

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6209892号  
(P6209892)

(45) 発行日 平成29年10月11日(2017.10.11)

(24) 登録日 平成29年9月22日(2017.9.22)

(51) Int.Cl.

F I

G 0 6 F 11/22 (2006.01)

G 0 6 F 11/22 6 7 3 Z

請求項の数 10 (全 18 頁)

(21) 出願番号	特願2013-157057 (P2013-157057)	(73) 特許権者	000005223
(22) 出願日	平成25年7月29日(2013.7.29)		富士通株式会社
(65) 公開番号	特開2015-26337 (P2015-26337A)		神奈川県川崎市中原区上小田中4丁目1番1号
(43) 公開日	平成27年2月5日(2015.2.5)	(74) 代理人	100104190
審査請求日	平成28年4月5日(2016.4.5)		弁理士 酒井 昭徳
		(72) 発明者	パリジ マチュー
			神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		(72) 発明者	▲高▼山 浩一郎
			神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		審査官	三坂 敏夫
			最終頁に続く

(54) 【発明の名称】 競合テスト用タイミング調整プログラム、競合テスト用タイミング調整方法および競合テスト用タイミング調整装置

(57) 【特許請求の範囲】

【請求項 1】

半導体装置の内部信号を任意のイベントで競合させる競合タイミングを調整する競合テスト用タイミング調整プログラムにおいて、

コンピュータに、

競合させたい一对のイベントの指定により、前記半導体装置の入力のトランザクションと前記イベントとに基づき、前記一对のイベントに関連する複数の前記トランザクションを1組毎に選択させ、

選択した1組の前記トランザクションに含まれる前記一对のイベントの時刻が一致するように、一方のトランザクションの入力タイミングを調整させ、

入力タイミングが調整された一方のトランザクションに含まれる他のイベントの時刻が、他方のトランザクションに含まれる同一のイベントの時刻と不一致となるかを判断させ、

判断結果として不一致が得られた1組のトランザクションと、調整した前記入力タイミングと、を競合させたい一对のイベントに対するタイミング調整結果として出力させる、処理を実行させることを特徴とする競合テスト用タイミング調整プログラム。

【請求項 2】

前記半導体装置の入力のトランザクションと、前記イベントとを関連付けたイベントテーブルを生成させ、

競合させたい一对のイベントの入力に基づき、前記イベントテーブルを参照して、前記

10

20

一对のイベントに関連する複数の前記トランザクションを1組毎に選択させることを特徴とする請求項1に記載の競合テスト用タイミング調整プログラム。

【請求項3】

前記判断結果として一致が生じた前記一方のトランザクションには低い優先度を付与することを特徴とする請求項1または2に記載の競合テスト用タイミング調整プログラム。

【請求項4】

前記判断結果として一致が生じた後、前記一对のイベントに関連する前記トランザクションの組が残っていない場合には、一对のイベントに対するタイミング調整が不可能であると判断することを特徴とする請求項1～3のいずれか一つに記載の競合テスト用タイミング調整プログラム。

10

【請求項5】

前記トランザクションをシミュレータに実行させ、前記イベントが生じた時刻を前記イベントテーブルに記録させることを特徴とする請求項2に記載の競合テスト用タイミング調整プログラム。

【請求項6】

前記タイミング調整結果により得られた複数組のトランザクションをシミュレータに出力させ、

前記シミュレータのシミュレーション結果に基づき、前記一对のイベントにおける競合条件が満たされた場合に、一对のイベントに対するタイミング調整に関する情報を競合シナリオとして出力させる、

20

ことを特徴とする請求項1～5のいずれか一つに記載の競合テスト用タイミング調整プログラム。

【請求項7】

前記判断結果として一致が生じた前記一方のトランザクションは、前記イベントテーブルの優先度が低い格納位置に変更させることを特徴とする請求項2または5に記載の競合テスト用タイミング調整プログラム。

【請求項8】

選択した1組の前記トランザクションに含まれる前記一对のイベントの時刻のうち、時刻が最大値のイベントの時刻が一致するように、一方のトランザクションの入力タイミングを調整させることを特徴とする請求項1～7のいずれか一つに記載の競合テスト用タイミング調整プログラム。

30

【請求項9】

半導体装置の内部信号を任意のイベントで競合させる競合タイミングを調整する競合テスト用タイミング調整方法において、

コンピュータが、

競合させたい一对のイベントの指定により、前記半導体装置の入力のトランザクションと前記イベントとに基づき、前記一对のイベントに関連する複数の前記トランザクションを1組毎に選択し、

選択した1組の前記トランザクションに含まれる前記一对のイベントの時刻が一致するように、一方のトランザクションの入力タイミングを調整し、

40

入力タイミングが調整された一方のトランザクションに含まれる他のイベントの時刻が、他方のトランザクションに含まれる同一のイベントの時刻と不一致となるかを判断し、

判断結果として不一致が得られた1組のトランザクションと、調整した前記入力タイミングと、を競合させたい一对のイベントに対するタイミング調整結果として出力する、

処理を実行することを特徴とする競合テスト用タイミング調整方法。

【請求項10】

半導体装置の内部信号を任意のイベントで競合させる競合タイミングを調整する競合テスト用タイミング調整装置において、

前記半導体装置の入力のトランザクションと、前記イベントとを関連付けたイベントテーブルと、

50

競合させたい一対のイベントの入力に基づき、前記イベントテーブルを参照して、前記一対のイベントに関連する複数の前記トランザクションを1組毎に選択し、

選択した1組の前記トランザクションに含まれる前記一対のイベントの時刻が一致するように、一方のトランザクションの入力タイミングを調整し、

入力タイミングが調整された一方のトランザクションに含まれる他のイベントの時刻が、他方のトランザクションに含まれる同一のイベントの時刻と不一致となるかを判断し、

判断結果として不一致が得られた1組のトランザクションと、調整した前記入力タイミングと、を競合させたい一対のイベントに対するタイミング調整結果として出力する競合シナリオ生成部と、

を有することを特徴とする競合テスト用タイミング調整装置。

10

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、半導体装置の競合テスト用タイミング調整プログラム、競合テスト用タイミング調整方法および競合テスト用タイミング調整装置に関する。

【背景技術】

【0002】

LSI等の半導体装置は、機能検証装置を用いてハードウェア記述言語により記載された論理回路を検証する。この機能検証の一つとして、複数信号のタイミング依存によるLSI内部信号の競合検証がある。

20

【0003】

競合検証では、例えば、競合の条件を登録したテーブルを参照し、複数のプログラムが並行して動作するシミュレーション中に発生した条件のうち競合させたい条件が一致するかをチェックする技術がある。この技術では、一部の条件一致時にシミュレーション実行を停止し、全部の条件一致時にシミュレーション実行を再開させる（例えば、下記特許文献1参照。）。

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特開平11-238002号公報

30

【発明の概要】

【発明が解決しようとする課題】

【0005】

しかしながら、従来技術では、データの入力側からみて深い位置に位置する競合点における競合の検証時に、複数のデータパス別の時間差を全て把握して競合を生じさせることは困難であり、競合の検証に手間がかかった。

【0006】

例えば、競合点として、コア内部の競合点（アービタ）の調停動作（優先出力）を確認するために、このアービタに対して2つの信号（データ1，データ2）が同時に入力されたときの検証を行う。この際、コアに対して外部からそれぞれ入力される2つのデータは、競合点（アービタ）に到達するまでにかかる時間が異なる場合がある。例えば、2つのデータパス上で競合点（アービタ）より手前の回路ブロック（例えば、フリップフロップ（FF））で所定のイベント発生が生じると、検証が必要な競合点（アービタ）にリクエストが届くまでの（遅延）時間が異なってしまう。

40

【0007】

図10は、競合の発生を説明するための図である。横軸は時間であり、縦軸には2つのデータ（データ1，データ2）を示す。はじめに、（a）に示すように、データ1，2を同時に（時期 $t=0$ ）入力させたとする。このときのアービタに対するリクエストが生じるまでの時間は、データ1が $t=1$ 、データ2が $t=2$ であったとする。この場合、データ1，2間のリクエストの時間差を波形等で確認する。この後、（b）に示すように、

50

時間差 に相当する分だけデータ 2 の入力タイミングを遅らせれば、競合点（アービタ）においてデータ 1 , 2 のリクエストが競合可能となる。

【 0 0 0 8 】

しかしながら、( b ) に示した入力タイミングを遅らせたとしても、競合点（アービタ）においてデータ 1 , 2 のリクエストのタイミングが競合しない場合がある。例えば、( c ) に示すように、データ 2 に対して時間差 を与えて入力しても、実際に競合点（アービタ）にリクエストが発生するまでの時間が  $t = 2'$  となり予想した  $t = 2$  と異なることがある。

【 0 0 0 9 】

この相違は、データパス上の複数の回路ブロック（ F F ）において所定のイベント発生等に基づく遅延量の積算によって生じる。従来、検証装置では、大規模な L S I 上での個々の回路ブロック（ F F ）の順序制御までは把握せずに、時間差 を更新して所定回数までに競合条件が満たされるか判断している。そして、検証装置による競合条件が満たされない場合、設計者により時間差 を手作業で更新する等の作業が必要となり、競合条件に収束するまでに時間と手間がかかった。また、手作業で行っても競合条件に収束しない結果となる場合もある。

【 0 0 1 0 】

ここで、データパス上の回路ブロック（ F F ）は膨大な数であるため、個々の回路ブロックの動作にかかる遅延量を全て把握することは困難であり、競合点（アービタ）における競合のタイミングが最適となるようにいかにして調整するか、その具体的解決が望まれている。データの入力側からみたデータパスの深い位置の競合点（アービタ）では、手前に多数のブロックが存在するため、複数のデータが競合する入力タイミングを得ることが困難であった。

【 0 0 1 1 】

一つの側面では、本発明は、複数のデータの任意の競合点における競合タイミングを最適に調整できることを目的とする。

【課題を解決するための手段】

【 0 0 1 2 】

一つの場合では、競合させたい一对のイベントの指定により、半導体装置の入力のトランザクションとイベントとに基づき、前記一对のイベントに関連する複数の前記トランザクションを 1 組毎に選択し、選択した 1 組の前記トランザクションに含まれる前記一对のイベントの時刻が一致するように、一方のトランザクションの入力タイミングを調整し、入力タイミングが調整された一方のトランザクションに含まれる他のイベントの時刻が、他方のトランザクションに含まれる同一のイベントの時刻と不一致となるかを判断し、判断結果として不一致が得られた 1 組のトランザクションと、調整した前記入力タイミングと、を競合させたい一对のイベントに対するタイミング調整結果として出力する。

【発明の効果】

【 0 0 1 3 】

一つの実施形態によれば、複数のデータの任意の競合点における競合タイミングを最適に調整できる。

【図面の簡単な説明】

【 0 0 1 4 】

【図 1】図 1 は、実施の形態にかかる競合テスト用タイミング調整装置の機能を示すブロック図である。

【図 2】図 2 は、イベントテーブルに格納されるイベントテーブル情報を示す図表である。

【図 3】図 3 は、イベントテーブル生成の処理手順の一例を示すフローチャートである。

【図 4】図 4 は、競合シナリオ生成にかかるトランザクション選択例を説明するための図表である。

【図 5】図 5 は、競合シナリオ生成のためのトランザクション選択処理の一例を示すフロ

10

20

30

40

50

ーチャートである。

【図6】図6は、競合シナリオ生成処理の一例を示すフローチャートである。

【図7】図7は、競合テスト用タイミング調整装置のハードウェア構成の一例を示すブロック図である。

【図8】図8は、競合テスト用タイミング調整装置の配置例を示す図である。

【図9】図9は、競合点でのタイミング競合の一例を示す図である。

【図10】図10は、競合の発生を説明するための図である。

【発明を実施するための形態】

【0015】

(実施の形態)

10

以下に添付図面を参照して、開示技術の好適な実施の形態を詳細に説明する。図1は、実施の形態にかかる競合テスト用タイミング調整装置の機能を示すブロック図である。競合テスト用タイミング調整装置100は、イベントテーブル生成部101と、イベントテーブル102と、競合シナリオ生成部103と、を含む。

【0016】

イベントテーブル生成部101は、コアに対する入力トランザクションの集合の情報( $Tr_1, Tr_2, \dots, Tr_n$ )110と、コア内部における内部イベントのリスト情報( $Ev_1, Ev_2, \dots, Ev_m$ )111に基づき、イベントテーブル情報112を生成し、イベントテーブル102に格納する。内部イベントのリスト情報111は、コア内部において競合点(アービタ等)における競合させたい(競合対象)のイベントを含む。

20

【0017】

競合シナリオ生成部103は、イベントテーブル102に格納されたイベントテーブル情報112と、競合させたいイベントの組の情報113の指定入力と、に基づいて、競合させたいイベントに対する競合シナリオの情報121を生成する。競合させたいイベントの組の情報113は、内部イベントのリスト情報111のなかからサブセットとして用意し、指定入力することができる。例えば、競合点がコア内部の所定のブロック(例えばアービタ)の場合、このブロック(アービタ)に対してリクエスト( $Req$ )を出力する1対のブロック(例えばFF)がイベントの組の情報として指定入力される。

【0018】

(イベントテーブルの生成について)

30

図2は、イベントテーブルに格納されるイベントテーブル情報を示す図表である。イベントテーブル生成部101は、入力トランザクションの集合の情報110の複数のトランザクション( $Tr_i$ )をシミュレータによりシミュレーション実行する。このシミュレーション実行時に、イベントテーブル生成部101は、内部イベントのリスト情報111に登録されたイベント( $Ev_j$ )が発生する時刻( $T_k$ )を一覧化したイベントテーブル情報112をイベントテーブル102に格納する。イベントテーブル情報112は、縦軸が各トランザクション $Tr_i$ 、横軸には各イベント $Ev_j$ を時刻 $T_k$ 毎に(右方向ほど時刻経過)格納する。

【0019】

例えば、各イベント $Ev_j$ は、コア内部におけるデータパス上の各FFに相当する。図2において、例えば、図中112aに示すトランザクション $Tr_3$ を実行した際に、イベント $Ev_1$ 、 $Ev_3$ は発生せず、イベント $Ev_2$ 、 $Ev_4$ 、 $Ev_5$ が発生した時刻がそれぞれ $T=50$ 、 $80$ 、 $100$ であることを示している。

40

【0020】

図3は、イベントテーブル生成の処理手順の一例を示すフローチャートである。イベントテーブル生成部101が行うイベントテーブル情報112の生成内容について説明する。はじめに、イベントテーブル生成部101は、トランザクション $Tr_i$ 実行の初期値 $i=0$ をセットする(ステップS301)。次に、イベントテーブル生成部101は、シミュレータにより、入力トランザクションの集合の情報110が示すトランザクション $Tr_i$ を実行させる(ステップS302)。

50

## 【 0 0 2 1 】

この後、イベントテーブル生成部 1 0 1 は、内部イベントのリスト情報 1 1 1 に登録されたイベント (  $E v_j$  ) が発生する時刻 (  $T_k$  ) をイベントテーブル情報 1 1 2 の該当するイベント (  $E v_j$  ) に記録する ( ステップ S 3 0 3 )。そして、変数  $i$  をインクリメントする。

## 【 0 0 2 2 】

そして、イベントテーブル生成部 1 0 1 は、 $n$  個の最後のトランザクション (  $T r_n$  ) のシミュレーション実行が終了したか判断する (  $i < n$  ) ( ステップ S 3 0 4 )。そして、未実行分のトランザクションがあれば ( ステップ S 3 0 4 : N o )、ステップ S 3 0 2 に戻り、全て実行済みであれば ( ステップ S 3 0 4 : Y e s )、以上のイベントテーブル情報 1 1 2 の生成の処理を終了する。

10

## 【 0 0 2 3 】

( 競合シナリオの生成について )

競合シナリオ生成部 1 0 3 は、以下の ( 1 )、( 2 ) の処理を行う。

( 1 ) 競合イベントが同時に発生するための時刻調整と、トランザクションの組の優先度の付与。この処理では、イベントテーブル情報 1 1 2 に基づいて、競合点において競合させたいイベントが発生するトランザクションの組を全て生成する。そして、競合させたいイベントが同時に発生するように、トランザクションの開始時刻を決定する。また、他のイベントの競合可能性により、生成したトランザクションの組に優先度を付与する。

## 【 0 0 2 4 】

20

( 2 ) 上記 ( 1 ) で付与された優先度順にトランザクションの各組み合わせのシミュレーション実行による競合発生の有無の確認と、競合シナリオの生成。この処理では、競合シナリオ生成部 1 0 3 は、トランザクションの各組み合わせに対し、優先度順にシミュレータによるシミュレーションの実行により、実際に競合点で競合が生じるか否かを確認し、競合が起こった組み合わせを競合シナリオ 1 2 1 として出力する。

## 【 0 0 2 5 】

( 1 : 競合イベントが同時に発生するための時刻調整と、トランザクションの組の優先度の付与の処理内容の説明 )

図 4 は、競合シナリオ生成にかかるトランザクション選択例を説明するための図表である。図 2 と同様のイベントテーブル情報 1 1 2 の例を用い、競合シナリオ生成部 1 0 3 が行う処理例について説明する。ここで、競合点 ( アービタ等 ) において競合させたいイベントの組が  $E v_4$  と  $E v_5$  であるとする。

30

## 【 0 0 2 6 】

はじめに、( a ) に示すように、競合シナリオ生成部 1 0 3 は、( 1 ) イベント  $E v_4$  を発生させるトランザクション  $T r$  を探す。そして、( 2 ) イベント  $E v_4$  が含まれるトランザクション  $T r_1$  を選択する。この後、( 3 ) イベント  $E v_5$  を発生させるトランザクションを探し、トランザクション  $T r_2$  を選択する。

## 【 0 0 2 7 】

この後、( b ) に示すように、( 4 ) 選択したトランザクション  $T r_1$  のイベント  $E v_4$  (  $T = 50$  ) と、トランザクション  $T r_2$  のイベント  $E v_5$  (  $T = 80$  ) を同時に発生させる時刻を判断する。この場合、競合シナリオ生成部 1 0 3 は、トランザクション  $T r_1$  と  $T r_2$  の時刻の差分 により、トランザクション  $T r_1$  の開始時刻を + 30 と判断する。

40

## 【 0 0 2 8 】

この後、( c ) に示すように、( 5 ) トランザクション  $T r_1$  のイベント  $E v_2$  の時刻が 50 となり (  $20 + 30$  )、トランザクション  $T r_2$  のイベント  $E v_2$  の時刻 50 と同じになる。これにより、イベント  $E v_2$  における競合による調停が働き、トランザクション  $T r_1$  または  $T r_2$  のいずれかの処理が待たされる ( 遅延が生じる ) 可能性がある。

## 【 0 0 2 9 】

イベント  $E v_2$  の発生は、競合させたいイベント  $E v_4$  と  $E v_5$  が発生する前 ( データ

50

パス上の手前)の位置であるため、競合シナリオ生成部103は、イベントE v 2における調停の結果、イベントE v 4とE v 5は同時に発生(競合)しない可能性があると判断する。このような場合、競合シナリオ生成部103は、トランザクションTr 1とTr 2の組み合わせを低優先度に設定する。

#### 【0030】

そして、(d)に示すように、(6)一方のトランザクションTr 1を基準として組み合わせたときの他方のトランザクションTr 2を優先度を下げ、対応してテーブルの格納順位を下げる。この後、(7)競合シナリオ生成部103は、再度イベントE v 5を生じるトランザクションTrを探し、(8)トランザクションTr 3を選択する。

#### 【0031】

この後、選択したトランザクションTr 1のイベントE v 4( $T = 50$ )と、トランザクションTr 3のイベントE v 5( $T = 100$ )を同時に発生させる時刻を判断する。この場合、競合シナリオ生成部103は、トランザクションTr 1とTr 3の時刻の差分により、トランザクションTr 1の開始時刻を $+50$ と判断する。

#### 【0032】

この場合、(e)に示すように、トランザクションTr 1について、(9)イベントE v 2の開始時刻は70となり( $20 + 50$ )、トランザクションTr 3の時刻50と異なるため、同時に発生しない(競合なし)と判断する。また、(10)イベントE v 3については、トランザクションTr 3側にイベントがないため、同時に発生しないと判断する。そして、(11)イベントE v 4の開始時刻は100であり( $50 + 50$ )、トランザクションTr 3の時刻80と異なるため、同時に発生しない(競合なし)と判断する。

#### 【0033】

以上により、競合シナリオ生成部103は、(12)トランザクションTr 1とTr 3の組み合わせを高優先度とする。

#### 【0034】

この後、競合シナリオ生成部103は、トランザクションの各組み合わせに対し、シミュレータによるシミュレーションの実行により、実際に競合点で競合が生じるか否かを確認し、競合が起こった組み合わせを競合シナリオ121として出力する。この際、図4に示した高優先度のトランザクションTr 1のイベントE v 4と、トランザクションTr 3のイベントE v 5との組み合わせ、を優先してシミュレーションを行う。

#### 【0035】

図5は、競合シナリオ生成のためのトランザクション選択処理の一例を示すフローチャートである。上記図4の処理例に示した競合シナリオ生成部103が行う処理内容について説明する。以下の処理では、競合させたいイベントを含む1組のトランザクションを選択して、1組のトランザクションに含まれる複数の他の(競合点以外の)イベントについての競合が発生しないトランザクションの組を選択する。競合シナリオ生成部103は、イベントテーブル102のイベントテーブル情報112からトランザクションの組を変更して読み出し、各トランザクションの組に対する以下の処理を行う。

#### 【0036】

はじめに、競合シナリオ生成部103は、イベントテーブル102のイベントテーブル情報112を読み出し、競合点におけるイベントE v jを含むトランザクションTr iを選択する(ステップS501)。次に、選択したトランザクションTr iに含まれる全てのイベントE v jを抽出したか判断し(ステップS502)、未抽出分が残っていれば(ステップS502: No)、残りのイベントE v jを抽出し(ステップS503)、ステップS501に戻る。全て抽出済みであれば(ステップS502: Yes)、ステップS504に移行する。

#### 【0037】

ステップS501~ステップS503の処理により、イベントテーブル情報112に含まれる全てのトランザクションTr iの全てのイベントE v jが抽出される。抽出した情報は、図示しないメモリに保持しておく。例えば、図4の(a)に示す例では、トランザ

10

20

30

40

50

クション  $Tr_1 \sim Tr_3$  に含まれる各イベント  $Ev$  が抽出される。

【0038】

以下の処理では、競合シナリオ生成部 103 は、競合点における競合を発生させる複数のイベント  $Ev_j$  を含む全てのトランザクション  $Tr_i$  について、1組（一対）のトランザクション  $Tr_i$  毎にタイミング（時刻）調整を行い、競合点以外でのイベントでの競合の発生の有無を確認する。

【0039】

まず、競合シナリオ生成部 103 は、1組のトランザクション  $Tr_i$  を選択する（ステップ S504）。例えば、図4に示したトランザクション  $Tr_1, Tr_2$  を選択する。次に、選択した1組のトランザクション  $Tr_i$  のうち、時刻  $T_{ij}$  が最大値のイベント  $Ev_j$  を探す（ステップ S505）。最大値の時刻  $T_{ij}$  におけるイベントは、コア上のデータベースにおいて最も奥部に位置するブロック（アービタ等）における競合を示す。例えば、図4の（b）に示す例では、1組のトランザクション  $Tr_1, Tr_2$  のうち最大値の時刻  $T_{ij}$  は、80である。

【0040】

なお、この例では、競合点を最も奥に位置するブロックとして時刻の最大値を探すこととしたが、競合点をデータベース上でより手前の任意のブロックに設定し、対応する時刻を探すこととしてもよい。

【0041】

次に、競合シナリオ生成部 103 は、選択した1組のトランザクション  $Tr_i$  のうち一方のトランザクション  $Tr_i$  の開始時刻を変更する（ステップ S506）。この際、競合シナリオ生成部 103 は、1組のトランザクション  $Tr_i$  における競合点における時刻が同じになる差分 を求め、一方のトランザクション  $Tr_i$  の開始時刻に差分 を加える。例えば、図4の（c）に示す例では、トランザクション  $Tr_1$  の開始時刻に差分 として 30を加える。

【0042】

次に、競合シナリオ生成部 103 は、差分 を加えたトランザクション  $Tr_i$  に含まれる各イベント  $Ev_j$  の時刻に差分 を加える（ステップ S507）。図4の（c）に示す例では、トランザクション  $Tr_1$  の各イベント  $Ev_2, Ev_3, Ev_4$  の時刻にそれぞれ差分 の 30 が加えられる。

【0043】

次に、競合シナリオ生成部 103 は、選択した一対のトランザクション  $Tr_i$  における各イベント  $Ev_j$  における競合の有無を確認する処理を行う。この際、所望する競合点に対応するイベント  $Ev_j$  の時刻は一致し、かつ、他のイベント  $Ev_j$  における時刻が不一致であることを条件として確認の処理を行う。

【0044】

ステップ S508 の処理は、1組のトランザクション  $Tr_i$  を更新する処理であるが、競合シナリオ生成部 103 は、初回については、ステップ S504 にて選択された1組のトランザクション  $Tr_i$  を選択する（ステップ S508）。

【0045】

以降の処理では、変数  $k$  のインクリメントにより、競合点における競合を行うトランザクション  $Tr_i$  の組が変更される。すなわち、競合点（アービタ）に対するデータ入力は、1組に限らず複数存在する可能性があるため、これら複数のデータ入力に対応する複数組のトランザクション  $Tr_i$  を順次選択する。例えば、図4の例では、トランザクション  $Tr_1$  と  $Tr_2$  の組の後、トランザクション  $Tr_1$  を代えずにトランザクション  $Tr_3$  を選択してトランザクション  $Tr_1, Tr_3$  を1組とする。

【0046】

ステップ S508 の後、競合シナリオ生成部 103 は、1組のトランザクション  $Tr_i$  について、各イベント  $Ev_j$  における時刻  $T_{ij}$  が不一致であるか判断する（ステップ S509）。例えば、図4の（c）に示す例では、トランザクション  $Tr_1, Tr_2$  のイベ

10

20

30

40

50



ント  $E v 2$  ,  $E v 3$  ,  $E v 4$  における時刻の不一致を判断する。

【 0 0 4 7 】

この結果、競合シナリオ生成部 1 0 3 は、イベント  $E v j$  における時刻の一致が生じた場合には ( ステップ  $S 5 0 9$  :  $N o$  )、該当するランザクション  $T r i$  の優先度を下げ、イベントテーブル情報 1 1 2 の最下段に位置変更する ( ステップ  $S 5 1 0$  )。例えば、図 4 ( d ) に示すように、1 組のランザクション  $T r i$  のうち一方のランザクション  $T r 2$  の優先度を下げてイベントテーブル情報 1 1 2 の最下段に位置変更する。これにより、イベントテーブル情報 1 1 2 には、上段ほど、優先度が高いランザクション  $T r i$  が格納されることになる。

【 0 0 4 8 】

次に、競合シナリオ生成部 1 0 3 は、1 組のランザクション  $T r i$  の一方のランザクションを新たなランザクション  $T r i$  に変更する ( ステップ  $S 5 1 1$  )。この際、ランザクションは競合点におけるイベント ( 時刻 ) を含むランザクションを選択する。図 4 の ( d ) の例では、新たなランザクション  $T r i$  としてランザクション  $T r 3$  を選択し、ランザクション  $T r 1$  ,  $T r 3$  が新たな 1 組のランザクションとなる。

【 0 0 4 9 】

次に、競合シナリオ生成部 1 0 3 は、この新たな 1 組のランザクション  $T r i$  を用いた上記のタイミング調整の処理を実行済みであるか確認する ( ステップ  $S 5 1 2$  )。未実行の組であれば ( ステップ  $S 5 1 2$  :  $N o$  )、ステップ  $S 5 0 9$  に戻るが、実行済みであれば ( ステップ  $S 5 1 2$  :  $Y e s$  )、全ての組み合わせのランザクション  $T r i$  においてタイミング調整の結果、競合点以外のイベントでの競合が生じたため、タイミング調整 ( 以降の競合シナリオ作成 ) は不可能と判断し ( ステップ  $S 5 1 3$  )、以上のタイミング調整にかかる処理を終了する。

【 0 0 5 0 】

また、上記ステップ  $S 5 0 9$  において、競合シナリオ生成部 1 0 3 は、イベント  $E v j$  における時刻の不一致が確認できた場合には ( ステップ  $S 5 0 9$  :  $Y e s$  )、競合点における他の組のランザクション  $T r i$  があるか判断する ( ステップ  $S 5 1 4$  )。そして、他の組のランザクション  $T r i$  が残っている場合には ( ステップ  $S 5 1 4$  :  $Y e s$  )、競合シナリオ生成部 1 0 3 は、変数  $k$  をインクリメントし ( ステップ  $S 5 1 5$  )、ステップ  $S 5 0 8$  に戻り、新たなランザクション  $T r i + k$  を用いた 1 組のランザクションにおけるタイミングの調整処理を行う。

【 0 0 5 1 】

一方、他の組のランザクション  $T r i$  がない場合には ( ステップ  $S 5 1 4$  :  $N o$  )、競合シナリオ生成部 1 0 3 は、競合点における全ての組のランザクション  $T r i$  に対するタイミング調整が有効であり、以降の競合シナリオの作成が可能であると判断し、以上のタイミング調整の処理を終了する。

【 0 0 5 2 】

( 2 : 優先度順にランザクションの各組み合わせのシミュレーション実行による競合発生の有無の確認と、競合シナリオの生成の処理説明 )

図 6 は、競合シナリオ生成処理の一例を示すフローチャートである。競合シナリオ生成部 1 0 3 が行う処理内容について説明する。図 6 は、図 5 に示した処理の実行後に競合シナリオ生成部 1 0 3 が実行し、選択した 1 組のランザクション毎の競合シミュレーションをシミュレータに実行させる。そして、競合が起こるランザクションの組み合わせを競合シナリオとして生成する。

【 0 0 5 3 】

はじめに、競合シナリオ生成部 1 0 3 は、初期設定を行う ( ステップ  $S 6 0 1$  )。ランザクション  $T r 1$  を  $d 1$  とし、他のランザクション  $T r 2$  以降を  $d 2 \sim d n$  とする。また、フラグを初期値 ( 0 ) に設定する。フラグは、イベントの開始時間変更時に 1 に変更される。

【 0 0 5 4 】

10

20

30

40

50

次に、競合シナリオ生成部 103 は、トランザクション  $d_1 \sim d_n$  の組み合わせをシミュレータに出力し、シミュレーションを実行させる（ステップ S602）。この際、図 4 および図 5 を用いて説明した選択処理により、優先度が高い 1 組のトランザクションを最初に選択し、シミュレータにシミュレーション実行させる。そして、競合シナリオ生成部 103 は、シミュレーション結果が、所望する一対のイベントにおいて実際に競合条件が満たされたかを判断する（ステップ S603）。

【0055】

シミュレーション実行により実際に競合条件が満たされたシミュレーション結果が得られた場合には（ステップ S603：Yes）、競合シナリオ生成部 103 は、競合テスト用タイミング調整が成功したと判断する。そして、競合シナリオ生成部 103 は、ステップ S602 にて実行したトランザクションの組み合わせの情報を示す競合シナリオを生成し（ステップ S604）、処理を終了する。

10

【0056】

一方、シミュレーション実行により競合条件が満たされないシミュレーション結果の場合には（ステップ S603：No）、競合シナリオ生成部 103 は、各組み合わせの一方のトランザクション  $d_1$  以外のトランザクション  $d(d_2 \sim d_n)$  について、各イベント  $E_{vj}$  に所定の時間差を加える（ステップ S605）。時間差により、上述したように、イベント  $E_{vj}$  の開始時間が変更される。

【0057】

次に、競合シナリオ生成部 103 は、フラグが 1 であるか判断する（ステップ S606）。フラグが 1 でなければ（ステップ S606：No）、開始時間の変更に対応してフラグを 1 に変更し（ステップ S607）、ステップ S602 に戻り、ステップ S602 以下の処理を再実行する。開始時間の変更後のシミュレーション実行により、競合条件が満たされれば（ステップ S603：Yes）、タイミング調整が成功した（ステップ S604）、と判断できるようになる。ステップ S602～ステップ S607 の処理は、既存の競合タイミング検証と同等の処理である。

20

【0058】

一方、ステップ S606 にて、フラグが 1 であれば（ステップ S606：Yes）、競合シナリオ生成部 103 に設けられたアサーション生成部は、アサーションの生成を行う（ステップ S608）。このアサーションは、各イベントにおける競合があり得るかの組の情報であり、例えば、 $(e_1 | - > ! e_2) || (e_1 | - > ! e_3) \dots (e_1 | - > ! e_n)$  との記載で表し、各イベント  $e_1, e_2 \dots$  からのそれぞれのリクエスト (Req) に関する情報を含む。そして、フォーマル検証ツールにこのアサーションを渡し、シミュレータによるシミュレーション時のフォーマル分析を実行させる（ステップ S609）。フォーマル検証ツールからは、分析結果としてアサーションが証明されたか否かを競合シナリオ生成部 103 に返答する。

30

【0059】

そして、競合シナリオ生成部 103 は、アサーションが証明された場合には（ステップ S610：Yes）、ステップ S612 に移行する。アサーションが証明されていない場合には（ステップ S610：No）、競合シナリオ生成部 103 は、次に高い優先度のトランザクションの組を選定し（ステップ S611：Yes）、ステップ S601 に戻る。

40

【0060】

ステップ S611 において、次に優先の高いトランザクションの組がない場合（ステップ S611：No）には、ステップ S612 に移行する。ステップ S612 では、競合シナリオの生成が不可能であり失敗であると判断し（ステップ S612）、以上の競合シナリオ生成の処理を終了する。

【0061】

（競合テスト用タイミング調整装置のハードウェア構成）

図 7 は、競合テスト用タイミング調整装置のハードウェア構成の一例を示すブロック図である。図 7 において、競合テスト用タイミング調整装置 700 は、CPU 701 と、R

50

read - Only Memory (ROM) 702と、Random Access Memory (RAM) 703と、を含む。また、半導体メモリやディスクドライブ等の記憶部704と、ディスプレイ708と、通信インターフェース(I/F)709と、キーボード710と、マウス711と、スキャナ712と、プリンタ713とを備えてもよい。これらCPU701～プリンタ713はバス714によってそれぞれ接続されている。

#### 【0062】

CPU701は、競合テスト用タイミング調整装置700の全体の制御を司る演算処理装置である。ROM702は、競合テスト用タイミング調整装置700の競合テスト用タイミング調整プログラムを記憶する不揮発性メモリである。RAM703は、CPU701による演算処理実行時のワークエリアとして使用される揮発性メモリである。

10

#### 【0063】

通信インターフェース709は、ネットワーク715と内部のインターフェースを司り、外部装置からのデータの入出力を制御する。具体的に、通信インターフェース709は、通信回線を通じてネットワーク715となるLocal Area Network (LAN)、Wide Area Network (WAN)、インターネットなどに接続され、ネットワーク715を介して他の装置に接続される。通信インターフェース709には、例えば、モデムやLANアダプタなどを採用することができる。

#### 【0064】

ディスプレイ708は、競合テスト用タイミング調整処理のための設定画面やタイミング調整結果について、カーソル、アイコンあるいはツールボックスをはじめ、文書、画像、機能情報などをデータ表示する装置である。ディスプレイ708には、例えば、Thin Film Transistor (TFT) 液晶ディスプレイ、プラズマディスプレイ、有機ELディスプレイなどを採用することができる。

20

#### 【0065】

上述した図1を用いて説明した競合テスト用タイミング調整の機能(イベントテーブル生成部101、競合シナリオ生成部103)は、図7に記載のCPU701がROM702に格納された競合テスト用タイミング調整プログラムを実行処理して得ることができる。

#### 【0066】

(競合テスト用タイミング調整装置の配置を示す図)

30

図8は、競合テスト用タイミング調整装置の配置例を示す図である。フォーマル検証ツール801と、シミュレータ802との間のインターフェースとして競合テスト用タイミング調整装置100(競合シナリオ生成部103)が配置される。

#### 【0067】

競合シナリオ生成部103には、上述したように、競合点での競合を確認するために、競合点で競合させたいイベントの組(例えば、Ev4, Ev5)の情報が入力される。競合シナリオ生成部103は、上述した(1)、(2)の処理により、このイベントの組の情報に基づいて、このイベントに対応するトランザクションの組を生成する。そして、競合点において競合させたいイベントが同時に起こるように、トランザクションのデータd1...dnに対するタイミング調整を更新する。

40

#### 【0068】

この更新は、図8のデータ入力シーケンス更新処理部103aが実行し、シミュレータ802に対してタイミング調整し、更新したトランザクションd1...dnを出力する。シミュレータ802は、シミュレーション結果t1...tnをデータ入力シーケンス更新処理部103aに返答する。

#### 【0069】

また、アサーション生成部103bは、アサーションを作成してフォーマル検証ツール801に出力する。フォーマル検証ツール801は、フォーマル分析結果をアサーション生成部103bに返答する。

#### 【0070】

50

そして、競合シナリオ生成部 103 は、データ入力シーケンス更新処理部 103 a、およびアサーション生成部 103 b からの返答に基づいて、競合点において競合させたいイベントが同時に起こり、かつ、他のイベント（競合点よりも前のデータパス上のブロック）における競合を回避する旨の競合シナリオを生成し、出力する。競合シナリオは、シミュレータ 802 に出力されるものであり、シミュレータ 802 に対する競合可能データ入力シーケンスをテキスト等により記載したものである。

#### 【0071】

（競合点でのタイミング競合の例）

図 9 は、競合点でのタイミング競合の一例を示す図である。コア 901 には、複数のデータパスが配置され、データパス 1（902 a）上の複数のブロック（FF1～FFn）と、データパス 2（902 b）上の複数のブロック（FF1～FFm）が配置されている。これらデータパス 1, 2 の入力側からみた奥部のブロックとしてアービタ 903 が配置されているとする。そして、このアービタ 903 におけるデータ入力の競合時の動作を検証する場合には、アービタ 903 の直前の FFn と FFm からのリクエスト（Req）を同時に発生させる必要がある。

#### 【0072】

このため、テストベンチ 910 は、データパス 1, 2 に対してデータを入力するドライバ 1（911 a）、ドライバ 2（911 b）の入力タイミングを調整する。テストベンチ 910 は、上述したフォーマル検証ツール 801、シミュレータ 802 および競合テスト用タイミング調整装置 100（競合シナリオ生成部 103）を含む。そして、競合シナリオ生成部 103 によって最適に調整されたタイミングを有してデータが入力され、アービタ 903 は、直前の FFn と FFm からのリクエスト（Req）を同時に発生させることができる。

#### 【0073】

これにより、コア 901 の奥部に位置するアービタ 903 等のブロックであっても、競合を発生させることができるようになり、競合発生時のブロックの動作（アービタ 903 の調停動作）をモニタ等で確認できるようになる。なお、データパス上の任意のブロック（イベント）を指定して競合テスト用タイミング調整装置に入力すれば、このブロックにおける競合についても同様に競合時の動作を確認できるようになる。

#### 【0074】

さらには、入力トランザクションのイベントのうち最大値の時刻を抽出するとともに、一对のイベントが入力されるブロックを特定してもよい。そして、イベントテーブル 112 を参照して、データパスの最奥部に位置するブロック（例えばアービタ 903）を競合点として競合点を自動探索することもできる。これにより、競合させたい一对のイベントの指定入力を省くことができる。

#### 【0075】

以上説明した実施の形態によれば、競合点において競合させたいイベントを同時に起こすことができるようにトランザクションの入力タイミングを調整することができる。この際、入力タイミングを調整したトランザクションにおいて競合点以外のイベントにおける競合の有無を確認する。これにより、競合点以外のデータパス上の前段のブロックでのイベント発生による遅延発生の可能性を確認でき、競合点において競合させたいイベントの入力タイミングを最適に調整できるようになる。

#### 【0076】

これにより、単にトランザクションの入力タイミング（開始時刻）を遅延させただけの場合に、他のブロックでのイベント発生を要因として、競合点で競合のタイミングがずれ、競合が生じない、状態となることを回避できる。そして、競合点で競合させたいイベントを同時に起こるトランザクションを用いて競合タイミングテスト（競合検証）を行えるようになる。

#### 【0077】

なお、本実施の形態で説明したプログラムは、予め用意されたプログラムをコンピュー

10

20

30

40

50

タで実行することにより実現することができる。また、このプログラムは、半導体メモリ、ハードディスク、フレキシブルディスク、CD-ROM、MO、DVD等のコンピュータで読み取り可能な記録媒体に記録され、コンピュータによって記録媒体から読み出されることによって実行される。また、このプログラムは、インターネット等のネットワークを介して配布してもよい。

【0078】

上述した実施の形態に関し、さらに以下の付記を開示する。

【0079】

(付記1) 半導体装置の内部信号を任意のイベントで競合させる競合タイミングを調整する競合テスト用タイミング調整プログラムにおいて、

10

コンピュータに、

競合させたい一对のイベントの指定により、前記半導体装置の入力のトランザクションと前記イベントとに基づき、前記一对のイベントに関連する複数の前記トランザクションを1組毎に選択させ、

選択した1組の前記トランザクションに含まれる前記一对のイベントの時刻が一致するように、一方のトランザクションの入力タイミングを調整させ、

入力タイミングが調整された一方のトランザクションに含まれる他のイベントの時刻が、他方のトランザクションに含まれる同一のイベントの時刻と不一致となるかを判断させ、

判断結果として不一致が得られた1組のトランザクションと、調整した前記入力タイミングと、を競合させたい一对のイベントに対するタイミング調整結果として出力させる、処理を実行させることを特徴とする競合テスト用タイミング調整プログラム。

20

【0080】

(付記2) 前記半導体装置の入力のトランザクションと、前記イベントとを関連付けたイベントテーブルを生成させ、

競合させたい一对のイベントの入力に基づき、前記イベントテーブルを参照して、前記一对のイベントに関連する複数の前記トランザクションを1組毎に選択させることを特徴とする付記1に記載の競合テスト用タイミング調整プログラム。

【0081】

(付記3) 前記判断結果として一致が生じた前記一方のトランザクションには低い優先度を付与することを特徴とする付記1または2に記載の競合テスト用タイミング調整プログラム。

30

【0082】

(付記4) 前記判断結果として一致が生じた後、前記一对のイベントに関連する前記トランザクションの組が残っていない場合には、一对のイベントに対するタイミング調整が不可であると判断することを特徴とする付記1～3のいずれか一つに記載の競合テスト用タイミング調整プログラム。

【0083】

(付記5) 前記トランザクションをシミュレータに実行させ、前記イベントが生じた時刻を前記イベントテーブルに記録させることを特徴とする付記2～4のいずれか一つに記載の競合テスト用タイミング調整プログラム。

40

【0084】

(付記6) 前記タイミング調整結果により得られた複数組のトランザクションをシミュレータに出力させ、

前記シミュレータのシミュレーション結果に基づき、前記一对のイベントにおける競合条件が満たされた場合に、一对のイベントに対するタイミング調整に関する情報を競合シナリオとして出力させる、

ことを特徴とする付記1～5のいずれか一つに記載の競合テスト用タイミング調整プログラム。

【0085】

50

(付記 7) 前記判断結果として一致が生じた前記一方のトランザクションは、前記イベントテーブルの優先度が低い格納位置に変更させることを特徴とする付記 2 ~ 6 のいずれか一つに記載の競合テスト用タイミング調整プログラム。

【 0 0 8 6 】

(付記 8) 選択した 1 組の前記トランザクションに含まれる前記一对のイベントの時刻のうち、時刻が最大値のイベントの時刻が一致するように、一方のトランザクションの入力タイミングを調整させることを特徴とする付記 1 ~ 7 のいずれか一つに記載の競合テスト用タイミング調整プログラム。

【 0 0 8 7 】

(付記 9) 半導体装置の内部信号を任意のイベントで競合させる競合タイミングを調整する競合テスト用タイミング調整方法において、

コンピュータが、

競合させたい一对のイベントの指定により、前記半導体装置の入力のトランザクションと前記イベントとに基づき、前記一对のイベントに関連する複数の前記トランザクションを 1 組毎に選択し、

選択した 1 組の前記トランザクションに含まれる前記一对のイベントの時刻が一致するように、一方のトランザクションの入力タイミングを調整し、

入力タイミングが調整された一方のトランザクションに含まれる他のイベントの時刻が、他方のトランザクションに含まれる同一のイベントの時刻と不一致となるかを判断し、

判断結果として不一致が得られた 1 組のトランザクションと、調整した前記入力タイミングと、を競合させたい一对のイベントに対するタイミング調整結果として出力する、

処理を実行することを特徴とする競合テスト用タイミング調整方法。

【 0 0 8 8 】

(付記 10) 半導体装置の内部信号を任意のイベントで競合させる競合タイミングを調整する競合テスト用タイミング調整装置において、

前記半導体装置の入力のトランザクションと、前記イベントとを関連付けたイベントテーブルと、

競合させたい一对のイベントの入力に基づき、前記イベントテーブルを参照して、前記一对のイベントに関連する複数の前記トランザクションを 1 組毎に選択し、

選択した 1 組の前記トランザクションに含まれる前記一对のイベントの時刻が一致するように、一方のトランザクションの入力タイミングを調整し、

入力タイミングが調整された一方のトランザクションに含まれる他のイベントの時刻が、他方のトランザクションに含まれる同一のイベントの時刻と不一致となるかを判断し、

判断結果として不一致が得られた 1 組のトランザクションと、調整した前記入力タイミングと、を競合させたい一对のイベントに対するタイミング調整結果として出力する競合シナリオ生成部と、

を有することを特徴とする競合テスト用タイミング調整装置。

【 0 0 8 9 】

(付記 11) 前記トランザクションをシミュレータに実行させ、前記イベントが生じた時刻を前記イベントテーブルに記録させることにより、前記半導体装置の入力のトランザクションと、前記イベントとを関連付けた前記イベントテーブルを生成するイベントテーブル生成部、

をさらに有することを特徴とする付記 10 に記載の競合テスト用タイミング調整装置。

【 符号の説明 】

【 0 0 9 0 】

1 0 0 競合テスト用タイミング調整装置

1 0 1 イベントテーブル生成部

1 0 2 イベントテーブル

1 0 3 競合シナリオ生成部

9 0 1 コア

10

20

30

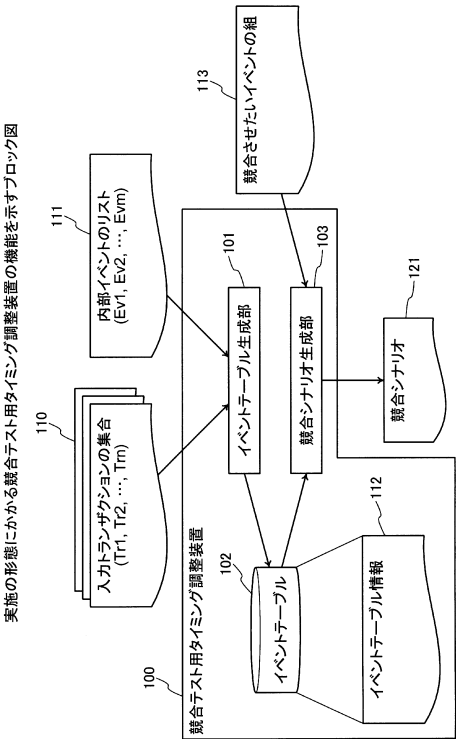
40

50

9 0 3    アービタ

9 1 0    テストベンチ

【 図 1 】



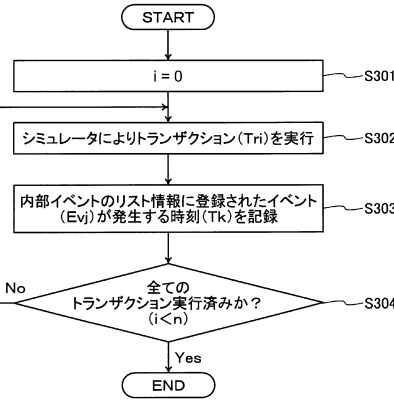
【 図 2 】

イベントテーブルに格納されるイベントテーブル情報を示す図表

	Ev1	Ev2	Ev3	Ev4	Ev5	...	Evm
Tr1	-	20	30	50	-	...	
Tr2		50	70	-	80	...	
Tr3	-	50	-	80	100	...	
⋮						...	
Trm						...	

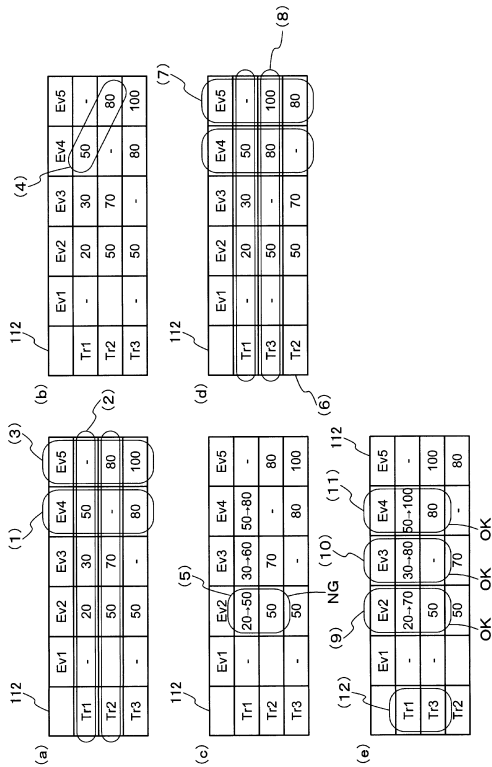
【 図 3 】

イベントテーブル生成の処理手順の一例を示すフローチャート



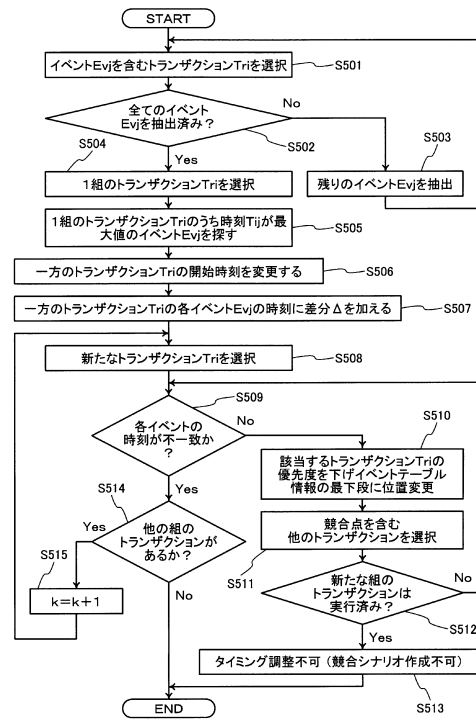
【図 4】

競合シナリオ生成にかかるトランザクション選択例を説明するための図表



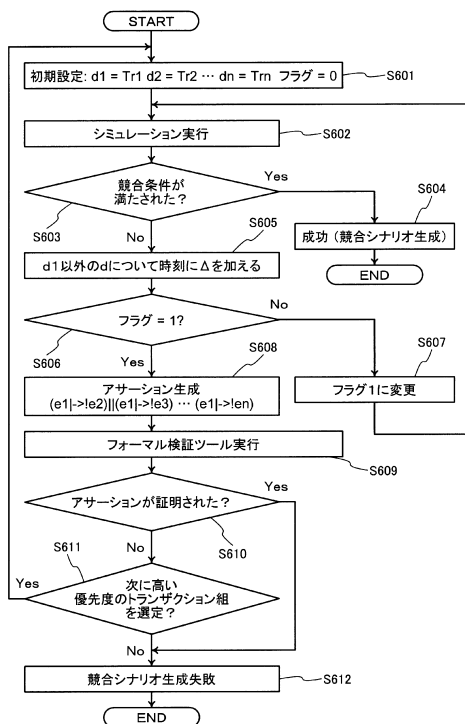
【図 5】

競合シナリオ生成のためのトランザクション選択処理の一例を示すフローチャート



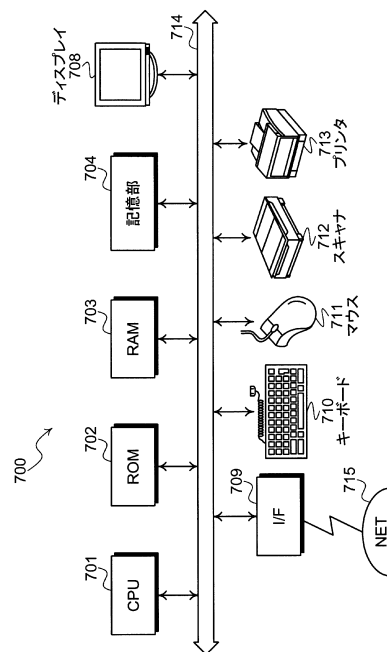
【図 6】

競合シナリオ生成処理の一例を示すフローチャート



【図 7】

競合シナリオ生成のためのハードウェア構成の一例を示すブロック図







---

フロントページの続き

(56)参考文献 特開平10-069503(JP,A)  
特開2010-244300(JP,A)  
特開2010-044622(JP,A)  
特開2001-222442(JP,A)  
米国特許第06772370(US,B1)

(58)調査した分野(Int.Cl., DB名)  
G06F 11/22-277