

Quad Tree 구조를 이용한 회로 추출기

(A Circuit Extractor Using the Quad Tree Structure)

李建培*, 鄭正和*

(Keon Bae Lee and Jong Wha Chong)

要 約

본 논문에서는 마스크 도면 정보를 나타내는 CIF 입력으로부터 회로도 추출하는 회로 추출기(circuit extractor)를 제안한다.

회로 추출기는 마스크 도면으로부터 트랜지스터와 그들간의 연결상태를 추출하고 기생 저항, 기생 캐패시턴스와 같은 회로 파라미터값을 계산한다.

기생 저항을 계산하기 위해 전류의 흐름 경로를 고려함으로써 저항값 근사화 과정에서 발생하는 오차를 줄였으며, 기생 캐패시턴스를 계산하는데 있어서 회로의 동작 특성에 영향을 미치는 결합 캐패시턴스 값을 고려하였다. 따라서, 이러한 회로 파라미터 값을 회로 시뮬레이션의 입력으로 사용하는 경우 지연시간 등과 같은 회로 특성을 정확히 예측할 수 있다.

마스크 정보인 2 차원 도형 정보를 저장하고 탐색하기 위한 데이터 구조로서는 multiple storage quad tree 개념을 도입하였으며 이 데이터 구조를 이용하여 회로 추출 알고리즘을 구성하였다.

또한, 여러가지 MOS 공정 기술에 따라 회로 추출 알고리즘의 변경없이 적용할 수 있는 기술 독립적인 회로 추출 알고리즘을 제안한다.

Abstract

This paper proposes a circuit extractor which extracts a netlist from the CIF input file containing the layout mask artwork informations.

The circuit extractor extracts transistors and their interconnections, and calculates circuit parameter such as parasitic resistance and parasitic capacitance from the mask informations.

When calculating the parasitic resistance, we consider the current flow path to reduce the errors caused by the resistance approximation. Similarly, we consider the coupling capacitance which has an effect on the circuit characteristics, when the parasitic capacitances are calculated.

Therefore, using these parameter values as an input to circuit simulation, the circuit characteristics such as delay time can be estimated accurately.

The presented circuit extraction algorithm uses a multiple storage quad tree as a data structure for storing and searching the 2-dimensional geometric data of mask artwork.

Also, the proposed algorithm is technologically independent to work across a wide range of MOS technologies without any change in the algorithm.

I. 서 론

*正會員, 漢陽大學校 電子工學科

(Dept. of Elec. Eng., Hanyang Univ.)

接受日字: 1987年 10月 13日

최근 집적회로의 복잡도가 증가함에 따라 레이아웃 설계시 마스크 도면 설계과정에서 설계상의 오류가 발

생활 가능성이 매우 높아지고 있으며 이러한 설계상의 오류를 칩 제작 공정 단계 이전에 추출하여 수정하기 위한 설계 검증 시스템의 개발이 필수 불가결하게 되었다.

실제 검증 방법으로는 회로 연결 상태와 같은 구조적인 검증을 위한 회로 연결도 검증과 회로 동작 특성을 예측하기 위한 회로 시뮬레이션 등이 있으나 이 검증 방법은 직접 마스크 도면으로부터 검증을 행할 수 없으므로 마스크 도면으로부터 검증 시스템의 입력 형태에 맞는 회로도들 추출하는 과정이 필요하게 되었다.

회로 추출(circuit extraction) 과정은 마스크 도면 정보를 나타내는 CIF¹¹⁾로부터 검증시스템의 입력으로 사용하기 위한 회로도를 추출하는 과정으로서, 트랜지스터와 그들의 연결도를 추출하는 Netlist 추출 단계와 기생 저항, 기생 캐패시턴스와 같은 회로 파라미터 값을 계산하는 회로 파라미터 추출 단계의 2단계로 나눌 수 있다.

회로 파라미터 추출 단계에 있어서 정확한 회로 파라미터 값을 계산하기 위하여는 많은 시간이 소요되므로 대부분의 디지털 시스템에서는 회로의 크기가 커짐에 따라 파라미터 값의 정확도 보다 빠른 계산 시간이 더 중요하게 되었다. 회로 파라미터 값의 계산 방법에는 여러 방법¹²⁻¹⁴⁾이 있으나 회로 파라미터 값 계산의 정확도와 계산 시간을 고려하는 파라미터 근사화 방법이 널리 이용되고 있다.

또한, 회로 추출과정은 마스크 도면 정보인 2차원 도형 정보를 이용하여 회로를 추출하므로 도형 정보를 가능한 한 적은 메모리를 사용하여 저장하면서 빠른 시간 내에 필요한 정보를 탐색할 수 있는 데이터 구조가 필요하다.

List 데이터 구조는 그 구조가 간단하나 정보탐색이 느리며, Corner stitching¹⁵⁾ 구조는 정보 탐색은 빠르나 list 구조보다 3 배 정도의 메모리가 소요되는 단점이 있다. Tree 구조중 multiple storage quad tree 구조는 기존의 list 데이터 구조보다 메모리는 더 소요되지만 point 탐색의 경우, $O(\log n)$ 의 빠른 탐색 능력을 갖는 장점이 있다.

본 논문에서는 빠른 탐색 능력을 갖는 multiple storage quad tree 데이터 구조¹⁶⁾를 이용한 새로운 회로 추출기를 제안한다.

주변 정보 탐색을 많이 필요로 하는 회로 추출 알고리즘에 multiple storage quad tree 데이터 구조를 사용함으로써 list 데이터 구조를 사용한 회로추출기^{17,18)}보다 빠른 회로 추출 알고리즘을 설계할 수 있다.

회로 파라미터 값 계산에 있어 branch 노드의 경우에 발생하는 근사화된 저항값의 오차를 줄이기 위하여

전류의 경로를 고려하였으며 전체 캐패시턴스 중 큰 비중을 차지하는 결합 캐패시턴스 값도 계산함으로써, 이러한 회로 파라미터 값을 회로 시뮬레이션의 입력으로 사용하는 경우 정확한 회로의 동작 특성을 예측할 수 있다. 또한, 기술 독립적(technology independent) 기법¹⁹⁾을 도입함으로써 회로 추출 알고리즘의 변경없이 다양한 공정 기술에 적용 가능하게 한다.

II. 회로 추출기의 데이터 구조

회로 추출기에서 취급하는 마스크 도면 정보는 마스크 층과 좌표값을 갖는 2 차원 평면 도형이므로 이 정보를 저장하고 빠른 시간 내에 탐색할 수 있는 데이터 구조를 사용해야만 그 회로 추출 시간을 단축할 수 있다.

Tree 데이터 구조²⁰⁾는 빠른 탐색 능력으로 인해 주변 정보 탐색 동작을 많이 필요로 하는 회로 추출기의 데이터 구조에 적합하다.

본 논문의 회로 추출기에서 취급하는 마스크 도면 정보는 수직, 수평 성분을 갖는 직사각형 도형으로 국한하며 마스크 정보를 저장하기 위한 내부 데이터 구조로서 multiple storage quad tree를 사용하였다.

Multiple storage quad tree는 기존 bisector quad tree^{21,22)}에 있어 필요한 정보를 탐색하기 위해 XY 분할선에 교차한 모든 정보를 탐색하는 것과는 달리 탐색 영역 내의 정보에 대해서만 탐색하는 장점으로 인해 빠른 데이터 탐색 능력을 가지며 메모리 소요량 면에서도 가장 간단한 데이터 구조인 linked list 구조보다 20~30% 정도 더 소요되는 데이터 구조이다.²³⁾

이 데이터 구조는 데이터의 빠른 탐색을 하는데 있어 같은 정보를 중복 저장하는 단점은 있으나 도형의 전 정보를 중복 저장하는 것이 아니라 단지 도형을 가리키는 포인터 값으로 중복 저장하므로 실제적인 메모리 사용면에서 과다한 메모리의 사용은 피할 수 있다.

특히 레이아웃 마스크 정보는 각각의 도형 정보의 크기가 전체에 비해 작고 균일하게 분포되어 있는 특징을 가지므로 이러한 정보를 저장하는 데이터 구조로서 적합하며 회로 추출 알고리즘에서 많이 사용하는 region 탐색과 point 탐색에 적합한 데이터 구조이다. 특히, point 탐색의 경우 linked list의 $O(n)$ 에 비해 $O(\log n)$ 의 빠른 정보 탐색을 행할 수 있다.

레이아웃 마스크 정보인 2 차원 도형 정보는 그림 1과 같고 이를 multiple storage quad tree로 표현하면 그림 2와 같다. Multiple storage quad tree는 tree-node와 leaf-node로 구성되어 있다. Tree-node는 4개의 sub-quad로 분할된 quad를 나타내므로 sub-quad에 관한 정보를 저장하고 있으며 leaf-node는 더 이상

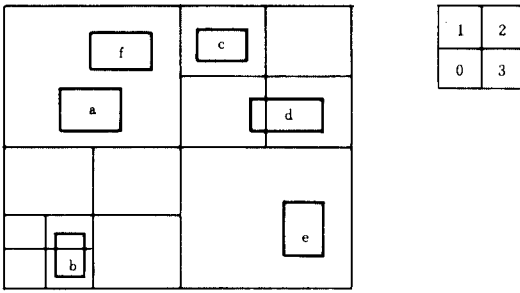


그림 1. 레이아웃 마스크 도면 정보
Fig. 1. A information of layout mask.

표 1. 마스크 평면
Table 1. A mask plane.

Active 평면	Interconnect 평면
polysilicon	metal
n-diffusion	contact
p-diffusion	
p-well (n-well)	
contact	

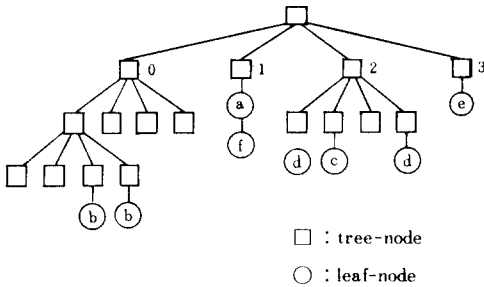


그림 2. Multiple storage quad tree 표현
Fig. 2. A representation of multiple storage quad tree.

분할되지 않는 quad를 나타내므로 이 quad 내에 존재하는 도형 정보를 가리키는 포인터를 저장하고 있다.

III. 회로 추출

1. 마스크 평면과 회로 표현

마스크 도면은 여러 개의 마스크 층(layer)으로 구성되어 있으므로 이 마스크층을 2 개의 마스크 평면에 나누어 저장한다. 마스크 평면은 트랜지스터와 같은 능동 소자를 형성하는 마스크 층을 저장하는 active 평면과 이들 소자간의 연결에 사용되는 마스크 층을 저장하는 interconnect 평면으로 나눌 수 있다.

이는 전기적으로 상호 영향을 미치는 층들을 같은 평면에 위치시킴으로써 서로 다른 층에 존재하면서 교차하는 도형간의 면적을 탐색하는 경우, 필요한 평면한 평면에서의 탐색으로 국한되므로 불필요한 탐색을 방지할 수 있다.

각 평면에 저장하는 마스크 층은 CMOS 공정의 경우 표 1 과 같다.

본 논문의 회로 추출기는 노드 모델¹¹⁾을 이용하여 회로를 표현한다. 노드는 트랜지스터간 또는 트랜지스터

와 연결된 영역중 등전위를 형성하는 영역으로 나타내며, 하나의 노드는 노드 이름, 기생 저항값, 그리고 접지 캐패시턴스 값을 갖는다.

RC network으로 회로를 표현하는 경우, 회로를 정확히 기술할 수 있는 장점이 있으나 회로가 대형화됨에 따라 그 복잡도로 인하여 과다한 회로 해석 시간이 요구된다. 노드 모델은 하나의 노드가 하나의 저항값과 캐패시턴스값을 가지므로 RC network과는 달리 회로를 간단하게 표현할 수 있다. 따라서 회로가 복잡하고 대형화됨에 따라 여러 CAD 시스템^{4,11)}의 회로 모델로 이용되고 있다.

2. 트랜지스터 추출

트랜지스터 추출은 회로 추출의 초기 단계로서, active 평면내의 polysilicon과 diffusion의 두 층이 서로 교차하여 발생하는 트랜지스터를 검출하여 그 정보를 저장하는 단계이다.

트랜지스터를 추출하기 위해 마스크 내의 polysilicon과 diffusion이 교차하는 부분을 탐색하여 교차하는 부분을 채널로 정의해야 하는데 이는 모든 diffusion 도형을 window로 설정하여 region 탐색에 의해 diffusion과 교차하는 polysilicon 도형을 탐색한다.

Region 탐색 알고리즘은 다음과 같다.

```

Region-Search (tree-node, window)
{
  for (quad 0; quad 3; ++ )
    if (window와 quad i가 교차) {
      if (quad i sub-divide)
        Region-Search (quad i, window);
      else {
        for (quad i 내의 모든 object)
          Relation (window, object);
      }
    }
} /*Region-Search*/
    
```

Relation (window, object) 동작은 window와 object의 각각의 2 대각선 좌표로부터 2 도형간의 인접 (abut), 교차 (overlap), 포함 (include) 등의 관계를 계산하여

window와 관련된 도형을 출력한다.

트랜지스터를 추출하기 위해 탐색된 2 층의 교차 부분에 대해 diffusion 영역을 그림 3 과 같이 세 부분으로 분할한 후 채널을 구성하는 부분 (C)과 source 또는 drain (S/D)을 구성하는 부분으로 나누어 Insert와 Delete 동작에 의해 tree 데이터 구조를 수정한다.

트랜지스터가 추출되면 트랜지스터에 이름을 부여하고 채널의 타입이나 채널의 폭과 길이와 같은 물리적인 정보와 위치 정보를 저장한다.

3. 회로 연결도 추출

트랜지스터가 추출되면 트랜지스터간의 연결 상태를 추출하기 위해 트랜지스터 (채널)를 제외한 마스크 도형에 노드 번호를 부여한다. 전기적으로 연결이 있는 마스크 도형에 같은 노드 번호를 부여하고 같은 노드 번호가 부여된 도형을 등전위로 하여 각 트랜지스터의 터미널에 연결된 노드를 찾는다. 따라서, 각각의 트랜지스터간의 연결 상태를 구함으로써 netlist를 구성한다.

각 도형에 노드 번호를 부여하는 Node-Find 알고리즘은 다음과 같다.

```

Node-Find(object)
{
  object를 window로 설정;
  Region-search(tree-node, window);
  for(window와 관련된 obj들에 대해)
  {
    if(obj == channel)
      Node-Find(obj);
    else {
      if(window와 obj가 전기적으로 연결)
        if(번호가 부여되지 않았으면) {
          같은 노드 번호 부여;
          Node-Find(obj);
        }
      else
        if(번호가 부여되지 않았으면) {
          다른 노드 번호 부여;
          Node-Find(obj);
        }
    }
  }
}
/*Node-find*/

```

모든 도형에 노드 번호가 부여되면 같은 노드 번호를 갖는 영역을 등전위로 하여 이 노드 정보와 입력 정보로부터 얻은 입·출력 단자와 power, 접지 등의 정보로부터 트랜지스터의 각 터미널에 연결된 노드 정보를 얻는다. 추출된 회로 연결도는 트랜지스터와 노드 (I/O 노드와 내부 노드)로 구성된다.

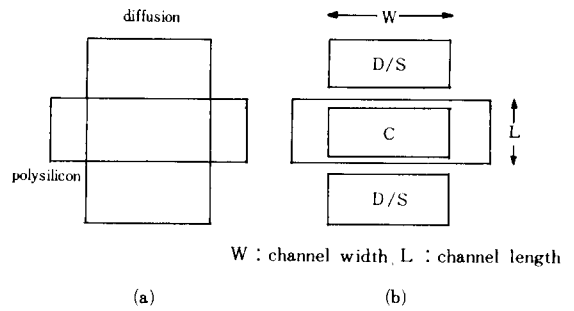


그림 3. 트랜지스터 검출과 diffusion 영역의 분할 Fig. 3. Transistor detection and a partition of diffusion area.

4. 회로 파라미터 값 계산

칩 공정 단계에서 생성되는 기생 저항 및 기생 캐패시턴스는 회로의 동작 특성을 결정짓는 중요한 요인이 되므로 회로 파라미터 값을 계산함으로써 회로 동작 특성을 예측할 수 있다.

(1) 기생 저항(parasitic resistance) 값 계산

기생 저항은 노드 모델의 잇점을 이용하여 노드를 구성하는 도형의 총 면적과 총 주변 길이를 계산한 후 이 값을 이용하여 직사각형 근사화 방법에 의해 구한다. 직사각형 근사화 방법의 빠른 계산 시간의 잇점이 있으나 branch 노드의 경우 실제의 저항값보다 큰 값으로 근사화되는 단점이 있다. 즉, 그림 4와 같이 (a)와 (b)의 두 노드는 같은 면적과 주변 길이를 가지므로 (c)와 같이 같은 저항값으로 근사화되나, 실제적으로 (b)와 같은 branch 노드가 더 작은 값을 갖게 된다.

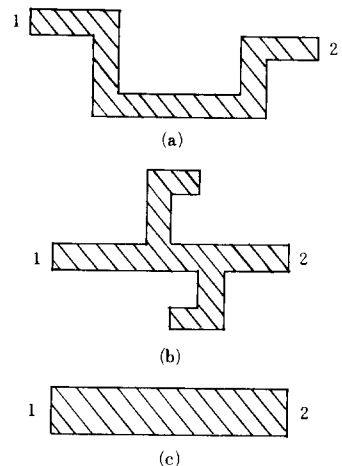


그림 4. 저항값의 근사화 Fig. 4. Approximating resistance.

본 논문에서는 직사각형 근사화의 장점을 이용하면서 전류의 흐름 경로를 고려함으로써 branch 노드의 경우에 발생하는 근사화의 오차를 줄일 수 있다.

저항값 계산 방법은 다음과 같다.

1) Maze 알고리즘을 이용하여 노드 상의 2 터미널 간의 최단 경로를 찾는다.

2) 경로 상에 존재하는 도형들로만 노드의 면적(A')과 주변길이(P')를 구한다.

3) 이 면적과 주변 길이를 이용하여 직사각형 근사화에 의해 저항값을 계산한다.

그림 5 (a)와 같이 두 터미널 1, 2 간의 최단 경로를 Maze 알고리즘을 이용하여 구하면 두 터미널 1, 2 간의 전류의 경로상에 존재하는 도형은 빗금 친 도형과 같다(그림 5 (b)). 이 도형들의 총 면적(A')과 총 주변길이(P')를 이용하여 직사각형 근사화한 결과는 그림 5 (c)와 같다.

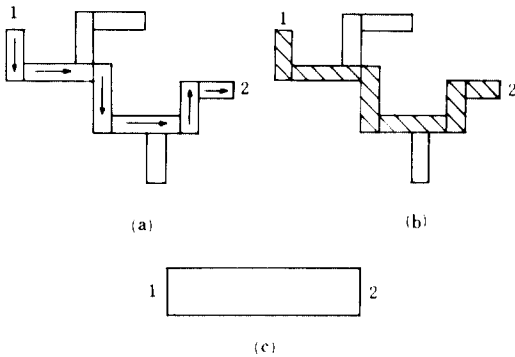


그림 5. 직사각형 근사화에 의한 저항값 계산

- (a) 전류의 경로 탐색
- (b) 경로 상에 존재하는 도형
- (c) 직사각형 근사화

Fig. 5. A calculation of resistance value using rectangular approximation.

- (a) Current path search.
- (b) Rectangles on current path.
- (c) Rectangular approximation.

(2) 기생 캐패시턴스(parasitic capacitance)값 계산
MOS 회로에 있어 기생 캐패시턴스는 다음과 같이 고려한다. 각 층과 substrate 간에 존재하는 접지 캐패시턴스(ground capacitance)와 인접하는 층간의 전기적인 간섭에 의해 발생하는 결합 캐패시턴스(coupling capacitance)의 2 개로 크게 나눈다. 최근 집적도가 급속히 향상됨에 따라 대부분의 MOS 회로에서 결합 캐패시턴스가 전체의 상당 부분을 차지하므로 결

합 캐패시턴스를 고려함으로써 동작 특성을 정확하게 예측할 수 있다.

1) 접지 캐패시턴스(Cg)

접지 캐패시턴스는 substrate에 존재하며 노드를 구성하는 마스크 층의 총 면적과 총 주변 길이에 비례하는 캐패시턴스 값을 갖는다.

$$C_g = C_a * A + C_p * P$$

- Ca : 단위 면적당의 캐패시턴스
- Cp : 단위 길이당의 캐패시턴스
- A : 노드의 총 면적
- P : 노드의 총 주변 길이

2) 결합 캐패시턴스

결합 캐패시턴스는 그림 6 과 같이 크게 3 가지로 구성된다.

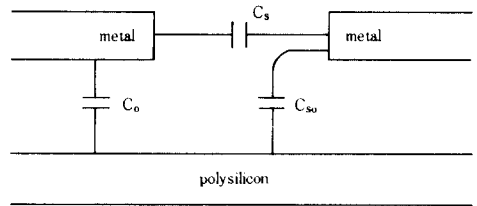


그림 6. 결합 캐패시턴스
Fig. 6. Coupling capacitance.

가. Overlap 캐패시턴스(Co)

서로 다른 층의 도형이 overlap 하는 경우 2 층간에 발생하며 overlap 한 면적에 비례하는 캐패시턴스 값을 갖는다.

$$C_o = C_{ij} * A$$

- Cij : i층과 j층간의 단위 면적당 캐패시턴스
- A : overlap한 면적

나. Sidewall 캐패시턴스(Cs)

서로 다른 노드에 속해 있는 도형이 나란히 지날때 2 노드 간에 발생하며 나란히 지나는 도형의 주변 길이에 비례하고 거리(간격)에 반비례하는 캐패시턴스 값을 갖는다.

Sidewall 캐패시턴스를 계산하기 위해 이웃하는 도형들을 점검하는 경우 간격이 임계값(T) 보다 큰 경우에는 실제적으로 상호간에 작용하는 영향력이 적으므로 T이내의 주변에 대하여만 sidewall 캐패시턴스를 고려한다(그림 7).

$$C_s = C_i * L / D$$

- Ci : i층의 단위 길이당의 캐패시턴스
- L : 평행한 주변 길이
- D : 주변 길이간의 간격

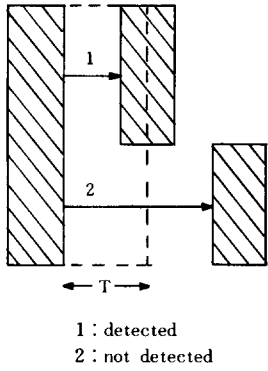


그림 7. 평행한 주변길이 간의 sidewall 캐패시턴스
Fig. 7. Parallel edge sidewall capacitance.

다. Sidewall-Overlap 캐패시턴스 (Cso)

한 층에 속하는 도형의 주변 길이와 다른 층의 도형과 overlap 하는 경우 두 층간에 발생하며 overlap 하는 주변 길이에 비례하는 캐패시턴스 값을 갖는다.

$$C_{so} = C_{ij} * L$$

Cij : i층과 j층간의 단위 길이당의 캐패시턴스

5. 기술 독립성 (technology independence)

반도체 공정 기술이 지속적으로 발전함에 따라 회로 파라미터와 설계 규칙이 변경되고 새로운 공정 기술이 개발되고 있으며 이러한 여러 공정 기술에 적용하기 위하여 기술 독립적인 회로 추출기의 설계가 필요하다.

기술 독립적 기법은 공정 기술에 관한 정보를 추출 알고리즘에 포함시키지 않고 technology file로부터 회로 추출에 필요한 정보를 이용함으로써 회로 추출 알고리즘의 변경없이 여러 공정 기술에 사용 가능하게 한다.

Technology file에는 특정 공정 기술에 대한 능동소자와 회로 파라미터 값에 관한 정보, 마스크층 간의 전기적인 연결 상태 등의 정보가 저장되어 있다.

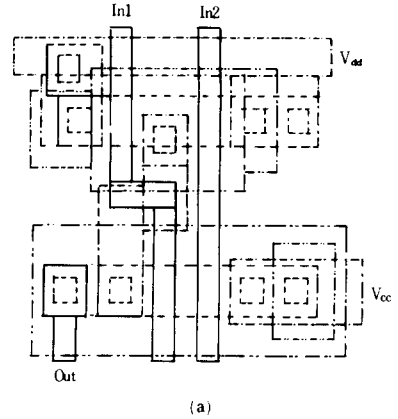
6. 출 력

회로 추출 단계에서 얻은 트랜지스터와 그들의 연결 상태, 회로 파라미터 값의 정보는 설계 검증을 위한 검증 tool의 종류에 따라 검증시스템의 입력 형태에 알맞는 netlist로 출력된다.

회로의 구조적 검증을 위한 연결도 검증 시스템의 입력으로는 트랜지스터와 그들의 연결 상태 정보만이 필요하나 회로의 동작 특성을 예측하기 위한 회로 또는 타이밍 시뮬레이션의 입력으로 netlist를 출력할 경우에는 연결도 정보와 계산된 회로 파라미터 값이 포함된다.

IV. 예 제

그림 8 (a)와 같은 CMOS p-well 공정, 2-입력 NAND 게이트의 마스크 도면을 나타내는 CIF file (그림 8 (b))을 입력으로 하여 회로 추출기에 의해 추출한 Netlist는 그림 9 와 같다.



DS;
LCP; B 8 14 19, 7; B 16 16 19, 22; B 8 48 55, 24; B 24 8 47, 52;
B 8 48 39, 80; B 8 104 71, 52; LCM; B 116 12 58, 95; B 22 22 21, 78;
B 32 22 95, 78; B 16 36 55, 59; B 16 24 39, 42; B 36 16 29, 22; B 48 20
103, 22; LCD; B 20 16 22, 91; B 96 16 64, 75; B 16 6 55, 64; B 80 16
71, 22; LCC; B 8 8 19, 22; B 8 8 39, 22; B 8 8 87, 22; B 8 8
103, 22; B 8 8 24, 75; B 8 8 55, 69; B 8 8 88, 75; B 8 8 104,
75; B 8 8 20, 91; LCW; B 114 40 64, 22; LCS; B 22 24 17, 72; B
56 38 56, 72; B 12 32 90, 75; B 22 30 106, 22;
LCP; In1 55, 0; In2 71, 0; Out 19, 0; LCM; Vdd 116, 95; Vss 127, 22;
DF;

그림 8. CMOS 2-입력 NAND의 예
(a) 마스크 도면 (b) CIF 입력
Fig. 8. An example of CMOS 2-input NAND.
(a) Layout mask. (b) CIF input.

```
CELL : NAND 2
fet1 pfet 8 4 Vdd In1 Vdd Out
fet2 pfet 8 4 Vdd In2 Vdd Out
fet3 nfet 8 4 Vss In2 #2 Vss
fet4 nfet 8 4 Vss In1 Out #2
node Vdd 50 69
node Vss 9 31
node In1 78 43
node In2 65 41
node #2 13 5
node Out 15 50
ccap In1 In2 4
```

그림 9. 회로 추출기에 의해 추출된 netlist
Fig. 9. Extracted netlist by circuit extractor.

Netlist는 트랜지스터(nfet, pfet)와 노드(node), 노드간의 결합 캐패시턴스(ccap)로 구성된다.

트랜지스터는 이름, 타입, 크기, 각 터미널과 연결된 노드 이름을 갖는다. 또한, 노드는 노드 이름과 기생 저항값, 기생 캐패시턴스 값을 갖는다.

V. 결 론

레이아웃 마스크 도면을 나타내는 CIF 정보로부터 검증 tool의 입력 형태에 맞는 회로도룰 추출해 내는 새로운 회로 추출기를 제안하였다.

수직, 수평성분 만을 갖는 직사각형 도형을 처리하는 본 논문의 회로 추출기는 내부 데이터 구조로 multiple storage quad tree를 사용하여 tree의 빠른 탐색 능력을 이용함으로써 효율적인 회로 추출 알고리즘의 수행을 가능하게 하였다.

저항값 계산에 있어 전류의 경로를 고려함으로써 branch 노드의 경우에 발생하는 근사화의 오차를 줄였으며 전체의 캐패시턴스 중 큰 비중을 차지하는 결합 캐패시턴스를 고려함으로써 정확한 회로의 동작 특성을 예측하게 하였다. 또한, 기술 독립적 기법을 도입함으로써 회로 추출 알고리즘의 변경없이 여러 공정기술에 사용할 수 있다.

본 논문의 회로 추출 알고리즘은 C언어로 프로그램되어 SSM-16의 UNIX O/S 하에서 실현하였다.

앞으로의 연구 과제로는 계층 구조에 적합한 회로추출기와 설계, 검증, 수정의 설계 사이클의 단축을 위한 incremental 회로 추출기의 개발에 관한 연구가 될 것이다.

參 考 文 獻

- [1] C. Mead, L. Conway, "Introduction to VLSI system," Addison Wesley, 1980.
- [2] R.L. Brown, "Multiple storage quad trees: a simpler faster alternative to bisector list quad trees," *IEEE Trans. on CAD* Vol. CAD-5, No. 3, July 1986.
- [3] M. Horowitz, R.W. Dutton, "Resistance extraction from mask layout data" *IEEE Trans. on CAD* vol. CAD-2, No. 3, July 1983.
- [4] H. Yoshimura, "An algorithm for resistance calculation from IC mask pattern information," *Proceeding of 1979 ISCAS*, pp. 478-181, 1979.
- [5] S. Mori, "Hierarchical capacitance extraction in an IC artwork verification system" *ICCAD 84*, pp.266-268, 1984.
- [6] S.P. McCormick, "EXCL: a circuit extractor for IC design," *21st DAC*. pp.616-623, 1984.
- [7] A. Goopata, "ACE: A circuit extractor" *20th DAC* pp.721-725, 1983.
- [8] D.T. Fitzpartrick "MEXTRA: A manhattan circuit extractor" *Electronics Res. Lab. Memo M82/42*, Electrons Research Lab. Univ. of California Berkeley, January, 1982.
- [9] A. Boothsaz, R.A. Cotrell, "A technology independent approach to hierarchical capacitance extraction in an IC artwork verification system" *ICCAD 84*, pp. 266-268, 1984.
- [10] Aho, Hopcroft, Ullman, "The design and analysis of computer algorithm," Addison wesley, 1974.
- [11] G. Kedem, "The Quad-CIF Tree: a data structure for hierarchical on-line algorithms," *19th DAC*, pp. 352-357, 1982.
- [12] J.B. Rosenberg, "Geographical data structure compared: a study of data structure supporting region queries" *IEEE Trans. on CAD* vol. CAD-4, No. 1, January 1985.
- [13] W.S. Scott and J.K. Ousterhout, "Magic's circuit extractor" *22nd DAC*. pp. 286-292, 1985.
- [14] J.K. Ousterhout "Crystal: a timing analyzer for nMOS VLSI circuit" *3rd Caltech Conference on VLSI, Pasadena, CA*. 21-23, March, 1983.