

시간 제한 조건을 가진 자유 선택 신호 전이 그래프로부터 비동기 회로의 합성

정 성 태[†] · 정 석 태[†]

요 약

본 논문에서는 시간 제한 조건을 가진 자유 선택 신호 전이 그래프로부터 비동기 회로를 합성하는 방법을 기술한다. 이 방법에서는 상태 그래프를 생성하지 않고 신호 전이 그래프로부터 직접 신호 전이들간의 관계를 구하여 비동기 회로를 합성한다. 본 논문의 합성 방법에서는 자유 선택 신호 전이 그래프를 선택 행위가 없는 결정성 신호 전이 그래프로 분할한 다음에 각 결정성 신호 전이 그래프에 대하여 타이밍 분석을 수행하여 임의의 두 신호 전이 사이의 시간 제약 병렬 관계와 시간 제약 인과 관계를 구한다. 다음에는 이 관계들을 이용하여 각 결정성 신호 전이 그래프에 대한 합성을 수행하고 그 결과를 합병함으로써 전체 회로를 합성한다. 실험 결과에 의하면 본 논문에서 제안한 합성 방법은 상태 공간이 큰 회로에 대하여 현저하게 합성시간을 단축시킬 수 있을 뿐 아니라 기존의 상태 그래프 기반 합성 방법과 비교하여 거의 같은 면적의 회로를 합성한다.

Synthesis of Asynchronous Circuits from Free-Choice Signal Transition Graphs with Timing Constraints

Sung Tae Jung[†] · Suck Tae Joung[†]

ABSTRACT

This paper presents a method which synthesizes asynchronous circuits from free-choice Signal Transition Graphs (STGs) with timing constraints. The proposed method synthesizes asynchronous circuits by analyzing the relations between signal transitions directly from the STGs without generating state graphs. The synthesis procedure decomposes a free-choice STG into deterministic STGs which do not have choice behavior. Then, a timing analysis extracts the timed concurrency and timed causality relations between any two signal transitions for each deterministic STG. The synthesis procedure synthesizes circuits for each deterministic STG and synthesizes the final circuit by merging the circuits for each deterministic STG. The experimental results show that our method achieves significant reductions in synthesis time for the circuits which have a large state space, and generates circuits that have nearly the same area as compared to previous methods.

키워드: 자유 선택 신호 전이 그래프(free-choice signal transition graph), 결정성 신호 전이 그래프(deterministic signal transition graph), 합성(synthesis), 시간 제약 비동기 회로(timed asynchronous circuit), 상태 그래프(state graph)

1. 서 론

비동기 회로는 클럭 스큐나 분배 문제가 없고 평균 속도로 동작할 수 있고 온도나 습도와 같은 처리 환경의 변화에 민감하지 않고 모듈화 설계가 용이하고 전력 소모가 적은 회로 구현에 이용될 수 있는 등 동기 회로에 비하여 여러 가지 장점을 가지고 있기 때문에 비동기 회로의 합성을 위한 많은 방법들이 제안되었다. 비동기 회로는 여러 부류로 분류될 수 있는데, 그 중의 한 부류인 속도 독립 회로(speed-independent circuit)는 논리 게이트의 지연 시간에 관계없이 올바르

게 동작하도록 보장되는 견고성을 가지고 있기 때문에 속도 독립 회로의 합성을 위한 많은 방법들이 제안되었다[1-7]. 그러나 어떠한 게이트 지연 시간의 변화에도 올바르게 작동할 수 있도록 하기 위해서는 회로의 면적이 더 많이 요구되고 회로의 속도도 느려질 수 있다.

이러한 문제를 극복하기 위해 제안된 회로가 시간 제약 비동기 회로(timed asynchronous circuit)이다. 시간 제약 비동기 회로는 주어진 타이밍 조건 하에서 올바르게 동작하므로 회로의 견고성은 속도 독립 회로보다 떨어지지만 회로의 면적이나 속도에 있어서 효율적이기 때문에 보다 실질적이고 실용적인 회로로서 인식되고 있다. 시간 제약 비동기 회로의 합성 방법[8, 9]들이 제안되었는데, ATACS[9]는 시간

* 이 논문은 2001년도 원광대학교의 교비 지원에 의해서 연구됨.

† 정 회 원 : 원광대학교 컴퓨터및정보통신공학부 교수
논문접수 : 2001년 7월 4일, 심사완료 : 2001년 11월 23일

제한 조건을 이용함으로써 속도 독립 회로보다 면적과 속도 면에서 성능이 우수한 회로를 보다 빠른 시간 내에 합성할 수 있었다.

그러나 시간 제약 비동기 회로 합성 방법들[8, 9]에서는 중간 단계 표현으로 상태 그래프를 이용하는데, 상태 그래프의 한 노드는 회로의 전체 신호들의 값을 나타낸다. 따라서, 신호의 수가 N 일 때 상태 그래프의 노드 수는 2^N 개가 될 수 있다. 이러한 이유로 이들 합성 방법들은 상태 수가 큰 경우에는 합성에 많은 시간이 소요되거나 메모리의 부족으로 인하여 합성에 실패할 수 있는 문제를 가지고 있다. 예를 들어, ATACS의 경우에는 상태수가 100만 개만 넘어가도 합성에 실패하였는데, 신호의 수가 20개만 되어도 상태 수는 100만 개 이상이 될 수 있기 때문에 문제점으로 지적되고 있다. 이러한 문제를 극복하기 위한 한 방법으로 신호 전이 그래프로부터 상태 그래프를 이용하지 않고 직접 신호 전이들의 관계를 이용하여 해저드가 없는 시간 제약 비동기 회로를 합성하는 방법[10, 11]이 제안되었다. 이 방법에서는 신호 전이 그래프에 대하여 직접 타이밍 분석을 수행하여 신호 전이 사이의 시간 차이를 구하고 그에 근거하여 시간 제약 병렬 관계(timed concurrency)와 시간 제약 인과 관계(timed causality)를 구한다. 그리고 이들 관계에 근거하여 신호 전이 사이의 우선 순위 그래프를 생성한 다음에 그래프 탐색을 통하여 시간 제약 비동기 회로를 합성한다. 이 방법은 상태 공간이 큰 신호 전이 그래프에 대하여 수행 시간을 현저히 단축시킬 수 있었고 기존의 상태 그래프 기반의 방법이 합성에 실패한 회로도 합성할 수 있었다. 그러나 이 방법은 신호 전이 그래프의 부 집합으로서 선택 행위를 포함하지 않는 결정성 신호 전이 그래프만 처리할 수 있다는 제한점을 가지고 있다.

따라서 이러한 제한점을 극복하고자 본 논문에서는 비동기 회로의 기술에 있어서 보편적으로 사용되는 자유 선택 신호 전이 그래프로부터 시간 제약 비동기 회로를 합성하는 방법을 제안한다. 본 논문에서 제안한 합성 방법에서는 자유 선택 신호 전이 그래프를 결정성 신호 전이 그래프로 분할하여 합성에 필요한 신호 전이들 사이의 시간 제약 인과 관계와 시간 제약 병렬 관계를 구한다. 그리고 분할된 각 결정성 신호 전이 그래프에 대하여 합성을 수행한 다음에 결과를 합병함으로써 최종 회로를 구한다. 본 논문에서 제안한 합성 방법을 기존의 다양한 벤치마크들에 적용하여 실험을 수행하였다. 본 논문에서 사용한 벤치마크에는 통신용 칩의 제어기[12], 바코드 생성 제어기[13], CD 플레이어 제어기[14], 캐시 메모리 제어기[15], SCSI 인터페이스 제어기[16], 메모리 접근 제어기[16] 등 실제로 설계된 칩에 포함되어 있는 제어 회로들이 포함되어 있다. 기존의 벤치마크에 대하여 ATACS와 본 논문의 합성 방법을 적용해본 결과 상태 수가 큰 회로에 대

하여 본 논문에서 제안한 방법이 합성 시간을 현저히 줄일 수 있었다. 또한 합성된 회로의 면적이 대부분 같았다. 따라서 기존의 상태 그래프 기반의 합성 방법을 적용할 수 없는 대규모 회로 설계에 본 논문의 합성 방법이 효과적으로 적용될 수 있을 것이다.

2. 시간 제한 조건을 가진 회로 명세

2.1 자유 선택 신호 전이 그래프

시간 제한 조건을 가진 신호 전이 그래프는 Petri 네트[17]의 전이를 신호의 전이로 해석한 것으로서 다음과 같이 정의된다.

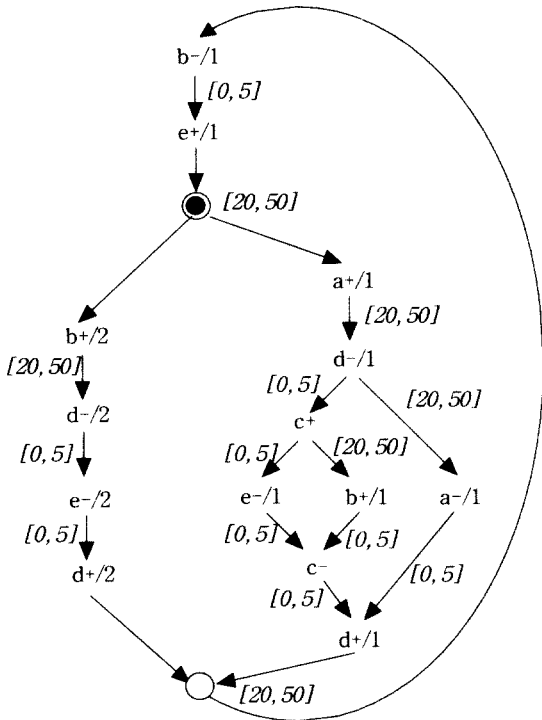
정의 1: 시간 제한 조건을 가진 신호 전이 그래프는 $G = \langle T, P, F, C, M_0 \rangle$ 이다. T 는 신호 전이의 집합을 나타내고 P 는 장소(place)의 집합을 나타낸다. F 는 신호 전이와 장소 사이의 흐름 관계를 나타내며 $F \subseteq (P \times T) \cup (T \times P)$ 의 관계를 갖는다. C 는 시간 제한 조건을 나타내는 것으로서 각 장소에 할당되며 $C = N \times (N \cup \infty)$ 의 값을 갖는다. ($N = \{0, 1, 2, 3, \dots\}$). M_0 는 초기 표식(marking)을 나타내는데, 표식이란 신호 전이 그래프의 각 장소에 토큰이라 불리는 0이상의 정수를 할당하는 함수로 정의된다.

만약에 $(t, p) \in F$ 의 관계가 만족되면 t 를 p 의 입력 전이라 하고 p 를 t 의 출력 장소라 한다. 반대로 $(p, t) \in F$ 의 관계가 만족되면 p 를 t 의 입력 장소라 하고 t 를 p 의 출력 전이라 한다. 본 논문에서는 자유 선택 신호 전이 그래프라 불리는 신호 전이 그래프를 다룬다.

정의 2: 두 전이 $(p, t_1) \in F$ 과 $(p, t_2) \in F$ 관계가 만족할 경우에 p 가 전이 t_1 과 전이 t_2 의 유일한 입력 장소인 신호 전이 그래프를 자유 선택 신호 전이 그래프라 한다.

(그림 1)에는 시간 제한 조건을 가진 자유 선택 신호 전이 그래프의 예가 나타나 있다. 신호 전이 그래프에서 신호는 s 와 같이 이름으로 내는데, (그림 1)에서 a, b 는 입력 신호이고 c, d, e 는 출력 신호이다. 신호 전이는 $s+$, $s-$ 등과 같이 나타낸다. $s+$ 는 신호 s 의 값이 0에서 1로 전이하는 것을 나타내고 $s-$ 는 1에서 0으로 전이하는 것을 나타낸다. 한 신호의 전이가 여러 번 발생하면 $s+/1$, $s-/1$ 과 같이 나타낸다. 원은 장소를 나타내고 원안의 검은색 점은 토큰을 나타낸다. 그림을 단순하게 하기 위해 입력 전이와 출력 전이가 하나씩인 장소는 그림에서 생략하였다. 장소와 전이 사이의 아크는

흐름 관계를 나타낸다. 각 장소에는 시간 제한 조건이 “[하한값, 상한값]”과 같은 형식으로 주어진다. 신호 전이는 모든 입력 장소가 토큰을 하나씩 가지고 있고 각 장소에 토큰이 있는 시간이 주어진 시간 제한 조건의 “하한값”을 지났으면 활성화된다. 활성화된 신호 전이가 일어나게 되면 입력 장소에 있던 토큰들은 제거되고 출력 장소에 토큰이 하나씩 추가된다. (그림 1)의 초기 표식은 $a+1$ 또는 $b+2$ 전이가 20에서 50 단위 시간 사이에 발생할 수 있음을 나타내고 있다. 다음에서 s^* 는 전이 $s+$ 또는 $s-$ 를 나타낸다고 하자. 전이 s^* 로부터 전이 t^* 로의 시간 제한 조건을 $TC(s^*, t^*)$ 로 나타내기로 한다. 그리고 $TC_L(s^*, t^*)$ 은 시간 제한 조건의 하한값 $TC_U(s^*, t^*)$ 은 상한값을 나타내기로 한다.



(그림 1) 시간 제한 조건을 가진 자유 선택 신호 전이 그래프 예

2.2 타이밍 분석

회로 합성을 위해서 필요한 타이밍 정보를 얻기 위해서 타이밍 분석을 수행해야 한다. 본 논문에서는 타이밍 정보를 얻기 위해 참고문헌 [7]의 타이밍 분석 알고리즘을 이용한다. 이 알고리즘은 결정성 신호 전이 그래프만을 처리할 수 있기 때문에 본 논문에서는 자유 선택 신호 전이 그래프로부터 분할된 결정성 신호 전이 그래프에 대하여 타이밍 분석을 수행한다.

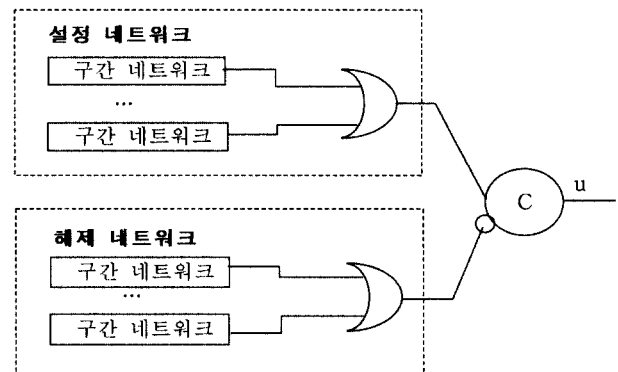
합성에 필요한 타이밍 정보는 결정성 신호 전이 그래프의 임의의 두 신호 전이 사이의 최소 시간 차이와 최대 시간 차이이다. 전이 s^* 를 기준으로 했을 경우에 s^* 와 t^* 의 최소

시간 차이를 $TD_L(s^*, t^*)$ 로 나타내기로 하고 최대 시간 차이를 $TD_U(s^*, t^*)$ 로 나타내기로 하자. 그리고 $TD(s^*, t^*) = [TD_L(s^*, t^*), TD_U(s^*, t^*)]$ 와 같이 나타내기로 한다. 예를 들어, $TD(s^*, t^*) = [10, 30]$ 는 s^* 전이가 발생한 후 10 단위 시간에서 30 단위 시간 사이에 t^* 전이가 발생한다는 것을 나타낸다. 그리고 $TD(s^*, t^*) = [-50, -20]$ 은 전이 s^* 가 발생하는 시간에서 20에서 50 단위 시간 전에 전이 t^* 가 발생한다는 것을 나타낸다.

3. 합성 절차

3.1 단일 큐브 회로 모델

본 논문의 합성 알고리즘에서는 출력 신호 u 에 대하여 (그림 2)와 같은 모델의 회로를 합성한다. 여기에서 C-원소는 저장 장치로서 두 입력이 모두 1이 되면 출력이 1이 되며, 두 입력이 모두 0이 되면 출력이 0이 되고 그 외의 경우에는 이전의 출력 값을 그대로 유지한다. 따라서 설정 네트워크의 출력이 1이 되고 해제 네트워크의 출력이 0이 되면 C-원소의 출력은 1이 된다. 반대로 해제 네트워크의 출력이 1이 되고 설정 네트워크의 출력이 0이 되면 C-원소의 출력은 0이 된다. 이 회로 모델에서 설정 네트워크와 해제 네트워크는 두 가지 방법에 의해 구현될 수 있다. 한 가지 방법은 AND, OR, NOT 게이트와 같은 단순 게이트를 사용하여 구현하는 것이다. 일반적으로 단순 게이트를 사용한 구현에서는 회로의 면적이 커지지만 단순한 게이트 라이브러리를 사용할 수 있으므로 실제로 구현하기에 용이하다. 다른 방법은 설정 네트워크와 해제 네트워크 각각을 하나의 복합 게이트로 구현하는 것이다. 복합 게이트를 사용하면 회로의 면적은 작아질 수 있지만 게이트 라이브러리에 여러 가지의 복합 게이트들을 가지고 있어야 하므로 라이브러리 구현이 어렵다는 문제점을 가지고 있다. 참고문헌 [10, 11]에서는 복합 게이트를 사용 한데 반하여, 본 논문의 합성 시스템은 단순 게이트를 사용하였다.



(그림 2) 목표 회로 모델

설정 네트워크와 해제 네트워크는 구간 네트워크들을 OR 게이트로 연결하여 구성되고 각 구간 네트워크는 하나의 AND 게이트로 구현된다. 여기에서, 구간 네트워크를 하나의 AND 게이트 형태로 합성하므로 단일 큐브 회로 모델이라 한다[6, 7]. 여기에서 구간이란 출력 신호의 한 전이, u^*/i 가 활성화되는 시간부터 그 전이의 다음의 역 방향 전이, $\overline{u^*/i}$ 가 활성화되는 시간 사이를 나타낸다고 하자. 그리고 이를 $u^*/i \mapsto \overline{u^*/i}$ 와 같이 나타내기로 하자. 여기에서, u^*/i 를 구간의 시작 전이라 하고 $\overline{u^*/i}$ 를 구간의 종료 전이라 하자. 그리고 설정 네트워크의 구간 네트워크를 설정 구간 네트워크라 하고 해제 네트워크의 구간 네트워크를 해제 구간 네트워크라 하기로 하자. 참고문헌 [10, 11]에서는 한 구간 네트워크가 하나의 구간 네트워크를 구현하는 경우만을 다루었는데, 본 논문에서는 한 구간 네트워크가 여러 구간을 구현할 수 있도록 확장하였다. 그에 따라, 회로 모델의 무해저드 조건도 정리 1과 같이 확장된다. 정리 1의 무해저드 조건은 기존의 속도 독립 회로 합성기 [3, 4, 6]에 사용되는 무해저드 조건과 동일하다. 단지 다른 점은 속도 독립 회로에서는 모든 신호 전이에 대하여 전이 s^* 로부터 전이 t^* 로의 시간 제한 조건이 $TC(s^*, t^*) = [0, \infty]$ 이라는 것이다. 속도 독립 회로에서 사용된 무해저드 조건에 대한 증명은 참고문헌 [3, 4, 6]에 나타나 있다.

정리 1: (회로 모델의 무해저드(hazard-free) 조건) 신호 u 에 대한 단일 큐브 회로 모델 구현은 각 구간 네트워크가 다음의 조건을 만족하면 주어진 시간 제한 조건 $T = \{u^*/i \mapsto \overline{u^*/i} \mid 1 \leq i \leq n\}$ 내에서 해저드를 갖지 않는다. 구간 네트워크가 구간의 집합을 구현한다고 가정하자. 여기에서, 설정 구간 네트워크의 경우에는 u^* 가 상향 전이(u^+)이고, 해제 구간 네트워크의 경우에는 하향 전이(u^-)이다.

- i) 구간 네트워크의 출력이 T 에 속하는 각 구간의 시작 전이 u^*/i 가 활성화 될 때에 1이 된다.
- ii) 구간 네트워크의 출력이 1이 된 다음에는 u^*/i 가 일어날 때까지는 1의 값을 유지하다가 $\overline{u^*/i}$ 가 활성화될 때까지는 0이 된다.
- iii) 구간 네트워크의 출력이 0이 된 다음에는 T 에 속하는 임의의 구간의 시작 전이 u^*/j 가 활성화 될 때까지 구간 네트워크의 출력이 0으로 유지된다.

(증명) 해저드가 발생하는 경우는 두 가지가 있다. 첫 번째는 출력 신호 u 가 0의 값을 유지하도록 예정된 구간에서 $0 \rightarrow 1$ 전이가 일어나는 경우이다. 두 번째는 출력 신호 u 가 1의 값을 유지하도록 예정된 구간에서 $1 \rightarrow 0$ 전이가 일어나는 경우이다. 먼저, 첫 번째 경우가 불

가능함을 보이도록 하자. 설명의 편의를 위하여 설정 네트워크의 출력을 S 라 하고 해제 네트워크의 출력을 R 이라 하자. 신호 (S, R) 이 다음 세 가지의 경우와 같이 전이할 때에 신호 u 에서 $0 \rightarrow 1$ 전이가 발생한다.

- 1) $(0, 0) \rightarrow (1, 0)$
- 2) $(0, 1) \rightarrow (1, 0)$
- 3) $(1, 1) \rightarrow (1, 0)$

위의 세 조건 모두 S 의 값은 1의 값을 가져야 한다. 먼저, u 가 0의 값을 유지해야 하는 구간에서 S 의 값이 0에서 1로 전이할 수 있는지 살펴보자. 설정 구간 네트워크는 조건 i)에 의해 반드시 u^+ 전이가 활성화될 때에 1로 전이하고, 그 이외의 경우에는 조건 ii)와 iii)에 의해 1로 전이하는 경우가 없다. 따라서, 출력 신호 u 가 0의 값을 가져야 하는 구간에서는 S 의 값은 0에서 1로 전이할 수 없다. 즉, u 의 값이 1로 설정되어야 할 때에만 S 의 값이 0에서 1로 전이된다. 그렇다면 u 가 0의 값을 유지해야 하는 구간에서 S 의 값이 1이 될 수 있는 경우는, u 의 값을 1로 설정하기 위해 1로 전이되었던 S 의 값이 계속해서 1로 유지된 경우밖에 없다. 그러나 u 의 값이 0으로 전이하기 위해서는 이미 S 의 값이 0이 되어 있어야 하기 때문에 S 의 값이 1로 유지되는 것은 불가능하다. 이와 같이 출력 신호 u 가 0의 값을 유지하도록 예정된 구간에서 $0 \rightarrow 1$ 전이가 일어날 수 없다. 또한 같은 이유로 출력 신호 u 가 1의 값을 유지하도록 예정된 구간에서 $1 \rightarrow 0$ 전이가 일어날 수 없다. ■

3.2 결정성 신호 전이 그래프의 분할

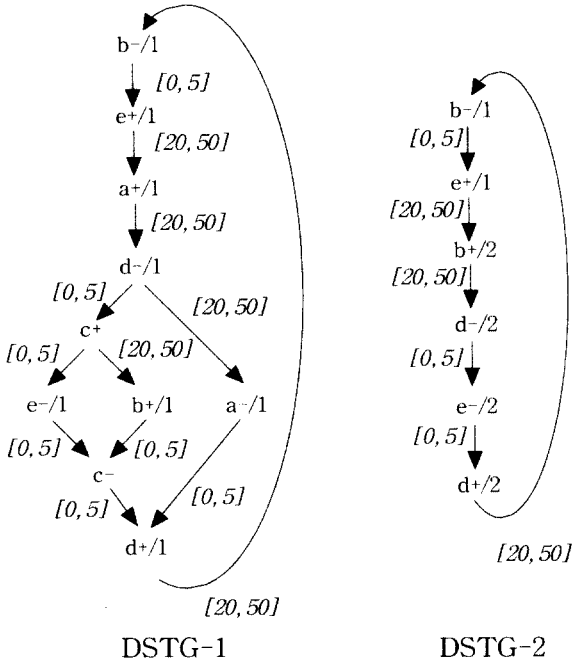
본 논문에서는 자유 선택 신호 전이 그래프를 결정성 신호 전이 그래프로 분할한 다음에 각 결정성 신호 전이 그래프에 대하여 합성을 수행하고 그 결과를 합병하는 방법을 사용한다. 합성된 회로가 해저드를 갖지 않기 위해서는 다음과 같은 조건을 만족해야 한다. 이 조건은 참고문헌 [6]에서 제안된 조건으로서 이에 대한 증명도 참고문헌 [6]에 나타나 있다.

정의 3: (결정성 신호 전이 그래프 분할에 대한 무해저드 조건) 신호 0를 신호 전이 그래프 G 의 출력 신호라 하고 1를 구간 네트워크 N 에 의해서 구현되어야 할 구간들의 집합이라 하자. 그리고 M 을 G 에 대한 결정성 신호 전이 그래프들의 집합이라 하자. 구간 네트워크 N 이 해저드를 갖지 않기 위해서는 다음의 조건이 만족되어야 한다.

- 1) 집합 I 에 속한 구간들의 일부 또는 전체를 포함하는 결정성 신호 전이 그래프에 대하여 구간 네트워크 N 은 정

리 1의 무해저드 조건을 모두 만족한다.

- 2) 신호 0의 구간 중에서 집합 1의 어떠한 구간도 포함하지 않지만 다른 구간을 포함하는 결정성 신호 전이 그래프에 대해서 구간 네트워크 N은 항상 0의 값을 가진다.
- 3) 신호 0의 어떠한 구간도 포함하지 않는 결정성 신호 전이 그래프에 대하여 구간 네트워크 N은 항상 0의 값을 가지거나 항상 1의 값을 가진다.



(그림 3) (그림 1)의 신호 전이 그래프에 대한 결정성 신호 전이 그래프 구성 요소

결정성 신호 전이 그래프의 분할은 참고문헌 [1]의 방법을 사용하였다. 이 방법에서는 각 선택 장소에 대하여 한 개의 출력 전이를 선택한 다음에, 선택된 출력 전이로부터 도달 가능한 전이들만 남기고 나머지 전이들을 모두 제거함으로써 결정성 신호 전이 그래프를 구한다. 본 논문의 합성 시스템에서는 자유 선택 신호 전이 그래프에 존재하는 모든 결정성 신호 전이 그래프를 필요로 하지는 않는다. 예를 들어, 선택 장소 A에 2개의 출력 전이 s_1, s_2 가 있고 선택 장소 B에 2개의 출력 전이 s_3, s_4 가 있을 때에, 결정성 신호 전이 그래프는 네 개가 존재하게 된다. 즉, (s_1, s_3) 를 선택한 경우, (s_1, s_4) 를 선택한 경우, (s_2, s_3) 를 선택한 경우, (s_2, s_4) 를 선택한 경우와 같이 네 가지의 경우에 대하여 결정성 신호 전이 그래프가 존재하게 된다. 그런데, 본 논문의 합성 시스템에서는 이들 중 두 개의 결정성 신호 전이 그래프만 사용해도 된다. 즉, (s_1, s_3) 를 선택한 경우와 (s_2, s_4) 를 선택한 경우의 결정성 신호 전이 그래프만 사용해도 되는데, 이 두 결정성 신호

전이 그래프만으로도 원래의 신호 전이 그래프의 모든 신호 전이를 포함하기 때문이다. 본 논문에서는 일단 모든 결정성 신호 전이 그래프를 구한 다음에 원래의 신호 전이 그래프의 모든 신호 전이를 포함하면서 결정성 신호 전이 그래프의 수가 최소가 되는 경우를 찾는 방법을 사용하였다. (그림 1)의 신호 전이 그래프에 대하여 구한 결정성 신호 전이 그래프가 (그림 3)에 나타나 있다.

3.3 불필요한 아크 제거와 신호 전이들의 관계 추출

본 논문의 합성 알고리즘에서는 타이밍 분석을 이용하여 신호 전이 그래프로부터 불필요한 아크를 제거하고 시간 제약 병렬 관계와 시간 제약 인과 관계를 구한다.

정의 4 : 전이 s^* 로부터 전이 t^* 로의 아크에 대하여 $TD_L(s^*, t^*) > TC_U(s^*, t^*)$ 의 관계가 만족되면 이 아크는 불필요한 아크이다.

예를 들어, (그림 3)의 결정성 신호 전이 그래프 DSTG-1에서는 $TD_L(e-/1, c-) = 15$ 이고 $TC_U(e-/1, c-) = 5$ 이므로 $e-/1$ 전이로부터 $c-$ 전이로의 아크가 불필요한 아크이다. 신호 전이 그래프에서 주어진 타이밍 제한 조건에 의하면 $e-/1$ 전이가 일어난 다음에 0에서 5 단위 시간 사이에 $c-$ 전이가 활성화된다고 되어 있다. 그러나, 타이밍 분석 결과에 의하면 $TD_L(e-/1, c-) = 15$ 으로 $c-$ 전이는 $e-/1$ 전이가 일어난 다음에 적어도 15 단위 시간 후에 일어날 수 있기 때문에 이 아크는 회로의 행위에 아무런 영향을 미치지 못하게 된다. 따라서, 이 아크는 불필요한 아크로서 제거될 수 있게 된다.

본 논문의 합성 알고리즘은 시간 제약 병렬 관계와 시간 제약 인과 관계에 기초하여 회로를 합성한다. 시간 제약 병렬 관계는 주어진 시간 제한 조건 내에서 두 신호 전이가 병렬로 일어날 수 있는가를 나타낸다.

정의 5 : 두 전이 s^* 와 t^* 가 다음 조건을 만족하면 시간 제약 병렬 관계를 갖는다. 이를 $s^* || t^*$ 와 같이 나타내기로 한다.

- 1) 두 전이 s^* 와 t^* 가 시간 제한 조건을 고려하지 않을 때에 서로 병렬 관계에 있다.
- 2) $TD_L(s^*, t^*) \leq 0$ 이고 $TD_U(s^*, t^*) \geq 0$ 이다.

정의 5의 조건이 만족되면 전이 t^* 가 전이 s^* 보다 전에 일어날 수도 있고 후에 일어날 수도 있다는 것을 의미하므로 두 전이는 병렬 관계를 가지는 것이다. 인과 관계는 한 신호 전이가 다른 신호 전이의 발생을 직접적으로 또는 간접적으로 일으킬 수 있는가 그렇지 않은가를 나타내기 위해

서 사용된다. 전이 s^* 가 전이 t^* 의 활성화 전이이면, 즉 전이 s^* 로부터 전이 t^* 로의 아크가 존재하고 이 아크가 불필요한 아크가 아니라면, 전이 s^* 는 전이 t^* 의 발생을 직접적으로 일으키게 된다. 전이 s^* 의 다음의 역방향 전이를 $\overline{s^*}$ 라 하자. 전이 s^* 가 전이 t^* 의 활성화 전이는 아니라도, 전이 s^* 가 일어난 다음에 전이 t^* 가 일어날 수 있고 전이 t^* 가 일어난 다음에 전이 $\overline{s^*}$ 가 일어날 수 있거나 또는 전이 t^* 와 전이 $\overline{s^*}$ 가 병렬로 일어날 수 있으면 전이 s^* 는 전이 t^* 의 발생에 간접적인 영향을 미칠 수 있다. 이러한 인과 관계가 신호 전이 그래프의 주어진 시간 제한 조건 내에서 만족될 경우에 이를 시간 제약 인과 관계라 하고 시간 제한 조건을 고려하지 않고 만족될 경우에 이를 비시간 제약 인과 관계라 하자. 시간 제약 인과 관계는 다음과 같이 정의될 수 있다.

정의 6: 전이 s^* , $\overline{s^*}$, t^* 가 다음의 조건을 만족하면 주어진 전이 s^* 는 전이 t^* 와 시간 제약 인과 관계를 가진다. 이를 $s^* \Rightarrow t^*$ 와 같이 나타내기로 한다.

- 1) $TD_L(s^*, t^*) > 0$ 이거나 전이 s^* 로부터 t^* 로의 경로가 존재하고 $TD_L(s^*, t^*) = 0$.
- 2) $TD_U(t^*, \overline{s^*}) \geq 0$.

시간 제약 병렬 관계는 동일한 결정성 신호 전이 그래프 구성 요소에 포함된 신호 전이 사이에서만 발생하므로, 결정성 신호 전이 그래프 내에서만 시간 제약 병렬 관계를 구한다. 그러나, 시간 제약 인과 관계는 서로 다른 결정성 신호 전이 그래프 구성 요소에 속한 신호 전이 사이에도 발생할 수 있다. 자유 선택 신호 전이 그래프의 모든 신호 전이 사이의 시간 제약 인과 관계를 구하기 위해서는 자유 선택 신호 전이 그래프 전체에 대하여 타이밍 분석을 수행해야 한다. 자유 선택 신호 전이 그래프 전체에 대하여 타이밍 분석을 수행하는 알고리즘[18]이 개발되어 있기는 하지만 분석 시간이 상태수에 지수승으로 비례하기 때문에, 너무 많은 시간이 소요되므로 본 논문에서는 적용될 수 없었다. 따라서, 본 논문에서는 서로 다른 결정성 신호 전이 그래프 구성 요소에 속하는 신호 전이 사이의 시간 제약 인과 관계를 구하기 위해서, 자유 선택 신호 전이 그래프 전체에 대해서는 그래프 탐색을 이용하여 비시간 제약 인과 관계를 구한 다음에 각 결정성 신호 전이 그래프 구성 요소 내에서 타이밍 분석을 통해 시간 제약 인과 관계를 구하는 방법을 사용한다. 결정성 신호 전이 그래프 내에서만 타이밍 분석을 하는 이유는 동일한 결정성 신호 전이 그래프 구성 요소에 속하지 않는 두 전이 s^* 와 t^* 가 비시간 제약 인과 관계를 가지기만 하면 s^* 와 t^* 가 시간 제약 인과 관계를 가지기 때문이다.

3.4 단일 큐브 구간 네트워크 합성

3.4.1 구간 네트워크의 공유

구간 네트워크를 합성하기 위해 먼저 해야 할 일은 한 구간 네트워크가 구현해야 할 구간들을 구하는 것이다. 두 구간의 시작 전이가 같은 활성화 상태를 가지면 그들은 하나의 구간 네트워크에 의해 구현되어야 한다. 본 논문에서는 활성화 큐브 정제화 방법에 의해 같은 활성화 상태를 가지는 전이들을 발견한다. 먼저, 신호 전이 그래프의 출력 신호의 각 전이에 대하여 최소 활성화 큐브를 구한다.

정의 7: 신호 전이 그래프에서 모든 신호들의 집합을 $S = \{s_1, s_2, \dots, s_n\}$ 이라 하자. 출력 신호의 전이 u^* 에 대한 최소 활성화 큐브, $C_{u^*} = c_1 c_2 \dots c_n$ 은 다음과 같이 정의된다.

- 1) 신호 s_i 의 한 전이가 u^* 와 시간 제약 병렬 관계를 가지면 $c_i = X$ 이다.
- 2) 신호 s_i 의 어떤 전이도 u^* 와 시간 제약 병렬 관계를 가지지 않고 $s_i^* \Rightarrow u^*$ 의 관계가 만족될 때에, s_i^* 가 상향 전이이면 $c_i = 1$ 이다. s_i^* 가 하향 전이이면 $c_i = 0$ 이다.

전이 u^*/i 와 u^*/j 를 신호 u 의 전이라 하고 두 전이의 방향이 동일하다고 하자. 만약에 두 전이의 최소 활성화 큐브가 도달 가능한 상태에서 서로 교차하면 두 전이는 하나의 구간 네트워크에 의해서 구현되어야 한다. 두 큐브 $C_{u^*/i} = c_1 c_2 \dots c_n$ 과 $C_{u^*/j} = d_1 d_2 \dots d_n$ 이 서로 교차하기 위해서는 모든 i 에 대하여 c_i 와 d_i 가 X 가 아닐 때에 $c_i = d_i$ 이어야 한다.

본 논문에서는 두 큐브가 도달 가능한 상태에서 교차하는지를 발견하기 위해서 큐브가 더 이상 도달할 수 없는 상태를 포함하지 않을 때까지 큐브들을 정제해 나가는 방법을 사용한다. 큐브의 정제는 그 큐브에 해당하는 전이와 시간 제약 병렬 관계에 있는 장소들의 최소 큐브를 이용하여 구한다. 장소에 대한 최소 큐브는 그 장소에 토큰이 있을 때의 상태들을 나타내는 것으로서 다음과 같이 정의된다.

정의 8: 신호 전이 그래프에서 모든 신호들의 집합을 $S = \{s_1, s_2, \dots, s_n\}$ 이라 하자. 장소 p 에 대한 최소 큐브, $C_p = c_1 c_2 \dots c_n$ 은 다음과 같이 정의된다.

- 1) 신호 s_i 의 한 전이가 p 와 시간 제약 병렬 관계를 가지면 $c_i = X$ 이다.
- 2) 신호 s_i 의 어떤 전이도 p 와 시간 제약 병렬 관계를 가지지 않고 전이 s_i^* 로부터 장소 p 로의 경로가 존재하고 그 경로에 s_i^* 의 역방향 전이가 포함되지 않을 때에, s_i^* 가 상향 전이이면 $c_i = 1$ 이다. s_i^* 가 하향 전이이면 $c_i = 0$ 이다.

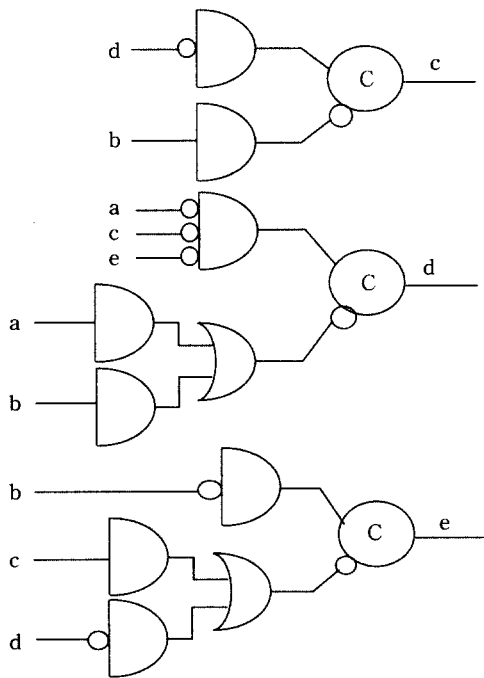
전이 u^* 가 장소 p_1, p_2, \dots, p_n 과 시간 제약 병렬 관계에 있을 때에 u^* 에 대한 큐브는 다음과 같이 정제한다.

$$C_{u^*}' = C_{u^*} \cap (C_{p_1} \cup C_{p_2} \cup \dots \cup C_{p_n})$$

(그림 1)의 신호 전이 그래프에서는 신호 d 의 전이 $d+1$ 과 $d+2$ 에 대한 구간 네트워크가 공유된다. 이들에 대한 최소 활성화 큐브는 각각 $C_{d+1}=01000$, $C_{d+2}=01000$ (여기에서 신호의 순서는 $abcde$ 라 하자)이므로 두 구간은 하나의 구간 네트워크에 의해 구현되어야 한다.

3.4.2 초기 큐브

이 단계에서는 각 구간 네트워크를 단일 큐브 형태로 합성한다. 여기에서는 구간들의 집합 $U = \{u^*/1 \mapsto \overline{u^*/1}, u^*/2 \mapsto \overline{u^*/2}, \dots, u^*/n \mapsto \overline{u^*/n}\}$ 에 대한 구간 네트워크 합성 과정을 설명하기로 한다. 무해저드 조건을 만족하는 구간 네트워크를 구하기 위한 첫 단계는 초기 큐브를 구하는 것이다. 초기 큐브는 $u^*/1, u^*/2, \dots, u^*/n$ 의 모든 여기 신호들의 곱으로 구성된다. 전이 s^* 가 전이 u^*/i 의 활성화 전이일 때 s^* 가 상향전이 이면 s 가 여기 신호가 되고 s^* 가 하향전이 이면 \overline{s} 가 여기 신호가 된다.



(그림 4) (그림 1)의 신호 전이 그래프에 대한 초기 큐브 네트워크

(그림 1)의 신호 전이 그래프의 각 구간 네트워크에 대한 초기 큐브가 (그림 4)에 나타나 있다. 초기 큐브들은 정리 1의 무해저드 조건을 만족하지 않을 수 있다. 모든 여기 신호

들이 1이 되면 u^*/i 전이가 활성화되므로 조건 i)이 만족된다. 그러나 이 여기 신호들은 $\overline{u^*/i}$ 전이가 활성화되기 전에 0이 되지 않을 수 있다. 또한 0으로 되었다라도 임의의 u^*/j 가 활성화 될 때까지 0으로 유지되지 않을 수 있으므로 무해저드 조건 ii)와 iii)이 만족되지 않을 수 있다. 이 조건이 만족되지 않을 경우에는 추가의 신호를 초기 큐브에 추가함으로써 조건이 만족되도록 만든다.

3.4.3 추가 신호 발견

추가 신호들을 구하기 위해서는 먼저 각 결정성 신호 전이 그래프 마다 필요한 추가 신호들을 구한 다음에 이들을 합병하는 방법을 사용한다. 최종 추가 신호의 수를 최소화하기 위해서는 각 결정성 신호 전이 그래프마다 모든 경우의 추가 신호들을 구해야 한다. 정의 3에 나타나 있는 바와 같이 결정성 신호 전이 그래프마다 무해저드 조건을 만족하기 위한 요건이 다르다. 출력 신호를 O 라 하고 구간 네트워크가 구현해야 할 구간들의 집합을 U 라 하자.

1) U 에 속한 구간들을 포함하는 결정성 신호 전이 그래프

이러한 경우에 구간 네트워크는 회로 모델의 무해저드 조건을 모두 만족해야 한다. 이러한 결정성 신호 전이 그래프에 대해서 모든 경우의 추가 신호들을 구하는 알고리즘이 (그림 5)에 나타나 있다. 이 알고리즘은 참고문헌 [10, 11]의 알고리즘을 두 가지 점에서 확장한 것이다. 한 가지는 참고문헌 [10, 11]에서는 한 구간 네트워크가 하나의 구간을 구현하지만 본 논문에서는 여러 구간을 구현할 수도 있다는 것이다. 다른 한 가지는 복합 게이트 대신에 단순 게이트를 사용한다는 것이다. 여기에서는 확장된 부분을 위주로 설명하기로 한다. 여러 구간을 구현할 수 있도록 하는 것은 합성할 수 있는 회로의 범위를 확장할 수 있다는 장점을 제공한다. 그리고 단순 게이트를 사용한다는 것은 게이트 라이브러리를 구축하기가 용이하다는 장점을 제공한다.

참고문헌 [10, 11]의 알고리즘에서는 우선 순위 그래프를 생성한 다음에 경로를 탐색함으로써 추가 신호를 발견한다. 구간 $u^* \mapsto \overline{u^*}$ 에 대하여 여기 신호로 구성된 초기 큐브가 무해저드 조건을 만족하지 않을 때에, 그 이유는 여기 신호의 값이 1이 되는 시간이 너무 길기 때문이다. 따라서 무해저드 조건을 위반하는 시간 동안에 0의 값을 가지는 신호를 추가함으로써 무해저드 조건이 만족될 수 있도록 하는 것이다. 추가되는 신호가 0의 값을 가져야 하는 시간은 우선 순위 그래프에서 출발 노드의 전이가 발생하는 시간부터 행선 노드의 전이가 발생하는 시간까지로 나타내진다. 그리고 출발 노드로부터 행선 노드로의 경로에 있는 전이들로부터 추가 신호를 구한다.

본 논문의 합성 알고리즘에서는 한 구간 네트워크가 여러

구간들을 구현할 수 있어야 한다. 구간 네트워크가 하나의 구간을 구현할 때와 여러 구간을 구현 할 때와 합성 알고리즘에서 달라지는 것은 우선 순위 그래프에서 행선 노드를 결정하는 조건에 있다. 구간 네트워크가 하나의 구간 $u^* \mapsto \overline{u^*}$ 만을 구현할 때에는 구간 네트워크의 출력은 전이 u^* 가 다시 활성화 될 때까지 0으로 유지되어야 한다. 구간 네트워크가 $u^*/1 \mapsto \overline{u^*/1}, u^*/2 \mapsto \overline{u^*/2}, \dots, u^*/n \mapsto \overline{u^*/n}$ 구간들을 구현한다고 하고 구간들이 이 순서대로 나타난다고 하자. 그러면 전이 u^*/i 가 활성화될 때에 1이 되었던 구간 네트워크의 출력은 $\overline{u^*/i}$ 전이가 활성화될 때까지 0이 되어야 하고 $u^*/i+1$ 전이가 활성화될 때까지 0으로 유지되어야 한다. 따라서 참고문헌 [10, 11]에서는 행선 노드가 구간의 시작 전이에 의해서 결정되었지만 본 논문의 알고리즘에서는 현재 구간의 다음 구간의 시작 전이에 의해서 결정된다는 것이 다르다.

단순 게이트를 사용한 합성을 위해 알고리즘에서 확장된 부분은 출발 노드가 각 행선 노드별로 독립적으로 결정된다는 것이다. 현재 추가 신호를 구하고자 하는 구간이 $u^*/i \mapsto \overline{u^*/i}$ 라 하고 e^*/j 가 u^*/i 의 활성화 전이라고 하자. 그러면 우선 순위 그래프에서 전이 $\overline{e^*/j-1}$ 이 행선 노드가 된다. 전이 e^*/j 에 해당하는 여기 신호는 전이 e^*/j 에 의해 1이 되는데, 이 때가 전이 u^*/i 가 활성화되는 때이다. 그런데, 이 여기 신호는 전이 $\overline{e^*/j}$ 에 의해서 0이 되었다가 전이 $e^*/j+1$ 에 의해서 다시 1이 된다. 전이 $\overline{e^*/j}$ 와 $e^*/j+1$ 가 발생할 동안에 다른 여기 신호와 추가 신호들이 1의 값을 가지고 있으면 구간 네트워크의 출력은 0으로 되었다가 다시 1이 되는데 이로 인하여 해저드가 발생하게 된다. 참고문헌 [10, 11]에서와 같이 복합 게이트로 구성된 회로를 합성할 때에는 구간 네트워크의 출력은 u^*/i 가 활성화 될 때에 1이 되었다가 $\overline{u^*/i}$ 전이가 활성화 될 때까지만 0이 되면 되고 그 사이에는 얼마든지 값이 변해도 해저드가 발생하지 않는다. 따라서 전이 u^*/i 와 전이 $\overline{u^*/i}$ 사이에 있는 전이들이 출발 노드가 되었다. 그러나 단순 게이트로 구성된 회로에서는 구간 네트워크의 출력은 u^*/i 가 활성화 될 때에 1이 되었다가 그 다음에 한번 0이 되면 계속해서 0의 값을 유지해야만 하고 다시 1의 값을 가지게 될 때는 반드시 다른 구간의 시작 전이가 활성화되는 때이어야만 해저드가 발생하지 않는다. 따라서 행선 노드 $\overline{e^*/j-1}$ 에 대한 출발 노드의 전이는 u^*/i 와 $\overline{u^*/i}$ 사이에 있어야 할 뿐만 아니라 전이 e^*/j 와 $e^*/j+1$ 사이에 있어야 한다. (그림 5)의 알고리즘에서 출발 노드들이 세 그룹 SA, SB, SC로 분류되어 있다. 전이 u^*/i 와 전이 $\overline{u^*/i}$ 사이에 있는 출발 노드는 SA 그룹에 포함되고 전이 e^*/j 와 전이 $e^*/j+1$ 사이에 있는 출발 노드는 SB 그룹에 포함된다.

```

shrink(G, U)
(
   $U_G = \{u^*, \overline{u^*} \mid u^*, \overline{u^*} \in U \text{ and } u^* \in G \text{ and } \overline{u^*} \in G, 1 \leq i \leq m\}$ 
  Foreach  $u^*/i \mapsto \overline{u^*/i}$  in  $U_G$ 
  /* 우선 순위 그래프 생성 */
   $\langle V, E \rangle = \langle \emptyset, \emptyset \rangle$ 

  /* 출발 노드와 행선 노드 결정 */
   $S_N = \{u^*\}$ 
  Foreach  $e^*/k$  (an enabling transition of  $u^*/i$ )
   $D_N = D_N \cup \{ \overline{e^*/k-1} \}$ 
   $SC_{\overline{e^*/k-1}} = \{u^*\}$ 

  Find  $e^*/k$  which satisfies  $e^*/k \Rightarrow u^*/i$ 
  If ( $e^*/k+1 \parallel \overline{u^*/i}$ )
  Foreach  $s^*/j$  in G
  If ( $u^*/i \Rightarrow s^*/j$  and not ( $s^*/j \parallel \overline{u^*/i}$ ), and  $\overline{s^*/j} \Rightarrow u^*/i$  (for some  $l$ ) and (( $e^*/k \Rightarrow s^*/j$  or  $\overline{e^*/k} \Rightarrow s^*/j$ ) and not ( $e^*/k+1 \parallel s^*/j$ )))
   $SC_{\overline{e^*/k-1}} = SC_{\overline{e^*/k-1}} \cup \{s^*\}$ 
  Else if ( $u^*/i \Rightarrow s^*/j$  and not ( $s^*/j \parallel \overline{u^*/i}$ ) and  $\overline{s^*/j} \Rightarrow u^*/i$  (for some  $l$ ))
   $SA_{\overline{e^*/k-1}} = SA_{\overline{e^*/k-1}} \cup \{s^*\}$ 
  Else if ( $u^*/i \Rightarrow s^*/j$  and  $\overline{s^*/j} \Rightarrow u^*/i$  (for some  $l$ ) and (( $e^*/k \Rightarrow s^*/j$  or  $\overline{e^*/k} \Rightarrow s^*/j$ ) and not ( $e^*/k+1 \parallel s^*/j$ )))
   $SB_{\overline{e^*/k-1}} = SB_{\overline{e^*/k-1}} \cup \{s^*\}$ 
  Else
  Foreach  $s^*/j$  in G
  If ( $u^*/i \Rightarrow s^*/j$  and not ( $s^*/j \parallel \overline{u^*/i}$ ) and  $\overline{s^*/j} \Rightarrow u^*/i$  (for some  $l$ ) and (( $e^*/k \Rightarrow s^*/j$  or  $\overline{e^*/k} \Rightarrow s^*/j$ ) and not ( $e^*/k+1 \parallel s^*/j$ )))
   $SC_{\overline{e^*/k-1}} = SC_{\overline{e^*/k-1}} \cup \{s^*\}$ 
   $S_N_{\overline{e^*/k-1}} = SA_{\overline{e^*/k-1}} \cup SB_{\overline{e^*/k-1}} \cup SC_{\overline{e^*/k-1}}$ 
   $V = V \cup S_N_{\overline{e^*/k-1}}$ 
   $V = V \cup D_N$ 

  /* 우선 순위 그래프 확장 */
  Foreach unprocessed node  $s^*/j$  in V
  Foreach  $t^*/j$  in G
  If (( $s^*/j \parallel t^*/k$  or  $s^*/j \Rightarrow t^*/k \overline{t^*/l} \Rightarrow u^*/i$ ) and  $t^*/k \Rightarrow \overline{s^*/j}$  and (for some  $l$ ))
   $V = V \cup \{t^*/k\}$ 
   $E = E \cup \{(s^*/j, t^*/k)\}$ 

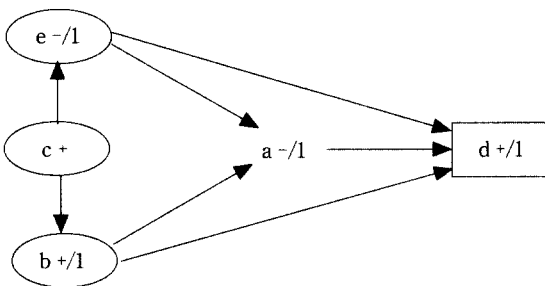
  Foreach  $d_j$  in  $D_N$ 
  Foreach  $s_k$  in  $S_{N_{d_j}}$ 
   $EX_{i,j,k} = \text{Find\_extra\_signals\_for\_each\_interval}(u^*, d_j, s_k)$ 
  Find_all_possible_extra_signal_sets(EX)
)

```

(그림 5) U에 속한 구간들을 포함하는 결정성 신호 전이 그래프에 대한 추가 신호 발견 알고리즘

두 조건을 모두 만족하는 출발 노드는 SC 그룹에 포함된다. 그런데, 전이 $e^*/j+1$ 과 전이 $\overline{u^*/i}$ 가 시간 제약 병렬 관계를 갖지 않으면 그룹 SB에 속하는 노드들은 SA 그룹에도 속하게 되므로 SC 그룹만 구하면 된다. 출발 노드와 행선 노드를 구한 다음에는 알고리즘에 나타나 있는 바와 같이 우선 순위 그래프를 확장한다. 그 다음에는 각 행선 노드에 대하여 해당 출발 노드로부터의 경로를 찾고 경로의 전이로부터 추가 신호들을 구한다. 출발 노드가 SC 그룹에 속할 경우에는 한 출발 노드로부터 행선 노드로의 경로에 있는 전이들만 고려하면 되는 반면에, SA와 SB에 속하는 노드들에 대해서는 SA에 속하는 한 출발 노드로부터의 경로와 SB에 속하는 한 출발 노드로부터의 경로 둘 다를 모두 고려해야 한다.

(그림 1)의 신호 전이 그래프에서 구간 집합 $U = \{c+ \mapsto c-\}$ 에 대하여 DSTG-1은 이 구간을 포함하므로 (그림 5)의 알고리즘이 적용된다. 이 경우에 대하여 생성된 우선 순위 그래프는 (그림 6)과 같고 최소 추가 신호 집합은 출발 노드로부터 행선 노드로의 최단 경로가 $b+1 \rightarrow d+1$ 과 $e-1 \rightarrow d+1$ 이므로 $E_{DSTG-1} = \{\overline{b}, e\}$ 이다. 그림에서 타원은 출발 노드를 사각형은 행선 노드를 나타낸다. 전이 $c+$ 는 구간의 시작 전이이고 전이 $e-1$ 과 $b+1$ 은 전이 $c+$ 와 전이 $c-$ 사이에서 일어나는 전이로서 (그림 5)의 알고리즘의 출발 노드 조건을 만족한다. 그리고 이들 세 전이 모두 SC 그룹에 대한 조건을 만족하므로 SC 그룹에 포함된다. 전이 $d-1$ 이 전이 $c+$ 의 활성화 전이이므로 이 전이의 바로 전의 역방향 전이인 $d+1$ 이 행선 노드가 된다. 중간 노드인 $a-1$ 과 아크들은 (그림 5)의 알고리즘에서 확장 조건에 따라 삽입되었다.



(그림 6) 구간 집합 $U = \{c+ \mapsto c-\}$ 에 대한 DSTG-1의 우선 순위 그래프

2) U에 속한 구간들을 포함하지 않지만 신호 0의 다른 구간을 포함하는 결정성 신호 전이 그래프

U에 속한 구간들을 포함하지 않지만 신호 0의 다른 구간을 포함하는 결정성 신호 전이 그래프를 G라 할 때에, 구간 네트워크는 G에 대하여 항상 0의 값을 가져야 한다. 본 논문의 합성 방법에서는 먼저 결정성 신호 전이 그래프 구성 요

소에 대해서 항상 0의 값을 가지는 신호인 0-안정 신호들을 구한다.

정의9: 결정성 신호 전이 그래프 구성 요소 G에 대하여 다음의 두 조건을 만족하는 신호 s를 0-안정 신호라 한다.

- i) G의 각 전이 t^* 는 신호 s의 하향 전이들 중의 한 전이 $s- /i$ 와 $s- /i \Rightarrow t^*$ 관계를 갖는다.
- ii) U에 속하는 각 구간의 시작 전이 u^*/i 는 신호 s의 상향 전이들 중의 한 전이 $s+ /j$ 와 $s+ /j \Rightarrow u^*/i$ 의 관계를 갖고 신호 s의 어떠한 전이와도 시간 제약 병렬 관계를 갖지 않는다.

정의 9와 유사하게 신호 \overline{s} 도 0-안정 신호가 될 수 있다. 만약에 신호 s 또는 \overline{s} 가 G에 대하여 0-안정 신호일 때에는 이 신호의 어떠한 전이도 G에 나타나지 않는다. 결정성 신호 전이 그래프 구성 요소 G에 대하여 0-안정 신호들이 존재하면 이 신호들 중의 한 신호만 구간 네트워크에 추가해도 G에 대해서는 무해저드 조건이 만족된다. 0-안정 신호들을 구한 다음에는 둘 또는 그 이상의 신호들로 구성되면서 G에 대하여 항상 0의 값을 갖는 큐브들을 구한다. G만을 고려할 때에는 0-안정 신호들만 구하면 추가 신호의 수가 최소화 될 수 있지만 여러 결정성 신호 전이 그래프 구성 요소에 대하여 전체적으로 추가되는 신호의 수를 최소화하기 위해서는 0-안정 신호뿐만 아니라 둘 이상의 신호들로 구성된 큐브들도 모두 구해야 한다. 그러나, 초기 큐브에 이미 0-안정 신호가 포함되어 있으면 G에 대해서는 추가의 신호가 필요하지 않으므로 둘 이상의 신호로 구성된 큐브들은 구할 필요가 없게 된다.

결정성 신호 전이 그래프 구성 요소에 대하여 항상 0의 값을 가지는 둘 이상의 신호로 구성된 큐브를 구하는 문제는 앞 절에서 추가 신호를 발견하기 위한 방법과 유사하게 해결할 수 있다. 앞 절에서 출발 노드로부터 행선 노드까지 한 경로에 대한 추가 신호들로 구성된 큐브는 출발 노드로부터 행선 노드까지의 기간 동안에 항상 0의 값을 가진다. 따라서, 출발 노드로부터 행선 노드까지의 기간이 결정성 신호 전이 그래프 구성 요소의 전체 사이클을 커버할 수 있도록 만들면 결정성 신호 전이 그래프 구성 요소에 대하여 항상 0의 값을 가지는 큐브들을 발견할 수 있는 것이다. 본 논문에서는 출발 노드와 행선 노드를 이와 같이 만들기 위하여 결정성 신호 전이 그래프 구성 요소를 두 사이클의 비순환 그래프로 펼친다. 그 다음에는 펼쳐진 그래프에 대하여 우선 순위 그래프를 생성하고 탐색함으로써 추가 신호들을 발견한다. 원래의 결정성 신호 전이 그래프 구성 요소에 전이 s^* 가 있을 때에, 펼쳐진 그래프의 첫 번째 사이클에서의 s^* 를 $\langle s^*, 0 \rangle$ 로 나타

내고 두 번째 사이클에서의 s^* 를 $\langle s^*, 1 \rangle$ 로 나타내기로 하자. 펼쳐진 그래프에 대한 우선 순위 그래프에서 $\langle s^*, 0 \rangle$ 와 $\langle s^*, 1 \rangle$ 에 각각 출발 노드와 행선 노드로 선택하면 그 사이의 경로는 결정성 신호 전이 그래프 구성 요소의 전체 사이클을 커버하게 된다. 따라서, 첫 번째 사이클의 전이들 중에서 해당 신호가 추가 신호로 사용될 수 있는 전이를 출발 노드로 선택하고 그 전이에 대한 두 번째 사이클의 전이를 행선 노드로 선택한다. 전이 $\langle s^*, 0 \rangle$ 가 출발 노드로 선택되고 전이 $\langle s^*, 1 \rangle$ 가 행선 노드로 선택되기 위해서는 해당 신호 (s^* 가 상향 전이이면 \bar{s} 이고 s^* 가 하향 전이이면 s)가 구간 네트워크의 추가 신호로 사용될 수 있어야 한다. 신호 s 또는 \bar{s} 가 추가 신호로 사용될 수 있기 위해서는 U 에 속한 각 구간의 시작 전이가 활성화 될 때에 그 신호가 1의 값을 가져야 하고 그 구간의 시작 전이가 발생될 때까지는 그 값을 유지해야 한다. 이를 위해서는 s^* 의 한 역방향 전이가 구간의 시작 전이와 시간 제약 인과 관계를 가져야 하고 신호 s 의 어떠한 전이도 구간의 시작 전이와 시간 제약 병렬 관계를 갖지 않아야 한다. 출발 노드와 행선 노드를 생성한 다음에는 노드들과 아크들을 삽입함으로써 우선 순위 그래프를 확장해 나간다. 우선 순위 그래프를 확장해 나가는 기본 원리는 앞 절과 같다. 여기에서는 펼쳐진 그래프에 대해서 우선 순위 그래프를 확장하는 것이 다른 점이다.

3) U 에 속한 구간들을 포함하지 않는 결정성 신호 전이 그래프

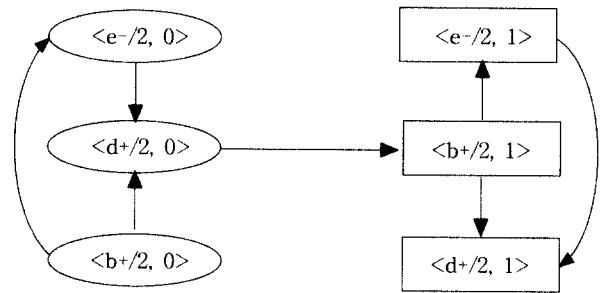
이러한 결정성 신호 전이 그래프에 대해서는 구간 네트워크는 항상 0의 값을 가지거나 항상 1의 값을 가져야 한다. 항상 0의 값을 가지도록 하는 추가 신호는 앞 절과 동일한 방법에 의해 구한다. 결정성 신호 전이 그래프에 대하여 구간 네트워크가 항상 1의 값을 가지기 위해서는 구간 네트워크의 모든 신호들이 결정성 신호 전이 그래프 전체에 대하여 항상 1의 값을 가져야 한다. 결정성 신호 전이 그래프 G 에 대하여 항상 1의 값을 가지는 신호인 1-안정 신호는 다음과 같이 정의된다.

정의 10: 결정성 신호 전이 그래프 G 에 대하여 다음의 두 조건을 만족하는 신호 s 를 1-안정 신호라 한다.

- i) G 의 각 전이 t^* 는 신호 s 의 상향 전이들 중의 한 전이와 $s+ / i \Rightarrow t^*$ 관계를 갖는다.
- ii) U 에 속하는 각 구간의 시작 전이 u^* / i 는 신호 s 의 상향 전이들 중의 한 전이와 $s+ / k \Rightarrow u^* / i$ 의 관계를 갖고 신호 s 의 어떠한 전이와도 병렬 관계를 갖지 않는다.

정의 10과 유사하게 신호 \bar{s} 도 1-안정 신호가 될 수 있다. 구간 네트워크가 1의 값을 갖도록 하기 위해서는 구간 네트

워크의 여기 신호들과 추가 신호들이 1-안정 신호이어야 한다. 그런데, 구간 네트워크에 추가되는 신호는 여러 결정성 신호 전이 그래프에 의하여 결정되므로 여기에서는 일단 1-안정 신호를 모두 구한 다음에 나중에 각 결정성 신호 전이 그래프에 대한 추가 신호들의 집합으로부터 전체적인 추가 신호를 구할 때에 모든 추가 신호들이 1-안정 신호인 경우가 있는지를 검사한다. 위에서 설명한 바와 같이 각 결정성 신호 전이 그래프에 대하여 필요한 추가 신호들의 집합을 구한 다음에 이들로부터 전체적으로 필요한 최소한의 추가 신호를 구한다.



(그림 7) 구간 집합 $U = \{c+ \mapsto c-\}$ 에 대한 DSTG-2의 우선 순위 그래프

(그림 1)의 신호 전이 그래프에서 $U = \{c+ \mapsto c-\}$ 일 때에 DSTG-2는 신호 c 의 전이를 포함하지 않으므로 해당 구간 네트워크는 항상 1의 값을 가지거나 0의 값을 가져야 한다. 이 구간 네트워크의 초기 큐브는 $I = \bar{d}$ 이고 DSTG-1에 대한 추가 신호 집합은 앞에서 설명한 바와 같이 $E_{DSTG_1} = \{\bar{b}, e\}$ 이었다. DSTG-2의 모든 전이 s^* 가 $c- \Rightarrow s^*$ 의 관계를 가지고 $c- \Rightarrow c+$ 의 관계를 가지므로 정의 10의 조건이 만족되어 이 구간에 대한 1-안정 신호는 \bar{c} 인데, 신호 \bar{d}, \bar{b}, e 가 1-안정 신호가 아니므로 이 구간 네트워크는 항상 1의 값을 가질 수 없게 된다. 따라서 항상 0의 값을 갖도록 하는 신호를 찾아야 한다. DSTG-2의 모든 전이 s^* 가 $a- \Rightarrow s^*$ 의 관계를 가지고 $a+ \Rightarrow c+$ 의 관계를 가지므로 정의 9의 조건이 만족되어 이 구간에 대한 0-안정 신호는 a 이다. 다음에는 우선 순위 그래프를 생성하여 탐색하는데, 이 구간에 대한 우선 순위 그래프가 (그림 7)에 나타나 있다. 전이 $b- / 1, d- / 1, e+ / 1$ 이 전이 $c+$ 와 시간 제약 인과 관계를 가지므로 $\langle b+ / 2, 0 \rangle, \langle d+ / 2, 0 \rangle, \langle e- / 2, 0 \rangle$ 가 출발 노드가 되고 $\langle b+ / 2, 1 \rangle, \langle d+ / 2, 1 \rangle, \langle e- / 2, 1 \rangle$ 이 행선 노드가 된다. 노드 사이의 아크는 알고리즘의 확장 조건에 따라 추가된다. 예를 들어, 전이 $e- / 2$ 와 전이 $e+ / 1$ 사이에 발생하는 전이 $d+ / 2$ 와 $b- / 1$ 중에 전이 $d+ / 2$ 의 역방향 전이가 전이 $c+$ 와 시간 제약 인과 관계를 가지므로 알고리즘의 조건을 만족하여 $\langle e- / 2, 0 \rangle$ 노드로부터 $\langle b+ / 2, 1 \rangle$ 노드로의 아크가 첨가된다.

이 그래프 탐색에서 최단 경로는 $\langle b+2, 0 \rangle \rightarrow \langle d+2, 0 \rangle \rightarrow \langle b+2, 1 \rangle$ 이므로 추가 신호는 \overline{b} 이다. 따라서 DSTG-2에 대한 추가 신호는 $E_{DSTG-2} = \{a, \overline{b}\}$ 이다. $I, E_{DSTG-1}, E_{DSTG-2}$ 로부터 최소의 구간 네트워크는 $\overline{b} \overline{d}$ 이 된다.

3.5 다중 큐브 구간 네트워크 합성

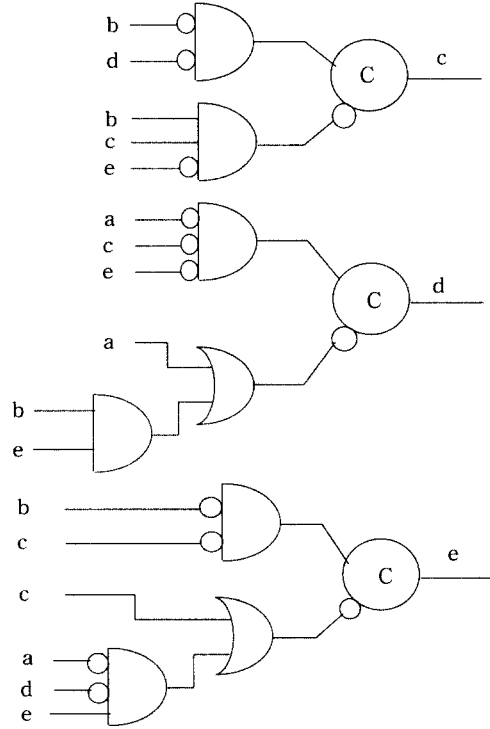
본 논문의 알고리즘은 조합형 네트워크를 발견함으로써 회로의 성능을 향상시킨다. 앞 단계의 단일 큐브 형태의 구간 네트워크 합성 알고리즘에서는 같은 수의 추가 신호들의 경우가 여러 개 있을 때에 조합형으로 만들어 주는 것을 우선적으로 선택하였다. 이 절에서는 앞 단계에서 합성한 단일 큐브 네트워크가 조합형이 아닌 경우에 그 구간 네트워크와 추가의 신호들을 OR 게이트로 연결하여 조합형 네트워크를 합성하는 알고리즘을 설명한다. 구간 네트워크가 조합형이라는 것은 구간네트워크의 출력이 $u*/i$ 전이가 활성화 될 때에 1이 되고 $\overline{u*/i}$ 전이가 활성화될 때에 0이 된다는 것을

의미한다. 따라서, 앞 단계에서 구한 구간 네트워크가 조합형이 아니라는 것은 구간 네트워크의 출력이 빨리 0이 된다는 것을 의미한다. 따라서 OR게이트를 추가하여 출력이 1이 되는 기간을 $\overline{u*/i}$ 전이가 활성화되는 때까지 연장시키는 것이다.

```

find_a_multi_cube_interval_network(STG G, transition u*/i, cube C)
{
    /* 우선 순위 그래프 생성 */
    <V, E> = <∅, ∅>
    /* 출발 노드와 행선 노드 결정 */
    Foreach s* in G
        if (occurrence_of_rising_and_falling_transition(s) = 1 and
            u*/i ⇒ s* and s* ⇒ u*/i and (t*/j ⇒ s* and
            not (t*/j || s*) for all t*/j, 여기에서 전이 t*/j는
            큐브 C의 상향 전이를 일으키는 전이임))
            SN = SN ∪ s*
    Foreach s*/j in G
        if (Is_a_non_redundant_trigger_transition(s*/j, u*/i))
            DN = DN ∪ {s*/j-1}
    V = SN ∪ DN
    /* 우선 순위 그래프 확장 */
    Foreach unprocessed node s* in V
        Foreach t* in G
            if (occurrence_of_rising_and_falling_transition(t) = 1 and
                (s* || t* or s* ⇒ t*) and t* ⇒ s* and
                t* ⇒ u*/i and u*/i ⇒ t*)
                V = V ∪ {t*}
                E = E ∪ {(s*, t*)}
    Foreach si ∈ SN
        Foreach dj ∈ DN
            Ei,j = Find_extra_signals(si, dj)
    Find_a_minimal_extra_signal_set(E)
}
    
```

(그림 8) 다중 큐브 조합형 네트워크 합성 알고리즘



(그림 9) 최종 합성 결과

(그림 8)에는 조합형 네트워크의 추가 신호를 구하는 알고리즘이 나타나 있다. 여기에서도 앞 단계에서와 유사하게 우선 순위 그래프를 이용하여 추가의 신호들을 구한다. 이 알고리즘은 앞 단계의 알고리즘과 출발 노드와 행선 노드를 구하는 조건이 다르고 그래프에서 경로에 s+ 전이가 있으면 추가되는 신호가 s가 되는 것이 다르다. 추가되는 신호의 하나는 앞 단계에서 구한 구간 네트워크의 출력이 0이 되기 전에 1이 되어야 한다. 그리고 구간의 마지막 전이가 활성화 될 때까지 이 둘 중의 적어도 한 신호의 값은 1의 값을 가져야 한다. 또한 추가되는 신호의 값은 이 구간 내에서만 1이 되어야 한다. 따라서 상향 전이와 하향 전이가 한 번만 일어나는 신호의 전이만을 고려한다. 출발 노드를 구하는 조건에서는 앞 단계에서 구한 단일 큐브 C를 0으로 만드는 전이보다 먼저 일어나는 가를 검사하는데, 이는 추가되는 신호가 구간 네트워크의 출력이 0이 되기 전에 1이 되도록 하기 위해서이다. 앞 단계에서는 $u*/i$ 전이가 활성화 될 때까지 0으로 유지되는 신호를 찾았기 때문에 행선 노드를 $u*/i$ 전이로부터 구했지만 여기에서는 $\overline{u*/i}$ 전이가 활성화 될 때까

지 1로 유지되는 신호를 찾기 때문에 $\overline{u*/i}$ 전이로부터 행선 노드를 구한다.

(그림 1)의 신호 전이 그래프에 대해서는 다중 큐브 구간 네트워크는 존재하지 않는다. (그림 9)에는 (그림 1)의 신호 전이 그래프에 대한 합성 결과가 나타나 있다.

4. 실험 결과

<표 1>에는 벤치마크들에 대한 실험 결과가 나타나 있다. 여기에서는 본 논문의 합성 결과와 참고문헌 [9]의 합성 결과를 리터럴의 수와 CPU 시간을 기준으로 비교하였다. 본 논문에서 제안한 합성 방법을 기존의 다양한 벤치마크들에 적용하여 실험을 수행하였다. 본 논문에서 사용한 벤치마크에는 통신용 칩의 제어기[12], 바코드 생성 제어기[13], CD 플레이어 제어기[14], 캐쉬 메모리 제어기[15], SCSI 인터페이스 제어기[16], 메모리 접근 제어기[16] 등 실제로 설계된 칩

에 포함되어 있는 제어 회로들이 포함되어 있다. 실험 결과에 의하면 큰 상태 공간을 가지는 회로에 대해서는 본 논문에서 제안하는 알고리즘이 합성 시간을 현저히 단축시킬 수 있다는 것을 알 수 있다. 표의 신호 전이 그래프 열에서 S는 신호 수, N은 노드 수, A는 아크 수를 나타낸다. 그리고 다른 열에서 N/A는 합성에 실패하거나 구할 수 없는 경우를 나타낸다. 표 1의 회로들은 SCSI 제어기와 master-read를 제외하고는 모두 선택행위를 포함하는 회로로서 기존의 직접 합성 방법 [10, 11]으로는 합성이 불가능하다.

본 논문의 합성 알고리즘은 C 언어를 사용하여 구현하였는데, <표 1>의 실험 결과는 운영체제가 Linux이며 주메모리가 384MB 이고 스왑 메모리가 700MB인 400MHz 펜티엄 II 컴퓨터에서 얻어진 것이다. 기존의 방법이 상태수가 큰 회로에 대하여 합성에 실패한 반면에 본 논문의 방법은 빠른 시간 내에 합성할 수 있다는 것이 주목할 만한 일이다. 그리고 속도가 빠르면서도 합성된 회로의 면적이 대부분 같다는

<표 1> 실험 결과 : ATACS와의 비교

| 회 로 | 신호전이그래프 (S / N / A) | 상 태 수 | ATACS | | 본 논문 | |
|-------------------|------------------------|--------|-------|---------|------|---------|
| | | | 리터럴수 | CPU 시간 | 리터럴수 | CPU 시간 |
| DME | 9/30/34 | 28 | 39 | 0.14 | 34 | 0.26 |
| cache-ctrl | 36/369/1098 | N/A | N/A | N/A | 811 | 390.30 |
| cache-ctrl-s2io1 | 52/738/2196 | N/A | N/A | N/A | 1061 | 4944.34 |
| cache-ctrl-s2io2 | 72/738/2196 | N/A | N/A | N/A | 1622 | 3190.09 |
| dff | 4/18/24 | 18 | 16 | 0.05 | 16 | 0.09 |
| dram-ctrl | 13/50/97 | 107 | 46 | 4.86 | 46 | 0.50 |
| elatchX | 26/52/103 | 9475 | 44 | 105.5 | 45 | 1.33 |
| factorial | 17/48/91 | 72 | 45 | 0.67 | 45 | 0.49 |
| hp-ir-it-ctrl | 13/47/75 | 71 | 60 | 0.45 | 60 | 0.66 |
| hp-ir-two-ticks | 6/23/33 | 26 | 8 | 0.10 | 7 | 0.09 |
| master-read | 18/28/40 | 2108 | 34 | 3.38 | 34 | 0.15 |
| po-alloc-outbound | 9/22/23 | 21 | 17 | 0.12 | 17 | 0.15 |
| po-rcv-setup | 5/15/20 | 14 | 8 | 0.05 | 8 | 0.04 |
| po-sbuf-read-ctrl | 8/24/29 | 20 | 13 | 0.07 | 13 | 0.09 |
| po-sbuf-send-ctrl | 8/24/29 | 33 | 36 | 0.14 | 36 | 0.16 |
| po-sbuf-send-pkt2 | 9/27/35 | 28 | 23 | 0.11 | 23 | 0.21 |
| p SCSI-ircv | 9/28/48 | 44 | 27 | 0.19 | 27 | 0.12 |
| p SCSI-isend | 10/49/81 | 74 | 59 | 0.41 | 59 | 1.02 |
| p SCSI-trcv | 8/26/39 | 35 | 23 | 0.12 | 23 | 0.09 |
| p SCSI-tsend | 10/45/72 | 65 | 55 | 0.34 | 55 | 0.81 |
| rlm | 6/13/14 | 12 | 7 | 0.04 | 7 | 0.05 |
| SCSI ctrl | 5/10/17 | 16 | 10 | 0.02 | 10 | 0.02 |
| 4 SCSI | 14/28/62 | 806 | 40 | 1.17 | 40 | 0.22 |
| 8 SCSI | 25/52/122 | 404006 | 80 | 4937.36 | 80 | 1.29 |
| 9 SCSI | 29/58/137 | N/A | N/A | N/A | 90 | 1.96 |
| sm | 3/10/12 | 8 | 7 | 0.06 | 7 | 0.03 |
| vme-master | 12/52/75 | 162 | 31 | 0.92 | 31 | 0.52 |
| vme-read | 13/34/46 | 124 | 25 | 0.48 | 24 | 0.28 |
| vme-write | 15/36/53 | 243 | 29 | 1.91 | 28 | 0.37 |

것도 주목할 만한 일이다. 제안된 방법의 회로의 면적이 작은 경우는 본 논문에서는 다중 큐브 형태의 조합형 네트워크를 합성하는 것에 반하여 기존의 방법에서는 이러한 회로를 합성하지 않기 때문이다. 기존의 방법이 보다 작은 회로를 합성하는 경우는 한 신호에 대하여 여러 구간이 존재할 경우에 여러 구간을 동시에 고려하는 것에 반하여 본 논문의 방법에서는 한 구간의 합성 시에 다른 구간을 고려하지 않기 때문이다. 이러한 두 경우는 드물었고 대부분의 경우에는 두 합성 방법이 같은 면적의 회로를 합성하였다. 본 논문에서 제안한 방법에 의하여 합성된 회로가 해저드가 없다는 것을 재확인하기 위하여 기존의 검증기[19]를 사용하였다. 검증기를 사용하여 회로의 동작을 확인한 결과 아무런 문제없이 올바르게 작동하는 것을 알 수 있었다.

5. 결 론

본 논문에서는 시간 제한 조건을 가진 신호 전이 그래프로부터 비동기 회로를 합성하는 방법을 제안하였다. 기존의 방법들이 지수승의 데이터 복잡도를 갖는 상태 그래프를 이용하는 반면에 본 논문에서는 신호 전이들의 관계를 이용하여 회로를 합성할 수 있음을 보였다. 그리고 필요한 신호 전이들의 관계는 타이밍 분석과 그래프 탐색에 의하여 구할 수 있음을 보였다. 본 논문의 실험 결과는 본 논문에서 제안하는 방법이 기존의 방법들의 문제점중의 하나인 상태 수의 급증 문제를 극복할 수 있음을 보였다. 또한 시스템 단계의 큰 회로를 다루는 상위 단계의 합성에서는 대규모의 신호 전이 그래프를 합성하는 것이 요구되기 때문에 본 논문의 합성 알고리즘은 상위 단계 합성 알고리즘 개발에 효과적으로 활용될 수 있을 것이다.

참 고 문 헌

[1] T. A. Chu, "Synthesis of Self-timed VLSI Circuits from Graph Theoretic Specifications," *Ph.D. Thesis*, Massachusetts Institute of Technology, 1987.
 [2] T. H.-Y. Meng, R. W. Broderson, and D. G. Messerschmitt, "Automatic synthesis of asynchronous circuits from high-level specifications," *IEEE Transactions on Computer-Aided Design*, Vol.8, No.11, pp.1185-1205, Nov. 1989
 [3] P. A. Beerel and T. H.-Y. Meng, "Automatic Gate-Level Synthesis of Speed-independent Circuits," *Proceedings of International Conference on Computer Aided Design*, pp.581-586, Nov. 1992.
 [4] A. Kondratyev, M. Kishinevsky, B. Lin, P. Vanvekbergen, and Yakovlev, "Basic Gate Implementation of Speed-independent Circuits," *In Proceedings of Design Automation Conference*, pp.56-62 Jun. 1994.

[5] V. I. Varshavsky, V. B. Marakhovshy, and V. V. Smolensky, "Designing self-timed devices using the finite automaton model," *IEEE Design & Test of Computers*, Vol.12, No.1, pp.14-23, Spring, 1995.
 [6] K. J. Lin, C. H. Kuo, and C. S. Lin, "Synthesis of Hazard-free Asynchronous Circuits Based on Characteristic Graph," *IEEE Transactions on Computers*, Vol.46, No.11, Nov. 1997.
 [7] E. Pastor, J. Cortadella, A. Kondratyev, and O. Roig, "Structural Methods for the Synthesis of Speed-Independent Circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol.17, No.11, pp.1108-1129, Nov. 1998.
 [8] L. Lavagno, K. Keutzer, and A. Sangiovanni-Vincentelli, "Algorithms for Synthesis of Hazard-Free Asynchronous Circuits," *Proceedings of the 28th Design Automation Conference*, 1991.
 [9] C. J. Myers, T. G. Rokicki, T. H.-Y. Meng, "POSET Timing and Its Application to the Synthesis and Verification of Gate-Level Timed Circuits," *IEEE Transactions on Computer-Aided Design*, 18(6), pp.769-786, June, 1999.
 [10] S. T. Jung and C. J. Myers, "Direct Synthesis of Timed Asynchronous Circuits," *Proc. International Conf. Computer-Aided Design (ICCAD)*, pp.332-337, November, 1999.
 [11] 김희숙, 정성태, "시간 제한 조건을 가진 결정성 신호 전이 그래프로부터 비동기 회로의 합성", *정보과학회논문지: 시스템 및 이론*, 제27권, 제2호, pp.216-226, 2000.
 [12] K. S. Stevens, S. V. Robinson, and A. L. Davis, "The Post Office - Communication Support for Distributed Ensemble Architectures," *In Proceeding of 6th International Conference on Distributed Computing Systems*, pp.567-571, 1986.
 [13] P. R. Panda and N. Dutt, "1995 High Level Synthesis Repository," *Tech. Rep. 95-04*, University of California, Irvine, U.S.A, 1995.
 [14] J. Kessels, K. van Berkel, R. Burgess, M. Roncken, and F. Schalij, "An Error Decoder for the Compact Disc Player as an Example of VLSI Programming," *Tech. Rep.*, Philips Research Laboratories, Eindhoven, The Netherlands, 1992.
 [15] S. M. Nowick, M. E. Dean, D. L. Dill, and M. Horowitz, "The Design of a High-Performance Cache Controller : A Case Study in Asynchronous Synthesis," *Integration, the VLSI Journal*, Vol.15, No.3, pp.241-262, Oct. 1993.
 [16] S. M. Nowick, K. Y. Yun, D. L. Dill, "Practical Asynchronous Controller Design," *in Proc. International Conference on Computer Design(ICCD)*, pp.341-345, Oct. 1992.
 [17] T. Murata, "Petri nets : Properties, Analysis and Applications," *Proceedings of IEEE*, Vol.77, No.4, pp.541-580, Apr. 1989.
 [18] C. J. Myers, T. G. Rokicki, and T. H.-Y. Meng, "POSET timing and its application to the synthesis and verification of gate-level timed circuits," *In IEEE Transactions on CAD*, Vol.18 No.6, pp.769-786, Jun. 1999.
 [19] T. G. Rokicki and C. J. Myers, "Automatic Verification of Timed Circuits," *In Proceedings of the conference on Computer Aided Verification*, Jun. 1994.



정 성 태

e-mail : stjung@wonkwang.ac.kr
1987년 서울대학교 컴퓨터공학과 졸업
1989년 서울대학교 컴퓨터공학과 석사학위
취득
1994년 서울대학교 컴퓨터공학과 박사학위
취득

1994년~1995년 한국전자통신연구소 박사후연수연구원
1999년~1999년 미국 Univ. of Utah 과학재단지원 해외 Post-Doc.
1995년~현재 원광대학교 컴퓨터및정보통신공학부 교수
관심분야 : VLSI / CAD, 영상 인식, 영상 기반 렌더링, 컴퓨터
그래픽스



정 석 태

e-mail : stjoung@wonkwang.ac.kr
1989년 전남대학교 전산학과 졸업
1996년 스쿠바대학 이공학연구과 석사학위
취득
2000년 스쿠바대학 공학연구과 박사학위
취득

2001년~현재 원광대학교 컴퓨터및정보통신공학부 전임강사
관심분야 : 공간 파서 생성기, 비주얼 시스템, 오감 정보통신