

## IP 설계를 위한 설계규칙 검사기 구현

백영석, 배영환, 조한진  
한국전자통신연구원 집적회로설계연구부

전화 : 042-860-6262

### Implementation of Design Rule Checker for IP Design

Young Seok Baek, Young Hwan Bae, and Hanjin Cho  
Integrated Circuit Design Research Department,  
Electronics and Telecommunications Research Institute

E-mail : ysbaek@etri.re.kr

#### Abstract

In this paper, we address the requirement of VHDL parser for design rule checker, and the structure and implementing method of design rule checker which checks if IP design is valuable to reuse. This checker builds the grammar trees from the design rules, and the internal graphs representation from IP design data. It maps the nodes of the grammar trees and the internal graphs to check if it violates the design rules. The design rule checker can do the cross reference between source codes and error messages to find error position easy.

게자가 1~2년 사이에 수백만 게이트를 설계할 수 있어야 하며, 또한 수백만 게이트급의 설계를 검증할 수 있어야 한다. 수백만 게이트 급의 설계는 하나의 설계자가 10만 게이트정도의 설계를 하는 기존의 설계 방법론으로는 불가능하다. 이러한 문제의 해결방법이 IP를 이용한 설계이다. 하나의 설계 블록이 수천에서 수만 게이트가 되는 재사용 가능한 설계를 이용함으로써 수백만 게이트 급의 설계가 가능할 수 있다.

본 논문은 IP를 목적으로 설계된 데이터가 재사용 가능하게 설계되었는가를 검사하는 설계규칙검사의 구현에 대하여 기술하며, 설계규칙검사의 구성 및 구현 방법의 순서로 기술한다.

#### II. 설계 규칙 검사기의 구조

#### I. 서론

반도체 설계는 제조공정의 발달과 설계 요구 복잡도 증가로 인하여 수백만 게이트급의 설계가 최근 이루어지고 있다. 이러한 설계 용량의 증가는 시스템을 하나의 칩에 구현하는 이른바 SOC(System On a Chip)라는 설계 방식을 도래하게 하였다. 수백만 게이트의 설계를 구현하기 위해서는 이를 가능하게 하는 설계 방법론이 필요하게 된다. 이러한 SOC 설계 방법론은 설

계 규칙 검사기는 IP, 블록 또는 시스템 구현을 위하여 설계된 데이터를 재사용에 적합하게 설계되었는가를 검사하는 CAD 소프트웨어로서 기본적인 구조는 그림 1과 같다. VHDL 하드웨어 기술언어로 설계된 IP는 VHDL 분석기를 통하여 내부의 중간 형식으로 데이터 베이스에 저장된다. 분석과정에서 하드웨어 기술언어에 대한 구문오류 및 의미오류 등의 정보를 설계자에게 제공하여 오류를 제거하게 한다.

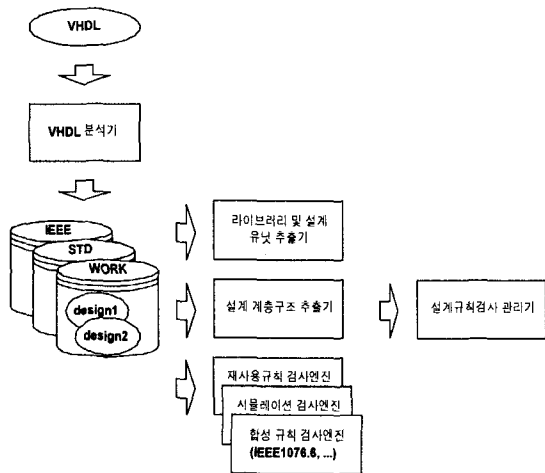


그림 1. 규칙검사기의 기본 구성도

(1) 설계규칙검사기를 위한 VHDL Parser의 요건

VHDL 언어를 사용하는 모든 CAD 소프트웨어는 VHDL parser를 사용하게 된다. Parser는 소프트웨어의 입력부분을 제어하므로 설계규칙검사기의 지원 능력에 영향을 미칠 수 있다. 일반적으로 VHDL parser가 지원하고 있는 부분 이외에 설계 규칙검사기를 위하여 parser가 가져야 하는 요건에 몇 가지에 대하여 기술한다.

(가) 대문자 소문자의 구분가능

VHDL 언어는 일반적으로 대소문자 구분이 없다. 따라서 대부분의 VHDL parser는 분석과정이 끝나면 대문자 또는 소문자로 변환을 하여 처리한다. 이러한 방식의 parser를 이용하는 설계규칙검사기는 RMM[1] 등에서의 요구하는 부분 신호선, 변수 포트의 이름은 소문자를 이용하며, 상수는 대문자를 이용하는 검사를 수행하지 못하는 문제가 생긴다. 따라서 규칙검사기는 설계자가 기술한 형태를 가공하지 않은 상태의 정보를 가지고 있어야 한다.

(나) 주석문의 사용가능

VHDL parser를 구현할 때 일반적으로 주석 문은 구문분석(lexical analysis) 과정에서 제거를 한다. 이러한 부분은 IEEE std 1076.6-1999 문서[2]에서 기술하는 다음과 같은 메타 명령어에 대한 검사를 불가능하게 한다. 설계규칙검사기가 이러한 종류의 메타 명령어를 검사하기 위하여 VHDL parser는 주석문의 정보를 내부적으로 지원하여야 한다.

- RTL\_SYNTHESIS OFF
- RTL\_SYNTHESIS ON

(다) 내부 표현형태에서 LRM(Language Reference

Manual) 상의 구조 유지

VHDL parser가 분석과정에서 VHDL LRM[3] 내에 기술되어 있는 언어의 문법 트리 구조를 유지하는 것은 새로운 설계규칙검사엔진 개발을 용이하게 한다.

(라) 구문상의 생략 가능한 부분의 내부 표현형태에서 생략 여부 표현

VHDL 언어 구문 상 임의의 부분은 생략이 가능하며, 이때 가정한 값을 가지게 된다. 그러나 IEEE1076.6 등의 설계규칙에서 임의의 부분이 생략되었는지 사용되었는지의 검사를 요구하게 되어 이러한 기능 검사를 위해서는 생략가능부분이 생략되었는지 사용되었는지에 대한 정보를 얻을 수 있어야 한다.

(마) 원시코드의 파일명과 내부구문의 라인번호의 정보

설계규칙검사기에서 오류정보의 위치를 알리기 위하여 VHDL parser에서 해당정보를 저장할 필요가 있다.

(2) 라이브러리 및 설계 유닛 탐색

라이브러리와 설계유닛 탐색은 사용 가능한 라이브러리와 해당 라이브러리의 설계 데이터 정보를 제공하여 설계규칙검사기가 규칙검사를 원하는 라이브러리와 설계 데이터를 선택할 수 있게 한다.

(3) 설계 계층구조 탐색기능

설계 계층구조 탐색은 규칙검사를 수행하는 설계 데이터의 설계 계층구조에 대한 정보를 제공하여 설계 규칙 검사기가 계층구조 정보를 통하여 IP 설계 전체를 검사할 수 있도록 도와준다.

(4) 설계 규칙검사기 엔진

설계 규칙검사기 엔진은 규칙검사를 담당하는 주요 부분으로 검사하고자 하는 설계 규칙을 선택함으로써 원하는 설계규칙을 검사할 수 있다. 설계 규칙검사기 엔진으로는 RTL 합성 설계규칙검사 엔진, 재사용 설계 규칙 검사엔진, 시뮬레이션 설계규칙 엔진 등의 다양한 엔진이 있다. 설계자 또는 회사내의 정의된 설계 규칙을 검사하는 사용자 정의 설계규칙 검사기 등도 VHDL parser를 이용하여 쉽게 구현이 가능하다.

(5) 설계규칙검사 관리

설계규칙검사 관리는 IP 설계자 및 IP 사용자가 IP의 성능을 검사할 수 있도록 관리하는 부분으로 Tcl/Tk를 이용하여 만든 사용자 인터페이스이다. VHDL 언어로 설계된 IP들을 분석하고, 규칙검사 대상을 선택하고, 설계 계층구조를 탐색하며, 설계규칙검사를 적용하는 등의 기능을 관리한다. 또한 규칙검사의

결과로 생성되는 검사정보를 사용자에게 제공하고 원 시코드와 상호참조기능을 제공하여 쉽게 IP 데이터를 개량할 수 있도록 도와준다.

### III. 설계 규칙검사의 구현

설계 규칙검사는 RTL 합성 설계규칙 엔진, 재사용 설계규칙엔진, 시뮬레이션 설계규칙 엔진 등이 있다. 합성 설계규칙 엔진은 IEEE std 1076.6-1999[X]에 기술되어 있는 레지스터 전송 수준의 합성을 위한 표준화 문서를 근간으로 개발되었다. 표준화문서는 Supported, Ignored, Not Supported의 3가지 카테고리로 분류되며, 설계규칙 검사는 Ignored 분류를 Warning으로 Not Supported 분류를 Error로 처리한다. 시뮬레이션 설계규칙 엔진은 사이클 기반 시뮬레이션에 적합한 설계로 되어 있는가를 검사하는 엔진으로 조합회로의 feedback 부분 등 사이클 기반 시뮬레이션에서 지원하지 않는 부분들을 탐지하여 사용자에게 제공한다. 재사용 설계규칙엔진은 Synopsys 사와 Mentor Graphics 사에서 만든 재사용 방법론 지침서를 근간으로 구현하였다.

설계규칙 엔진의 구현 방법은 설계규칙이 VHDL 구문 상에서 구문의 지원 여부에 대한 검사를 하는 것과 VHDL 구문에서 추론과정을 통하여 회로를 추출하고 이에 대한 검사를 하는 방법이 있다.

VHDL 구문상의 지원여부를 판단하는 검사의 예로 그림 2에 IEEE std 1076.6-1999의 신호선 할당구문에 대한 설계규칙과 내부표현 형태를 보여준다. 해당 설계규칙은 신호선 할당 구문이 있으면 VHDL 구문에서 label이 기술되었는가(Error), delay\_mechanism이 기술되었는가(Warning)와 waveform\_element가 2번 이상 기술되었는가(Error), 그리고, waveform\_element 내에 after 구가 있는가(Warning) 등을 검사하여 사용자에게 알려준다. 이러한 일이 수행은 내부표현형태를 이용하여 이루어진다. 신호선 할당구문의 내부표현에서 label의 기술여부는 label 항의 값이 NULL의 값을 가지는가를 검사함으로써 수행된다. delay\_mechanism의 기술여부도 신호선 할당 구문의 delay\_mechanism의 값이 NULL인지를 검사함으로써 수행된다. VHDL 구문 상에서 검사할 수 있는 설계규칙들을 parser에 의해서 생성된 내부표현이 VHDL 문법구조와 1대 1로 대응되기 때문에 쉽게 검사를 할 수 있다.

```
signal_assignment_statement ::=
    {label+} target <= [ delay_mechanism ] waveform ;
```

```
waveform ::=
    waveform_element {, waveform_element}
    | unaffected
```

```
waveform_element ::=
    value_expression [after time_expression]
    | null [after time_expression]
```

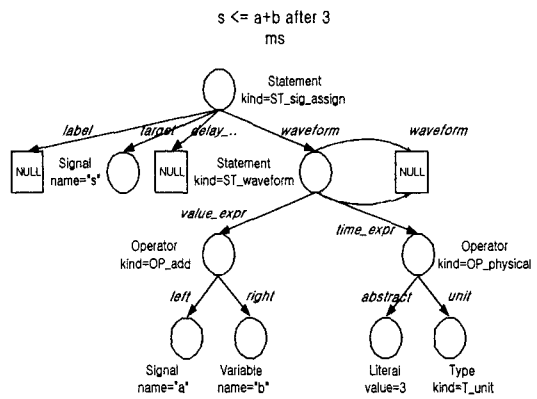


그림 2. 신호선 할당구문의 설계규칙과 내부표현

VHDL 구문 상에서 하드웨어 추론과정 이용한 설계 규칙검사 방법의 예로 그림 3과 4에 gated clock의 VHDL 기술과 추론된 하드웨어를 보여준다. RMM의 설계규칙에서는 gated clock의 사용을 피하라고 권고하고 있다.

```
process(clk)
begin
    wait until clk'event and clk='0';
    out1 <= data1;
end process;

g_clk <= out1 and clk;

process(g_clk)
begin
    wait until g_clk'event and g_clk='0';
    out2 <= data2;
end process;
```

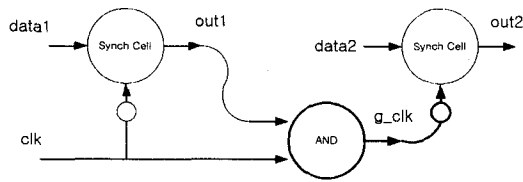


그림 4. Gated clock의 기술과 하드웨어 추론

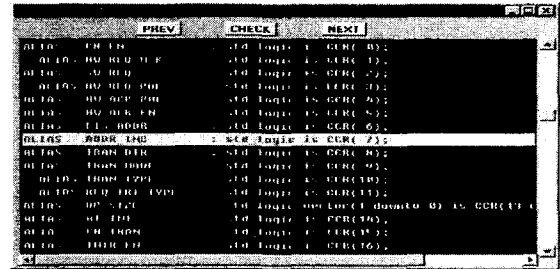


그림 7. 원시코드 창의 예

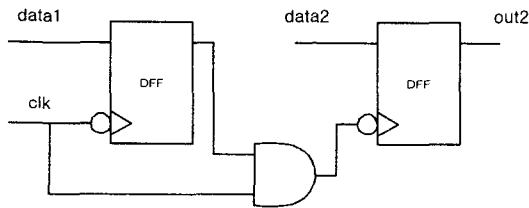


그림 6. 최종 추론 회로

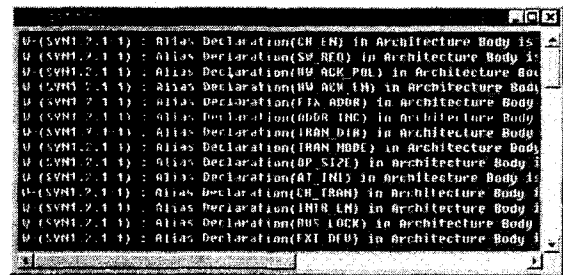


그림 8. 오류정보 창의 예

상기의 설계규칙 검사는 추론된 동기소자의 clock 단자로부터 입력의 방향으로 탐색을 하여 인버터가 아닌 논리소자가 추론되어 있는지를 검사함으로써 수행한다.

설계규칙 검사는 Tcl/Tk를 이용하여 사용자 인터페이스를 구현하였으며, 설계규칙검사를 수행 시에 나타나는 계층구조 탐색 창과 원시코드 탐색 창과 오류정보 창의 예를 그림5, 그림6, 그림7에 각각 보여준다. 상기 3개의 창은 서로 상호참조를 수행하여 계층구조창에서 설계를 선택하면 해당하는 설계의 원시코드가 원시코드 창에 나타나며 설계규칙검사를 수행한다. 실행으로 나타난 오류정보들은 오류정보 창에 표시되며 임의의 오류를 선택하면 해당 원시코드를 반전시켜 쉽게 IP 설계를 개선할 수 있다.

#### IV. 결론

본 연구는 IP를 목적으로 설계된 데이터가 IP로서 재사용 가능하게 설계되었는가를 검사하는 설계규칙검사의 구현에 대하여 기술하였다. 설계규칙은 다양한 엔진을 통하여 사용자가 선택적으로 이용할 수 있으며, 설계그룹 내부에 정의되어 있는 설계규칙 등도 새로운 설계규칙 엔진으로 쉽게 구현 및 등록이 가능하다. 설계규칙기의 사용자 인터페이스 부분은 Tcl/Tk로 구현하여 이 기종간의 이식이 용이하다. 본 연구의 결과는 IP를 목적으로 설계된 데이터의 성능을 판단하는데 도움이 될 것으로 사료된다. 추가적인 연구는 새로운 설계규칙검사 엔진의 개발과 설계규칙 기술언어의 연구를 통한 효율적인 규칙검사기를 연구하는 것이다.

#### 참고문헌

- [1] Reuse Methodology Manual, Second Edition, Kluwer Academic Publisher, 1999.
- [2] IEEE Standard for VHDL Register Transfer Level(RTL) Synthesis, IEEE Inc, 1999.
- [3] IEEE Standard Language Reference Manual, IEEE Inc, 1993.

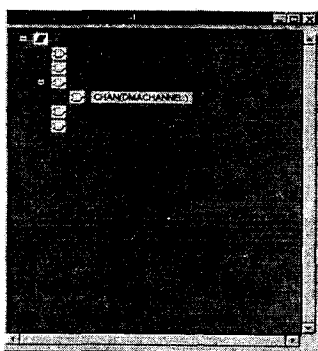


그림 7 계층구조 탐색 창의 예