰 단위(더 큰 캐패시턴스)로 변환될 때 증가된 캐패시턴스는 후속 게이트의 증가된 지연에 대해서 최적화가 어렵게 된다. 더욱이, 더 작은 단위(더 작은 캐패시턴스)로 게이트를 사이징함으로서 짧은 경로의 지연을 증가 시킬 경우 후속 게이트에 대한 영향에 의해 최적화 과정이 느려진다. 게이트 사이징에 의한 이러한 간접 영향은 단순화를 위해 생략 했지만 앞 노드의 캐패시턴스를 증가 시킴으로써 쉽게 고려 할 수 있다.

$C_p(L^*S)$ 에 의한 회로 성능 최적화 알고리즘이 표 1에 제시 되었다. 초기화 단계로서 L, S, L^* , 그리고 S^* 이 II장에 나타난 알고리즘에 의해 계산 된다. 클럭 주기 $C_p(L^*S)$ 이 요구되는 클럭 주기 C_p^* 보다 큰 경우, 사이징할 게이트는 $C_p(L^*S)$ 에서 $D(S)$ 을 증가시키기 위한 게이트 선택 방법에 의해 선택된다. 선택된 게이트를 사이징 한 후 새로운 L, S, L^*, S^* , 그리고 $C_p(L^*S)$ 이 다음 반복 단계의 게이트 선택을 위해 계산 된다. 게이트 선택 휴리스틱의 결과가 empty 일 경우, 알고리즘이 멈추고 새로운 클럭 주기

$C_p(L^*S)$ 이 마지막 결과로서 반환 된다. 그렇지 않을 경우, 또 다른 게이트가 선택되고 $C_p(L^*S)$ 값이 더 이상 감소 하지 않을 때까지 위의 과정이 반복 된다. 더 작은 단위로 게이트를 사이징한 후 클럭 주기 $C_p(L^*S)$ 가 여전히 C_p^* 보다 클 경우, 게이트는 $D(L^*)$ 를 감소시키기 위한 게이트 선택 방법에 의해 선택된 게이트가 사이징 되게 된다. 클럭 제한 $C_p(L^*S)$ 을 만족 시키기 위해서, 게이트들을 $D(L^*)$ 을 감소시키기 위해 반복해서 사이징 한다. 클럭 주기 $C_p(L^*S)$ 이 여전히 C_p^* 보다 크다면 사이징을 위한 단위를 더 작은 값으로 변형 한 후 위의 알고리즘을 반복 한다.

표 1. $C_p(L^*S)$ 를 최적화하는 알고리즘

Table 1. Algorithm of Optimizing $C_p(L^*S)$.

```

Compute  $L, S, L^*, S^*$  of the circuit by Loose Path
Sensitization Criterion
 $C_p(L^*S) \leftarrow D(L^*) - D(S) + t_c$ 
 $\Delta_{inc} \leftarrow factor 1$ 
 $\Delta_{dec} \leftarrow factor 2$ 
if  $C_p(L^*S) \leq C_p^*$ 
    done
else
repeat
repeat
    Select a gate  $G_i$  by
    Gate_selection Method for increasing
     $D(S)$  in  $C_p(L^*S)$ 
    resize  $G_i$  by  $d(G_i) \times \Delta_{inc}$ 
    Compute the new  $L, S, L^*, S^*$  of the
    circuit
     $C_p(L^*S) \leftarrow D(L^*) - D(S) + t_c$ 
until  $((C_p(L^*S) \leq C_p^*) \text{ or } (\text{Gate_selection Method return nil}))$ 
repeat
Select a gate  $G_i$  by
Gate_selection Method for decreasing
 $D(L^*)$  in  $C_p(L^*S)$ 
resize  $G_i$  by  $d(G_i) \times \Delta_{dec}$ 
Compute the new  $L, S, L^*, S^*$  of the
circuit
 $C_p(L^*S) \leftarrow D(L^*) - D(S) + t_c$ 
until  $((C_p(L^*S) \leq C_p^*) \text{ or } (\text{Gate_selection Method return nil}))$ 
 $\Delta_{inc} \leftarrow new\_factor 1$ 
 $\Delta_{dec} \leftarrow new\_factor 2$ 
until  $((C_p(L^*S) \leq C_p^*) \text{ or } (\text{Gate_selection Method return nil}))$ 
return  $C_p(L^*S)$ 

```

V. 실험 결과

제시된 알고리즘은 C언어를 사용해서 구현되었고 ISCAS85 벤치 마크 회로를 사용해 Sparc Workstation에서 검증 되었다. 사이징 요소로서, 케이트의 지연은 단순화를 위해 원래의 지연의 2배로 증가 하거나 1/2로 감소한다고 가정하였다. 전체 회로 면적에서의 효율성을 위해 짧은 경로에 있는 게이트들이 긴 경로에 있는 게이트 보다 먼저 사이징 된다. 짧은 경로에 있는 게이트를 더 느리게 변환 되면 회로의 전체 캐페시턴스가 줄어 드는 장점이 있다. 그러므로 클럭 주기를 줄이면서 성능을 최적화 하는 동안 회로의 전력 소모가 줄어들게 된다. 요구되는 클럭 주기 C_p^* 가 $C_p(LS)$ 의 클럭 주기의 80%로 가정하고 C_p^* 에 도달 할 때까지 반복적으로 사이징을 수행한다.

제안된 게이트 선택 방법을 사용 했을 때와 사용하지 않았을 때의 클럭 주기의 세 가지 다른 세팅에 따라서 알고리즘이 구현되고 실험 되었다. [5]에 있는 방법을 사용해서, $C_p(LS)$ 를 최적화 하기 위해 $G(\bar{L}S)$ 에 속하는 게이트들이 $D(S)$ 를 증가 시키기 위해 임의로 선택 된다. 그리고 나서 $G(L\bar{S})$ 에 속하는 게이트들이 $D(L)$ 을 감소 시키기 위해 임의로 선택 된다. 세가지 클럭 세팅에 따른 최적화와 제안된 게이트 선택 방법에 의해 게이트 사이징을 수행한 결과가 표 2에 제시 되었다. 예측 했던 것처럼 (A)와 (B) 게이트 선택 방법 모두 $C_p(L^*S)$ 와 $C_p(LS^*)$ 에 의한 알고리즘이 $C_p(LS)$ 보다 더 좋은 결과를 얻었다. 그러나 감작 경로에 의한 타이밍 분석에 의해 계산 시간이 더 걸린다. 반면에, $C_p(L^*S)$ 와 $C_p(LS^*)$ 의 알고리즘은 회로의 특성에 따라 약간의 차이 밖에 없는 거의 같은 결과를 내었다.

VII. 결 론

이 논문에서는 긴 감작 경로와 짧은 불안정 감작 경로 상에 있는 게이트들을 사이징 함으로서 클럭 주기를 최적화하는 새로운 알고리즘을 제시하였다. 클럭 주기에 의한 성능 최적화는 경로의 지연을 바꿈으로써 얻어질 수 있기 때문에 최단 경로와 최장 경로 사이의 지연 차이를 감소시키기 위해 지연을 늘이거나 줄여야 하는 게이트들은 신중하게 결정 되어야 한다.

표 2. ISCAS 벤치 마크 회로의 클럭 주기 최적화 실험 결과 (a) 케이트 사이징 방법 [5] (b) 제안된 사이징 방법

Table 2. Clock Period Optimization on ISCAS85 Benchmark Circuits. (a) Gate Sizing Method in [5] (b) Proposed Sizing Method

Circuit	C432	C499	C880	C1355	C1908	C2670	C3540	C5315	C7552
$C_p(LS)$	50.80	32.00	49.00	31.90	69.60	86.90	88.40	97.90	85.30
	50.80	32.00	49.00	31.90	59.70	85.10	79.90	94.30	83.80
	50.80	31.80	49.00	31.90	69.60	84.60	79.90	97.90	84.10
Target C_p	40.64	25.60	39.20	25.52	55.68	69.52	66.72	78.32	68.24
A LS	Gates Sized	56	250	3	224	24	34	148	58
	Increased Area(%)	47.2	183	12	98	48	54	105	45
	CPU(sec)	23.1	28.1	14	210.4	384.2	204	142.5	50.4
A LS	Gates Sized	49	242	3	248	4	28	57	45
	Increased Area(%)	41.9	143	13	182	0.0	44	58	29
	CPU(sec)	44.5	280.2	41	490.3	731.3	77.4	671.3	144.2
A LS	Gates Sized	50	237	3	249	7	34	60	53
	Increased Area(%)	42.4	143	12	19.0	-0.1	40	55	28
	CPU(sec)	39.6	310.3	30	313.4	980.4	802	664.4	160.5
B LS	Gates Sized	7	101	2	125	30	31	118	78
	Increased Area(%)	0.6	-25	4	11.6	2.5	29	9.5	4.7
	CPU(sec)	3.8	33.9	15	125.5	399.1	21.5	102.2	62.1
B LS	Gates Sized	7	101	2	93	2	27	33	88
	Increased Area(%)	0.6	-25	6	0.8	0.0	26	15	5.7
	CPU(sec)	10.2	122.5	21	202.5	6889	50.7	5312	2023
B LS	Gates Sized	8	101	2	91	25	26	33	63
	Increased Area(%)	0.4	-24	6	0.8	-0.3	25	1.6	5.5
	CPU(sec)	12.4	147.3	20	180.0	6757	44.9	4906	190.4

제안된 게이트 선택 방법에 의해 요구되는 클럭 주기는 적은 수의 게이트 사이징과 적은 수행 시간으로 구해질 수 있다. 실험 결과로 성능에서 20%의 이득을 얻기 위해서는 거의 모든 회로의 경우 면적이 5% 이하로 증가 ($C499$ 와 $C1908$ 회로는 면적이 감소 하였음) 하였다.

참 고 문 헌

[1] D. Brand and V. Iyengar, Timing Analysis

- using Functional Analysis, Technical Report, IBM Thomas J. Watson Research Center, 1986.
- [2] P. McGeer and R. Brayton, Efficient Algorithms for Computing the Longest Viable Path in a Combinational Network, *26th DAC*, 1989, PP 561-567.
- [3] H. Chen and David Du, Path Sensitization in Critical Path Problem, *ICCAD*, 1991
- [4] H-C. Chen, Study of Timing Verification and Timing Optimization of Combinational Circuits, Ph.D. Thesis, Dept. Of Computer Science, Univ. of Minnesota, Feb. 1992
- [5] S. Cheng, H. Chen, David Du, and A. Lim, The Role of Long and Short Paths in Circuit Performance Optimization, *29th DAC*, 1992.
- [6] L. Cotton, Maximum-rate pipeline sys-
- tems, *AFIPS Proc.*, 1969 Spring Joint Computer Conference, vol 34, pp 581-586.
- [7] B. Ekroot, Optimization of pipelined processors by insertion of combinational logic delay, Ph.D. Thesis, Department of Electrical Engineering, Stanford University, Sept. 1987.
- [8] S. Yen, D. Du, and S. Ghanta Efficient Algorithms for Extracting K Most Critical Paths in Timing Analysis, *International journal of computer Aided VLSI Design* 3, 1991, pp 193-215.
- [9] J. Kim, Y. Hsu, and D. Du A New Gate Selection Method for Resizing to Circuit Performance Optimization, *Proceedings of International Symposium on Circuits and Systems* 1996, vol. 4, pp 461-464

저자 소개



金 周 虎(正會員)

1963년 10월 26일생. 1987년 미네소타 주립대학 전산과 학사. 1995년 미네소타 주립대학 전산과 박사. 1995년 ~ 1996년 Cadence Design Systems Inc. San Jose, California, Senior Member of Technical Staff. 현재 서강대학교 전자계

산학과 조교수로 재직. 주관심분야는 CAD 및 하드웨어 시스템 설계등임