

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4933533号
(P4933533)

(45) 発行日 平成24年5月16日(2012.5.16)

(24) 登録日 平成24年2月24日(2012.2.24)

(51) Int. Cl. F I
GO 1 R 31/28 (2006.01) GO 1 R 31/28 H

請求項の数 18 (全 14 頁)

(21) 出願番号	特願2008-508894 (P2008-508894)	(73) 特許権者	591069226
(86) (22) 出願日	平成18年4月12日 (2006.4.12)		テラダイン・インコーポレーテッド
(65) 公表番号	特表2008-539429 (P2008-539429A)		TERADYNE INCORPORATED
(43) 公表日	平成20年11月13日 (2008.11.13)		ED
(86) 国際出願番号	PCT/US2006/013546		アメリカ合衆国マサチューセッツ州018
(87) 国際公開番号	W02006/118758		64, ノース・リーディング, リバーパーク・ドライブ 600
(87) 国際公開日	平成18年11月9日 (2006.11.9)	(74) 代理人	100140109
審査請求日	平成19年12月28日 (2007.12.28)		弁理士 小野 新次郎
(31) 優先権主張番号	60/676,005	(74) 代理人	100075270
(32) 優先日	平成17年4月29日 (2005.4.29)		弁理士 小林 泰
(33) 優先権主張国	米国 (US)	(74) 代理人	100080137
(31) 優先権主張番号	11/192,927		弁理士 千葉 昭男
(32) 優先日	平成17年7月29日 (2005.7.29)	(74) 代理人	100096013
(33) 優先権主張国	米国 (US)		弁理士 富田 博行

最終頁に続く

(54) 【発明の名称】 サイトアウェアオブジェクト

(57) 【特許請求の範囲】

【請求項1】

複数の被試験デバイス(DUT)を収容する複数のサイトを有する自動試験装置(ATE)と共に使用するための方法であって、該方法は、

前記複数のサイトと共に使用するオブジェクトを定義するステップであって、該オブジェクトは、前記複数のサイトに関連するデータを含み、前記複数のサイトのうちのアクティブなサイトにおけるアクティブ状態を判断し、所定の時間間隔に渡って前記アクティブなサイトにおけるアクティブ状態の変化を記録するものであり、該アクティブ状態は、前記アクティブなサイトと関連づけられた一つ又はそれ以上のDUTが試験に適していることを示す、ステップと、

前記ATEにより実行される、アクティブなサイトと関連付けられたDUTの試験中に、前記オブジェクトを使用するステップと、を含む方法。

【請求項2】

前記ATEは、関連するATEソフトウェアサイト管理インフラストラクチャを有することによって、前記複数のサイトを管理し、前記オブジェクトは、前記ATEソフトウェアサイト管理インフラストラクチャと通信して、どのサイトがアクティブであるかを判断する、請求項1に記載の方法。

【請求項3】

前記試験中に前記オブジェクトを使用するステップは、

前記オブジェクトを介して前記A T Eをプログラミングすること、
前記A T Eからの計器の読み取り値を前記オブジェクトに記憶すること、
前記オブジェクトを使用して前記計器の読み取り値を処理すること、及び
試験結果を前記オブジェクトに記録すること、
のうちの1つ又は複数を含む、請求項1に記載の方法。

【請求項4】

前記処理することは、前記計器の読み取り値を変更すること及び/又は該読み取り値を
試験サブシステムへ報告することを含む、請求項3に記載の方法。

【請求項5】

前記オブジェクトは、プログラミング言語で定義され、該プログラミング言語により、
単一のデータエレメントが該プログラミング言語で取り扱われるように取り扱われること
が可能である、請求項1に記載の方法。

10

【請求項6】

前記データは、どのサイトがアクティブであるかを識別するブールデータを含む、請求
項1に記載の方法。

【請求項7】

前記データは、ピンリストデータを含み、該ピンリストデータは、少なくとも1つのD
U Tの個々のピンにおける計器測定値に対応するピンデータを含む、請求項1に記載の方
法。

【請求項8】

20

前記オブジェクトはプログラミング言語で定義され、前記方法は、
前記プログラミング言語で演算を定義し、該演算は前記オブジェクトに特有のものであ
るステップをさらに含む、請求項1に記載の方法。

【請求項9】

前記演算は、前記プログラミング言語のネイティブ変数に対して行われる数学演算に対
応する数学的演算を含む、請求項8に記載の方法。

【請求項10】

複数の被試験デバイス(D U T)を収容する複数のサイトを有する自動試験装置(A T
E)と、

前記A T Eと通信する処理デバイスと、
を備えるシステムにおいて、前記処理デバイスは、

30

前記複数のサイトと共に使用するオブジェクトを定義する命令であって、

前記複数のサイトに関連するデータと、

前記複数のサイトのうちのアクティブなサイトにおけるアクティブ状態を判断し、所
定の時間間隔に渡る前記アクティブなサイトにおけるアクティブ状態の変化を記録するた
めのコンピュータ・コードであって、前記アクティブ状態が、前記アクティブなサイトと
関連付けられた一つ又はそれ以上のD U Tが試験に適していることを示すコンピュータ・
コードと、

を含むオブジェクトを定義する命令と、

前記A T Eにより実行される、アクティブなサイトと関連付けられたD U Tの試験中に
前記オブジェクトを使用する命令と、
を実行する、システム。

40

【請求項11】

前記A T Eは、関連するA T Eソフトウェアサイト管理インフラストラクチャを備える
ことによって、前記複数のサイトを管理し、前記オブジェクトは、前記A T Eソフトウェ
アサイト管理インフラストラクチャと通信して、どのサイトがアクティブであるかを判断
する、請求項10に記載のシステム。

【請求項12】

前記試験中に前記オブジェクトを使用することは、

前記オブジェクトを介して前記A T Eをプログラミングすること、

50

前記 A T E からの計器の読み取り値を前記オブジェクトに記憶すること、
前記オブジェクトを使用して前記計器の読み取り値を処理すること、及び
試験結果を前記オブジェクトに記録すること

のうちの1つ又は複数を含む、請求項 10 に記載のシステム。

【請求項 13】

前記処理することは、前記計器の読み取り値を変更すること及び/又は該読み取り値を
試験サブシステムへ報告することを含む、請求項 12 に記載のシステム。

【請求項 14】

前記オブジェクトは、前記処理デバイス上でプログラミングされるプログラミング言語
で定義され、該プログラミング言語により、単一のデータエレメントが該プログラミング
言語で取り扱われるように取り扱われることが可能である、請求項 10 に記載のシステム

10

【請求項 15】

前記データは、どのサイトがアクティブであるかを識別するブールデータを含む、請求
項 10 に記載のシステム。

【請求項 16】

前記データは、ピンリストデータを含み、該ピンリストデータは、少なくとも1つの D
U T の個々のピンにおける計器測定値に対応するピンデータを含む、請求項 10 に記載の
システム。

【請求項 17】

20

前記オブジェクトは、前記処理デバイス上でプログラミングされるプログラミング言語
で定義され、前記処理デバイスは、前記プログラミング言語で演算を定義する命令であっ
て、該演算が前記オブジェクトに特有のものである命令を実行する、請求項 10 に記載の
システム。

【請求項 18】

前記演算は、前記プログラミング言語のネイティブ変数に対して行われる数学演算に対
応する数学的演算を含む、請求項 17 に記載のシステム。

【発明の詳細な説明】

【技術分野】

【0001】

30

本発明は、概括的には、複数のデバイスを試験するのに使用される複数のサイトを有す
る自動試験装置 (A T E) に関し、より詳細には、複数のデバイスを試験する際に A T E
によって使用されるサイトウェアオブジェクトに関する。

【0002】

[優先権の主張]

本願は、米国仮特許出願第 60 / 676 , 005 号及び米国特許出願公開第 11 / 19
2 , 927 号の優先権を主張する。これらの出願の双方の内容は、本明細書にすべてが述
べられているように、参照により本出願に援用される。

【背景技術】

【0003】

40

A T E は、半導体、電子回路、プリント回路基板アセンブリ等のデバイスを試験する自
動化された、通常はコンピュータ駆動の手法を指す。半導体デバイス等、A T E によって
試験されるデバイスは、被試験デバイス (D U T) と呼ばれる。

【0004】

現代の A T E は、マルチサイト試験をサポートしている。マルチサイト試験をサポート
している A T E は、スロット、すなわち「サイト」を含む。試験されるデバイスは、各サ
イトに挿入され、A T E は、そのサイトのデバイスに対して必要ないかなる試験も行う。
コンピュータプログラムが、通常、試験に関与する。これらのコンピュータプログラムの
コードを変更することなく複数のサイトでデバイスを試験する能力は、「サイト透明性」
と呼ばれる。より詳細には、サイト透明性は、ユーザがこのようなタスクにほとんど責任

50

を持つことがなく、そうすることが必要とされるまで、自身の試験がマルチサイトであることをかろうじて考慮する必要だけがあるような、サイト及びサイトデータの自動管理を指す。有効である場合、ユーザは、プログラミングのサイト次元性 (site dimensionality) を意識し続けることはない。

【0005】

サイト透明性によって、ユーザは、試験プログラムコードに対して大幅な変更を行うことなく、複数のデバイスを試験する構成にすることが可能になるので、サイト透明性はATEにおける重要な利点である。

【発明の開示】

【課題を解決するための手段】

【0006】

本発明は、コンピュータプログラム製品を含み、サイトウェア (site-aware) オブジェクトを実施する、装置及び方法を記載する。

一態様においては、本発明は、一般的にサイトを有する自動試験装置 (ATE) と共に使用するための方法であって、サイトのそれぞれは被試験デバイス (DUT) を収容する方法を対象とする。その方法は、複数のサイトと共に使用するオブジェクトを定義し、オブジェクトは、複数のサイトの少なくともいくつかに関連するデータを含み、どのサイトがアクティブであるかを判断すること、及びATEによるDUTの試験中にオブジェクトを使用することを含む。この態様は、以下の特徴のうちの1つ又は複数も含むことができる。

【0007】

ATEは、関連するATEソフトウェアサイト管理インフラストラクチャを備えることができ、それによって、サイトを管理する。オブジェクトは、ATEソフトウェアサイト管理インフラストラクチャと通信して、どのサイトがアクティブであるかを判断することができる。試験中のオブジェクトの使用は、オブジェクトを介してATEをプログラミングすること、ATEからの計器の読み取り値をオブジェクトに記憶すること、オブジェクトを使用して計器の読み取り値を処理すること、及び試験結果をオブジェクトに記録することのうちの1つ又は複数も含むことができる。処理は、計器の読み取り値を変更すること及び/又は読み取り値を試験サブシステムへ報告することを含むことができる。

【0008】

オブジェクトは、プログラミング言語で定義されることができる。単一のデータエレメントが、プログラミング言語で取り扱われているように、オブジェクトは、プログラミング言語によって取り扱い可能である。データは、どのサイトがアクティブであるかを識別するブールデータを含むことができる。データはピンリストデータを含むことができる。ピンリストデータは、少なくとも1つのDUTの個々のピンにおける計器測定値に対応するピンデータを含むことができる。

【0009】

オブジェクトは、プログラミング言語で定義することができ、方法はまた、プログラミング言語で演算を定義し、演算は、オブジェクトに特有のものであることを含むことができる。演算は、プログラミング言語のネイティブ変数に対して行われる数学演算に対応する数学的演算を含むことができる。

【0010】

別の態様では、本発明は、一般に、サイトを有するATEと、ATEと通信する処理デバイスとを含むシステムを対象にする。ここで、サイトのそれぞれは、DUTを収容するように構成されている。処理デバイスは、上記態様の方法を遂行する命令を実行する。この態様は、上記態様と同じ特徴又は異なる特徴も含むことができる。

【0011】

1つ又は2つ以上の例示の詳細は、添付図面及び以下の説明で述べられている。本発明のさらに別の特徴、態様、及び利点は、説明、図面、及び特許請求の範囲から明らかになるであろう。

10

20

30

40

50

【 0 0 1 2 】

異なる図における同じ参照符号は、同じ要素を示す。

【 発明を実施するための最良の形態 】

【 0 0 1 3 】

図 1 は、複数の D U T を試験する際に使用されるシステム 1 0 を示している。システム 1 0 は A T E 1 2 を含む。A T E 1 2 は、デバイスインターフェース基板 (D I B) 1 4 を含む。D I B 1 4 は、A T E にデバイスを接続するのに使用される。D I B 1 4 は、複数のサイト 1 6 a ~ 1 6 d を有する。D U T は、A T E を使用して試験を行うために各サイトに挿入することができる。図 1 には 4 つのサイトしか示されていないが、A T E 1 2 は、任意の個数のサイトを含むことができる。また、図 1 の例では、1 スロットにつき 1 つの D U T があるが、すべてのサイトが占有される必要はない。

10

【 0 0 1 4 】

各 D U T は、A T E を使用して試験を行うことができるデバイスならば、どのタイプのデバイスであってもよい。D U T の例には、半導体デバイス及び電子アセンブリが含まれるが、これらに限定されるものではない。

【 0 0 1 5 】

A T E 1 2 は、デジタル信号プロセッサ (D S P) 1 8 等の内部処理デバイスを含む。D S P 1 8 は、A T E ソフトウェア 2 0、すなわち、実行可能コードから成るコンピュータプログラム (複数可) の全部又は一部を実行して、サイト 1 6 a ~ 1 6 d の D U T を試験することができる。A T E ソフトウェア 2 0 は、A T E 1 2 のメモリ (図示せず) に記憶することもできるし、後述するコンピュータ 2 2 等の外部処理デバイスから提供することもできる。A T E ソフトウェア 2 0 は、サイト 1 6 a ~ 1 6 d を管理する A T E ソフトウェアサイト管理インフラストラクチャ 2 4 を含む。

20

【 0 0 1 6 】

特に、A T E ソフトウェアサイト管理インフラストラクチャは、A T E ソフトウェア 2 0 とサイト 1 6 a ~ 1 6 d との間における情報の交換を制御する。この情報は、サイトの D U T からのデータ、及び / 又は、サイトの D U T へ送信されるデータを含むことができる。D U T へのデータの例には、波形又は他の刺激等の試験データ、及び、D U T をプログラミングするために A T E ソフトウェア 2 0 によって使用される構成データが含まれるが、これらに限定されるものではない (一実施形態では、D S P は D U T からの信号を処理するだけであることに留意されたい)。D U T からのデータの例には、サイトがアクティブであるか否かに関する表示、及び、D U T の個々のピンからの読み取り値を含む、サイトの D U T からの読み取り値が含まれるが、これらに限定されるものではない。サイトは、そのサイトがピンアウト (bin-out) し、そのサイトの D U T が必要なすべての試験を終えて合格したか又は D U T が 1 つ又は 2 つ以上の必要な試験に不合格であったことを意味する場合に、非アクティブであるとみなすことができることに留意されたい。いずれの場合も、それ以上試験は行われぬ。或いは、サイトは、そのサイトに D U T がいない場合、又は、サイトが、試験を実行しているアドミニストレータによって非アクティブを指定された場合に非アクティブであるとみなすことができる。

30

【 0 0 1 7 】

A T E 1 2 は、コンピュータ 2 2 と通信する。コンピュータ 2 2 は、あらゆるタイプの処理デバイスとすることができる。このあらゆるタイプの処理デバイスには、従来のパーソナルコンピュータ (P C)、デスクトップデバイス若しくはハンドヘルドデバイス、マルチプロセッサコンピュータ、マイクロプロセッサベースの消費者用電子デバイス若しくはプログラマブル消費者用電子デバイス、ミニコンピュータ、メインフレームコンピュータ、及び / 又はパーソナルモバイルコンピューティングデバイスが含まれるが、これらに限定されるものではない。図 1 の例では、コンピュータ 2 2 は、1 つ又は 2 つ以上のプロセッサ (単に「プロセッサ 2 6」と呼ばれる) 及びメモリ 2 8 を含む。メモリ 2 8 は、試験ソフトウェア 3 0 を記憶する。試験ソフトウェア 3 0 は、A T E 1 2 で D U T を試験するためにプロセッサ 2 6 によって実行される。試験中、試験ソフトウェア 3 0 は、上述し

40

50

た A T E ソフトウェア サイト管理 インフラストラクチャを含めて、A T E ソフトウェア 20 と相互作用する。たとえば、試験ソフトウェア 30 は、サイト 16 a ~ 16 d の D U T に試験データを提供し、且つ、このような試験の結果を試験ソフトウェア 30 又は指定されたサブシステムへ報告するように A T E ソフトウェア 20 に命令することによって試験を制御することができる。このために、試験ソフトウェア 30 は、1 つ又は 2 つ以上のグラフィカルユーザインターフェース (G U I) (図示せず) を生成することができる。この G U I を通じて、アドミニストレータは、A T E 12 における試験を制御することができる。

【 0 0 1 8 】

これまで、コンピュータ 22 の試験ソフトウェアは、手続き型開発環境 (P D E : procedural development environment) と呼ばれる対話型グラフィカル環境と共に動作していた。P D E は、サイト透明性を実施するコードを含み、それによって、試験ソフトウェア開発者はプロセスフローに集中することが可能であった。その結果作成される試験ソフトウェアは、したがって、P D E に依存していた。一方、システム 10 は、P D E を含む必要がない。その代わりに、試験ソフトウェア 30 は、試験中に、サイトウェアオブジェクト (後述) を利用して、サイト透明性を実施する。

【 0 0 1 9 】

この実施形態では、試験ソフトウェア 30 は、ビジュアルベーシックフォアアプリケーション (Visual Basic for Application) (ビジュアルベーシック) で記述される。ビジュアルベーシックは、ベーシックと呼ばれる以前の D O S バージョンから発展した高水準プログラミング言語である。ビジュアルベーシックは、視覚的なイベント駆動型プログラミング言語である。これが、以前のベーシックとの主な相違である。ベーシックでは、プログラミングは、テキストのみの環境で行われ、プログラムは、シーケンシャルに実行される。ビジュアルベーシックでは、プログラミングは、オブジェクトを使用するグラフィカル環境で行われる。

【 0 0 2 0 】

オブジェクトは、データと、そのデータを操作するために実行されるコードとの双方を含む自己完結的エンティティである。ユーザは、ビジュアルベーシックにおいてオブジェクトをランダムに選択することができるので、各オブジェクトは、このような行動に対応できるように独立してプログラミングされる。ビジュアルベーシックプログラムは、このように、多数のサブプログラムから構成することができ、それらの多数のサブプログラムのそれぞれは、それ自身のプログラムコードを有し、独立して、且つ、他のサブプログラムと同時に実行することができる。

【 0 0 2 1 】

A T E 12 によって使用されるサイトウェアオブジェクトは、マルチサイトデータを表すように設計されたデータオブジェクトである。このようなデータは、A T E 計器をプログラミングし、測定された読み取り値を保持し、それらの読み取り値を操作し、且つ / 又は、それらの読み取り値を、D U T の配置 (dispositioning) (たとえば、区分け (ビンニング : binning)) 及び試験結果の記録 (データログ) を担当するサブシステムへ報告するのに使用することができる。サイトウェアオブジェクトという名前は、それらのサイトウェアオブジェクトが A T E ソフトウェア サイト管理 インフラストラクチャ 24 に内部接続され、したがって、A T E のシステムサイトの状況に対する経時的な変更、たとえば、毎時間、毎分、毎秒等に検出される定期的な変更を認識する (アウェア : aware) ということに起因している。サイトウェアオブジェクトは、表面的には対立する 2 つのモデルをユーザに提供する。サイトウェアオブジェクトは、サイトの概念を基本的に無視しつつ試験コードを書き込むことをユーザに可能にすると同時に、必要な時にサイト特有のデータにアクセスすることをユーザに可能にする。

【 0 0 2 2 】

図 2 を参照すると、この実施形態では、サイトウェアオブジェクト 32 は、D I B 14 のサイト 16 a ~ 16 d のそれぞれを識別するデータ 34 を含む。また、サイトウェア

10

20

30

40

50

アオブジェクト34は、サイトのそれぞれにおけるDUTに関連するデータ36も含む。DUTに関連するデータは、読み取り値、試験結果等、DUTから受信されたデータの場合もあるし、プログラム情報、試験データ等、DUTへ転送するデータの場合もある。

【0023】

サイトウェアオブジェクト32は、サイトウェアオブジェクト32が、サイトを、それらのサイトのDUTに関連するデータに関連させる意味でアレイに類似している。一方、サイトウェアオブジェクト32は、試験中にどのサイトがアクティブ/非アクティブであるかを知っている点でアレイとは異なる。このために、サイトウェアオブジェクト32は実行可能コード38も含む。実行可能コード38は、どのサイト16a~16dがアクティブ/非アクティブであるかを判断するために、ATEソフトウェアサイト管理インフラストラクチャ24と通信し、それに応じて、自身の記憶データを更新し、そして、試験ソフトウェア30と通信するのに使用される。

10

【0024】

この実施形態では、サイトウェアオブジェクトは、ビジュアルベーシックで使用される。しかしながら、サイトウェアオブジェクトは、ビジュアルベーシックの状況における使用に限定されるものではなく、あらゆるプログラミング言語で実施できることに留意されたい。一方、以下の説明は、ビジュアルベーシックと共に使用されるサイトウェアオブジェクトに適用される。

【0025】

ビジュアルベーシックでは、最も汎用の変数タイプはVariant(バリエーション)である。この変数タイプは、ビジュアルベーシックプログラミング環境で使用できる他のあらゆる変数タイプを保持するように設計されている。この実施形態では、SiteVariant(サイトバリエーション)と呼ばれるVariantに相当するサイトウェアが定義される。SiteVariantは、あらゆるタイプのマルチサイト値を保持することができる。

20

【0026】

SiteVariantオブジェクトの振る舞いは、以下のように要約することができる。SiteVariantオブジェクトは、複数のサイトのデータを含む。このデータは、単一の代入文で単一のスカラー値を代入される場合には、サイト一律(site-uniform)となることができ、測定によって返される場合には、サイト特有となることができる。SiteVariantオブジェクトは、「サイトウェア」である。SiteVariantは、使用される時、試験プログラムが現在サイトループ(site-loop)にあるか否かが分かっている。サイトループでは、1つのサイトのみが現在選択されている。SiteVariantオブジェクトが、サイトループにある(且つ、単一のサイトが選択されている)とき、又は、サイト一律である(且つ、すべてのサイトについて単一のスカラー値を有する)とき、SiteVariantオブジェクトは、他のあらゆるビジュアルベーシックのスカラーと同様に使用することができる。そうではなく、SiteVariantがサイトループの外部にあり、サイト特有のデータを保持するとき、SiteVariantは、マルチサイトとして取り扱われる。以下のコードは、SiteVariantがマルチサイトデータを保持するか否か、及び、SiteVariantがサイトループの内部で使用されるのか、又は外部で使用されるのかに応じて、SiteVariantの2重機能を実例で説明するものである。以下の例のDebug.Printの使用は、単一のスカラー変数を必要とするあらゆる用法のプロキシである。この例では、YYのような2重文字の変数名は、その変数がマルチサイトデータを保持することを示している。Xのような単一文字の変数名は、スカラーを保持し、この例では、標準的なビジュアルベーシックのDouble(ダブル)と全く同様に機能する。

30

40

【0027】

【数 1】

```
Dim Site As Variant
Dim X As New SiteDouble
Dim YY As New SiteDouble
```

```
X = 2.5 'Variable holds one number -- same number for all sites
```

```
Debug.Print X 'Prints 2.5
```

```
YY = thehdw.DCVI.Pins("Vdd").Meter.Read 'Meter Read returns site-specific data
-- a separate value for each site.
```

```
Debug.Print YY ' Runtime Error: "Object contains site-specific data" 10
```

```
For Each Site In theExec.Sites
```

```
    Debug.Print Site
```

```
    Debug.Print YY 'Works here. Object always evaluates to a scalar in a site
```

```
loop
```

```
Next Site
```

【0028】

SiteVariantは、サイト一律又はサイト特有のいずれかである。「サイト一律」は、すべてのサイトが、単一の代入による1つの共通値を共有することを意味する。「サイト特有」は、各サイトがそれ自身の値を有することを意味する。これらの用語は、単一の値を共有することを指し、値の数字が等しいことを指すものではないことに留意されたい。たとえば、サイトループがSiteDouble(サイトダブル)ですべてのサイトに5.0を代入していた場合、たとえすべてのサイトが同じ値を有していても、そのサイトループは依然としてサイト特有である。

【0029】

より特有のオブジェクトタイプは、特有のビジュアルベーシックの変数タイプを保持するSiteVariantから派生することができる。これらのオブジェクトタイプは、ビジュアルベーシックにネイティブな変数タイプに対応することができる。これらのより特有のオブジェクトタイプの例には、SiteDouble、SiteBoolean(サイトブール)、SiteLong(サイトロング)、及びPinData(ピンデータ)が含まれる。オブジェクトタイプSiteDouble、SiteBoolean、及びSiteLongは、ネイティブなビジュアルベーシックのタイプDouble(ダブル)(64ビット精度浮動小数点数)、Boolean(ブール)(論理の真又は偽)、及びLong(ロング)(32ビット整数)にそれぞれ対応するSiteVariantデータを表す。PinDataは、ピン名及びチャネル識別子等のデバイスピンに関係する特性を含むようにSiteVariantを拡張する。PinDataは、マルチピン・マルチサイト計器読み取り値をカプセル化するPinListData(ピンリストデータ)収集クラスで使用される。すなわち、PinListDataは、複数のサイト及びアクティブなサイトについて、DUTの個々のピンと、それらの個々のピンのそれぞれの測定に関連するデータ値とを識別する。

【0030】

より具体的には、PinListDataオブジェクトは、新しいSiteVariant使用モデルに統合されている。PinListDataオブジェクトは、ピン寸法をマルチサイトデータに追加する。したがって、PinListDataオブジェクトは、各ピン及び各サイトのデータを含むマトリックスと考えることができる。PinListDataオブジェクトは、各ピンにつき1つのPinDataオブジェクトを保持する。PinListDataの各ピンスライスは、ピン名を有するSiteVariantである。これは、PinDataオブジェクトと呼ばれる。実際には、単一のピンを含むどのPinListDataオブジェクトも、SiteVariantとして使用することもできるし、保持されたデータタイプが、そのより特有の派生物のデータタイプと一致した場合には、その派生物の1つとして使用することもできる。この理由により、ユーザは、PinListDataオブジェクトのドライバシntaxが単一のピンを含む場合には、もはやPinListDataオブジェクトを使用する必要はない。たとえば、ユーザは、単一ピンの測定値をSiteDoubleに直接書き込む

10

20

30

40

50

オプションを有する。PinListDataオブジェクトは、SiteVariantを保持するので、同じサイトウェアの振る舞い及び同じサイト一律/サイト特有の属性を有する。PinListDataのレガシーシンタックスは、まだ機能するが、このシンタックスの多くは、SiteVariantモデルと一致するインターフェースの方が選ばれることにより、隠れている。単一のコールで複数のサイト及び複数のピンに対する数学処理を行うPinListData数学処理メソッドが提供される。以下のコードは、4つのサイトの3対のピンにメータ測定を行い、その後、すべてのサイトにおけるすべての対の間のデルタを単一の文でデータログするものである。

【 0 0 3 1 】

【 数 2 】

```
Dim HighReading As New PinListData
Dim LowReading As New PinListData
Dim Delta As New PinListData
```

```
HighReading = thehdw.DCVI.Pins("A_pos, B_pos, C_pos").Meter.Read
LowReading = thehdw.DCVI.Pins("A_neg, B_neg, C_neg").Meter.Read
Delta = HighReading.Subtract(LowReading)
TheExec.Flow.TestLimit Delta
```

【 0 0 3 2 】

上記オブジェクトタイプは、派生させることができる唯一のタイプではない。Variantに現れ得るどのタイプも、SiteVariantの派生物の基礎を形成することができる。これらのタイプには、ユーザ定義のタイプ及び非ネイティブのビジュアルベーシックのタイプが含まれ得るが、これらに限定されるものではない。

【 0 0 3 3 】

サイトウェアオブジェクトは、可能な範囲内で、それらのサイトウェアオブジェクトに対応するビジュアルベーシックのタイプの振る舞いを模倣する。したがって、SiteVariantのプログラミングモデルは、Variantのプログラミングモデルとほぼ同じであり、SiteDoubleのプログラミングモデルは、Doubleのプログラミングモデルとほぼ同じであり、他のものも同様である。オブジェクトタイプとネイティブタイプとの間の1つの相違は、オブジェクトタイプが定義されて割り当てられるということである。もう1つの相違は、ビジュアルベーシックでは、ネイティブタイプに対するほとんどの数学演算が、たとえば「+」、「-」、「*」、「/」といった記号演算子の使用を伴うことである。一方、サイトウェアオブジェクトのプログラミングインターフェースは、これらの演算子に対応するメソッドを定義している。

【 0 0 3 4 】

より具体的には、ビジュアルベーシックプログラミング言語を使用すると、サイトウェアオブジェクトで実行できる演算（たとえば、数学演算、論理演算等）を定義することが可能である。これらの演算は、ネイティブなビジュアルベーシック演算に対応するが、コードを使用して定義される。これは、サイトウェアオブジェクトが、ネイティブなビジュアルベーシック変数とは異なる構造を有することから必要となる。例として、

$$y = x + 0.1$$

の加算演算は、ビジュアルベーシックのネイティブ変数「y」及び「x」に対して、0.1を「x」に加算し、且つ、その結果の値を「y」に代入するように実行することができる。同様の演算は、数学演算子「+」を使用せずに、サイトウェアオブジェクト「X」及び「Y」についてもビジュアルベーシックで行うことができる。その代わりに、関数「add」が、サイトウェアオブジェクトと共に使用するためにビジュアルベーシックで定義され得る。この関数を使用すると、以下のように、サイトウェアオブジェクト「X」の（1つのサイトについての）各データエントリーに0.1の値を加算して、それらの値を「Y」の対応するエントリーに代入することが可能である。

【 0 0 3 5 】

10

20

30

40

50

$Y = X.add(0.1)$

異なる関数を使用して、他の演算を定義することもできる。このような関数を使用すると、単一のネイティブデータエレメントがビジュアルベーシック（又は他のプログラミング言語）で取り扱われる場合に、サイトウェアオブジェクトは、ビジュアルベーシック（又は他のプログラミング言語）により取り扱うことができる。この機能によって、サイトウェアオブジェクトを本質的にビジュアルベーシックのオブジェクトとして取り扱うことが可能になり、それによって、プログラマは、ATE 12の複数のサイトを考慮するように試みる代わりに、プロセスフローに集中することが可能になるので、この機能は、サイト透明性を促進させる。

【 0 0 3 6 】

上記演算定義は、サイトウェアオブジェクトに対する固有の限定ではない。そうではなく、演算定義の必要性は、ビジュアルベーシックプログラミング環境の特徴である。ビジュアルベーシックプログラミング環境は、演算子の多重定義（operator overloading）をサポートしていない。異なるプログラミング環境では、サイトウェアオブジェクトは、記号演算子のサポートを有する場合がある。

【 0 0 3 7 】

図3は、サイトウェアオブジェクトと共にATE 12を使用するためのプロセス40を示している。プロセス40は、サイト16a～16dと共に使用するためのサイトウェアオブジェクトを定義する（42）。サイトウェアオブジェクトは、ユーザ入力に従って定義することができる。サイトウェアオブジェクトは、サイト16a～16dのDUTの試験中、ATE 12によって使用される（44）。上述したように、サイトウェアオブジェクトは、ATEソフトウェアサイト管理インフラストラクチャ24と通信して、どのサイトがアクティブであるかを判断する。詳細には、サイトウェアオブジェクトは、特定のDUTが選択されていることの表示を受信し（46）、その表示をサイトウェアオブジェクトに記憶する（48）。

【 0 0 3 8 】

試験中、サイトウェアオブジェクトは、あらゆる目的に使用することができる。この目的には、ATEのプログラミング、ATEからの読み取り値のサイトウェアオブジェクトへの記憶、試験結果のサイトウェアオブジェクトへの記録、及びサイトウェアオブジェクトを使用する読み取り値の処理が含まれるが、これらに限定されるものではない。読み取り値の処理には、読み取り値の変更及び/又は読み取り値の試験サブシステムへの報告が含まれる。

【 0 0 3 9 】

この実施形態では、SiteVariantオブジェクトは、ビジュアルベーシックのようなスカラー変数使用モデルとして提供される。これは、ビジュアルベーシックの特徴、すなわち、オプションのパラメータを有するデフォルトの隠れプロパティ、並びに、同じプロパティの異なるタイプのLet及びGetを行う能力の組み合わせを通じて達成される。各SiteVariantオブジェクトは、Long=-1として、オプションのパラメータSite（サイト）を有する隠れデフォルトプロパティ.Value()を有する。デフォルトのSiteパラメータは、サイトウェアの作用を起動し、サイトループにおけるすべての選択されたサイト又は単一の選択されたサイトに対して操作を行う。以下の同等なシンタックスを比較する（X及びZはSiteDoubleである一方、yは通常のビジュアルベーシックのDoubleである）。

【 0 0 4 0 】

用法：	同等な用法
X = 5	X.Vaule(-1)=5
y = X	y=X.Value(-1)
Z = X	Z.Value(-1)=X

上記は、.Value及び-1が共にデフォルトであるので同等である。

【 0 0 4 1 】

.Valueプロパティを隠れデフォルトにすることは、この使用モデルの近道だけではなく

10

20

30

40

50

それ以上のものである。デフォルトメカニズムが、多様な作用を追加するのに使用される。これらのSiteVariantは、シンタックスに応じて数字又はオブジェクトとして機能する。ビジュアルベーシックは、指定されたプロパティもメソッドも有しないオブジェクトに遭遇すると、どのタイプがその用法で必要とされているかに応じて、そのオブジェクトをオブジェクトそのものとして解釈することもできるし、オブジェクトのデフォルトプロパティへのコールの結果として解釈することもできる。上記 $y = X$ のラインでは、Doubleが左端に必要とされる。XはDoubleではないので、ビジュアルベーシックは、オブジェクトXのデフォルトプロパティを試み、デフォルトプロパティが確かにDoubleを返すことを見出す。他方、 $Z = X$ のラインでは、Zのデフォルトプロパティがオブジェクト (Variant) を受け付け、その結果、ビジュアルベーシックは、オブジェクトであるXのZへの直接代入を行う。たとえば、.Value()プロパティのGetバージョンがDoubleであっても、ビジュアルベーシックによって、.Value()プロパティのLetバージョンがVariantであることが可能になることから、この直接代入は可能である。デフォルト用法の簡易性及びそのビジュアルベーシックの変数との一貫性によって、.Valueプロパティを隠蔽することが可能になる。

10

【 0 0 4 2 】

本明細書で説明したプロセス及びサイトウェアオブジェクトは、どの特定のハードウェアとの使用にも、どの特定のソフトウェアとの使用にも、どの特定のプログラミング言語との使用にも限定されない。これらのプロセス及びサイトウェアオブジェクトは、どのコンピューティング環境又はどの処理環境においても適用性を見出すことができ、マシン可読命令を実行できるどのタイプのマシンとも適用性を見出すことができる。これらのプロセス及びサイトウェアオブジェクトの全部又は一部は、デジタル電子回路機構で実施することもできるし、コンピュータのハードウェア、ファームウェア、ソフトウェアで実施することもできるし、それらの組み合わせで実施することもできる。

20

【 0 0 4 3 】

プロセス及びサイトウェアオブジェクトの全部又は一部は、データ処理装置によって実行されるか又はデータ処理装置のオペレーションを制御するコンピュータプログラム製品、すなわち、情報担体に有形に具現化されたコンピュータプログラムとして実施することができる。この情報担体は、たとえば、マシン可読ストレージデバイス又は伝搬信号である。データ処理装置は、たとえば、プログラマブルプロセッサ、コンピュータ、又は複数のコンピュータである。コンピュータプログラムは、コンパイラ型言語又はインタープリタ型言語を含むあらゆる形のプログラミング言語で記述することができ、スタンドアロンプログラムとして、又は、モジュール、コンポーネント、サブルーチン、若しくはコンピューティング環境での使用に適した他のユニットとしての形を含むあらゆる形で配備することができる。コンピュータプログラムは、1つのサイトの1つのコンピュータ、又は、複数のサイトにわたって分散し通信ネットワークにより相互接続された複数のコンピュータで実行されるように配備することができる。

30

【 0 0 4 4 】

プロセス及びサイトウェアオブジェクトに関連する方法ステップは、それらのプロセス及びサイトウェアオブジェクトの機能を実行するために1つ又は2つ以上のコンピュータプログラムを実行する1つ又は2つ以上のプログラマブルプロセッサが実行することができる。また、方法ステップは、たとえばFPGA (フィールドプログラマブルゲートアレイ) 及び/又はASIC (特定用途向け集積回路) といった専用論理回路機構によって実行することもでき、プロセス及びサイトウェアオブジェクトは、そのような専用論理回路機構として実施することができる。

40

【 0 0 4 5 】

コンピュータプログラムの実行に適したプロセッサには、例として、汎用マイクロプロセッサ及び専用マイクロプロセッサの双方、並びに、あらゆる種類のデジタルコンピュータのあらゆる1つ又は2つ以上のプロセッサが含まれる。一般に、プロセッサは、読み出し専用ストレージエリア若しくはランダムアクセスストレージエリア又はその双方から命

50

令及びデータを受け取る。コンピュータの要素は、命令を実行するためのプロセッサ、並びに、命令及びデータを記憶するための1つ又は2つ以上のストレージエリアデバイスを含む。一般に、コンピュータは、たとえば、磁気ディスク、光磁気ディスク、若しくは光ディスクといった、データを記憶するための1つ若しくは2つ以上のマストレージデバイスも含むか、又は、そのようなマストレージデバイスからのデータの受信若しくはそのようなマストレージデバイスへのデータの転送若しくはそれらの双方を行うよう動作可能に結合されている。コンピュータプログラム命令及びデータを具現化するのに適した情報担体には、すべての形の揮発性ストレージエリアが含まれる。このすべての形の揮発性ストレージエリアには、例として、たとえばEPROM、EEPROM、及びフラッシュストレージエリアデバイスといった半導体ストレージエリアデバイス、内部ハードディスク又は着脱可能ディスクといった磁気ディスク、光磁気ディスク、並びにCD-ROMディスク及びDVD-ROMディスクが含まれる。

10

【0046】

プロセス及びサイトウェアオブジェクトの全部又は一部は、たとえばデータサーバといったバックエンドコンポーネントを含むコンピューティングシステム、又は、たとえばアプリケーションサーバといったミドルウェアコンポーネントを含むコンピューティングシステム、又は、たとえばグラフィカルユーザインターフェースを有するクライアントコンピュータといったフロントエンドコンポーネントを含むコンピューティングシステム、又は、このようなバックエンドコンポーネント、ミドルウェアコンポーネント、若しくはフロントエンドコンポーネントのあらゆる組み合わせを含むコンピューティングシステムで実施することができる。このシステムのコンポーネントは、たとえば通信ネットワークといったあらゆる形又は媒体のデジタルデータ通信によって相互接続することができる。通信ネットワークの例には、LAN、及び、たとえばインターネットといったWANが含まれる。

20

【0047】

プロセス及びサイトウェアオブジェクトに関連する方法ステップは、並べ替えることができ、且つ/又は、1つ若しくは2つ以上のこのようなステップは、本明細書で説明した結果と同じ結果又は類似の結果を達成するように省略することができる。

【0048】

本明細書で説明したプロセスは、オブジェクト指向型プログラミングではなく、手続き型言語の状況で使用することができる。たとえば、サイトウェアオブジェクトX、Yは、以下のように定義することができる。

30

【0049】

```
X = Instrument.ReadMeasurement(PinA)
Y = X.Add(0.1)
```

C等の手続き型言語では、X、Yに対応するものは、以下のように、データ構造体(以下のX、Y)及び外部関数(Add())を使用して実施することができる。

【0050】

```
X = ReadMeasurement(InstrumentID, PinA)
Y = Add(X, 0.1)
```

40

1つの相違は、手続き型言語では、X及びYがデータしか含まず、何らかの処理を実行するAdd()等の外部関数が存在するという点である。

【0051】

本明細書で説明したものと異なる実施形態の要素を組み合わせ、具体的に上述していない他の実施形態を形成することができる。本明細書で具体的に説明していない他の実施形態も、特許請求の範囲に含まれる。

【図面の簡単な説明】

【0052】

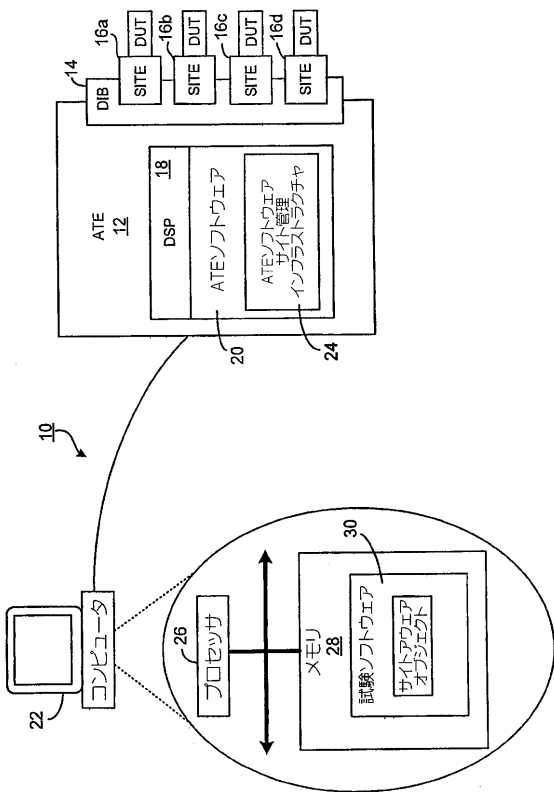
【図1】ATEシステムのブロック図である。

【図2】サイトウェアオブジェクトの図表である。

50

【図3】試験中にサイトウェアオブジェクトを使用するためのプロセスのフローチャートである。

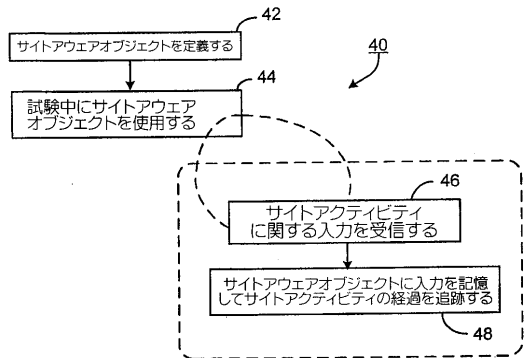
【図1】



【図2】

サイトウェアオブジェクト 32		38 実行可能 コード
34	36	
サイト16Aを識別するデータ	サイト16AのDUTに関連するデータ	
サイト16Bを識別するデータ	サイト16BのDUTに関連するデータ	
サイト16Cを識別するデータ	サイト16CのDUTに関連するデータ	
サイト16Dを識別するデータ	サイト16DのDUTに関連するデータ	

【図3】



フロントページの続き

(74)代理人 100087424

弁理士 大塚 就彦

(72)発明者 フロトヤク, スティーヴン・ジェイ

アメリカ合衆国マサチューセッツ州02478, ベルモント, ウォーリック・ロード 47

(72)発明者 スティムソン, ランドル・ピー

アメリカ合衆国マサチューセッツ州02116, ボストン, ウェスト・ニュートン・ストリート
221, アpartment 2

(72)発明者 ソーントン, ダニエル・ピー

アメリカ合衆国マサチューセッツ州02052, メドフィールド, ステイジコウチ・ロード 9

審査官 菅藤 政明

(56)参考文献 国際公開第2004/072669(WO, A1)

特開平11-190767(JP, A)

(58)調査した分野(Int.Cl., DB名)

G01R 31/28