

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4026979号

(P4026979)

(45) 発行日 平成19年12月26日(2007.12.26)

(24) 登録日 平成19年10月19日(2007.10.19)

(51) Int. Cl.		F I		
H04L 12/40	(2006.01)	H04L 12/40		Z
G06F 13/38	(2006.01)	G06F 13/38	350	

請求項の数 5 (全 19 頁)

<p>(21) 出願番号 特願平11-67052 (22) 出願日 平成11年3月12日(1999.3.12) (65) 公開番号 特開2000-32010(P2000-32010A) (43) 公開日 平成12年1月28日(2000.1.28) 審査請求日 平成18年3月13日(2006.3.13) (31) 優先権主張番号 特願平10-84709 (32) 優先日 平成10年3月30日(1998.3.30) (33) 優先権主張国 日本国(JP)</p>	<p>(73) 特許権者 000001007 キヤノン株式会社 東京都大田区下丸子3丁目30番2号 (74) 代理人 100090273 弁理士 園分 孝悦 (72) 発明者 小林 崇史 東京都大田区下丸子3丁目30番2号 キ ヤノン株式会社内 (72) 発明者 大西 慎二 東京都大田区下丸子3丁目30番2号 キ ヤノン株式会社内 審査官 岩田 玲彦</p>
--	---

最終頁に続く

(54) 【発明の名称】 データ通信システム

(57) 【特許請求の範囲】

【請求項1】

被送信データを送信するソースと、
前記被送信データを受信するディスティネーションと、
前記ソースと前記ディスティネーションとの間の通信を管理するコントローラとを有し

、
前記ソースは、
前記被送信データ内の各データを、前記ディスティネーションが有する受信バッファの
一部を指定するのに用いられるオフセットアドレスとともに送信する機能と、
バスリセットが発生した場合において、前記被送信データの送信を中断する機能とを有
し、

前記ディスティネーションは、
前記バスリセットが発生した後において、前記バスリセットが発生する前に正常に受信
されたデータに対応する前記オフセットアドレスを前記ソースに通知する機能を有し、

前記ソースはさらに、
前記バスリセットが発生した後において、前記ディスティネーションから通知された前
記オフセットアドレスに従って前記被送信データの送信を再開する機能を有することを特
徴とするデータ通信システム。

【請求項2】

前記コントローラはさらに、前記バスリセットが発生した後において、前記ディスティ

10

20

ネーションに再設定された通信アドレスを前記ソースに通知する機能を有することを特徴とする請求項 1 に記載のデータ通信システム。

【請求項 3】

前記コントローラはさらに、前記バスリセットが発生した後において、前記ソースに再設定された通信アドレスを前記ディスティネーションに通知する機能を有することを特徴とする請求項 1 又は 2 に記載のデータ通信システム。

【請求項 4】

前記被送信データ内の各データは、前記オフセットアドレスとともに、IEEE 1394 - 1995 規格で規定されたアシンクロナス (Asynchronous) 転送方式により送信されることを特徴とする請求項 1 から 3 のいずれかに記載のデータ通信システム。

10

【請求項 5】

前記バスリセットは、IEEE 1394 - 1995 規格で規定された機能であることを特徴とする請求項 1 から 4 のいずれかに記載のデータ通信システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、被送信データを送信するソースと、当該被送信データを受信するディスティネーションと、当該ソースと当該ディスティネーションとの間の通信を管理するコントローラとを有するデータ通信システムに関する。

【0002】

20

【従来の技術】

従来、パーソナルコンピュータ (以下、PC) の周辺機器の中で、最も利用頻度が高かったのはハードディスクやプリンタであった。これらの周辺機器は、専用の入出力インタフェース或いは SCSI (small computer system interface) インタフェース等の汎用性のあるデジタルインタフェースによって PC と接続されていた。

【0003】

一方、近年、デジタルカメラ、デジタルビデオカメラ等の AV (Audio/Visual) 機器も PC の周辺装置の 1 つとして注目を浴びている。これらの AV (Audio/Visual) 機器も専用インタフェースを介して PC と接続されていた。

【0004】

30

従来の専用インタフェースや SCSI インタフェースでは、特に AV 機器の有する静止画像や動画のような大容量のデータを扱う場合において、データ転送レートが低い、パラレル通信のため通信ケーブルが太い、接続できる周辺機器の数と種類が少ない、接続方式に制限がある、リアルタイムなデータ転送が行えない等の多くの問題があった。

【0005】

このような問題点を解決する次世代の高速、高性能デジタルインタフェースの一つとして、IEEE (The Institute of Electrical and Electronics Engineers, Inc.) 1394 - 1995 規格が知られている。

【0006】

IEEE 1394 - 1995 規格に準拠したデジタルインタフェース (以下、1394 インタフェース) には、次のような特徴がある。

40

- (1) データ転送速度が高速である。
- (2) リアルタイムなデータ転送方式 (即ち、Isochronous 転送方式) と Asynchronous 転送方式をサポートしている。
- (3) 自由度の高い接続構成 (トポロジ) を構築できる。
- (4) プラグ・アンド・プレイ機能と活線挿抜機能をサポートしている。

【0007】

【発明が解決しようとする課題】

しかしながら、IEEE 1394 - 1995 規格では、コネクタの物理的、電氣的な構成、最も基本的な 2 つのデータ転送方式等については定義しているが、どのような種類のデ

50

ータを、どのようなデータ・フォーマットで、どのような通信プロトコルに基づいて送受信するのかについては定義していなかった。

【0008】

又、IEEE 1394 - 1995規格のIsochronous転送方式では、送出パケットに対する応答が規定されていないため、各Isochronousパケットが確実に受信されたかは保証されない。したがって、連続性のある複数のデータを確実に転送したい場合、或いは1つのファイルデータを複数のデータに分割して確実に転送したい場合、Isochronous転送方式を使用することはできなかった。

【0009】

又、IEEE 1394 - 1995規格のIsochronous転送方式では、転送帯域に空きがある場合でも、通信の総数が64個に制限される。このため、少ない転送帯域で数多くの通信を行いたい場合、Isochronous転送方式を使用することはできなかった。

10

【0010】

又、IEEE 1394 - 1995規格では、ノードの電源のON/OFF、ノードの接続/切り離し等に応じてバスリセットが生じた場合、データの転送を中断しなければならない。ところが、IEEE 1394 - 1995規格では、バスリセットや伝送時のエラーによってデータ転送が中断した場合、どのような内容のデータを失ったのかについては知ることができなかった。更に、一度中断した転送を復帰するためには、非常に煩雑な通信手順を踏む必要があった。

20

【0011】

ここで、バスリセットとは、新たなトポロジの認識と各ノードに割り当てられるアドレス(ノードID)の設定とを自動的に行う機能である。この機能により、IEEE 1394 - 1995規格では、プラグ・アンド・プレイ機能と活線挿抜機能とを提供することができる。

【0012】

又、IEEE 1394 - 1995規格に準拠した通信システムにおいて、リアルタイム性は必要とされないが信頼性が要求される比較的データ量の多いオブジェクトデータ(例えば、静止画像データ、グラフィックスデータ、テキストデータ、ファイルデータ、プログラムデータ等)を、1以上のセグメントデータに分割して連続的に転送するための通信プロトコルは具体的に提案されていなかった。

30

【0013】

又、このような通信システムでは、通常バスリセットが発生した場合、転送中であったセグメントを再び始めからやり直さなければならなかった。特に、ディスティネーションのバッファ容量が大きく、しかもセグメントのほとんどの転送が完了していた場合には、転送の再開に時間がかかり、同じデータを無駄に転送してしまう問題があった。

【0014】

また、前述の問題を解決するために、ディスティネーションのバッファサイズを予め小さく設定し、廃棄されるデータ量と再送されるデータ量とを減らすことが考えられる。しかしながら、そのように構成した場合でも、セグメント単位に転送される制御コマンドとそれに対するレスポンスの通信が増加し、伝送効率が悪くなってしまいう問題があった。

40

【0015】

そこで、本発明は、転送の再開にかかる時間と重複して転送されるデータ量とを減らすことができるデータ通信システムを提供することを目的とする。

【0016】

【課題を解決するための手段】

本発明に係るデータ通信システムの一つは、被送信データを送信するソースと、前記被送信データを受信するディスティネーションと、前記ソースと前記ディスティネーションとの間の通信を管理するコントローラとを有し、前記ソースは、前記被送信データ内の各データを、前記ディスティネーションが有する受信バッファの一部を指定するのに用いら

50

れるオフセットアドレスとともに送信する機能と、バスリセットが発生した場合において、前記被送信データの送信を中断する機能とを有し、前記ディスティネーションは、前記バスリセットが発生した後において、前記バスリセットが発生する前に正常に受信されたデータに対応する前記オフセットアドレスを前記ソースに通知する機能を有し、前記ソースはさらに、前記バスリセットが発生した後において、前記ディスティネーションから通知された前記オフセットアドレスに従って前記被送信データの送信を再開する機能を有することを特徴とする。

【0021】

【発明の実施の形態】

以下、本発明のデータ通信システム、データ通信方法、データ通信装置及び記憶媒体の一実施の形態を図面を参照して説明する。 10

図1は、本実施例の通信システムの構成について説明する図である。図1において、各機器は、IEEE1394-1995規格（以下、1394規格）に準拠したデジタルインタフェース105を具備している。

【0022】

図1の通信システムは、TV101、デジタルビデオテープレコーダ（以下、DVTR）102、プリンタ103、デジタルカムコーダ（以下、DVCR）104により構成されている。

【0023】

図1において、TV101、DVTR102、DVCR104は、制御部106、信号処理部107を具備し、プリンタ103は制御部106、画像処理部109を具備している。各機器は、1394規格に準拠した通信ケーブルを介して接続されている。 20

【0024】

ここで、通信ケーブルには、4ピンケーブルと6ピンケーブルの2種類がある。4ピンケーブルは、データの転送や調停信号の通信に用いられる2組のシールド付きのより対線により構成されている。6ピンケーブルは、2組のより対線と電源供給用のペア線からなる。2組のより対線を用いて伝送されるデータは、DS-Link方式により符号化されたデータである。

【0025】

次に、図2を用いて本実施例のデジタルインタフェース105の構成について詳細に説明する。 30

デジタルインタフェース105は、機能的に複数のレイヤ（階層）から構成されている。図2において、デジタルインタフェース105は、IEEE1394-1995規格に準拠した通信ケーブル201を介して他の機器のデジタルインタフェース105と接続される。又、デジタルインタフェース105は、1つ以上の通信ポート202を有し、各通信ポート202はハードウェア部に含まれるフィジカル・レイヤ203と接続される。

【0026】

図2において、ハードウェア部は、フィジカル・レイヤ203とリンク・レイヤ204とから構成されている。フィジカル・レイヤ203は、他のノードとの物理的、電気的なインタフェース、バスリセットの検出とそれに伴う処理、入出力信号の符号化/復号化、バス使用権の調停等を行う。又、リンク・レイヤ204は、通信パケットの生成、各種の通信パケットの送受信、サイクルタイムの制御等を行なう。又、リンク・レイヤ204は、後述する通信プロトコルにおいて規定するパケットの生成及び送受信の機能を提供する。 40

【0027】

又、図2において、ファームウェア部は、トランザクション・レイヤ205とシリアル・バス・マネージメント206とを含んでいる。トランザクション・レイヤ205は、Asynchronous転送方式を管理し、各種のトランザクション（リード、ライト、ロック）を提供する。

【0028】

又、トランザクション・レイヤ205は、後述する通信プロトコルにおいて規定するトラ 50

ンザクションを管理する機能を提供する。シリアル・バス・マネージメント206は、IEEE1212CSR規格に基づいて、自ノードの制御、自ノードの接続状態の管理、自ノードのID情報の管理、シリアルバスネットワークの資源管理を行う機能を提供する。又、シリアル・バス・マネージメント206は、後述する通信プロトコルに関する各種の処理動作を制御する機能を提供する。

【0029】

図2に示すハードウェア部及びファームウェア部が実質的に1394インタフェースを構成するものであり、それらの基本構成は、IEEE1394-1995規格により規定されている。

【0030】

又、ソフトウェア部に含まれるアプリケーション・レイヤ207は、使用するアプリケーションソフトによって異なり、どのようなオブジェクトデータをどのように転送するかを制御する。

【0031】

後述する本実施例の通信プロトコルは、デジタルインタフェース105を構成するハードウェア部及びファームウェア部の機能を拡張するものであり、ソフトウェア部に対して新規な転送手順を提供するものである。

【0032】

前述のデジタルインタフェース105では、電源投入時、新たな機器の接続や切り離し等の接続構成の変化に応じてバスリセットを自動的に行う。ここで、バスリセットとは、通信システムを構成する各機器(以下、ノード)が、今までの認識していた通信システムの接続構成と各機器の通信アドレス(以下、ノードID)とを初期化し、新たな接続構成の再認識と通信アドレスの再設定とを行うため処理である。

【0033】

以下、バスリセットの処理手順を簡単に説明する。この手順は、通信システムにおける階層的な接続構成の認識と各ノードに対する物理的な通信アドレスの付与からなる。

【0034】

接続構成の認識は、バスリセットの開始後、各ノードが親子関係を宣言し合うことによって実行される。各ノードは、各ノード間の親子関係を決定することにより、通信システムをツリー構造(階層構造)として認識する。尚、各ノード間の親子関係は、通信システムの接続状態や各ノードの機能に依存するため、バスリセット毎に同じ関係になることはない。

【0035】

例えば、図1の通信システムにおいて、デジタルインタフェース105では、まず、プリンタ103(以下、ノードD)とDVTR102(以下、ノードC)との間で親子関係を設定する。次に、DVCR104(以下、ノードB)とTV101(以下、ノードA)との間、及びノードCとノードAとの間で親子関係を設定する。

【0036】

最終的に全てのノードの親(或いは上位)と認識された機器がルート・ノードとなり、この通信システムのバス使用权の調停を管理する。図1の通信システムでは、ノードAがルート・ノードとなる。

【0037】

ルート・ノードの決定後、通信システムを構成する各ノードは、ノードIDの設定を自動的に開始する。ノードIDの設定は、基本的に親ノードがポート番号の若い通信ポートに接続された子ノードに対して物理アドレスの設定を許可し、更にその子ノードが自分の子ノードに対して順番に設定の許可を与えることによって実行される。自己のノードIDを設定したノードは、セルフIDパケットを送出し、自己に付与されたノードIDを他のノードに対して通知する。最終的に全ての子ノードのID設定が終了した後、親ノードは自己のノードIDを設定する。

【0038】

10

20

30

40

50

以上の処理を繰り返し実行することによって、ルート・ノードのノードIDが一番最後に設定される。尚、各ノードに割り当てられるノードIDは、各機器の親子関係に依存するため、バスリセット毎に同じノードIDが設定されることはない。

【0039】

次に、図1を用いてノードIDの自動設定処理を説明する。尚、以下の説明では、接続構成の認識後、ノードAがルート・ノードとなった場合について説明する。

【0040】

図1において、ルート・ノードであるノードAはまず、「ポート1」の通信ポートに接続されているノード、即ちノードBに対してノードIDの設定を許可する。

【0041】

ノードBは、自己のノードIDを「#0」に設定し、その結果をセルフIDパケットとして通信システムを構成する全てのノードに対してブロードキャストする。ここで、ブロードキャストとは、所定の情報を不特定多数のノードを宛先として送出することである。

【0042】

この結果、全てのノードは、「ノードID「#0」は割り当て済である」と認識し、次にノードIDの設定を許可されたノードは「#1」を設定する。

ノードBの設定後、ノードAは、「ポート2」の通信ポートに接続されているノード、即ちノードCに対してノードIDの設定を許可する。

【0043】

ノードCは更に、子ノードの接続されている通信ポートの内、最も若いポート番号の通信ポートから順に設定の許可を与える。つまり、ノードDに対して許可を与え、その許可を受けたノードDがノードID「#1」を設定した後、セルフIDパケットをブロードキャストする。

【0044】

ノードDの設定後、ノードCが自己のノードIDを「#3」に設定し、最後にルート・ノードであるノードAが自己のノードIDを「#4」に設定して接続構成の認識を終了する。

【0045】

このようなバスリセット処理により、各ノードのデジタルインタフェース105は、通信システムの接続構成の認識と各ノードの通信アドレスの設定とを自動的に行うことができる。そして各ノードは、この前述のノードIDを用いることにより各ノード間の通信を行なうことができる。

【0046】

次に、デジタルインタフェース105の具備するデータ転送方式について、図3を用いて説明する。

図3の通信システムは、データ転送方式としてIsochronous転送モードとAsynchronous転送モードとを具備している。Isochronous転送モードは、1通信サイクル期間(125 μ s)毎に一定量のパケットの送受信を保証するため、ビデオデータや音声データのリアルタイムな転送に有効である。また、Asynchronous転送モードは、制御コマンドやファイルデータ等を必要に応じて非同期に送受信する転送モードであり、Isochronous転送モードに比べて優先順位が低く設定されている。

【0047】

図3において、各通信サイクルの始めには、サイクル・スタート・パケット301と呼ばれる各ノードの計時するサイクル時間を調整するための通信パケットが送出される。

【0048】

サイクル・スタート・パケット301の転送後、所定の期間がIsochronous転送モードに設定されている。Isochronous転送モードでは、Isochronous転送モードに基づいて転送されるデータの夫々に対してチャンネル番号を付すことにより、複数のIsochronous転送を実行することができる。

【0049】

10

20

30

40

50

例えば、図3において、DVC R 1 0 4 から Isochronous 転送されるデータ 3 0 2 にチャンネル番号「c h 0」、DVT R 1 0 2 から Isochronous 転送されるデータ 3 0 3 にチャンネル番号「c h 1」、TV 1 0 1 から Isochronous 転送されるデータ 3 0 4 にチャンネル番号「c h 2」が割り当てられている場合、各データは、1 通信サイクル期間内において時分割に Isochronous 転送される。

【 0 0 5 0 】

各 Isochronous 転送が終了した後、次のサイクル・スタート・パケット 3 0 1 の転送される期間までが Asynchronous 転送に使用される。例えば、図3では、Asynchronous 転送に基づくデータ 3 0 5 が DVC R 1 0 4 から プリント 1 0 3 に転送される。

【 0 0 5 1 】

図4は、Asynchronous 転送モードを用いた本実施の形態の通信プロトコルの基本構成について説明するシーケンスチャートである。図4において、オブジェクトデータ、例えば静止画像を順次 Asynchronous 転送するノード、即ちソース 4 0 2 を DVC R 1 0 4 とする。また、ソース 4 0 2 から Asynchronous 転送されたオブジェクトデータを受信するノード、即ちディスティネーション 4 0 3 を プリント 1 0 3 とする。更に、ソース 4 0 2 とディスティネーション 4 0 3 との間の通信を管理するノード、即ちコントローラ 4 0 1 を TV 1 0 1 とする。

【 0 0 5 2 】

本実施の形態の通信プロトコルは、3つのフェーズからなっている。第1のフェーズ 4 0 4 は、コネクションフェーズで、コントローラ 4 0 1 は、ディスティネーション 4 0 3 の受信バッファサイズ、後述するディスティネーションオフセット、受信可能か否かを問い合わせ、ディスティネーション 4 0 3 を受信待機状態にセットする。また、コントローラ 4 0 1 は、ソース 4 0 2 から Asynchronous 転送されるオブジェクトデータを選択して、ソース 4 0 2 を転送待機状態にセットする。

【 0 0 5 3 】

第2のフェーズ 4 0 5 は、伝送フェーズで、コントローラ 4 0 1 は、ソース 4 0 2 とディスティネーション 4 0 3 とを制御し、オブジェクトデータを1つ以上のパケットにより順次 Asynchronous 転送する。

【 0 0 5 4 】

第3のフェーズ 4 0 6 は、コネクションリリースフェーズで、コントローラ 4 0 1 は、ディスティネーション 4 0 3 の受信バッファを自己の管理下から解放し、同様にソース 4 0 2 の送信バッファを自己の管理下から解放する。

【 0 0 5 5 】

図5は、ソース 4 0 2 から Asynchronous 転送されるオブジェクトデータとディスティネーション 4 0 3 の受信バッファとの関係を説明する図である。

ソース 4 0 2 から Asynchronous 転送されるオブジェクトデータ 5 0 1 は、コントローラ 4 0 1 から通知されたディスティネーション 4 0 3 の受信バッファサイズに等しい1つ以上のセグメント 5 0 2 に分割される。

【 0 0 5 6 】

それぞれのセグメント 5 0 2 に含まれるデータは、1つ以上の Asynchronous 転送モードに基づく通信パケット 5 0 3 (以下、Asynchronous パケット 5 0 3) にパケットサイズされ、ソース 4 0 2 からディスティネーション 4 0 3 に順次転送される。

【 0 0 5 7 】

ディスティネーション 4 0 3 は、ソース 4 0 2 からの Asynchronous パケット 5 0 3 を順次受信し、一時的に受信バッファ 5 0 4 に書き込む。1セグメント分のオブジェクトデータの転送が終了した後、ディスティネーション 4 0 3 は、受信バッファ 5 0 4 に格納された1セグメント分のデータを内部メモリ 5 0 5 に順次書き込む。

【 0 0 5 8 】

次に、図6は用いて、ディスティネーション 4 0 3 を含む全てのノードの有する受信バッファ 5 0 4 について詳細に説明する。受信バッファ 5 0 4 は、IEEE 1212 CSR (

10

20

30

40

50

Control and Status Register Architecture) 規格 (又は、ISO/IEC 13213:1994 規格) に準拠した 64 ビットのアドレス空間により管理されている。IEEE 1212 CSR 規格とは、シリアルバス向けの制御、管理、アドレス割り振りを規定した規格である。

【0059】

図 6 (a) は、64 ビットのアドレスにより表される論理的なメモリ空間である。又、図 6 (b) は、図 6 (a) に示すアドレス空間の一部であり、例えば、上位 16 ビットが FFF_{16} となるアドレス空間である。受信バッファ 504 は、図 6 (b) に示すアドレス空間の一部を使用し、アドレスの下位 48 ビットを示すデスティネーション・オフセットにより指定される。このデスティネーション・オフセットは、各 Asynchronous 10
 s パケットのヘッダ部により指定される。

【0060】

図 6 (b) において、例えば、 $000000000000_{16} \sim 0000000003FF_{16}$ は予約された領域であり、実際にオブジェクトデータ 501 の書き込まれる領域は、アドレスの下位 48 ビットが $FFFFF0000400_{16}$ 以降となる領域である。

【0061】

(第 1 の実施の形態)

以下、第 1 の実施の形態の通信プロトコルを、図 1 に示す通信システムに適用した場合について説明する。ここで、第 1 の実施の形態におけるコントローラ 401 を TV101、ソース 402 を DVCR104、デスティネーション 403 をプリンタ 103 とする。 20

【0062】

第 1 の実施の形態において、ソース 402 は、図 5 に示すように、1 つのオブジェクトデータ (例えば、画像データ、音声データ、グラフィックスデータ、テキストデータ等) 501 を 1 つ以上のセグメント 502 に分割した後、各セグメント 502 を 1 つ以上の Asynchronous パケット 503 にて Asynchronous 転送する。また、デスティネーション 403 は、ソース 402 から Asynchronous 転送された 1 つ以上の Asynchronous パケット 503 を受信バッファ 504 に書込み、1 セグメント 502 毎に内部メモリ 505 に格納する。

【0063】

更にコントローラ 401 は、デスティネーション 403 のバッファサイズの間合せ、ソース 402 から転送されるオブジェクトデータの選択、ソース 402 にて生成されるセグメントのサイズの指定等、ソース 402 とデスティネーション 403 との間の通信を管理する。 30

【0064】

また、第 1 の実施の形態の通信プロトコルは、図 4 に示すように、3 つのフェーズ、即ち、コネクションフェーズ、伝送フェーズ、コネクションリリースフェーズから構成されている。

【0065】

ここで、第 1 の実施の形態におけるコネクションフェーズとコネクションリリースフェーズとは、図 4 に示す第 1 のフェーズ 404、第 3 のフェーズ 406 と同様に実行することができる。従って、第 1 の実施の形態では、伝送フェーズについて詳細に説明する。 40

【0066】

図 7 は、第 1 の実施の形態の伝送フェーズについて詳細に説明するシーケンスチャートである。

図 7 において、コントローラ 401 は、デスティネーション 403 に対して、いくつかの通信パケットで Asynchronous 転送される 1 セグメントのデータを受信するように指示する (704)。また、コントローラ 401 は、ソース 402 に対して、オブジェクトデータを 1 つ以上のセグメントに分割し、各セグメントをいくつかの通信パケットで Asynchronous 転送するように指示する (705)。

【0067】

ここで、コントローラ 401 からソース 402 に転送されるコマンドの構成を図 8 を用い 50

て説明する。このコマンドは、例えば、ソース402に1セグメントの転送の開始を指示するコマンド（以下、転送開始コマンド）であり、Asynchronous転送モードにより転送される。

【0068】

図8において、最初のフィールド801には、転送の開始等を指示するコマンドデータが格納される。フィールド802には、送信するセグメントの順番を示すセグメントナンバーが格納される。フィールド803には、ディスティネーション403のノードIDが格納される。

【0069】

フィールド804には、ディスティネーション403の具備する受信バッファ504の先頭アドレスが格納される。フィールド805には、ディスティネーション403の具備する受信バッファのサイズ、つまり1セグメントのサイズ情報が格納される。フィールド806には、ディスティネーション403の受信可能な通信パケットの最大サイズ情報が格納される。フィールド807には、各種のステータス情報等が格納される。

【0070】

また、図8において、フィールド807の所定の領域には、再送識別ビットを格納する領域808が設定されている。ソース402は、この領域808を読み出すことによって、通常の転送処理を行うか、再送処理を行うかを判断する。例えば、コントローラ401が、ソース402に対して通常のセグメントの転送を指示する場合、この領域808には“0”が格納される。

【0071】

第1の実施の形態において、転送開始コマンドを受信したソース402は、フィールド804に格納された受信バッファ504の先頭アドレスの値を所定の内部レジスタに格納するように構成されている。ここで、内部レジスタは、各機器の具備するデジタルインタフェース105或いは各機器の動作を制御する制御部106（図1参照）に含まれている。

【0072】

図7において、コントローラ401からの指示後、ソース402は、1セグメント502を1つ以上のAsynchronousパケット503にパケットサイズし、各Asynchronousパケット503をディスティネーション403へ順次転送する（706）。

【0073】

ここで、各Asynchronousパケット503には、ディスティネーション403の具備する受信バッファ504の所定の領域を指定するアドレス（ディスティネーションオフセットアドレス）が格納されている。例えば、あるセグメントの最初のAsynchronousパケット503には、コントローラ401により通知された受信バッファ504の先頭アドレスが格納されている。また、それ以降のAsynchronousパケットには、その受信バッファ504の所定の領域を順次指定するオフセットアドレスが格納される。

【0074】

あるセグメント502の転送中にバスリセット707が生じた場合、そのソース402はそのセグメントの転送を一時中断する。ディスティネーション403は、そのセグメントの受信を中断すると共に、バスリセットが発生する前に正常に受信された最後のAsynchronousパケットに含まれるオフセットアドレスを前述の内部レジスタに格納する。また、ディスティネーション403は、受信バッファ504に格納されたデータ（即ち、途中まで転送されたセグメント）を廃棄せず、そのまま保持する。

【0075】

バスリセット707を検出したコントローラ401は、バスリセット処理の終了後、ソース402とディスティネーション403のノードIDが変化したか否かを調べる。その後、コントローラ401は、ソース402及びディスティネーション403に対してデータ転送の再開を指示する（708、709）。

【0076】

ここで、ソース402に対して転送される転送の再開を指示するコマンドには、バスリセ

10

20

30

40

50

ット707にて再設定されたディスティネーション403のノードIDが格納されている。また、このコマンドは、図8のように構成されており、フィールド807内の領域808(再送識別ビット)には、“1”が格納されている。ソース402は、この領域808を読み出すことによって、再送処理の実行を認識する。

【0077】

同様に、ディスティネーション403に対して転送される転送の再開を指示するコマンドには、バスリセット707にて設定されたソース402のノードIDが格納されている。

【0078】

コントローラ401からの指示を受けたディスティネーション403は、ソース402に対して内部レジスタに格納しておいたオフセットアドレスを通知すると共に、ソース402から再送されるセグメントを待機する(710)。

10

【0079】

また、コントローラ401からの指示を受けたソース402は、ディスティネーション403からのオフセットアドレスの通知を待機する。オフセットアドレスの通知後、ソース402は、セグメントの途中からAsynchronous転送を再開する(711)。

【0080】

ここで、ソース402は、前述の内部レジスタに格納されたディスティネーション403の具備する受信バッファ504の先頭アドレスの値と、ディスティネーション403から通知されたオフセットアドレスの値とを比較し、その差をとることにより転送を開始すべきデータを識別している。

20

【0081】

例えば、図9に示すように、内部レジスタに格納されている受信バッファ504の先頭アドレスの下位16bitが「0E00h」、前述のオフセットアドレスの下位16bitが「0E04h」であった場合、ソース402は、バスリセットにより転送の中断されたセグメントの内、5Byte目のデータ901からAsynchronous転送を再開する。

【0082】

このセグメントのAsynchronous転送が完了した後、ソース402は、転送の完了をコントローラ401に報告する(712)。また、ディスティネーション403も同様に、1セグメントの受信が完了したことをコントローラ401に報告する(713)。

【0083】

このように、1セグメント分のデータの転送中にバスリセットが生じても、図7の708~713に示す手順を実行することにより、そのセグメントの転送を始めからやり直すことなく再開することができる。

30

次以降のセグメントの転送を開始する場合には、コントローラ401、ソース402、ディスティネーション403が、再び704~713に示す手順を繰り返し行えばよい。

【0084】

以上のように、第1の実施の形態では、コントローラ401がバスリセットを検出した後、ソース402とディスティネーション403とに送信の再開を要求する。そして、ソース402は、この要求を受けたディスティネーション403から通知されたオフセットアドレスを用いて、転送の中断されたセグメントから送信すべきデータを選択し、それを順次Asynchronous転送するように構成されている。

40

【0085】

これにより、各セグメントの転送中にバスリセットが生じても、転送の再開にかかる時間と重複して転送されるデータ量とを削減することができると共に、伝送効率の低下を回避することができる。

【0086】

(第2の実施の形態)

以下、第2の実施の形態の通信プロトコルを、第1の実施の形態と同様に、図1に示す通信システムに適用した場合について説明する。ここで、第2の実施の形態におけるコントローラ401をTV101、ソース402をDVC R104、ディスティネーション40

50

3をプリンタ103とする。

【0087】

以下、第2の実施の形態において、第1の実施の形態と同一あるいはそれに相当する部材或いは機能については同一符号を用いて説明を省略する。

第2の実施の形態の通信プロトコルは、第1の実施の形態と同様に、図4に示すように、3つのフェーズ、即ち、コネクションフェーズ404、伝送フェーズ405、コネクションリリースフェーズ406から構成されている。以下、第2の実施の形態では、第1の実施の形態と同様に、伝送フェーズについて詳細に説明する。

【0088】

図10は、第2の実施の形態の伝送フェーズについて詳細に説明するシーケンスチャートである。 10

図10において、バスリセット707が発生するまでの手順、即ち図7の704～706までの手順は、第1の実施の形態と同様に動作するため、その説明を省略する。以下、バスリセット707が発生した後の処理について説明する。

【0089】

あるセグメントのAsynchronous転送中にバスリセット707が生じた場合、ソース702はそのセグメントの転送を中断する。転送を中断すると共に、ソース702は、バスリセットが発生する前に送信された最後のAsynchronousパケット503に含まれるオフセットアドレスを内部レジスタに格納する。

【0090】

ここで、オフセットアドレスが格納される内部レジスタは、各機器の具備するデジタルインタフェース105或いは各機器の動作を制御する制御部106に含まれている。尚、前述のオフセットアドレスは、受信バッファ504の先頭アドレスが格納される内部レジスタとは異なる内部レジスタに格納される。 20

【0091】

又、ディスティネーション403は、そのセグメントの受信を中断すると共に、受信バッファ504に格納されたデータ(即ち、途中まで転送されたセグメント)を廃棄せず、そのまま保持する。

【0092】

バスリセット707を検出したコントローラ401は、バスリセット処理の終了後、ソース402とディスティネーション403のノードIDが変化したか否かを調べる。その後、コントローラ401は、ソース402及びディスティネーション403に対してデータ転送の再開を指示する(1001、1002)。 30

【0093】

ここで、ソース402に対して転送される転送再開コマンドには、バスリセット707にて再設定されたディスティネーション403のノードIDが格納されている。また、このコマンドは、図2のように構成されており、フィールド807内の領域808(再送識別ビット)には、“1”が格納されている。ソース402は、この領域808を読み出すことによって、再送処理の実行を認識する。

【0094】

同様に、ディスティネーション403に対して転送される転送の再開を指示するコマンドには、バスリセット707にて設定されたソース402のノードIDが格納されている。 40

【0095】

コントローラ401からの指示を受けたディスティネーション403は、ソース402から再送されるデータを待機する。

また、コントローラ401からの指示を受けたソース402は、前述の内部レジスタに格納しておいたオフセットアドレスを読み出し、そのアドレスに対応するデータからのAsynchronous転送を再開する(1003)。

【0096】

ここで、ソース402は、前述の内部レジスタに格納されたディスティネーション403 50

の具備する受信バッファ504の先頭アドレスの値と、前述のオフセットアドレスの値とを比較し、その差をとることにより転送を開始すべきセグメントの一部を識別している。

【0097】

例えば、図9に示すように、内部レジスタに格納されている受信バッファ504の先頭アドレスの下位16bitが「0E00h」、前述のオフセットアドレスの下位16bitが「0E04h」であった場合、ソース402は、バスリセットにより転送の中断されたセグメントの内、5Byte目のデータ901からAsynchronous転送を再開する。

【0098】

このセグメントのAsynchronous転送が完了した後、ソース402は、転送の完了をコントローラ401に報告する(712)。また、ディスティネーション403も同様に、1セグメントの受信が完了したことをコントローラ401に報告する(713)。

10

【0099】

このように、1セグメント分のデータの転送中にバスリセットが生じても、図10に示す手順を実行することにより、そのセグメントの全ての転送を始めからやり直すことなく再開することができる。

【0100】

次以降のセグメントの転送を開始する場合には、コントローラ401、ソース402、ディスティネーション403が、再び図10に示す手順を繰り返し行えばよい。

【0101】

以上のように、第2の実施の形態では、コントローラ401がバスリセットを検出した後、ソース402とディスティネーション403とに送信の再開を要求する。そして、ソース402が、バスリセットの前に正常に転送されたAsynchronousパケット503に含まれるオフセットアドレスを用いて、転送の中断されたセグメントデータから送信すべきデータを選択し、それを順次Asynchronous転送するように構成されている。

20

【0102】

これにより、各セグメントの転送中にバスリセットが生じても、第1の実施の形態と同様に、転送の再開にかかる時間と重複して転送されるデータ量とを削減することができる。共に、伝送効率の低下を回避することができる。

【0103】

(第3の実施の形態)

30

以下、第3の実施の形態の通信プロトコルを、図1に示した通信システムに適用した場合について説明する。ここで、第3の実施の形態におけるコントローラ401をTV101、ソース402をDVCR104、ディスティネーション403をプリンタ103とする。

【0104】

以下、第3の実施の形態において、第1の実施の形態と同一あるいはそれに相当する部材或いは機能については同一符号を用いて説明を省略する。

第3の実施の形態において、ソース402は、オブジェクトデータ501を1つ以上のセグメント502に分割した後、各セグメント502を1つ以上のAsynchronousパケット503にてAsynchronous転送する。

40

【0105】

また、ディスティネーション403は、ソース402からAsynchronous転送された1つ以上のAsynchronousパケット503を受信し、1セグメント毎に内部メモリに格納する。更にコントローラ401は、ディスティネーション403のバッファサイズの間合せ、ソース402から転送されるオブジェクトデータの選択、ソース402にて生成されるセグメントのサイズの指定等、ソース402とディスティネーション403との間の通信を管理する。

【0106】

また、第3の実施の形態の通信プロトコルは、第1の実施の形態と同様に3つのフェーズ、即ち、コネクションフェーズ404、伝送フェーズ405、コネクションリリースフェ

50

ーズ406から構成されている。以下、第3の実施の形態では、伝送フェーズについて詳細に説明する。

【0107】

図11は、第3の実施の形態の伝送フェーズについて詳細に説明するシーケンスチャートである。

図11において、バスリセット707が発生するまでの手順、即ち、図11の104～106までの手順は、第1の実施の形態と同様に動作するため、その説明を省略する。以下、バスリセット707が発生した後の処理について説明する。

【0108】

あるセグメントのAsynchronous転送中にバスリセット707が生じた場合、ソース402はそのセグメントの転送を中断する。転送を中断すると共に、ソース402は、バスリセットが発生する前に正常に送信された最後のAsynchronousパケット503に含まれるオフセットアドレスを内部レジスタに格納する。 10

【0109】

ここで、オフセットアドレスが格納される内部レジスタは、各機器の具備するデジタルインタフェース105或いは各機器の動作を制御する制御部106に含まれている。尚、前述のオフセットアドレスは、受信バッファ504の先頭アドレスが格納される内部レジスタとは異なる内部レジスタに格納される。

【0110】

ディスティネーション403は、そのセグメントの受信を中断すると共に、受信バッファ504に格納されたデータ、即ち、途中まで転送されたセグメントを廃棄せず、そのまま保持する。 20

【0111】

バスリセット707を検出したコントローラ401は、バスリセット処理の終了後、ソース402とディスティネーション403のノードIDが変化したか否かを調べる。その後、コントローラ401は、ソース402に対して、正常に送信された最後のAsynchronousパケット503に含まれるオフセットアドレスを問い合わせる(1101)。

【0112】

ここで、ソース402に対して転送される問合せコマンドのデータ部は、図8のように構成されており、そのコマンドのフィールド807内の領域808(再送識別ビット)には、“1”が格納されている。ソース402は、この領域808を読み出すことによって、再送処理の実行を認識する。 30

【0113】

この問い合わせに対し、ソース402は、前述の内部レジスタに格納していたオフセットアドレスを読み出し、コントローラ401に通知する(1102)。ソース402からの通知を受けたコントローラ401は、ディスティネーション403のノードIDを通知すると共に、そのオフセットアドレスに対応するデータからAsynchronous転送を再開するように指示する(1103)。

【0114】

コントローラ401から転送の再開の指示を受けたソース402は、前述のオフセットアドレスに対応するデータからのAsynchronous転送を再開する(1104)。 40

【0115】

ここで、ソース402は、ディスティネーション403の具備する受信バッファ504の先頭アドレスの値と、前述のオフセットアドレスの値とを比較し、その差をとることにより転送を開始すべきセグメントの一部を識別している。

【0116】

例えば、図9に示すように、内部レジスタに格納されている受信バッファ504の先頭アドレスの下位16bitが「0E00h」、前述のオフセットアドレスの下位16bitが「0E04h」であった場合、ソース402は、バスリセットにより転送の中断されたセグメントの内、5Byte目のデータ901からAsynchronous転送を再開する。 50

【 0 1 1 7 】

この 1 セグメントの Asynchronous 転送が完了した後、ソースは、転送の完了をコントローラ 4 0 1 に報告する (7 1 2)。また、ディスティネーション 4 0 3 も同様に、1 セグメントの受信が完了したことをコントローラ 4 0 1 に報告する (7 1 3)。

【 0 1 1 8 】

このように、1 セグメント分のデータの転送中にバスリセットが生じても、図 1 1 に示す手順を実行することにより、そのセグメントの全ての転送を始めからやり直すことなく再開することができる。

【 0 1 1 9 】

次以降のセグメントの転送を開始する場合には、コントローラ 4 0 1、ソース 4 0 2、ディスティネーション 4 0 3 が、再び図 1 1 に示す手順を繰り返し行えばよい。 10

【 0 1 2 0 】

以上のように、第 3 の実施の形態では、コントローラ 4 0 1 がバスリセットを検出した後、バスリセットの前に正常に転送された Asynchronous パケット 5 0 3 に含まれるオフセットアドレスをソース 4 0 2 に問い合わせる。そして、コントローラ 4 0 1 は、ソース 4 0 2 から通知されたオフセットアドレスに対応するデータから Asynchronous 転送を再開するようにソース 4 0 2 に指示するように構成されている。

【 0 1 2 1 】

これにより、各セグメントの転送中にバスリセットが生じても、第 1 の実施の形態と同様に、転送の再開にかかる時間と重複して転送されるデータ量とを削減することができる。 20

【 0 1 2 2 】

(他の実施の形態)

前述の第 1 ~ 第 3 の実施の形態は、以下のように実現することも可能である。例えば、前述した実施の形態の機能を実現するソフトウェアのプログラムコードを記録した記録媒体を、本実施の形態のコントローラ 4 0 1、ソース 4 0 2、ディスティネーション 4 0 3 の具備する制御部 (マイクロコンピュータを含む) に供給するように構成することもできる。

【 0 1 2 3 】

そして、本実施の形態のコントローラ 4 0 1、ソース 4 0 2、ディスティネーション 4 0 3 の具備する制御部が、その記録媒体に格納されたプログラムコードを読み出し、前述した実施の形態の機能を実現するようにシステム或いは装置の動作を制御するように構成しても本発明の実施の形態を実現することができる。 30

【 0 1 2 4 】

例えば、第 1 の実施の形態の図 7、第 2 の実施の形態の図 1 0、第 3 の実施の形態の図 1 1 に示した処理及び機能を実現するプログラムコードを格納した記録媒体を、コントローラ 4 0 1、ソース 4 0 2、ディスティネーション 4 0 3 の制御部 1 0 6 に供給する。

【 0 1 2 5 】

そして、コントローラ 4 0 1、ソース 4 0 2、ディスティネーション 4 0 3 の制御部 1 0 6 が、その記録媒体に格納されたプログラムコードを読み出し、各実施の形態の機能を実現するように、図 1 に示すコントローラ 4 0 1、ソース 4 0 2、ディスティネーション 4 0 3 の具備する各処理回路を動作させるようにしてもよい。 40

【 0 1 2 6 】

この場合、記録媒体から読み出されたプログラムコード自体が前述した実施の形態の機能を実現することになり、そのプログラムコードを記憶した記録媒体は、本発明の一部の構成要件になる。

【 0 1 2 7 】

プログラムコードを供給するための記録媒体としては、例えば、フロッピディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。 50

【 0 1 2 8 】

また、制御部上で稼動しているOS（オペレーティングシステム）或いはアプリケーションソフト等が、記録媒体より読み出したプログラムコードの指示に基づき、本実施の形態のシステム或いは装置の動作を制御することにより、前述した実施の形態の機能を実現する場合も本発明に含まれることは言うまでもない。

【 0 1 2 9 】

更に、記録媒体から読み出されたプログラムコードが、制御部に接続された機能拡張ボード或いは機能拡張ユニットの具備するメモリに書き込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットの具備する制御部が本実施の形態のシステム或いは装置の動作を制御することにより、前述した実施の形態の機能が実現される場合も本発明に含まれることは言うまでもない。尚、本発明はその精神、またはその主要な特徴から逸脱することなく、様々な形で実施することができる。

10

【 0 1 3 0 】

例えば、第1～第3の実施の形態において、コントローラ401、ソース402、ディスティネーション403の機能を夫々別の機器が具備している場合について説明したが、それに限るものではない。例えば、第1～第3の実施の形態に示したコントローラ401の機能とソース402の機能とを同一の機器に具備させるように構成してもよい。

【 0 1 3 1 】

このように構成した場合、コントローラ401とソース402との間の通信は、各機器に共通する伝送路を介することなく行われるため、通信システム全体の伝送効率をより向上させることができる。

20

【 0 1 3 2 】

しかし、本発明は、コントローラ401とソース402とが同一の機器内に存在する場合よりも、第1～第3の実施の形態に示すように、コントローラ401、ソース402、ディスティネーション403とが夫々別の機器として存在する場合において、より高い効果を奏するものである。このことは、コントローラ401の機能とディスティネーション403の機能とが同一の機器内に存在する場合についても同様である。したがって、前述の実施の形態はあらゆる点において単なる例示に過ぎず、限定的に解釈してはならない。

【 0 1 3 3 】

【 発明の効果 】

本発明に係るデータ通信システムによれば、転送の再開にかかる時間と重複して転送されるデータ量とを削減することができる。

30

【 図面の簡単な説明 】

【 図 1 】 本実施の形態の通信システムの構成を示すブロック図である。

【 図 2 】 本実施の形態の1394インタフェースの構成を示す図である。

【 図 3 】 1394インタフェースの有する転送方式を説明する図である。

【 図 4 】 本実施の形態の通信プロトコルの基本手順を説明するシーケンスチャートである。

【 図 5 】 オブジェクトデータの転送モデルを説明する図である。

【 図 6 】 各ノードの有するアドレス空間を説明する図である。

40

【 図 7 】 第1の実施の形態の通信プロトコルを説明する図である。

【 図 8 】 コントローラからソースに転送される通信パケットの構成を示す図である。

【 図 9 】 ソースの具備する内部アドレスを説明する図である。

【 図 10 】 第2の実施の形態の通信プロトコルを説明するシーケンスチャートである。

【 図 11 】 第3の実施の形態の通信プロトコルを説明するシーケンスチャートである。

【 符号の説明 】

101 TV

102 DVTR

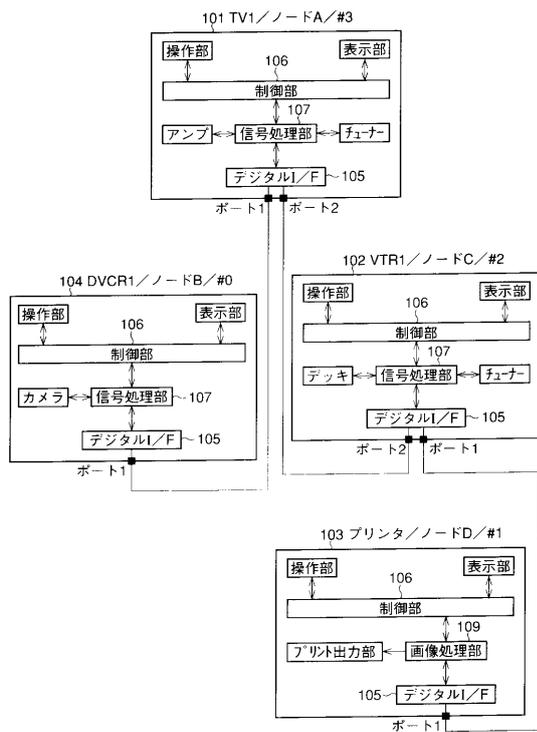
103 プリンタ

104 DVC R

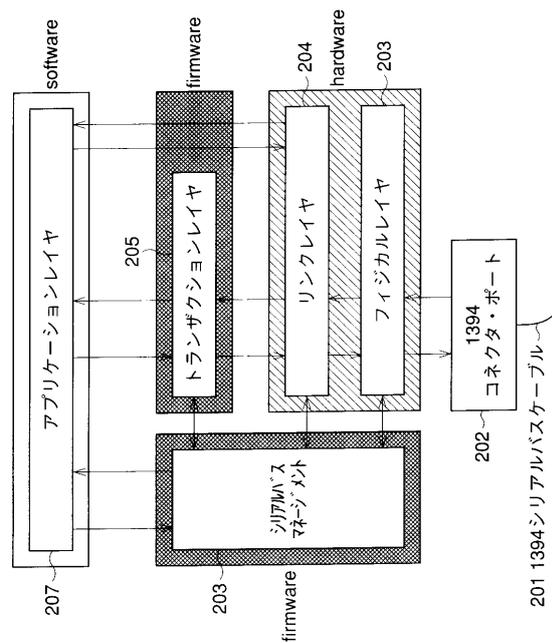
50

- 105 デジタルインターフェース
- 106 制御部
- 107 信号処理部
- 401 コントローラ
- 402 ソース
- 403 ディスティネーション

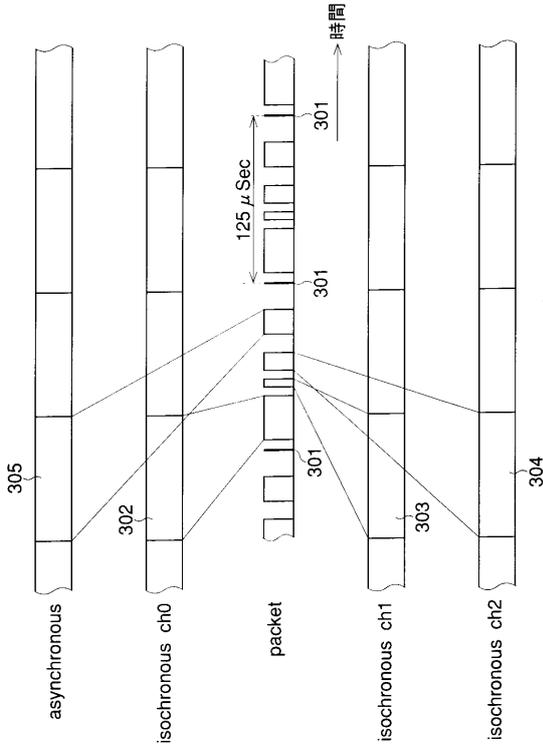
【図1】



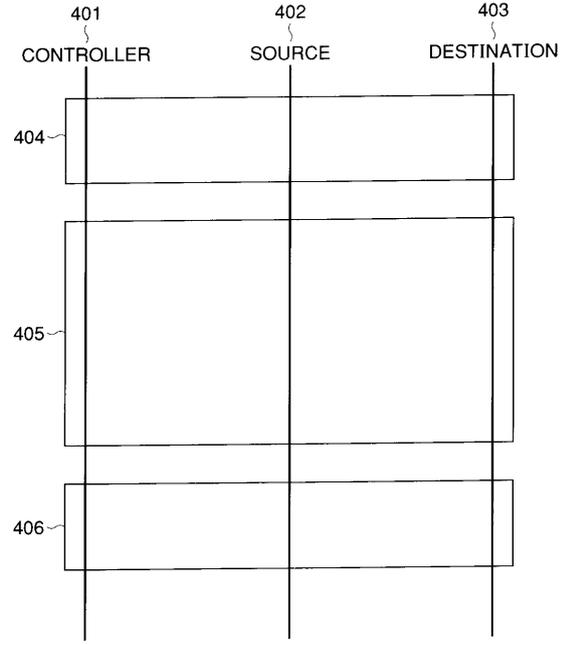
【図2】



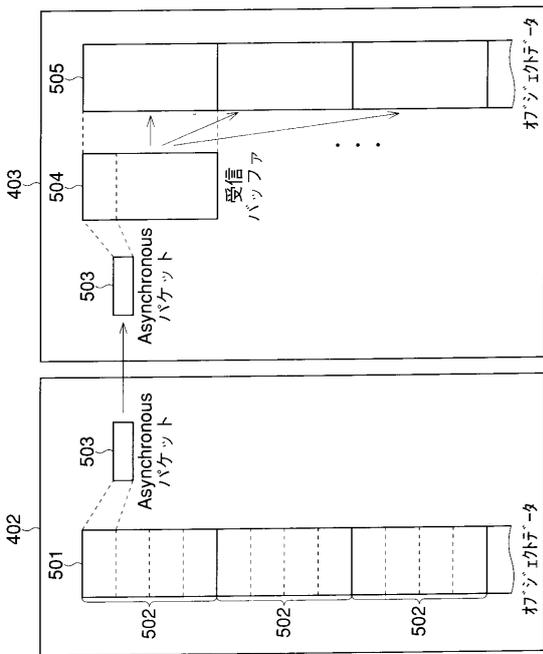
【 図 3 】



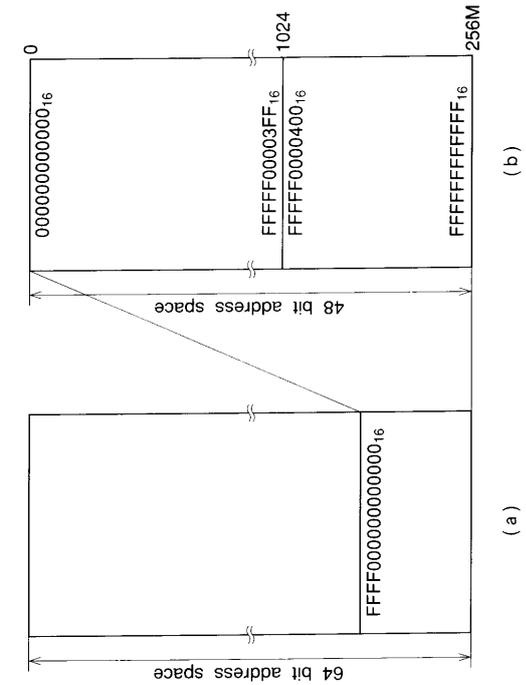
【 図 4 】



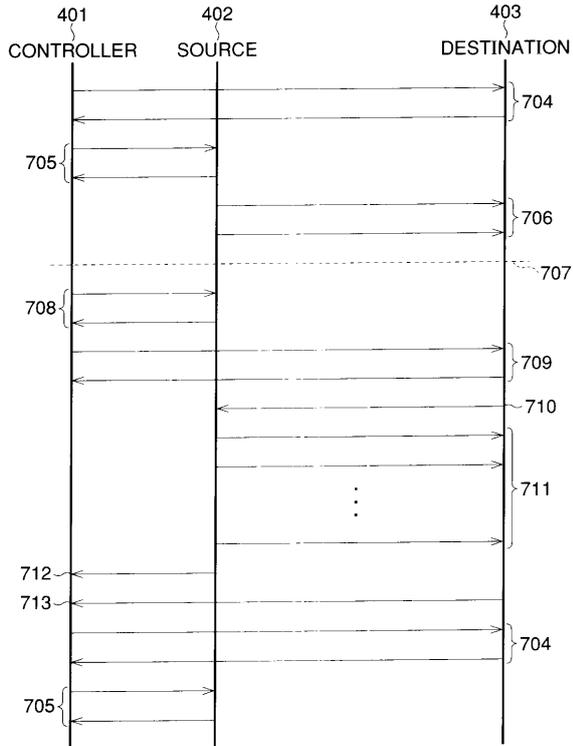
【 図 5 】



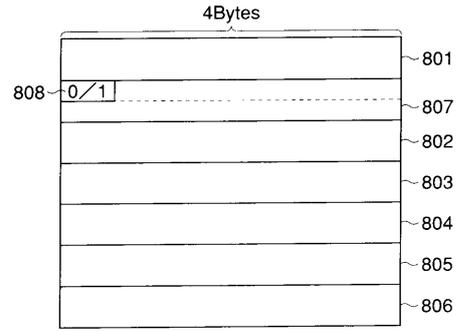
【 図 6 】



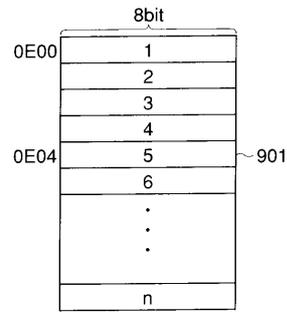
【 図 7 】



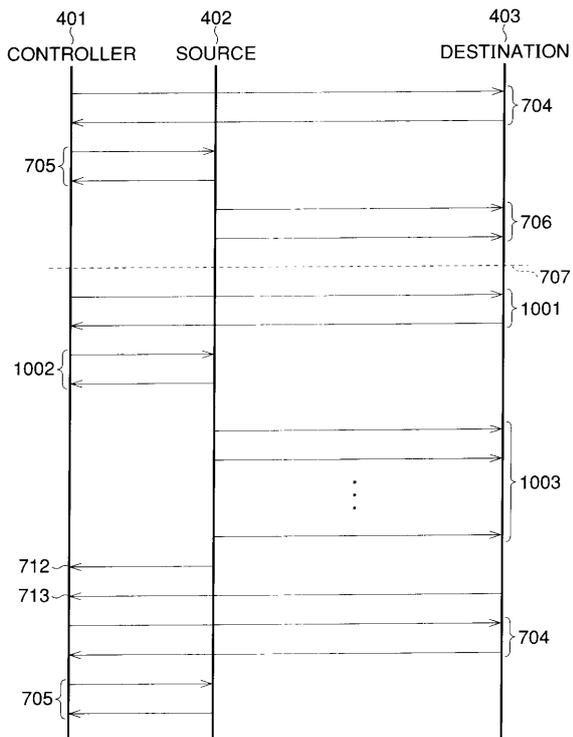
【 図 8 】



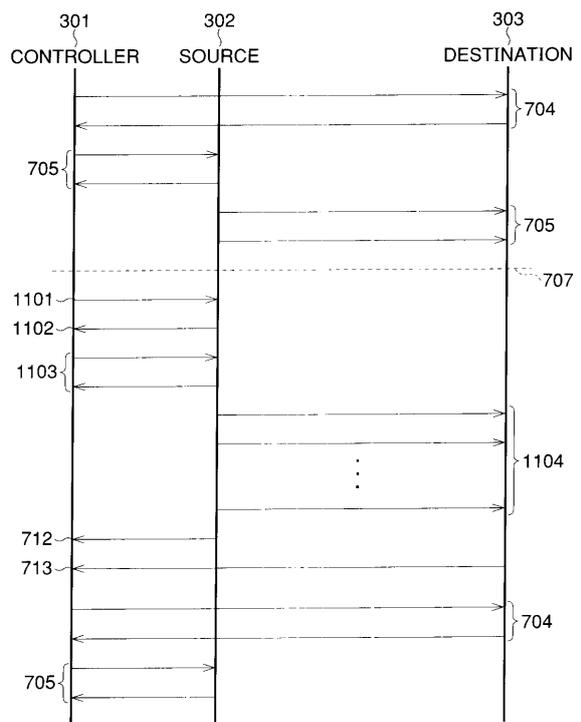
【 図 9 】



【 図 10 】



【 図 11 】



フロントページの続き

(56)参考文献 特開平07-250084(JP,A)
特開平09-298578(JP,A)

(58)調査した分野(Int.Cl., DB名)

H04L 12/40

G06F 13/38