

(19) 日本国特許庁 (JP)

## (12) 特 許 公 報 (B2)

(11) 特許番号

特許第4105382号  
(P4105382)

(45) 発行日 平成20年6月25日 (2008. 6. 25)

(24) 登録日 平成20年4月4日 (2008. 4. 4)

(51) Int. Cl.

F I

<b>G06F 13/14</b>	<b>(2006. 01)</b>	G06F 13/14	320H
<b>G06F 13/38</b>	<b>(2006. 01)</b>	G06F 13/38	350
<b>H04L 29/02</b>	<b>(2006. 01)</b>	H04L 13/00	301Z
<b>H04N 5/44</b>	<b>(2006. 01)</b>	H04N 5/44	Z

請求項の数 10 (全 22 頁)

(21) 出願番号 特願2000-504512 (P2000-504512)  
 (86) (22) 出願日 平成10年7月23日 (1998. 7. 23)  
 (65) 公表番号 特表2001-511558 (P2001-511558A)  
 (43) 公表日 平成13年8月14日 (2001. 8. 14)  
 (86) 国際出願番号 PCT/IB1998/001172  
 (87) 国際公開番号 W01999/005603  
 (87) 国際公開日 平成11年2月4日 (1999. 2. 4)  
 審査請求日 平成17年6月20日 (2005. 6. 20)  
 (31) 優先権主張番号 97401793.1  
 (32) 優先日 平成9年7月24日 (1997. 7. 24)  
 (33) 優先権主張国 欧州特許庁 (EP)

(73) 特許権者 500222364  
 カナル プラス ソシエテ アノニム  
 フランス国 エフ-75711 パリ セ  
 デックス 15 クアイ アンドレ シト  
 ロエン 85/89  
 (74) 代理人 100070150  
 弁理士 伊東 忠彦  
 (72) 発明者 メリク, ジェローム  
 フランス国 エフ-60300 センリス  
 リュ デ ミュークス 55  
 (72) 発明者 デクレリック, クリストフ  
 フランス国 エフ-28210 セナンテ  
 ス リュ ディス オームズ ダンコート  
 3

最終頁に続く

(54) 【発明の名称】 I E E セット・トップ・ボックス装置ドライバ

## (57) 【特許請求の範囲】

## 【請求項 1】

アプリケーションと、対応するインタフェース識別子が割り当てられている少なくとも一つの機能を有するインタフェースとの間で、装置ドライバを介してデータを通信する方法であって、前記インタフェース識別子の前記機能への割当ては可変であり、当該方法は、前記装置ドライバにおいて、

少なくとも一つの前記機能に関して、対応する論理識別子を記憶する工程と、

前記装置ドライバと前記アプリケーションとの間の前記対応する機能に関連する通信を指示するために前記論理識別子を前記アプリケーションに供給する工程と、

前記一つのあるいは各々の機能に割り当てられたインタフェース識別子とは独立に前記一つのあるいは各々の論理識別子と前記一つのあるいは各々の機能との間の対応を保持し、所与の論理識別子を使用して指示された前記アプリケーションと前記装置ドライバとの間の通信が、前記機能への対応するインタフェース識別子の割り当ての変更の後でも、前記対応する所定の機能との関連を保持したままであるようにする工程と、  
 を行うことを特徴とする方法。

## 【請求項 2】

前記インタフェースと前記装置ドライバとの間の通信は、前記一つまたは各々のインタフェース識別子に基づいて指示されることを特徴とする請求項 1 に記載の方法。

## 【請求項 3】

前もって決められた規準を満たす全ての機能について、論理識別子と、対応するインタ

10

20

フェース識別子とのリストをコンパイルする工程を含むことを特徴とする請求項 1 又は 2 記載の方法。

【請求項 4】

前記の装置ドライバは、論理識別子に割り当てられた前記インタフェース識別子を要求のあったときにアプリケーションに通知するように構成されていることを特徴とする請求項 1 ~ 3 のいずれかに記載の方法。

【請求項 5】

前記の装置ドライバは、インタフェース識別子の代わりに少なくとも一つの論理識別子を使って、バスに接続された物理装置間の接続を定義するために、アプリケーションからの要求を受け入れるように構成したことを特徴とする請求項 1 ~ 4 のいずれかに記載の方法。

【請求項 6】

アプリケーションと、インタフェース識別子が割り当てられている少なくとも一つの機能を有するインタフェースとの間で、通信を行う装置ドライバであって、前記一つのまたは各々のインタフェース識別子は少なくとも一つのイベントに応じて可変であり、当該装置ドライバは、

それぞれのインタフェース識別子に対応する少なくとも一つの論理識別子を記憶する手段と、

当該装置ドライバと前記アプリケーションとの間の前記対応する機能に関連する通信を指示するために前記論理識別子を前記アプリケーションに供給する手段と、

前記一つのあるいは各々の機能に割り当てられたインタフェース識別子とは独立に前記一つのあるいは各々の論理識別子と前記一つのあるいは各々の機能との間の対応を保持し、所与の論理識別子を使用して指示された前記アプリケーションと当該装置ドライバとの間の通信が、前記機能への対応するインタフェース識別子の割り当ての変更の後でも、前記対応する所定の機能との関連を保持したままであるようにする手段と、  
で構成したことを特徴とする装置ドライバ。

【請求項 7】

前記の装置ドライバは、ソフトウェアで実現されており、好ましくは前記一つあるいは各々のアプリケーションを実行する処理手段によって実行可能であることを特徴とする請求項 6 に記載の装置ドライバ。

【請求項 8】

前もって決められた基準を満たす全ての機能について、論理識別子と、対応するインタフェース識別子とのリストをコンパイルするように構成したことを特徴とする請求項 6 又は 7 記載の装置ドライバ。

【請求項 9】

論理識別子に割り当てられたインタフェース識別子を、要求のあったときにアプリケーションに通知する手段を含むことを特徴とする請求項 6 ~ 8 のいずれかに記載の装置ドライバ。

【請求項 10】

インタフェース識別子の代わりに少なくとも一つの論理識別子を使って、バスに接続された物理装置間の接続を定義するために、アプリケーションからの要求を受け入れる手段を含むことを特徴とする請求項 6 ~ 9 のいずれかに記載の装置ドライバ。

【発明の詳細な説明】

【0001】

本発明は、物理装置（周辺装置）に対するアプリケーション・プログラムのインタフェースに関するが、特に排他的にではなくデジタル・テレビジョン・システム用の受信機／復号器に関する。

【0002】

デジタル伝送システムの出現は、このようなシステムを他の目的に使う可能性を開いた。その一つは、エンドユーザに対話性を提供することである。ここで使われる「デジタル

10

20

30

40

50

伝送システム」という用語は、例えば主としてオーディオ・ビジュアルの、あるいはマルチメディアのデジタル・データを伝送または放送するための如何なる伝送システムをも含んでいる。本発明は、特に放送デジタル・テレビジョン・システムに適用可能であるが、本発明はまた、マルチメディア・インターネット・アプリケーション用の一定の電気通信網、閉回路テレビジョン等にも適用可能である。ここに用いられる用語「デジタル・テレビジョン・システム」は、例えば如何なる衛星システム、地上システム、ケーブル・システム、その他のシステムを含んでいる。

#### 【 0 0 0 3 】

本発明は、受信された信号が受信器を経由して受信器／復号器に渡され、それからテレビジョン受像機に渡される放送デジタル・テレビジョン・システムに、固有の用途を見いだしている。ここで使われる用語「受信器／復号器」は、符号化信号または非符号化信号のいずれか、例えばテレビジョン信号および／またはラジオ信号を受信するための受信装置を意味することもあり、これらの信号は何か別の手段で放送または伝送されることもある。この用語はまた、受信された信号を復号するための復号器を意味することもある。このような受信器／復号器の実施形態は、例えば物理的に分離している受信器と組み合わせて機能する復号器といった、あるいはウェブ・ブラウザーまたはビデオレコーダまたはテレビジョンといった付加機能を含む復号器といった、「セット・トップ・ボックス」における、受信信号を復号するための受信器と統合された復号器を含むこともある。

#### 【 0 0 0 4 】

この受信器／復号器は、圧縮されたMPEG形式の信号をテレビジョン受像機用のテレビジョン信号に復号する。これは、セット・トップ・ボックスまたはSTBとしても知られている受信器／復号器内のインタフェースを介して遠隔制御送受器によって制御される。用語MPEGは、国際標準化機構のワーキング・グループ「動画専門家グループ」によって開発されたデータ伝送規格を指しているが、特に排他的にではなくデジタル・テレビジョン用に開発されたMPEG-2規格を指しており、またドキュメントISO13818-1、ISO13818-2、ISO13818-3、およびISO13818-4に詳述されている。本特許出願に関しては、この用語は、デジタル・データ伝送の分野に適用可能なMPEGの変形、修正または発展形のすべてを含んでいる。

#### 【 0 0 0 5 】

前述の対話性を与える一つの方法は、テレビジョン信号が経由して受信される受信器／復号器上でアプリケーションを実行することである。種々のアプリケーションが種々の物理装置とトランスペアレントな仕方で通信することを可能にすることが望ましい。筆者らの同時係属中の出願PCT/EP97/02115とPCT/EP97/02116は、一つ以上のアプリケーションが受信器／復号器によってダウンロードされることが可能で、また各装置ごとの装置（デバイス）ドライバおよび全体の装置（デバイス）マネージャ用の装置ドライバを用いて、並列インタフェース、直列インタフェース、スマートカード・リーダーといった、受信器／復号器内の物理装置と通信することができるシステムについて述べている。ここで使われる「スマートカード」という用語は、チップ・ベースのカード装置、または例えばマイクロプロセッサおよび／または記憶装置を有する同様の機能、性能のものを含むが、これらに限定するわけではない。カードに代わる物理形態を有する装置、例えばTV復号器システムでしばしば使われるようなキー形状の装置もこの用語に含まれる。

#### 【 0 0 0 6 】

本発明によれば、高速デジタル・インタフェース上で他のオーディオ・ビジュアル装置、例えばデジタル・ビデオ・レコーダと通信できる能力を受信器／復号器のために用意することが提案されている。最近開発されたIEEE1394規格は、100Mb/s以上のシリアル通信速度を備えた有望で柔軟なインタフェース・プロトコルを提供している。

#### 【 0 0 0 7 】

IEEE1394インタフェースを使用することによる問題は、受信器／復号器以外のバ

10

20

30

40

50

スに接続された装置によって、インタフェース・バスがリセットされるか、あるいはパラメータが変更されることがあると言うことであり、これはアプリケーションに問題を引き起こす可能性がある。このために、インタフェースを処理することの出来る更に複雑なアプリケーションを実行するために、より大きなメモリと処理パワーを必要とするようになる。これは、各受信器／復号器のコストとアプリケーションの開発とデバッグのコストとに上記二つの必要性を追加することになる。

【 0 0 0 8 】

本発明の特徴は、このようなインタフェースにアプリケーションをインタフェースさせる問題を軽減しようとするものである。本発明は、I E E E 1 3 9 4 その他のインタフェースに受信器／復号器をインタフェースさせる際に多くの利点を提供するが、本発明はインタフェースのパラメータがアプリケーションの制御の範囲外で変化するインタフェースに他のアプリケーションをインタフェースさせることに適用することが出来るということが認められるであろう。

10

【 0 0 0 9 】

第1の特徴において本発明は、アプリケーションと、インタフェース識別子が割り当てられている少なくとも一つの機能を有するインタフェースとの間で装置ドライバを介してデータを交信する方法であって、前記一つのあるいは各々のインタフェース識別子は少なくとも一つのイベントの後に変化しやすいものである前記方法において、少なくとも一つの前記機能に関して、対応する論理識別子を記憶することと、前記装置ドライバとアプリケーションとの間での、前記対応する機能に関連する通信を指示するために前記論理識別子を前記アプリケーションに供給することと、所定の論理識別子を使用して指示された前記アプリケーションと装置ドライバとの間の通信が、対応するインタフェース識別子の前記機能への割り当ての変更が続いて前記対応する所定の機能との関連を保持するように、前記一つのあるいは各々の論理識別子に割り当てられたインタフェース識別子とは独立に前記一つのあるいは各々の論理識別子と前記一つのあるいは各々の機能との対応を保持することとからなる、前記方法を提供する。

20

【 0 0 1 0 】

このようにして、インタフェース識別子と機能との結合は、ときどき変化することがあるが、このような変化は、アプリケーションにとって実質的にトランスペアレントにすることができ、その結果さらに簡単にすることができる。

30

【 0 0 1 1 】

インタフェースと装置ドライバとの間の通信は好ましくは、一つのあるいは各々のインタフェース識別子に基づいて指示されるが、これはそのインタフェースとの通信を容易にする。

【 0 0 1 2 】

論理識別子は、一つ以上のアプリケーションによって指定される機能にのみ割り当てることが出来る。これによって、必要とされる論理識別子の数を減らすことができる。それに代わって装置ドライバは、論理インタフェースと、前記機能すべてに関する、あるいは前もって決められた規準を満足する機能すべてに関する、対応するインタフェース識別子のリストをコンパイルするように、また好ましくは、ある機能が追加または除去または変更される度毎に、あるいは何らかのインタフェースが変更された場合にこのリストを更新するように整えることができる。

40

【 0 0 1 3 】

この方法は、アプリケーションがインタフェース識別子を知る必要性を除去するが、好ましくは装置ドライバは、要求のあったときに論理識別子に割り当てられたインタフェース識別子をアプリケーションに通知するように整えられる。インタフェースおよび関連の装置ドライバが所望のように機能しているかどうかをハイレベル・アプリケーションが決定できるので、これは、システムのテストを著しく容易にすることが分かっている。

【 0 0 1 4 】

好ましくは、装置ドライバは、インタフェース識別子の代わりに少なくとも一つの論理識

50

別子を使ってバスに接続された物理装置間の接続を定義するようにというアプリケーションからの要求を受け入れるように整えられる。これは、アプリケーションによる接続の管理を容易にする。

【 0 0 1 5 】

アプリケーションは、装置マネージャ手段を介して装置ドライバと通信するように整えられることが好ましい。装置マネージャ手段を備えることによって、行うべき通信の全体的制御が可能になるので、競合せずに多数のアプリケーションが多数の装置と通信できるようになる。

【 0 0 1 6 】

第1の好適な実施形態では、インタフェースの少なくとも一つの前記機能は、インタフェースに接続された周辺装置を含み、対応するインタフェース識別子は、その周辺装置に割り当てられた物理アドレス（時にはノード・アドレスとしても知られている）を含み、この論理識別子は、周辺装置に割り当てられた論理アドレス（論理周辺装置識別子と呼ばれることもある）を含む。このようにして、所定の論理アドレスを使用するアプリケーションは、周辺装置の物理アドレスが変更されても（例えば別の周辺装置の、バスへの接続とそれに続くバス・リセットに続いて）、所定の周辺装置（例えばデジタル・ビデオレコーダ）との通信を継続することができる。

10

【 0 0 1 7 】

このような場合、対応を維持することは、一つのあるいは各々のイベント、例えばバス・リセットといったイベントに続いて周辺装置に割り当てられた物理アドレスを決定するために、論理アドレスが割り当てられている各々の周辺装置に問い合わせることを含むことが好ましい。これによって、物理アドレスのありそうな変更に続いて割当てを更新することが可能になる。

20

【 0 0 1 8 】

このような場合にも、所定の周辺装置に関するインタフェース識別子を通知することがその周辺装置の物理（あるいはノード）アドレスを通知することを含んでおり、またその周辺装置の更なる識別子、例えばその周辺装置を識別する更なる情報を含む一意のノード識別子を通知することを含んでいれば、これは特に好ましい。この一意のノード識別子は、製造業者および／または販売業者および／または周辺装置のモデル番号を識別することができ、また通し番号を含むこともある。この一意ノード識別子は、少なくとも4バイトであることが好ましく、8バイト長であれば更に好ましい。

30

【 0 0 1 9 】

第2の好適な実施形態によれば、インタフェースの少なくとも一つの前記機能は、そのインタフェースを介して利用可能な定義されたパラメータのチャンネルを含み、またそれに対応するインタフェース識別子は、インタフェース・チャンネル番号（あるいは所謂チャンネル識別子）を含んでおり、論理識別子は論理チャンネル識別子を含んでいる。このようにして、変化する可能性のあるインタフェース・チャンネル番号をアプリケーションが追跡・記憶していることは必要でなくなる。これらのチャンネルは、定義された帯域幅を有する等時性チャンネルであることが好ましい。

40

【 0 0 2 0 】

装置ドライバは好ましくは、定義されたパラメータのチャンネル（例えば、定義された最大帯域幅を有するチャンネル）を割り当てるというアプリケーションからの要求を受け取るように、また割当てが成功した場合には論理チャンネル識別子を返却するように整えられる。アプリケーションは、インタフェース・チャンネル番号を知る必要はないが、もし装置ドライバが、好適なインタフェース・チャンネル番号を受け入れるように、また利用可能であればその好適なインタフェース・チャンネル番号を割り当てるように、またその好適なインタフェース・チャンネル番号が利用可能でない場合または好適なチャンネル・インタフェースが指定されない場合には、空きチャンネルを割り当てるように、整えられていれば、好適である。インタフェース・チャンネルを指定する能力を備えることによって、すべてのアプリケーションがインタフェース・チャンネル番号を認識することを必要とせずに、ある適当なア

50

アプリケーションによるインタフェースの制御とテストとを容易にすることができる。装置ドライバは好ましくは、好適なインタフェース・チャンネルの識別子を受け取るように、また好適なインタフェース・チャンネルを指定せずに有効なインタフェース・チャンネル番号の代わりに前もって決められたキーを認識するように、そして他の無効なインタフェース・チャンネル番号が指定された場合にはアプリケーションにエラーを報告するように整えられており、これはアプリケーションをデバッグする際に助けとなり得る。

【0021】

また装置ドライバは、インタフェース・チャンネル識別子をアプリケーションに通知するように整えられていることが好ましく、また好ましくはそのチャンネルに割り当てられている最大速度と現在利用可能な速度とそのチャンネルを使っている接続（もしあれば）の数とそのチャンネルを使っている各接続の識別子との内の少なくとも一つを含む他のパラメータを通知するように整えられていることも好ましい。これによって、すべてのアプリケーションがインタフェースを使用するためにこのようなパラメータを取り扱う必要がなく、ある適当なアプリケーションによる通信の高度な管理が可能となる。

10

【0022】

最も好ましくは、第1、第2の好適な実施形態の両者が一緒に使用されていて、装置ドライバは、論理アドレスと論理チャンネル識別子とを参照することによって、インタフェースに取り付けられた周辺装置間の一つ以上の接続を定義するというアプリケーションからの要求を受け入れるように整えられる。このようなこれら二つの実施形態の組合せは、関連する周辺装置のアドレスの詳細、あるいは接続が確立されるインタフェース・チャンネルの詳細を追跡・記憶していることを必要とせずに、アプリケーションは接続を確立できるといふ相乗作用的利益をもたらす。装置ドライバは好ましくは、特定の周辺装置間の2点間接続と同報接続のうちの少なくとも一つを確立するように整えられている。

20

【0023】

インタフェース・パラメータが変化しやすいバス・リセットといったイベントの最中に通信が中断されることがある。装置ドライバは、アプリケーションからの入力を要求することなく一定のイベントを処理できるが、装置ドライバは、バスのリセット（好ましくはリセットの開始と終了とを通知する別々のイベント）とかバス・トポロジーマたはチャンネルまたは接続パラメータの変更の内の少なくとも一つを含む一つ以上のイベントをアプリケーションに通知する（アプリケーションが要求すれば）ように整えられていることが好ましい。

30

【0024】

第2の特徴において本発明は、アプリケーションと、インタフェース識別子が割り当てられている少なくとも一つの機能を有するインタフェースとの間で通信を行うための装置ドライバであって、前記一つのあるいは各々のインタフェース識別子は少なくとも一つのイベントの後で変化しやすいものである前記ドライバにおいて、それぞれのインタフェース識別子に対応する少なくとも一つの論理識別子を記憶する手段と、前記装置ドライバと前記アプリケーションとの間で対応する機能に関連する通信を指示するために前記アプリケーションに前記論理識別子を供給する手段と、前記アプリケーションと所定の論理識別子を使って指示された前記装置ドライバとの間の通信が、対応するインタフェース識別子の前記機能への割当ての変更が続いて、対応する所定の機能との関連を維持できるように、前記一つのあるいは各々の機能に割り当てられたインタフェース識別子とは独立に前記一つのあるいは各々の論理識別子と前記一つのあるいは各々の機能との間の対応を維持するための手段とを含むことを特徴とする、前記装置ドライバを提供する。

40

【0025】

前記装置ドライバは、ハードウェア例えば専用の集積回路に実現することができ、これによって動作速度を向上させることができる。しかしながら前記装置ドライバは、少なくとも部分的にはソフトウェアに実現されることが更に好ましく、アプリケーションを実行する処理手段によって実行されることが好ましい。これによって柔軟性がより大きくなり、必要なコンポーネントの数が減少し、装置ドライバは、更に容易に更新可能となる。

50

## 【 0 0 2 6 】

第 3 の特徴において本発明は、アプリケーションを実行するための実行時エンジン手段と、少なくとも一つの装置に接続するためのインタフェース手段において、前記インタフェースはインタフェース識別子が割り当てられている少なくとも一つの機能を有しており、前記一つのあるいは各々のインタフェース識別子が少なくとも一つのイベントの後で変化しやすいものである、前記インタフェース手段と、装置ドライバ手段とを含むデータ処理システムであって、前記装置ドライバ手段は、それぞれのインタフェース識別子に対応する少なくとも一つの論理識別子を記憶する手段と、前記装置ドライバとアプリケーションとの間に対応する機能に関連する通信を指示するために前記アプリケーションに前記論理識別子を供給する手段と、前記アプリケーションと所定の論理識別子を使って指示された前記装置ドライバとの間の通信が、対応するインタフェース識別子の前記機能への割当ての変更が続いて、対応する所定の機能との関連を維持できるように、前記一つのあるいは各々の機能に割り当てられたインタフェース識別子とは独立に前記一つのあるいは各々の論理識別子と前記一つのあるいは各々の機能との間の対応を維持するための手段とを含むことを特徴とする、前記データ処理システムを提供する。

10

## 【 0 0 2 7 】

前記第 1 の特徴の好適な機能は、前記第 2、第 3 の特徴に適用することができる。

## 【 0 0 2 8 】

前記データ処理システムは、同報データを（衛星またはケーブル経由で）受信する手段を含む受信器／復号器（例えばセット・トップ・ボックス）に実現されることが好ましく、そのインタフェースは、受信データの少なくとも一部の表示用または記憶用のデジタル・ビデオレコーダまたはデジタル表示装置またはコンピュータへの接続用に整えられていることが好ましい。前記装置ドライバは、前記インタフェースに渡すために変更されたデータ・ストリームを生成するために前記受信したデータ・ストリームを変更する装置手段と協同動作するように整えられることが好ましい。

20

## 【 0 0 2 9 】

インタフェースは、I E E E 1 3 9 4 規格、またはその修正版、改良版、または変形版に従うことが好ましい。データは、I E E E 1 8 8 3 規格に従ってトランスポートすることができる。

## 【 0 0 3 0 】

アプリケーションは、解釈された言語で実行されることが好ましく、装置ドライバは、コンパイルされることが好ましい。

30

## 【 0 0 3 1 】

本発明は、アプリケーションが例えば I E E E 1 3 9 4 バス上でデジタル・ビデオレコーダと通信することを可能にするための受信器／復号器に採用されることが最も好適である。装置ドライバは、アプリケーションからデジタル・ビデオレコーダを制御するためのコマンドを送信するように、および／またはデジタル・ビデオレコーダに記録された情報に関するデータを受信するように整えることができ、このようにして、受信器／復号器内で動作する対話型アプリケーションは、プログラムあるいはその他のデータの記録とプレーバックを制御できる。送信されるデータは、M P E G フォーマット（これは基本 M P E G フォーマットの如何なる変形または発展形をも意味する）データであることが好ましいが、他のフォーマットでも使用できる。

40

受信器／復号器の原理

## 【 0 0 3 2 】

本発明を具体化した装置ドライバを説明する前に、好適なプラットフォーム、デジタル衛星受信器／復号器の基本機能を簡単に説明する。

## 【 0 0 3 3 】

図 1 を参照すれば、本実施例の装置ドライバが設置を意図しているデジタル対話型テレビジョン・システムでの使用のための受信器／復号器 2 0 2 0 あるいはセット・トップ・ボックスが概略図的に描かれている。適当なデジタル対話型テレビジョン・システム

50

の詳細については、例えば、筆者らの同時係属中の出願 P C T / E P 9 7 / 0 2 1 0 6 から 0 2 1 1 7 に見出すことができる。装置ドライバの機能の理解を促すために、以下、受信器 / 復号器の基本構成を要約する。

【 0 0 3 4 】

図 1 を参照すれば、受信器 / 復号器 2 0 2 0 は、数個のインタフェースを含んでおり、明確に言えば M P E G 信号フロー用のチューナー 4 0 2 8 と、シリアル・インタフェース 4 0 3 0 と、パラレル・インタフェース 4 0 3 2 と、1 個はシステムの一部を構成するスマートカード用であり、1 個は銀行カード用または他のスマートカード（支払いやホームバンキングその他に使われる）用である 2 個のカード・リーダー 4 0 3 6 とを含んでいる。またこの受信器 / 復号器は、ユーザがテレビジョン信号（プログラム）制作者に好み等を応答指示できるように、テレビジョン信号制作者へのモデム通信バック・チャンネル 4 0 0 2 に対するインタフェース 4 0 3 4 を含んでいる。受信器はまた、実行時エンジン 4 0 0 8 と装置マネージャ 4 0 6 8 と一つ以上のアプリケーション 4 0 5 6 を実行するための複数の装置 4 0 6 2 とを含んでいる。

【 0 0 3 5 】

この明細書の目的のために、アプリケーションは、好ましくは受信器 / 復号器 2 0 2 0 のハイレベル機能を制御するための一つのコンピュータ・コードである。例えばエンドユーザがテレビジョン受像機 2 0 2 2 の画面上で見られるボタン・オブジェクトにリモート・コントローラの焦点を位置決めして確認キーを押下すると、そのボタンに関連した命令列が実行される。

【 0 0 3 6 】

対話型アプリケーションは、メニューを提示し、エンドユーザの要求でコマンドを実行して、そのアプリケーションの目的に関するデータを提供する。アプリケーションは、常駐アプリケーション、すなわち受信器 / 復号器 2 0 2 0 の R O M （または F L A S H メモリまたはその他の不揮発性メモリ）内に記憶されたアプリケーションか、受信器 / 復号器 2 0 2 0 の R A M または F L A S H メモリに同報・ダウンロードされたアプリケーションのいずれかでもよい。

【 0 0 3 7 】

前述のアプリケーションに更に詳細に説明されたアプリケーションの幾つかの例は、下記のものである：

- ・ M P E G - 2 環境で受信器 / 復号器 2 0 2 0 を直ちに動作させ得るモジュールの適応可能な集まりである開始 ( I n i t i a t i n g ) アプリケーション。
- ・ダウンロードか常駐か、いずれかのアプリケーションが受信器 / 復号器 2 0 2 0 上で動作することを可能にする起動 ( S t a r t u p ) アプリケーション。
- ・プログラミングに関する十分な情報を与える対話型アプリケーションであるプログラム案内 ( P r o g r a m G u i d e ) 。
- ・エンドユーザが現在のイベントを買うことを可能にするデジタル T V ブーケの各 P P V チャンネル上で利用可能な対話型サービスである番組有料視聴制 ( P a y P e r V i e w ) アプリケーション。
- ・エンドユーザがこの P C ダウンロード・アプリケーションを使ってコンピュータ・ソフトウェアをダウンロードすることを可能にする P C ダウンロード ( P C D o w n l o a d ) アプリケーション。
- ・画面上のボタンを使うエンドユーザ・ナビゲーションによる映像の周期的ビデオ放送を含む雑誌閲覧 ( M a g a z i n e B r o w s e r ) アプリケーション。
- ・商品の売出しを受信器 / 復号器 2 0 2 0 に伝送してテレビジョン 2 0 2 2 上に表示可能とし、またユーザが購入する特定の品目を選択することを可能にするテレショッピング ( T e l e s h o p p i n g ) アプリケーション。

【 0 0 3 8 】

アプリケーションは、受信器 / 復号器 2 0 2 0 内の記憶場所に格納され、資源ファイルとして表現される。これらの資源ファイルは、前述の明細書に詳細に説明したように、グラ

10

20

30

40

50



フィック・オブジェクト記述ユニット・ファイルと、変数ブロック・ユニット・ファイルと、命令列ファイルと、アプリケーション・ファイルと、データ・ファイルとを含んでいる。

【0039】

MPEGデータ・ストリームにおいて各モジュールは、1群のMPEGテーブルを含んでいる。各MPEGテーブルは多数のセクションとしてフォーマット化できる。MPEGデータ・ストリームでは各セクションは、最大4キロバイトの「サイズ」を持っている。シリアル・パラレル・ポート経由のデータ転送のために例えば、モジュールは複数のテーブルとセクションとに同じように分割され、セクションのサイズはトランスポート媒体によって変化する。

10

【0040】

モジュールは、データ・ストリームのそれぞれのタイプ、例えばビデオ・データ・ストリーム、オーディオ・データ・ストリーム、およびテレテキスト・データ・ストリームの内で一般的には188バイトのデータ・パケットの形式のMPEGデータ・ストリームでトランスポートされる。各パケットの先頭には13ビットのパケット識別子(Packet Identifier)(PID)があり、MPEGデータ・ストリームでトランスポートされる各パケット毎に1個のPIDがある。プログラム・マップ・テーブル(PMTテーブル)は、異なるデータ・ストリームのリストを含んでおり、それぞれのPIDにしたがって各データ・ストリームの内容を定義している。PMTを使って識別されるPIDは、データ・ストリーム内にアプリケーションが存在することを装置に警告する。

20

【0041】

復号器は、RAMボリュームとFLASHボリュームとROMボリュームとに分割されたメモリを含んでいるが、この物理構成は論理構成とは別個のものである。メモリは更に、種々のインタフェースに関連する複数のメモリ・ボリュームに分割されることもある。メモリは、一つの見方からすれば、ハードウェアの一部と見なすこともでき、別の見方からはハードウェアとは別に示される、システム全体をサポートするか、あるいは含むものと見なすこともできる。

【0042】

システムは、仮想マシン4007の一部を形成する実行時エンジン4008を中心としていると考えることができる。これは、一方(「ハイレベル」側)ではアプリケーションに連結されており、他方(「ローレベル」側)では以下に論じる種々の中間的な論理ユニットを介して受信器/復号器ハードウェア4061に連結されている。この受信器/復号器ハードウェアは、前に論じたような種々のポートあるいはインタフェース(送受器2026用のインタフェース2030、MPEGストリーム・インタフェース4028、シリアル・インタフェース4030、パラレル・インタフェース4032、カードリーダー4036へのインタフェース、モデム通信バック・チャンネル4002へのインタフェース4034)を含んでいると考えることができる。

30

【0043】

図2を参照すれば、種々のアプリケーション4056は、ユニット4007に連結されている。より一般的に使われるアプリケーションの幾つかは、4057で示すように、システム内に多少とも永続的に常駐することできるが、その他のアプリケーションは例えばMPEGデータ・ストリームから、あるいは要求にしたがって他のポートからシステム内にダウンロードされるであろう。

40

【0044】

ユニット4007は、実行時エンジン4008に加えてツールボックス4058を含む幾つかの常駐ライブラリ機能4006を含んでいる。このライブラリは、エンジン4008によって使われるC言語の種々雑多な機能を含んでいる。これらには、データ構造の圧縮、伸長または比較、線の描画等といったデータ操作が含まれる。ライブラリ4006はまた、ハードウェア・ソフトウェアのバージョン番号と利用可能なRAM空間といった受信器/復号器2020内のハードウェア4061に関する情報と新しい装置4062をダウ

50

ンロードするときに使われる機能に関する情報とを含んでいる。各機能は、このライブラリ内にダウンロードできてフラッシュ・メモリまたはRAMメモリ内に格納される。

【0045】

実行時エンジン4008は、装置マネージャ4068に連結され、装置マネージャ4068は1組の装置4062に連結され、装置4062は装置ドライバ4060に連結され、装置ドライバ4060はポートまたはインタフェースに連結されている。広い意味で装置ドライバは、論理インタフェースを定義すると考えることができるので、二つの異なる装置ドライバは一つの共通な物理ポートに連結できる。装置は、通常二つ以上の装置ドライバに連結されるであろう。もし装置が単一の装置ドライバに連結されるのであれば、その装置は通常、通信に必要とされる全機能を組み入れるように設計されるであろうから別の装置ドライバの必要はなくなる。ある幾つかの装置は、それらの装置間で通信することができる。

10

【0046】

以下に述べるように、各装置4062から実行時エンジンまでの通信の形式には、変数とバッファと1組のイベント待ち行列に渡されるイベントとによって三つの形式がある。受信器/復号器2020の各機能は、装置4062として表される。装置は、ローカルにもリモートにもなり得る。ローカル装置4064は、スマートカードとSCARTコネクタ信号とモデムとシリアル・パラレル・インタフェースとMP EGビデオ・オーディオ・プレーヤーとMP EGセクション・テーブル・抽出器とを含む。遠隔位置で実行されるリモート装置4066は、ポートと手順とが、受信器/復号器の製造業者によって提供され、設計された装置と装置ドライバによってではなく、システム管理者またはシステム設計者によって定義されなければならないと言う点で、ローカル装置と異なっている。

20

【0047】

新しい装置4062が生成されると、その装置は、関連するアプリケーション4056を放送センターからダウンロードすることによって既存の受信器/復号器2020内にインストールすることができる。この受信器/復号器2020内へのダウンロードは、ハードウェア・ソフトウェアのバージョンをチェックし、もし正しければその新しい装置4062を表すソフトウェア・モジュールをロードし、ライブラリ4006の手順にファームウェア内(フラッシュ・メモリ内)にその新しい装置コードをインストールするように依頼するアプリケーション4056によって行われる。これによって、ソフトウェアの残りの部分に影響を与えずに受信器/復号器2020内に新しい機能の柔軟で確実なインストールを行うことができる。

30

【0048】

装置マネージャ4068は、アプリケーション4056と受信器/復号器2020の固有の機能との間の共通のソフトウェア・インタフェースである。装置マネージャ4068は、装置4062へのアクセスを制御し、予期しないイベントの受信を宣言し、また共用メモリを管理する。

【0049】

実行時エンジン4068は、マクロプロセッサと共通アプリケーション・プログラミング・インタフェースの制御の下で動作する。これらは、すべての受信器/復号器2020がアプリケーションの点から同じであるようにすべての受信器/復号器2020内にインストールされる。

40

【0050】

このエンジン4008は、受信器/復号器2020上でアプリケーション4056を実行する。これは、対話型アプリケーション4056を実行し、受信器/復号器2020の外部からイベントを受信し、図形とテキストを表示し、サービス用の装置を呼び出し、そして特定の計算のためにエンジン4008に接続されたライブラリ4006の機能を使用する。

【0051】

実行時エンジン4008は、各受信器/復号器2020内にインストールされた実行可能

50

なコードであり、アプリケーションを解釈して実行するためのインタプリタを含んでいる。本エンジン4008は、シングル・タスクのオペレーティング・システム(MS-DOSなど)を含む如何なるオペレーティング・システムにも適応可能である。本エンジン4008は、プロセス・シーケンサ・ユニット(キー押下といった種々のイベントを取り込んで種々のアクションを実行する)に基づいており、異なるハードウェア・インタフェースからのイベント待ち行列を管理するそれ自身のスケジューラを含んでいる。これはまた、図形とテキストの表示を処理する。プロセス・シーケンサ・ユニットは、1組のアクション・グループを含んでいる。各イベントは、プロセス・シーケンサ・ユニットをイベントの性格にしたがって現アクション・グループから他のアクション・グループに移動させて、その新しいアクション・グループのアクションを実行させる。

10

#### 【0052】

エンジン4008は、アプリケーション4056を受信器/復号器メモリ2028内にロードおよびダウンロードするコード・ローダーを含んでいる。最適の使用を保証するために、必要なコードだけがRAMまたはFlashメモリにロードされる。ダウンロードされたデータは、アプリケーション4056の如何なる修正をも、あるいは如何なる無認可アプリケーションの実行をも防止するために認証機構によって検証される。エンジン4008は更に、デコンプレッサを含む。アプリケーション・コード(中間コードの形式)は、メモリ空間を節約することと、MPEG-2トランスポート・ストリームからのダウンロード、または内蔵受信器/復号器モードを介してのダウンロードを高速化するために圧縮されているので、このコードはRAMにロードする前にデコンプレス(復元)しなくてはならない。エンジン4008はまた、種々の変数値を更新して状態変化を決定するためにアプリケーション・コードを解釈するインタプリタと、エラー・チェッカーとを含んでいる。

20

#### 【0053】

如何なる装置4062のサービスを使用する前にも、プログラム(アプリケーション命令列のような)が「クライアント」として、すなわち装置4062または装置マネージャ4068への論理的アクセス経路として宣言されなくてはならない。このマネージャは、この装置に対するすべてのアクセス時に参照されるクライアント番号をクライアントに与える。装置4062は、数個のクライアントを持つことができ、各装置4062のクライアントの数は、装置4062のタイプに依存して指定される。クライアントは、手順「Device-Open Channel」によって装置4062に導入される。この手順は、クライアントにクライアント番号を割り当てる。クライアントは、手順「Device-Close Channel」によって装置マネージャ4068クライアント・リストから取り出すことができる。

30

#### 【0054】

装置マネージャ4068によって与えられる装置4062へのアクセスは、同期式、非同期式どちらでも可能である。同期式アクセスのためには手順「Device:Call」が使用される。これは、直ちに利用可能であるデータをアクセスする、または所望の応答を待つことを必要としない機能をアクセスする手段である。非同期式アクセスのためには手順「Device:I/O」が使用される。これは、応答を待つことを必要とするデータ、例えば、マルチプレックスを見つけるためにチューナー周波数を走査すること、あるいはMPEGストリームからテーブルを取り戻すことを必要とするデータをアクセスする手段である。要求された結果が利用可能であると、イベントはその到着を知らせるためにエンジンの待ち行列内に入れられる。更なる手順「Device:Event」は、予期しないイベントを管理する手段を提供する。

40

#### 【0055】

上述のように実行時エンジンのメイン・ループは、種々のプロセス・シーケンサ・ユニットに連結されており、このメイン・ループが適当なイベントに出会うと、制御は一時的にこれらのプロセス・シーケンサ・ユニットの一つに移される。

#### 【0056】

50

図3を参照すれば、装置マネージャは、装置からのイベントが一時的記憶のために渡される待ち行列100を含んでいる。適当な間隔で仮想マシンは、この待ち行列に信号を送ってそこから第1の項目を抽出する。このイベント項目は、仮想マシン内の待ち行列構造101に移される。このイベント項目の優先順位にしたがって、このイベント項目は、5個の待ち行列0から4のうちの適当な一つに挿入される。イベント項目は、実行時エンジンの制御下で待ち行列セクタ・ユニット102によって待ち行列構造101から抽出される。

#### 【0057】

あるイベントが待ち行列構造101から選択されると、そのイベントはプロセス・シーケンサ・ユニット・エンジン104に渡されるが、このエンジン104はプロセス・シーケンサ・ユニット・ドライバ105と1組のプロセス・シーケンサ・ユニット106とからなっている。各プロセス・シーケンサ・ユニットは、1組の互いに連結されたアクション・グループであるので、一つのアクション・グループから次のアクション・グループへの各ステップは、一般に現アクション・グループとそのイベントの性質とに依存している。異なるプロセス・シーケンサ・ユニットは、異なるサイズと複雑さを持っており、「次の」アクション・グループが、すなわちあるイベントに応答してシステムがステップを進めるアクション・グループがそのイベントの性質にのみ依存していて現アクション・グループからは独立しているものを含んでいる。またプロセス・シーケンサ・ユニット・ブロックの右側に示すように、例えば単一のポートを介して複数の同じプロトコルを使って幾つかの別々のデータ・ストリームを取り扱うために、あるプロセス・シーケンサ・ユニットの幾つかのコピー、すなわち幾つかの同じプロセス・シーケンサ・ユニットが存在することもある。

#### 【0058】

あるイベントが選択されると、そのイベントは適当なプロセス・シーケンサ・ユニットに渡される。これは、そのプロセス・シーケンサ・ユニット上の現アクション・グループから適当な出口を選択する。この結果、適当な次のアクション・グループが選択されて、例えば装置マネージャへのメッセージの送付とか命令列の実行とかを含めてそのアクション・グループ内のアクションが実行される。プロセス・シーケンサ・ユニット内のアクション・グループはまた、イベント・メッセージを他のプロセス・シーケンサ・ユニットに送ることもできる。

#### 【0059】

ある命令列が選択されると、その命令列の識別情報が命令列セクタ107に送られる。これは、命令列メモリ108から所望の命令列を取得して、それを命令列インタプリタ109に渡し、その命令列インタプリタ109が命令列を実行する。

#### 【0060】

システムはまた、フィルタ110を含んでおり、このフィルタ110は、例えばプロセス・シーケンサ・ユニット106からイベント・タイプをロードされる。あるイベント項目が装置マネージャ内の待ち行列100から仮想マシン内の待ち行列構造101に渡されると、そのタイプまたは文字がフィルタ110内のリストと照合され、もしそれが認められないタイプであれば、それは拒否される。これによって、もし仮想マシンが取り扱えないタイプのイベントを装置マネージャがキーボードが発生させても、これらのイベントは待ち行列構造101に渡されないということが保証される。(もしこの種のイベントが待ち行列構造101に渡されたとすると、これらのイベントはその待ち行列に累積されるか、プロセス・シーケンサ・ユニット・エンジン104の誤動作を引き起こすかのどちらかになるであろう。)

#### 【0061】

このようにして、われわれの基本的受信器/復号器プラットフォームは、アプリケーションが種々の装置と通信することを可能にする際に、かなりの柔軟性を提供するということが理解できる。

〔IEEE1394バス用の装置ドライバ〕

10

20

30

40

50

## 【0062】

図4を参照すれば、IEEE 1394バス・ドライバは、アプリケーションと、IEEE 1394バスに接続されたデジタル・ビデオレコーダといった周辺装置との間の通信を容易にするために前述の方式にしたがって動作するということが理解できる。

## 【0063】

データの高速通信のためには、例えばMPEG実時間データの記憶のためには、アプリケーションによる制御にとって比較的簡単である、従来のシリアル・パラレル・インタフェースは十分に高速とは言えない。下記の装置ドライバは、アプリケーションがIEEE 1394バスを効率的にアクセスすることを可能にする多数の新しい機能を組み入れており、例えば比較的単純なアプリケーションによる、バスに接続されたデジタル・ビデオレコーダの制御を可能にすることができる。

10

## 【0064】

この装置ドライバは、コマンドを介してアプリケーションによって別々にアクセス可能である多数の機能ユニットを含んでいると考えることができる。各コマンドは、他の装置にも共通である上述の三つの標準手順の一つを用いて、装置マネージャ4068の制御下で動作する装置4062を介してアプリケーションとインタフェースする。情報は、アプリケーションと装置ドライバとの間でパラメータ・テーブルを用いて渡すこともできる。参照を容易にするために三つの基本的手順を下記に簡単に要約する：

## 【0065】

1) Device : Call このコマンドは、同期コマンドまたはデータ転送を実行するためにアプリケーションによって使うことができる。このアプリケーションの実行は、装置ドライバによる動作が完了して制御が返却されるまで停止される。これによって、厳密な順序で実行しなくてはならない動作を高い信頼度で制御することが可能になる。

20

2) Device : I/O このコマンドは、非同期動作を可能にする。すなわち、アプリケーションは、データ転送の要求または装置ドライバによって実行すべき特定の機能の要求を送出でき、またそのアプリケーションの実行は、そのデータ転送または機能が装置ドライバによって実行されている間、続行できる。

3) Device : Event このイベントをトラップする機能は、イベントが装置ドライバによってアプリケーションに通知されることを可能にし、またコードとは独立にイベントに応答してアプリケーションによって取られる特定のアクションのために、アプリケーションは、そのイベントが通知されるときに実行しており、アプリケーションは効果的に中断される。各イベントは、優先順位付けすることができる。イベントは、バス・リセットといったインタフェース上で発生するイベントを通知するために使うことができる。

30

## 【0066】

さて本発明を具体化した装置ドライバに備えられたコマンドを説明する。各コマンドは、Device : CallまたはDevice : I/O問題のいずれかを介してパラメータとしてコマンドの識別子を渡すことによってアプリケーションによってアクセスできる。下記のコマンドのすべてが備えられる必要はなく、またコマンドの機能は、変更可能である。これらのコマンドは、認められるように、独立に備えることも変更することも可能であるが、上述のコマンドによって与えられる組合せ機能から幾つかの相乗効果が得られる。

40

## 【0067】

アプリケーションによって見られるように、オプションの好適な機能と一緒に各コマンドによって与えられる特徴と機能とに関して、これらのコマンドを説明する。与えられた情報と明細書とによれば、これらの機能を実際実現することは、本技術に精通する人にとって簡単であろうし、また精密な詳細は、実施者に委ねられる。一例として各コマンドは、ソフトウェアに実現でき、好ましくはCプログラミング言語で書かれ、また好ましくはアプリケーションを実行するために使われるプロセッサ上で動作するようにコンパイルされる。しかしながら装置ドライバは、別々のプロセッサ上で実行でき、また一部または全

50

部のコマンドは専用のハードウェアによって実現できる。Call コマンドまたは I O コマンドを使って装置ドライバは、アプリケーションに情報を通知したり、あるいはそのアドレスが装置ドライバに渡されるメモリ内に格納されたパラメータ・テーブルに値を設定することによってパラメータをアプリケーションに戻したりできる。

#### 【0068】

コマンドに関する下記の機能は、装置ドライバによって実現される幾つかの潜在する機能、例えば論理周辺装置識別子と論理チャネル識別子とを取り扱う機能などを意味しており、装置ドライバは、論理周辺装置識別子と論理チャネル識別子とがそれらの対応するインタフェース機能（それぞれ物理アドレスまたはインタフェース・チャネル番号）に相関させられることを可能にする、論理周辺装置識別子と論理チャネル識別子のそれぞれのテーブルを維持する手段を組み入れているということが認められるであろう。更にバス・リセットといったイベント発生の場合に装置ドライバは、新しい物理アドレスとチャネル番号を確認するように、またアプリケーションによって見られる通り、遷移が比較的シームレスであるようにテーブルを更新するように整えられる。

10

#### 【0069】

更に、本装置ドライバは、実際にインタフェースと通信を行うための、そしてメモリ割当てと割当て解除といった必要なハウスキーピング（付帯演算）タスクを実行するための手段を含んでいることは言うまでもない。これらの機能の一部は、図5に概略的に図示してある。これらの詳細は、使用される特定の物理ハードウェアに依存するであろうが、本明細書に提示された案内に基づいて、また I E E E 1394 規格文書（その開示内容は参考のために本書に組み入れてある）の適当な部分を参照して実施するのは本技術に精通する人にとって簡単であろうから、ここでは説明しない。

20

(Command: Bus 1394 Set)

#### 【0070】

このコマンドは、基本的インタフェース・パラメータがアプリケーションによって設定されることを可能にし、好ましくは割り当てるべきデータ受信バッファのサイズとインタフェースを介して非同期コマンドを送出するときに使われる通信再試行の回数とがアプリケーションによって設定されることを可能にする。これらのパラメータは、前もって設定でき、コマンドは省略することも可能であるが、このコマンドを備えることは、通信が、異なるアプリケーション用に最適化されることを可能にする。このようなコマンドは、非同期的に非常にうまく設定できるが、装置パラメータが安定化された後にだけ後続のアプリケーション・コマンドが実行されるように、Call 方式を介してこのコマンドにアクセスすることが好ましいということが分かっている。このコマンドは好ましくは、装置ドライバが周辺装置からデータを受信する過程にある場合にエラーをアプリケーションに通知する。

30

(Command: Bus 1394 Info)

#### 【0071】

このコマンドは、バス・トポロジに関する基本的情報をアプリケーションに返す。このコマンドは、より時間的にクリティカルでないので、I O コマンドを介して非同期的にアクセスされることが好ましい。

40

#### 【0072】

好ましくは、このコマンドおよび本当にすべての、あるいは少なくとも幾つかの非同期コマンドは、応答（またはあるコード、例えば最大時間を意味しないゼロ）に必要とされる最大時間（例えばミリ秒単位で）を経過するように整えられる。これによって装置ドライバが要求を優先順位付けすることが可能になる。

#### 【0073】

このコマンドは好ましくは、バスによって管理される最大データ速度と、呼出しの瞬間に利用可能なデータ速度（すなわちバス上で既にアクティブな接続を考慮に入れて）と、そのバスに物理的に接続された周辺装置の数およびそれらに対応する論理識別子（更に後で論じられる）と、その呼出しの時に利用可能である論理チャネルとに関する情報を返す。

50

## 【 0 0 7 4 】

I E E E 1 3 9 4 バスによれば、このバスに接続された各周辺装置は、ときどき変化する可能性のある物理アドレスを割り当てられる。

## 【 0 0 7 5 】

このコマンドを特に備えることは任意であるが、各実行ごとにアプリケーションが、単一の論理アドレスを使って、対応する周辺装置を一意的に明白に識別できるように、各周辺装置に関して一定である論理アドレス（論理周辺装置識別子とも呼ばれる）のテーブルを装置ドライバが維持する（所定のアプリケーションに関して所定のセッションの間；もし受信器／復号器がリセットされればこれらの論理アドレスは変化する可能性がある）ことが望ましい。チャンネルに割り当てられたチャンネル番号もまた変化する可能性があるので、装置ドライバもまた、論理チャンネル番号のテーブルを保守する。それから装置ドライバは、該当するテーブルからデータを調べることによって簡単に情報要求に応答することができる。

10

## 【 0 0 7 6 】

好ましくは、チャンネルの利用可能性に関する情報は、2進形式でビットマップとして、好ましくは、各ビットが64個の論理チャンネルの一つの利用可能性を符号化する情報の8バイトとして渡される（例えばチャンネルが既に割り当てられていることを意味する「0」とチャンネルが使用可能であることを意味する「1」）。

( Command : Bus 1394 Info Periph )

## 【 0 0 7 7 】

20

このコマンドは、論理周辺装置識別子を示すパラメータを受信するように、またインタフェース上の周辺装置に割り当てられた物理アドレスに対応する2バイトの物理アドレス（ノードIDとしても知られている）を返すように、また周辺装置をグローバルに好ましくは一意的に識別する、あるいは少なくとも周辺装置の販売業者番号またはモデル番号を識別する8バイトの一意ノード識別子を返すように、整えられる。これによって適切に高度なアプリケーションに例えば特定の周辺装置を識別する情報に基づいて装置の固有の能力を決定する能力が与えられる。

## 【 0 0 7 8 】

このコマンドは好ましくは、インタフェースが機能的なI E E E 1 3 9 4 バスに物理的に接続されていない場合か、論理周辺装置識別子が無効である（例えば、前もって決められた最大値、好ましくは63よりも大きい）場合にエラーを通知するように、また指定された論理周辺装置識別子が分からない場合、または装置が指定された時間内に応答できなかった場合にバス・リセット保留中というエラーを通知するように、整えられる。

30

## 【 0 0 7 9 】

このコマンドは好ましくは、Device : I / O手順を用いて非同期的にアクセスされ、完了または不成功を示す信号はパラメータ・ブロックを用いて渡される。

( Command : Bus 1394 Alloc Channel )

## 【 0 0 8 0 】

このコマンドは、好ましくは所望の通信速度と好ましくは使用する所望のインタフェース・チャンネルとを指定するチャンネル割当ての要求を受信するように整えられる。前もって決められたコード（例えばOFFh）は、特定のインタフェース・チャンネルではなく、その場合あるいは所望のチャンネルが占有されている場合は、装置ドライバは利用可能なチャンネルを割り当てるということを示すために使われる。

40

## 【 0 0 8 1 】

このコマンドは、成功の場合、割り当てられた論理チャンネル識別子を返し、好ましくはBus - 1394 - Info - Periphコマンドに関して前述の該当する場合、あるいは利用可能なチャンネルが無い場合、あるいは要求されたデータ速度が利用可能な最大速度より高い場合にはエラーを通知する。

## 【 0 0 8 2 】

例えば非常に限定された数のチャンネルを使う、装置ドライバの単純化された実施形態では

50

、このコマンドと次に説明する関連する二つのコマンドとは、幾らか柔軟性を犠牲にして省略することができる。

【0083】

このコマンドは好ましくは、Device: I/O手順によって非同期的にアクセスされ、完了または不成功を示す信号がパラメータ・ブロックによって渡される。

(Command: Bus 1394 Info Channel)

【0084】

このコマンドは、指定された論理チャネルの特性に関する情報をアプリケーションに返すように整えられている。好適にもこのコマンドは、チャネルに割り当てられた最大速度 (Kbit/s 単位で) と、呼出し時にチャネル経由で利用可能な速度と、実チャネル識別子 (すなわち、装置ドライバによってではなくインタフェースによって割り当てられたもの) と、そのチャネルを使う接続の数と、そのチャネルを使う各接続の論理識別子とを返す。

【0085】

このコマンドは好ましくは、指定されたチャネル番号が割り当てられない場合、無効識別子の場合、バス・リセット保留中の場合、あるいはそのインタフェースが物理的に接続されていない場合に、エラーを通知する。

【0086】

このコマンドは好ましくは、Device: I/O手順によって非同期的にアクセスされ、完了または不成功を示す信号がパラメータ・ブロックによって渡される。

(Command: Bus 1394 Free Channel)

【0087】

このコマンドは、パラメータとして指定された論理チャネルに関する接続を切り離すことによって通信用のチャネルを解放する (しかし接続識別子を割当て解除しないことが好ましい)。このコマンドは好ましくは、非同期的に動作して、あるイベントによって選択されたチャネル内で通信がまだ保留中であることを通知する。

(Command: Bus 1394 Open Connect)

【0088】

このコマンドは、論理チャネル識別子と好ましくは接続タイプとを指示する要求を受信するように、また二つの装置間の2点間接続を開始、または指定された接続タイプに依存して同報内または同報外接続を開始するように、整えられている。2点間接続が指定される場合には、二つの周辺装置の論理周辺装置識別子もまた、装置ドライバに渡されなくてはならない。このコマンドの変形は、物理アドレスとインタフェース実チャネル番号とを使って動作できるが、論理パラメータに基づく動作は、上述のようにアプリケーションの動作が単純化されるという利点をもたらす。

【0089】

このコマンドは、成功の場合、論理接続識別子を返す。

【0090】

単純化された実施形態は、指定すべき2点間接続を定義する能力を省略できる。典型的なアプリケーションでは、バスに接続されたデジタル・ビデオレコーダといった単一の装置が存在するだけであるから、同報接続で十分である。

【0091】

装置ドライバの幾つかの実施形態では、特定の接続の開始は、受信器/復号器内の他の信号経路の再経路指定を自動的に誘発することがある。例えば同報内接続の開始は、デマルチプレクサーがIEEE 1394バス上で受信される着信データを処理するために利用可能であるように、デマルチプレクサー入力からフロントエンドを自動的に切り離すことがある。

【0092】

このコマンドは好ましくは、接続の本数が最大に達したとき、あるいは他のコマンドに関して上述の他の該当ケースの場合に、エラーを通知する。



## 【0093】

このコマンドは好ましくは、Device: I/O手順によって非同期的にアクセスされ、完了または不成功を示す信号が、あるイベントによって渡される。

(Command: Bus 1394 Close Connect)

## 【0094】

このコマンドは、論理接続識別子を受信し、その接続上の通信を停止させ、その後に再使用のためにその接続識別子を解放する。

## 【0095】

接続の開始時に受信器/復号器内で信号が自動的に再経路指定される場合には、この装置は好ましくは、その接続を閉じる際、または最後の関連接続を閉じる際に、接続を以前の状態に戻す。例えばデマルチプレクサ入力は、接続中の最後の同報を閉じる際にフロントエンドに再接続されることがある。

10

## 【0096】

このコマンドは好ましくは、Device: I/O手順によって非同期的にアクセスされ、完了または不成功を示す信号が、あるイベントによって渡される。

(Command: Bus 1394 List Connect)

## 【0097】

このコマンドは、呼出し時に利用可能な、好ましくは接続の本数と、各接続に関して論理接続識別子と、接続のタイプ(2点間、同報内、同報外)を示すフラグとを含むリストの形式の、復号器それ自身に関連している、アクティブな接続のリストを返す。

20

## 【0098】

このコマンドおよび/または以下に述べるコマンドは、単に単純な接続が提供される場合には、装置の単純な実施形態では省略可能である。しかしながらこのようなコマンドを備えることによって、アプリケーションがそれ自身で確立した接続を監視できるばかりでなく、二つ以上のアプリケーションが同時にその装置ドライバを使用できる場合には、他のアプリケーションによって確立された接続をも監視でき、また何れかの接続が不意に閉じられたかどうかを監視できるようになる。

## 【0099】

このコマンドは好ましくは、Device: Call手順によって同期的にアクセスされるが、これは、接続は頻繁に変化しやすく、またアプリケーションは、そうでなければ古くなった情報に基づいて通信を制御したり、さもなければ装置ドライバから応答のポーリングを要求したりしようとする可能性があるからである。

30

(Command: Bus 1394 Info Connect)

## 【0100】

このコマンドは、論理接続識別子を受け入れて、接続が確立された論理チャンネル番号を返す。このコマンドは好ましくは、接続のタイプの表示をも返し、また2点間接続の場合には関連する周辺装置の論理アドレスを返す。

List-Connectコマンドと同様に、このコマンドは好ましくは同期的にアクセスされる。

(Command: Bus 1394 Reset)

40

## 【0101】

このコマンドは、バス・リセット手順を起動するか、あるいはバス・リセットが既に保留中であればエラーを返す。このコマンドは、アプリケーションがリセット直後にIEEE 1394バスの制御を捕捉することを可能にするために使用でき、また好ましくは同期的にアクセスされる。更に以下に論じられるが、装置ドライバは好ましくは、あるイベントによってバス・リセットの完了を通知する。

(Command: Bus 1394 Send FCP)

## 【0102】

このコマンドは特に、省略してもよく、異なるように実現してもよい。下記の説明は、IEEE 1394バス上で非同期的にデータを送るための構成の一例についてのものである

50

。

#### 【 0 1 0 3 】

このコマンドは、コマンドとして非同期的に送られるメッセージ、あるいは I E E E 1 3 9 4 バス上の周辺装置に応答するメッセージを含むパラメータ・ブロックを受信する。このパラメータ・ブロックは好ましくは、メッセージのタイプと、応答用に割り当てられるべきバッファのサイズと、宛て先周辺装置の論理周辺装置識別子と、メッセージの長さ、メッセージそのものの表示を含んでいる。

#### 【 0 1 0 4 】

このコマンドは好ましくは、成功した送信を表示し、あるいは前もって決められた回数以内の再試行で送信が不成功であった場合、または I n f o - P e r i p h コマンドに関して前に述べたような該当ケースの場合にはエラーを通知する。

10

#### 【 0 1 0 5 】

潜在的には大量のデータが転送される可能性があるので、このコマンドは好ましくは、転送が継続されている間にアプリケーションが実行を継続することを可能にするように非同期的にアクセスされる。

#### 【 0 1 0 6 】

好ましくはこのコマンドは、前もって定義された論理周辺装置識別子が指定された場合、すべての周辺装置（例えば 6 3 台）にメッセージを同報するように整えられている。

#### 【 0 1 0 7 】

装置ドライバの単純化された実施形態ではこのコマンドは、例えばディジタル・ビデオレコードにコマンドを送信するためには十分である 3 2 バイトの固定長のメッセージの伝送に限定されることもある。

20

#### 【 0 1 0 8 】

好ましくは装置ドライバは、ほぼ同時に多数の要求を受信および送信でき、また多数の応答を報告できる。しかしながら単純化された実施形態は、単一で逐次的な要求についての能力を備えているだけである。

#### 【 0 1 0 9 】

アプリケーションが装置ドライバにコマンドを送ることを可能にするこれらのコマンドに加えて、装置ドライバは、装置マネージャのイベント処理機能を介してアプリケーションにイベントを通知するように整えられている。装置ドライバは下記のイベントを実施する：

30

( E v B u s 1 3 9 4 R c v F C P )

#### 【 0 1 1 0 】

このイベントは、周辺装置からの F C P フレームの受信を通知して、発信元周辺装置の論理アドレスと、メッセージのタイプと長さ、内容とを含むパラメータ・ブロックを供給する。

( E v B u s 1 3 9 4 C h a n n e l )

#### 【 0 1 1 1 】

このイベントは、チャンネル割当てとチャンネル割当て解除とを知らせて、どのチャンネルが割り当てられたか、好ましくは I n f o コマンドに関して前に説明したように 2 進形式で符号化されたかを知らせるリストを渡す。

40

( E v B u s 1 3 9 4 C o n f i g )

#### 【 0 1 1 2 】

このイベントは、周辺装置の接続と切り離しとを知らせ、また接続された周辺装置の数とそれらの論理アドレスとを含むリストを供給する。

#### 【 0 1 1 3 】

このイベントと前述の C h a n n e l イベントとに関するインタフェース上の変化は、装置ドライバがこのようなイベントをアプリケーションに知らせない場合でも更新された論理識別子とインタフェース識別子との間に対応テーブルを保持するために装置ドライバによって監視しなくてはならないと言うことは認められるであろう。

50

( E v B u s 1 3 9 4 C o n n e c t )

【 0 1 1 4 】

このイベントは、接続の切り離しを知らせるために使われ、また切り離された接続のアプリケーションに論理識別子を供給し、また好ましくは I n f o - C o n n e c t i o n コマンドに関して前に説明したものと類似のフォーマットでその切り離された接続に関する更なる情報を含むリストを供給する。

( E v B u s 1 3 9 4 L o E v e n t s )

【 0 1 1 5 】

このイベントは、一つ以上のローレベル・インタフェース・エラー、例えば周辺装置が許可されたよりも長い間バスを保持していることとか、データ・エラーや C R C エラーとか、予期しないトランザクションとか、未知のチャンネル番号やトランザクション・コードなどを知らせることができる。このイベントは、主としてデバッグ用に有用であり、また装置ドライバの単純化された実施形態では省略してもよい。

( E v B u s 1 3 9 4 H i E v e n t s )

【 0 1 1 6 】

このイベントは、バス・リセットの開始と完了の内の少なくとも一つ（好ましくは両方）と、ケーブルの電源障害またはバス内でのループの検出または装置ドライバが多数回の再試行後、独力では回復できない重大障害といったイベントとを含む一つ以上のハイレベル・バス状態を知らせることができる。

( E v B u s 1 3 9 4 O f f )

【 0 1 1 7 】

更なるイベントとしてこのイベントは、受信したメッセージを格納すべき利用可能なバッファを持たないといった、装置ドライバにとって内部的なエラーを知らせるために使用できる。

【 0 1 1 8 】

上述のコマンドとイベントは、単に例示的なものであって、本発明は種々の仕方で実現可能であり、また特に幾つかのコマンドは類似の機能を実行する他のコマンドと組み合わせることが可能であり、あるいはあるコマンドは単純化された実施形態では省略することもできる。これらの機能の各々のハードウェアとソフトウェアの実施形態は、複数のコマンド間でも単一のコマンド内でも自由に組み合わせることが可能である。ハードウェア実施形態は、より高速に動作して処理能力を解放でき、これに対してソフトウェア実施形態はより容易に更新できる。ハードウェア、コンピュータ・ソフトウェアなどによって実行されるこれらの機能は、電気などの信号について、あるいはそれらの信号を使って実行される。ソフトウェア実施形態は、ROM また F L A S H に格納することができ、あるいは F L A S H 内でパッチすることもできる。

【 0 1 1 9 】

本発明は、純粋に例として今まで説明してきたものであり、本発明の範囲内で細部の修正が実施可能であることは理解されるであろう。

【 0 1 2 0 】

本説明で開示された各機能と、（適当な場合に）特許請求項と図面とは、それぞれ独立に、あるいはいずれでも適当に組み合わせて設けることも可能である。

【図面の簡単な説明】

さて本発明の好適な機能は、下記の付属図面を参照して純粋に例として説明される：

【図 1】図 1 は、受信器 / 復号器のインタフェースの概略図であり、

【図 2】図 2 は、受信器 / 復号器の機能ブロック図であり、

【図 3】図 3 は、更に詳細な仮想マシンと実行時エンジンの幾つかのコンポーネントを示しており、

【図 4】図 4 は、装置ドライバを経由するアプリケーションと遠隔周辺装置との間の通信の流れを説明するための概略図であり、

【図 5】図 5 は、装置ドライバの幾つかのコンポーネントを示す概略図である。

10

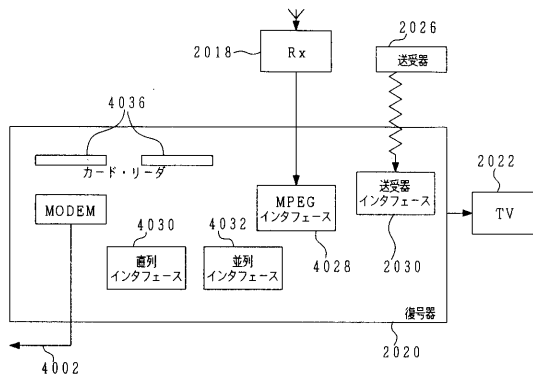
20

30

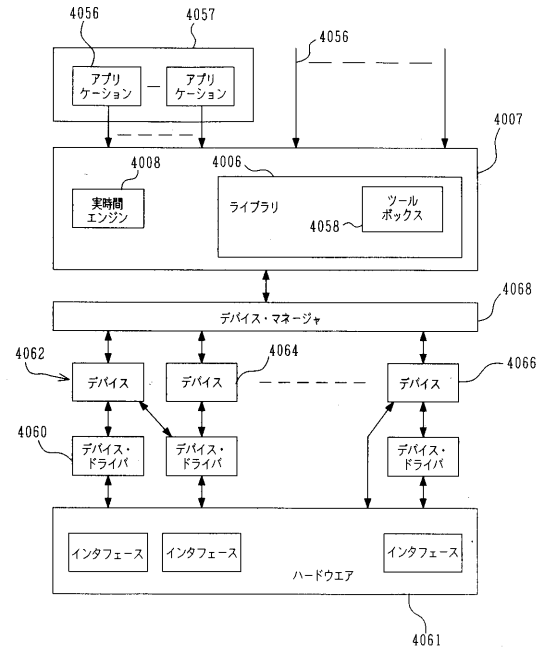
40

50

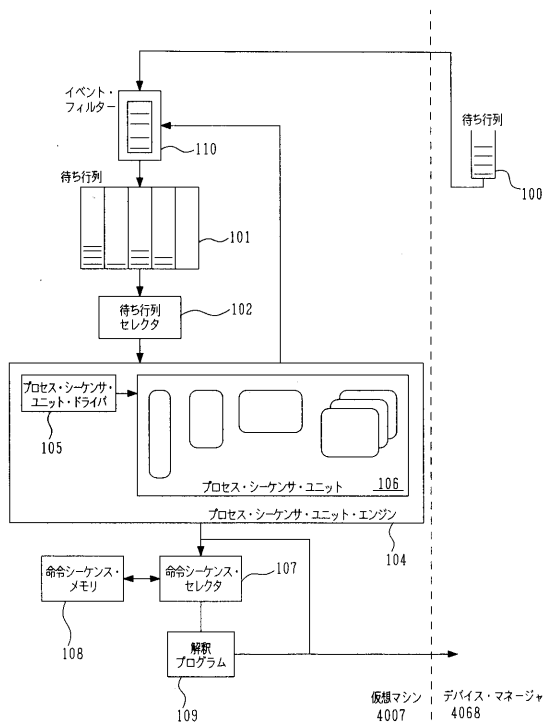
【図 1】



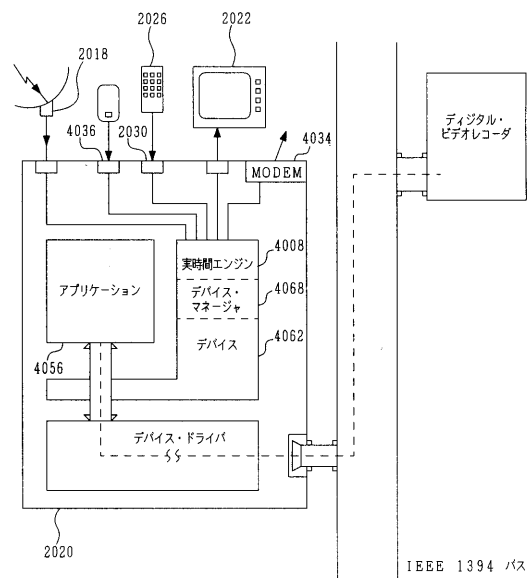
【図 2】



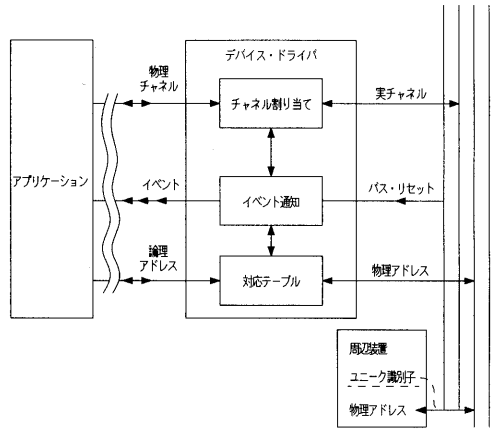
【図 3】



【図 4】



【図 5】



---

フロントページの続き

審査官 石井 茂和

- (56)参考文献 特開平 0 6 - 2 4 4 8 4 9 ( J P , A )  
特開昭 6 2 - 2 8 7 3 5 5 ( J P , A )  
特開昭 6 3 - 2 5 3 4 5 8 ( J P , A )

(58)調査した分野(Int.Cl. , D B 名)

G06F 13/14

G06F 13/38

H04L 29/02

H04N 5/44

WPI(DIALOG)

JSTPlus(JDream2)