

변형기법을 이용한 비동기 시스템의 상위수준 합성기법

유동훈, 이동익

광주과학기술원 정보통신공학과

전화: 062-970-2267 / 핸드폰: 016-229-0521

High-Level Synthesis for Asynchronous Systems using Transformational Approaches

Dong-Hoon Yoo and Dong-Ik Lee

Dept. of Info. & Comm., K-JIST

Email: dhyoo@kjist.ac.kr

Abstract

Although asynchronous designs have become a promising way to develop complex modern digital systems, there is a few complete design framework for VLSI designers who wish to use automatic CAD tools. Especially, high-level synthesis is not widely concerned until now.

In this paper we proposed a method for high-level synthesis of asynchronous systems as a part of an asynchronous design framework. Our method performs scheduling, allocation, and binding, which are three subtasks of high-level synthesis, in simultaneous using a transformational approach. To deal with complexity of high-level synthesis we use neighborhood search algorithm such as Tabu search.

I. 서론

최근 저전력 소모, 클럭 분배 문제의 해결, 평균적인 계산시간, 모듈성(modularity), 견고성, EMI 문제의 해결 등의 장점을 가진 비동기 시스템의 설계 방법에 대해 많은 연구가 이루어지고 있다. 목적 시스템을 최적화하는데 있어서 상위수준합성은 여러 설계 단계 중에서 가장 중요한 단계이며, 따라서 많은 상위수준 합성 기법이 동기식 설계를 위해 연구되어 왔다. 그러나, 기존의 동기식 설계에 사용되어 왔던 상위수준 합성기법은 비동기식 설계에는 적합하지 않다. 이는 기존의 방법들이 동기식에서의 클럭 주기를 이용한 알고리즘을

사용하기 때문이다. 비동기식 설계에서는 각각의 기능 모듈이 입력 데이터에 의존한 지연시간을 가지므로, 이를 이용하여 설계자가 원하는 최적의 결과를 얻어낼 수 있는 상위수준 합성기법을 개발할 필요가 있다.

이를 위해 [1]과 [2]에서는 비동기 시스템을 위한 상위수준 합성기법을 제안하였다. [1]은 리스트 스케줄링(list scheduling)을 변형한 방법을 이용하였고, [2]는 데이터 흐름도에 순서를 정의하는 아크를 삽입함으로써 스케줄링을 수행하였다. 하지만, 두 방법 모두 최적화에 있어 제어흐름을 고려하지 않았으며, 목적 시스템의 복잡도가 커질 경우 효과적인 수행이 어렵다는 단점을 가지고 있다.

본 논문은 변형기법을 이용하여 상위수준합성의 세 단계인 스케줄링, 자원할당, 바인딩을 동시에 해결하는 비동기식 상위수준 합성기법을 제안한다. 클럭 주기를 사용하지 않는 비동기식 구조수준합성을 하기 위해서는 먼저 입력으로 주어진 HDL(Hardware Description Language)을 내부적으로 표현할 설계표현식(design representation)을 정의하고, HDL로부터 생성된 설계표현식에 여러 가지 변형동작을 가함으로써 설계자가 요구하는 결과를 얻어낸다. 제어흐름과 데이터흐름을 동시에 표현하는 설계표현식을 변형시키면서 최적의 결과를 얻어내는 과정은 해법공간(solution space)에서 최적(optimal)의 해를 탐색하는 방법과 동일하다. 이러한 탐색을 효율적으로 수행하기 위해 본 연구에서는 탐색 알고리즘의 일종인 Tabu Search를 이용한다.

본 논문의 2장에서는 비동기 시스템을 위한 상위수준합성의 특성을 알아보고, 3장에서는 설계정보를 표현하고 분석하기 위한 설계표현식에 대해 설명한다. 4장에서는 3장에서 정의한 설계표현식을 이용하여 설계공간을 탐색하는 변형기법에 대해 설명하며, 마지막으로

5장에서 요약과 함께 결론을 맺는다.

II. 상위수준합성(High-Level Synthesis)

상위수준합성은 스케줄링(scheduling), 자원할당(allocation), 바인딩(binding)의 작업들로 이루어진다. 동기식 설계에서의 스케줄링은 각각의 연산을 모든 연산들의 최악지연시간으로 정의되는 제어단계(control step)에 할당하는 작업이다. 이에 반해 자원할당과 바인딩은 주어진 설계양식에 대응하는 하드웨어 자원을 선택하고 할당하는 작업들이다. 상위수준합성에서 가장 중요한 작업은 스케줄링이며, 주로 자원할당과 바인딩은 스케줄링의 결과를 기반으로 하여 이루어진다.

비동기 시스템의 스케줄링에서 사용하는 시간모델(timing model)은 동기식에 대하여 다음과 같은 차이점을 가진다.

- 시간(time)은 연속적(continuous)이다. 따라서, 각 연산의 시작과 끝은 아무때나 일어날 가능성이 있는 사건(event)로 여겨진다.
- 각각의 연산은 변동적이며 데이터에 의존한 지연시간(delay)을 갖는다.

따라서, 비동기식 설계에서의 스케줄링은 연산의 제어 단계에 따라 결정되는 것이 아니라 연산간의 부분순서(partial ordering)에 의해 결정되어진다. 다시 말해서, 비동기식 설계에서의 스케줄링은 연산간의 순서를 결정하는 작업으로 정의된다.

연산의 순서를 결정하기 위해서는 우선 연산간의 데이터 의존성을 검사하여야 한다. 두 연산간의 데이터 의존성이 있을 경우에 이 두 연산을 동시에 수행할 수 없으므로 만약 이 두 연산이 동일한 유형의 연산이라면 하나의 데이터 모듈을 공유하도록 할 수 있다. 또한, 두 연산간에 데이터 의존성이 없을 경우에는 각각의 연산은 서로 다른 데이터 모듈을 사용하여 수행되어야만 한다. 전자의 경우, 두 연산 중 한 연산의 수행이 끝난 다음에 나머지 연산을 수행할 수 있으므로 이 두 연산은 직렬순서(sequential order)관계를 갖는다. 후자의 경우, 각각의 연산을 다른 연산의 수행 결과와 상관없이 독립적으로 수행할 수 있으므로 병렬순서(parallel order)관계를 갖는다.

직렬순서관계에 있는 연산은 병렬순서관계에 있는 연산보다 더 많은 연산시간을 가지지만 더 적은 데이터 모듈을 사용한다. 반대로, 병렬순서관계에 있는 연산은 직렬순서관계에 있는 연산보다 더 적은 연산시간을 가지지만 더 많은 데이터 모듈을 사용한다. 이처럼 의존성 검사를 통해 연산간의 순서관계를 정할 수 있으며, 병렬순서관계에 있는 연산을 의존성을 파괴하지 않는 범위에서 직렬순서관계로 변형하거나, 혹은 직렬순서관계에 있는 연산을 병렬순서관계로 변형함으로써 설계자가 원하는 스케줄링의 결과를 얻을 수 있다.

이러한 순서관계를 결정하기 위한 방법으로는 생성

가능한 모든 탐색공간을 탐색하는 완전탐색(exact search)과 탐색공간을 효율적으로 줄여서 탐색하는 경험적탐색(heuristic search)이 있다. 완전탐색방법은 항상 최적의 결과를 얻을 수 있으나, 탐색공간이 커질 경우에는 매우 비효율적이라는 단점을 가지고 있다. 반면, 경험적탐색방법은 항상 최적의 결과를 얻을 수 있다는 보장은 없지만, 큰 탐색공간을 효과적으로 탐색할 수 있다는 장점을 가지고 있다. 설계하고자 하는 시스템의 크기가 점점 커져 가고 있는 요즘의 설계환경에서는 큰 탐색공간을 효과적으로 탐색할 수 있는 경험적탐색방법을 이용하는 것이 더 나은 선택이라고 할 수 있겠다.

III. 설계표현식(Design Representation)

대부분의 VLSI 설계도구에서는 주어진 설계대상의 효과적인 분석과 합성을 위하여 VHDL이나 Verilog 등과 같은 HDL로 기술되어진 입력사양을 설계표현식으로 변환한다. 효과적인 분석을 위하여 대부분의 설계표현식은 그래프의 형태로 나타내어지며, 입력사양에 주어진 모든 정보를 표현할 수 있어야 한다. 또한, 효과적인 합성을 위해 설계표현식은 상위수준의 기술에서부터 상호수준의 기술까지 다양한 수준의 표현식을 나타낼 수 있어야 한다.

현재 가장 널리 사용되고 있는 설계표현식으로는 데이터흐름도(data flow graph: DFG)와 제어/데이터흐름도(control/data flow graph: CDFG)가 있다[3]. DFG에서 각각의 연산은 그래프의 노드로 표현하며, 데이터의 흐름은 아크로 표현한다. CDFG에서는 데이터의 연산에 관련된 부분은 DFG로 표현하고, 제어에 관련된 부분은 CFG(control-flow graph)로 표현한다. 기본적인 DFG는 비주기적(acyclic)이어서 반복수행(loop)을 표현하기에는 적합하지 않다. 이를 보완하기 위해 [4]에서 각각의 반복수행을 하위계층의 그래프로 표현하는 계층적인 DFG를 제안하였다. CDFG는 제어흐름을 관장하는 노드들을 이용하여 반복수행이나 선택수행(branch) 등을 표현하고 있으나 아직까지 정형화된 형태를 가지지 못하고 있다.

페트리 넷[5]은 비동기식 설계에 있어서 설계표현식으로 가장 널리 사용되고 있으며 정형화된 형태를 가지고 있다. 페트리 넷은 장소(place)와 변이(transition)의 노드들과 그들의 연결로 구성되며, 병행시스템을 기술하기에 적합한 도구로 알려져 있다. 페트리 넷을 설계표현식으로 이용하여 DFG나 CDFG의 단점을 보완할 수 있으며, 근본적으로 병행성을 내재하고 있는 비동기 시스템을 기술하거나 표현하는데 있어 페트리 넷이 적합한 도구라고 할 수 있겠다. 또한, 페트리 넷은 노드의 연결관계에 따라 직렬순서관계와 병렬순서관계를 명확히 표현하기 때문에 연산의 순서관계에 의해 스케줄링이 결정되는 비동기식 설계에 더욱 적합하다고 할 수 있다. 그러나, 신호전이그래프(signal transition graph:

```

while x < a do
  x1 := x + dx;
  u1 := u - (3*x*u*dx) - (3*y*dx);
  y1 := y + (u * dx);
  x := x1; u := u1; y := y1;
end
    
```

그림 1. 미분방정식 해법기의 기술

STG) 등과 같은 페트리 넷의 변형을 제어회로를 기술하는 도구로 사용한 연구는 있지만, 비동기 시스템의 상위수준합성에 페트리 넷을 사용하는 예는 전무한 형편이다.

Zebo Peng은 [6]에서 동기식 상위수준합성을 위한 설계표현식으로 ETPN(Extended Timed Petri Net)을 제안하였다. ETPN은 일반적인 페트리 넷을 시간과 조건적 선택(conditional choice)의 표현이 가능하도록 확장한 것으로서 각 장소들은 데이터 패스의 한 모듈에 대응하며 그 모듈의 지연시간을 포함한다. Zebo Peng이 제안한 상위수준합성 방법에서는 ETPN의 모든 장소들이 동기식 시스템의 한 클럭 주기 만큼의 시간을 갖는다고 가정한다. ETPN의 장점은 페트리 넷을 이용하여 제어회로를 표현할 뿐만 아니라, 이와 동시에 DFG의 변종을 이용하여 제어회로에 대응하는 데이터 패스도 효율적으로 표현할 수 있다는 데 있다.

본 논문에서는 설계표현식으로 ETPN을 비동기식 설계에 맞도록 변형하여 사용한다. Zebo Peng의 방법에서처럼 ETPN의 각 장소는 한 클럭 주기 만큼의 지연 시간을 갖는 것이 아니라, 그 장소에 대응하는 데이터 모듈의 평균지연시간을 갖는다고 가정한다. 그림 1의 미분방정식 해법기의 기술[4]을 ETPN으로 나타낸 그림이 그림 2에 나타나있다. 그림 2의 (a)는 제어회로의 동작을 ETPN으로 나타낸 그림이고, (b)는 (a)에 대응하는 DFG를 나타낸 그림이다. (a)의 ETPN에서 장소가 토큰을 갖게 되면 (b)에서 동일한 숫자로 표시된 아크를 통해 데이터가 모듈의 입력으로 들어가 작업이 수행된다.

IV. 변형기법 (Transformational Approach)

대부분의 상위수준합성 작업은 NP-complete인 문제로 알려져 있다[4]. 따라서, 이러한 복잡성을 좀더 효과적으로 다루기 위해 다음의 두 가지 방법을 고려할 수 있다.

1. 합성작업을 여러 개의 부작업(sub-task)로 나누고, 한번에 한 부작업을 수행한다.
2. 일련의 변형단계를 거쳐 합성작업을 수행한다.

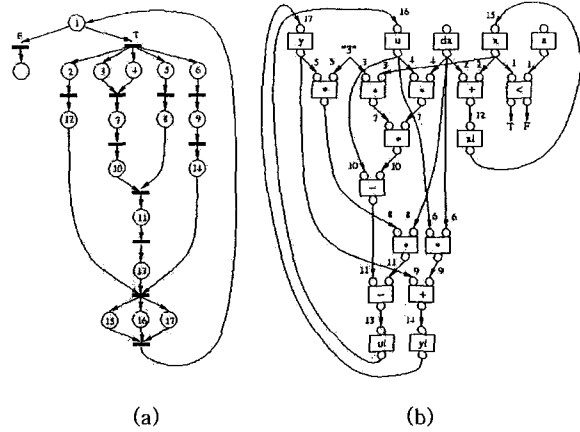


그림 2. 미분방정식 해법기의 (a) ETPN과 (b) DFG 표현

각 변형단계에서는 이전 작업에 약간의 수정을 가해 새로운 변형단계를 만들어냄으로써 점차적으로 원하는 결과에 접근해간다.

후자의 방법을 변형기법(transformational approach)이라고 한다. 전통적인 방법과 변형기법의 비교가 그림 3에 나타나있다. 전통적인 상위수준 합성기법은 행위기술(behavioral specification)으로부터 자원할당, 스케줄링, 바이딩 등을 통해 최종결과를 생성하는 반면, 변형기법은 행위기술을 일련의 맵핑(mapping)을 통해 초기의 결과를 생성하고 이를 조금씩 변형시켜가면서 최적화된 결과를 생성한다.

변형기법은 다음과 같은 장점을 갖는다.

1. 합성과정이 일련의 변형단계로 나뉘어지기 때문에, 만약 각 변형단계가 의미보전(semantic preserving)변형이라는 것이 증명되면 전체 합성작업이 반드시 올바른 결과를 생성한다는 것을 보장할 수 있다.
2. 경험적최적화방법(heuristic optimization method)을 더 효과적으로 사용할 수 있도록 해준다. 이때, 변형기법은 일종의 주변탐색(neighborhood search) 최적화 문제로 생각되어질 수 있다.
3. 각 변형단계에서의 결과가 모두 상위수준합성의

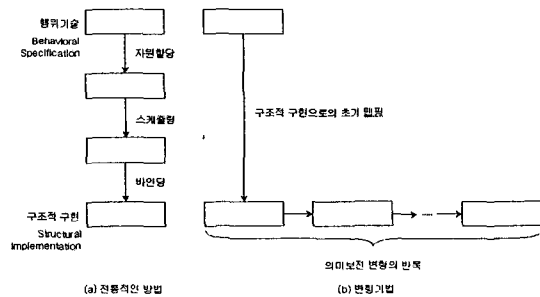


그림 3. 전통적인 방법과 변형기법의 비교

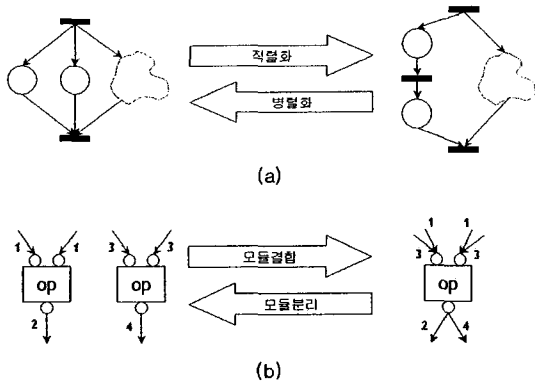


그림 4. 기본적인 변형패턴

완전한 결과로서 사용될 수 있기 때문에 설계자는 설계시의 제약조건을 만족하는 다양한 해답 중에서 자신이 원하는 결과를 선택할 수 있다.

변형의 목적은 최적의 설계(optimal design)에 도달하는 것이고, 이는 해답공간(solution space)에서 최적의 해답(optimal solution)을 찾는 것과 동일한 작업이다. 따라서, 변형기법에 Tabu search, genetic algorithm, simulated annealing 등과 같은 기존의 주변탐색 방법들을 사용할 수 있다. 본 논문에서는 Tabu search 알고리즘을 이용하여 상위수준합성을 수행하고자 한다.

주변탐색 알고리즘을 이용하기 위해서는 기본적인 변형패턴을 정의해야 한다. ETPN을 이용하여 비동기 시스템의 상위수준합성을 하기위한 기본 변형패턴으로 다음의 변형을 생각할 수 있다.

1. 직렬화(serialization): 병렬관계에 있는 두개의 장소를 직렬관계로 변환한다. 이때 시스템의 지연시간(latency)은 증가하나 모듈결합을 수행할 수 있게 되어 모듈의 개수를 줄일 수 있는 가능성을 만들어준다. 직렬화는 제어/데이터 의존성에 상관없이 사용할 수 있다. (그림 4의 (a))
2. 병렬화(parallelization): 직렬관계에 있는 두개의 장소를 병렬관계로 변환한다. 이때 시스템의 지연시간은 감소하나 모듈분리를 해야 하므로 모듈의 개수는 증가한다. 병렬화는 제어/데이터 의존성이 없는 경우에만 가능하다. (그림 4의 (a))
3. 모듈결합(module merge): 서로 다른 두개의 모듈을 하나의 모듈로 결합하는 변형으로 두 모듈의 기능이 동일할 경우에만 행해진다. 이때 시스템의 지연시간은 증가하나 사용되는 모듈의 개수는 감소한다. 모듈결합은 결합하고자 하는 두 모듈에 대응하는 장소들간에 병행성이 없을 경우에만 가능하다. (그림 4의 (b))
4. 모듈분리(module split): 하나의 모듈을 두개의 모듈로 분리하는 변형이다. 이때 시스템의 지연시간은 감소하나 사용되는 모듈의 개수는 증가한다. 모듈분리는 제어/데이터 의존성에 상관없이

사용할 수 있다. (그림 4의 (b))

HDL의 킵파일을 통해 얻어진 초기의 설계표현식을 위의 기본변형을 통해 조금씩 변형시켜 나가면서 주변탐색 알고리즘을 이용하여 최적화된 결과를 얻을 수 있다. 이를 위해서 지연시간과 하드웨어 자원에 대한 평가함수(cost function)를 정의하고, 각 변형단계에서의 구현이 설계제약조건을 만족하는지를 검사한다.

V. 결론

동기 시스템과는 달리 전역클럭을 사용하지 않는 비동기 시스템은 기존의 동기식 상위수준합성과는 다른 방법을 요구한다. 또한, 목적 시스템의 크기가 커질수록 상위수준합성의 복잡도 높은 여러 작업들을 효과적으로 수행할 수 있는 방법을 필요로 한다.

본 논문에서는 변형기법을 이용하여 비동기 시스템의 상위수준합성을 수행하는 방법을 제안하였다. 변형기법을 이용하여 상위수준합성의 세가지 작업인 스케줄링, 자원할당, 바인딩을 동시에 수행하고, 주변탐색 알고리즘인 Tabu search를 이에 적용하여 복잡도 높은 작업을 효과적으로 수행하도록 하였다.

본 논문에서는 시스템의 지연시간과 면적만을 비교한 상위수준합성을 제안하였지만, 전력소비도 함께 고려한 상위수준합성의 연구도 현재 수행 중이다.

감사의 글

본 연구는 한국과학재단의 한일국제공동연구과제(20006-302-01-2) 및 광주과학기술원 초고속 광 네트워크 연구센터를 통한 한국과학재단 우수연구센터 지원금에 의한 것입니다.

참고문헌

- [1] R. M. Badia and J. Cortadella, "High-Level Synthesis of Asynchronous Systems: Scheduling and Process Synchronization," European Design Automation Conference, Pages 70-74, February 1998.
- [2] B. M. Bachman, Architectural-Level Synthesis of Asynchronous Systems, MS thesis, University of Utah, December 1993.
- [3] Daniel Gajski and Nikil Dutt, *High-Level Synthesis: Introduction to Chip and System Design*, Kluwer Academic Publishers, 1992.
- [4] G. D. Micheli, *Synthesis and Optimization of Digital Circuits*, McGraw-Hill, 1994.
- [5] T. Murata, "Petri nets: Properties, analysis, applications," Proceedings of the IEEE, Vol. 77, No. 4, pages 541-580, 1989.
- [6] Zebo Peng, *A formal methodology for automated synthesis of VLSI systems*, PhD thesis, Dept. of Computer and Information Science, Linköping University, 1987.