

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4002335号

(P4002335)

(45) 発行日 平成19年10月31日(2007.10.31)

(24) 登録日 平成19年8月24日(2007.8.24)

(51) Int. Cl.	F I
HO 4 L 12/46 (2006.01)	HO 4 L 12/46 V
GO 6 F 13/00 (2006.01)	GO 6 F 13/00 3 5 5
HO 4 L 12/66 (2006.01)	HO 4 L 12/66

請求項の数 5 (全 20 頁)

(21) 出願番号	特願平9-360774	(73) 特許権者	591030868
(22) 出願日	平成9年12月26日(1997.12.26)		コンパック・コンピューター・コーポレーション
(65) 公開番号	特開平10-224382		COMPAG COMPUTER CORPORATION
(43) 公開日	平成10年8月21日(1998.8.21)		アメリカ合衆国テキサス州77070, ヒューストン, ステイト・ハイウェイ 249, 20555
審査請求日	平成16年12月24日(2004.12.24)		20555 State Highway 249, Houston, Texas
(31) 優先権主張番号	775021		77070, United States of America
(32) 優先日	平成8年12月27日(1996.12.27)	(74) 代理人	100089705
(33) 優先権主張国	米国(US)		弁理士 社本 一夫

最終頁に続く

(54) 【発明の名称】 ネットワーク・デバイスの仮想LANを再構築する方法及び装置

(57) 【特許請求の範囲】

【請求項1】

多ポート・ブリッジング・デバイスの仮想ローカル・エリア・ネットワーク(LAN)を構築する方法において、

予め定義されている各仮想LANに関して、ポートをグループ化することにより、複数の仮想LANを予め定義するステップと、

スパンニング・ツリー手順を実行し、各仮想LANに対するルート識別子およびルート・ポートを決定するステップと、

各仮想LANの前記ルート識別子を、少なくとも1つの他の仮想LANのルート識別子と比較するステップと、

いずれか2つの仮想LANのルート識別子が等しい場合、該2つの仮想LANの前記ルート・ポートを比較するステップと、

いずれか2つの仮想LANに関して、前記ルート識別子が同一であり、前記ルート・ポートが異なる場合、前記2つの仮想LANを併合し、新しい仮想LANを形成する併合ステップと

からなることを特徴とする方法。

【請求項2】

請求項1記載の方法において、前記併合ステップは、前記2つの仮想LANの前記ポートの全てを結合することにより、前記新しい仮想LANを形成するステップを含むことを特徴とする方法。

【請求項 3】

請求項 1 記載の方法において、該方法は更に、前記併合ステップの後に、前記新しい仮想 LAN に対してスパニング・ツリー手順を実行するステップを含むことを特徴とする方法。

【請求項 4】

請求項 1 記載の方法において、該方法は更に、

いずれか 2 つの仮想 LAN に関して、前記ルート・ポートの前記ルート識別子および前記ルート・ポートが同一である場合、該 2 つの仮想 LAN のいずれかの共有ポートが、前記 2 つの仮想 LAN の一方に対してブロックされ、他方に対してブロックされているか否かについて判定を行うステップと、

前記 2 つの仮想 LAN の共有ポートが、前記 2 つの仮想 LAN の一方に対してブロックされ、他方に対してはブロックされていない場合、前記 2 つの仮想 LAN を併合し、他の新しい仮想 LAN を形成するステップとを含むことを特徴とする方法。

【請求項 5】

ネットワーク・デバイスにおいて、複数のポートと、

前記複数のポートの複数の仮想 LAN を定義する仮想 LAN 定義回路と、

前記仮想 LAN 定義回路に結合され、前記複数の仮想 LAN の各々に対して、ルート識別子およびルート・ポートの決定を含むスパニング・ツリーの定義を決定するスパニング・ツリー回路と、

前記スパニング・ツリー回路および前記仮想 LAN 定義回路に結合され、前記複数の仮想 LAN の各々の前記スパニング・ツリー定義を、前記複数の仮想 LAN の少なくとも他の 1 つと比較し、前記仮想 LAN のいずれか 2 つの双方の前記ルート識別子が同一であり、前記 2 つの仮想 LAN 双方の前記ルート・ポートが同一でない場合、前記 2 つの仮想 LAN を併合する処理回路と

からなることを特徴とするネットワーク・デバイス。

【発明の詳細な説明】**【0001】****【発明の属する技術分野】**

本発明は、一般的に、ネットワーク構築デバイス(networking device)に関し、更に特定すれば、仮想 LAN を動的に再構築し、効率的なネットワーク動作のための機能的スパニング・ツリー(functional spanning tree)を得る方法および装置に関するものである。

【0002】**【従来の技術】**

2 台以上のコンピュータ間で、ファイルや資源を共有したり、その他の場合には通信を可能にするネットワークやネットワーク・システムには多くの異なる種類がある。「ネットワーク・デバイス」という用語は、総称的に、ネットワーク・インターフェース・カード(NIC: network interface card)を通じてネットワークにリンクされているコンピュータ、またはリピータ(中継器)即ちハブ、ブリッジ、ネットワーク・スイッチ(交換機)、ルータおよびブルータ(brouter)等の、ネットワークにおいて特殊な機能を実行する装置のことを意味する。ネットワークは、メッセージ容量、ノードが分散される範囲、ノードまたはコンピュータ・タイプ、ノード関係、接続形態(トポロジ)即ち論理および/または物理レイアウト、ケーブル・タイプおよびデータ・パケット・フォーマットに基づくアーキテクチャまたは構造、アクセスの可能性等のような、種々の構造および機能に基づいて分類することができる。例えば、ネットワークの範囲は、1 つの建物のオフィスまたはフロア内のローカル・エリア・ネットワーク(LAN: local area network)、大学のキャンパス全体にわたるワイド・エリア・ネットワーク(WAN: wide area network)、あるいは国境を超えて広がる都市または州および地球領域ネットワーク(GAN: global area network)のように、ノードが分散される距離のことを言う。

【0003】

ネットワークは、1つ以上のリピータ、ブリッジ、交換機または同様のタイプのデバイスを用いることによって拡張を図ることができる。リピータとは、電気信号の再発生、タイミングの再生および増幅を行うことにより、一方のネットワーク・セグメントから他方のネットワーク・セグメントに全てのパケットを移動させる装置のことである。ブリッジとは、OSI (Open Systems Interconnection) 基準モデルのデータ・リンク・レイヤにおいて動作するデバイスであり、一方のネットワークから他方のネットワークにパケットを受け渡し、パケットのフィルタ処理を行うことによって各ネットワーク・セグメント上の不要なパケット伝搬量を減らすことにより、効率を高めるデバイスである。交換機すなわちネットワーク・スイッチは、多ポート・ブリッジと機能的には同様のネットワーク・デバイスであるが、複数のポートを含み、いくつかの同様のネットワークに結合され、これらのネットワーク間でネットワーク・トラフィックを指揮しようというものである。リピータおよびネットワーク・スイッチは、別の組のポートも含み、1つ以上のアップリンク・ポートのように、高速化したネットワーク・デバイスに結合することも可能である。

10

【0004】

ネットワークの拡張の結果として、アドレス衝突(address conflict)の問題だけでなく、ブロードキャスト (同報通信) ストーム(storm)のような、望ましくないネットワーク・パケットの二重化(duplication)や不所望の送信の原因となるループがしばしば発生する。標準的なスパニング・ツリー手順が、ブリッジ、ルータおよびネットワーク・スイッチのような、ネットワーク・ブリッジング・デバイスに定義され、ネットワークのブリッジング・デバイスが、ループのない即ち「スパニング」・ツリーを形成するあらゆる接続形態のサブセットを動的に発見できるようになっている。American National Standards InstituteおよびInstitute of Electrical and Electronics Engineers, Incによるスパニング・ツリー手順は、ANSI/IEEE Std. 802.1Dとして知られている仕様書において公表されている。スパニング・ツリー手順を実行すると、ネットワーク・システム内のいずれか2つのデバイス間にネットワーク経路が形成される。この経路は、ネットワーク・システムの変更に応答して動的に更新される。各ブリッジング・デバイスは、コンフィギュレーション・メッセージを送信する。このメッセージは、コンフィギュレーション・ブリッジ・プロトコル・データ・ユニット (BPDU: bridge protocol data units) とも呼ばれており、ネットワーク内の他のブリッジング・デバイスが、スパニング・ツリーを決定する際に用いる。スパニング・ツリーを決定するために用いられるコンフィギュレーション・メッセージは、48ビットのメディア・アクセス制御 (MAC: media-access control) アドレスを基本とし、これは、業界標準によれば、一意的であることが保証されている。最も低いMACアドレスを有するブリッジング・デバイスは、スパニング・ツリーのルートとして選択され、他のブリッジング・デバイスは、ルート・デバイスのコスト(cost)、即ち、ルート・デバイスからの距離を判定する。

20

30

【0005】

【発明が解決しようとする課題】

多ポート・ブリッジ、ネットワーク・スイッチ、ルータ等のようなブリッジング・デバイスの中には、ユーザに1つ以上の仮想LAN (VLAN) を定義させる機能を含むものがある。別個のVLANは、別個のアドレス空間を各VLANと関連付けることができる。ユーザまたはネットワーク管理者は、VLANの定義の中に1つ以上のポートを一緒に集合化することによって、1つ以上のVLANを定義する。この場合、ブリッジング・デバイスは、効果的に各VLANを他のポートから分離する。1つ以上のポートを2つ以上のVLAN間で共有することができ、共有ポートを有するVLAN間でトラフィックを伝送する。かかる共有ポートは許可されているものの、ループ化問題、アドレスの衝突、および/またはブロードキャスト・ストームの原因となる。また、ユーザは外部ループを形成する外部ハードウェアを介して、意図的にまたは不用意に、2つの互いに排他的なVLANの2つ以上のポートを接続してしまう場合がある。かかるループは、必ずしも標準のスパニング・ツリー手順によって処理されず、その結果、望ましくないパケットの二重化お

40

50

よび送信、アドレスの衝突、またはブロードキャスト・ストームを発生する可能性がある。

【0006】

L V A N定義およびハードウェア接続のそれぞれにおいて、放置しておけばネットワークの問題を発生し、スパニング・ツリーの目的を損なう恐れがある、問題のリンクおよびループを検出することが必要とされている。更に、ネットワークの構成を変更し、トラブルのない機能的なネットワーク・システムを得ることも必要とされている。

【0007】

【課題を解決するための手段】

本発明による、多ポート・ブリッジング・デバイスのV L A Nをコンフィギュレーションする（構築する）方法は、複数のV L A Nを予め定義するステップと、スパニング・ツリー手順を実行し各V L A Nに対してルート識別子とルート・ポートとを決定するステップと、各V L A Nのルート識別子を少なくとも1つの他のV L A Nと比較するステップと、いずれか2つのV L A Nのルート識別子が等しい場合、これら2つのV L A Nのルート・ポートを比較するステップと、いずれか2つのV L A Nに対して、ルート識別子が同一であり、ルート・ポートが異なる場合、これら2つのV L A Nを併合して新しいV L A Nを形成するステップとを含む。好ましくは、各V L A Nをこのようにして他の各V L A Nと比較する。V L A Nを併合する際、双方のV L A Nのポート全てを結合し、新しいV L A Nを定義する。次に、新しいV L A Nについてスパニング・ツリー手順を実行し、次いで、同じように、新しいV L A Nを他のV L A Nと比較する。また予め定義されたV L A Nの内、互いに等価なあらゆるV L A Nを併合し、あらゆるサブセットV L A Nを、各サブセットV L A Nのポート全てを含む上位のV L A Nと併合する。ルート識別子およびルート・ポートが等しい場合、本発明による方法は、更に、2つのV L A Nのいずれかの共有ポートが、一方に対してブロックされ、他方に対してブロックされていないか否かについて判定を行う。そうである場合、これら2つのV L A Nを併合して、新しいV L A Nを形成する。

【0008】

本発明によるネットワーク・デバイスは、複数のポートと、1つ以上のポートを集合化することにより、複数のV L A Nを定義するV L A N定義回路と、V L A Nの各々に対して、ルート識別子およびルート・ポートの決定を含む、スパニング・ツリー定義を決定するスパニング・ツリー回路と、V L A Nのスパニング・ツリー定義を比較し、いずれか2つのV L A Nのルート識別子が同一であり、これら2つのV L A Nのルート・ポートが同一でない場合、これら2つのV L A Nを併合する処理回路とを含む。更に、処理回路は、2つのV L A N双方のルート識別子およびルート・ポートが同一であり、共有ポートが一方に対してブロッキングであり、他方に対してブロッキングでない場合、これら2つのV L A Nを併合する。処理回路はV L A Nを併合する際、併合するV L A Nのポート全てを含む新しいV L A Nを決定する。スパニング・ツリー回路、処理回路およびV L A N定義回路を含むネットワークの回路は、ネットワーク機能を実行するために必要に応じておよび適切に相互結合された、プロセッサ、メモリおよびポート回路から成る。

【0009】

ネットワーク・デバイスの仮想L A Nを動的に再構成する方法および装置は、本発明がなければ所与のV L A N定義および／または特定のネットワーク・コンフィギュレーションによって発生するであろう、データ・パケットのループ化、アドレスの衝突およびブロードキャスト・ストームのようなネットワーク問題を防止することができる。このようなV L A N併合は、スパニング・ツリー手順によって対処しようとしても、V L A N定義またはハードウェア・コンフィギュレーションによって再度引き起こされる、ネットワークの問題を防止するものである。標準的なスパニング・ツリー手順では解決できない、いずれか2つのV L A Nに発生する衝突は、かかるV L A Nを1つに併合し、新しいV L A Nにスパニング・ツリーを再度計算することによって解決する。かかる併合は、潜在的な問題を根絶し、新しいネットワーク問題を引き起こすことなく、有効なネットワークを得るこ

10

20

30

40

50

とができる。管理情報ベース（MIB）を拡張し、元のVLAN定義およびあらゆるVLAN併合の結果を反映することも可能である。ユーザまたはネットワーク管理者は、必要に応じて、VLANまたはハードウェアの再構成を行うことができる。

【0010】

【発明の実施の形態】

図1を参照すると、本発明にしたがって実施された多ポート・ネットワーク・デバイス102を含む、ネットワーク・システム100のブロック図が示されている。ネットワーク・デバイス102は、P1, P2, P3, . . . Pnで示す、複数のポート104を有する。ネットワーク・デバイス102は、必要な数だけポート104を有することができ、各ポート104は、イーサネット（Ethernet）等のようないずれかのタイプのネットワーク・プロトコルにしたがって動作する。また、ポート104は、あるプロトコルにしたがって動作するいくつかのポート、および異なるプロトコルにしたがって動作する他のポートを含んでもよい。例えば、ポート104のいくつかは、Ethernet 10Base-Tにしたがって、毎秒10メガビット（Mbps）で動作し、残りのポートはEthernet 100Base-TXにしたがって、100Mbpsで動作するものとしてもよい。ポート104は、ツイスト・ペア（捩り線対）・ケーブル、光ファイバ・ケーブル、同軸ケーブル等のようなネットワーク・セグメント、あるいは当業者に知られている無線接続を通じて、ネットワーク・デバイス102を複数の他のネットワーク・デバイスに接続可能とする。他のネットワーク・デバイスには、ブリッジ、ネットワーク・スイッチ、リピータ、ルータ、ブルータ等の、任意タイプのデータ端末機器（DTE：Data Terminal Equipment）デバイスが含まれる。データ端末機器デバイスは、ネットワーク・インターフェース・カード（NIC：network interface card）等を有するコンピュータ・システムを含むあらゆるデータの送信（発信）元または宛先である。

【0011】

ネットワーク・デバイス102は、多ポート・ブリッジまたはネットワーク・スイッチであることが好ましく、処理ロジック即ちプロセッサ106、ポート回路108、およびメモリ110を含む。メモリ110は、リード・オンリ・メモリ（ROM）、ランダム・アクセス・メモリ（RAM）等、不揮発性メモリおよびプログラム可能メモリのあらゆる組み合わせを含む。プロセッサ106、ポート回路108、およびメモリ110は、いずれの望ましい態様においても、実施しかつ結合することができる。例えば、メモリ110は、中央メモリとし、必要に応じてプロセッサ106およびポート回路108の双方に結合してもよく、または、メモリ110は、プロセッサ106やポート回路108、そして個々のポートP1～Pnのいずれか1つ以上のための別個のメモリ・モジュールのように、ネットワーク・デバイス102内で分散される。

【0012】

概して言えば、プロセッサ106は、ネットワークネットワーク・スイッチ102の初期化およびコンフィギュレーションを含む処理機能および手順を実行する。プロセッサ106は、その種々の機能を実行するために、メモリ110に記憶されているルーチンを実行する。例えば、プロセッサ106は、第4. 7. 1章による初期化手順を含むANSI/IEEE Std. 802.1D規格に準拠するスパニング・ツリー機能、および第4. 8. 1章によるスパニング・ツリー構成を実行する。また、プロセッサ106は、以下で更に詳しく説明する、本発明による仮想LAN（VLAN）定義の動的再コンフィギュレーションを含む、オンゴーイング（出立）管理機能を処理の間に実行する。ポート回路108は、ポート104の各々に結合され、ポート104間のパケット・データの流れを制御すると共に、プロセッサ106が必要に応じてポート104の動作を制御できるようにする。

【0013】

メモリ110は、必要に応じてポート104のあらゆる集合化すなわちグループ化を定義する、プログラム可能なVLAN定義部112を含む。プロセッサ106は、ユーザまたはシステム管理者がメモリ110をプログラムするのを補佐するルーチンを実行する。図1に示すように、ポートP1, P2, P3を含むVLAN V1 150およびポートP

10

20

30

40

50

4, P 5, P 6を含むV L A N V 2 1 5 2を含む、少なくとも2つの別個のV L A Nが定義される。したがって、V L A N V 1及びV 2は、ポート1 0 4の互いに排他的な集合を含む。好適実施例では、V L A N定義部1 1 2は、各ポート毎に1つのV L A Nポート・マスクを含み、各ポート・マスクは、ネットワーク・デバイス1 0 2の各ポート毎に1ビットを含む。ポート・マスク内の1つのビットを論理高レベル（論理1のような）にセットすると、V L A N内の各ポートを含むことになり、論理低レベル（論理0のような）にクリアすると、当該ポートをマスクすることになる。このように、各ポートのポート・マスクは、同一V L A N群に属する他の各ポート1 0 4を識別することにより、当該ポートに対するV L A Nを定義する。勿論、各ポート1 0 4は、それ自体のV L A Nとして定義することもでき、あるいは、ポート1 0 4の全てを1つのグローバルV L A Nに含

10

【0 0 1 4】

ネットワーク・セグメント1 1 2は、D T Eデバイス1 1 4をポートP 1に結合する。ネットワーク・セグメント1 1 6は、ネットワーク・デバイス1 1 8をポートP 2に結合する。ネットワーク・デバイス1 1 8は、多数のポートを有するリピータ等であり、1つ以上のD T Eデバイス1 2 0, 1 2 2を別個のネットワーク・セグメントを通じて結合するものであることが好ましい。リピータ1 1 8の1つのポートは、ネットワーク・セグメント1 2 4を通じて、他のネットワーク・デバイス1 2 6の1つのポートに結合されている。ネットワーク・デバイス1 2 6はブリッジであり、ネットワーク・セグメント1 2 8を通じて他のリピータ1 3 0に結合されている他のポートを含む。リピータ1 3 0は、別個のネットワーク・セグメントを通じて、1つ以上のD T Eデバイス1 3 2, 1 3 4を結合し、ネットワーク・セグメント1 3 6を通じて、他のブリッジ1 3 8の1つのポートに結合されている。ブリッジ1 3 8は、ネットワーク・セグメント1 4 0を通じて、ネットワーク・デバイス1 0 2のポートP 6に結合されている、他のポートを含む。D T Eデバイス1 4 4は、ネットワーク・セグメント1 4 2を通じて、ポートP 4に結合されている。図1に示す構成は、ネットワーク・デバイス1 0 2のポートP 2, P 6間のハードウェア・ループを示しており、このハードウェア・ループはデバイス1 1 8, 1 2 6, 1 3 0, 1 3 8、およびネットワーク・セグメント1 1 6, 1 2 4, 1 2 8, 1 3 6, 1 4 0を含む。

20

【0 0 1 5】

図2は、異なるV L A N定義を有するネットワーク・デバイス1 0 2を示す簡略ブロック図である。即ち、第1 V L A N V 1 2 0 2はポートP 1, P 2, P 3を含み、第2 V L A N V 2 2 0 4はポートP 1～P 6を含む。このように、V 1はV 2のサブセットである。共有するポートP 1, P 2, P 3のいずれかが、あるV L A Nに対してはブロッキング(blocking)、他のV L A Nに対しては非ブロッキング(non-blocking)（リスニング(listening)／学習(learning)／フォワード即ち送出(forwarding)）としてコンフィギュレーションされると、サブセットのV L A N定義は問題を発生する場合がある。各ポート1 0 4および各V L A N定義について、ネットワーク・デバイス1 0 2は、ユーザ定義パラメータおよびスパニング・ツリー処理の結果に基づいて、ポート状態を判定する。ポートがブロックされている場合、ネットワーク・デバイス1 0 2は通常、当該ポートにおいて受信されるあらゆるデータ・パケットを無視し、このポートを通じたデータ・パケットの送出を全く行わない。コンフィギュレーションの間にポートがリスニング状態(listening status)モードにセットされると、コンフィギュレーション・パケットの送出および受信が行われる。このパケットは、他の呼び方では、ブリッジ・プロトコル・データ・ユニット(B P D U: bridge protocol data units)としても知られている。ポートが学習状態モードにセットされると、当該ポートに結合されているデバイスのネットワーク・アドレスを検査し格納する、学習ブリッジ機能を実行する。ポートが送出モード(forwarding mode)にセットされると、データ・パケットの送出および受信が行われる。

30

40

【0 0 1 6】

図3は、別のV L A N定義を有するネットワーク・デバイス1 0 2の他の簡略ブロック図

50

であり、VLAN V1 302はポートP1～P4を含み、VLAN V2 304はポートP4～P7を含む。このように、ポートP4は、VLAN V1, V2間で共有される。ポートP2, P5は、外部ネットワーク306に結合されている。外部ネットワーク306は、簡略化ブリッジ識別子(ID:図ではBID)が5であるブリッジ・デバイス308を含む。通常、各ブリッジ・デバイスには少なくとも1つのブリッジIDが割り当てられており、このIDは産業用の幅である48ビットの一意のアドレスである。ネットワークネットワーク・スイッチ102は、ブリッジID12をVLAN V1に割り当て、ブリッジID13をV2に割り当てる。ブリッジ・デバイス308のブリッジID5が、スパンニング・ツリー手順による最低の値である場合、ブリッジ・デバイス308は、ネットワーク・デバイス102を含むネットワーク306に対するROOTブリッジとなる。ポートP2はVLAN V1に対するROOTポートとなり、ポートP5はネットワーク・デバイス102のVLAN V2に対するROOTポートとなる。V1およびV2は共有ポートおよび異なるROOTポートを有するため、VLAN V1, V2を横切るループが存在するので、二重パケット、アドレス衝突、ブロードキャスト・ストーム等の、潜在的な問題の原因となる。

10

【0017】

図4は、図3に示したVLAN定義と同じ定義のネットワーク・デバイス102の他の簡略ブロック図を示し、V1はポートP1～P4を含み、V2はポートP4～P7を含む。しかしながら、ネットワーク・システム306は共有ポートP4に結合されている。したがって、ポートP4はVLAN V1, V2双方に対するROOTポートとなる。

20

【0018】

図5は、他のVLAN定義を有するネットワーク・デバイス102の他の簡略ブロック図であり、VLAN V1 502はポートP1～P5を含み、VLAN V2 504はポートP3～P7を含む。この場合、ポートP3～P5は、VLAN V1, V2間で共有される。初期化および構成の後、P3は、V1, V2双方に対するROOTポートに決定される。ポートP4は、その中の記号「X」で示されるように、V1, V2双方に対してブロッキングとして構成される。しかしながら、ポートP5は、V1に対してはブロッキング、V2に対しては送出にセットされている。2つ以上のポートを数個のVLAN間で共有することは通常許されているが、共有ポートが一方のVLANに対してはブロッキングであり、共有VLANの他のいずれかに対しては非ブロッキングである場合、ネットワークの問題が発生する。

30

【0019】

図6は、更に他のVLAN定義を有するネットワーク・デバイス102の他の簡略ブロック図であり、VLAN V1 602はポートP2～P4を含み、VLAN V2 604はポートP3～P7を含み、VLAN V3 606はポートP4, P8, P9を含む。このように、ポートP3は、VLAN V1, V2間で共有され、ポートP4は、3つのVLAN V1～V3全ての間で共有される。通常、かかる多数のポート共有は許されているが、標準的なスパンニング・ツリー手順によるコンフィギュレーションについて既に説明したように、問題を生ずる可能性がある。例えば、ポートP3がV1に対してはブロッキングであり、V2に対しては送出である場合、アドレス衝突および／または二重パケットの問題が発生する。

40

【0020】

図7は、上述のあらゆるVLAN構成の問題を解消するために、ネットワーク・デバイス102が実行する再コンフィギュレーション手順を示すフロー・チャートである。この手順は、ユーザ定義のVLAN定義を変更するもので、プロセッサ106、ポート回路108、またはその組み合わせによって実行される。最初のステップ702において、VLAN定義部112に定義されているような、VLANの割り当てがメモリ110から読み出される。次のステップ704において、あらゆる等価なVLANが共に併合される。等価なVLANとは、同じポート割り当てを有するもののことである。また、あらゆるサブセットVLANも、それらの対応するサブセットVLANと一緒に共に併合される。2つの

50

VLAN V_i , V_j は、以下の式 (1) および (2) にしたがって併合される。

【0021】

【数1】

$$V_i'(fwd_mask) := V_i(fwd_mask) OR V_j(fwd_mask) \quad (1)$$

【数2】

$$V_j'(fwd_mask) := V_j(fwd_mask) OR V_i(fwd_mask) \quad (2)$$

ここで、fwd_maskは、指示されたVLANのポート・マスクを示し、VLAN V_i' , V_j' は「新しい」VLAN定義であり、「:=」演算子は「等しくセットする」を意味し、「OR」演算子はビット毎の論理OR演算である。式 (1) および (2) に示すように、新しいVLAN定義は、新しいVLAN双方に対して同一であり、元のVLAN定義のポートのスーパーセット(superset)を含む。一旦併合されると、VLAN割り当ての内の一方のみがアクティブとなり、スパニング・ツリーを決定する目的のために用いられ、他方のVLAN定義は、管理目的のためのネットワーク・デバイス102の管理情報ベース(MIB: management information base)に保持されるが、スパニング・ツリーを決定する目的のためには用いられない。影響が及ぶポート各々のポート・マスク割り当ては、VLAN定義112において変更され、影響が及ぶと共に中に含まれる各ポートに対しては、新しいVLANポート・マスク定義が用いられる。また、元のVLAN定義もネットワーク管理の目的のために保持されることに留意する必要がある。

【0022】

図2を例として用いると、V1はポート・マスクfwd_mask=000007h(ここで、「h」は16進表記であることを示す)を有し、V2はポート・マスクfwd_mask=00003Fhを有する。併合処理の後、新しいVLAN V_1' , V_2' の双方に対するポート・マスクは、fwd_mask=00003Fhとなる。新しいポート・マスク定義 V_1' がアクティブであり、スパニング・ツリーの目的に使用され、 V_2' に対する「新しい」ポート・マスク定義は管理の目的のために保持され、併合VLANと印されるが、アクティブではないので、スパニング・ツリーを決定するためには用いられない。ポートP1～P6の各々に対するポート・マスクも、新しいfwd_mask=00003Fhに等しくセットされる。

【0023】

次のステップ706において、新しいネットワーク・デバイス102は、アクティブな各VLANに一意のブリッジIDを割り当てる。次のステップ708において、ANSI/IEEE Std. 802.1Dの第4.8.1章に定義されているように、スパニング・ツリー初期化手順にしたがって、アクティブな各VLANの各ポートを初期化する。次のステップ710において、ANSI/IEEE Std. 802.1Dの第4.7.1章に定義されている、スパニング・ツリー・コンフィギュレーション

(構成)手順を、アクティブな各VLANの各ポートについて実行する。標準的なスパニング・ツリー手順に従ってコンフィギュレーション手順を実行した後、各VLANにはルート識別子(ROOT ID)およびROOTポートが割り当てられている。例えば、図3に示すように、VLAN V1にはROOT ID 5が割り当てられ、ポート2がそのROOTポートであり、VLAN V2にはROOT ID 5が割り当てられ、ポート5がそのROOTポートである。

【0024】

次のステップ712, 714において、アクティブな各VLANを1つ置ききのVLANと比較し、前述のようなあらゆる潜在的なネットワーク問題を確認する。以下で説明するように、アクティブで残っているVLAN全てについて検査しコンフィギュレーションするまで、問題のVLAN対を新しいVLAN定義に併合し、新しいVLANを残りの各VLANと比較する。ステップ714において、新しいアクティブなVLAN対を選択する。次のステップ716において、選択されたVLANのROOT IDを比較する。選択された対のROOT IDが等しくない場合、処理はステップ712に戻り、他のアクティブなVLANの対を選択する。ROOT IDが等しくない場合、これら2つのVLANは共にリンクされず、問題のループを含んでいない。したがって、スパニング・ツリーは

10

20

30

40

50

良好である。しかしながら、選択された対の R O O T I D が等しい場合、処理はステップ 7 1 8 に進み、選択された対の R O O T ポートが等しいか否かについて検査を行う。図 3 に示すように、選択された対の R O O T I D が等しいが、R O O T ポートが等しくない場合、処理はステップ 7 2 2 に進み、これら 2 つの V L A N を併合する。

【 0 0 2 5 】

先に示した式 (1) および (2) は、ステップ 7 2 2 において 2 つの V L A N を併合するために用いられる。例えば、図 3 に示すように、V L A N V 1 はポート・マスク fwd_mask=00000Fh を有し、V L A N V 2 はポート・マスク fwd_mas=000078h を有する。新しい V L A N V 1 ' , V 2 ' の各々は、併合処理の結果、新しいポート・マスク fwd_mask=00007Fh を有する。上述のように、V L A N V 1 ' が継続するスパニング・ツリー処理に選択され、V L A N V 2 ' は併合されたものとマークされて使用されない。ポート P 1 ~ P 7 の各々に対するポート・マスク定義は変更され、00007Fh に等しくセットされる。その結果得られた V L A N V 1 ' 8 0 2 を図 8 に示す。次のステップ 7 2 4 において、ブリッジ I D を新しい V L A N V 1 ' に割り当てる。新たなブリッジ I D は、全く新しいブリッジ I D でも、元の V L A N V 1 , V 2 に割り当てられたブリッジ I D の一方でもよい。図 8 に示すように、新しい V L A N V 1 ' には、元の V L A N V 1 のブリッジ I D = 1 2 が割り当てられる。次のステップ 7 2 6 において、ステップ 7 0 8 についての説明と同様に、新しい V L A N のポート全てを初期化し、ステップ 7 1 0 についての説明と同様に、新しい V L A N に対するスパニング・ツリー (樹) のコンフィギュレーション (構成) を実行する。

【 0 0 2 6 】

図 8 に示すように、新しい V L A N V 1 ' に対する新たなコンフィギュレーションの結果、R O O T ポートの 1 つが、新しい V L A N V 1 ' 8 0 2 に対する R O O T ポートとして選択される。何故なら、いずれの所定の V L A N に対しても、選択される R O O T ポートは 1 つのみとされているからである。ステップ 7 2 8 から、処理はステップ 7 1 2 に戻り、他の V L A N 対を比較する。ここで、新しい V L A N も結果的に他の各 V L A N と比較される。ステップ 7 1 8 に戻り、図 4 および図 5 に示したように、R O O T I D および R O O T ポートが等しい場合、処理はステップ 7 2 0 に進む。ステップ 7 2 0 において、ある V L A N についてはブロックされ、他のいずれかの V L A N についてはブロックされていない共有ポートがあるか否かについて検査を行う。図 4 に示すように、かかる共有ポートがない場合、処理はステップ 7 1 2 に戻り、比較のための他の V L A N 対を選択する。しかしながら、図 5 に示す構成のポート P 5 のように、ある共有ポートが、1 つの V L A N に対してはブロックされ、他の V L A N に対してはブロックされていないという状況があった場合、処理はステップ 7 2 2 に進み、これらの V L A N を併合する。例えば、図 5 において、ポート P 5 は V L A N V 1 に対してはブロックされているが、V L A N V 2 に対してはフォワード状態となっている。この場合、V L A N V 1 , V 2 が併合され、その結果、図 9 に示すように、新しい V L A N V 1 ' 9 0 2 が得られる。初期化ステップ 7 2 6 およびコンフィギュレーション・ステップ 7 2 8 を実行した後、新しい V L A N に割り当てられたブリッジ I D によっては、R O O T ポートが同一のまま残っている可能性がある。異なる V L A N に対して元々異なるステータスの設定を有していたポート (図 5 ではポート P 5) には、送出、ブロッキング等のような新しい状態の設定が、図 9 に示す新しい V L A N V 1 ' に対して与えられる。

【 0 0 2 7 】

アクティブな V L A N および新しく併合された V L A N の全てについて、ステップ 7 1 2 において比較判定を行った後、処理は元の V L A N 定義については完了したことになる。しかしながら、一実施例では、処理の間に、V L A N 定義を変更する場合もある。V L A N 定義を処理の間に再コンフィギュレーションする場合、定義された新しい V L A N 各々について、そして、V L A N の併合の結果、新しい V L A N が生成されたなら、それらの各々について、図 7 に示す手順を繰り返す。

【 0 0 2 8 】

10

20

30

40

50

ネットワーク・デバイス 102 のブリッジ MIB (RFC 1493) は、必要に応じて、管理目的にも拡張することができる。ブリッジ MIB は、メモリ 110 に格納しておくことが好ましい。MIB 拡張部は、ユーザまたはネットワーク管理者が、併合 VLAN を決定し、VLAN 定義を再コンフィギュレーションしたり、あるいは必要に応じてハードウェアの変更を行うことを可能にするものである。可能な MIB 拡張部の一例を以下に示す。

1. ブリッジ・アドレス、ブリッジ内に存在するポート数等を含む、dot1dBase 群ベース拡張部
2. 仮想ブリッジ全ておよびそれらのスパニング・ツリーに対処する dot1dStp 群拡張部
3. 各仮想ブリッジに対する dot1dTp
4. VLAN に基づいてスタティック・アドレスをコンフィギュレーションしなければならない場合の dot1dStatic 群拡張部。

10

【0029】

VLAN 管理 MIB の一例は以下の通りである。

【表 1】

-- VLAN Config Table
 -- This table is used to statically configure N VLANs at
 -- initialization. The vlanPortConfigTable is then used
 -- to add/delete ports to this VLAN.

vlanConfigTable	OBJECT-TYPE	
SYNTAX	SEQUENCE OF vlanConfigEntry	10
ACCESS	not-accessible	
STATUS	mandatory	
DESCRIPTION		
	"A table containing a list of VLANs configured for this device."	
	::= { vlanConfigConfig 2 }	

20

vlanConfigEntry	OBJECT-TYPE	
SYNTAX	VlanConfigEntry	
ACCESS	not-accessible	
STATUS	mandatory	
DESCRIPTION		
	"A list of VLANs, their Ids etc."	
INDEX { vlanConfigIndex }		30
::= { vlanConfigTable 1 }		

VlanConfigEntry ::= SEQUENCE {		
vlanIfIndex	INTEGER,	
vlanConfigBridgeAddress	MacAddress,	
vlanConfigNumofEntries	INTEGER,	
vlanConfigReconfigStatus	INTEGER,	40
vlanConfigReconfigCookie	INTEGER,	

vlanConfigName DisplayString

}

vlanIndex OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-only

STATUS mandatory

10

DESCRIPTION

”The unique identifier for this VLAN configuration entry.”

::= { vlanConfigEntry 1 }

vlanConfigBridgeAddress OBJECT-TYPE

SYNTAX MacAddress

20

ACCESS read-only

STATUS mandatory

DESCRIPTION

” The MAC address that is used by this VLAN. It is taken from the pool of addresses available for the bridge. ”

::= { vlanConfigEntry 2 }

30

vlanConfigNumofEntries OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-only

STATUS mandatory

DESCRIPTION

” The total number of ports configured for this VLAN. Please note that this number includes only number of ports configured by the user. If VLANs are merged, the

40

total number of ports present in a VLAN will be different
 from this number ”
 ::= { vlanConfigEntry 3 }

vlanConfigReconfigStatus OBJECT-TYPE

SYNTAX INTEGER{

userDefined(1),
 reconfigured(2)
 }

10

ACCESS read-only

STATUS mandatory

DESCRIPTION

” This Object indicates if VLAN merging has occurred. If
 the value of this object is reconfigured (2), then the
 vlanConfigReconfigCookie provides a unique means of
 identifying all VLAN entries that have been merged”
 ::= { vlanConfigEntry 4 }

20

vlanConfigReconfigCookie OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-only

STATUS mandatory

DESCRIPTION

” A unique value that identifies all VLAN entries that
 have been merged ”
 ::= { vlanConfigEntry 5 }

30

vlanConfigName OBJECT-TYPE

40

SYNTAX DisplayString

ACCESS read-write

STATUS mandatory

DESCRIPTION

" A textual description field for this VLAN. "

::= { vlanConfigEntry 6 }

vlanPortConfigTable OBJECT-TYPE

10

SYNTAX SEQUENCE OF vlanPortConfigEntry

ACCESS not-accessible

STATUS mandatory

DESCRIPTION

"A table containing a list of ports configured for a VLAN."

::= { vlanConfig 2 }

20

vlanPortConfigEntry OBJECT-TYPE

SYNTAX VlanPortConfigEntry

ACCESS not-accessible

STATUS mandatory

DESCRIPTION

"A list of ports, their Ids etc."

30

INDEX { vlanPortVlanIndex, TlanPortIndex }

::= { vlanConfigTable 1 }

-- VLAN Port Config Table

-- This table is used to add/remove ports from a VLAN

VlanPortConfigEntry ::= SEQUENCE {

40

vlanPortVlanIndex

INTEGER,

```

        vlanPortIndex          INTEGER,
        vlanPortName           DisplayString,
        vlanPortConfigStatus EntryStatus
    }
vlanPortVlanIndex             OBJECT-TYPE
    SYNTAX                     INTEGER
    ACCESS                      read-only
    STATUS                      mandatory
    DESCRIPTION
        "The index of the VLAN. This value is the same as the
        value of vlanIndex in the vlanConfigTable."
    ::= { vlanPortConfigEntry 1 }

vlanPortIndex                 OBJECT-TYPE
    SYNTAX                     INTEGR
    ACCESS                      read-only
    STATUS                      mandatory
    DESCRIPTION
        "The port number of the port that is a member of this
        VLAN. This value should be the same as the if Index value
        for this interface"
    ::= { vlanPortConfigEntry 2 }

vlanPortName                  OBJECT-TYPE
    SYNTAX                     DisplayString
    ACCESS                      read-write
    STATUS                      mandatory
    DESCRIPTION
        " A textual description place-holder for this port. "
    ::= { vlanPortConfigEntry 3 }

```

10

20

30

40

vlanPortConfigStatus OBJECT-TYPE

SYNTAX EntryStatus

ACCESS read-write

STATUS mandatory

DESCRIPTION

"The row control mechanism for this entry. "

10

::= { vlanPortConfigEntry 4 }

-- Reset Reconfiguration

-- Starts reinitialization of All VLANs whose port masks have merged

vlanConfigResetReconfig OBJECT-TYPE

SYNTAX INTEGER{

invalid (1),

reset (2)

}

20

ACCESS read-write

STATUS mandatory

DESCRIPTION

"Setting this value to reset(2) will restart spanning tree process on all merged VLANs. "

30

::= { vlanConfig 3 }

--Traps

vlanConfigReconfig TRAP-TYPE

ENTERPRISE vlanConfig

40

VARIABLES {vlanConfigReconfigCookie,

`vlanIfIndex(first), vlanIfIndex(second)....}`

DESCRIPTION

"A vlanConfigReconfig TRAP is sent to the management station when the bridge reconfigures the port masks after it detects a VLAN misconfiguration"

以上の説明から、ネットワーク・デバイスの仮想LANを動的に再コンフィギュレーションする方法および装置が、特定のVLAN定義および／またはハードウェア構成によって発生する、データ・パケットのルーピング、アドレス衝突および／またはブロードキャスト・ストームのような、ネットワーク問題を防止することが明らかであろう。等価なVLANおよびサブセットVLANは潜在的にネットワーク問題を発生する可能性があるので、これらをまず結合する。次に、初期化およびコンフィギュレーションを含むスパニング・ツリー手順を実行する。初期のスパニング・ツリー手順を実行した後、VLANを比較して、問題のあるコンフィギュレーションを確認する。同一のルート識別子を有する互いに排他的なVLANを共に接続し、外部的に1つのループを形成することによって、これらのVLANを併合する。共有ポートが異なるルート・ポートを有するVLANも併合する。共有ポートおよび同一ルート・ポートを有するVLANは、いずれかの共有ポートが一方のVLANに対してブロッキングであり、他方に対してはそうでない場合、併合する。このようなVLANの併合によって、VLAN定義および／またはハードウェア・コンフィギュレーションの重複のために、スパニング・ツリー手順では対処できないネットワーク問題を防止する。また、VLANの併合の結果、新たな問題を招くことなく、有効なネットワークが得られる。

【0030】

以上、好適実施例との関連において本発明によるシステムおよび方法について説明したが、ここに記載した特定形態に限定されることを意図する訳ではなく、逆に、特許請求の範囲に規定される本発明の技術思想および範囲に合理的に含まれ得る代替物、変更物、均等物等は、本発明に含まれることを意図するものである。

【図面の簡単な説明】

【図1】本発明にしたがって実施した、多ポートネットワーク・デバイスを含むネットワーク・システムを示すブロック図である。

【図2】サブセットVLANを示すVLANの定義と共に、図1のネットワーク・デバイスを示す簡略ブロック図である。

【図3】1つの共有ポートおよび異なるルート・ポートを有する2つのVLANを示すVLANの定義と共に、図1のネットワーク・デバイスを示す簡略ブロック図である。

【図4】1つの共有ポートおよび同一ルート・ポートを有する2つのVLANを示すVLAN定義と共に、図1のネットワーク・デバイスを示す簡略ブロック図である。

【図5】多数の共有ポート、同一ルート・ポート、およびVLANに対して異なるステータスを有する1つの共有ポートを有する2つのVLANを示すVLAN定義と共に、図1のネットワーク・デバイスを示す簡略ブロック図である。

【図6】共有ポートを有する多数のVLANを示す3つのVLAN定義と共に、図1のネットワーク・デバイスを示す簡略ブロック図である。

【図7】本発明にしたがって実施した再コンフィギュレーション手順を示すフローチャートである。

【図8】図3の元のVLAN定義を新たなVLAN定義に併合した、図1のネットワーク・デバイスを示す簡略ブロック図である。

【図9】図5の元のVLAN定義を新たなVLAN定義に併合した、図1のネットワーク・デバイスを示す簡略ブロック図である。

【符号の説明】

10

20

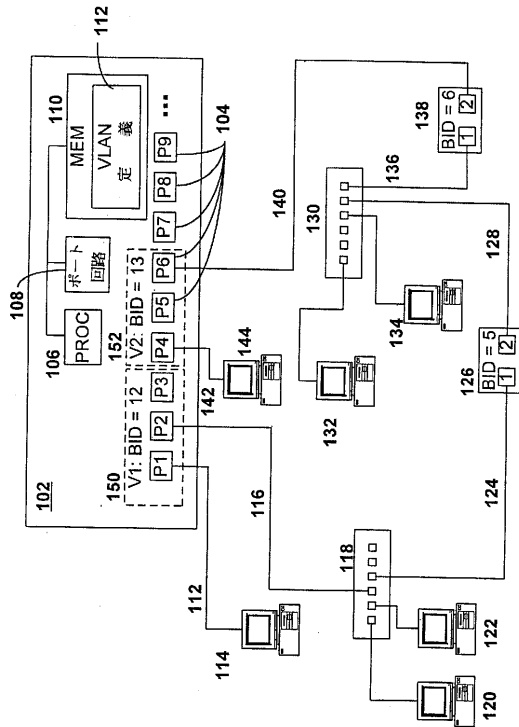
30

40

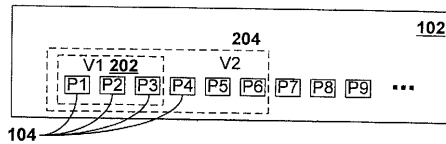
50

1 0 0	ネットワーク・システム	
1 0 2	多ポート・ネットワーク・デバイス	
1 0 4	ポート	
1 0 6	プロセッサ	
1 0 8	ポート回路	
1 1 0	メモリ	
1 1 2	ネットワーク・セグメント	
1 1 4	D T E デバイス	
1 1 6	ネットワーク・セグメ	
1 1 8	ネットワーク・デバイス	10
1 2 0 , 1 2 2	D T E デバイス	
1 2 4	ネットワーク・セグメント	
1 2 6	ネットワーク・デバイス	
1 2 8	ネットワーク・セグメント	
1 3 0	リピータ	
1 3 2 , 1 3 4	D T E デバイス	
1 3 6	ネットワーク・セグメント	
1 3 8	ブリッジ	
1 4 0	ネットワーク・セグメント	
1 4 2	ネットワーク・セグメント	20
1 4 4	D T E デバイス	
2 0 2	第 1 V L A N V 1	
2 0 4	第 2 V L A N V 2	
3 0 2	V L A N V 1	
3 0 4	V L A N V 2	
3 0 6	外部ネットワーク	
3 0 8	ブリッジ・デバイス	
5 0 2	V L A N V 1	
5 0 4	V L A N V 2	
6 0 2	V L A N V 1	30
6 0 4	V L A N V 2	
6 0 6	V L A N V 3	

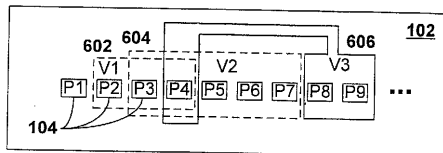
【図 1】



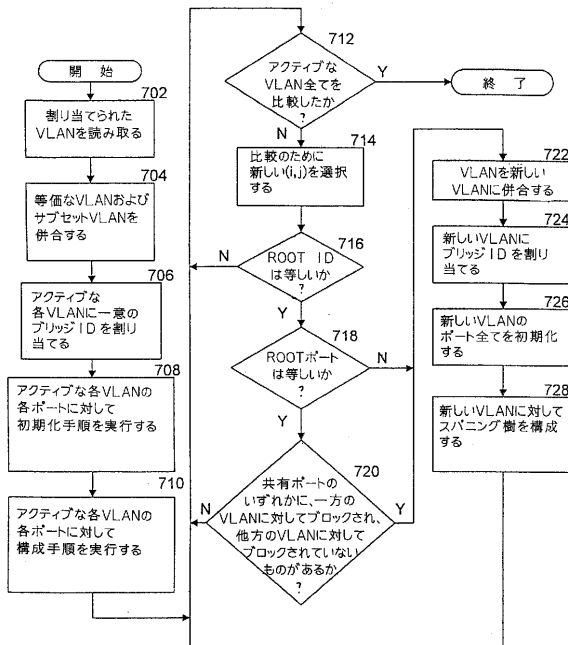
【図 2】



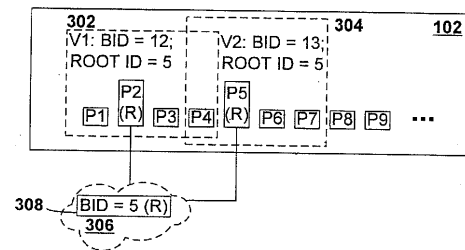
【図 6】



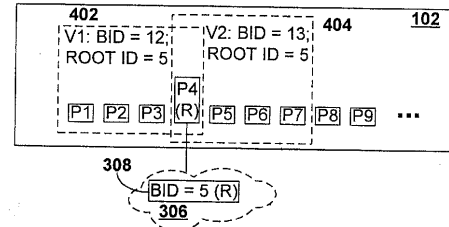
【図 7】



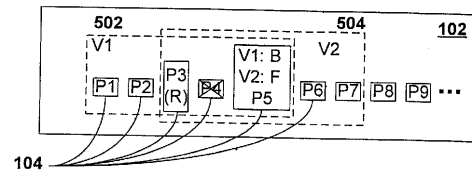
【図 3】



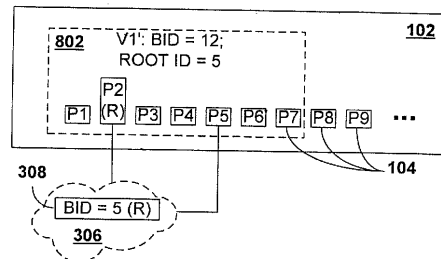
【図 4】



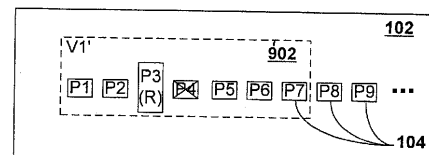
【図 5】



【図 8】



【図 9】



フロントページの続き

(74)代理人 100071124

弁理士 今井 庄亮

(74)代理人 100076691

弁理士 増井 忠弐

(74)代理人 100075236

弁理士 栗田 忠彦

(74)代理人 100075270

弁理士 小林 泰

(74)代理人 100096068

弁理士 大塚 住江

(74)代理人 100107696

弁理士 西山 文俊

(72)発明者 ペラム・マリムズ

アメリカ合衆国テキサス州77339, キングウッド, キングウッド・ドライブ 3700, アパ
ートメント・ナンバー 1508

審査官 矢頭 尚之

(56)参考文献 特開平7-264244 (JP, A)

特開平8-32611 (JP, A)

(58)調査した分野(Int.Cl., DB名)

H04L 12/46

H04L 12/66