

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2009-518717

(P2009-518717A)

(43) 公表日 平成21年5月7日(2009.5.7)

(51) Int.Cl.	F I	テーマコード (参考)
G 0 6 F 17/50 (2006.01)	G 0 6 F 17/50 6 5 4 K	5 B 0 4 6
H 0 1 L 21/82 (2006.01)	H 0 1 L 21/82 C	5 F 0 6 4

審査請求 未請求 予備審査請求 未請求 (全 16 頁)

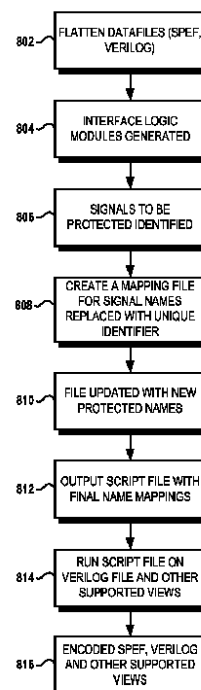
(21) 出願番号 特願2008-543551 (P2008-543551) (86) (22) 出願日 平成18年10月26日 (2006.10.26) (85) 翻訳文提出日 平成20年5月21日 (2008.5.21) (86) 国際出願番号 PCT/US2006/060261 (87) 国際公開番号 W02007/102887 (87) 国際公開日 平成19年9月13日 (2007.9.13) (31) 優先権主張番号 11/290,368 (32) 優先日 平成17年11月30日 (2005.11.30) (33) 優先権主張国 米国 (US)	(71) 出願人 504199127 フリースケール セミコンダクター イン コーポレイテッド アメリカ合衆国 78735 テキサス州 オースティン ウィリアム キャノン ドライブ ウェスト 6501 (74) 代理人 100070150 弁理士 伊東 忠彦 (74) 代理人 100091214 弁理士 大貫 進介 (74) 代理人 100107766 弁理士 伊東 忠重
--	--

最終頁に続く

(54) 【発明の名称】 EDAツール設計ビューにおける情報を保護する方法およびプログラム・プロダクト

(57) 【要約】

ある種のEDAツール設計ビュー内のモジュール名および構造といった、HDLインターフェース論理モデルにおける機微な回路設計情報(806)が、設計ビューを回路設計のシミュレーション・モデルの一部として移転するのに先立って、選択されたインスタンスおよびネット名を、無関係な一意的な識別子(808)で置き換えることによって消去される。結果として、その情報の許諾されていない使用が避けられる。IC設計の種々の設計ビューにおける信号名をエンコードする方法は、複数の設計データベースに含まれる名前のリストを提供し(806)、名前のリスト内の各名前を保護された名前に変え(808)、各設計ビュー・データベース中の関連付けられた保護された名前をそれぞれの変更された名前に置き換える(810、812、814、816)ことを含む。



【特許請求の範囲】**【請求項 1】**

IC設計の第一のビューをスキャンして名前を探す段階と；

スキャンから見出された一つまたは複数の名前のそれぞれを、関連付けられる名前に似ていない関連付けられた一意的な識別子にマッピングする段階と；

前記第一のビューにおいて、マッピングされた名前の一つまたは複数の生起を、その関連付けられた一意的な識別子で置換し、IC設計の第一の保護されたビューを生成する段階と；

IC設計の第二のまたはより多くのビューにおいて、前記マッピングされた名前の一つまたは複数の生起を、その関連付けられた一意的な識別子で置換し、IC設計の第二の保護されたビューを生成する段階とを有する、方法。

10

【請求項 2】

前記一つまたは複数の名前が、前記第一のビューにおけるインスタンス名およびネット名のうちから選択される、請求項 1 記載の方法。

【請求項 3】

各名前が、一つまたは複数のインスタンス名と一つまたは複数のネット名との間の階層的関連付けを表現する、請求項 1 記載の方法。

【請求項 4】

前記第一のビューがインターフェース論理モデルである、請求項 1 記載の方法。

20

【請求項 5】

前記第一のビューおよび第二のビューがネットリストである、請求項 1 記載の方法。

【請求項 6】

前記第一のビューおよび第二のビューがハードウェア記述言語で書かれている、請求項 1 記載の方法。

【請求項 7】

前記第二のビューがVerilogでコーディングされている、請求項 1 記載の方法。

【請求項 8】

前記第一のビューが、標準寄生交換フォーマットでフォーマットされている、請求項 1 記載の方法。

30

【請求項 9】

請求項 1 記載の段階を実行するために埋め込まれているプログラム論理を含む機械可読媒体を有する製造物。

【請求項 10】

請求項 1 記載の段階を実行する手段を有するシステム。

【請求項 11】

種々の情報を含む複数のデータベース中の信号名をエンコードする方法であって：

複数のデータベースに含まれる名前のリストを提供する段階と、ここで、前記複数のデータベースの各データベースはIC設計に関係する異なる情報を有しており；

名前の前記リスト内の各名前を保護された名前に変える段階と；

40

前記複数のデータベースの各データベース中の関連付けられた保護された名前ですべての変更された名前を置き換える段階とを有する、方法。

【請求項 12】

前記名前のリストがインスタンス名およびネット名を含む、請求項 11 記載の方法。

【請求項 13】

前記名前のリストに含まれる一つまたは複数の名前が、一つまたは複数のインスタンス名と一つまたは複数のネット名との間の階層的関連付けを表現する、請求項 11 記載の方法。

【請求項 14】

50

前記複数のデータベースがインターフェース論理モデルを含む、請求項 1 1 記載の方法。

【請求項 1 5】

複数のデータベースがネットリストを含む、請求項 1 1 記載の方法。

【請求項 1 6】

前記複数のデータベースがハードウェア記述言語で書かれている、請求項 1 1 記載の方法。

【請求項 1 7】

前記複数のデータベースのうち少なくとも一つのデータベースが標準寄生交換フォーマットでフォーマットされている、請求項 1 1 記載の方法。

【請求項 1 8】

前記複数のデータベースのうち少なくとも一つがVerilogでコーディングされている、請求項 1 1 記載の方法。

【請求項 1 9】

請求項 1 1 記載の段階を実行するために埋め込まれているプログラム論理を含む機械可読媒体を有する製造物。

【請求項 2 0】

請求項 1 1 記載の段階を実行する手段を有するシステム。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本発明は、概括的にはデジタル回路のための計算機援用設計の分野に、より詳細には電子回路設計自動化 (EDA: electronic design automation) モデルに含まれる設計情報の保護に関する。

【背景技術】

【0 0 0 2】

多層深さのサブミクロンの相補型金属 - 酸化物半導体 (CMOS) 技術のようなシリコン技術における現代の進歩は、より大きくより複雑な設計が単一チップ上に形成されることを許容し、それにより同じシリコン基板上に完全な「システム・オン・チップ (SOC)」設計が配置されることを可能にするのに与っている。これらの複雑な設計は、たとえば、一つまたは複数のプロセッサ・コンポーネント、デジタル信号処理 (DSP) コンポーネント、メモリ、いくつかの通信インターフェースおよびグラフィック・サポート・コンポーネントを組み込んでいてもよい。

【0 0 0 3】

SOC設計の到来は電子システム産業を、既存の回路ブロックまたは「コア」の使用に基づく新たな設計パラダイムへとシフトさせている。このパラダイムのもとでは、システムは、一つまたは複数の既存の回路ブロックを同じシリコン基板上に統合することによってまとめられる。IC設計者にとって、新しいIC設計のたびに必要なハードウェアすべてを設計し直すよりも、すでに開発されているコンポーネントを組み込むほうが効率的かつ実際的である。このシフトからの自然な帰結として、異なるチップ設計ベンダーが既存の回路ブロックを、システム・オン・チップ (SOC) を開発するシステム・レベルの統合者または製造業者に供給するようになった。しばしばこれらのコアは、IC統合者以外のそのようなベンダーまたはグループによって所有される知的財産 (IP) に関わる。

【0 0 0 4】

市場の需要により、電子システム製造業者はシステムをより迅速かつ効率的に開発するよう迫られるので、電子システム統合者または製造業者にとって、SOCのシミュレーションおよび設計統合を許容するためにコアの設計モデルを受け取ることが有用である。IC設計シミュレーション・モデルは、IC設計者が、あるコンポーネントが別のコンポーネントと協働して設計者のニーズを達成するかどうかを知ることが許容する。設計および方法論の複雑さが増すにつれ、可用性 (usability) と統合 (integration) を許容するためには

10

20

30

40

50

、コア設計へのますます多くの詳細および可視性 (visibility) が要求される。残念ながら、SOC統合者に移転されるモデルが精確で詳細になるほど、それらの設計の盗用および不許諾操作といった濫用のリスクが大きくなる。

【0005】

コア設計およびモデルは、物理的には記憶デバイス上のデータまたはコードとして存在するので、コピー、偽造および再設計するのが比較的簡単である。さらに、コアの存在によってもたらされる効率向上も、それらのアイテムの不許諾使用、再使用、移転または販売の誘因となる。結果として、設計および方法論の複雑さからはSOC設計者によるコアの可用性を許容するための著しい量の情報の移転が要求される一方、SOC設計に参加する顧客、ベンダーおよびサードパーティーに移転されるIPを保護するための努力を払うことが

10

【発明の開示】

【発明が解決しようとする課題】

【0006】

シミュレーション・モデルが設計の動作および機能に対するそのような貴重な洞察を提供するので、当該設計のIPを保有する者は典型的には、共同する当事者間で念入りに起草された法的合意なしにはシミュレーション・モデルを開示しない。SOC設計者がある設計または発想を試したいとすると、そのような合意を取り付けることが、迅速で、効率的で、コスト効率のよい設計開発の著しい障害となる。さらに、権利保護されたコアの不正なコピー、偽造、移転または再使用を検出することが難しいため、そのような法的手段そのものの有用性も限られている。特定のユーザーによってコアが不正に取得されたかどうか、あるいはある回路設計内のどのような機微な情報がその電子的な形で入手されたのかを判別するのはきわめて困難でありうる。したがって、コアの提供者は、コアの設計、開発および入手に費やされた資源からくる恩恵を奪われることがないよう、自分たちの設計を保護する効果的な方法を必要としている。

20

【0007】

本発明について、以下の記述では図面を参照して好ましい実施形態の形で述べる。図面において、同様の番号は同じまたは類似の要素を表す。

【0008】

本発明の例示的な実施形態の以下の詳細な説明では、該説明の一部をなす付属の図面が参照される。図面は本発明が実施されうる個別的な例示的な実施形態の図解として示される。

30

【発明を実施するための最良の形態】

【0009】

本発明の例示的な実施形態の以下の詳細な記述では、本発明が実施されうる個別的な例示的な実施形態が、当業者が本発明を実施できるようにするのに十分な詳細さで記述される。他の実施形態が利用されてもよく、本発明の精神または範囲から外れることなく論理的、アーキテクチャ的、プログラムの、機械的、電気的およびその他の変更がなされてもよいことは理解されるものとする。したがって、以下の詳細な記述は、限定する意味に解釈されるべきではなく、本発明の範囲は付属の請求項によってのみ定義される。特に、当業者には、ハードウェア記述言語 (HDL) 表現または他の記号表現を使ってIC設計またはIC設計のシミュレーション・モデルIC設計を表現する任意の仕方が、本発明の範囲内で考えられることは明らかであろう。

40

【0010】

ここに記載される一つまたは複数の実施形態において、回路設計のシミュレーション・モデルの一部として移転される機微な回路設計情報を保護し、その結果としてその情報の不許諾使用を回避する方法およびプログラム・プロダクトが提供される。ここで、機微な情報とは、ある種のEDAツール設計ビュー (EDA tool design view) 内でのモジュール名および構造といったものである。このようにして、本発明の方法およびシステムは、シミュレーション・モデル内に含まれる秘匿情報へのアクセスを許可することなく、SOC設

50

計者が、シミュレーション・モデルのコアおよび諸コンポーネントの動作が該設計者のニーズを満たすかどうかを判別するためにコア設計のシミュレーション・モデルを使うことを可能にする。これは、普通ならシミュレーション・モデルに具現されるかもしれない機微なコア設計情報を保護するためのSOC設計者（または会社）とIP保有者との間の秘匿性合意への依存を、部分的には軽減する

設計プロセスの一部として、典型的には、仮想コンポーネント・ブロックが生成される。設計プロセスは、特定の回路設計のために所望される抽象化のレベルに依存していくつかのステップを含むことができる。チップ設計者はしばしば、設計プロセスを支援し、プロトタイプ作成や生産に先立ってチップ設計のシミュレーションを許容するための電子回路設計自動化（EDA）ソフトウェア・ツールを使用する。EDAソフトウェア・ツールを使ったチップ設計は一般に、チップ設計が徐々に完成されていく逐次過程を含む。典型的には、チップ設計者は、コンピュータ・ワークステーションにおいて情報を入力することによって回路を構築する。コンピュータ・ワークステーションは一般に、必要とされる回路設計の諸部分を表示するよう高品質グラフィック機能を有している。普通、たとえばVerilogまたはVHDLのようなハードウェア記述言語（HDL）を使ってトップダウン式の設計方法論が用いられる。これによると、設計者は回路の機能コンポーネントを階層的に定義することによって集積回路を作成し、各コンポーネントをどんどん小さなコンポーネントに分解していく。

【0011】

最初に、集積回路のさまざまなコンポーネントが、その機能的な動作ならびに関連する入力および出力によって定義される。これらの設計状態の間、設計者は一般に、かなりの階層情報を使って回路を構築し、典型的には設計において相当な規則性を与える。デジタル回路の設計は、コンピュータ・システムのメモリ内で種々の方法で表現または指定されることができる。デジタル回路表現の各型はビュー（view）と呼ばれる。異なるビューは、回路構造に関する異なる量および型の情報を含む。回路構造に関するより多くの情報を含むビューは、より多くのコンピュータのメモリを必要とし、構築および操作するためにより多くのコンピュータの時間を必要とする。

【0012】

HDLまたは他の高レベルの記述から、典型的には論理合成によって、実際の論理セル実装が決定される。論理合成は回路の機能的な記述を特定の回路実装に変換する。次いで論理セルは「配置され（placed）」（すなわち、回路レイアウトにおいて特定の座標位置を与えられる）、「経路形成される（routed）」（すなわち、設計者の回路定義に従って結線または相互接続される）。配置および経路形成ソフトウェア・ルーチンは一般に、論理合成プロセスによって生成されたフラットにされたネットリストを入力として受け容れる。このフラットにされたまたは階層的なネットリストが、目標標準セル・ライブラリから特定の論理セル・インスタンスを同定し、特定のセルからセルへの接続性を記述する。

【0013】

ICチップは、半導体基板の表面上に形成されたセルおよびセル間の接続を含む。セル（cell）とは、トランジスタ、コンデンサおよび他の基本回路素子といった一つまたは複数の回路素子が、ある機能を実行するようにまとめられたグループである。ICは多数のセルを含み、セル間の複雑な接続を要求することもある。ICのセルのそれぞれは、一つまたは複数のピンを有しうる。各ピンは今度はICの一つまたは複数の他のピンに導線で接続される。ピンをICに接続する導線も、チップの表面上に形成される。

【0014】

ネット（net）とは、二つ以上の相互接続されたピンおよび導線の集合であり、それによりピンを有する論理回路を接続する。典型的なチップはさまざまな組み合わせで接続される必要のある何千、何万または何十万のピンを有するので、チップはまた、何千、何万または何十万の「ネット」すなわちピンの集合の接続の定義をも含む。あるチップについてのネットの数は典型的にはそのチップ上のセルの数のオーダーと同じオーダーである。ネットの大半は接続されるべき二つのピンしか含まないことも普通だが、三つ以上のピン

を含むネットも多い。いくつかのネットは接続されるべき何百または何万のピンを含むことがある。

【0015】

ICモデル（IC model）とは、典型的にはHDLで表現されている相互接続された回路セルおよびネットの集合であり、本稿では時に交換可能にモジュール（module）、コア（core）、知的財産（IP: intellectual property）、IPブロック、ブロックまたはコンポーネントと称される。コア（core）とは、ハードウェアで実装可能な電子回路設計であり、典型的にはHDLで指定され、また典型的には最上位のネットリストで表現される。「ネットリスト（netlist）」とは接続されるピンの名前を含むネット（「ネット名」）のリストまたはセルのピンに接続するネットの名前を含むセル（「インスタンス名」）のリストである。最上位のRTLは、より低いレベルの抽象化でゲート・レベルのネットリストを生成するために合成される。設計の合成後、ゲート・レベルのネットリストが検証され、回路のレイアウトが決定され、ICが製造される。

10

【0016】

今日、ほとんどのデジタル集積回路（IC）の設計は、HDL（ハードウェア記述言語）方法論に基づく高度に構造化された過程である。HDL方法論は、ICが仕様文書によって規定されることで始まる。次いで、IC仕様が、レジスタ転送言語（RTL: Register Transfer Language）と称される設計抽象化のレベルで高レベルのHDLコードに書かれる。RTLコードは、回路をレジスタ、ブール式および「if-then-else」文や複雑なイベント・シーケンスのような制御論理の集合として記述する。RTLコードは典型的には、RTLレベルの抽象化でVerilog-HDL（「Verilog」）またはVHDLのようなHDL言語を使って実装される。ここでは、IC設計は、回路入力、出力およびクロックされるレジスタの間に流れるデータに対して実行される動作を記述することによって、規定される。RTLレベルにおいて、設計者は、設計階層性およびパーティション分割、クロック方式、リセット方式およびレジスタ位置といったすべての主要な設計上の決定をしなければならない。それらのすべての決定は、RTLコードに含まれ、反映される。RTLコードは一般に技術から独立であり、設計ツールからも独立である。

20

【0017】

RTLコードによって表現されるIC設計は次いで合成されて、ゲート・レベルの記述すなわちネットリストを生成する。合成（synthesis）とは、RTLコードによって表現される設計のアーキテクチャ的および機能的な記述を、論理レベルおよびゲート・レベルの記述のような、設計のより低レベルの表現に翻訳するために行われるステップである。すなわち、仕様およびRTLコードは一般には当該設計を実装するために使われる厳密なゲートまたは論理デバイスを規定しない。IC設計のゲート・レベルの記述は技術依存でありうる。これは、合成プロセスの間に、合成ツールが、技術独立なRTLコードを技術依存のゲート・レベルのネットリストにマッピングするために所与の技術ライブラリを使用することがあるからである。しかしながら、ネットリストは汎用でも技術固有でもよい。汎用ネットリストは、技術固有のセル・ライブラリに相関されていない、RTLコードから生成されたネットリストである。技術固有のネットリストまたはマップ・ネットリストは、IC設計が特定の技術固有のセル・ライブラリにマッピングされたあとに作成されたネットリストである。

30

40

【0018】

ここに記載される本発明の諸実施形態は、システム・レベルのマクロ、メガセルまたはシステム・チップ設計において使用される埋め込まれたソフトウェア「コア」のいずれに関係するものであれ、幅広い多様な仮想コンポーネント・ブロックに適用可能である。仮想コンポーネント・ブロックは一般に、設計段階における抽象化のレベルと同じくらいその「ファームさ（firmness）」に従って分類できる。仮想コンポーネント・ブロックはたとえば、「ソフト」「ファーム」および「ハード」の三つの範疇のうちの一つに分類されてもよい。ファームさの範疇は、機能的な指定から具体的な製造プロセスまでの範囲で開発段階を特徴付ける。

50

【0019】

「ソフト」仮想コンポーネント・ブロックは典型的には、SOC統合者に、合成可能なハードウェア記述言語（たとえばVHDLまたはVerilogのような）の形で届けられる。ソフト仮想コンポーネントは高度に柔軟であり、汎用かつ構成可能でありうる。「ファーム」仮想コンポーネント・ブロックは主として、汎用技術のために最適化された構造型（structural-type）モデルである。ファーム仮想コンポーネント・ブロックの詳細のレベルはたとえば、RTLサブブロックの領域配置が指定される回路設計から、相対的に配置されたデータ経路を介して、完全に配置されたネットリストを有する回路設計までの範囲にわたる。ファーム仮想コンポーネント・ブロックは一般に経路形成を含まず、目標技術を横断して移植可能である。「ハード」仮想コンポーネントは個別の技術にマッピングされており、一般に目標技術におけるパワー、サイズおよびパフォーマンスのために最適化されている。ハード仮想コンポーネントの例は、カスタム物理レイアウトまたは完全に配置され、経路形成され、所与の技術のために最適化されたネットリストを含みうる。ハード仮想コンポーネント・ブロックはプロセスまたはベンダー固有であり、通例、GDSIIのような標準的な設計言語で表される。ユーザーは一般に、ハード仮想コンポーネント・ブロックを効果的に使うために、GDSIIファイルと一緒に、高レベルの行動モデルおよび完全な物理的／タイミング・モデルへのアクセスを与えられる。

10

【0020】

SOCの設計および解析のタスクを単純化するため、複雑な設計はブロックに、すなわちチップ内の任意のユーザー定義またはシステム定義の階層に分割できる。この際、ブロックはインターフェース・モデルで置換できる。有利には、そのようなモデルは、たとえばブロックのインターフェース情報をブロックのSOC統合者と交換するコンパクトな手段を提供できる。あるブロックまたはコアについて完全なゲート・レベルのネットリストの代わりにインターフェース・モデルを使用することは、特にゲート数が500万ゲートを超える複雑な設計については、チップ・レベルの解析およびシミュレーションに関連するパフォーマンスおよび容量を改善することに向けた鍵である。さらに、そのようなインターフェース論理モデルは、チップ設計者が、インターフェース論理モデル（ILM: interface logic model）と一緒に移転されることが必要とされない内部的なゲート・レベル設計を保持し、保護することを許容する。

20

【0021】

自動モデル生成は、ILMの使用を含むことができる。ILMは、設計上の内部的なレジスタ間経路に関連するネットリスト、制約およびバックアノテーション（back-annotation）を破棄し、それにより精度を犠牲にすることなく著しい実行時間およびパフォーマンスの改善をもたらす。さらに、ILMは元のネットリストのインターフェース論理のみを含んでいる。たとえば、あるブロック上のレジスタ間経路のみに含まれる論理はILMには入れられていない。ILMは、入力ポートで始まり、エッジ・トリガーされるレジスタまたは出力ポートのいずれかで終わるタイミング経路（すなわちシーケンシャル論理）に含まれる諸セル（すなわち、論理および関連するネット（ここではインターフェース・ネットと呼ばれる）の組み合わせ）を含む。そのようなタイミング経路において透明なラッチに遭遇する場合、それは組み合わせ論理として扱われ、経路追跡はラッチを通過して、エッジ・トリ

30

40

【0022】

ILMは、設計のインターフェース・タイミングのシミュレーション・モデル表現として使うことができる。ILMモデルは、設計Verilogネットリストおよび適切な寄生情報から生成できる。コンポーネントのILMモデルは、そのコンポーネントがどのように動作し、他のコンポーネントとタイミングに関して相互作用するかの詳細な理解および探求を可能にする。

【0023】

ここで図面を参照し、特に図1を参照すると、本発明のある実施形態のブロック図が示されている。IC設計情報（Verilog設計）102はVerilogで実装されている。IC設計寄生

50

情報 (SPEF) 1 0 4 は IC 設計情報 1 0 2 に関連しており、標準寄生交換フォーマット (SPEF: Standard Parasitic Exchange Format) である。Verilog 設計 1 0 2 および SPEF 1 0 4 は一緒になって、コアまたは設計ブロックのための設計情報の重要な集合を提供する。標準的な EDA ツールを使って、SPEF 1 0 4 は ILM 1 0 6 に変換でき、Verilog 設計 1 0 2 は Verilog ILM 1 0 8 に変換できる。ILM を生成するとき、EDA ツールは、回路シミュレーション、タイミング検証ならびに回路配置および経路形成のために EDA ツールによって時に使われるようなフラットなゲート・レベルのネットリストをフォーマットする。SPEF ILM 1 0 6 および Verilog ILM 1 0 8 のそれぞれは、回路設計のモデル・ビューを表現する。理解されるであろうように、各モデル・ビューは、当該モジュールの動作および機能に対する洞察を提供しうる秘匿シミュレーション情報を含んでいる。

10

【0024】

図 2 は、フラットなゲート・レベルの SPEF の例である。ネットリスト 2 0 0 は、名前マップ 2 0 2、ポート寄生情報 2 0 4 ならびにネット接続および寄生情報 2 0 6 を含んでいる。

【0025】

図 3 は、Verilog ビューにおけるモジュールまたはコア設計の例を示している。フラットなゲート・レベルのネットリスト 3 0 0 は、各「インスタンス名」、各「ネット名」およびインスタンス名とネット名との間の階層的関連付けを定義する。インターフェース 3 0 2 は、モジュールの最上位レベルで入力および出力に接続するネットを同定する。ネット名 3 0 4 は、モジュールの最上位レベルで入力や出力に接続しないネットの各インスタンスを同定する。インスタンス名 3 0 6 は、ビュー内のインターフェース・セルの各インスタンスを同定する。図 4 は、インスタンス名 3 0 6 のフォーマットを、インスタンス名 4 0 2、複数の入力 4 0 4、4 0 6 (またはもっと) および出力 4 0 8 によって同定されるものとして示している。

20

【0026】

図 1 に戻って参照すると、設計保護システム 1 1 0 は、シミュレーション・モデル、コアおよび／またはコンポーネントの権利保護された部分の外部設計者への開示を遮蔽しつつ、その設計者に該コンポーネント、コアまたはモデルが設計目標に合うかどうかについての理解を与える。ある好ましい実施形態によれば、設計保護システム 1 1 0 は、IC モデルに含まれるインスタンス名およびネット名ならびに可能性としては関連付けられた階層情報を定義し直し、それにより秘匿情報を保護する。設計保護システム 1 1 0 は、一貫した仕方ですべての設計ビューにわたって個別的なデータを定義し直す。設計の第一のビューは、該ビュー内のある種のインスタンス名およびネット名を定義し直すために暗号を使うことによってエンコードされる。インスタンス名、ネット名またはフラットなネットリストにおける階層的関連付けが完全な論理経路を同定しているとき、これらは、そのインスタンスまたはネットについての単一の一意的な識別子で置換される。次いで、複数のビューにわたって一貫性を維持するため、関係する相補的なビューが厳密に同じ仕方でエンコードされる。

30

【0027】

ある好ましい実施形態では、設計保護システム 1 1 0 内のスクリプトが保護された SPEF ILM 1 1 2 および保護された Verilog ILM 1 1 4 を生成する。これらは、機微な情報または秘匿情報の一部または全部をはぎ取られているが、それでもサードパーティー・ユーザーにモデルとの完全な相互作用をもつことを許容する。保護された SPEF ILM 1 1 2 および保護された Verilog ILM 1 1 4 は、標準的な EDA ツールと完全に互換なフォーマットでのコアのシミュレーションにつながる。

40

【0028】

図 5 は、本発明のある好ましい実施形態に基づくビュー 1 0 2、1 0 4 または ILM 1 0 6、1 0 8 のそれぞれにおいて見出されるインスタンス名、ネット名および階層的関連付けをエンコードするための設計保護システム 1 1 0 内のスクリプトによって実行される置換法のマッピングを示している。記載の目的のため、ILM 1 0 6、1 0 8 (例示の目的の

50

ため、例示的なビュー 200、300 から導出されたと考えられる) を保護する設計保護システム 110 が記述されるが、設計保護システム 110 は、たとえばビュー 102、104 を含め、ほとんど任意の設計ビューを保護するよう動作可能である。マッピング・キー 500 の各行は、SPEF 104 内のインスタンス名を指定する識別子 502 (最初の二つの「!」の間として定義される) および識別子 502 を置換するエンコードされた識別子 504 (二番目と三番目の「!」の間として定義される) を有する。たとえば、行 506 のインスタンス名 502 (すなわち、名前マップ 202 の行「*12」に見出される「and_gate_3」) は、行 506 に示されている一意的な識別子 (すなわち、「JWW12JWW」) で置換される。この例では、エンコードされた識別子 504 は、SPEF ILM 106 (SPEF 104 から変換後の) においてそのインスタンス名が同定されている行を同定し、その前または後に所定の文字を付加することによって作成された。当業者は、ネットリスト中のインスタンス名、ネット名および階層的関連付けが一意的かつ共有されていない識別子を与えられる限り、いかなる名付けまたはエンコード方式を使うこともできることを理解するであろう。マッピング・キー 500 は、エンコードされたインスタンス名およびエンコードされていないインスタンス名の両方を使ったネット名のエンコードをも提供する。たとえば、図 5 で行 508 に示されているように、以前に名前を変更されたインスタンス「JWW12JWW」とセル「nand_gate」を接続するネットを同定するために、一意的な識別子「JWW17JWW」が生成されている。

10

【0029】

ある好ましい実施形態では、設計保護システム 110 は、エンコードされるべきインスタンス名およびネット名を同定するために、ある名前マップ (SPEF 200 内の名前マップ 202 から導出される) について SPEF ILM 106 をスキャンするスクリプトによって実装される。該スキャンから、設計保護システム 110 は、ネットリストに含まれる名前または他の秘匿情報に無関係な命名法を使って、各同定された名前を一意的に同定するためのエンコード・マップ 500 を生成する。設計保護システム 110 は次いで、完全に機能するが、選択された機微な情報または秘匿情報がネットリストからはぎ取られている ILM 設計ビューとして、SPEF ILM 112 を生成する。

20

【0030】

図 6 は、本発明の好ましい実施形態に基づく、保護された SPEF の例を示している。この例では、図 2 の例における名前マップ 202 の行「*12」～「*17」が秘匿情報であると判別されており、保護された SPEF 600 を生成するにあたって、マッピング・キー 500 を使って、一意的なコード 602 にエンコードされている。

30

【0031】

次いで設計保護システム 110 は、同じマッピング・キー 500 を使ってコア設計の他のすべての関連する設計ビューを保護することに進む。ある好ましい実施形態では、設計保護システム 110 は、マッピング・キー 500 を使って Verilog ILM 108 から保護された Verilog ILM 114 を生成する。

【0032】

Verilog 300 にマッピング・キー 500 を適用することによって生成された保護された Verilog ファイルを HDL ネットリスト 700 として実装した例が図 7 に示されている。Verilog 300 中のネット 304 は、保護されたネット名 702 として示されている一意的に名付けられた接続にエンコードされている。同様に、インスタンス名 306 は、マッピング・キー 500 に従って、保護されたインスタンス名 704 にエンコードされている。たとえば、インスタンス名 306 の最初の「NAND3x1」は、関連のないインスタンス名「JWW17JWW」および一意的な同定された出力「JWW30JWW」を使って、(704 の) 第一の保護されたインスタンス名にエンコードされている。

40

【0033】

ここで図 8 を参照すると、本発明のある好ましい実施形態に基づく、EDA ツールの設計ビュー中の秘匿情報を保護するプロセスの流れ図が示されている。プロセス 800 はステップ 802 で始まり、ここで、IC 設計を記述する HDL コードがフラットにされ、階層構造

50

が除かれる。たとえば、シミュレーション目的のためにサードパーティーに移転されるべきファイルを保護するとき、SPEFおよびVerilogファイルはフラットにされる。次いでプロセスはステップ804に進み、ここで、ILMモデルが生成される。次いでプロセスはステップ806に進み、ここで、保護されるべき信号名が同定される。ここで、プロセスはSPEFファイルを構文解析して、SPEFファイル中で留まるべき、あるいは名前をつけ直されるべき信号数の数および位置を判別する。たとえば、SPEFファイルは、モジュール・ポートの数および位置を判別してそれらが元の形で保持されるようにするために構文解析されるであろう。

【0034】

その後、プロセスはステップ808に進み、ここで、信号名を保護された信号名に関連付けるマッピング・ファイルが生成される。これは、選択された信号名を、元の信号名と無関係だが一意的に各信号名が互いに区別されるよう同定する一意的な識別子にマッピングすることによる。ポート名に含まれる信号名は、ある好ましい実施形態ではマッピングまたは置換されない。次いでプロセスはステップ810に進み、ここで、SPEFファイルは、該ファイルをスキャンして元の信号名のインスタンスを探し、該インスタンスをそのマッピングされた保護された名前で置換することによって、保護された信号名でエンコードされる。その後、プロセスはステップ812に進み、ここで、信号名をその一意的な保護された信号名にマッピングするためのコードを含むスクリプト・ファイルが生成される。その後、プロセスはステップ814に進み、ここで、スクリプト・ファイルはVerilog設計ファイルおよび第一のファイルと同じ信号名を含むICの他の支援設計ビューに対して実行される。実行されると、スクリプトは、ステップ806、808および810を対象設計ビュー・ファイルに対して実行し、選択された信号名の各インスタンスを、その一意的なマッピングされた保護された信号名で置き換えることによって保護する。プロセスはステップ816で、エンコードされたSPEF、Verilogおよび他のサポートされる設計ビューを、各設計ビューどうしの間の一貫性を保持し、該設計ビューのサードパーティー受領者によるEDAツールにおける完全な可用性を提供しつつ、モジュール名および構造を隠す、保護されたフォーマットで出力することによって終了する。

【0035】

理解されるであろうように、本発明の前記の好ましい実施形態は、複数のIC設計ビューが、効果的な仕方で秘匿情報をフィルタ除去され、複数のビューを横断して標準的なEDAツールによる利用を許容するための必要な情報を失うことなく保護された設計ビューが呈示されることを許容する。保護された設計ビューをエンコードすることは、マッピング・キーを使って実行されるので、元の設計者は、保護されたファイル設計ビューのユーザーからのコメントまたは技術的論点を簡単に翻訳できる。これは、技術的設計に関する多くの論点または問題について、保護された設計ビューと対話する顧客をサポートするために重要である。

【0036】

理解されるであろうように、本発明の好ましい実施形態におけるプロセスは、コンピュータ・プログラミング・ソフトウェア、ファームウェアまたはハードウェアの任意の組み合わせを使って実装されうる。本発明をソフトウェアで実施することに向けた準備ステップとして、ある好ましい実施形態に基づくコンピュータ・プログラミング・コード（ソフトウェアであろうとファームウェアであろうと）が典型的には、固定（ハード）ドライブ、ディスク、光ディスク、磁気テープ、ROM、PROMなどのような半導体メモリといった一つまたは複数の機械可読記憶媒体に記憶され、それにより本発明に基づく製造物を作る。前記コンピュータ・プログラミング・コードを有する製造物は、記憶装置から直接コードを実行することによって、記憶装置からコードをハードディスク、RAMなどといった別の記憶装置にコピーすることによって、あるいはコードをリモート実行のために送信することによって、使われる。本発明の方法形態は、本発明に基づくコードを含む一つまたは複数の機械可読記憶媒体を、そこに含まれるコードを実行するための適切な標準的なコンピュータ・ハードウェアと組み合わせることによって実施されうる。本発明を実施する

装置は、本発明に基づいてコーディングされたコンピュータ・プログラム（単数または複数）へのネットワーク・アクセスを含むまたは有する、一つまたは複数のコンピュータおよび記憶システムであることができる。

【0037】

本発明は、好ましい諸実施形態を参照して具体的に図示され、記述されてきたが、当業者は、本発明の精神および範囲から外れることなく、そこに形態および詳細におけるさまざまな変更がなしうることを理解するであろう。記載されている実施形態への任意の変更、修正、追加および改良が可能であり、付属の請求項において詳述される本発明の範囲内にはいりうるものである。

【図面の簡単な説明】

【0038】

【図1】 本発明のある実施形態のブロック図である。

【図2】 フラットなゲート・レベルのSPEFの例を示す図である。

【図3】 Verilogビューにおけるモジュールまたはコア設計の例を示す図である。

【図4】 インスタンス名の例示的なフォーマットを示す図である。

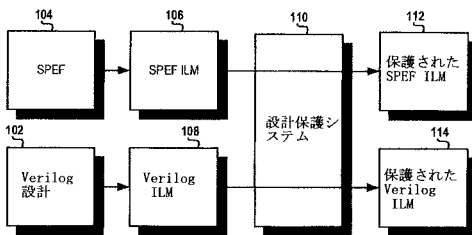
【図5】 本発明のある好ましい実施形態に基づく設計保護システムによって実行される置換法のマッピングを示す図である。

【図6】 本発明のある好ましい実施形態に基づく保護されたSPEFの例を示す図である。

【図7】 本発明のある好ましい実施形態に基づく保護されたVerilogを示す図である。

【図8】 本発明のある好ましい実施形態に基づくEDAツールの設計ビューにおける秘匿情報を保護するためのプロセスの流れ図である。

【図1】



【図2】

```

200 → {
202 {
    *NAME_MAP
    *1 in_and1_a
    *2 in_and1_b
    *3 in_and1_c
    *4 in_and2_a
    *5 in_and2_b
    *6 in_and2_c
    *7 in_and3_a
    *8 in_and3_b
    *9 in_and3_c
    *10 result
    *11 clk
    *12 and_gate_3
    *13 and_gate_2
    *14 and_gate_1
    *15 and_gate_1vinv_gate
    *16 nor_gate
    *17 and_gate_3vhand_gate
204 {
    *PORTS
    *1 O *L 6.000000 *S 0.0 0.0
    *2 O *L 6.000000 *S 0.0 0.0
    *3 O *L 6.000000 *S 0.0 0.0
    *D_NET *1 10.000000
    *CONN
    *1 *19:B *L 10.000000
    *P *1 I
    *CAP
    1 *19:B 10.000000
    2 *1 10.000000
    *RES
    1 *19:B *1 10.000000
    *END
206 {
    *D_NET *2 10.000000
    *CONN
    *1 *19:A *L 10.000000
    *P *2 I
    *CAP
    1 *19:A 10.000000
    2 *2 10.000000
    *RES
    1 *19:A *2 10.000000
    *END
}
}

```

FIGURE 2

【 図 3 】

```

module top_example ( in_and1_a, in_and1_b, in_and1_c, in_and2_a, in_and2_b,
in_and2_c, in_and3_a, in_and3_b, in_and3_c, result, clk );
302 input in_and1_a, in_and1_b, in_and1_c, in_and2_a,
in_and2_b, in_and2_c, in_and3_a, in_and3_b, in_and3_c, clk;
output result;

wire land_gate_3/and_z_reg_reg;
wire land_gate_3/nand_z;
wire land_gate_3/nand_z_reg;
wire land_gate_3/and_z_reg;
wire land_gate_2/and_z_reg_reg;
304 wire land_gate_2/nand_z;
wire land_gate_2/nand_z_reg;
wire land_gate_1/and_z_reg;
wire land_gate_1/nand_z;
wire land_gate_1/inv_gate/and_z_reg;
wire land_gate_1/nand_z_reg;

NAND3x1 land_gate_3/nand_gate ( A(in_and3_b), B(in_and3_a), C(in_and3_c), Z(land_gate_3/
nand_z));
NAND3x2 land_gate_2/nand_gate ( A(in_and2_b), B(in_and2_a), C(in_and2_c), Z(land_gate_2/
nand_z));
NAND3x1 land_gate_1/nand_gate ( A(in_and1_b), B(in_and1_a), C(in_and1_c), Z(land_gate_1/
nand_z));
DFFx1 land_gate_3/delay_1 ( D(land_gate_3/nand_z), CK(clk), Q(land_gate_3/nand_z_reg));
INVx8 land_gate_3/inv_gate ( A(land_gate_3/nand_z_reg), Z(land_gate_3/and_z_reg));
DFFx1 land_gate_3/delay_2 ( D(land_gate_3/and_z_reg), CK(clk), Q(land_gate_3/
and_z_reg_reg));
306 DFFx1 land_gate_2/delay_1 ( D(land_gate_2/nand_z), CK(clk), Q(land_gate_2/nand_z_reg));
INVx1 land_gate_2/inv_gate ( A(land_gate_2/nand_z_reg), Z(land_gate_2/and_z_reg));
DFFx1 land_gate_2/delay_2 ( D(land_gate_2/and_z_reg), CK(clk), Q(land_gate_2/
and_z_reg_reg));
DFFx1 land_gate_1/delay_1 ( D(land_gate_1/nand_z), CK(clk), Q(land_gate_1/nand_z_reg));
INVx1 land_gate_1/inv_gate/inv_gate ( A(land_gate_1/nand_z_reg), Z(land_gate_1/inv_gate
and_z_reg));
DFFx1 land_gate_1/delay_2 ( D(land_gate_1/inv_gate/and_z_reg), CK(clk),
Q(land_gate_1/and_z_reg_reg));
NOR3x1 nor_gate ( A(land_gate_2/and_z_reg_reg), B(land_gate_3/and_z_reg_reg), C(
and_gate_1/and_z_reg_reg), Z(result));
endmodule

```

FIGURE 3

【 図 4 】

インスタンス名	入力A	入力B	出力
402	404	406	408

【 図 6 】

```

*NAME_MAP
*1 in_and1_a
*2 in_and1_b
*3 in_and1_c
*4 in_and2_a
*5 in_and2_b
*6 in_and2_c
*7 in_and3_a
*8 in_and3_b
*9 in_and3_c
*10 result
*11 clk
602 {
*12 JWW12JWW
*13 JWW13JWW
*14 JWW14JWW
*15 JWW15JWW
*16 JWW16JWW
*17 JWW17JWW

*PORTS
*1 O 1.6 0.000000 *S 0.0 0.0
*2 O 1.6 0.000000 *S 0.0 0.0
*3 O 1.6 0.000000 *S 0.0 0.0

*D_NET *1 10.000000
*CONN
*1 *19 B | *L 10.000000
*P *11
*CAP
*1 *19 B 10.000000
*2 *1 10.000000
*RES
*1 *19 B *1 10.000000
*END
*D_NET *2 10.000000
*CONN
*1 *19 A | *L 10.000000
*P *21
*CAP
*1 *19 A 10.000000
*2 *2 10.000000
*RES
*1 *19 A *2 10.000000
*END

```

FIGURE 6

【 図 5 】

```

506 ~ %sland_gate_3/UWW12JWWlg
~ %sland_gate_2/UWW13JWWlg
~ %sland_gate_1/UWW14JWWlg
~ %sJWW14JWWlg/inv_gate/UWW15JWWlg
~ %sJWW12JWWlg/inv_gate/UWW16JWWlg
508 ~ %sJWW12JWWlg/nand_gate/UWW17JWWlg
~ %sJWW13JWWlg/nand_gate/UWW18JWWlg
~ %sJWW14JWWlg/nand_gate/UWW19JWWlg
~ %sJWW12JWWlg/delay_1/UWW20JWWlg
~ %sJWW12JWWlg/inv_gate/UWW21JWWlg
~ %sJWW12JWWlg/delay_2/UWW22JWWlg
~ %sJWW13JWWlg/delay_1/UWW23JWWlg
~ %sJWW13JWWlg/inv_gate/UWW24JWWlg
~ %sJWW13JWWlg/delay_2/UWW25JWWlg
~ %sJWW14JWWlg/delay_1/UWW26JWWlg
~ %sJWW15JWWlg/inv_gate/UWW27JWWlg
~ %sJWW14JWWlg/delay_2/UWW28JWWlg
~ %sJWW12JWWlg/and_z_reg/UWW29JWWlg
~ %sJWW12JWWlg/nand_z/UWW30JWWlg
~ %sJWW30JWWlg/reg/UWW31JWWlg
~ %sJWW12JWWlg/and_z_reg/UWW32JWWlg
~ %sJWW13JWWlg/and_z_reg/UWW33JWWlg
~ %sJWW13JWWlg/nand_z/UWW34JWWlg
~ %sJWW34JWWlg/reg/UWW35JWWlg
~ %sJWW13JWWlg/and_z_reg/UWW36JWWlg
~ %sJWW14JWWlg/and_z_reg/UWW37JWWlg
~ %sJWW14JWWlg/nand_z/UWW38JWWlg
~ %sJWW15JWWlg/and_z_reg/UWW39JWWlg
~ %sJWW38JWWlg/reg/UWW40JWWlg
x

```

FIGURE 5

【 図 7 】

```

module top_example ( in_and1_a, in_and1_b, in_and1_c, in_and2_a, in_and2_b,
in_and2_c, in_and3_a, in_and3_b, in_and3_c, result, clk );
input in_and1_a, in_and1_b, in_and1_c, in_and2_a, in_and2_b, in_and2_c,
in_and3_a, in_and3_b, in_and3_c, clk;
output result;

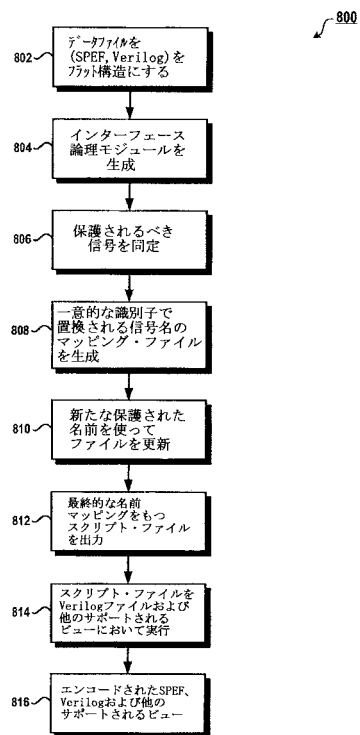
wire UWW29JWW;
wire UWW30JWW;
wire UWW31JWW;
wire UWW32JWW;
wire UWW33JWW;
702 {
wire UWW34JWW;
wire UWW35JWW;
wire UWW36JWW;
wire UWW37JWW;
wire UWW38JWW;
wire UWW39JWW;
wire UWW40JWW;

NAND3x1 UWW17JWW ( A(in_and3_b), B(in_and3_a), C(in_and3_c), Z(UWW30JWW));
NAND3x2 UWW18JWW ( A(in_and2_b), B(in_and2_a), C(in_and2_c), Z(UWW34JWW));
NAND3x1 UWW19JWW ( A(in_and1_b), B(in_and1_a), C(in_and1_c), Z(UWW38JWW));
DFFx1 UWW20JWW ( D(UWW30JWW), CK(clk), Q(UWW31JWW));
INVx8 UWW21JWW ( A(UWW31JWW), Z(UWW32JWW));
DFFx1 UWW22JWW ( D(UWW32JWW), CK(clk), Q(UWW29JWW));
DFFx1 UWW24JWW ( D(UWW35JWW), CK(clk), Q(UWW36JWW));
INVx1 UWW25JWW ( A(UWW36JWW), Z(UWW33JWW));
DFFx1 UWW26JWW ( D(UWW38JWW), CK(clk), Q(UWW39JWW));
INVx1 UWW27JWW ( A(UWW40JWW), Z(UWW39JWW));
DFFx1 UWW28JWW ( D(UWW39JWW), CK(clk), Q(UWW37JWW));
NOR3x1 JWW16JWW ( A(UWW33JWW), B(UWW29JWW), C(UWW37JWW), Z(result));
endmodule

```

FIGURE 7

【図 8】



【国際調査報告】

60900020042



INTERNATIONAL SEARCH REPORT

International application No.

PCT/US06/60261

A. CLASSIFICATION OF SUBJECT MATTER		
IPC: G06F 17/50(2006.01)		
USPC: 716/12		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols) U.S.: 716/12		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) Please See Continuation Sheet		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,E	US 7,168,041 B1 (DURRILL) 23 January 2007 (23.01.2007)	1-3, 5-7, 9-13, 15-16 and 18-20
Y	US 2005/0229128 A1 (SRIPADA) 13 October 2005 (13.10.2005)	4 and 14
Y,E	US 2007/0050745 A1 (DINTER et al) 01 March 2007 (01.03.2007)	8 and 17
☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family		
Date of the actual completion of the international search 27 August 2008 (27.08.2008)		Date of mailing of the international search report 03 SEP 2008
Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US Commissioner for Patents P.O. Box 1450 Alexandria, Virginia 22313-1450 Facsimile No. (571) 273-3201		Authorized officer Marianne Seidel Telephone No. 571-272-1600 <i>Janice Ford for</i>

Form PCT/ISA/210 (second sheet) (April 2007)

19.1.2009

2₁**INTERNATIONAL SEARCH REPORT****International application No.**
PCT/US06/60261**Continuation of B. FIELDS SEARCHED Item 3:****Search Text:** US-PGPUB;USPAT;FPRS;EPO;JPO;DERWENT;IBM_TDB**Search Terms:** signal, different, exclusive; individual, unique, view or window; protect, identifier, name

フロントページの続き

(81)指定国 AP(BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), EP(AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW

(72)発明者 デュア, マイケル ダブリュ

アメリカ合衆国, テキサス州 78630, シーダー・パーク, ピー. オー. ボックス 1081

(72)発明者 ウールシィ, ジョン ダブリュ

アメリカ合衆国, テキサス州 78681, ラウンド・ロック, ノース・サマークラスト 4210

Fターム(参考) 5B046 AA08 BA02 BA03 KA00

5F064 BB35 DD02 EE02 HH06 HH08 HH09 HH10 HH11