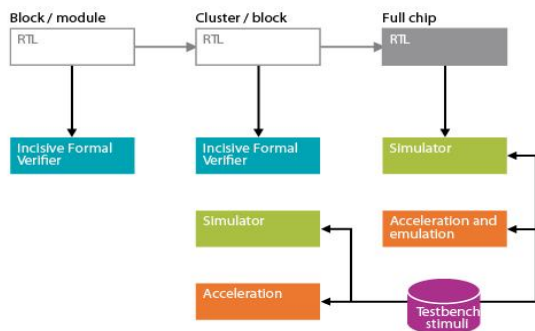


INCISIVE FORMAL VERIFIER

Incisive® Formal Verifierは、フォーマル解析を手軽に利用可能とすることで生産性向上と製品品質向上を実現するソリューションです。Incisive機能検証プラットフォームの中核部品として、完全なアサーションベース検証 (ABV: Assertion-Based Verification) フローを構成し、検証の数ヶ月前倒し実施を可能とし、リスピンの削減・Time-to-Marketの加速を実現します。

Incisive 機能検証プラットフォーム

ナノメートルスケール IC での機能検証は、速度と効率性の双方を強く必要とします。しかし、今日の分断されたメソッドでは、速度と効率性を最適化することは困難と言わざるを得ません。それぞれの検証ステージは、それぞれ独自のメソッド、ツール、モデル、ユーザ・インターフェースを持っているため、あるステージの成果物を他のステージにおいて再利用することは難しく、設計者はそれぞれのステージで、毎回、新たに設計および検証モデルを作成する必要があります。ケイデンスの Incisive 検証プラットフォームは、フォーマル解析、シミュレーション、アクセラレーションおよびエミュレーションが有機的に結合され、業界最高水準の速度と効率性を提供する統一されたメソッドをサポートした、世界初の機能検証プラットフォームです。



INCISIVE FORMAL VERIFIER

Incisive Formal Verifier は、業界最先端のフォーマル解析テクノロジーを採用し、パフォーマンス、容量、使用性において高い性能を提供いたします。実際の製品設計で確かめられた信頼性の高いこのテクノロジーにより、Incisive Formal Verifier は生産性と製品品質の両面の向上を実現します。Incisive 機能検証プラットフォームの提供する完全なアサーションベース検証ソリューションの一部として、Incisive Formal Verifier は、Incisive シミュレーション、カバレッジ、アクセラレーション、エミュレーションと同じアサーション言語をサポートしています。お客様との共同プロジェクトを通して開発した効率的メソッドを用いることで、アサーションベース検証およびフォーマル解析の導入は、最大の投資収益率をもたらします。

Incisive Formal Verifier はテストベンチを必要としないため、テストベンチ開発を待つことなく、数ヶ月前倒しに検証を開始できます。フォーマル手法によって、バグの原因特定はピンポイントに可能となり、ブロックレベルのデバッグやブロックの結合の時間を短縮できます。

フォーマル解析では、数学的手法を用いて網羅的にチェックを行うため、Incisive Formal Verifier は従来手法では検出困難であったような、極めて特殊な状況下でのみ発生するようなコーナーケースの機能的なバグを検出可能です。これらの機能により、生産性の向上、リスピンのリスクの削減、製品品質の向上、Time-to-Market の加速を実現します。

利点

- リスク削減と予測性向上による Time-to-Market の加速
- テストベンチ開発前の検証実施による生産性向上
- 従来手法では検出困難であったコーナーケース・バグを検出可能とすることによる、品質向上とリスピンの削減
- ブロック・デザインの労力削減とデバッグ工数短縮
- ブロック結合のための工数削減
- シミュレーションと同様の使用感の先進的デバッグ環境による移行・導入の容易性

特長

使い勝手よく自動化された最先端フォーマル・エンジン

実際の製品設計で確かめられた信頼性のテクノロジーをベースに、世界的に著名な開発陣によって強化拡張された複数のエンジンにより、業界最高水準のパフォーマンスおよび容量を実現しています。使用性を高めかつ最大のパフォーマンスを得るために、最適なエンジンの選択は自動的に行なわれます。また、分散プロセスによる複数エンジンの並列実行も可能であり、より高いパフォーマンスを得ることが可能です。

広範なデザイン言語サポート

Incisive Formal Verifier は、Verilog®、SystemVerilog、VHDL およびこの混在言語と広いデザイン言語をサポートしており、設計チーム・グループのデザイン文化に依存せずに使用することが可能です。構文解析器は数千の設計で証明された高い信頼性が保証されています。

広範なアサーション言語のサポートと相互互換性

Incisive Formal Verifier は、Incisive シミュレーション、カバレッジ、アクセラレーション、エミュレーションと同じアサーション言語をサポートしています。アサーション言語としては、PSL(Property Specification Language)、SVA(SystemVerilog Assertions)、OVL(Open Verification Library)の各種標準言語/ライブラリおよびオープンソースである IAL(Incisive Assertion Library)が利用可能です。

自動アサーション・エクストラクション

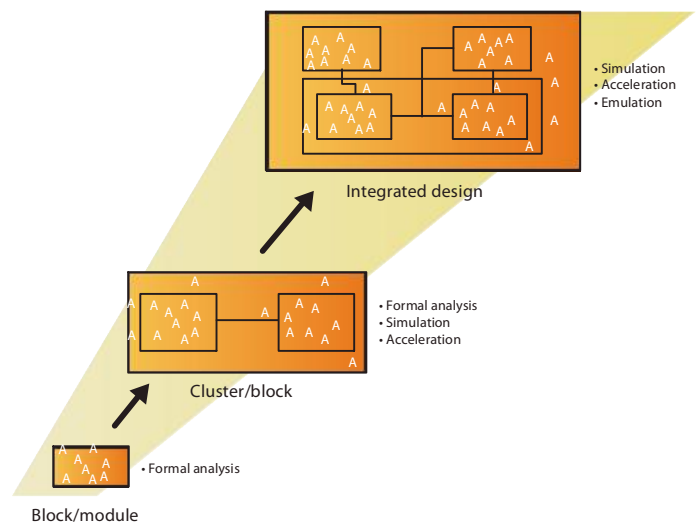
Incisive Formal Verifier は、FSM(有限状態遷移機械)の状態や状態遷移、分岐におけるデッドコード、合成プラグマなどについて、一般的なデザイン構造上のチェックを行なうためのアサーションを自動的に生成します。この機能により、設計者の方は設計スペシフィックなアサーションの記述に注力することで、全体としての品質および生産性の向上が可能となります。

シミュレーションとの相乗効果

Incisive Formal Verifier は、サードパーティーのシミュレータとの接続も可能ですが、Incisive Unified Simulator と接続したときに最大の効果を発揮します。

Incisive プラットフォームの構成ツールとして、両社の間では共通の構文解析器、アサーション、リントチェック、解析、カバレッジ、デバッグを使用しています。

さらに、Incisive Formal Verifier はブロックレベル検証のための初期化機能を組み込みで持っています。この機能は業界標準の VCD インターフェースを介しても使用可能であり、複雑な初期化が必要なデザインにおいて、シミュレーションのトレース情報を利用することができます。



強力かつ使いやすいデバッグ・解析機能

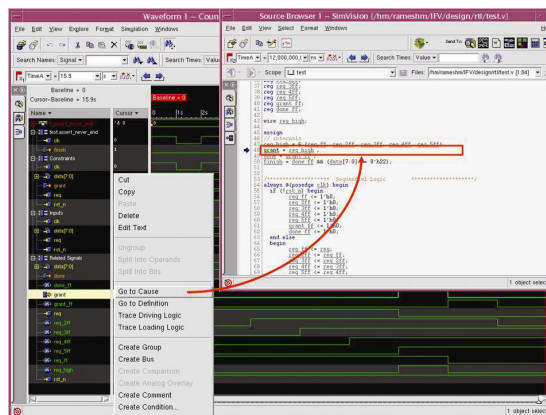
Incisive プラットフォームとして統合された GUI および Tcl インターフェースにより、Incisive Unified Simulator と同様の使用感が継承されています。このため、Incisive Unified Simulator に慣れ親しんだ設計者の方であれば、無理なく Incisive Formal Verifier を使用してデバッグ・解析を行なうことが可能です。このデバッグ・解析機能にはリントチェック、ソース・コードとリンクされた波形ビューア、ソース・コードのアノテーションおよびトレース機能、構造解析、アサーション間の矛盾チェック、カバレッジ・レポートなどが含まれています。

Incisive Formal Verifier がアサーション違反を検出したとき、デバッグ用に反例の波形図を生成します。また、このバグの確認もしくはリグレッション・テストのためにシミュレーションを行なうための簡単なテストベンチの作成も行ないます。これらの機能により、シミュレーション中心のフローに Incisive Formal Verifier フローを容易に追加できます。

フォーマル・カバレッジ

フォーマル解析を使用するとき、十分なアサーションを書いたかどうか、どれだけのアサーションがチェックされたのか、そして、証明がどの範囲にわたって行なわれたのかを知る必要があります。Incisive Formal Verifier は、効率的なフォーマル解析を行なうために、これらのすべてについてレポートを作成します。

Incisive Formal Verifier は、シミュレーションでのカバレッジの抜けを調べるための到達性解析などの機能も持っています。アサーションを用いた機能カバレッジと FSM の状態・状態遷移やデッドコードなどのコードカバレッジの両方の見地から、より大きな効果を得ることができます。Incisive Formal Verifier は、シミュレーション、アクセラレーションおよびエミュレーションでは見つけられない機能的な問題を指摘し、検証漏れを防ぎます。



COMPREHENSIVE METHODOLOGY

Incisive Formal Verifier は、単体でも強力なアサーションベース検証 (ABV: Assertion-Based Verification) が可能ですが、Incisive のアサーションベース検証 (ABV: Assertion-Based Verification) フローの中で最大の効果を発揮します。検証の効率を最適化するためには、ツールの長所を生かした検証フローを構築する必要があります。フォーマル解析メソドロジーを含む Incisive のアサーションベース検証フローは、アサーションの導入を容易にし、検証全体の工数の削減を品質向上を満たしながら実現することができます。

フォーマル解析は設計者が RTL およびアサーションを書き進めている検証フロー初期、すなわち意味のあるテストベンチが出来上がる数ヶ月前に実行されます。この手法により、機能的なバグは設計サイクルの初期段階において検出でき、より低いコストで修正可能となります。また、コーナーケース・バグは従来の検証フローでは検出が困難で取りこぼされてきましたが、フォーマル解析を用いることで容易に検出でき、リスピンのリスクを削減できます。ブロックレベルで記述されたアサーションは後の設計サイクルでのシミュレーション、アクセラレーション、エミュレーションでも再利用されます。このアサーションベース検証フローは、既存の検証フローからの移行が容易でありながら、Time-to-Market を大きく加速できるメソドロジーです。

仕様

設計言語サポート

- Verilog (IEEE 1364-1995, IEEE 1364-2001)
- SystemVerilog (IEEE 1800)
- VHDL (IEEE 1076-1987, IEEE 1076-1993)
- 上記の混在記述

アサーション言語サポート

- PSL (Property Specification Language)
- SVA (SystemVerilog Assertions)

アサーションライブラリサポート

- OVL (Open Verification Library)
- IAL (Incisive Assertion Library)

HDL 解析

- 500 以上のルールを持つリント解析
- 合成可能性
- レース条件
- コード再利用性
- FSM 符号化
- Verilog, VHDL, 混在記述サポート
- デザイン・サブセット

結果解析

デバッグおよび GUI

- アサーション・マネージャ
- 波形ビューア
- ソース・コード・ブラウザ
- ソース・コード値アノテーション
- シグナル・フロー・ブラウザ
- アサーション矛盾チェッカ
- アサーション・トリガ

レポート機能

- ステータスおよび証明範囲レポート
- アサーション・カバレッジ・レポート
- フォーマル・カバレッジ・レポート
- Cone-of-Influence 解析

インターフェース

- Tcl コマンドインターフェース
- VCD および SST2 インターフェース

プラットフォーム

- Sun Solaris
- HP-UX
- Linux

cadence®

日本ケイデンス・デザイン・システムズ社

本社 / 〒222-0033 神奈川県横浜市港北区新横浜 2-100-45
営業本部
TEL.(045)475-8410 FAX.(045)475-8415 URL <http://www.cadence.co.jp/>
〒541-0054 大阪府大阪市中央区南本町 2-6-12 サンマリオン NBF タワー16F
TEL.(06)6121-8095 FAX.(06)6121-7510

* © 2011 Cadence Design Systems, Inc. All rights reserved worldwide.
CadenceおよびCadenceロゴは、Cadence Design Systems, Inc. の登録商標です。
その他記載されている製品名および会社名は、各社の商標または登録商標です。
* 掲載の内容は、2011年8月現在のものです。



販売代理店 イノテック株式会社 IC ソリューション本部

〒222-8580 神奈川県横浜市港北区新横浜 3-17-6
TEL.(045)474-2290,2291,2293 (営) FAX.(045)474-2395
URL <http://www.innotech.co.jp/>
〒541-0054 大阪府大阪市中央区南本町 2-6-12 サンマリオン NBF タワー16F
TEL. (06)6121-7703 (営) FAX. (06)6121-7720