

# 다항식 MOS 지연시간 모델을 이용한 스위치레벨 논리 시뮬레이터

## (Switch Level Logic Simulator Using Polynomial MOS Delay Model)

金永鉉\*, 金起\*, 朴松培\*

(Young Hyun Jun, Ki Jun and Song Bai Park)

### 要 約

NMOS와 CMOS 논리 회로에 대한 새로운 방식의 스위치레벨 논리 시뮬레이션이 제안되었다. 간단한 인버터의 상승 또는 하강 지연시간은 입력 파형의 기울기, 부하 커패시턴스(capacitance)와 소자의 크기의 다항식(polynomial)의 곱으로서 근사하였고 다항식의 계수는 주어진 공정에 대하여 SPICE 시뮬레이션 결과에 가장 잘 맞게 결정되었다. 이 방법은 쉽고 정확하게 다입력 전이에 대해서도 적용할 수 있다. 시뮬레이션 결과들은 제안된 방법이 몇가지 회로에 대하여 SPICE 시뮬레이션 결과와 비교하여 5% 이내의 오차로서 지연시간을 예측 할 수 있음을 보여 주었으며 수행속도도 수 천배 빨라졌음을 알 수 있었다.

### Abstract

A new technique is proposed for switch-level logic simulation for NMOS and CMOS logic circuits. For the simple inverter the rise or fall delay time is approximated by a product of polynomials of the input waveform slope, the output loading capacitance and the device configuration ratio, the polynomial coefficients being so determined as to best fit the SPICE simulation results for a given fabrication process. This approach can easily and accurately be extended to the case of multiple input transitions. The simulation results show that proposed method can predict the delay times within 5% error and with a speed up by a factor of three orders of magnitude for several circuits tested, as compared with the SPICE simulation.

### I. 서 론

반도체 기술의 발전으로 MOS 논리 회로의 집적

도는 날로 증가 되므로 MOS 대형회로의 설계검증을 위해서는 시뮬레이션의 중요성이 강조 되고있다.

MOS 대형회로에 적합한 시뮬레이터에는 회로 시뮬레이터<sup>1)</sup>, 시간 시뮬레이터<sup>2,3)</sup>, 스위치 레벨 논리 시뮬레이터<sup>4,5)</sup>, 논리 시뮬레이터<sup>6)</sup> 등이 있다.

SPICE<sup>7)</sup>와 같은 회로 시뮬레이터는 전체회로의 비

\*正會員, 韓國科學技術院 電氣 및 電子工學科

(Dept. of Electrical Eng., KAIST)

接受日字: 1988年 3月 22日

선형 미분 방정식의 정확한 수치해를 구하므로 정확한 해를 보장하지만 회로의 크기가 증가하면 수행속도가 느리고 많은 양의 메모리를 요구한다. 시간 시뮬레이터는 전체회로를 여러개의 부회로로 분할하여 각각의 부회로를 독립적으로 해석하므로 수행시간이 적게 걸리나 강하게 결합된(tightly coupled) 소자나 피이드백 회로에서 수렴 문제와 많은 반복 횟수 때문에 때로는 회로 시뮬레이터 보다 수행시간이 오래 걸리거나 아예 수렴을 못 하는 경우도 있다. 반면 논리 시뮬레이터는 회로의 논리 검증에 사용되며 해석 단위는 논리 게이트 이고 각 게이트마다 지연이 일정한 값으로 정해지므로 부정확한 시간 정보를 조래한다.

스위치 레벨 논리 시뮬레이터는 회로와 논리 시뮬레이터의 중간 수준에 해당한다. 이 시뮬레이터의 장점은 속도면에서 논리 시뮬레이터보다는 느리지만 회로 시뮬레이터 보다는 빠른 수행시간으로 회로 시뮬레이터에 근접한 지연을 계산할 수 있다.

스위치 레벨 논리 시뮬레이터는 회로내의 트랜지스터를 논리적으로 모델링하나 소자의 크기와 부하 캐패시턴스에 의존하는 지연을 계산한다. Nham 과 Bose 에 의해 개발된 시뮬레이터<sup>6)</sup>가 여기에 속하는데, 이 시뮬레이터는 트랜지스터들을 on-off 상태로 모델링하고 여러형태의 MOS 트랜지스터의 크기(configuration) 들은 각각 표준화된 크기(standard configuration) 으로 변화시킨다음 각각의 표준화된 크기에 대해서 여러가지의 입력, 출력 캐패시턴스에 따라 지연이 회로 시뮬레이터 또는 직접측정에 의해 계산되어 표에 저장된다. 표준화된 크기가 아닌 경우에는 표준화된 크기에 대응하는 크기 상수(scale factor)가 계산된다. 이 때 출력 캐패시턴스는 크기 상수로 나누어서 증가 캐패시턴스를 구한다.

스위치 레벨 논리 시뮬레이터중에서도 진보된 ED MOS 전용의 MOSTIM<sup>7)</sup> 시뮬레이터는 정확한 지연계산을 위해 3 가지 상태의 논리레벨을 가지며 입력선이 기울기를 포함하고있다. 이 시뮬레이터는 회로를 다기능 블록(multi-functional block : MFB) 과 패스 트랜지스터 블록(pass transistor block : PTB) 으로 구분하여 해석한다. 이 시뮬레이터의 단점은 첫째 EDMOS 회로만 해석되며 둘째 지연표 계산(table look-up) 방법을 사용하므로 지연계산에 있어서 많은 보간(interpolation) 과 그과정에서 오차가 발생한다. 세째로는 입력 기울기와 출력 캐패시턴스만의 함수로서 표가 만들어져 트랜지스터의 크기가 충분히 고려되지 못한다.

이와같은 MOSTIM 의 단점을 보완하고자 Over-

hauser 가 Shichman-Hodges MOS 모델을 이용하여 물리적인 의미를 갖는 지연 방정식<sup>8)</sup> 을 만들었으나 이 방정식은 CMOS 회로에만 적용되며 사용되는 모델 변수가 4 개로 제한되어 있다. 그래서 많은 변수를 요구하는 short channel MOS 대해서는 적용할 수 없다.

본 논문에서는 CMOS, EDMOS 두가지 경우에 모두 적용되며 long channel 이나 short channel 에서도 사용이 가능한 다항식 MOS 지연 방정식을 제안하며 이 지연 방정식을 스위치 레벨 논리 시뮬레이터인 JKSIM<sup>12)</sup> 에서 구현되었다.

## II. 스위치레벨 논리 시뮬레이션

### 1. 스위치레벨 논리 시뮬레이션

스위치레벨 논리 시뮬레이션의 목적은 정확도 면에서 회로 시뮬레이션에 근접한 시간 정보를 얻으면서 수행시간에 있어서는 논리 시뮬레이션에 근접한 빠르기를 얻기 위함이다. 일반적으로 스위치레벨 논리 시뮬레이터들은 대부분 구조가 비슷하지만 지연시간의 정의와 계산에 있어서 제각기 다르다. 본 논문의 시뮬레이터에서도 일반적인 방식을 그대로 따르나 지연시간의 계산에 있어서 서로운 방법을 제안하였다.

### 2. 3 상태 파형

스위치레벨 시뮬레이터를 효율적으로 정확하게 하기 위하여는 기울기를 포함하는 3 논리 상태를 갖는 파형이 적합하다. 논리 상태들은 0,U,1로서 0은 논리 low에, 1은 논리 high에 대응되고 U는 0에서 1로 또는 1에서 0으로 전이하는 동안의 상태를 나타낸다. 여기서 3 가지 논리에 대응하는 전이의 범위는 다음과 같다.

$$\begin{aligned} 0 &: V(t) \leq V_{TL} \\ 1 &: V(t) \geq V_{TH} \\ U &: V_{TL} < V(t) < V_{TH} \end{aligned}$$

이 경우 4 가지 형태의 전이가 생기는데 0에서 1, U에서 1, 1에서 U, U에서 0이다. 이 4 가지 상태는 각각 1,2,3,4의 형태로서 표현한다. 만약 시간 t에서 0에서 U로 전이가 일어난다면 (1,t)로서 파형이 기술된다.

파형의 전이 순서는 다음과 같이 나타내어진다.

$$S = P(w_i, t_i)$$

여기서  $w_i$ 는 전이 형태이며  $t_i$ 는 전이가 일어나는 시간이다. 그림 1은 3 상태 논리 파형을 보여주고 있다.

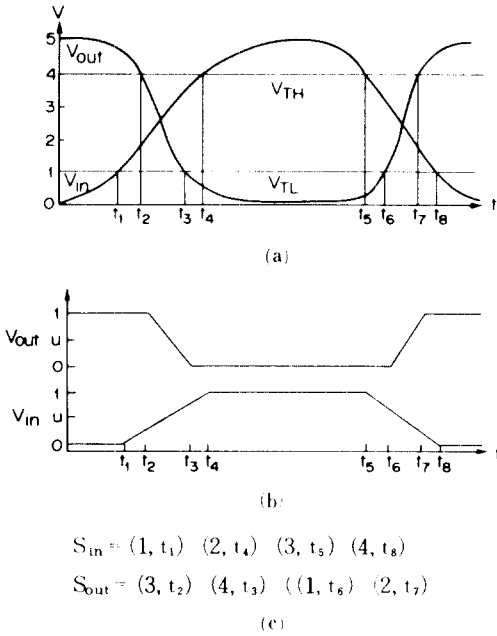


그림 1. 상태 논리파형

- (a) 아날로그 파형
- (b) 논리 파형
- (c) 파형의 표현

Fig. 1. State logic waveform.

- (a) analog waveform.
- (b) logic waveform.
- (c) representation of waveform.

3. 3-상태 파형 이완 방법.

각각의 부회로들은 정하여진 순서대로 해석되는데 부회로마다 전체 해석구간 0에서 T까지의 입력에 대해서 출력을 계산하는 것이다. 이 방식을 파형이완<sup>2</sup> 방식이라 한다. 그러나 0에서 T까지 모든 시간에서 해석하면 메모리 이용이나 시간면에서 비효율적이다. 스위치레벨 논리 시뮬레이션은 입력이 변화하지 않는 부분에서는 출력변화가 일어나지 않으므로 입력 변화가 일어나는 시간만 저장하고 해석하면 된다. 따라서 파형은 그림1(c)와 같이 파형을 천이 순서로서 나타낸다.

임의의 부회로를 해석하려면 먼저 해석구간 들을 만들어서 linked-list 형태로 저장한다. 이 구간들은 입력 천이가 일어나는 구간들이다. 그림 2는 인버터의 입력과 해석구간 list 를 나타내고 있다. 한편 2-입력 NAND 게이트의 해석구간은 2-입력간의 다중 천이(multi-transition)나 겹침 천이(overlap-transition)가 일어나면 2 개의 입력 천이가 포함되도록 해석구간을 조정하여야 한다.

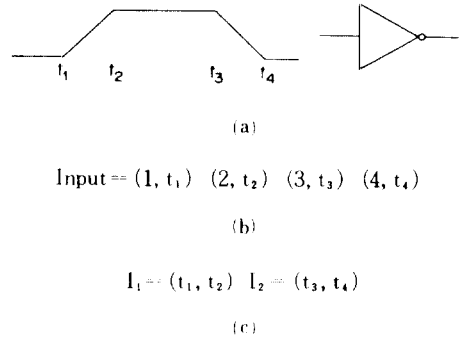


그림 2. 인버터에 대한 파형이완 방법

- (a) 인버터 회로
- (b) linked list로 표현된 인버터의 입력
- (c) 인버터의 해석구간

Fig. 2. Waveform relaxation in analysis of inverter.

- (a) inverter circuit.
- (b) input of inverter represented by linked list.
- (c) analysis interval of inverter.

스윗치레벨 논리 시뮬레이션에서 파형이완 방법의 예를들면 2단 인버터 chain 해석의 경우 제 1단 인버터를 전체 입력파형에 대하여 출력파형을 구한다음 출력파형의 천이 시간들을 메모리에 저장시킨다. 이때 제 1단 출력은 제 2단의 입력이 된다. 위의 방법을 제 2단에 적용시킨다.

4. Dynamic windowing

피이드백 회로(SCC)의 해석은 피이드백으로 인해서 특수한 처리가 요구된다. Dynamic windowing<sup>8</sup> 방법은 스위치레벨 시뮬레이션에서 피이드백 루우프로 형성된 SCC 회로를 해석할때 중복 해석을 피하여 해석시간을 줄이는 방법이다. SCC 회로를 해석할때 SCC 회로의 해석 상태를 고려하여 SCC 내의 각 부회로들은 각 부회로들의 입력이 천이 할때만 해석된다. 따라서 SCC의 해석구간 목록(list)은 SCC 내의 부회로들의 입력이 천이 하는 시간들만 으로서 만든다. 이때 해석구간 목록중의 한 해석구간을 검지(window)라고 부른다. 그림 3은 SCC회로를 보여주며 각 노우드에서의 천이 순서와 해석구간 들을 나타내고 있다. 그림 3에서 해석구간 I<sub>1</sub>는 다음과 같은 정보를 갖고 있다.

- 1) 해석구간의 시작 시간.
- 2) 해석구간의 끝 시간.
- 3) 해석구간 안에서 입력이 천이하는 부회로들의 목록.

그리고 두개의 해석구간이 겹칠때는 그림3 (c) 와 같이 하나의 해석구간으로 만든다. 다음 프로시듀어 (procedure) 는 SCC 회로 해석의 순서이다.

프로시듀어 SCC 해석 :

SCC에 있는 모든 입력 j에 대하여

begin

입력 j에 대한 구간의 순서 목록 L<sub>j</sub> 를 구성한다 ;

end

구간의 순서 목록 L<sub>j</sub> 를 전체 구간 목록 L에 넣는다 ;

L이 비어있지 않다면

begin

L으로부터 처음 구간 I<sub>k</sub>를 제거한다 ;

I<sub>k</sub>동안 변하는 각 부회로 b<sub>i</sub>에 대하여

begin

I<sub>k</sub>구간 동안 b<sub>i</sub>를 해석한다 ;

b<sub>i</sub>의 출력 m에 대하여

begin

출력m에 의하여 생겨나는 구간 L<sub>m</sub>의 목록을 구성한다 ;

만일 L<sub>m</sub>의 구간과 I<sub>k</sub>의 구간과 겹치면

begin

그 부회로에 대하여 반복이 필요하다 ;

end

end

L<sub>m</sub>을 L과 합쳐서 새로운 구간 목록 L을 만든다 ;

end

만일 부회로들이 반복이 필요하다면

begin

이러한 부회로에 대하여 반복을 한다 ;

end

end

### III. 지연시간

#### 1. 지연시간의 정의

MOS 인버터의 입력, 출력 파형에 관계되는 변수 들은 다음과 같이 정의 된다.

고(저) 문턱 전압 전압 V<sub>TH</sub> (V<sub>TL</sub>) :

전압원 V<sub>DD</sub>의 80% (20%)

입력 상승(하강) 시간 T<sub>ILH</sub> (T<sub>ILL</sub>) :

상승(하강) 하는 입력이 V<sub>TL</sub> (V<sub>TH</sub>)에 도달하는 시간 과 V<sub>TH</sub> (V<sub>TL</sub>)에 도달하는 시간사이의 간격.

출력 상승(하강) 시간 T<sub>OLH</sub> (T<sub>OLL</sub>) :

상승(하강) 하는 출력이 V<sub>TL</sub> (V<sub>TH</sub>)에 도달하는 시간과 V<sub>TH</sub> (V<sub>TL</sub>)에 도달하는 시간사이의 간격.

출력 상승(하강) 오프셋(offset) 시간 T<sub>LHO</sub> (T<sub>LLO</sub>) :

하강(상승) 하는 입력이 V<sub>TH</sub> (V<sub>TL</sub>)에 도달하는 시간

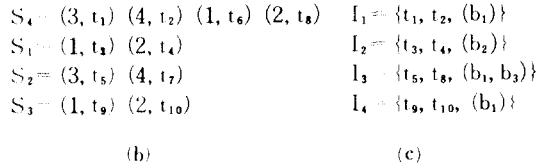
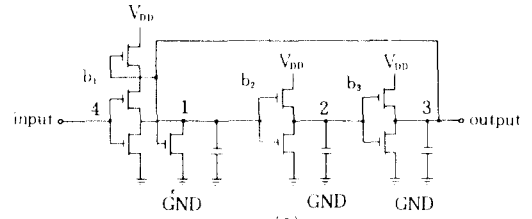


그림 3. SCC 회로의 해석

(a) SCC 회로  
 (b) 각 노우드에서의 파형  
 (c) 해석구간

Fig. 3. Analysis of SCC circuit.

(a) SCC circuit.  
 (b) waveform at each node.  
 (c) analysis interval.

에서 상승(하강) 하는 출력이 V<sub>TL</sub> (V<sub>TH</sub>)에 도달하는 시간 간격.

위에서 정의한 T<sub>ILH</sub>, T<sub>ILL</sub>, T<sub>LHO</sub>, T<sub>LLO</sub>의 4 가지 부분 지연시간은 그림 4에서 보여준다.

전체 출력 지연시간은 상승지연시간의 경우 T<sub>r</sub> = T<sub>LHO</sub> + T<sub>ILH</sub>로 하강지연시간의 경우 T<sub>f</sub> = T<sub>LLO</sub> + T<sub>ILL</sub>로 정의 된다. 위의 4 가지 부분 지연시간은 부회로의 입력 기울기 τ, 출력 캐패시턴스 C<sub>out</sub>, 소자 크기의 비 β에 대한 함수가 된다. 여기서 β는 CMOS 회로에 있어서는 상승 지연의 경우 PMOS의 소자 크기이고, 하강 지연의 경우 NMOS의 소자 크기이며,

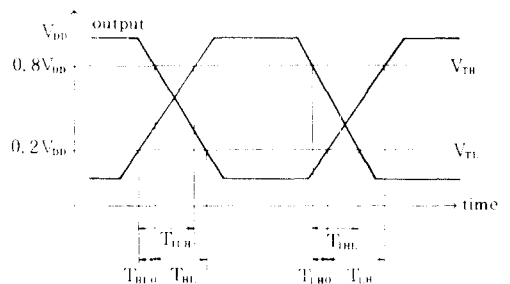


그림 4. 지연시간의 정의

Fig. 4. Definitions of delay time.

EDMOS 회로에 있어서는 풀업(pullup) 트랜지스터와 풀다운(pulldown) 트랜지스터의 소자 크기의 비가 된다.

2. 지연시간의 수식화

지연시간함수의 형태를 알아보기 위하여 상승지연 시간  $T_{LH}$ 를 일정한  $\beta$ 에 대하여 여러가지의  $C_{out}$ 과  $\tau$ 에 대하여 SPICE 시뮬레이션한 결과가 그림 5의 실선과 같다. 그림 5에서 볼 수 있듯이 모든 곡선은  $\tau$ 에 대한 1차 함수로서 표현된다. 그러므로  $T_{LH}$ 는 다음과 같은 1차 함수가 된다.

$$T_{LH} = f(\tau, C_{out}, \beta) = a_1 + a_2 \tau \quad (1)$$

이때  $a_1$ 과  $a_2$ 는  $C_{out}$ 과  $\beta$ 의 함수이다.  $a_1$ 는 수직축과의 교점이 되고  $a_2$ 는 직선의 기울기를 나타낸다. 직선들과 수직축과의 교점이  $C_{out}$ 에 대하여 등간격으로 벌어져 있으므로  $C_{out}$ 에 대한 1차 함수가 된다.

$$a_1 = b_1(\beta) + b_2(\beta) C_{out} \quad (2)$$

또한 직선의 기울기는  $C_{out}$ 에 대하여 조금씩 변하므로  $C_{out}$ 에 대한 2차 함수로 가정한다.

$$a_2 = b_3(\beta) + b_4(\beta) C_{out} + b_5(\beta) C_{out}^2 \quad (3)$$

각각의 계수  $b_i$ 는 그림 6에서 나타나듯이  $\beta$ 에 대하여 2차 함수로 근사시킬 수 있다.

$$b_i(\beta) = c_{i1} + c_{i2} \cdot \beta + c_{i3} \cdot \beta^2 \quad (i=1, 2, \dots, 5) \quad (4)$$

이때  $c_{ij}$  ( $i=1, 2, \dots, 5$ ;  $j=1, 2, 3, 1$ )은 계수(coefficient)이다. 식(1)에 식(2), (3), (4)를 대입하면 다음과 같은  $T_{LH}$ 의 식을 얻을 수 있다.

$$T_{LH}(\tau, C_{out}, \beta) = \{ (C_{11} + C_{12} \cdot \beta + C_{13} \cdot \beta^2) + (C_{21} + C_{22} \cdot \beta + C_{23} \cdot \beta^2) \cdot C_{out} \} + \{ (C_{31} + C_{32} \cdot \beta + C_{33} \cdot \beta^2) + (C_{41} + C_{42} \cdot \beta + C_{43} \cdot \beta^2) \cdot C_{out} + (C_{51} + C_{52} \cdot \beta + C_{53} \cdot \beta^2) \cdot C_{out}^2 \} \cdot \tau \quad (5)$$

$C_{ij}$ 의 값은 식(5)으로부터 계산된  $T_{LH}$ 와 SPICE에서 얻어진 결과( $T_{LH}^*$ )와의 RMS(root mean square) 오차  $\epsilon$ 가 최소가 되도록 최소 자승 방법을 사용하여 최적화 시킨다.

$$\epsilon = \sum \frac{(T_{LH} - T_{LH}^*)^2}{T_{LH}^{*2}}$$

식(5)의 15개의  $C_{ij}$ 가 결정되면 이들은 메모리에 저장되어 주어진  $\tau, C_{out}, \beta$ 에 대하여 보간(interpolation) 없이 지연시간을 계산하는데 사용된다.

다른 지연시간  $T_{HL}, T_{LH0}, T_{HL0}$  등도 위와 같은 방

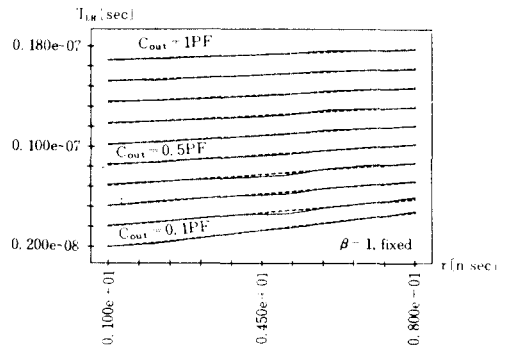


그림 5.  $\beta=1$ 일때  $\tau$ 와  $C_{out}$ 에 대한 EDMOS 인버터의  $T_{LH}$ 의 변화

Fig. 5. Variations of  $T_{LH}$  of EDMOS inverter with  $\tau$  and  $C_{out}$  at  $\beta=1$ .

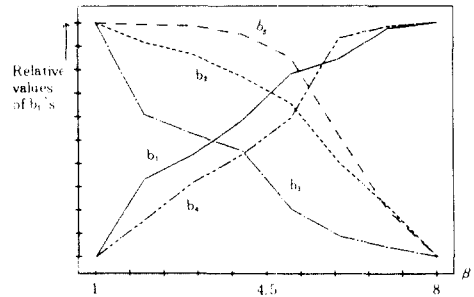


그림 6.  $\beta$ 에 대한  $b_i$  ( $i=1, 2, \dots, 5$ )의 변화

Fig. 6. Variations of  $b_i$ 's ( $i=1, 2, \dots, 5$ ) with  $\beta$ .

법으로 수식화 할 수 있다. 본 논문에서 사용한 3가지 변수  $\tau, C_{out}, \beta$ 값의 범위는 EDMOS의 경우에는 각각 1nsec-8nsec, 0.1PF-1PF, 1-8이고 CMOS의 경우에는 0.01nsec-8nsec, 0.01PF-1PF, 1-8이다.

3. 다입력 게이트의 해석

1) 유효(effective)  $\beta$ 의 계산

다입력 게이트는 MOS를 저항으로 생각하여 등가적인 인버터로서 만들 수 있다. MOS의 컨덕턴스(conductance)는 채널(channel) 폭(width)  $W$ 와 채널 길이(length)  $L$ 과의 비  $W/L$ 에 비례한다. 따라서 두 개의 트랜지스터가 병렬로 연결되어 있으면 여기에 등가한 트랜지스터의  $W/L$ 은 두 트랜지스터의  $W/L$ 을 더한 것과 같다. 반면, 직렬로 연결되어 있으면 등가한 트랜지스터의  $W/L$ 은 두 트랜지스터로  $L/W$ 를 더한 것의 역수와 같다. 이와같은 원리를 이용하여

다입력 게이트는 인버터와 같이 하나의 부하(load) 트랜지스터와 구동(driver) 트랜지스터로 바뀐다. 트랜지스터가 직렬로 또는 병렬로 연결 되었을때 유효한 입력과 W/L은 다음과 같은 방법으로 구하여진다.

병렬 연결 :  $(W/L)_{eff} = (W/L)_1 + (W/L)_2$

직렬 연결 :  $(W/L)_{eff} = \frac{(W/L)_1 \cdot (W/L)_2}{(W/L)_1 + (W/L)_2}$

CMOS 다입력 게이트의 경우 출력이 상승할 경우에는 PMOS 만을 감축(reduction) 하고 출력이 하강할 경우에는 NMOS 만을 감축한다. 그러나 EDMOS의 경우에는 상승, 하강 지연이 모두 풀업 트랜지스터와 풀다운 트랜지스터의 크기의 비  $(W/L)_{pd}/(W/L)_{pu}$  에 의존한다. 즉 이와 같은 유효  $\beta$ 의 값을 간략히 나타내면 다음과 같다.

$$\beta_{eff} = \begin{cases} \frac{(W/L)_{eff,driver}}{(W/L)_{eff,load}} & \text{(NMOS 경우)} \\ (W/L)_{eff,load} & \text{(CMOS, 상승 출력)} \\ (W/L)_{eff,driver} & \text{(CMOS, 하강 출력)} \end{cases}$$

2) 유효 입력의 결정

트랜지스터들이 병렬로 연결 되었을때, 종전의 방법에서는<sup>5,6,9)</sup> 감축된 하나의 등가 트랜지스터에 선택되는 입력은 다음과 같다. 입력 상승(하강)의 경우에는, 다중 입력 전이가 일어난 입력 중에서  $V_{TH}$  ( $V_{TH}$ )에 제일 먼저 도달한 입력을 effective 입력으로서 선택한다. 이러한 입력 선택은 지연 계산에 있어서 오차를 생기게 한다.

본 논문에서는 이러한 오차를 줄이기 위하여 새로운 방법을 제안한다. 설명의 편의를 위하여 CMOS 회로를 생각한다. 그림7(a)와 (b)는 두개의 다른시간  $t_1$ 과  $t_2$ 에서 상승(하강)하기 시작하고  $t_3$ 과  $t_4$ 에서 끝나는 2개의 파형을 보여준다. 우리는  $\gamma$ 라는 변수를 다음과 같이 정의한다.

$$\gamma = \frac{V_c}{V_{DD}}$$

여기서  $V_c$ 는 뒤지는 입력이 문턱 전압과 만나는 시간에 있어서 앞서는 입력의 상승(하강) 전압이다(그림 7).

만일  $\gamma > 0.5$  이면, 그에 대응하는 유효 입력은 다음의 두개의 시간에 의하여 정해진다.

$$t_s = t_1 + \gamma \cdot (t_2 - t_1)$$

$$t_e = t_3 + \gamma \cdot (t_4 - t_3)$$

만일  $\gamma \leq 0.5$  이면  $\gamma = 0.5$ 로 고정된다.

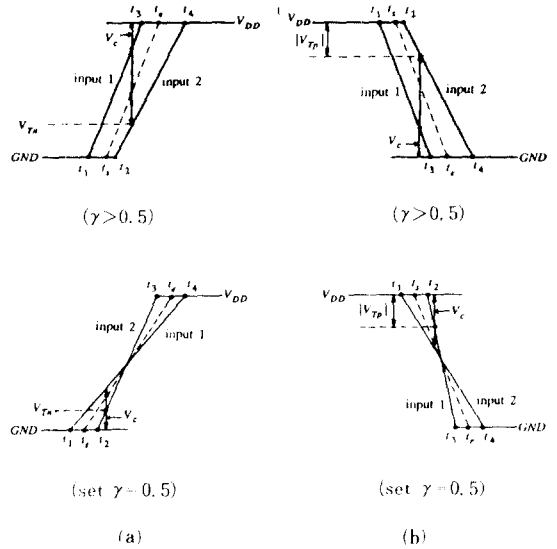


그림 7. 두개의 다른 기울기로서 상승 (a) 하강 (b) 하는 두개의 입력파형.  $V_{TH}$ =NMOS threshold 전압,  $V_{TP}$ =PMOS threshold 전압  
**Fig. 7.** Two input waveforms starting to rise (a) or fall (b) at different instants with different slopes.  $V_{TH}$ =NMOS threshold voltage,  $V_{TP}$ =PMOS threshold voltage.

그림 8은 유효 입력 결정에 있어서 제안된 방법과 종전의 방법을 사용하여 CMOS NOR 게이트를 해석한 결과를 보여준다. 여기서 우리는 제안된 방법이 매우 정확함을 알수있다.

트랜지스터가 직렬로 연결 되었을때는 종전의 방법대로, 하나의 등가 트랜지스터에 선택되는 입력은 다중입력 전이중에서 가장늦게 문턱 전압에 도달하는 입력을 선택한다. 이러한 선택은 만족할만한 정확도를 보여주므로 본 논문에서는 이 방법을 그대로 채택한다.

4. 전송(transmission) 게이트의 지연 시간

1) MOSFET의 지향

MOSFET은 게이트 전압에 의하여 전류가 제어되는 소자로서 저항값은 비선형 특성을 나타낸다. 이와같은 MOSFET은 다음과 같은 3개의 영역에서 동작되며 대응하는 전류방정식은 다음과 같이 나타내어진다(NMOS의 경우).

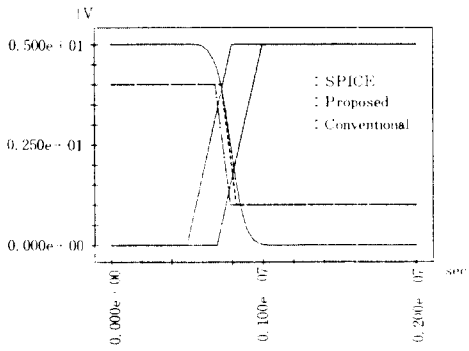


그림 8. 제안된 방법과 충전의 방법에 의한 CMOS NOR gate의 해석결과, 각 NMOS의  $\beta = 16/5$ ,  $C_{out} = 0.2pF$  그리고  $V_{Tn} = 0.67$ 이다  
 Fig. 8. Simulation results of a CMOS NOR gate by the proposed and conventional method.  $\beta = 16/5$ ,  $C_{out} = 0.2pF$  and  $V_{Tn} = 0.67V$ .

차단 영역 ( $V_{KS} < V_T$ )

$$I_{ds} = 0 \quad (6)$$

선형 영역 ( $V_{KS} \geq V_T, 0 \leq V_{ds} < V_{KS} - V_T$ )

$$I_{ds} = \frac{1}{2} \cdot \mu_0 \cdot C_{ox} \cdot \frac{W}{L} \cdot (2(V_{KS} - V_T) - V_{ds}) \cdot V_{ds} \quad (7)$$

포화 영역 ( $V_{KS} \geq V_T, V_{ds} \geq V_{KS} - V_T$ )

$$I_{ds} = \frac{1}{2} \cdot \mu_0 \cdot C_{ox} \cdot \frac{W}{L} \cdot (V_{KS} - V_T)^2 \quad (8)$$

위의 3 개의 동작 영역에서 MOSFET의 드레인-소오스 간의 채널 저항  $R_c$ 은 나타낼수 있는 영역은 선형 영역이 된다.

$$G_c = \lim_{V_{ds} \rightarrow 0} \frac{\partial I_{ds}}{\partial V_{ds}} = \lim_{V_{ds} \rightarrow 0} \mu_0 \cdot C_{ox} \cdot \frac{W}{L} \cdot (V_{KS} - V_T - V_{ds}) \quad (9)$$

$$G_c = \mu_0 \cdot C_{ox} \cdot (V_{KS} - V_T) \cdot \frac{W}{L}$$

$$R_c = k_p \cdot \frac{L}{W} \left( k_p \cdot \frac{1}{\mu_0 \cdot C_{ox} \cdot (V_{KS} - V_T)} \right)$$

이와같이 channel 저항  $R_c$ 는 MOSFET의 크기를 나타내는 채널 폭  $W$ 와 채널 길이  $L$ 의 비로써 나타내어진다. 위의  $R_c$ 에서  $k_p$ 값은 NMOS의 경우는  $k_{pn}$ 이고 PMOS의 경우는  $K_{pp}$ 이다.

2) CMOS 전송 게이트의 지연시간

CMOS 전송 게이트의 지연시간 계산에서는 Elmore

지연시간 개념을 적용하면 지연시간이 계산된다. Elmore 지연수의 적용범위는 선형 회로의 경우에만 적용되므로 MOS의 저항을 선형 저항으로 근사화한다음 적용시켜야만 하므로 약간의 오차가 발생한다.

(1) 동기 상태(synchronous mode)

① 출력 상승의 경우

그림 9에서  $V_{c1}$ 의 값이 입력 전압에 의하여 안정화된 다음 전송 게이트가 도통 되는 상태이므로 부하 커패시턴스  $C_2$ 는 전압원  $V_{DD}$ 에 의하여 충전된다. 지연시간  $T_{LH0}$ 와  $T_{LH}$ 를 구하기 위하여 Elmore 지연수  $\tau_e$ 에 의하여 유도된 유효 병수는 다음과 같다.

$$\tau_e = R_1 \cdot C_1 + (R_1 + R_2) \cdot C_2 \quad (10)$$

$$\text{여기서 } R_1 = \frac{1}{k_{pp}} \cdot \left( \frac{L}{W} \right)_{M_1} \quad (11)$$

$$R_2 = \frac{1}{\frac{1}{k_{pp}} \cdot \left( \frac{L}{W} \right)_{M_3} + \frac{1}{k_{pn}} \cdot \left( \frac{L}{W} \right)_{M_4}} \quad (12)$$

$$R_{eff} = R_1 + R_2 \cdot \left( \frac{L}{W} \right)_{eff} \cdot \frac{1}{k_{pp}} \quad (13)$$

$$\beta_{eff} = \frac{W}{L} \cdot \frac{1}{L_{eff}} = \frac{1}{k_{pp} \cdot R_{eff}} \quad (14)$$

$$C_{eff} = \frac{\tau_e}{R_{eff}} \quad (15)$$

$$\tau_{eff} = \tau \text{ 전송 게이트의 입력} \quad (16)$$

전송 게이트의 상승지연  $T_{LH0}$ 와  $T_{LH}$ 는 위의 3개

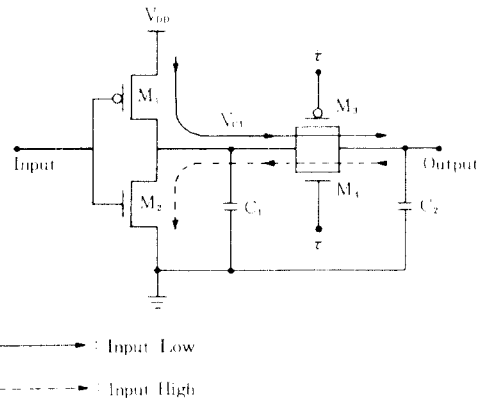


그림 9. CMOS 동기 전송 게이트에서의 전류의 흐름  
 Fig. 9. Current flows in CMOS synchronous transmission gates.

의 유효 변수  $\beta_{eff}$ ,  $C_{eff}$  와  $\tau_{eff}$  를 식 (5) 에 대입하여 구한다.

② 출력 하강의 경우

그림 9 에서  $C_2$  에 충전된 전하는  $M_2$  를 통하여 방전 하게 된다. 지연 시간  $T_{HL}$  와  $T_{HLO}$  를 구하기 위하여 Elmore 시상수  $\tau_e$  를 구한다. 그 결과는, 식(10) 과 (13) 에서  $k_{pp}$  를  $k_{pn}$  으로, 식(11) 에서  $M_1$  을  $M_2$  로 대치한 것이다. 전송 게이트의 상승지연  $T_{HLO}$  와  $T_{HL}$  는 위의 3 개의 유효 변수  $\beta_{eff}$ ,  $C_{eff}$  와  $\tau_{eff}$  를 식(5) 에 대입하여 구한다.

(2) 비동기 상태 (asynchronous mode)

비동기 상태 에서는 전송 게이트의 제어 신호가 안정된 상태에서 입력의 변화에 따라 동작을 하게 된다. CMOS 회로의 지연시간의 계산에서는 모든 트랜지스터를 완전히 도통 상태나 차단 상태로 취급하였으므로 동기 상태의 동작과 비동기 상태의 동작에 있어서는 차이가 없으므로 앞의 동기 상태에서 설명한 방법을 그대로 사용하면 된다. 단지 전송 게이트의 게이트 제어 클럭(clock) 이 입력단에 게이트를 제어하는 클럭으로 대치될 뿐이다. EDMOS 전송 게이트의 경우에도 CMOS 전송 게이트의 경우와 마찬가지로 할 수 있다.

IV. 시뮬레이션

1. 프로그램의 구현

본 논문의 스위치레벨 논리 시뮬레이터는 VAX 11/750 상에서 C 를 사용하여 구현되었다. 전체 프로그램은 입력 전 단(pre-processor) 1000 줄(line)과 주 해석단(main analysis part) 4000 줄로서 나누어졌으며 입력 전 단 부분은 회로의 매크로 확장(macro expansion)과 해석순서를 정하고 해석 부분은 회로의 해석을 한다.

2. 시뮬레이션 결과

본 논문에서 제안한 지연시간 모델의 정확도를 시험하기 위하여 각 단마다 MOS 의 크기와 부하 커패시턴스를 다르게 한 12단 인버터 체인을 시뮬레이션 하였고 그 결과는 그림10에 있다. 그리고 다입력 게이트의 정확도를 시험하기 위하여 CMOS NAND-NOR 게이트(그림11) 를 시뮬레이션 하였다.

그 외에 전송 게이트의 정확성을 보기 위하여 2-phase D latch(그림12) 와 1 bit full adder 회로(그림13) 에 대하여서도 시뮬레이션 하였다.

어러가지 회로에 대해서 시뮬레이션한 결과 정확도에 있어서는 SPICE 결과와의 오차가 7% 이내 에 들고 수행시간은 수 천배 이상 빨라짐을 알수 있

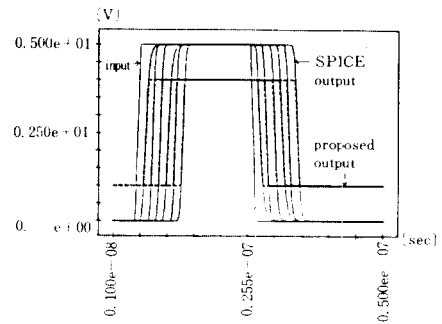
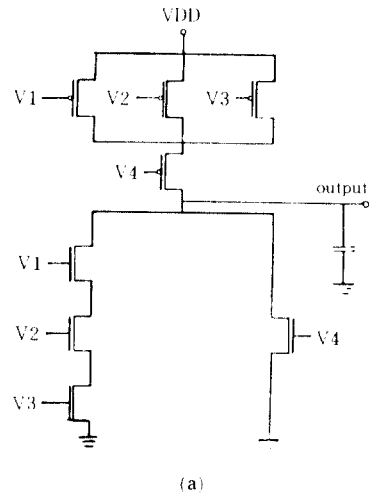
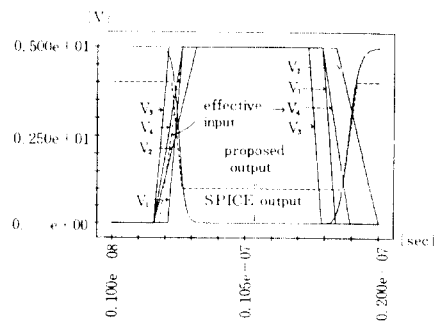


그림10. CMOS 12단 인버터 체인의 시뮬레이션 결과

Fig. 10. Simulation result of CMOS 12 stage inverter chain.



(a)



(b)

그림11. CMOS NAND-NOR 게이트의 시뮬레이션 (a) CMOS NAND-NOR 게이트 (b) 시뮬레이션 결과

Fig. 11. Simulation result of CMOS NAND-NOR gate.

(a) CMOS NAND-NOR gate. (b) result.



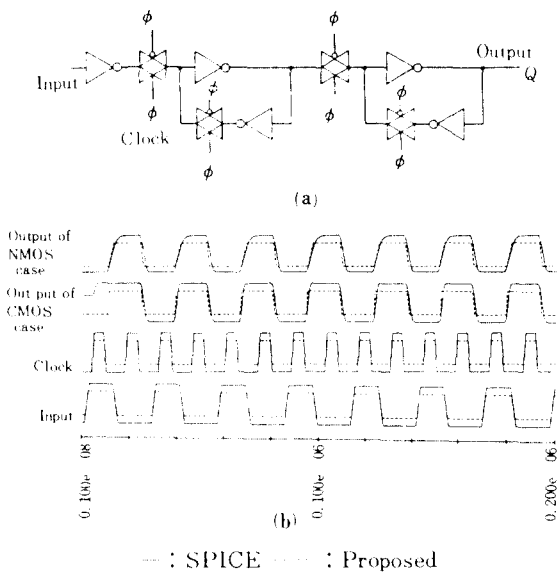


그림 12. 2-phase D latch의 시뮬레이션  
 (a) 2-phase D latch 회로  
 (b) CMOS와 EDMOS 회로의 해석결과  
**Fig. 12.** Simulation results of 2-phase D latch.  
 (a) 2-phase D latch circuit.  
 (b) simulation Result of CMOS and EDMOS circuits.

었다. 각 예제 회로에 대해서 본 논문의 시뮬레이터와 SPICE와의 정확도와 수행시간의 비교를 표 1에 나타내었다.

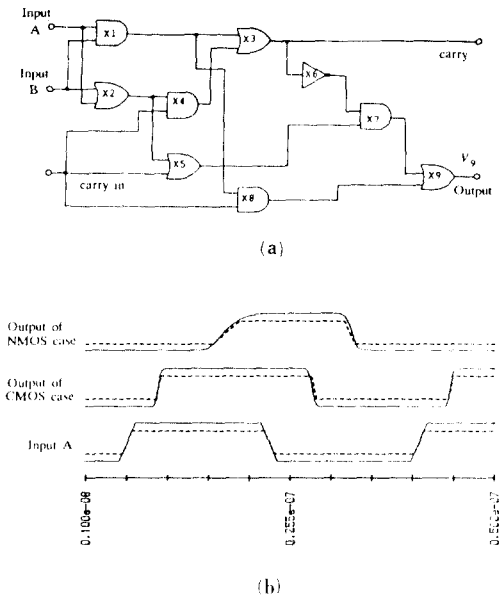
V. 결 론

본 논문에서는 다항식 MOS 지연시간 모델을 이용한 지연시간의 계산이 제안되었다. 제안된 지연시간 모델은 입력 기울기, MOS의 크기 그리고 부하 캐패시턴스의 영향을 모두 고려하였으며 SPICE를 이용해서 주어진 지연시간과 오차가 최소화 되도록 지연방정식을 만들었다. 또한 감속 방법을 사용하여 다압력 게이트를 효과적으로 해석하였으며 전충 게이트 해석에 있어서는 전충 게이트의 지형을 고려해서 효과적으로 감속 하였다. 제안된 모델은 넓은  $\beta$ 와  $C_{out}$  범위에 대하여 적용가능하며 SPICE와 비교하여 정확도가 7% 이내에서 유지됨을 알수 있었다.

새로운 지연시간 모델을 적용한 JKSIM은 VAX 11/750 상에서 C 언어로 구현되었다. 제안된 모델의 정확도가 SPICE와 비교하여 분석되었고 수행 시간이 SPICE에 비하여 수 천배 빠르다는 것이 입증되었다.

표 1. VAX/UNIX-750에서의 CPU 시간과 정확도의 비교  
**Table 1.** Comparisons of CPU time and accuracy on VAX/UNIX-750.

Circuit	# of MOS	Proposed model(sec)	SPICE (sec)	Speed-up factor	Error from SPICE
CMOS complex gate	8	0.2	61.2	306	< 5%
EDMOS NOR gate	3	0.183	33.37	182	< 5%
CMOS 12 inverter chain	24	0.35	379.1	1083	< 3%
EDMOS 12 inverter chain	24	0.334	453.6	1358	< 3%
CMOS 1 bit full adder	50	0.317	557.2	1757	< 5%
EDMOS 1 bit full adder	42	0.52	562.3	1081	< 5%
CMOS latch	14	4.2	1071	255	< 3%
CMOS 8 bit counter	208	20.2	106537.4	5274	< 7%
CMOS ALU	512	13	21870	1682	< 7%
CMOS CPU	782	33	559453	16953	< 5%



--- : SPICE    - - - - : Proposed

그림 13. 1 bit full adder 회로의 시뮬레이션  
 (a) 1 bit full adder 회로  
 (b) CMOS와 EDMOS 회로의 해석결과

Fig. 13. Simulation result of 1bit full adder circuit.  
 (a) 1 bit full adder circuit.  
 (b) simulation result of CMOS and EDMOS circuits.

参 考 文 献

[1] L.W. Nagel, "SPICE2: A computer program to simulate semiconductor circuits," Electronics Research Laboratory, Memorandum no. ERL-M520, University of California, Berkeley, May 1975.

[2] E. Lelarsmee and A. Sangiovanni-Vincentelli, "RELAX: A new circuit simulator for large scale MOS integrated circuits," *Proceedings of the ACM IEEE 19th Design Automation Conference*, pp. 682-690, June 1982.

[3] Y.H. Jun, C.W. Lee, K.J. Lee, and S.B. Park, "Timing simulator by waveform

relaxation considering feedback effect," *Proceedings of the IEEE International Symposium of Circuit and System*, May 1987.

[4] T.W. Kweon, "Gate and Functional Level logic simulation with 8-state signal model," M.S. Thesis, KAIST, 1983.

[5] H.N. Nham and Bose, "A multiple delay simulator for MOS LSI circuits," *Proceeding of the ACM IEEE 17th design automation conference*, pp. 610-617, June 1980.

[6] V.B. Rao, T.N. Trick, and I.N. Hajj, "A table-driven delay-operator approach to timing simulation of MOS VLSI circuit," *Proceedings of the 1983 IEEE International Conference on Computer Design*.

[7] V.B. Rao, "Switch-Level timing simulation of MOS circuits," Ph.D. Thesis, University of Illinois, 1985.

[8] V.B. Rao and T.N. Trick, "A new approach to processing strongly connected circuit blocks in a waveform relaxations switch-level timing simulator," *Proceedings of the 1984 IEEE international conference on circuits and computers*, pp. 502-506, Oct. 1984.

[9] D.V. Overhauser, "A new approach to switch-level timing simulation of CMOS VLSI circuits," M.S. Thesis, University of Illinois, 1985.

[10] W.C. Elmore, "The transient response of damped linear networks with particular regard to wideband amplifiers," *Journal of applied physics*, vol. 19, pp. 55-62, Jan. 1948.

[11] J.L. Wyatt Jr, C. Zukowski, L.A. Glasser, P. Bassett, and P. Penfield Jr, "The waveform bounding approach to timing analysis of digital MOS IC's," *Proceedings of the IEEE international conference on computer-aided design*, pp. 392-395, Nov. 1983.

[12] K. Jun, "Switch level logic simulator using polynomial MOS delay model," M.S. Thesis, Korea Advanced Institute of Science and Technology, 1988.