

Timing Analysis by Concurrent Event Propagation

韓 昌 昊*

(Chang Ho Han)

Abstract - This paper proposes concurrent event propagation technique for timing analysis. The technique makes it possible to find several input vectors and sensitizable paths at the same time. The concurrent event propagation technique is based on the event driven simulation and the timing analysis technique with boolean equations. The technique propagates as many events as possible at the same time while preventing propagation of boolean terms which do not sensitize paths. Since events do not propagate through false paths, the longest path which successfully propagates events to one of the primary outputs is one of the longest sensitizable paths. The technique can speed up timing analysis by unifying path sensitization and maximum delay calculation.

Key Words : timing analysis, path sensitization, false path, critical path, delay

1. 서 론

칩 하나에 수백만개의 트랜지스터를 포함하는 마이크로프로세서나 메모리를 대량생산할 수 있는 반도체 기술은 설계 자동화 기술의 발전에 크게 도움을 받았다고 할 수 있다. 대규모 집적회로의 설계를 검증하고 시험하는데 필요한 대표적인 도구들은 설계검증 도구, 타이밍분석 도구, 시험패턴 생성 도구등 여러 가지가 있다. 디지털 회로의 설계시에 회로의 최대 지연시간(maximum delay)이 주어진 값을 초과하지 않는지를 확인하는 과정을 타이밍 검증(timing verification)이라고 한다. 정확한 타이밍 검증을 위해서는 시뮬레이션 기법으로 모든 가능한 입력에 대해 회로의 출력 및 지연시간을 구해보면 되겠지만 이는 검증 시간이 회로의 주입력(primary input)의 수에 따라 지수적으로 증가하므로 회로의 주입력의 수가 많아지면 현실적으로 실현이 불가능해진다. 따라서 보다 효율적으로 회로의 최대 지연시간을 분석하는 방법이 필요한데 이러한 방법들을 타이밍 분석(timing analysis)이라고 한다

타이밍 분석은 완성된 회로가 주어진 클럭 속도에서 적절하게 동작할 수 있는지의 여부를 검증하는 중요한 과정이며 최근의 시스템들이 매우 빠른 클럭 속도를 요구함에 따라 더욱 그 중요성이 커지고 있다. 타이밍 분석은 신호의 전달이 가능한, 즉 활성화(sensitization)가 가능한 가장 긴 경로를 구하는 것으로 먼저 가장 긴 경로를 구하여 해당 경로의 활성화 여부를 검증하는 방식으로 이루어지는 것이 보통이다[1,2]. 즉, 먼저 주입력(primary input)부터 주출력(primary

output)까지의 최장경로(longest path)를 구하고 그 경로상의 모든 게이트에서 활성화에 필요한 경로 제한조건(path constraint)들을 모두 만족하는지의 여부를 확인하는 방법이 흔히 사용된다. 이러한 방법들에는 먼저 선택된 경로가 거짓 경로(false path)로 확인되면 동일한 과정을 다시 반복하게 되며 거짓 경로가 많은 경우 상당한 수행시간이 소요된다.

경로의 활성화를 검증하는 방법에는 가장 긴 경로부터 하나의 경로씩 순차적으로 활성화 여부를 확인하는 방법[1,2]과 가능한 한 빨리 거짓 경로(false path)를 찾아냄으로써 활성화가능 경로를 찾는 시간을 줄이는 방법[3]등이 사용된다. 활성화 경로를 찾는 방법으로는 ATPG(Automatic Test Pattern Generation) 알고리즘을 이용하는 방법[1,4], BDD(Binary Decision Diagram)나 부울대수를 활용하는 방법[2,5,6,7]등으로 나눌 수도 있다. ATPG는 완성된 회로의 테스트에 필요한 검사패턴을 생성하는 방법인데 결합의 활성화 과정이 경로의 활성화 과정과 유사하므로 타이밍 분석에 직간접으로 응용이 가능하다. ATPG를 응용하는 방법은 다양한 지연시간 모델과 경로 활성화 모델을 수용할 수 있고 큰 회로에도 적용이 가능하지만 분석시간은 다소 긴 편이다.

BDD는 타이밍 분석 외에도 여러 가지 CAD(Computer - Aided Design) 알고리즘에 사용될 수 있는 방법인데 일단 BDD를 구성하면 효율적인 분석이 가능하지만 회로에 따라서는 메모리 사용량이 과다하여 BDD로 표현할 수 없는 경우가 있다[2,8,9].

지연시간 부울대수(Timed Boolean Algebra :TBA)[5,6,7]에 기반을 둔 방법은 부울대수에 시간 정보를 첨가하여 연산할 수 있도록 하였다. 이렇게 하여 회로의 지연시간과 연관된 입력 조합이 경로를 통하여 함께 전달되도록 하며 활성화 경로를 찾는 문제도 부울 연산으로 손쉽게 해결할 수

*正 會 員 : 仁河大 電子計算工學科 副教授 · 工博
 接受日字 : 1999年 8月 6日
 最終完了 : 1999年 8月 30日

있다. 각각의 출력에 대한 가능한 입력조합과 연관된 지연 시간을 동시에 구하므로 효율적이지만 회로의 크기에 따라 지연시간 부울항의 수가 기하 급수적으로 증가하므로 임계치를 두어 항의 수가 어느 이상 불어나지 못하도록 해야 한다. 예를 들어 ISCAS85 benchmark 회로[10]의 대부분에서 부울항의 수를 제한하지 않으면 메모리 부족으로 동작되지 않는다. 부울항을 제한하면 일부 거짓 경로가 활성화 경로인 것으로 판정될 가능성이 있다. 타이밍 분석시간의 단축을 위하여 계층적인 타이밍 분석 방법도 사용되지만 정확성은 다소 떨어질 수 있다[11,12].

경로활성화에 대한 정의는 문헌에 따라 다소 다르지만 어떤 주입력에서 주어진 신호의 변화 즉, 사건(event)이 해당 경로를 따라서 전달되는 것을 나타낸다[1,2,3,5,13].

본 논문에서 제안하는 병렬 사건전파 방식은 지연시간 부울대수[5,6,7]와 유사하게 부울연산을 이용하지만 지연시간 부울대수에서는 모든 가능한 항들을 연산하여 나중에 활성화 여부를 조사하는 반면, 병렬 사건전파 방식은 신호의 변화에 해당하는 항들만 경로에 따라 전달하기 때문에 불필요한 거짓경로에 해당되는 항들은 전달되지 않는다. 따라서 항들이 급격히 증가하는 것을 방지할 수 있어 정확하고 효율적인 분석이 가능하다.

2. 병렬 사건전파

본 절에서는 본 논문에서 제안하는 병렬 사건전파(concurrent event propagation) 방식의 개요와 사용되는 게이트 모델에 대하여 설명한다.

정의1] 정값, 부값 : 회로의 게이트들을 연결하는 신호선(net)을 V 라고 할 때 V 는 정값 v 와 부값 \bar{v} 를 가지며 $V(v, \bar{v})$ 로 표시한다. 정(부)값 $v(\bar{v})$ 는 신호선 V 가 1(0)값을 갖도록 하는 주입력(primary input)의 조합을 나타내는 부울식이다.



그림 1 정값, 부값의 표시방법
Fig. 1 Notation of positive value and negative value

예를 들어 그림 1과 같이 AND 게이트가 주입력 A, B 에 연결되어 있다면 신호선 F 의 정값은 $f=ab$, 부값은 $\bar{f}=\bar{a}+\bar{b}$ 이며 $F(f, \bar{f})=F(ab, \bar{a}+\bar{b})$ 로 표시될 수 있다.

정의2] 사건 : 어떤 시간에 신호의 값이 0에서 1로 또는 1에서 0으로 변화되도록 하는 입력 조합의 값이 도착하는 것을 사건(event)이라고 한다. 사건에 의해 신호선은 정값(0에서 1로 변할 때) 또는 부값(1에서 0으로 변할 때)을 갖게 된다. 정값을 갖게 하는 사건을 정사건, 부값을 갖게 하는 사건을 부사건이라 한다. 신호선 V 에 정사건과 부사건이 도착한 시간을 각각 t_+, t_- 라고 하면 신호선 V 와 연관된 사건 E_v 는 $E_v=(E_{v+}, E_{v-})=(t_+, t_-; \bar{v})$ 로 표시한다.

어떤 게이트의 입력에 사건이 도착하면 게이트의 사건전파 모델에 의하여 출력사건이 구해지고 출력사건은 입력사

건의 도착시간(현재시간)에 해당 게이트의 지연시간을 더한 시간에 스케줄된다. 스케줄된 사건은 해당시간이 되어 출력 신호선의 값을 변경시키고 해당 신호선과 연결된 게이트에 입력사건을 제공한다. 사건은 주입력(primary input)에서 발생하여 게이트와 신호선들을 통하여 주출력(primary output)까지 전파되는데 사건은 단지 활성화된 경로를 통해서만 주출력까지 전파되고 거짓경로를 통해서만 주출력까지 전파되지 않으므로, 주출력까지 사건이 전파되는 시간이 회로의 지연시간이 되고 해당 전파경로는 활성화된 경로이다. 최종시간에 전파된 사건들과 관련된 경로는 위험경로(critical path)이며 이때 전파된 정값 또는 부값은 각각 해당 출력을 1 또는 0으로 활성화시키는 입력조합이 된다.

2.1 AND 게이트를 통한 사건의 전파

AND 게이트를 통한 정사건의 전파 : 시간 t_y 에서 주변 입력 Y 의 정값이 y 가 되었고 $t_y < t_x$ 인 시간 t_x 에서 AND 게이트의 현재입력 X 에 사건이 도착하여 정값이 x 가 되었다면 출력신호 Z 에 대하여 시간 $t_z = t_x + d_r$ 에서 $z=xy$ 의 정값을 가지는 사건이 스케줄된다. 단, d_r 은 게이트의 상승 지연시간이다. 만약 $t_y < t_x$ 인 모든 t_y 에 대하여 $y(t_y|t_y < t_x) = \phi$, 즉 t_x 이전에 Y 에 도착한 정사건이 없어 정값이 NULL이라면 X 의 정값 x 는 AND게이트를 통해 전파되지 않는다.

예) 다음의 그림 2에서와 같이 상승 지연시간이 3이고 하강 지연시간이 1인 AND 게이트가 있다면, $t_y=1$ 에서 Y 의 정값 $y=bc$ 이고 $t_x=2$ 에서 X 의 정값 x 가 a 로 변할 때 AND게이트 출력 Z 에는 $t_z=5$ 에 정값 $z=abc$ 가 스케줄된다.

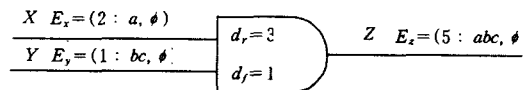


그림 2 AND 게이트를 통한 정사건의 전파
Fig. 2 Propagation of a positive event through an AND gate

AND 게이트를 통한 부사건의 전파 : 시간 t_x 에서 현재 입력 X 의 부값이 \bar{x} 가 되었고 $t_y < t_x$ 인 모든 t_y 에 대하여 주변입력 Y 의 부값이 $\bar{y}(t_y) = \phi$ 이라면, 즉 t_x 이전에 도착한 부사건이 없다면 출력신호 Z 에 대하여 $t_z = t_x + d_f$ 에서 $\bar{z} = \bar{x}$ 의 부값을 가지는 사건이 스케줄된다. 단, d_f 는 게이트의 하강 지연시간이다. 만약 $\bar{y}(t_y|t_y < t_x) \neq \phi$ 이고 $y = y(t_y|t_y < t_x) \neq \phi$ 이면, 즉 t_x 이전에 Y 에 부사건과 정사건이 모두 도착한 적이 있다면, $t_z = t_x + d_f$ 에서 $\bar{z} = \bar{x} \cdot y$ 이다. 그 밖의 경우에는 X 의 부값 \bar{x} 는 AND 게이트를 통해 전파되지 않는다.

예) 다음의 그림 3에서 입력 Y 에 어떠한 사건도 도착한 적이 없다면 $t_x=3$ 에서 도착한 부사건에 의한 X 의 부값 $\bar{x}=bc$ 는 게이트의 하강 지연시간 $d_f=2$ 이후, 즉 $t_z = t_x + d_f = 5$ 에서 출력 Z 의 부값이 $\bar{z}=bc$ 가 되도록 하는 부사건을 스케줄시킨다.

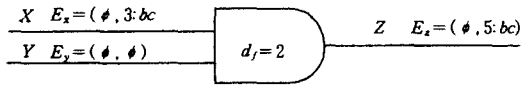


그림 3 AND 게이트를 통한 부사건의 전파
Fig. 3 Propagation of a negative event through and AND gate

2.2 게이트의 사건전파 모델

입력이 X, Y이고 출력이 Z인 AND게이트 또는 NAND 게이트의 현재 입력 X에 부사건이 발생할 때 주변입력 Y의 값에 따른 Z에 스케줄되는 값들을 표1에 요약하였다. 표 2는 입력 X에 정사건이 도착하는 경우이다. 표3(표4)는 OR과 NOR에서 부(정)사건이 입력 X에 도착할 경우를 요약한 것이다.

표 1 입력 X에 부사건이 도착하는 경우 출력에 스케줄되는 사건

Table 1 Output events when a negative event arrives at input X

| 현재입력 X 부값 \bar{x} | 주변입력 Y | | d_f 이후에 스케줄되는 사건 | |
|------------------------|-------------|--------------|--------------------|-------------|
| | 정값 y | 부값 \bar{y} | AND의 부값 \bar{z} | NAND의 정값 z |
| \bar{x} | ϕ 또는 y | ϕ | \bar{x} | \bar{x} |
| \bar{x} | ϕ | \bar{y} | ϕ | ϕ |
| \bar{x} | y | \bar{y} | $\bar{x} y$ | $\bar{x} y$ |

표 2 입력 X에 정사건이 도착하는 경우 출력에 스케줄되는 사건

Table 2 Output events when a positive event arrives at input X

| 현재입력 X 정값 x | 주변입력 Y | | d_f 이후에 스케줄되는 사건 | |
|----------------|--------|---------------------|--------------------|--------------------|
| | 정값 y | 부값 \bar{y} | AND의 정값 z | NAND의 부값 \bar{z} |
| x | ϕ | ϕ 또는 y | ϕ | ϕ |
| x | y | ϕ 또는 \bar{y} | $x y$ | $x y$ |

표 3 입력 X에 부사건이 도착하는 경우 출력에 스케줄되는 사건

Table 3 Output events when a negative event arrives at input X

| 현재입력 X 부값 \bar{x} | 주변입력 Y | | d_f 이후에 스케줄되는 사건 | |
|------------------------|-------------|--------------|--------------------|-------------|
| | 정값 y | 부값 \bar{y} | OR의 부값 \bar{z} | NOR의 정값 z |
| \bar{x} | ϕ 또는 y | ϕ | ϕ | ϕ |
| \bar{x} | ϕ 또는 y | \bar{y} | $\bar{x} y$ | $\bar{x} y$ |

표 4 입력 X에 정사건이 도착하는 경우 출력에 스케줄되는 사건

Table 4 Output events when a positive event arrives at input X

| 현재입력 X 정값 x | 주변입력 Y | | d_f 이후에 스케줄되는 사건 | |
|----------------|--------|--------------|--------------------|-------------------|
| | 정값 y | 부값 \bar{y} | OR의 정값 z | NOR의 부값 \bar{z} |
| x | ϕ | ϕ 또는 y | x | x |
| x | y | ϕ | ϕ | ϕ |
| x | y | \bar{y} | $x y$ | $x y$ |

2.3 사건 전파 과정

그림 4의 회로[5]에 대하여 사건의 전파 과정을 도표화한 것이 표 5이다.

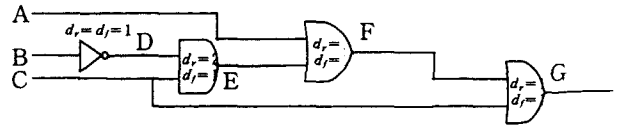


그림 4 예제회로

Fig. 4 An example circuit

표 5 예제 회로에 대한 사건의 전파 과정

Table 5 Event propagation for the example circuit

| t | a | \bar{a} | b | \bar{b} | c | \bar{c} | d | \bar{d} | e | \bar{e} | f | \bar{f} | g | \bar{g} |
|-----------|-----|-----------|------------|-----------|------------|-----------|------------|-----------|-------------|-----------|---|-----------|--------------------|-------------------|
| $-\infty$ | 값 | | | | \bar{b} | | | | | | | | | |
| | 스케줄 | $0a$ | $0\bar{a}$ | $0b$ | $0\bar{b}$ | $0c$ | $0\bar{c}$ | | | | | | | |
| 0 | 값 | \bar{a} | b | \bar{b} | c | \bar{c} | | | | | | | | |
| | 스케줄 | | | | | | $1\bar{b}$ | $1b$ | $1\bar{c}$ | $1a$ | | | $1c$ | |
| 1 | 값 | | | | | | \bar{b} | b | \bar{c} | a | | | | \bar{c} |
| | 스케줄 | | | | | | | | $3\bar{b}c$ | $2bc$ | | | $2\bar{a}\bar{c}$ | $2ac$ |
| 2 | 값 | | | | | | | | | bc | | | $\bar{a}\bar{c}$ | \bar{c} |
| | 스케줄 | | | | | | | | | | | | $3\bar{a}bc$ | $3\bar{c}\bar{a}$ |
| 3 | 값 | | | | | | | | $\bar{b}c$ | | | | $\bar{a}bc$ | \bar{c} |
| | 스케줄 | | | | | | | | | | | | $4\bar{a}\bar{b}c$ | \bar{c} |
| 4 | 값 | | | | | | | | | | | | $\bar{a}bc$ | \bar{c} |
| | 스케줄 | | | | | | | | | | | | | |

먼저 $t = -\infty$ 에서는 모든 신호선의 값들이 정의되지 않고 모든 주변입력들에 대하여 $t=0$ 에서 정값, 부값들을 갖도록 스케줄된다. $t=0$ 에서 주입력들의 값이 주어지면서 이들을 입력으로 하는 게이트들의 값이 계산되어 출력에 사건이 스케줄된다. 예를 들어 입력 A에 연결된 OR 게이트의 출력 F를 보자. $t=0$ 에서 A에 사건이 도착할 때 주변입력 E에는 사건이 도착한 적이 없어 정값 e와 부값 \bar{e} 가 정의되지 않으므로 표 3, 4에서 알 수 있듯이 F의 정값 $f=a$ 가 시간 1에 스케줄되고 부값 \bar{f} 는 정의되지 않는다. 표 5에서 1: a는 시간 1에 a의 값이 스케줄됨을 의미한다. 즉 A의 부값 \bar{a} 에 의한 부사건은 F를 통해 전파되지 않는다. 같은 방식으로 주입력과 연결된 모든 게이트에 대하여 연산을 수행하여 전파되는 사건들을 스케줄한다.

시간 0에서 더 이상의 사건이 없으면 시간을 1로 증가시켜 스케줄된 사건들에 따라 해당 신호값들을 갱신하고 해당 사건들에 의해 입력이 갱신된 게이트들의 출력 사건들을 구한다. 예를 들어 F의 정값 f가 시간 1에서 a가 되었으므로 이에 따라 G의 정값이 변할 수 있다. 표 4에 따라 주변입력 C의 정값이 없으면 g 는 a가 되지만 C의 정값이 $t=0$ 에서 c가 되었으므로 $g=f \cdot \bar{c} = \bar{a}\bar{c}$ 가 되며 상승 지연 시간이 1이므로 시간 $t=2$ 에 스케줄된다. 같은 방법으로 시간 1에 스케줄된 모든 게이트에 대한 연산을 마치면 현재시간을 2로 증가시키고 스케줄된 값들을 갱신한다. 이때 주출

력 G 의 정값이 $g = \overline{ac}$ 가 되므로 $A=1, C=0$ 가 경로 활성화 유발하는 입력이고 경로의 길이는 현재 시간인 2가 된다. 활성화 경로는 사건이 전달되어온 경로인 $A \rightarrow F \rightarrow G$ 가 된다.

같은 방식으로 더 이상의 사건이 발생하지 않을 때까지 반복하면 표 5와 같이 g 에는 더 이상의 사건이 도착하지 않으므로 정값에 대하여 이 경로가 활성화 가능한 최장경로이다. 그러나 부값에 대하여 $\overline{g} = \overline{a} \overline{c}$ 가 시간 3에 도착하므로, 회로의 최대 지연시간은 3이고 정·부값 모두에 대한 활성화 가능한 최장경로, 즉 위험경로(critical path)는 $\overline{a} \overline{c}$ 가 전파되어온 경로인 $C \rightarrow F \rightarrow G$ 이고 위험경로를 활성화하는 입력은 $A=0, C=0$ 이다.

이 결과는 문헌 [5]와 같지만 본 논문에서의 사건의 전파 과정이 [5]에서의 지연시간 부울합의 연산보다 훨씬 간단한 것을 알 수 있다. 또한 [5]에서는 주출력에 도착한 부울항들에 대하여 활성화 여부에 대한 테스트를 수행하여야 하나 제안된 방식에서는 활성화 가능한 부울항들만이 사건 전파 과정을 통하여 전달되므로 추가적인 테스트를 수행하지 않는다. 회로가 더욱 복잡할 경우 [5]의 경우 항의 수가 크게 증가하지만 본 논문에서 제안한 병렬 사건전파 방식에서는 불필요한 사건들이 전파되지 않으므로 보다 효율적인 연산이 가능하다.

3. 시스템의 구조

병렬 사건전파 방식의 타이밍 분석을 위한 시스템의 구조는 일반적인 타이밍 분석 시스템보다는 사건구동형 시뮬레이터[14]와 유사하다. 개념적으로는 가능한 모든 입력에 대하여 동시에 시뮬레이션을 수행하는 것으로도 볼 수 있다.

시스템의 동작을 간략히 기술하면 다음과 같다. 먼저 VHDL 또는 다른 형식으로 표시된 회로도도 전처리과정을 거쳐 DAG(Directed Acyclic Graph)로 변환된다. 게이트 타이밍 모델은 AND, OR, NAND, NOR, NOT, BUFFER 등의 게이트에서의 사건전파 방식을 나타낸다. 타이밍 모델에 따라 상승지연, 하강지연이 같거나 다르게 할 수 있다.

타이밍 분석은 시간 0에서 모든 주입력에 동시에 정사건과 부사건을 스케줄해 줌으로써 시작된다. 회로가 너무 큰 경우에는 동시에 스케줄하는 사건의 수를 다소 줄이고 반복 수행해야 할 경우도 있다. 각 주입력에 도착하는 사건에 의하여 출력이 변화되는, 즉 사건이 발생하는 모든 게이트에 대하여 해당 신호 값을 부울식 형태로 스케줄한다. 타이밍 큐에 스케줄되는 사건들은 해당 게이트의 상승/하강 지연시간을 현재 시간에 더한 시간에 해당 신호값을 변화시킨다.

현재 시간에서 모든 입력에 대한 처리가 끝나면 시간을 증가시키고 해당 시간에 스케줄된 모든 신호값을 변화시킨다. 그리고 신호값들의 변화에 따라 그 신호값들을 입력으로 하는 게이트들의 출력을 계산하고 그 값들을 스케줄하는 과정을 반복하여 모든 사건이 소멸될 때까지 반복하면 마지막으로 주출력에 나타난 신호값이 해당 주출력을 활성화하는 입력이고 그 신호값이 전달되어온 경로가 위험경로(critical path)이며 이때의 현재시간이 회로의 최대 지연시간이 된다.

4. 실험 및 결과

제안된 병렬 사건전파(CEP : Concurrent Event Propagation)기법을 주메모리 128MB의 Spark 1 Workstation에서 C언어로 구현하여 실험하였다. 실험 결과의 비교를 위하여 동일한 시스템에서 지연시간 부울대수를 이용한 프로그램(TBTP : Timed Boolean Term Propagation)도 구현하여 수행결과를 비교하였다.

TBTP에서는 지연시간 부울항의 증가를 제안하는 절삭임계값을 2000으로 설정하였다. TBTP는 모든 회로에 대하여 절삭임계값을 적용하지 않으면 시스템에서 사용가능한 약 120MB의 메모리를 모두 사용하고 메모리 부족으로 실험결과를 얻을 수 없었다. CEP에서는 C7552를 제외한 모든 경우에서 모든 주입력에 동시에 정사건과 부사건을 스케줄하여 실험을 수행하였다. C7552의 경우에는 이렇게 할 경우 약 120MB의 메모리를 모두 소진하게 되므로 동시에 스케줄하는 사건을 한번에 193개의 주입력으로 제한하였다. 그 결과 수행시간이 매우 길게 나왔다. 그러나 TBTP에서와 같이 항의 수를 2000으로 제한한 경우에는 신속한 응답을 얻을 수 있었다. CEP의 경우 C7552를 제외한 경우 대략 10MB~80MB의 메모리가 사용되었다. TBTP는 실험한 모든 대상에서 절삭임계값이 적용되어 실험결과에서 지연시간이 실제보다 길게 나올 가능성을 가지고 있으나 본 논문에서 제안된 CEP의 경우에는 항의 수를 제한할 필요가 없어 정확한 결과를 보장할 수 있었다. 단 C7552에서는 수행시간이 너무 길어 실용적이지 못하므로 항의 수를 제한하는 편이 바람직하였다. 항의 수가 제한된 경우에는 TBTP에서와 같이 다소 부정확한 결과가 나올 가능성을 감수해야 한다. 또한 모든 경우에 본 논문에서 제안한 CEP가 TBTP보다 수행속도가 매우 빠른 것을 확인할 수 있었다.

표 6 실험 결과

Table 6 Experimental results * : 부울항의 수를 2000으로 제한

| 회로 | TBTP 수행시간 (초) | CEP 수행시간 (초) |
|-------|---------------|---------------------|
| C880 | 202.2 * | 4.9 |
| C1355 | 27.1 * | 4.8 |
| C1908 | 372.7 * | 13.5 |
| C2670 | 248.8 * | 21.8 |
| C3540 | 385.9 * | 25.0 |
| C5315 | 306.9 * | 62.5 |
| C6288 | 3844.3 * | 143.0 |
| C7552 | 608.8 * | 575967.8 330.0 * |

5. 결론

본 논문에서는 병렬 사건전파 방식에 의한 타이밍 분석 기법을 제안하였다. 구현된 시스템의 ISCAS85 회로에 대한 실험결과, 지연시간 부울대수를 이용한 기존의 방식과 비교하여 모든 경우에 우수한 결과를 얻었다.

기존의 방식중 지연시간 부울대수를 이용한 방법은 유연한 지연시간 모델링이 가능하면서도 빠른 시간에 위험경로와 활성화 입력을 찾을 수 있지만 회로가 커지면 부울항의

수가 크게 증가하여 이를 제한할 수 밖에 없고 이러한 경우 거짓경로를 위험경로로 판정할 수도 있다. 제안된 기법에서는 모든 부울항이 경로를 따라 전파되지 않고 단지 사건과 관련된 부울항만 전파되므로 거짓경로가 많을 경우 부울항의 수가 크게 증가하지 않는다. 따라서 기존의 방법보다 메모리 사용량이 매우 적고 대부분의 경우 항의 수를 제한할 필요가 없으며 수행시간도 크게 단축할 수 있다.

향후 연구에서는 데이터 구조의 개선 등을 통하여 메모리 사용량을 더욱 줄일 수 있는 방법이 모색되어야 할 것이다.

감사의 글

본 연구는 한국과학재단의 특정기초연구비 (96-0102-16-01-3) 지원으로 수행되었으며 지원에 감사드립니다.

참고문헌

[1] H. Chang, and J. A. Abraham, "VIPER : An Efficient Vigorously Sensitizable Path Extractor," Proc. of the 30th DAC, pp. 112-117, 1993.
 [2] Y. C. Ju, and R. A. Saleh, "Incremental Techniques for the Identification of Statically Sensitizable Critical Paths," Proc. of the 28th DAC, pp. 541-546, 1991.
 [3] C. Oh, and M. R. Mercer, "Efficient Logic-Level Timing Analysis Using Constraint-Guided Critical Path Search", IEEE Transactions on VLSI systems, Vol 4(No. 3), pp. 346-355, 1996.
 [4] P. Ashar, and S. Malik, "Functional Timing Analysis Using ATPG", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 14(No. 8), pp. 1025-1030, 1995.
 [5] S. T. Huang, T. M. Parng, and J. M. Shyu, "A New Approach to Solving False Path Problem in Timing Analysis," IEEE Int. Conf. Computer-Aided Design, pp. 216-219, 1991.
 [6] S. T. Hung, T. M. Parng, and J. M. Shyu, "A Polynomial-Timing Heuristic Approach to Approximate a Solution to the False Path Problem," Proc. of the 30th DAC, pp. 118-122, 1993
 [7] S. T. Huang, T. M. Parng, and J. M. Shyu "A Polynomial-Time Heuristic Approach to Solving the False Path Problem," IEEE Transactions on Circuits and Systems-Fundamental Theory and Applications. Vol. 43(No. 5), pp. 386-395, 1996.
 [8] R. E. Bryant, "Graph-Based Algorithms for Boolean Function Manipulation," IEEE Transactions on Computers, Vol. C-35, No.8 pp. 677-691, Aug.1986.
 [9] R. I. Bahar, E. A. Frohm, C. M. Gaona, G. D. Hachtel, E. Macii, A. Pardo, and F.Somenzi, "Algebraic Decision Diagrams and Their Applications," Proc. of the 30th DAC, pp. 188-191, 1993.

[10] ISCAS-85 Benchmarks, Special Session: Recent Algorithms for Gate Level ATPG with Fault Simulation and Their Performance Assessment, IEEE International Symposium on Circuits and Systems, June 1985.
 [11] Yuji Kukimoto, and Robert K. Brayton, "Hierarchical Functional Timing Analysis," Proc. of the DAC, pp. 580-585, 1998.
 [12] Y. Blaquiere, M. Dagenais, and Y. Savaria, "Timing Analysis Speed-up Using a Hierarchical and a Multimode Approach," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 15(No. 2) 1996.
 [13] H. C Chen, and D. H. C. Du, "Path Sensitization in Critical Path Problem", Proc. of the ICCAD, pp. 208-211, 1991.
 [14] S. A. Szygenda and E. W. Thomson, "Digital Logic in a Time Based Table Driven Environment Part 1 ; Design Verification." in Computer, pp. 38-49 March 1975.

저 자 소 개



한 장 호 (韓 昌 昊)

1980년 성균관대학교 전자공학과 졸업.
 1982년 서울대학교 대학원 전자공학과 졸업 (석사). 1991년 The Univ. of Texas at Austin 전기 및 컴퓨터공학과 (공학).
 1983년~86년 한국전자통신연구소연구원.
 1991년~92년 Cadence Design Systems,

Inc., Senior Member of Technical Staff. 1992년~현재 인하대학교 전자계산공학과 부교수.

Tel : (032) 860-7449, Fax : (032) 874-1435 (부서 공용)

E-mail : chhan@inha.ac.kr