

타이밍 분석을 위한 효율적인 시간 지연 계산 도구

김준희[°], 김부성, 갈원광, 맹태호, 백종홍, 김석윤
숭실대학교 컴퓨터학부

An Efficient Delay Calculation Tool for Timing Analysis

Joon-Hee Kim[°], Boo-Sung Kim, Won-Koang Kal, Tae-Ho Maeng, Jong-Hyun Baek, Seok-Yoon Kim
Soongsil Univ. School of Computing

Abstract - As chip feature size decrease, interconnect delay gains more importance. A accurate timing analysis required to estimate interconnect delay as well as cell delay. In this paper, we present a timing-level delay calculation tool of which the accuracy is bounded within 10% of SPICE results. This delay calculation tool generates delay values in SDF(Standard Delay Format) for parasitic data extracted in SPEF(Standard Parasitic Exchange Format). The efficiency of the tool is easily seen because it uses AWE(Asymptotic Waveform Evaluation) algorithm for interconnect delay calculation, and precharacterized library and effective capacitance model for cell delay calculation.

1. 서 론

현 사회가 정보화 사회가 되어감에 따라 정보기기 및 하드웨어의 기능 및 성능에 대한 요구조건도 계속적으로 강화되고 있으며, 이를 기기 및 하드웨어에 사용되는 반도체 칩들 또한 제조 기술의 발달에 힘입어 고집적화, 고속화되어 가고 있다. 미국 SIA(Semiconductor Industry Association)의 연례 보고서에 의하면 향후 10년 이내에 회로선폭 0.1 μm 이내, 2GHz의 칩내부 력속도(on-chip clock rate)를 가진 반도체 소자시대가 도래할 것이라고 예측하고 있다[1]. Feature size 가 작아지고, 동작 주파수가 커짐에 따라 연결선의 시간 지연은 전체 시간지연에서 큰 부분을 차지하게 되었고, 결과적으로 연결선은 고속의 집적회로, multichip, 시스템 등의 성능최적화를 위한 중요한 고려요소가 되었다. 본 논문에서는 날로 중요시되고 있는 칩 상에서의 시간지연 계산에 대하여 보다 빠르고 정확한 측정을 가능하게 하는 도구를 소개한다.

연결선 및 게이트 지연 해석의 중요성은 정확한 타이밍 분석을 필요하게 만들었으며, 근래에 와서는 칩 제작에 있어서 timing analyser의 사용이 당연시되어 폭넓게 사용되고 있다. 이러한 timing analyser의 사용에 의한 시간지연의 측정은 각각의 연결선과 셀의 문제가 아니라 하나의 시스템으로 구성되어 있는 칩 전체의 문제가 되었다. 이러한 칩의 timing level에서의 시간지연 예측은 그 수행시간 때문에 말미암아 SPICE를 사용하여 측정하기에는 부적당하므로 예측시간을 줄일 수 있는 많은 방법들이 연구되어 왔으며, 현재에도 주된 연구 분야로 부각되고 있다. 이중에서 셀 지연의 계산 문제만 보더라도 비선형 회로 해석을 포함하므로 정확한 해석을 위해서는 시간소모가 크다. 그러나 셀을 정밀하게 모델링하여 SPICE와 같은 simulator를 사용하는 방법은 그 수행 속도로 인하여 사용하기에 부적당하다. 이러한 문제를 풀기 위해 많은 회로 수준 도구들이 cell

library의 형태를 이용하는 근사적인 방법을 사용하고 있다. 이 cell library는 미리 cell에 대한 자연시간을 계산하고, 이 값을 데이터베이스화하여 cell의 자연시간을 얻고자 할 때, 셀 데이터베이스로부터 값을 추출하여, 셀의 자연시간을 얻는 방법이다. 이러한 방법은 계산시간을 획기적으로 단축할 수 있으며, 현재의 대부분의 회사에서 이러한 방법을 사용하고 있다. 본 논문에서는 연결선의 시간지연을 측정하는 방법으로는 moment matching 기법을 사용한 AWE기법[5]을 사용하여 SPICE보다 수 배에서 수 만배 이상의 속도를 얻고, 셀의 시간지연 측정을 위해서는 Precharacterization된 셀 library를 사용함으로써 빠른 속도를 얻을 수 있으며, 셀 library에서 값 추출시에 C_{eff} 모델링 방법[4]을 사용하여 보다 정확한 값 추출할 수 있는 도구를 소개한다.

본 도구는 전체 칩 설계 흐름에서 그림 1의 진하게 표시된 위치를 차지하고 있다. Netlist 작성 후의 설계 검증과정에서 논리 검증을 거친 다음, timing level에서 검증을 하게 된다. 각 검증단계에서 오류가 발생시에는 feed-back함으로써 오류부분은 수정한 후에 다시 각 검증 단계를 거쳐서 설계 오류의 유·무를 판단하게 된다.

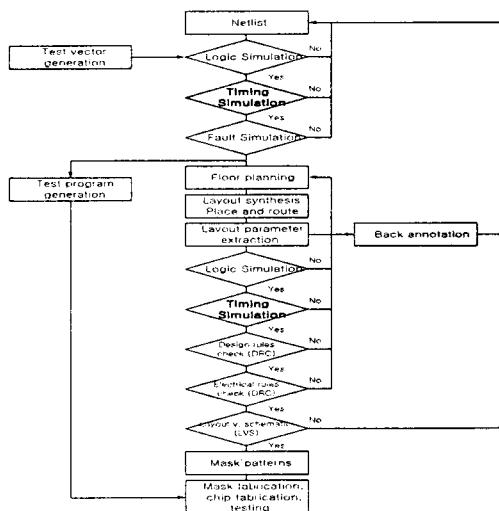


그림 1. 칩 레이아웃 설계의 흐름

각 설계 단계 관점에서 파일의 형태를 보면, 처음의 물리적인 형태의 설계를 표현한 것을 레이아웃 데이터(Layout data)라 한다. 이 데이터는 원하는 회로의 형태를 물리적으로 폭과 길이 등의 요소로서 표현된 데이터를 말한다. 이러한 데이터 형으로는 GDSII 형태와 Magic의 mag 형태 등이 있다. 이러한 데이터 형태는 회로적인 표현으로 교환 가능하게 된다. 즉 각 선폭과

길이 등으로 표현된 물리적 형태를 저항, 커파시턴스, 인덕턴스의 회로적인 기생성분으로 표현가능한데, 이것을 기생 데이터(parasitic data)라 한다. 이러한 데이터의 형태에는 SPEF(Standard Parasitic Exchange Format), SPICE 등이 있다. 이러한 기생 데이터를 입력으로 하여 칩 혹은 셀의 특성 중에서 필요로 하는 시간지연과 같은 timing level에 관련된 여러 가지 정보를 얻을 수 있다. 이러한 정보를 담고 있는 데이터를 timing 데이터라 하며, 이러한 데이터는 SDF(Standard Delay Format)라는 표준화된 파일 규격으로 표현이 가능하다. 본 논문에서 제시한 지연 계산 도구는 이러한 표준화된 폭넓게 사용되어지고 있는 파일 형태를 입·출력으로 사용한다.

아래의 그림2는 각 설계 단계에서 사용되는 파일의 형태이다.

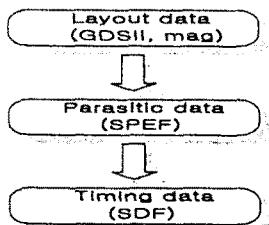


그림 2. 각 설계 단계에서의 파일의 형태

2. 본 론

2.1 시스템의 구성

본 논문의 시간지연 계산 도구는 연결선의 기생성분에 대한 정보를 담고 있는, SPEF 파일형으로 구조화된 파일을 입력으로 받아들여 입력 파일에 표현된 연결선과 관련되는 각 셀의 시간지연과 연결선의 시간지연을 계산하여 각 값을 SDF형의 구조를 갖는 출력 파일로 생성하는 도구이다.

즉, 다음 그림과 같이 본 도구는 SPEF 파일을 입력으로 하고 SDF 파일을 출력으로 갖고 있다.

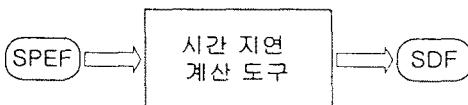


그림 3. 타이밍 분석기의 입출력 구조도

2.2 시스템의 내부 구성

프로그램의 내부흐름은 그림4와 같이 나타내진다. SPEF 파일을 입력으로 받아들여 parsing하는 동안에 두 개의 구조체 그룹을 만들면서 연결선 시간지연 계산을 위한 SPICE 형태의 RLC의 내용이 들어 있는 파일을 만든다. 이렇게 구조화된 연결선과 셀에 대한 정보를 recursive 함수를 사용하여 게이트 시간지연과 연결선의 시간지연을 구하면서 구조체의 시간지연 저장부분에 각 시간지연 값을 저장한다. 최종적으로 각 구조체의 저장된 시간지연 값을 SDF 형태에 맞는 파일형으로 출력하고 종료하는 과정으로 구성되어진다.

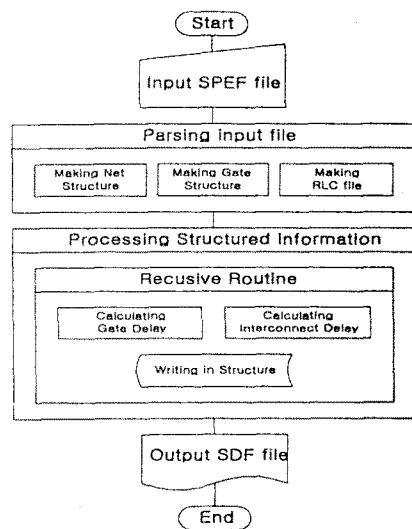


그림 4. 시간 지연 계산 도구의 flowchart

2.3 내부 알고리즘

입력 파일을 읽음으로써 얻어진 각각의 구조체를 가지고 각 연결선의 시간지연과 각 게이트의 시간지연을 계산하기 위해서는 다음과 같이 표2의 단계를 거친다.

- ① 게이트의 출력단에 연결된 연결선의 moment값을 구한다.
- ② 연결선의 moment값으로 연결선의 C_{eff} 값을 구한다
- ③ 구해진 C_{eff} 값과 게이트 입력단의 입력 천이시간을 입력으로 하여 precharacterization된 cell library로부터 게이트 입력단부터 출력단까지의 시간지연과 출력단에서의 천이시간을 얻는다.
- ④ 게이트 출력단의 천이시간을 입력으로 하여 연결선의 시간지연과 연결선 출력단의 천이시간을 얻는다.

표 2. 게이트와 연결선의 시간지연 계산 방법

앞의 4가지 단계를 반복적으로 수행하여 각 게이트와 연결선의 시간지연 값을 얻을 수 있다.

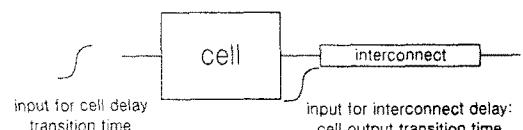


그림 5. 게이트와 연결선의 시간지연 계산 흐름

개발 도구의 주요한 부분인 시간지연과 천이시간의 계산은 다음의 순서로 계산되어진다.

- ① 입력 파일의 시스템에서 입력 Port와 연결된 첫 연결선의 시간지연과 천이시간을 계산한다.
- ② 시스템의 출력단과 연결된 게이트의 핀 값을 입력으로 하는 recursive 함수를 수행한다.
- ③ 최종 출력단과 연결된 게이트와 출력 포트 사이의 시간지연과 천이시간을 계산한다.

표 3. 도구의 주요 시간지연 계산 부분

다음의 표는 앞의 표4에서 설명된 세 가지 단계를 보여주고 있다.

```

Analysis_Delay(FILE *IN, FILE *OUT)
입력포트에 연결된 연결선의 지연 및 천이시간 계산
net=NULL일 때까지 반복수행
IF net->pin!=NULL)
  IF net->pin->direction=='O'
    THEN
      node2=net->s_inst->node;
      node1=net->s_inst->gate->node
      node1이 NULL일때까지 반복수행
      IF node1->inst->direction=='I'
        THEN
          recursive(IN,node1->inst, node2->inst)
          게이트의 지연 및 천이시간 계산
          node1=node1->next
        net=net->next
      출력포트에 연결된 연결선의 지연 및 천이시간 계산

```

표 4. 해석의 흐름

그림6은 recursive한 특성을 갖는 프로그램의 수행을 설명하고 있다. recursive 함수는 게이트 입력단의 시간지연 값이 계산되어 있을 때까지 이전 게이트로 옮겨간다. 그 후에 게이트의 입력단의 시간지연 값이 있으면 return하여 게이트의 시간지연 계산과 연결선의 시간지연 계산을 수행하며 진행된다. 그러나가 recursive 함수를 벗어나면 현재의 게이트 시간지연을 계산하고 다시 다음의 게이트 입력단을 따라 동일한 방법을 반복한다. 그 후에 출력단에 연결된 게이트의 시간지연이 모두 계산되면 출력 Port와 연결된 게이트 사이의 시간지연을 계산하고 종료하게 된다.

그림6의 ①로 표시된 부분이 위의 표3의 ①을 수행하는 부분이고, 그림의 ②는 표3의 ②를, 그림의 ③은 표3의 ③을 수행하는 부분이다.

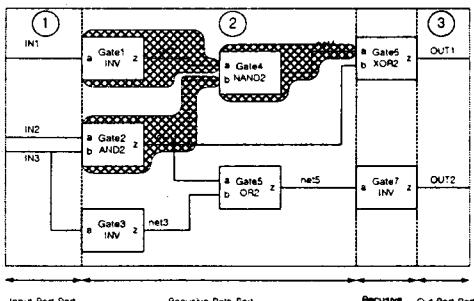


그림 6. Recursive 함수

아래의 표5는 recursive 함수의 알고리즘을 나타낸다.

```

recursive(Inst *first, Inst *second)
IF first->rising_delay가 NULL과 같은가?
  THEN
    IF first->net->s_inst->gate->check가 '0'과 같은가?
      THEN
        first->net->s_inst->gate->check='1'
        node1=first->net->s_inst->gate->node
        node1이 NULL일때까지 다음을 반복 수행한다.
        IF node1->inst->direction이 입력인가?
          THEN
            recursive(node1->inst, first->net->s_inst) 실행
            게이트 지연 계산 함수 실행
            node1=node1->next
        연결선 지연 계산 함수 실행

```

표 5. Recusive Routine

3. 결 론

침의 고속화·고집적화로 인하여 침 설계시에 고려해야 할 요소들이 더욱 증가하게 되었다. 과거에는 중요시 여기지 않았던 부분인 연결선에서의 시간 지연은 이제 다른 부분보다 더 중요한 문제로 부각되었으며, 과거의 논리 검증 완료만으로 동작하곤 하던 침들은 고속·초미세선폭 회로시대에서는 타이밍 분석을 통과하지 않고서는 작동 가능성을 보장받지 못하게 되었다.

본 시간지연 계산 도구는 SPEF 파일 형태를 입력으로 하여 기생 정보를 읽어들여 timing 데이터를 생성하여 규격화된 SDF 파일 형태로 출력파일을 형성하는 도구이다. 그러므로 입·출력 파일의 규격성으로 인하여 여타의 프로그램들과 호환성을 갖고 있으며 연결선의 시간지연 계산에는 AWE 알고리즘을 적용하여 그 속도를 빠르게 하였으며, 게이트의 시간지연 계산에는 C_{eff} 를 찾아보니 정확한 셀의 시간지연과 천이시간을 얻을 수 있도록 구성된 도구이다. 향후 과제로는 물리적레이아웃 정보로부터 SPEF 형태로 변환하는 알고리즘 및 도구에 관한 연구와 개개의 회사마다 다르게 설정되어 있는 timing library에 대한 연동이 중요한 연구 과제가 될 것이다. 이렇게 되면 설계중인 침의 성능을 보다 정확하게 예측하는데 큰 도움이 될 것이다.

(참 고 문 헌)

- [1] SIA(Semiconductor Industry Association). Annual Report, 1994.
- [2] Standard Delay Format Specification Version 3.0. Open Verilog International, May 1995.
- [3] Standard For Integrated Circuit Delay and Power Calculation System. Prepared by the Delay and Power Calculation Working Group of the Design Automation Standards Committee. v0.9.15, pp 239-264, August 22, 1997.
- [4] 김부성, 김성만, 김석윤, "효율적인 타이밍 수준 게이트 지연 계산 알고리즘", 전기학회 98년 추계 학술대회 계재 예정
- [5] L. T. Pillage, R. A. Rohrer, "Asymptotic Waveform Evaluation for Timing Analysis," IEEE Tran. on computer-aided design, vol. 9, No. 4, April, 1990.