

# SmartSpice의 Verilog-A 파서에 대한 자세한 정보

## 소개

최근 몇 년간, Verilog-AMS 하드웨어 표현 언어 (Verilog-A)는 아날로그, 혼합 신호 설계에 널리 사용되었습니다. 이에 발맞추어, 대부분의 EDA 기업은 Verilog-A 용 시뮬레이션 툴을 제공하고 있습니다. 실바코는 몇 년 전 SmartSpice에서 Verilog-A에 대한 지원을 추가하였습니다. 최근 SmartSpice의 Verilog-A 파서가 일부 개선되었으며, 이는 시뮬레이션 및 프로젝트 관리의 성능 향상에 큰 도움이 될 것입니다. 본 애플리케이션 노트에서 이를 자세히 소개합니다.

## 파일 관리 개선

SmartSpice에서 Verilog-A 파서는 사용자의 소스 코드를 처리한 후 일부 파일을 생성해야 합니다. 원래 이런 파일은 프로젝트 파일과 동일한 위치에 저장되었습니다. 입출력 파일이 같이 있게 되면, 프로젝트 관리에 불편을 초래할 수 있습니다. 이를 해결하기 위해, SmartSpice Verilog-A 파서를 개선하였습니다. 이제 Verilog-A 파서가 프로젝트 파일 위치에 디렉토리 트리를 생성하여, 모든 결과 파일을 이 디렉토리에 기록합니다. 디렉토리 트리는 그림 1과 같이 생성됩니다.

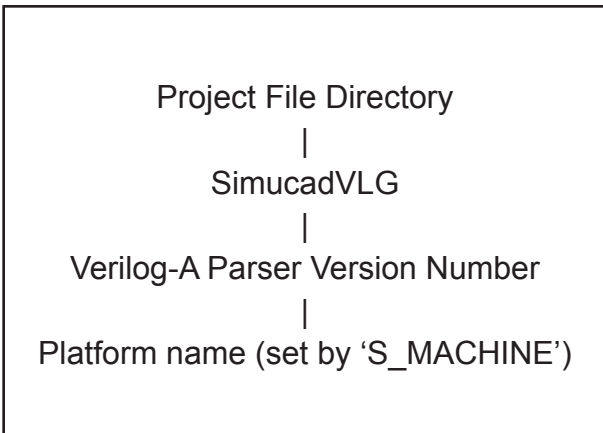


그림 1. 디렉토리 트리 패턴

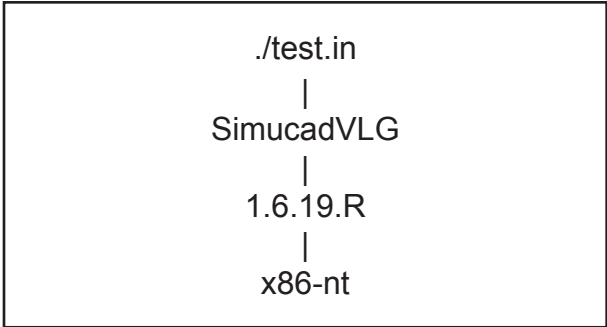


그림 2. 디렉토리 예제

예를 들어, `project file=./test.in`, `Verilog-A Parser version=1.6.19.R`, `S_MACHINE=x86-nt` 경우 Verilog-A 파서는 그림 2와 같은 디렉토리 트리를 생성합니다. 이 디렉토리 트리에서 입출력 파일이 분리되므로, Verilog-A 파서에서 생성한 결과 파일을 쉽게 관리할 수 있습니다.

## Verilog-A 파서 결과의 재사용

SmartSpice에서 Verilog-A ("SCI" 외의 모드)로 작성된 파일을 발견하면, Verilog-A 파서가 파일을 처리하고 SmartSpice 시뮬레이션에 사용할 공유 라이브러리를 생성합니다. 이는 시간이 오래 걸릴 수 있습니다. 모든 시뮬레이션에서 변경되지 않은 Verilog-A 파일을 처리할 필요가 없으며, 이는 시간 낭비입니다. SmartSpice에서 공유 라이브러리를 재사용할 수 있도록 개선이 이루어졌습니다. Verilog-A 소스 파일과 그 종속 파일이 바뀌지 않았다면, 전에 이 파일로부터 생성된 공유 라이브러리는 파서로 전처리할 필요없이 재사용할 수 있습니다. 따라서 Verilog-A 파일은 Verilog-A 파서에서 한 번만 처리하면 되고, SmartSpice에서 그 결과를 반복적으로 사용하여 처리 시간을 단축할 수 있습니다.

SmartSpice 출력 메시지를 통해 파일 처리에 Verilog-A 파서를 사용하는지 또는 이전 결과를 재사용하는지 쉽게 파악할 수 있습니다. 예를 들어 다음과 같은 메시지가 있습니다:

```
(VERILOGA): Compiling file ..... 'D:\test_case\laplace.va'.
```

"laplace.va" 파일을 Verilog-A 파서로 처리하고 있음을 나타냅니다. 반면에 다음 메시지는,

```
(VERILOGA): 'D:\test_case\laplace.va'의 기존 모델을 사용합니다.
```

이전 결과를 재사용할 수 있으므로, "laplace.va"를 파서에 서 처리할 필요가 없음을 나타냅니다.

## 독립 Verilog-A 파서

과거에, Verilog-A 파서는 별도의 프로그램이 아니었습니다. 그것은 단지 SmartSpice의 모듈이었으며, SmartSpice에서만 호출할 수 있었습니다. 또한 새로운 Verilog-A 버전은 SmartSpice를 다시 설치해야 합니다. 현재의 SmartSpice에서, Verilog-A 파서는 SmartSpice에서 호출하거나 또는 별도로 실행할 수 있는 프로그램입니다. 따라서 SmartSpice를 사용하지 않고 파서만 실행하여 Verilog-A 파일을 처리할 수 있으며, 파서가 생성한 공유 라이브러리는 SmartSpice가 시뮬레이션에서 직접 활용할 수 있습니다. 또한 새로운 버전의 Verilog-A 파서가 출시되면, 파서만 설치하면 되고 SmartSpice는 변경할 필요가 없으므로 관리가 편리합니다. 또한, 독립적인 Verilog-A 파서는 하나의 명령줄로 여러 개의 파일을 처리할 수 있습니다. 예를 들어, 다음 셸 명령어는:

```
> veriloga -l *.va -vcc
```

현재 작업 디렉토리의 모든 Verilog-A 파일을 처리하며, 이는 강력하고 간단합니다.

명령줄에서 Verilog-A 파서를 실행할 때 몇 가지 유의할 점이 있습니다. 먼저, Verilog-A 파서에서 생성한 로그 파일 내에 오류 디버깅에 관하여 특히 유용한 정보가 있으므로, "-"을 항상 설정하는 것이 좋습니다. 둘째, 윈도우에서 환경 매개 변수 "LIBVGL\_PATH"가 설정되지 않은 경우, "-libvlgpath" 옵션을 사용하여 윈도우 C 컴파일러가 libVGL 라이브러리를 가져올 수 있는 경로를 정확하게 지정해야 합니다. 셋째, 옵션 "-f"는 입력 파일 이름을 대체하는 모든 메시지에서 표시되는 파일 이름을 설정하는 데 사용됩니다. 예를 들어, 아래 명령어는

```
>veriloga -l .\test\example1\example.va
-f example.va -cc
```

표시된 모든 메시지에서 "example.va"을 사용하여 ".\test\example1\example.va"를 대체합니다. 이 옵션은 메시지 길이를 줄이고 읽기 쉽게 합니다. 그러나 명령줄에 여러 개의 입력 파일이 포함된 경우, 이 옵션을 사용할 수 없습니다.

SmartSpice와 마찬가지로, 이제 사용자는 명령줄에서 "-V" 옵션을 사용하여 Verilog-A 파서의 버전을 지정할 수 있습니다. 예를 들어, 셸 명령어는

```
>veriloga -V 1.6.17.R *.va -vcc
```

Verilog-A 파서 버전 1.6.17.R을 사용하여 파일을 처리합니다. 지정된 버전이 설치되지 않은 경우, 다음 메시지가 나타납니다:

```
"Command line -V "1.6.17.R" not found.
Exiting.
```

이 플랫폼에 사용할 수 있는 버전은 1.6.15.R과 1.6.13.R입니다.

SmartSpice에서 Verilog-A 파일을 처리할 때, SmartSpice ModelLib 설정 파일에서 지정된 libVGL 라이브러리에 해당 버전의 Verilog-A 파서가 설치되어 있지 않으면, SmartSpice 출력 창에서도 이 메시지를 표시합니다. 버전 일치 요구 사항이 중요합니다. SmartSpice를 사용하려면 libVGL 라이브러리 버전이 Verilog-A 파서 버전과 일치해야 합니다. libVGL 라이브러리 버전은 ModelLib 설정 파일에 지정되어 있습니다. SmartSpice는 정확한 버전의 Verilog-A 파서가 사용되는 것을 보장하므로, 일반적으로 문제가 되지 않습니다. 그러나 사용자가 Verilog-A 파서를 독립적으로 사용하여 먼저 Verilog-A 파일을 처리하고 나중에 생성된 결과를 SmartSpice에서 바로 재사용하려면, 전처리를 수행하는 Verilog-A 파서의 버전이 SmartSpice에서 사용하는 libVGL 라이브러리 버전과 일치하는지 확인해야 합니다. 그렇지 않으면 SmartSpice가 올바른 Verilog-A 버전을 호출하여 해당 Verilog-A 파일을 처리하므로, Verilog-A 파서 결과를 재사용할 이유가 없습니다.

## 결론

본 애플리케이션 노트에서, SmartSpice의 Verilog-A 파서에 대한 최신 변경사항을 소개합니다. 이러한 개선에 의해 프로젝트 관리가 쉬워지고, 시뮬레이션 성능이 향상되며 사용자에게 보다 많은 유연성을 제공합니다. 이러한 개선 사항을 이해하면, 툴을 보다 효율적으로 사용하여 작업을 완료하는 데 도움을 줍니다.