

RTL에서 테스트를 고려한 설계

획득 가능한 테스트 범위의 평가 수준 향상

Atrenta의 SpyGlass DFT 툴을 사용할 경우, Verilog 또는 VHDL로 코딩을 하는 설계자들은 RTL 기술로부터 획득 가능한 테스트 (고장) 범위를 상당히 높은 수준까지 평가할 수 있다. 뿐만 아니라, 설계자들은 테스트 범위에 대한 설계 변화의 영향을 확인할 수 있다. 특히, 설계자들은 ATPG 툴을 구동하거나 복잡한 테스트 벤치를 개발하지 않더라도 이러한 툴을 통해 달성할 수 있었던 테스트 범위에 대한 신뢰성 있는 평가를 얻을 수 있다. <편집자 주>

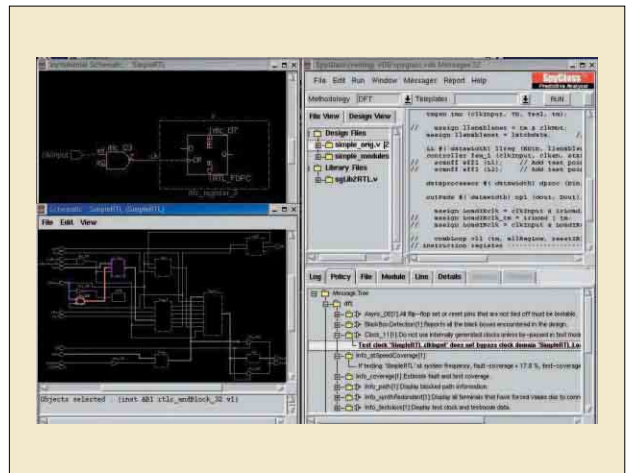
글: Ralph Marlett 제품 이사/Atrenta Inc.
www.atrenta.com

DFT(Design-for-Test)는 대부분의 기업에서 테스트를 위한 재설계(ReDesign-for-Test)로 생각되어 왔다. 뿐만 아니라, 테스트는 추가적인 것으로 생각되었다. 우선 설계는 코딩된 다음 시뮬레이션 되고 합성된다. 설계 주기 내에서 수 개월이 소요되는 이 모든 작업이 완료된 후 설계는 테스트 가능성을 보장하기 위해 테스트 팀에 넘겨진다. 하지만, 대부분의 경우 이들 설계는 테스트가 불가능하다.

가장 일반적으로 사용되고 있는 DFT 툴들은 게이트 레벨에서 동작하며, 시뮬레이션 및 합성 과정 이후에도 다시 이 레벨에서 동작한다. 게이트 레벨에서 어떠한 변화가 발생할 경우, 게이트와 RTL 코드 사이의 일대일 상관성은 존재하지 않는다. 원래의 RTL을 게이트 레벨의 변화와 일치하도록 되돌리는 것은 매우 어렵다.

합성 툴은 게이트에서 RTL로의 상관성을 제공하지 않기 때문에 특정 게이트 레벨 구조를 교정하기 위해 RTL을 조정하는 작업은 상당한 시간과 노력을 필요로 한다. 수백 개의 설계 파일과 수많은 계층 구조 레벨들로 인해 대규모 설계에서 게이트 레벨 위반에 대한 정확한 원인을 찾는 작업은 실로 엄두를 내기가 힘든 것이 사실이다. RTL이 변화하지 않는다면 동일한 진단 활동을 수행할 수 있으며, 그 코드를 재사용한다면 관련 설계 변화를 게이트 레벨에서 재수행해야 할 것이다.

목표는 물론 테스트를 위해 재설계 하는 것이 아니라 초기 단



계에서 테스트가 가능한 설계를 개발하는 것이다. 초기 설계 단계에서부터 RTL 설계자가 적절히 테스트 가능성 규칙 (testability rule)을 적용하지 않는다면, 설계에 대대적인 변화를 가하지 않는 한 열악한 테스트 범위를 가지거나 테스트 자체가 불가능할 수도 있다. 적절한 설계 기법을 통해 소중한 테스트 엔지니어링 시간을 절감할 수 있을 뿐만 아니라 상대적으로 재사용 가능성이 높은 코드를 얻을 수 있다. 테스트 가능성 문제는 RTL 코드에 거의 반영되지 않는 게이트 레벨에서의 변화를 형성하기 위한 요구사항을 형성한다.

RTL에서 테스트 범위 분석

Atrenta의 SpyGlass DFT 툴을 사용할 경우, Verilog 또는 VHDL로 코딩을 하는 설계자들은 이제 RTL 기술로부터 획득 가능한 테스트 (고장) 범위를 상당히 높은 수준까지 평가할 수 있다. 뿐만 아니라, 설계자들은 테스트 범위에 대한 설계 변화의 영향을 확인할 수 있다. 설계자들은 ATPG 툴을 구동하거나 복잡한 테스트 벤치를 개발하지 않더라도 이러한 툴을 통해 달성할 수 있었던 테스트 범위에 대한 신뢰성 있는 평가를 얻을 수 있다.

가장 일반적으로 사용되는 DFT 방식인 'Scan Design'은 일반적으로 테스트 영역을 회로의 플립플롭으로 이동시키고 그 테스트 결과를 외부로 이동시켜 테스트 기기가 실제 결과를 기대치와 비교할 수 있도록 하기 위해서 테스트 모드 제어를 필요로 한다. 이러한 테스트 모드 제어는 플립플롭 클록과 비동기식 설정 및 재설정 핀이 회로 상태에 영향받지 않도록 보장해야 한다.

예를 들어, 다음의 코드 프래그먼트(code fragment)에 설명된 것과 같이 저전력 애플리케이션은 물론 다양한 다른 애플리케이션에서 gated clock이 일반적으로 사용된다:

```
assign dknet = clk & dken;
always @ (posedge dknet);
```

Scan Design은 gated clock이 바이패스 로직을 반드시 가지고 있어서 클록들이 지원 로직(enable logic)을 제어하는 어떠한 플립플롭에도 영향을 받지 않아야 한다. 다음과 같이 조정된 코드는 테스트 모드에서 신호 "clk"가 시스템 클록 지원 로직으로부터 어떠한 영향도 받지 않도록 보장할 수 있다.

```
assign tdken = dken | Testmode;
assign dken = clk & tdken;
always @ (posedge dken);
```

스캔 이동 동작 시에 설정 또는 재설정이 활성화되면, 스캔중이거나 스캔이 완료된 데이터가 손상될 수 있기 때문에 파생된 설정 및 재설정 신호들은 물론 테스트 고려사항들을 요구한다. 최종 결과가 잘못된 칩을 합격시키거나 양호한 칩에 대해 고장

판단을 할 수 있다. 어떠한 경우여라도 이것은 상당한 문제를 야기한다. 다음의 코드 프래그먼트는 내부적으로 생성된 신호를 통해 재설정되는 간단한 플립플롭의 예를 나타낸 것이다.

```
always @ (posedge clk or negedge derived_reset);
if (derived_reset) q <= 1'b0;
else q <= data;
```

테스트 관점에서 다음에 설명한 테스트 모드 신호를 통해 내부적으로 생성된 재설정을 표시해야만 한다:

```
assign tmreset = derived_reset | test_shift;
always @ (posedge clk or negedge tmreset);
if (tmreset) q <= 1'b0;
else q <= data;
```

RTL에서 기억해야 할 수많은 규칙들

모든 RTL 설계자들이 간단하게 만들 수 있는 간단한 변화들을 피상적으로 나타낸 이러한 2가지 예는 단지 방산의 일각에 불과하다. 기억해야 할 너무나 많은 규칙들이 있기 때문에 고도로 숙련된 설계팀들조차도 상당한 어려움을 겪는다. 따라서 많은 기업들이 숙련된 설계자들이 프로젝트에 대한 모든 코드를 감시할 수 있도록 RTL 코드 검사를 수행한다. 하지만 불행하게도 코드 검사를 통해서도 이러한 모든 문제들을 인지할 수는 없다.

설계를 구성하는 수백 개의 설계 파일이 존재할 수 있기 때문에 문제의 증상(예를 들어, 스캔을 대체할 수 없는 플립플롭)이 실제 원인으로부터 제거되거나 문제를 해결하기 위한 최선의 방법이 될 수 있다.

엄격한 일정과 해결해야 할 상충되는 시스템 요구사항들로 인해 RTL 설계자들은 DFT 문제들을 처리해야 할 필요성을 느끼지 못한다. 특히, DFT 변화에 대한 대가 역시 그리 명확하지 않기 때문이다. 하지만, 이제 일반적으로 사용되는 측정기준인 테스트 범위가 설계에 대한 아주 작은 변동의 영향까지도 그래픽적으로 나타내기 위해 RTL 레벨에서도 제공된다.

테스트 범위 평가의 목적은 테스트 범위에 대한 빠르면서 상당히 정확한 평가를 제공하는 것이다. 결과적으로 상용 ATPG

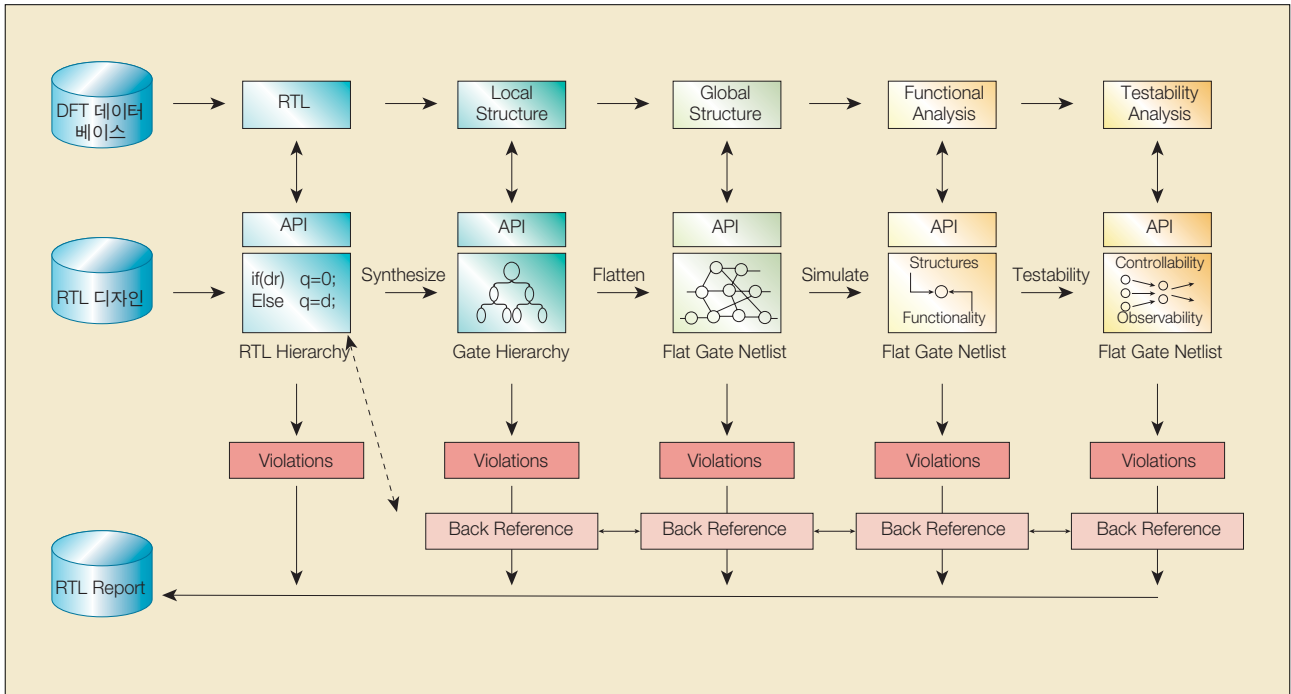


그림. SpyGlass 엔진의 DFT 문제를 위해 RTL을 처리하는 과정. SpyGlass는 설계의 RTL 규칙 위반 사항을 검사한다. 하지만, RTL 코드는 제한된 정보만을 제공하기 때문에 SpyGlass는 게이트의 계층 구조를 검사하기 위해 고속 합성 엔진을 사용한다. SpyGlass는 합성 결과를 평탄화하고 전체 구조를 검사하여 기능 및 테스트 가능성 분석을 수행한다. 모든 결과는 RTL에 다시 반영된다.

툴은 설계 플로우에 있어서 이보다 늦게 이러한 평가를 제공한다. Atrenta의 SpyGlass DFT 제품은 이러한 테스트 범위 추정을 신속하게 제공할 수 있다.

RTL에서 ATPG에 대한 분석

ATPG 툴은 특정 값(0 또는 1)에 고정된 핀을 대상으로 설계 되었으며, 실행 태스크와 전파 태스크 모두를 수행한다. ATPG가 0 또는 1에 고정된 실제 핀에 대해 이들 태스크를 모두 실행하는지의 여부에 상관없이 필요조건들이 동시에 존재할 수 있기 때문에 판명된 고장에 고정된 핀을 감지한다.

2개의 동등한 테스트 가능성 분석 태스크를 통해 2개의 기본적인 ATPG 태스크를 RTL에서 추정할 수 있다. 고장 실행은 '제어 가능성(controllability)'으로 대체되며 고장 전파는 '관찰 가능성(observability)'으로 대체된다. 2개의 태스크는 ATPG보다 훨씬 간단하며, 테스트 가능성 분석 분야에서 상당히 일반적인 개념이다.

테스트 가능성 분석은 입력 핀에서 게이트 레벨 회로 내의 노드를 제어할 수 있는지, 그리고 출력 핀에서 관측 가능하지의 여부를 결정한다. 과거에 수많은 알고리즘이 게이트 레벨 기술에서 사용되었다. 게이트 레벨 넷리스트를 빠르게 생성하면서 논리적 정확성을 유지하는 것이 핵심이다.

테스트 포인트 삽입

테스트 포인트 삽입에 대한 우선순위를 부여하기 위해 테스트 범위에 근거한 테스트 가능성 분석을 사용할 수도 있다. 앞서 열거한 코드 예제들에서 테스트 모드 신호를 조합하여 성공적인 테스트 생성으로부터 시스템 로직을 블록화할 수 있다. 관측 테스트 포인트가 추가되어야 할 곳에 배치하는 것은 간단한 공정이지만, 고장 감지 기능을 증가시키기 위해 이러한 각각의 위치를 분석한다면 평가된 고장 수의 증가는 각각의 것과 관련될 수 있기 때문에 설계자를 위한 소중한 지침을 제공할 수 있다.

실제 설계에서 SpyGlass DFT 내에 구축된 이러한 기능을 사용하고 운영함으로써 테스트 범위가 상당히 향상되는 것을 알 수 있었다.

RTL에서 가치 있는 정보를 제공하는 예측적 분석

예측적 분석 기능을 통합한 Atrenta의 SpyGlass DFT와 같은 설계 툴을 사용할 경우, 특히 게이트 레벨 표현을 생성하는 고속 합성 기능을 포함하고 있다면 이와 같은 테스트들을 상당히 단순화시킬 수 있다. 이를 통해 RTL 설계 단계에서 구조적 분석을 수행할 수 있다.

본질적으로 이를 통해 툴은 복잡한 테스트 가능성 문제뿐만 아니라 클록 영역 교차, 동기화, 3상 버스 디코딩(tri-state bus coding), 조합 루프(synthesis loop), 로직 콘 심도(logic cone depth) 등과 같은 복잡한 설계 문제들을 RTL에서 감지할 수 있으며, 직접적으로 원래의 RTL 파일과 라인 번호에 오류들을 보고할 수 있다. 그래픽적인 디버깅 환경과 도식 뷰어를 통해 신속하게 문제를 차단할 수 있다. 이때 회로 내의 노드가 0 또는 1로 조정될 수 있는지, 또는 각 노드가 회로 또는 테스트 출력에서 관측이 가능한지 여부를 결정하는 최적화된 테스트 가능성 분석기가 포함될 경우, 예측적 분석 기반 DFT는 매우 유용하다.

테스트 범위를 평가하기 위해 테스트 가능성 분석을 사용하기 위한 타당성은 다음의 2가지 원칙에 근거한다.

첫 번째 원칙은 고장이 0 또는 1로 실행될 수 있다면 1 또는 0으로 제어될 수 있는가이다. 두 번째 원칙은 고장이 출력에 전달될 수 있다면 해당 출력에서 관측이 가능한가이다.

DFT에 근거한 예측적 분석을 통해 테스트가 가능하지 않은 고장뿐만 아니라 사용되지 않는 고장까지 판단할 수 있으며, 따라서 테스트 범위를 평가할 수 있다.

노드에 대한 제어 가능성과 관찰 가능성은 이러한 조건들이 동시에 만족될 수 있는지에 대해 그 어떤 것도 설명하지 않는다. 그럼에도 불구하고 제어 가능성과 관찰 가능성에 기반한 평가는 실제 ATPG 성능에 거의 근접한다. 더구나 문제 해결을 위한 대안적인 방법들을 비교하는 데에도 평가를 사용할 수 있다. 이러한 경우, 정확한 값의 중요성이 상대값보다 낮을 수 있다.

RTL에서 테스트를 위한 실제 설계

설계가 한층 복잡해짐에 따라 테스트를 위한 설계가 RTL 설계 플로우 내에 통합될 필요가 있다. 이후 설계 주기에 문제를 야기할 상당한 가능성을 가진 RTL 코드와 영역을 검사하고 표시하는 공정을 자동화시킨 Atrenta의 SpyGlass DFT를 통해 이러한 문제를 해결할 수 있다. 설계자들은 그들의 머리 속에 반드시 기억해야 할 규칙들과 고려해야 할 많은 문제들이 있다. SpyGlass는 자동적으로 코드를 분석하고 문제 영역을 부각시킨다.

이외에도 설계팀은 SpyGlass 툴에 자체 규칙을 추가할 수 있다. 많은 대기업들이 최상의 실행을 위해 SpyGlass 규칙에 대한 기초를 개발하는데 참여했다. SpyGlass는 이미 이러한 규칙들 중 상당부분을 포함하고 있으며, 일부 기업들은 RTL 코드를 하나의 설계에서 뿐만 아니라 다음 설계에서도 재사용 가능하도록 하기 위해 보다 많은 규칙들을 추가하고 있다.

엔지니어링의 다양한 분야에서 뿐만 아니라 IC 설계에서도 초기에 규칙을 올바르게 설계하는 것이 중요하다. 이제 설계자들은 설계가 완벽하게 테스트 가능하고 최상의 실행 규칙들을 충족하는지를 확인하기 위해 기존 시뮬레이션 및 합성 공정으로 이동하기에 앞서 자신의 IC 설계 코드를 신속하게 평가할 수 있다. ^{RTL} _{time}

“벤더와 유저를 잇는 가교”

PRODUCT FEATURE의 지면은 항상 열려 있습니다. 이 코너는 귀사의 솔루션을 단순히 소개하는 차원을 넘어, 유저의 입장에서 다양한 활용 테크닉을 소개할 수 있습니다.

아울러, 차별화된 기능과 특징을 남아 있는 1%까지 소개함으로써 귀사의 솔루션을 유저들이 100% 활용할 수 있을 것입니다.

제품기사 담당 최찬용 기자
e-mail: cco@techworld.co.kr