크로스톡 회피를 위한 게이트 사이징을 이용한 타이밍 윈도우 이동

이 형 우, 장 나 은, 김 주 호
서강대학교 전자계산학과
전화 : 02-706-3997 / 핸드폰 : 011-1761-1267

Timing Window Shifting by Gate Sizing for Crosstalk Avoidance

Hyungwoo Lee, Naeun Jang, Juho Kim
CAD & VLSI Lab. Dept. of Computer Science Sogang University
E-mail : hwlee@sogang.ac.kr

Abstract

This paper presents an efficient heuristic algorithm to avoid crosstalk which effects to delay of CMOS digital circuit by downsizing and upsizing of Gate. The proposed algorithm divide into two step, step1 performs downsizing of gate, step2 performs upsizing, so that avoid adjacent aggressor to critical path in series. The proposed algorithm has been verified on LGSynth91 benchmark circuits and Experimental results show an average 8.64% Crosstalk Avoidance effect. This result proved new potential of proposed algorithm..

1 서론

휴대용 계산기기와 모바일 시스템의 급격한 성장으로 고성능 고집적화에 대한 요구가 증가하고 있다. 하지만 고성능 고집적화에 따른 노이즈의 효과는 더욱 증폭되고 있는 실정이다. 현재 누락전류(crosstalk)에 대한 접근 방법은 회로에 대한 최악의 경우 지연시간(worst case delay)을 구하기 위한 노이즈의 분석에 중점을 두고 있다. 이 접근 방법은 누락전류를 어떻게 모델링하며, 내부결선간의 거리는 가까우나 논리적으로 영향을 미치지 않는 어그레서들이나, 시간적으로 서로 영향을 미치지 않는 어그레서들을 찾아서 최악 경우 지연시간의 계산에 참고하지 않는 것이다.

일반적으로 게이트 사이징(gate sizing)은 CMOS 디지털 회로에서 신호 천이에 의해 발생하는 동적 전력 소모(dynamic power dissipation)에서 기능에 영향을 미치지 않는 불필요한 기능성 천이(functional transition)인 글리치(glitch)를 제거하기 위해서 사용되어 왔다. 본 논문에서는 누락전류를 게이트 사이징으로 회피하는 방법을 제안할 것이다. 이 방법은 게이트 사이징에 대한 이해와 누락전류의 이해에서 출발하며 아직까지 시도되지 않은 알고리즘이다.

논문은 다음과 같이 구성된다. 제 2장에서는 기존 프루닝 기법에 의해 언급하며, 제 3장에서 흐름도 및 알고리즘에서 사용될 timing window model 및 게이트 사이징에 의한 누락전류의 회피에 대해서 논의하며 4장에서는 구현된 알고리즘에 의한 실험결과를 보여주며 마지막으로 5장에서는 앞으로 계획하고 있는 연구에 대한 언급을 하며 결론으로 끝을 맺는다.

2 이론적 배경

2.1 Temporal Pruning

누락전류 노이즈는 넷 각각의 타이밍 윈도우와 관련이 있다. 예를 들어 그림 4에서 A1의 타이밍 윈도우가 V의 타이밍 윈도우와 겹쳐 있는데 이것은 A1이 V에게 영향을 미친다는 것을 의미한다. 반면에 A2는 V와 겹치지 않는데 이것은 넷 V에게 어떠한 누락전류 노이즈도 발생하지 않는다는 것을 의미한다. 이러한 접근은 정적 노이즈 분석 모델(static noise analysis model)에 반영되어 있는데 기능적(functional) 정보는 무시한 채 시간(timing) 정보만 포함된 것이다.

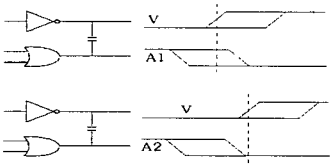

그림 1 Temporal Relationship between Victim Net and Aggressor Nets

2.2 Functional Pruning

경로의 기능적인(functional) 관계에 의해서 노이즈에 대해서 절대 반응하지 않는 경로(path)들을 어그레서에서 제거하는 것이며 그 알고리즘에 따라 이진 결정 다이어그램(BDD)을 이용하거나 path sensitization을 이용하기도 한다.[5] 이 접근은 zero-delay model을 가정하고 있으며 역시 시간(temporal) 정보는 무시된다.

3 게이트 사이징을 이용한 크로스톡 회피

기존의 방법은 누락전류의 분석에 의미를 두어왔으며 더욱 정확한 분석을 통해서 functional, temporal pruning을 수행했다. 하지만 보다 적극적인 방법으로 전력 감소에만 쓰이던 게이트 사이징(gate sizing)을 사용하여 누락전류가 발생하는 부분을 가능한 회피할 수 있는 휴리스틱 알고리즘을 본 논문에서 제시할 것이다. 본 논문은 누락전류의 기능적인(functional)적인 측면은 고려하지 않는다.

3.1 알고리즘 개념도

전체적인 알고리즘 흐름도는 그림 5와 같다. Crosstalk Avoidance 알고리즘의 입력으로는 회로에 대한 도착시간(arrival time)과 요구시간(require time) 그리고 회로의 어그레서와 빅팀 정보이다. 출력은 그림과 같이 게이트 사이징으로 수정된 crosstalk avoided netlist이다.

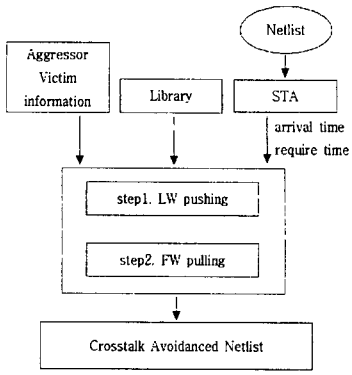

그림 5 Overall Flow Graph

3.2 timing window modeling

기존의 타이밍 윈도우 모델링과는 달리 본 논문에서는 빅팀과 어그레서의 구분을 두지 않는다. 단지 둘 중 어떤 타이밍 윈도우가 우선하느냐에 따라서, 우선하는 노드의 타이밍 윈도우를 FW(Former Window)로 정의하며 우선하지 않는 노드의 타이밍 윈도우를 LW(Latter Window)로 정의한다. FWLT(Former Window Latest Time)는 FW의 스위칭이 끝나는 시간을 가리키며 LWET(Latter Window Earliest Time)는 LW의 스위칭이 시작하는 시간을 가리킨다. Td(timing difference)는 FWLT에서 LWET를 뺀 값으로 정의한다. 그림 3.2.1은 본 논문에서 사용하는 타이밍 윈도우 모델링을 표현하고 있다.

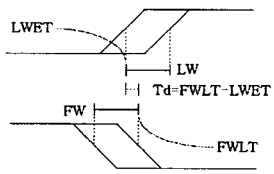

그림 6 Timing Window modeling

3.3 제안된 크로스톡 회피 알고리즘

3.3.1 step1 LW pushing

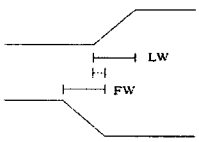

그림 7
누락전류(crosstalk) 발생

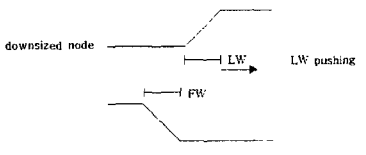

그림 8 LW pushing (slack≥Td)

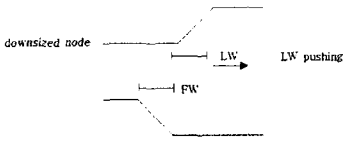

그림 9 LW pushing (slack<Td)

그림 7과 같이 타이밍 윈도우가 서로 겹치게 되어서 누락전류가 발생하는 경우 LW측의 노드의 게이트의 사이즈를 작게 하여 게이트의 동작 속도를 느리게 한다. 그렇게 하면 그림 8와 같이 서로 겹치는 부분은 사라지게 되며 누락전류를 회피하게 된다. 만약 LW의 슬랙이 Td보다 작다면 슬랙만큼 pushing을 수행한다. 이 경우는 그림 9에 나타나 있으며 완벽한 회피가 되지는 않는다. 하지만 그림 7보다는 겹치는 부분이 적어지므로 그만큼 누락전류의 효과가 약해진다는 것을 의미한다. 여기서 고려해야 할 사항은 해당 노드의 슬랙이다. 슬랙을 넘어서 pushing을 할 경우에 다음노드의 도착시간(arrival time)이 요구시간(required time)을 넘어서는 경우가 발생하므로 절대 슬랙을 넘어서 pushing을 하지는 않는다.

만약 라이브러리에 요구조건을 만족하는 delay를 가진 게이트가 존재하지 않을 경우 step1에서는 회피를 고려하지 않는다.

3.3.2 step2 FW pulling

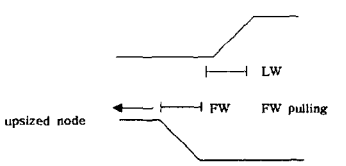

그림 10 FW pulling ($d(G_j) \geq Td$)

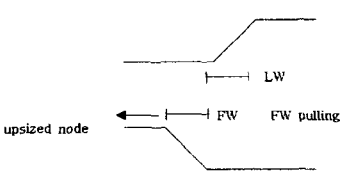

그림 11 FW pulling ($d(G_j) < Td$)

그림 7과 같이 타이밍 윈도우가 서로 겹치게 되어서 누락전류가 발생하는 경우 FW측의 노드의 게이트의 사이즈를 크게 하여 게이트의 동작 속도를 빠르게 한다. 그렇게 하면 그림 10과 같이 서로 겹치는 부분은 사라지게 되며 누락전류를 회피하게 된다. 만약 FW측 노드의 게이트를 가장 큰 게이트를 넣어도 이전 게이트와 사이징된 게이트의 차가 Td를 넘지 못할 경우 가능한 큰 게이트로 사이징함으로써 가능한 만큼만 pulling을 수행한다. 이 경우는 그림 11에 나타나 있으며 그림 9의 pushing 경우처럼 완벽한 회피가 되지는 않는다. 하지만 그림 7보다는 겹치는 부분이 적어지므로 그만큼 누락전류의 효과가 약해진다는 것을 의미한다.

만약 라이브러리에 이러한 delay를 만족하는 게이트가 존재하지 않을 경우 step2에서 역시 회피를 고려하지 않는다.

FW pulling의 경우 FW를 당기다 보면 앞 노드의 타이밍 윈도우와 겹칠 수 있다. 이는 또 다른 누락전류를 발생시키는 요인이 된다. 이 경우를 회피하기 위해서 앞의 노드와 타이밍 윈도우가 겹치는 경우는 step2: FW pulling의 대상에서 제외한다.

3.3.3 전체적인 알고리즘

그림 5의 알고리즘 흐름도에서 묘사됐듯이 정적 시간 분석기(Static Timing Analysis)에 의해 시간(delay) 정보를 가진 넷리스트(netlist), 빅팀 어그레서 정보, 그리고 라이브러리를 입력으로 받는다. 주어진 입력으로 임계경로를 탐색하면서 step1: LW pushing, gate downsizing, step2: FW pulling, gate upsizing으로 gate sizing을 하며 사이징이 이루어 질 때마다 전체 지연시간 정보를 갱신하며 제거된 어그레서를 회로에 반영한다. 그리고 누락전류 회피 알고리즘에 의해서 게이트 사이징된 넷리스트를 출력한다.

4 실험 결과

제안된 알고리즘은 Ultra Sparc workstation상에서 C 언어로 구현되었으며 *LGSynth91* 벤치마크 회로를 사용해 검증되었다. 회로들의 초기 매핑은 SIS[1]를 이용하였고 실제공정에서 사용되는 0.5㎛ 표준 셀 라이브러리를 목표로 하였다. 실험에서 사용된 라이브러리는 각각 5종류의 사이즈를 가지는 AND, OR, NAND, NOR, BUFFER, INVERTER로 구성되어 게이트의 INERTIAL DELAY는 게이트의 전파 지연과 같은 크기를 갖는 것으로 가정하였다. 표 1에 실험결과를 제시하였다. 표 1의 '# of aggressor by critical path'항목은 입력으로 제공되는 회로의 임계경로에 인접한 어그레서의 개수이며 '# of avoided aggressor'항목은 논

문에서 제공되는 알고리즘을 적용한 프로그램을 거친 뒤 제거된 어그레서의 개수이다. 그리고 'Aggressor avoidance ratio'는 제거된 어그레서에 대한 비율을 나타내고 있다. 제시한 Crosstalk Avoidance by gate sizing 알고리즘을 수행한 후 평균적으로 8.64%의 어그레서 감소가 있었다. 실험결과에서 나타나는 어그레서의 감소는 일반적으로 언급되는 활동하지 않는 어그레서의 제거가 아닌 실제 어그레서의 제거를 의미한다.

Circuit Name	#of gates	# of Nodes	# of aggressor by critical path	# of avoided aggressor	Aggressor avoidance ratio (%)
comp	212	244	12	1	8.3
C432	222	258	16	2	12.5
C880	378	438	24	2	8.3
C499	602	643	28	3	10.7
C1908	672	705	39	3	7.7
rot	796	931	45	3	6.7
C1355	978	1019	61	6	9.8
C3540	1321	1371	65	4	6.1
pair	2160	2333	103	8	7.7
Average					8.64

표 1 Crosstalk Avoidance by Gate Sizing 실험결과

5 결 론

본 논문은 임계경로의 delay에 영향을 미치는 어그레서들을 게이트 사이징을 이용한 누락전류 회피 알고리즘으로 최대한 회피시키는 방법을 제안했다. 논문의 알고리즘은 step1의 LW pushing과 step2의 FW pulling을 거치면서 게이트의 다운사이징 업사이징을 수행한다. 그 결과는 기대한 만큼 누락전류 회피 효과가 있었으며 앞으로 deepsub micron 공정이 점점 진행됨에 따라 더욱 높은 효율이 기대된다. 더군다나 이 알고리즘은 기존에 나왔던 일반적인 분석 중심의 프루닝(pruning)이 아닌 여태까지 시도하지 않았던 다른 방법인 적극적인 프루닝이 가능하다는 것을 입증한 데에 의의를 가진다. 특히 기존의 분석 접근 방법의 프루닝과는 전혀 다른 개념의 알고리즘이므로 이전 프루닝 방법과 병행 사용이 가능하다. 그리고 본 논문에서 제시된 실험으로 게이트 사이징이 글리치 제거에 의한 전력 최적화(power optimization)에만 사용되지 않고 누락전류 제거에까지 사용될 수 있음을 보여준다.

앞으로 연구할 과제는 스케줄링 알고리즘을 도입한 누락전류 회피 알고리즘을 고려하고 있으며 보다 효율적인 알고리즘이 나올 것으로 기대하고 있다.

참고 문헌

[1] Pinhong Chen, Kurt Keutzer, 'Toward True Crosstalk Noise Analysis' on Dept. of EECS, Univ. of California at Berkeley, in 1999 IEEE

[2] Yasuhiko Sasaki, Giovanni De Micheli, 'Crosstalk Delay Analysis using Relative Window Method' on stanford University Computer System Laboratory, in 1999 IEEE

[3] Tong Xiao, Malgorzata Marek-Sadowska 'Worst Delay Estimation in Crosstalk Aware Static Timing Analysis' on Department of Electrical and Computer Engineering University of California, in 2000 IEEE

[4] Pinhong Chen, Yuji Kukimoto, Kurt Keutzer, 'Refining Switching Window by Time Slots for Crosstalk Noise Calculation', on Dept. of EECS, U.C. Berkeley, Cadence Design System Inc, in 2002 IEEE

[5] Jae-Seok Yang, Jeong-Yeol Kim, Joon-Ho Choi, Moon-Hyun Yoo, Jeong-Taek Kong ,'Elimination of false aggressors using the functional relationship for full-chip crosstalk analysis' on CAE team, Memory Division, Dept. of Device solution Network, Samsung Electronics, in 2003 IEEE

[6] Donald Chai, Alex Kondratyev, Yajun Ran, Kenneth H. Tseng, Yosinori Watanabe, Malgorzata Marek-Sadowska, 'Temporofunctional Crosstalk Noise Analysis', on Cadence Design System, in DAC 2003, June 2-6, Anaheim, California, USA

[7] M. Hashimoto, H. Onodera, and K. Tamaru, A Power Optimization Method Considering Glitch Reduction by Gate Sizing, in Proceedings of the Inter national Symposium on Low Power Design, pp. 221-226, August 1998.

[8] J. Kim, C. Bamji, Y. Jiang, and S. Sapatnekar, Concurrent Transistor Sizing and Buffer Insertion by Considering Cost-Delay Tradeoffs, in Proceedings of the International Symposium on Physical Design, pp. 130-135, April 1997.