

# VHDLとVerilog-HDLが 混在したデザインを検証する



森田栄一

本誌付属のCD-ROMに収録した評価版ツールを利用する「EDA ツール活用チュートリアル」の第6回である。今回はVHDLのデザインとVerilog-HDLのデザインが混在した設計データを一括してシミュレーションする。LSIの集積度の向上にともない、設計再利用の意識が高まっている。しかし、過去の設計資産のなかには異なる設計環境のもとで開発されたものが少なくない。とくにハードウェア記述言語については、VHDLとVerilog-HDLの両方が業界標準になっているため、両方のデザインが引き継がれているケースがよくある。最近では、こうした状況を考慮してVHDLとVerilog-HDLの混在したデータを扱えるHDLシミュレータが市販されている。今回のチュートリアルで使用するModel Technology社製のModelSimも、その一つである。いつものように設計済みのサンプル・データを用意した。機能検証だけでなく、タイミングを取り扱うためのSDF(Standard Delay Format)アノテーションなどにも触れる。混在シミュレーションを体験することによって、VHDLとVerilog-HDLの言語仕様の違いについても、理解が進むだろう。

(編集部)

## 1 はじめに

「携帯端末」や「モバイル」といった言葉を耳にする機会が増えています。こうした製品の開発では、複数の半導体チップを1チップにまとめる作業が非常に重要になります。これは、基板の面積を縮小したり、ブロック(コンポーネントまたはモジュール)間を高速な内部バスで接続することによって、高速なデータ転送を実現できるからです。

このような設計では、再利用性の高いHDL設計手法が威力を発揮します。基本的に、過去の半導体チップのHDLデータにシステム・バスのインターフェースの記述を追加するだけですむからです。

ただし、過去の半導体チップが異なる設計環境で開発されている場合には、いくつかの問題が生じます。たとえば、あるブロックはVHDLの設計環境で、別のブロックはVerilog-HDLの設計環境で開発されていたとしましょう。合成については、ブロック単位に作業を進めて、ネットリストの

段階でブロック同士を接続することになります。一方、チップ全体の検証(アーキテクチャやRTLの検証)を行うには、別の工夫が必要になります。

最近では、上記のようなVHDLとVerilog-HDLの「混在」に対応したHDLシミュレータを、いくつかのEDAベンダが出荷しています。ここでは、こうした検証を「混在シミュレーション<sup>※</sup>」と呼ぶことにします。混在シミュレーションを利用すると、それぞれのモジュールのHDLソース・コードを変更する必要がありません。つまりVHDLデータをVerilog-HDLで書き直したり、Verilog-HDLデータをVHDLで書き直す必要がありません。

このチュートリアルでは、混在シミュレーションの機能をサポートするHDLシミュレータの一つである「ModelSim 評価版(Model Technology社製)」を利用して、混在シミュレーションの作業工程を体験していただきたいと思います。ModelSim 評価版とサンプルのソース・コードは本誌付属のCD-ROMに収録されています。

読者の方には今回のチュートリアルを通して、

- VHDLとVerilog-HDLの特徴
- VHDLとVerilog-HDLが混在したデザインの扱い方

を理解していただきたいと思います。

本稿では、次の順番で話を進めます。

2節で、まず混在シミュレーションの概要を説明し、その後、実際にツールを操作してシミュレーションを実行します。

3節ではVHDLとVerilog-HDLの間の信号値の変換規則について説明します。Verilog-HDLの信号強度を含む値がVHDL上でどのように変換されるのかについて示します。

4節ではVHDLのライブラリ概念をVerilog-HDLでどのように取り扱うのかについて説明します。また、混在設計におけるSDF(Standard Delay Format)アノテーションの概要などにも触れます。

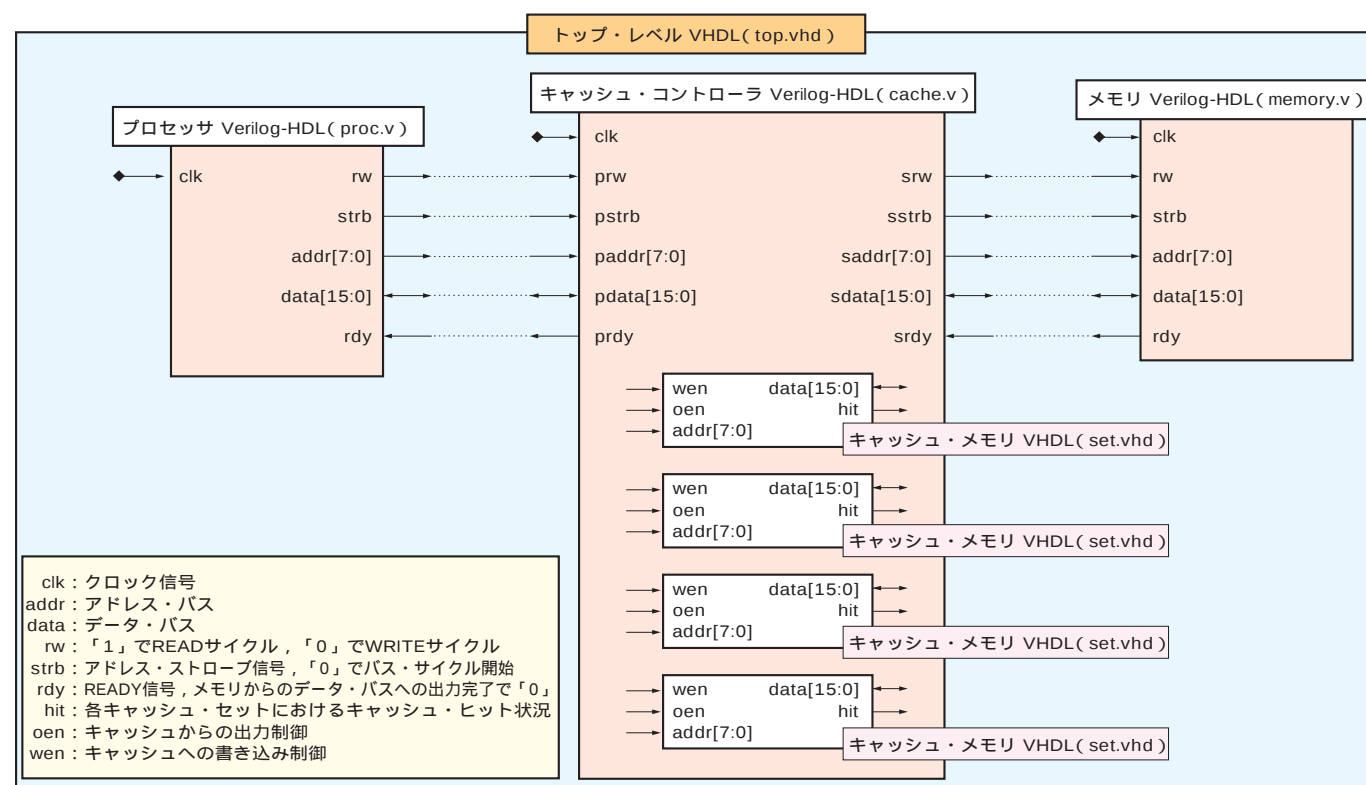
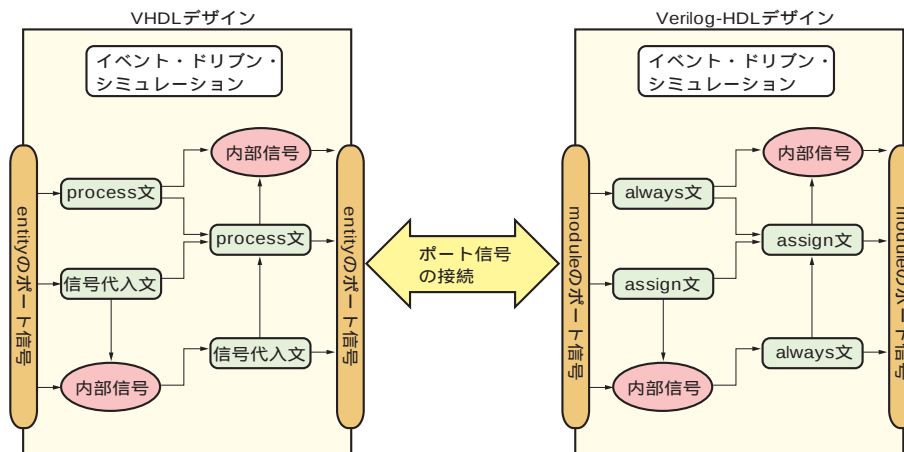
なお、今回のチュートリアルでは、読者の方々にVHDLやVerilog-HDLなどの基本的な知識があるものとして、話を進めます。たとえばHDLの文法や設計フローについての説明は省略します。これらについては、ほかの教科書や雑誌記事などを参考にしてください。

注：ここでの「混在シミュレーション」はミックスド・レベルの論理シミュレーション(たとえばゲート・レベルとRTLの混在シミュレーション)やアナログ/デジタル混在シミュレーション(ミックスド・シグナル・シミュレーション)、ハードウェア/ソフトウェア協調シミュレーションではないことに注意されたい。

【図1】

## 混在シミュレーションの概念

各ブロック(コンポーネントまたはモジュール)は、単一のHDLで記述された場合と同じ動作を行う。2種類のHDLのデザインは、ポート信号の接続部でのみインターフェースする。



【図2】混在回路の例

プロセッサとメモリ・キャッシュからなる回路。トップ・レベルはVHDL, 下位階層はVerilog-HDLのブロックである。三つのブロックのうち、一つのブロック(キャッシュ・コントローラ)の下位階層にはVHDLのブロックが含まれる。

# 2

## 混在シミュレーションを体験する

ここでいう「VHDLとVerilog-HDLの混在」とは、単一のソース・コードのなかに2種類のHDLの記述が含まれているということではありません。図1のように、VHDLで記述されたデザインとVerilog-HDLで記述されたデザインをそれぞれ独立したブロックとして配置し、それらのポート信号を接続することを指します。異なる言語で記述したモジュールを接続することを除けば、接続方法そのものは単一言語の場合と同じです。詳しい記

述方法については、後述します。

さて、VHDLのシミュレーションとVerilog-HDLのシミュレーションには、イベント処理や信号代入のメカニズムに微妙な違いがあります。しかし、イベント・ドリブン型の論理シミュレータを使用する限り、基本的な処理は変わりません。デザインのなかのブロック(process文, always文)で生じた信号値の変化(イベント)がほかのブロックに伝播していくことでシミュレーションが進行します。このときにイベントを伝播するものが信号(VHDLのsignal, Verilog-HDLのネット)になります。

混在シミュレーションにおいても、VHDLとVerilog-HDLの各ブロック