Artgraphics

SystemVerilog 設計・検証ツール製品紹介

Document Revision: 1.1, 2019.01.14

注意事項

ここで紹介する内容は、最新版と異なる場合があります。ご了承下さい。

SystemVerilog 設計・検証ツール製品紹介

〒124-0012 東京都葛飾区立石 8-14-1
www.artgraphics.co.jp

SystemVerilog Tools Summary

8-14-1, Tateishi, Katsushika-ku, Tokyo, 124-0012 Japan
www.artgraphics.co.jp

1 製品の目的

設計・検証作業における生産性向上の必要性は、必然的に、より汎用的・抽象的な記述を余儀なくさせます。更に、RTL から TLM への移行を経験する現状を鑑みて、設計・検証技術者は最新の技術、及び、ツールを駆使しなければなりません。

最先端の検証技術（VMM、UVM 等）を効果的に適用する為には、最適な GUI の使用が不可欠です。SystemVerilog IDE 、及び、SVChecker は時代に即した機能を提供します。

検証機能は説明の余地が無い為、この文書では GUI 機能を中心にして紹介します。

2 製品概要

2.1 SVDesigner

SVDesigner は設計・検証機能を統合した GUI ベースの開発環境です。以下の様な機能を備えています。

- デザイン・マネージャー（プロジェクトのファイルを管理し、コンパイルからシミュレーションまでの一連の処理を行ないます）
- コード開発と支援機能（シンタックス・ハイライト・テキスト・エディタ、ナビゲータ、リント、記述ルール・チェック）
- クイック参照機能（コード・スニペット、ブックマーク）
- SystemVerilog コンパイラー、及び、シミュレータ
- RTL 論理合成（業界標準のテクノロジー・ライブラリーをサポート）
- 検証機能（アサーション、カバレッジ、制約付き乱数発生機能）
- 検証ビューワー（VCD、カバレッジ、アサーション）
- UVM サポート（UVM は複雑なコンストラクトである為、GUI の使用は不可欠です）
- HTML 文書生成（ SystemVerilog 記述 を HTML 形式に変換します）
- ワークベンチ（データ構造ウィザード、UVM クラス・ウィザード、テストベンチ生成等）
- 自動バックアップ機能
- ソフトウェア更新機能（ユーザは DownloadMgr を使用して適宜ソフトウェアを更新する事が出来ます）

2.2 SVChecker

SVChecker は SystemVerilog に依る設計・検証を支援する開発環境です。SVDesigner と異なり、SVChecker は検証機能を持ちません。既に検証ツールを保有しているユーザ向けの GUI 環境です。具体的には、以下の様な機能を備えています。

- プロジェクト・マネージャー（プロジェクトのファイルを管理します）
- コード開発と支援機能（シンタックス・ハイライト・テキスト・エディタ、ナビゲータ、リント、記述ルール・チェック）
- クイック参照機能（コード・スニペット、ブックマーク）
- SystemVerilog コンパイラー
- UVM サポート（UVM は複雑なコンストラクトである為、GUI の使用は不可欠です）
- ワークベンチ（データ構造ウィザード、UVM クラス・ウィザード、テストベンチ生成等）
- 自動バックアップ機能
- ソフトウェア更新機能（ユーザは DownloadMgr を使用して適宜ソフトウェアを更新する事が出来ます）

3　機能紹介

IDE としての特徴は、使用者の生産性を向上する事を目的としている事です。コード入力の負荷軽減、ソース・コードを見易くする支援、参照資料の管理、プロジェクトに必要なファイル管理等様々な観点から設計者・検証技術者の支援をします。以下では、GUI としての先進的な機能を紹介します。

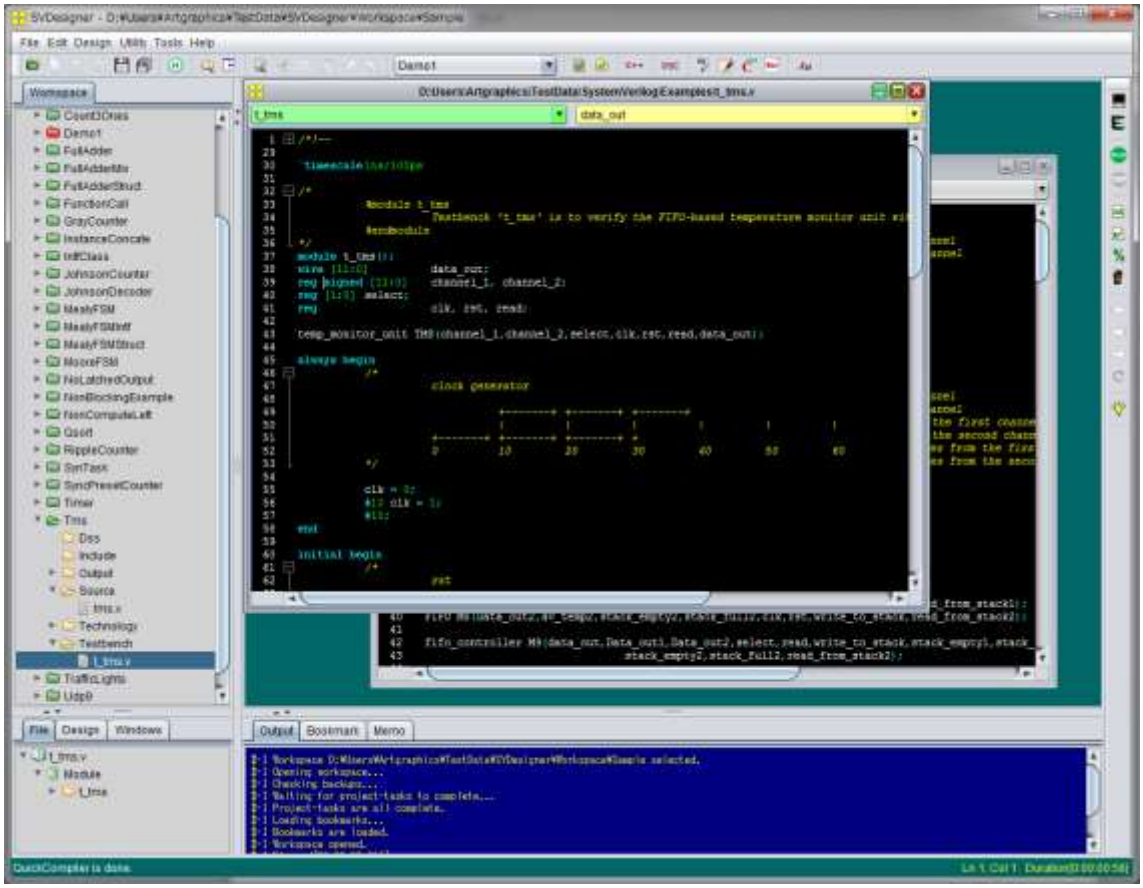
3.1　SVDesigner と SVChecker の差異

二つの製品の違いを以下の様に総括する事が出来ます。

	SVDesigner	SVChecker
画面管理法	MDI (Multiple Document Interface)	TDI (Tabbed Document Interface)
開発機能	有り	有り
検証機能	有り	無し

3.2　SVDesigner

SVDesigner は下図に示す様な外観を持ちます。



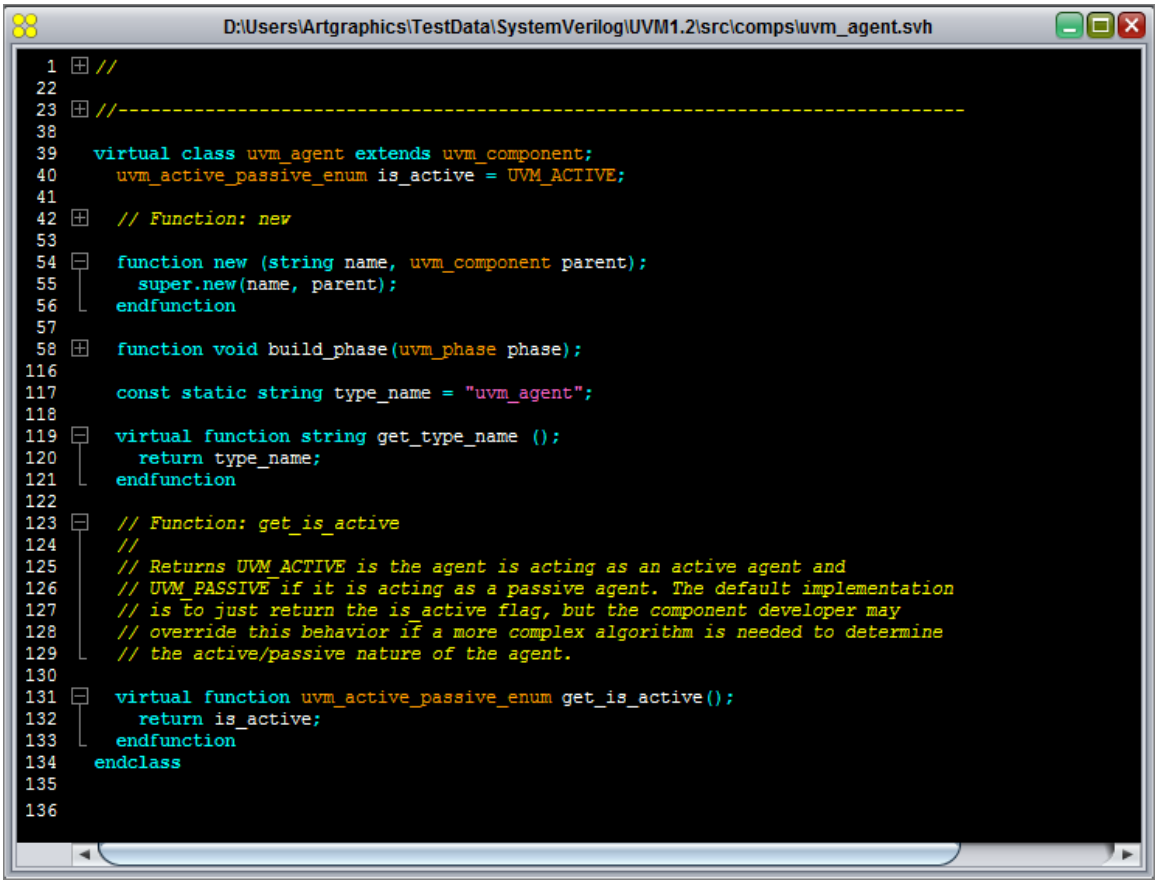
3.2.1 テキスト・エディタの特徴

テキスト・エディタの特徴を以下の様に纏める事が出来ます。

- SystemVerilog を意識したシンタックス・ハイライト機能を備えています。
- 行番号を最新状態に保ちます。
- 行の折り畳み機能を備えています。
- テキスト・エディタの画面を２分割する事が出来ます。
- トークン上でマウスを静止するとトークンの情報が表示されます。
- 入力に従いナビゲータがガイドします。
- 各種のショートカット・キーを備えています。
- その他クラシックな編集機能を備えています。

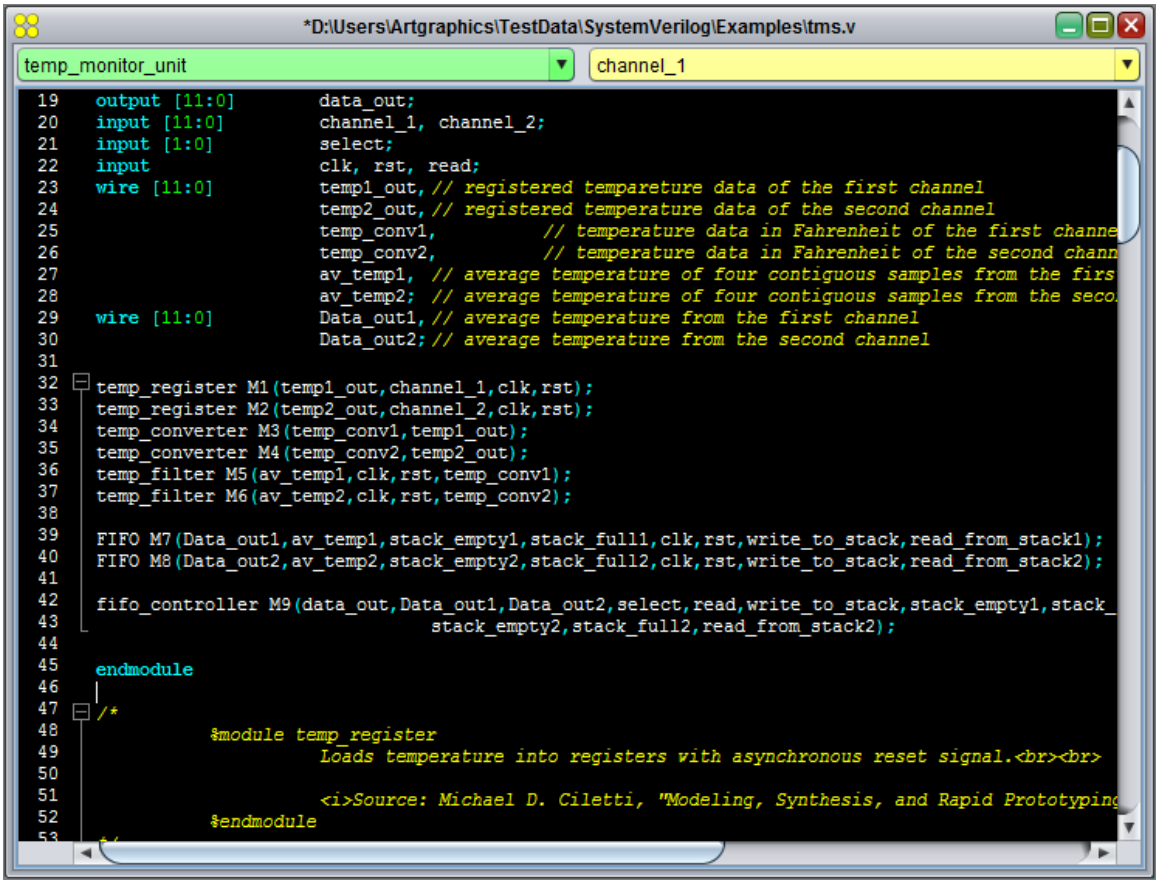
3.2.2 行の折り畳み

行の折り畳み機能は非常に便利です。特に、クラス内に複雑なタスク、及び、ファンクションが定義されている場合、注目したい箇所を中心に表示する事が出来ます。例えば、下図は、uvm_agent クラスを表示しています。１ページに全容が表示されています。行を折り畳まない場合には、２ページとなり１画面に収まりません。その場合、クラス全体の内容を理解し難くなります。



3.2.3 ユーザ固有の折り畳み定義

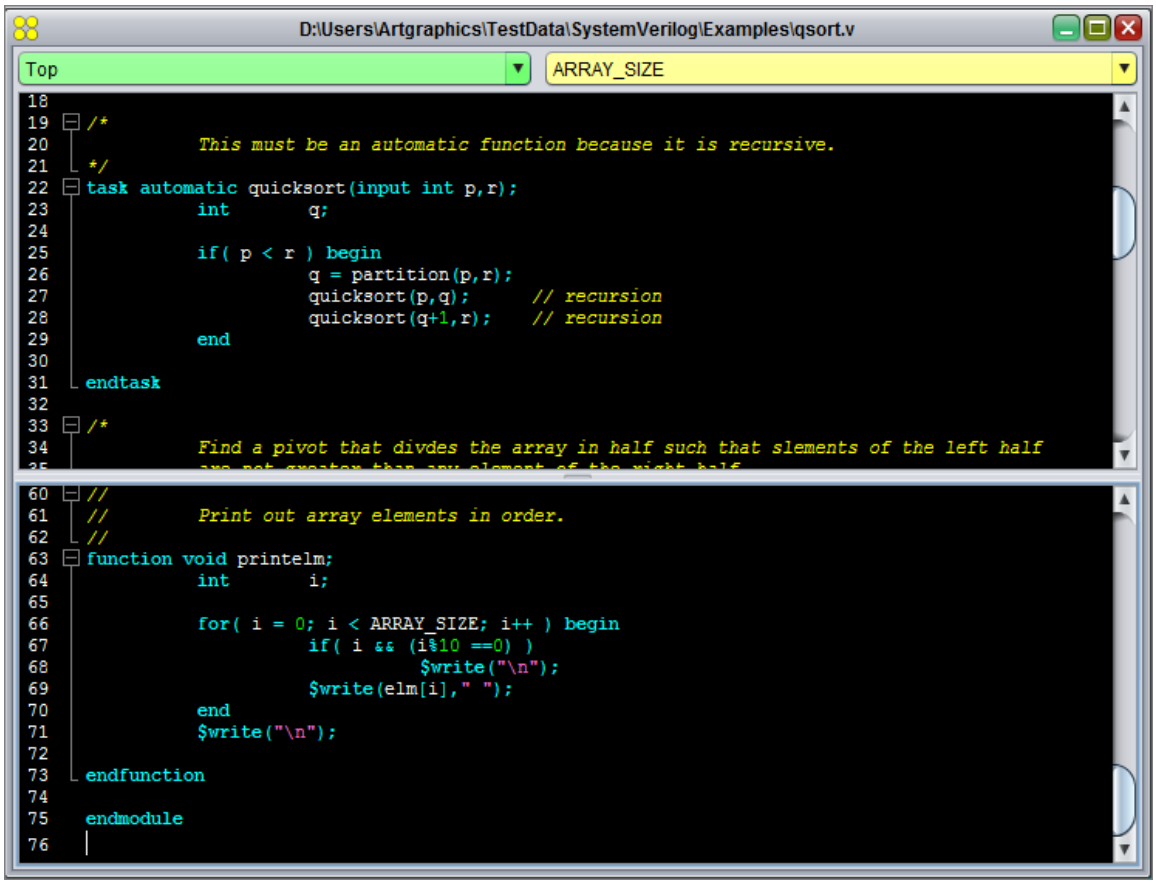
ファイルを開くと行の折り畳みが自動的に設定されますが、ユーザ自身で折り畳みを定義する事が出来ます。例えば、下図はモジュール・インスタンス全体を折り畳みとして定義した状態を示しています。



同様に、折り畳みを容易に解除する事が出来ます。

3.2.4　テキスト・エディタの画面分割

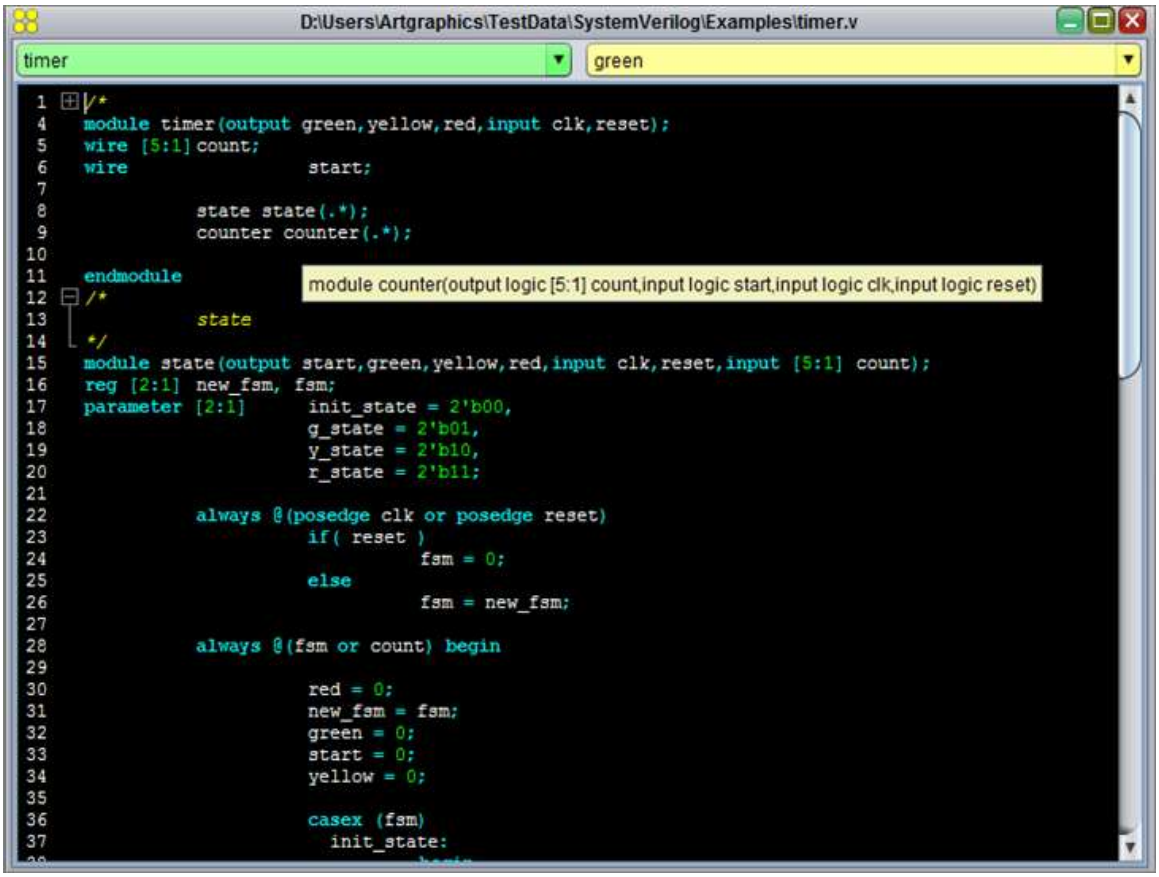
テキスト・エディタの画面を２分割する機能は、同一ファイルの異なる場所を参照する事を容易にします。



分割された個々の画面で作業をする事が出来ます。コード生成時に強力なツールとなります。

3.2.5 ヒント機能

次の図は、モジュール counter 上にマウスを静止した状態を示しています。counter の使用法がヒントとして表示されます。



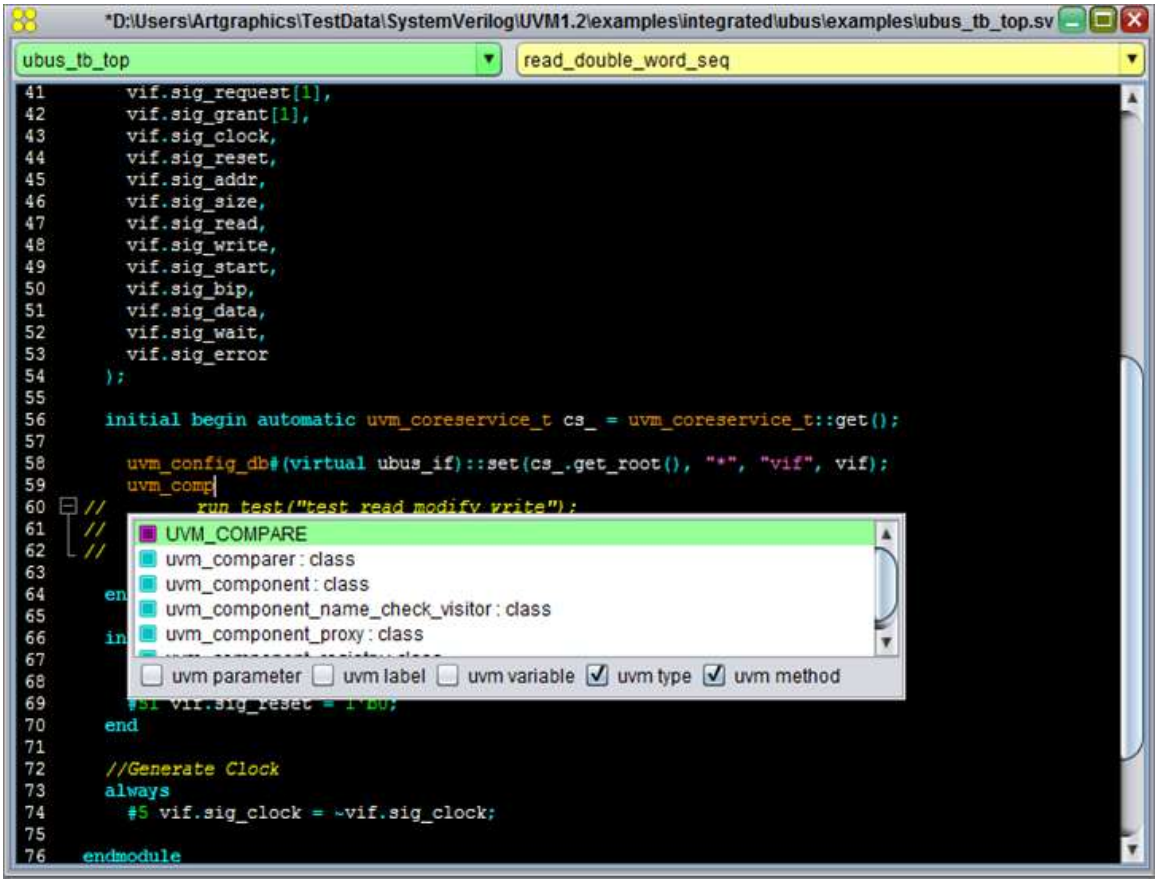
3.2.6 入力ナビゲータ

SystemVerilog は多くの機能を持つ為、コード入力時の負荷が増加します。特に、UVM を使用する場合、タイプする作業は決して楽ではありません。入力ナビゲータはその負荷を軽減する為に開発されました。

3.2.6.1 一般トークン

下図は、uvm_comp を入力した状態を示しています。図で紹介している様に、ナビゲータは考えられる全ての候補をリスト・アップしています。引き続き入力をするとリストは絞り込まれます。Enter をタイプすると現在選択されている項目が入力文字に置き換わります。

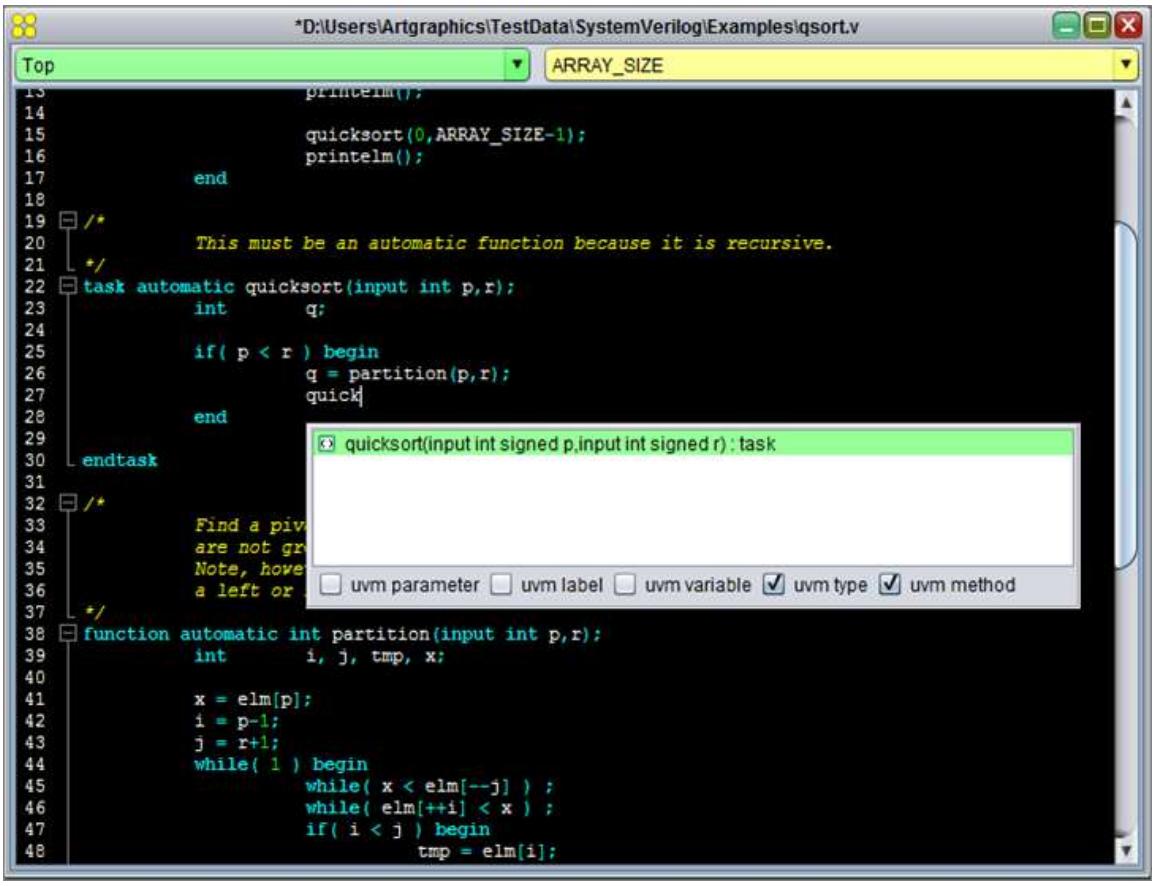
ナビゲータは入力位置の近辺に自動的に表示されます。Escape キーを押すとナビゲータは消滅します。



3.2.6.2　メソッド（タスク、ファンクション）入力

メソッド名の一部を入力した場合には、ナビゲータは完全なメソッド名、及び、引数を表示します。候補を選択して Enter をタイプするとメソッドの呼び出しが完了します。

下図は、`quick` をタイプした様子を示しています。



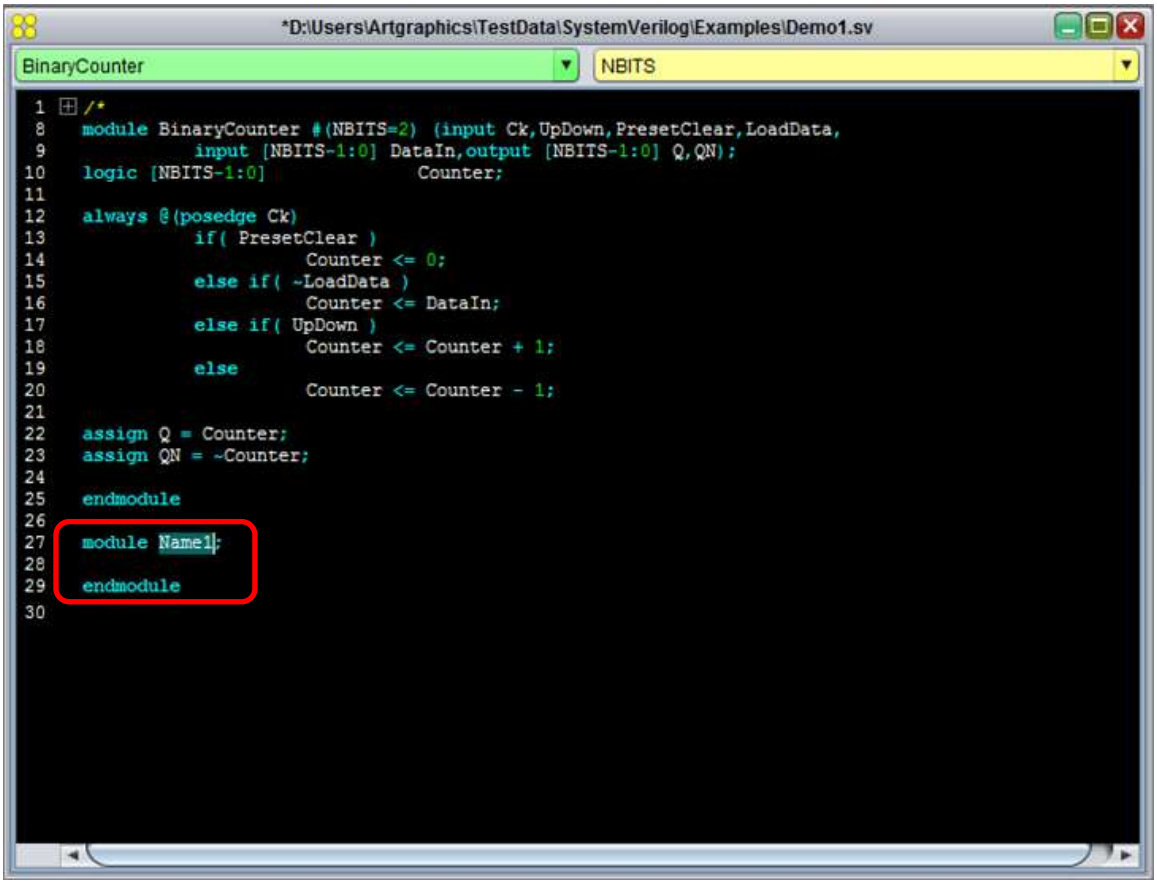
この状態で Enter をタイプすると `quicksort(p,r)` が入力されます。

3.2.7　ショートカット・キー

SystemVerilog には対として使用されるキーワードが多く存在します。以下の様なショートカット・キーが入力負荷を軽減します。

キー操作	意味
CTRL+SHIFT+A	always begin と end の対を挿入します。
CTRL+SHIFT+B	ブロック・コメント・タグ（/* */）を挿入します。
CTRL+SHIFT+C	class と endclass のキーワード対を挿入します。
CTRL+SHIFT+G	covergroup と endgroup のキーワード対を挿入します。
CTRL+SHIFT+H	checker と endchecker のキーワード対を挿入します。
CTRL+SHIFT+I	interface と endinterface のキーワード対を挿入します。
CTRL+SHIFT+L	initial begin と end の対を挿入します。
CTRL+SHIFT+M	module と endmodule のキーワード対を挿入します。
CTRL+SHIFT+N	function と endfunction のキーワード対を挿入します。
CTRL+SHIFT+P	package と endpackage のキーワード対を挿入します。
CTRL+SHIFT+R	program と endprogram のキーワード対を挿入します。
CTRL+SHIFT+S	全てのファイルを保存します。
CTRL+SHIFT+T	task と endtask のキーワード対を挿入します。
CTRL+SHIFT+U	モジュール・インスタンスを生成します。

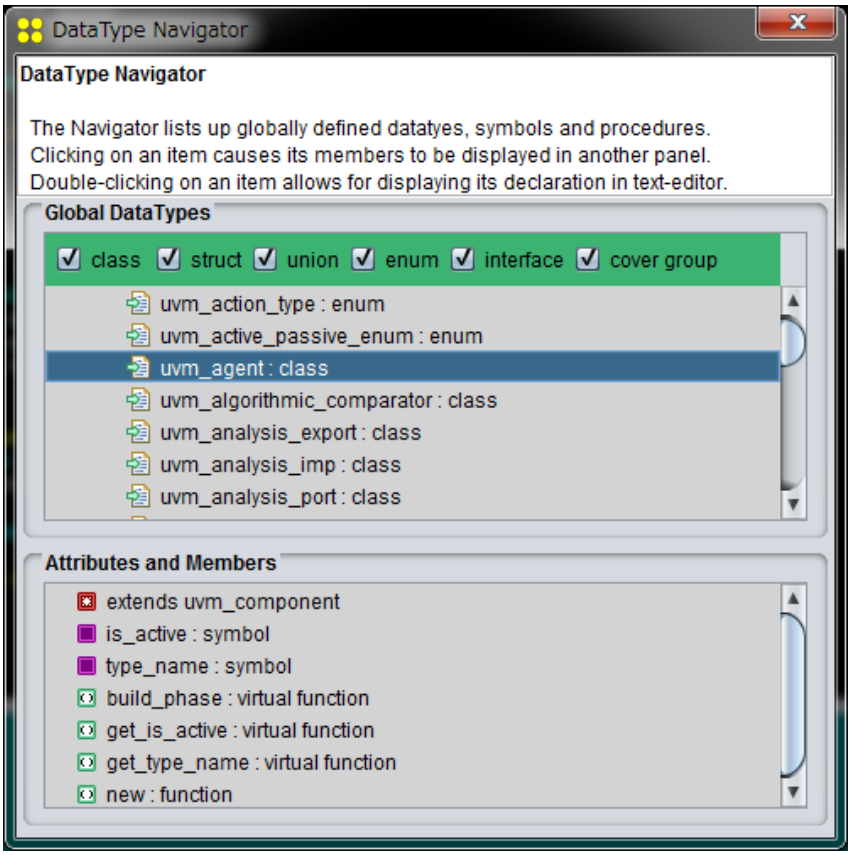
下図は、CTRL+SHIFT+M をタイプした状態を示しています。モジュール名称 Name1 を所望の名称に変更すれば、module/endmodule 構造が完了します。



3.2.8　データタイプ・ネビゲータ

強力なクロスリファレンス機能です。特に UVM を使用する場合、必須の機能です。UVM Reference マニュアルの代用としての機能があります。

下図は、データタイプ・ナビゲータを起動して、uvm_pkg を表示している状態を示しています。



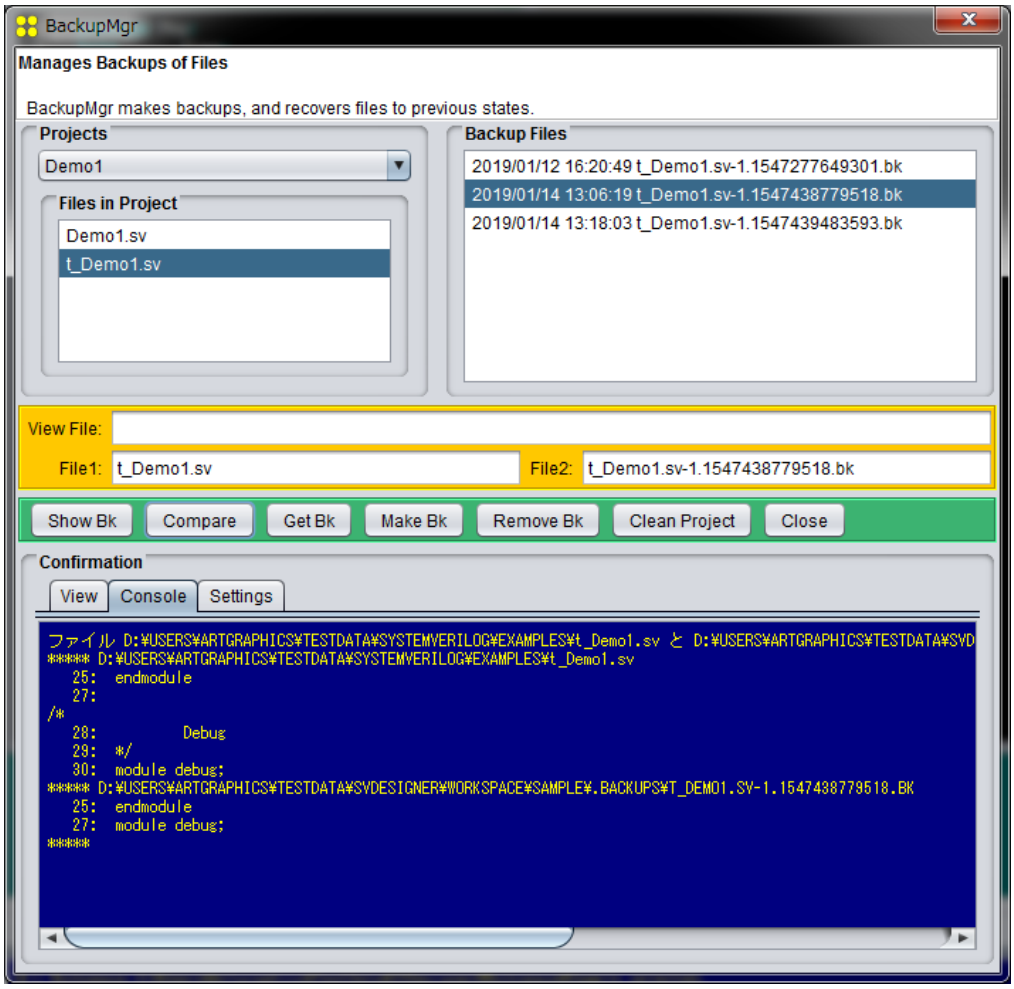
ナビゲータは、uvm_pkg に定義されている全てのクラスにアクセスする機能を提供します。それぞれのクラス名をクリックするとクラス内で定義されているメンバー、及び、メソッドが下のパネルに表示されます。クラス名をダブル・クリックするとクラスの内容がテキスト・エディタに表示されます。

3.2.9　自動バックアップ機能

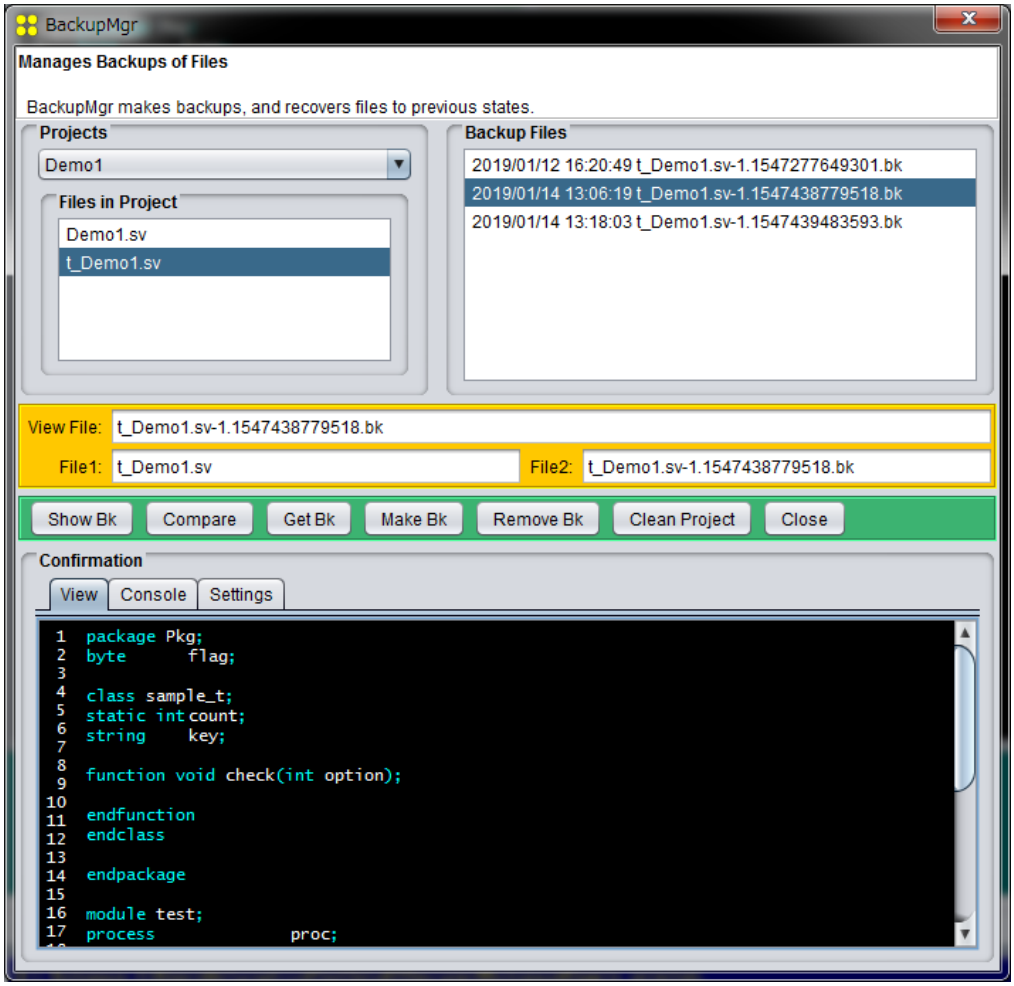
SystemVerilog IDE は自動バックアップ機能を備えています。この機能により、ファイルが壊滅状態になってもセッション開始時点のファイルを復元する事が出来ます。バックアップは自動的に行なわれる為、使用者には手間がかかりません。何よりも、バックアップを取る機会を忘れる事はありません。

バックアップはシステム固有の場所に保管されます。バックアップの作成は自動的に行なわれる為、ユーザに求められる特別な操作はありません。ユーザがバックアップ・ファイルを直接操作する為には、BackupMgr を使用します。以下の操作が可能です。

① バックアップ・ファイルの内容を表示する事
② ソース・ファイルとバックアップ・ファイルの差異を調べる事
③ バックアップ・ファイル同士の比較を行う事
④ ソース・ファイルをバックアップ・ファイルから復元する事
⑤ ソース・ファイルから新しいバックアップを作成する事
⑥ 不要になったバックアップを削除する事
⑦ プロジェクトに使用されているファイルのバックアップを整理する事
⑧ バックアップが保管されているディレクトリを知る事
⑨ バックアップが使用しているディスク・スペースの大きさを知る事

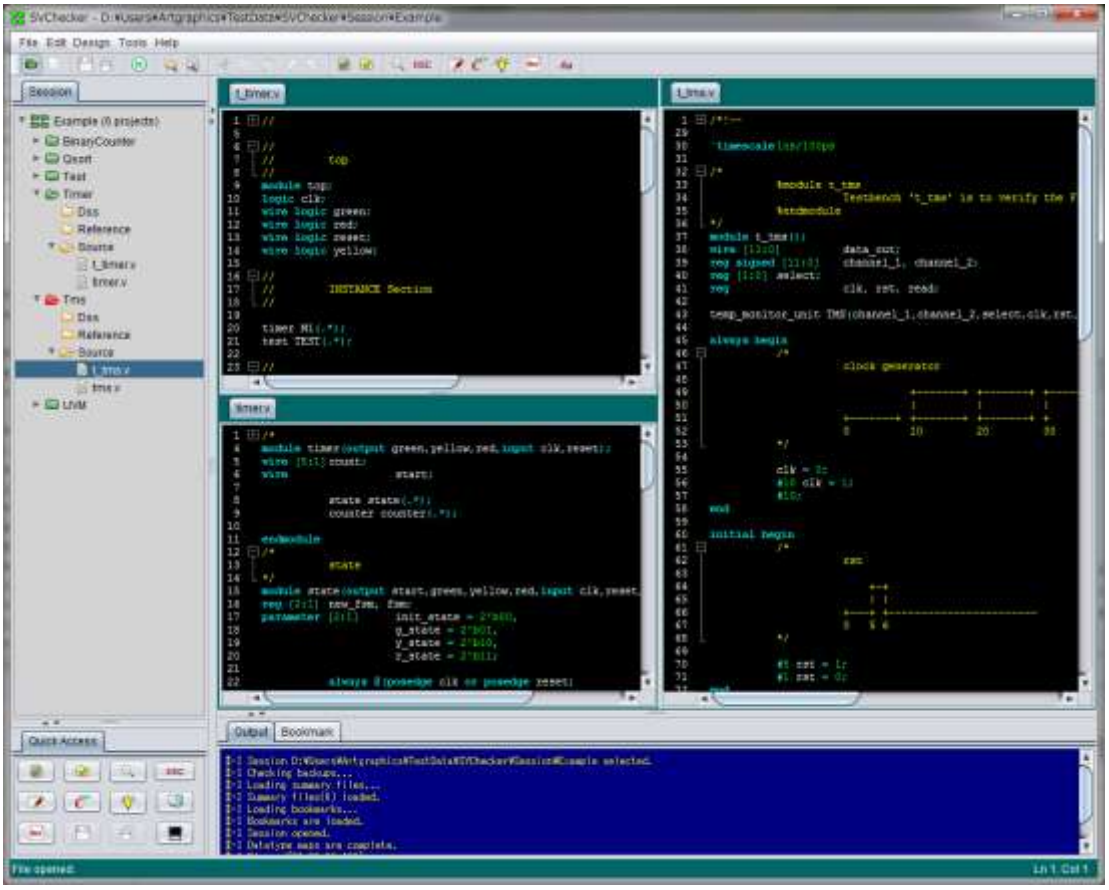


前図は BackupMgr を使用してファイルの比較をしている状態を示しています。ソース・ファイルとバックアップ・ファイルとの差異を知る事により作業履歴を思い出す事が出来ます。バックアップ・ファイルを選択して Show Bk ボタンをクリックすると、バックアップ・ファイルの内容が View パネルに表示されます。バックアップ・ファイルからファイルを復元する場合は、Get Bk ボタンをクリックするだけで済みます。また、下図は、バックアップ・ファイルの内容を表示している状態を示しています。



3.3 SVChecker

SVChecker は下図の示す様な外観を持ちます。



タブを移動する事により、テキスト・エディタの画面のレイアウトを変更する事が出来ます。タブをキャンバス外に移動するとファイルが閉じます。全て直感的な操作である為、作業が楽に進みます。

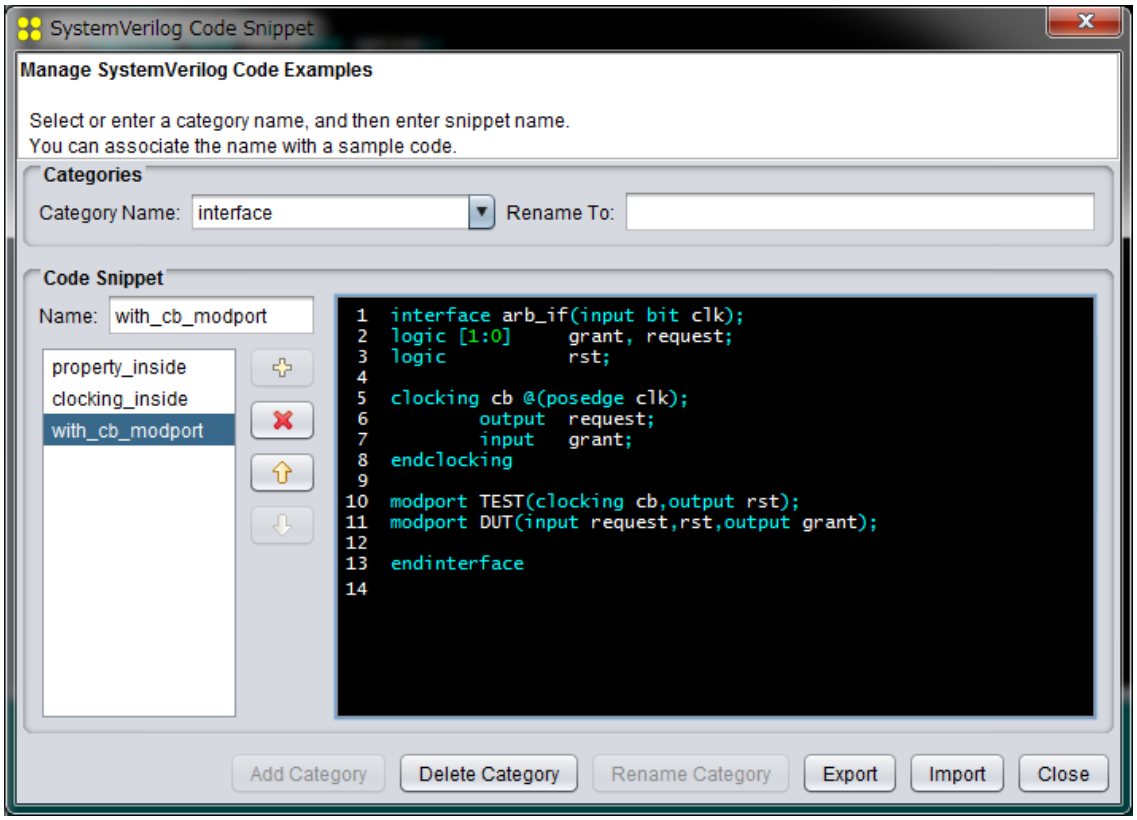
テキスト・エディタのもつ機能は SVDesigner と同じなので、省略します。

4 参照機能

4.1 コード・スニペット

スニペットは、テキスト文字列から構成された情報群を意味します。例えば、サンプル・コードの断片がスニペットに相当します。マニュアルに書かれた複雑なシンタックスを読み返すよりも過去の使用例を見た方が確実で理解が速いのが常です。

以下にスニペットの例を示します。

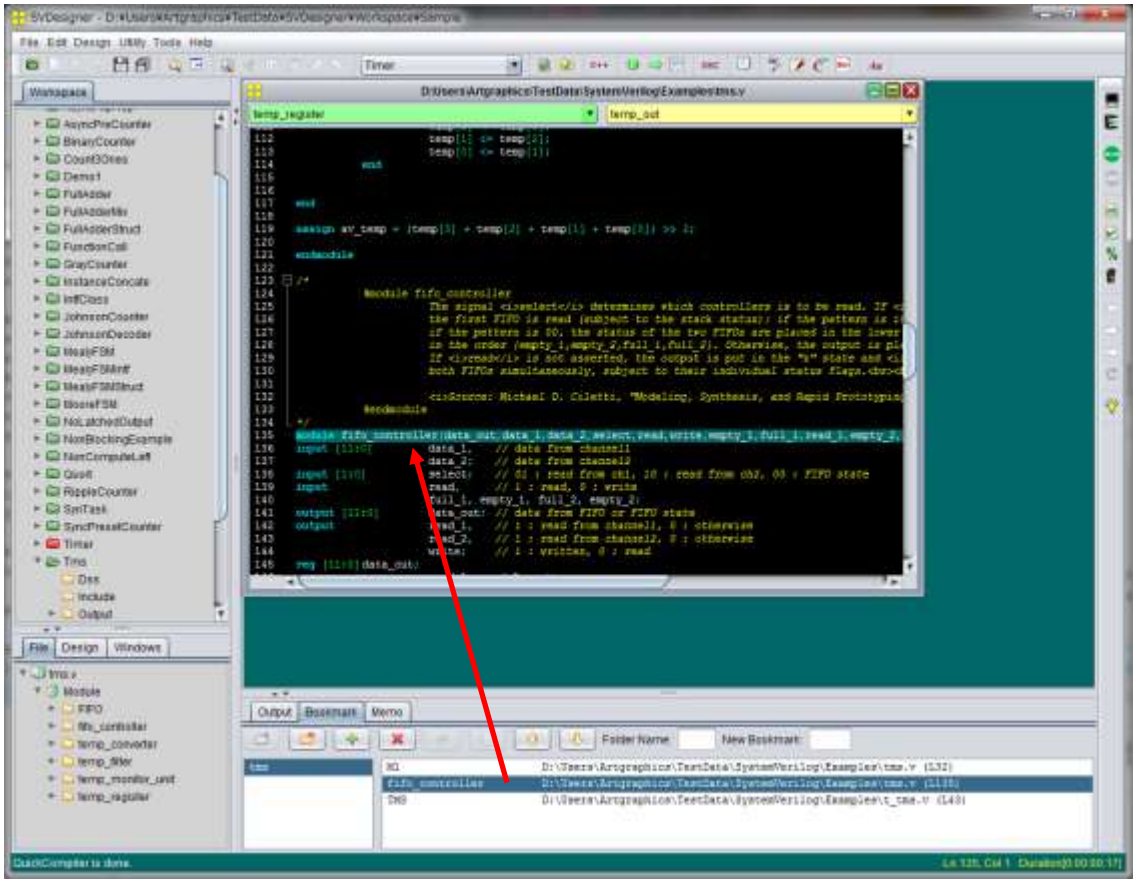


登録したコード・スニペットをソース・コードに簡単に貼り付ける事が出来ます。また、Export、及び、Import 機能を使用する事によりコード・スニペットを他の人と共有する事が出来ます。

4.2 ブックマーク

コード・スニペットはユーザ指定の名称に対してコードの断片をマップする機能です。それに対して、ブックマーク機能はユーザ指定の名称に対してソース・ファイル内の行番号を割り当てます。コード・スニペット機能、及び、ブックマーク機能の組み合わせを利用する事により、コード開発時の生産性を向上する事が出来ます。次の様に機能します。

- ブックマークをダブル・クリックするとファイルが開き、該当する行がテキスト・エディタ内に選択表示されます。
- ブックマークを目的別に分類する事が出来ます。
- ブックマークに名称を設定する事が出来ます。



頻繁にアクセスするファイルをブックマークとして登録しておくと便利です。