

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4612821号  
(P4612821)

(45) 発行日 平成23年1月12日 (2011. 1. 12)

(24) 登録日 平成22年10月22日 (2010. 10. 22)

(51) Int. Cl.

F I

H O 4 L 29/06 (2006. 01)

H O 4 L 13/00 3 O 5 C

H O 4 L 12/56 (2006. 01)

H O 4 L 12/56 3 O O D

請求項の数 11 (全 26 頁)

(21) 出願番号 特願2004-264645 (P2004-264645)  
 (22) 出願日 平成16年9月10日 (2004. 9. 10)  
 (65) 公開番号 特開2006-81033 (P2006-81033A)  
 (43) 公開日 平成18年3月23日 (2006. 3. 23)  
 審査請求日 平成19年9月10日 (2007. 9. 10)

(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 國分 孝悦  
 (72) 発明者 川島 淳  
 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 森村 和彦  
 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 森川 健一  
 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

最終頁に続く

(54) 【発明の名称】 通信制御装置及び方法

(57) 【特許請求の範囲】

【請求項 1】

送信データを送信するためのフレーム個数及びフレーム長を算出する算出手段と、  
 前記算出手段による算出の結果を基に送信データをフレーム単位に分割するペイロード  
 分割手段と、

前記ペイロード分割手段による送信データの分割と並列して、前記算出手段による算出  
 の結果を基に前記フレーム単位の送信データに付加すべきヘッダを生成するヘッダ生成手  
 段と、を有し、

前記ヘッダ生成手段は、複数レイヤのプロトコルのヘッダを並列して生成することを特  
 徴とする通信制御装置。

【請求項 2】

前記算出手段は、データの送信要求に基づき前記フレーム個数及び前記フレーム長を算  
 出し、

前記ヘッダ生成手段は、前記ペイロード分割手段が送信データを入力している間に送信  
 データに付加すべきヘッダを生成することを特徴とする請求項 1 記載の通信制御装置。

【請求項 3】

前記ペイロード分割手段は、フレーム単位の送信データを加算してフレーム毎のチェッ  
 クサムを算出し、

前記ヘッダ生成手段は、前記ヘッダ内に前記チェックサムを付加することを特徴とする  
 請求項 1 又は 2 記載の通信制御装置。

## 【請求項 4】

前記ペイロード分割手段は、複数フレームの送信データを入力している間でも 1 フレーム分の送信データを入力して分割する毎にフレーム単位の分割完了通知を出力し、

前記ヘッダ生成手段は、前記フレーム単位の分割完了通知を基に前記フレーム単位のヘッダを生成することを特徴とする請求項 1 ~ 3 のいずれか 1 項に記載の通信制御装置。

## 【請求項 5】

前記ヘッダ生成手段は、前記ペイロード分割手段が複数フレームの送信データを入力している間でも各フレームのヘッダを生成する毎にフレーム単位のヘッダ生成完了通知を出力し、

さらに、前記フレーム単位のヘッダ生成完了通知を基に前記フレーム単位の送信データ及びヘッダを合成するヘッダ/ペイロード合成手段を有することを特徴とする請求項 1 ~ 4 のいずれか 1 項に記載の通信制御装置。

10

## 【請求項 6】

さらに、フレームの生成に必要なヘッダ生成情報を収集し、ヘッダ生成指示及びヘッダ生成情報をフレーム毎に出力するプロトコル制御手段を有し、

前記ヘッダ生成手段は、前記プロトコル制御手段から前記ヘッダ生成指示及び前記ヘッダ生成情報を入力すると、ヘッダを生成し、ヘッダ生成完了通知を出力することを特徴とする請求項 5 記載の通信制御装置。

## 【請求項 7】

さらに、前記ヘッダ/ペイロード合成手段により合成されたフレームを送信するフレーム送信手段を有することを特徴とする請求項 5 又は 6 記載の通信制御装置。

20

## 【請求項 8】

さらに、前記ペイロード分割手段により分割されたフレーム単位の送信データを格納するためのメモリを有することを特徴とする請求項 7 記載の通信制御装置。

## 【請求項 9】

さらに、前記メモリからフレーム単位の送信データを読み出して前記ヘッダ/ペイロード合成手段に出力するペイロード読み出し手段を有することを特徴とする請求項 8 記載の通信制御装置。

## 【請求項 10】

前記ヘッダ生成手段は、

T C P ヘッダを生成する T C P ヘッダ生成手段と、

U D P ヘッダを生成する U D P ヘッダ生成手段と、

I P ヘッダを生成する I P ヘッダ生成手段と、

M A C ヘッダを生成する M A C ヘッダ生成手段とを有し、

前記プロトコル制御手段は、前記 T C P ヘッダ生成手段、前記 U D P ヘッダ生成手段、前記 I P ヘッダ生成手段及び前記 M A C ヘッダ生成手段にヘッダの生成指示を出力することができることを特徴とする請求項 6 ~ 9 のいずれか 1 項に記載の通信制御装置。

30

## 【請求項 11】

通信制御装置が行う通信制御方法であって、

送信データを送信するためのフレーム個数及びフレーム長を算出する算出ステップと、

前記算出ステップにおける算出の結果に基づいて、送信データをフレーム単位に分割する分割ステップと、

40

前記分割ステップにおける送信データの分割と並列して開始されるステップであり、前記算出ステップにおける算出の結果に基づいて、前記フレーム単位の送信データに付加すべきヘッダを生成するヘッダ生成ステップと、を有し、

前記ヘッダ生成ステップでは、複数レイヤのプロトコルのヘッダを並列的に生成することを特徴とする通信制御方法。

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

50

本発明は、通信制御装置及び方法に関するものである。

【背景技術】

【0002】

パーソナルコンピュータ等でなくとも、機器を直接ネットワークに接続する機会や需要が高まっている。例として携帯端末、プリンタ、カメラ、コピー機、ディスプレイ、ビデオ機器、音響装置などがあげられる。これらの機器は比較的低価格・低処理能力のCPUを搭載、又は全く搭載していない場合もあり、その様な機器資源を利用しネットワークにおいて標準的に使用されているTCP/UDP/IPといったインターネット標準プロトコルを実現しようとすれば、TCP/UDP/IPのプロトコルスタックを低処理能力CPUで低ビットレート（帯域）ではあるが実現するか、又はTOE（TCP/IPオフロードエンジン）といったプロトコル処理に特化した補助的デバイスを利用する事で比較的広帯域なネットワーク通信を実現するか、の方法しかない。とはいったものの機器が扱う情報の量は写真、動画、音楽、いずれにおいても大きくなる一方で、機器のコストを低価格に抑える必要がある、といった背景においてはTOEによるネットワーク通信機能実現のアプローチが今後増加してくることは想像に難くない。

10

【0003】

TOE実現（TCP/IPプロトコルスタックのハードウェアによる実現）において、最も簡単にこれを実現する為には、ただ単に今迄ソフトウェアで実現していた方法をそのままハードウェア化すればよいと言う考え方がある。ソフトウェアで実現されているTCP/UDP/IPプロトコルスタックは十分に枯れた方法により実現されており、この考え方は1つの妥当な考え方ではある。しかしソフトウェア（CPU向け）に考案された方法をそのままハードウェア化したときに、それがゲートやレジスタ（フリップフロップ）やメモリといったプリミティブなデバイスで実現される事を考えるとソフトウェアの実現方法がそのままハードウェアの適切な実現方法であるかといえ、違うと言わざるを得ない。ハードウェアは複雑なポインタ処理や、変数の動的生成や消滅、再帰処理、多くのプロセス生成、といった処理方法は苦手であり、ソフトウェアで実現されたTCP/IPプロトコルスタックは複雑なポインタ処理と変数やメモリ領域の動的生成や消滅といった手順が到るところに存在している。

20

【0004】

従来手順をそのままハードウェアで実現すると大きく以下の流れになる。（1）アプリケーションデバイスから送信指示された一連の送信データを一時的に連続した共有メモリ領域に保存する。（2）共有メモリに保存された一連の送信データを、最も上位のプロトコル制御ブロックが参照、分割、ヘッダ追加を行い、次のプロトコル制御ブロックに共有メモリのアクセス権利と共に制御を移す。各プロトコル制御ブロックは順次、各々共有メモリの参照、分割、ヘッダ追加を行う。（3）ヘッダとペイロードを連続したメモリ空間に隣接して配置する。ペイロードの全てを連続したメモリ空間に配置できない場合は別のメモリ空間にペイロードのみを配置する。（4）ヘッダと隣接したペイロードや隣接しないペイロード、フレームの連続関係はメモリ制御情報として必要な数だけメモリ内に保存される（一般にmbufと呼ばれる）。

30

【0005】

即ち、複雑なペイロード分割とペイロードやヘッダの管理情報（mbuf）の生成後にヘッダが生成され、この処理は各プロトコルスタックで各々順次実行される。

40

【0006】

また、下記の特許文献1には、汎用データの送受信を行うプロセッサにより構成されたプロトコル汎用処理手段と、ヘッダ情報を固定的に定め又は定型的に変化させたヘッダ情報と送信データとを結合しパケットを構成する専用のプロトコル専用処理手段と、送信データによってプロトコル処理手段を適宜選択する通信制御装置が記載されている。

【0007】

また、下記の特許文献2には、通信ネットワークから受信したパケットを格納する第1受信バッファと、第1受信バッファに格納されたパケットの一部である第1部分データに

50

基づき、第1の処理を行う第1処理回路と、第1の処理と少なくとも一部を並行して、第1受信バッファに格納されたパケットにおける、第1部分データとは異なる部分である第2部分データを、第1受信バッファから受け取って格納する第2受信バッファと、第2受信バッファに格納された第2部分データに基づき、第1の処理より長い処理時間を要する第2の処理を行う第2処理回路とを備える半導体回路デバイスが記載されている。

【0008】

また、下記の特許文献3には、選択された分割データに対するプロトコル制御情報を生成して、分割データの先頭前方又は末尾後方に書き込み、パケットを生成し、プロトコル制御情報及び分割データをまとめて1パケットとして送信するデータ伝送装置が記載されている。

10

【0009】

【特許文献1】特開2000-59463号公報

【特許文献2】特開2004-48392号公報

【特許文献3】特開平11-215204号公報

【発明の開示】

【発明が解決しようとする課題】

【0010】

しかしながら以下の問題が発生する。(1)ハードウェアが元々持つ並列処理能力を發揮できない。また(2)メモリ管理方法もポインタ処理が多用されていてハードウェアで処理しにくい。

20

【0011】

本発明の目的は、ハードウェアが持つ並列処理能力を有効活用することにより、高速なフレーム生成を実現することである。

【課題を解決するための手段】

【0012】

本発明の通信制御装置は、送信データを送信するためのフレーム個数及びフレーム長を算出する算出手段と、前記算出手段による算出の結果を基に送信データをフレーム単位に分割するペイロード分割手段と、前記ペイロード分割手段による送信データの分割と並行して、前記算出手段による算出の結果を基に前記フレーム単位の送信データに付加すべきヘッダを生成するヘッダ生成手段と、を有し、前記ヘッダ生成手段は、複数レイヤのプロトコルのヘッダを並列して生成することを特徴とする。

30

【0013】

また、本発明の通信制御方法は、通信制御装置が行う通信制御方法であって、送信データを送信するためのフレーム個数及びフレーム長を算出する算出ステップと、前記算出ステップにおける算出の結果に基づいて、送信データをフレーム単位に分割する分割ステップと、前記分割ステップにおける送信データの分割と並列して開始されるステップであり、前記算出ステップにおける算出の結果に基づいて、前記フレーム単位の送信データに付加すべきヘッダを生成するヘッダ生成ステップと、を有し、前記ヘッダ生成ステップでは、複数レイヤのプロトコルのヘッダを並列的に生成することを特徴とする。

40

【発明の効果】

【0014】

本発明によれば、ハードウェアが持つ並列処理能力を有効に活用し、送信データをフレーム単位に分割する処理と、複数レイヤのプロトコルのヘッダを生成する処理と、を並列して実行するので、高速なフレーム生成を実現できる。

【発明を実施するための最良の形態】

【0015】

(第1の実施形態)

図1は、本発明の第1の実施形態による通信制御装置の構成例を示すブロック図である。以下、図1を用いてその構成を説明する。

【0016】

50

## 【アプリケーションデバイス】

アプリケーションデバイス(101)は、送信要求(106)と送信データ長(107)をフレーム個数フレーム長算出部(102)に出力し、送信データ(110)をペイロード分割部(103)に出力する。アプリケーションデバイス(101)には、2つの種類が想定される。1つ目はコンスタントビットレート(単位時間あたりの情報量が一定)のデバイスである。2つ目はバリエابلビットレート(単位時間あたりの情報量が一定ではないが、単位時間辺りの情報量は明示される)のデバイスである。その他にバリエابلビットレートのデバイスからの情報を一時的にバッファリングし定期的に一定量の情報を出力する様シェーピングするデバイスである場合も想定されるがこれは1つ目の種類のデバイスと同じと見なせる。また、サンプリング周波数で予め規定された量子化ビットレートでRAW情報を出力してくるデバイスも想定できるが、これは1つ目の種類のデバイスと同様に見なすことが出来る。これら全てのアプリケーションデバイス(101)に接続される本実施形態による通信制御装置は、アプリケーションデバイス(101)とのインタフェースにおいて、送信要求(106)、送信データ長(107)、送信データ(110)を受ける事とする。

10

## 【0017】

## 【フレーム個数フレーム長算出部】

フレーム個数フレーム長算出部(102)は、アプリケーションデバイス(101)から入力された送信要求(106)をトリガに、同じく入力された送信データ長(107)を基に送信データを幾つのフレームに分割して出力すべきかを算出する。

20

## 【0018】

図22を用いてペイロード長とヘッダ数算出の流れについて詳説する。フレーム個数フレーム長算出部(102)は、アプリケーションデバイス(101)からの送信要求(106)を受信すると併せて、その一度の送信要求で送信したいデータの長さを送信データ長(107)として受け取る(ステップ301、302)。するとフレーム個数フレーム長算出部(以後FFCと呼ぶ)はフレーム数(FLM\_COUNT)、先頭フレームペイロード長(HEAD\_PAY\_LEN)、中間フレームペイロード長(MID\_PAY\_LEN)、最終フレームペイロード長(FIN\_PAY\_LEN)を算出する必要があるが、これらをまず初期化する(ステップ303)。フレームをどのような長さの単位で送信するかについてはTCP/UDPを想定した場合、インタフェースのMTU又はTCPのMSSが想定されるが、ここではまずインタフェースのMTUを取得する(ステップ304)。また、TCP/UDP/IPから形成されるフレームを扱う場合にフレームのペイロード長を算出する為には各々のプロトコルヘッダの長さを鑑みる必要があるのでIPHL(IP Header Length)及びUDPHL(UDP Header Length)及びTCPLH(TCP Header Length)を規定する(ステップ305)。このステップ305では、それぞれ基本のヘッダ長を示してあるが各プロトコルにおいてオプションヘッダを想定する場合はこの場でオプションヘッダ長まで加味したヘッダ長を規定する。次にコネクションの種別についてTCPかUDPか又はRAWであるか、を識別する。コネクション種別は予めFFC(102)にて保持しておく方法とコネクション識別子を基に導出する方法が考えられる(ステップ306)。

30

40

## 【0019】

UDPの場合は送信データ長がインタフェース層MTU - IPHL - UDPHL以下であることを確認する(ステップ311)。そうである場合は1つのフレームに与えられた送信データが全て格納可能であるのでフレーム数1、先頭フレームペイロード長=送信データ長である、とする(ステップ308)。そうでない場合は送信データを複数のフレームに分割する必要がある。この分割数や各ペイロード長を算出する作業は仮にフレームを送信していった場合にどうなるかという事を想定しながら分割作業を進めていく。まず仮の送信済みデータ長(DONE\_LEN)を0とする(ステップ312)。次に未送信データ長がフラグメントペイロード長以下であることを確認する。これは最終フラグメントであるかという判断をおこなっており、未送信データ長は送信データ長から仮の送信済みデー

50

タ長 ( D O N E \_ L E N ) を減算して求め、これと M T U - I P H L の長さを比較する等という方法が想定できる (ステップ 3 1 3)。ここまでの事前チェックの流れにおいては、初めての比較では未送信データ長がフラグメントペイロード長以下であるということは起き得ない。次にフレーム数 ( F L M \_ C O U N T ) をインクリメントする (ステップ 3 1 4)。送信済みデータ長が 0 であれば (ステップ 3 1 5)、先頭フレームであると判断されるので先頭フレームのペイロード長は M T U - I P H L - U D P H L であり、仮の送信済みデータ長は M T U - I P H L - U D P H L であると決定される (ステップ 3 1 7)。先頭フレームではない場合は中間フレームであるから中間フレーム長を M T U - I P H L、仮の送信済みデータ長 ( D O N E \_ L E N ) を  $D O N E _ L E N + ( M T U - I P H L )$  とする (ステップ 3 1 6)。これらの後再び未送信データ長がフラグメントペイロード長以下であるか、即ち最終フラグメントであるかを判断し (ステップ 3 1 3)、もし最終フラグメントであれば最終フレームは送信データ長から D O N E \_ L E N を減算する事で求められる (ステップ 3 1 8)。その後最終フラグメントフレームの分のフレーム数をカウントアップ ( F L M \_ C O U N T + + ) し (ステップ 3 1 9)、フレーム数、先頭フレームペイロード長、中間フレームペイロード長、最終フレームペイロード長が決定出力される (ステップ 3 1 0)。

10

#### 【 0 0 2 0 】

コネクション種別が R A W の場合はフレーム数は 1 つで、フレームペイロード長は M T U - I P H L の範囲内に収まるべきであり、もしそうでないとしてもその様に送信データを切り落として送信する事を想定している。又はアプリケーションデバイス ( 1 0 1 ) にエラーとして応答を送るという方法も考えられる (ステップ 3 0 8、3 0 9)。

20

#### 【 0 0 2 1 】

コネクション種別が T C P の場合はフレーム長は M S S (マキシマムセグメントサイズ) が基本となるので取得する (ステップ 3 2 0)。次に仮の送信済みデータサイズ ( D O N E \_ L E N ) をゼロに初期化し (ステップ 3 2 1)、(仮の)未送信データ長が M S S 以下であるかをチェックする (ステップ 3 2 2)。もし M S S 以下であれば与えられた送信データは次の 1 セグメントで送信完了できる事が判断できる。もし M S S より大きければフレーム数をインクリメント (ステップ 3 2 3) し、先頭セグメントであれば (この判断は D O N E \_ L E N が 0 であるか否かで行う) (ステップ 3 2 4)、先頭フレーム長は M S S であり、D O N E \_ L E N は M S S となる (ステップ 3 2 6)。再び仮の未送信データ長が M S S 以下であるかを確認し、まだ M S S より大きい場合はフレーム数をカウントアップし今度は中間フレームペイロード長を M S S とし送信済みデータ長  $D O N E _ L E N = D O N E _ L E N + M S S$  とする (ステップ 3 2 5)。再び未送信データ長が M S S 以下であるかを確認し、もし最終セグメントであると判断されれば最終フレームとして最終フレーム長を送信データ長から D O N E \_ L E N を減算した値とし (ステップ 3 2 7)、最終フレームを送信フレーム数としてカウントする (ステップ 3 2 8)。

30

#### 【 0 0 2 2 】

上述の様にフレーム数、先頭フレームペイロード長、中間フレームペイロード長、最終フレームペイロード長が決定出力される (ステップ 3 1 0)。これら 4 つの値をフレーム個数とフレーム長に関する値とし、ペイロード分離部 ( 1 0 3 ) とプロトコル制御部 ( 1 1 5 ) 及びヘッダ/ペイロード合成部 ( 1 2 0 ) に出力する。

40

#### 【 0 0 2 3 】

[ フレーム個数フレーム長算出部が期待するタイミングについて ]

本通信制御装置ではアプリケーションデバイス ( 1 0 1 ) からの送信要求 ( 1 0 6 ) と送信データ長 ( 1 0 7 ) を基に、まずフレーム個数とフレーム長の算出を行い、その後、ペイロード分割とヘッダ生成を開始する。これはペイロード分割とヘッダ生成を同時に開始する事を狙ったものである。

#### 【 0 0 2 4 】

この事について図 2 を用いて説明する。アプリケーションデバイス ( 2 0 1 ) が送信指示 ( 2 1 0 ) と送信データ長 ( 2 1 1 ) をフレーム個数フレーム長算出部 ( F F C ) ( 2

50

02)に送信(これら2つの送信は同期して同時に行われる事を想定)すると、FFC(202)では先に述べた手順でペイロード長とフレーム個数を算出しこの情報をペイロード分割部(203)とプロトコル制御部(205)に送出する(ヘッダ/ペイロード合成部(120)にも送出するが図2の説明においては省略する)。ペイロード長(212)とフレーム個数(213)は同期して同時に送信されることを想定し、ペイロード分割部(203)とプロトコル制御部(205)はほぼ同時にペイロード長(212)とフレーム個数(213)を受信することを想定している。この受信時刻を $t_1$ とする。この時刻 $t_1$ よりペイロード分割部(203)ではアプリケーションデバイス(201)からのデータ受信を開始し、ヘッダ生成部ではヘッダ生成を開始する。ペイロード分割部(203)がアプリケーションデバイス(201)からの送信データ受信を開始すると(214)、アプリケーションデバイス(201)は送信データを先頭から順次ペイロード分割部(203)に送信していく。この時アプリケーションデバイス(201)とペイロード分割部(203)は同期をとりながらデータの送受信を行っていく。例えばアプリケーションデバイス(201)からの送信データが3フレーム相当の長さであったとすれば、アプリケーションデバイス(201)からのデータの送信時間は(254)(255)(256)で示された時間かかる。アプリケーションデバイス(201)からペイロード分割部(203)にデータ送信を行う際、1フレーム目に相当する長さのデータを送信終了した時刻を $t_3$ 、2フレーム目に相当する長さのデータを送信終了した時刻を $t_4$ 、3フレーム目に相当する長さのデータを送信終了した時刻を $t_5$ とする。図ではアプリケーションデバイス(201)からペイロード分割部(203)に送信するデータのフレーム区切りにまでの所要時間を(254)(255)(256)にて示しているが、アプリケーションデバイス(201)にとってはフレームの切れ目の意識はなく、連続して(但し同期を取りながら、例えば(222)(227)(232)の様に)ペイロード分割部(203)にデータを送信する。ペイロード分割部(203)では連続して受信したアプリケーションデバイスデータをフレーム単位に分割しながらフレームペイロード格納メモリ(204)にアプリケーションデータをフレーム単位で格納していく(216)(225)(230)。一方、プロトコル制御部(205)はペイロード長(212)とフレーム個数(213)を受信すると時刻 $t_1$ より各ヘッダ生成部(206)(207)(208)(209)にヘッダ生成の依頼を出しヘッダ生成を開始する。各ヘッダ生成部ではペイロード長(212)やフレーム個数(213)が判明しているのでフレームペイロードがアプリケーションデバイス(201)から受信されるより先にヘッダ生成を進めることが出来る。ヘッダ生成を行っている期間を(240)(241)(242)(243)(244)(245)(246)(247)で示している。ここでは1つの送信データについて3つのフレーム用にヘッダを生成することを想定し図示している。ここで生成されるヘッダの長さはTCP、UDP、IP、MACヘッダのいずれも数バイトや20-30バイト程度である。これに対し1つのフレームやセグメントのペイロード長は、数百から1500バイト程度(インタフェース層がイーサネット(登録商標)の場合)と想定される。即ちアプリケーションデバイス(201)から1フレーム分の送信データを受信完了した時刻 $t_3$ に対し、ヘッダ生成部がヘッダ生成する時間は十分に短く済むはずであり、1つ目のフレームのヘッダ生成完了時刻は $t_2$ で図示される。また、2つ目3つ目のフレーム用のヘッダを生成したとしても時刻 $t_3$ 近辺又は $t_3$ よりずっと以前に、全ヘッダを生成完了している可能性が高い。即ち $t_3$ には1つ目のフレームは送信可能となる。

#### 【0025】

ところで、ここでチェックサムに関する問題が発生する。ヘッダの殆どの部分はペイロードとの依存なく生成可能だが、データグラムチェックサムやIPパケットチェックサムは送信データ(ペイロード)から算出しなければならない。但しUDPにはチェックサムを無効とするオプションがあり、この場合は図2に示す様なUDPとIPのヘッダ生成とペイロード分割の完全な独立並行処理が可能となる。

#### 【0026】

また、TCPヘッダのチェックサムを考慮した場合も本実施形態によるヘッダ生成とペ

10

20

30

40

50

イロード分割の独立並行処理がスループットに貢献する事ができる。これについて図4を用いて説明する。アプリケーションデバイス(201)から1セグメント分のデータを受信する時刻がt7とすればt7直前には1セグメントのチェックサム値は算出可能であり、予め時刻t1直後に期間(263)にて生成された1つ目のセグメントのTCPヘッダに時刻t7直前に算出されたセグメントチェックサムを追加して1つ目のセグメント用のTCPヘッダ生成を完了すればt7直後には1つ目のセグメントを送信可能となる。同様に時刻t8直前には2つ目のセグメントのチェックサム値が算出可能であり、そのチェックサム値を予め期間(264)で生成された2つ目のTCPヘッダに追加して2つ目のTCPヘッダを完成しt8直後には2つ目のセグメントが送出可能になる。さらにt9直後には3つ目のセグメントが送出可能になる。

10

【0027】

UDPヘッダのデータグラムチェックサムを考慮した場合について図3に示す。データグラム用のUDPヘッダを予め時刻t1直後に期間(275)で生成したとしても1データグラムのデータグラムチェックサムが算出されるのはデータグラム全体をアプリケーションデバイス(201)からペイロード分割部(203)が受信完了した時刻t10でありこの時点で予め生成してあったUDPヘッダにUDPデータグラムチェックサムを追加し、t10直後にUDPフレームが送出される事になる。この場合は2つ目、3つ目のフレームは既に生成完了しているので1つ目のフレーム送信直後に2つ目、3つ目のフレームを送信可能である。また、もしフレームの順番を保持して送出する必要がないのであれば、2つ目、3つ目のフレームを送出してから1つ目のUDPヘッダつきフレームを送出する事も出来る。

20

【0028】

[ペイロード分割部の分割]

ペイロード分割部(103)ではフレーム個数フレーム長算出部(102)から与えられたフレーム個数とフレーム長に関する入力(109)を基にアプリケーションデバイス(101)から与えられた送信データ(110)を分割しながらフレームペイロードとしてペイロードTXメモリ(104)に書き込み保存を行う(111)。ペイロード分割部(103)は、フレームペイロード書き込みが完了したらヘッダ/ペイロード合成部(120)に対しフレームペイロードの送信準備が出来た事を通知する(136)。

【0029】

[ペイロード分割部のチェックサム]

ペイロード分割部(103)ではフレーム個数フレーム長算出部(102)から与えられたフレーム個数とフレーム長に関する入力(109)を基にアプリケーションデバイス(101)から与えられた送信データ(110)を分割しながらフレームペイロードとしてペイロードTXメモリ(104)に書き込み保存を行う(111)際、フレームペイロードのチェックサムをフレーム毎に求め、UDPヘッダ生成部(117)及び、TCPヘッダ生成部(116)にチェックサム値を伝える(135)。

30

【0030】

[TXメモリ]

TXメモリ(104)では1つの送信要求に伴う一連の送信データを、送信要求毎、フレームペイロードFIFOにフレームペイロードの列として保存している。図5に示す様にTXメモリ(104)内部には1つの送信要求に伴い1つのFIFOが割り当てられ(340)、例えばここでは1つの送信要求による送信データが5つのフレームペイロードから成ることを示している。本図ではフレームペイロード1(先頭フレーム)からフレームペイロード5(最後尾フレーム)がFIFOに蓄えられているがフレーム単位での送信が行われる場合はこの様に5つのフレームが溜まることはない。各々のフレームペイロードはFIFOに挿入された直後にペイロード読み出し部(105)により抜き出され送信される事も考えられる。

40

【0031】

[ペイロード読み出し部]

50



ペイロード読み出し部(105)ではヘッダ/ペイロード合成部(120)からの、1つのペイロード読み出し指示(114)に対し、送信すべきフレームペイロードをTXメモリ(104)から1つ取り出し、ヘッダ/ペイロード合成部(120)へ出力する(113)。

#### 【0032】

[プロトコル制御部]

プロトコル制御部(115)ではフレーム個数フレーム長算出部(102)からフレーム個数、先頭フレームのペイロード長、中間フレームのペイロード長、最終フレームのペイロード長の4つの値(108)を受信すると、送信フレームがTCPなのかUDPなのかを判断し、フレーム送信に必要なプロトコルヘッダを生成するための情報を収集し、ヘッダ生成指示とヘッダ生成情報を各プロトコルヘッダ生成部(116~119)に同時配信する。収集する各種情報は静的に設定されている場合と動的にテーブル検索を行う場合がある。動的なテーブルとしては、コネクションに付随する宛先IPアドレス、送信元IPアドレス、宛先ポート、送信元ポートを割り出すアドレステーブルや、IPのルーティングテーブルなどがあげられる。

#### 【0033】

UDP/IPのフレーム送信時に送信データグラムが3つのフラグメントフレームに分割された場合の例を図6に示す。プロトコル制御部(115)はフレーム個数フレーム長算出部(102)からの通知により、プロトコルタイプがUDP、フレーム個数が3個、ペイロード長が先頭フレームがPL0の長さ、中間フレームがPL1の長さ、最終フレームがPL2の長さとして判断する。次にUDP、IP、MACヘッダ生成に必要な情報を収集し、各プロトコルヘッダ生成部にヘッダ生成指示と共にヘッダ生成情報を出力する(123~125)。UDPヘッダ生成部(117)に出力するヘッダ生成情報としては、送信元ポート番号、宛先ポート番号、UDPデータ長を出力(123)する。IPヘッダ生成部(118)に出力するヘッダ生成情報としてはヘッダ生成数3とヘッダ長、TOS、識別子、TTL、プロトコル、送信元IPアドレス、宛先IPアドレスを1フレーム分と、全長、フラグ、フラグメントオフセットを3フレーム分を出力(124)する。この時、先頭フレームはIPヘッダ長とUDPヘッダ長とPL0、中間フレームはIPヘッダ長とPL1、最終フレームはIPヘッダ長とPL2を加算した値が全長として出力される。MACヘッダ生成部(119)に出力するヘッダ生成情報としては、ヘッダ生成数3と宛先MACアドレス、送信元MACアドレス、タイプレングスを1フレーム分を出力(125)する。

#### 【0034】

TCP/IPのフレーム送信時に送信データグラムが3つのセグメントに分割された場合の例を図7に示す。プロトコル制御部(115)はフレーム個数フレーム長算出部(102)からの通知により、プロトコルタイプがTCP、フレーム個数が3個、ペイロード長が先頭フレームPL0、中間フレームPL1、最終フレームPL2と判断する。次にTCP、IP、MACヘッダ生成に必要な情報を収集し、各プロトコルヘッダ生成部にヘッダ生成指示と共にヘッダ生成情報を出力する。TCPヘッダ生成部(116)に出力するヘッダ生成情報としてはヘッダ生成数が3と送信元ポート番号、宛先ポート番号、シーケンス番号、確認応答番号、ヘッダ長、フラグビット、ウィンドウサイズ、緊急ポイントを3フレーム分出力(122)する。IPヘッダ生成部(118)に出力するヘッダ生成情報としてはヘッダ生成数3とヘッダ長、TOS、フラグ、オフセット、TTL、プロトコル、送信元IPアドレス、宛先IPアドレスを1フレーム分と、全長、識別子を3フレーム分出力(124)する。この時、先頭フレームはIPヘッダ長とTCPヘッダ長とPL0、中間フレームはIPヘッダ長とTCPヘッダ長とPL1、最終フレームはIPヘッダ長とTCPヘッダ長とPL2を加算した値が全長として出力される。MACヘッダ生成部(119)に出力するヘッダ生成情報としては、ヘッダ生成数3と宛先MACアドレス、送信元MACアドレス、タイプレングスを1フレーム分出力(125)する。

#### 【0035】

10

20

30

40

50

## [ TCPヘッダ生成部 ]

TCPヘッダ生成部(116)ではプロトコル制御部(115)からのヘッダ生成指示及びヘッダ生成情報に基づきヘッダ生成を開始し、ヘッダ生成完了後にヘッダ/ペイロード合成部(120)にTCPヘッダ生成完了通知(126)を行う。ヘッダ生成のフローチャートを図8に示す。ステップS801においてTCPヘッダ生成指示をプロトコル制御部(115)から受信したかを確認する。もしTCPヘッダ生成指示を受信したら、プロトコル制御部(115)からTCPヘッダ情報を取得する(ステップS802)。TCPヘッダ情報はヘッダ生成数とヘッダ生成数分の送信元ポート番号、宛先ポート番号、シーケンス番号、確認応答番号、ヘッダ長、フラグビット、ウィンドウサイズ、緊急ポイント及び擬似ヘッダ用の宛先IPアドレス、送信元IPアドレス、セグメント長(TCPヘッダ長+ペイロード長)である。取得したヘッダ生成情報から擬似ヘッダとチェックサム以外のTCPヘッダを作成する(ステップS803)。次にフレームごとのペイロードサムをペイロード分割部(103)より受信したかの判断を行い、もし受信していれば(ステップS804:Yes)、擬似ヘッダとチェックサム以外のTCPヘッダとペイロードサムからTCPのチェックサムを算出する(ステップS805)。ステップS805で算出したチェックサムをステップS806にてTCPヘッダのチェックサムフィールドに設定し、1フレーム分のTCPヘッダの生成を完了する。1フレーム分のTCPヘッダの生成が完了すると、ステップS807においてヘッダ/ペイロード合成部(120)にTCPヘッダ生成完了通知(126)を行い、ヘッダ生成数分のヘッダを生成したかの判断を行う(ステップS808)。もし、ヘッダ生成数分のヘッダを生成していない場合(ステップS808:No)は、ステップS803に戻り次のヘッダの生成を開始する。ヘッダ生成数分のヘッダを生成した場合(ステップS808:Yes)は、送信データグラムに付加しなければならないTCPヘッダを全て生成したとしてTCPヘッダ生成を終了する。TCPヘッダ生成部(116)で作成するヘッダのフォーマットを図14に、擬似ヘッダフォーマットを図15に示す。

## 【0036】

また、TCPヘッダ生成部(116)はヘッダ/ペイロード合成部(120)からの出力要求(126)を受信すると、生成した順番にTCPヘッダをヘッダ/ペイロード合成部(120)に出力(130)する。

## 【0037】

## [ UDPヘッダ生成部 ]

UDPヘッダ生成部(117)ではプロトコル制御部(115)からのヘッダ生成指示及びヘッダ生成情報に基づきヘッダ生成を開始し、ヘッダ生成完了後にヘッダ/ペイロード合成部(120)にUDPヘッダ生成完了通知(127)を行う。ヘッダ生成のフローチャートを図9に示す。ステップS901においてUDPヘッダ生成指示をプロトコル制御部(115)から受信したかを確認する。もしUDPヘッダ生成指示を受信したら、プロトコル制御部(115)からUDPヘッダ生成情報を取得する(ステップS902)。UDPヘッダ生成情報は、フレーム個数、送信元ポート番号、宛先ポート番号、UDPデータ長である。取得したヘッダ生成情報から擬似ヘッダとチェックサム以外のUDPヘッダを作成する(ステップS903)。次にフレームごとのペイロードサムをペイロード分割部(103)より受信したかの判断を行い、もし受信していれば(ステップS904:Yes)、ステップS905にてフレームごとのペイロードサムを加算する。次にステップS906にてフレーム個数分のペイロードサムをペイロード分割部(103)より受信したかの判断を行い、もし全てのペイロードサムを受信していなければ(ステップS906:No)、ステップS904に戻り次のフレームペイロード受信待ち状態になる。もし全てのペイロードサムを受信していれば(ステップS906:Yes)、ステップS907にて擬似ヘッダとチェックサム以外のUDPヘッダとペイロードサムの合計からUDPチェックサムを算出する。ステップS907で算出したチェックサムをステップS908にてUDPヘッダのチェックサムフィールドに設定し、UDPヘッダの生成を完了する。UDPヘッダの生成が完了すると、ステップS909においてヘッダ/ペイロード合成部

( 1 2 0 ) に U D P ヘッダ生成完了通知 ( 1 2 7 ) を行い、U D P ヘッダ生成を終了する。U D P ヘッダ生成部 ( 1 1 7 ) で作成するヘッダのフォーマットを図 1 6 に、擬似ヘッダフォーマットを図 1 7 に示す。

【 0 0 3 8 】

また、U D P ヘッダ生成部 ( 1 1 7 ) はヘッダ / ペイロード合成部 ( 1 2 0 ) からの出力要求 ( 1 2 7 ) を受信すると、生成した U D P ヘッダをヘッダ / ペイロード合成部 ( 1 2 0 ) に出力 ( 1 3 1 ) する。

【 0 0 3 9 】

[ I P ヘッダ生成部 ]

I P ヘッダ生成部 ( 1 1 8 ) ではプロトコル制御部 ( 1 1 5 ) からのヘッダ生成指示及びヘッダ生成情報に基づきヘッダ生成を開始し、ヘッダ生成完了後にヘッダ / ペイロード合成部 ( 1 2 0 ) に I P ヘッダ生成完了通知 ( 1 2 8 ) を行う。ヘッダ生成のフローチャートを図 1 0 に示す。ステップ S 1 0 0 1 において I P ヘッダ生成指示をプロトコル制御部 ( 1 1 5 ) から受信したかを確認する。もし I P ヘッダ生成指示を受信したら、プロトコル制御部 ( 1 1 5 ) から I P ヘッダ情報を取得する ( ステップ S 1 0 0 2 ) 。次にステップ S 1 0 0 3 にてヘッダ生成数が 2 個以上であるかの判断を行う。ヘッダ生成数が 2 個以上であるということは、T C P の場合はデータグラムがセグメント化されていることを意味し、U D P の場合はデータグラムがフラグメント化されていることを意味する。もし、ヘッダ生成数が 2 個以上でない場合は ( ステップ S 1 0 0 3 : N o ) 、ヘッダ生成情報から I P ヘッダを生成し ( ステップ S 1 0 0 4 ) 、ヘッダ / ペイロード合成部 ( 1 2 0 ) に I P ヘッダ生成完了通知を行い ( ステップ S 1 0 0 5 ) 、I P ヘッダ生成を終了する。ヘッダ生成数が 2 個以上である場合 ( ステップ S 1 0 0 3 : Y e s ) は、まずフレームごとに变化しない I P ヘッダフィールドを設定する ( ステップ S 1 0 0 6 ) 。フレームごとに变化しない I P ヘッダフィールドとは上位レイヤが T C P の場合は、バージョン、ヘッダ長、T O S、フラグ、フラグメントオフセット、T T L、プロトコル、送信元 I P アドレス、宛先 I P アドレスを示し、上位レイヤが U D P の場合はバージョン、ヘッダ長、T O S、識別子、T T L、プロトコル、送信元 I P アドレス、宛先 I P アドレスを示す ( 図 1 8 及び図 1 9 参照 ) 。次にステップ S 1 0 0 7 において先頭フレームのフレームごとに变化する I P ヘッダフィールドを設定する。フレームごとに变化する I P ヘッダフィールドとは上位レイヤが T C P の場合は、全長、識別子、ヘッダチェックサムを示し、上位レイヤが U D P の場合は、全長、フラグ、フラグメントオフセット、ヘッダチェックサムを示す ( 図 1 8 及び図 1 9 参照 ) 。フレームごとに变化する I P ヘッダフィールドの設定が終了するとステップ S 1 0 0 8 においてヘッダ / ペイロード合成部 ( 1 2 0 ) に I P ヘッダ生成完了通知を行う。次にステップ S 1 0 0 9 において次に生成する I P ヘッダでセグメント化もしくはフラグメントされた最終のフレームかの判断を行う。もし、次に生成する I P ヘッダで最終でない場合は ( ステップ S 1 0 0 9 : N o ) 、ステップ S 1 0 1 0 にて中間フレームのフレームごとに变化する I P ヘッダフィールドを設定し、ステップ S 1 0 1 1 にてヘッダ / ペイロード合成部 ( 1 2 0 ) に I P ヘッダ生成完了通知を行い、再び次に I P ヘッダで最終でないかの判断 ( ステップ S 1 0 0 9 ) を行う。ステップ S 1 0 0 9 において、もし次に生成する I P ヘッダで最終である場合はステップ S 1 0 1 2 にて最終フレームのフレームごとに变化する I P ヘッダフィールドを設定し、ステップ S 1 0 1 3 にてヘッダ / ペイロード合成部 ( 1 2 0 ) に I P ヘッダ生成完了通知 ( 1 2 8 ) を行い、送信データグラムに付加しなければならない I P ヘッダを全て生成したとして I P ヘッダ生成を終了する。

【 0 0 4 0 】

また、I P ヘッダ生成部 ( 1 1 8 ) はヘッダ / ペイロード合成部 ( 1 2 0 ) からの出力要求 ( 1 2 8 ) を受信すると、生成した順番に I P ヘッダをヘッダ / ペイロード合成部 ( 1 2 0 ) に出力 ( 1 3 2 ) する。

【 0 0 4 1 】

[ M A C ヘッダ生成部 ]

MACヘッダ生成部(119)ではプロトコル制御部(115)からのヘッダ生成指示及びヘッダ生成情報に基づきヘッダ生成を開始し、ヘッダ生成完了後にヘッダ/ペイロード合成部(120)にMACヘッダ生成完了通知(129)を行う。MACヘッダ生成部(119)はプロトコル制御部(115)からMACヘッダ生成指示を受信したら、MACヘッダ情報を取得する。MACヘッダ生成情報は、ヘッダ生成数、送信元MACアドレス、宛先MACアドレス、タイプレングスである。1つの送信データグラムがフラグメントやセグメント化によって複数のMACヘッダを生成する必要がある場合においても、そのヘッダの内容は同一であるため1つのMACヘッダ生成が完了するとMACヘッダ生成完了通知(129)をヘッダ/ペイロード合成部(120)にヘッダ生成数回出力する。MACヘッダ生成部(119)で作成するMACヘッダのフォーマットを図21に示す。

10

【0042】

また、MACヘッダ生成部(119)はヘッダ/ペイロード合成部(120)からの出力要求(129)を受信すると、生成したMACヘッダをヘッダ/ペイロード合成部(120)に出力(133)する。

【0043】

[ヘッダ/ペイロード合成部]

ヘッダ/ペイロード合成部(120)はフレーム個数フレーム長算出部(102)からのフレーム個数から生成すべきヘッダの個数を割り出す。次に、各プロトコルヘッダ生成部からのヘッダ完了通知とペイロード分割部(103)からのペイロード送信準備完了通知により、送信フレームに必要なヘッダとペイロードの準備が完了していることを認識し、TCP/IPの場合はMACヘッダ生成部(119)、IPヘッダ生成部(118)、TCPヘッダ生成部(116)、ペイロード読み出し部(105)の順に出力要求を行い、ヘッダ及びペイロードを読み出してフレームとして合成し、フレーム送出部(121)に出力する。UDP/IPの場合は先頭フレームはMACヘッダ生成部(119)、IPヘッダ生成部(118)、UDPヘッダ生成部(117)、ペイロード読み出し部(105)の順に出力要求を行い、中間、最終フレームはMACヘッダ生成部(119)、IPヘッダ生成部(118)、ペイロード読み出し部(105)の順に出力要求を行い、ヘッダ及びペイロードを読み出してフレームとして合成しフレーム送出部(121)に出力する。ペイロードを持たないフレームを送信する場合はヘッダのみ読み出して、フレームとして合成し、フレーム送出部(121)に出力する。

20

30

【0044】

[フレーム送出部]

フレーム送出部(121)では、ヘッダ/ペイロード合成部(120)から出力されるフレームを通信制御装置外に送信する機能を有する。IEEE802.3に準拠し、プリアンブルの付加、フレームパディング、CRCの生成等を行う。

【0045】

(第2の実施形態)

図11は、本発明の第2の実施形態による通信制御装置の構成例を示すブロック図である。図11を用いてその構成を説明する。

【0046】

[アプリケーションデバイス]

アプリケーションデバイス(101)は、送信要求(106)と送信データ長(107)をフレーム個数フレーム長算出部(102)に出力し、送信データ(110)をペイロード分割部(103)に出力する。アプリケーションデバイス(101)には、2つの種類が想定される。1つ目はコンスタントビットレート(単位時間あたりの情報量が一定)のデバイスである。2つ目はバリアブルビットレート(単位時間あたりの情報量が一定ではないが、単位時間辺りの情報量は明示される)のデバイスである。その他にバリアブルビットレートのデバイスからの情報を一時的にバッファリングし定期的に一定量の情報を出力するシェーピングするデバイスである場合も想定されるがこれは1つ目の種類のデバイスと同じと見なせる。また、サンプリング周波数で予め規定された量子化ビットレ

40

50

トでRAW情報を出力してくるデバイスも想定できるが、これは1つ目の種類のデバイスと同様に見なすことが出来る。これら全てのアプリケーションデバイス(101)に接続される本実施形態による通信制御装置は、アプリケーションデバイス(101)とのインタフェースにおいて、送信要求(106)、送信データ長(107)、送信データ(110)を受ける事とする。

#### 【0047】

##### [フレーム個数フレーム長算出部]

フレーム個数フレーム長算出部(102)は、アプリケーションデバイス(101)から入力された送信要求(106)をトリガに、同じく入力された送信データ長(107)を基に送信データを幾つのフレームに分割して出力すべきかを算出する。ペイロード長とヘッダ数算出の流れは、図22の上記の説明と同じである。上述の様にフレーム数、先頭フレームペイロード長、中間フレームペイロード長、最終フレームペイロード長をフレーム個数とフレーム長に関する値とし、ペイロード分離部(103)とプロトコル制御部(115)及びヘッダ/ペイロード合成部(120)に出力する。

#### 【0048】

##### [ペイロード分割部の分割]

ペイロード分割部(103)ではフレーム個数フレーム長算出部(102)から与えられたフレーム個数とフレーム長に関する入力(109)を基にアプリケーションデバイス(101)から与えられた送信データ(110)を分割しながらフレームペイロードとしてペイロードTXメモリ(104)に書き込み保存を行う(111)。ペイロード分割部(103)は、フレームペイロード書き込みが完了したらプロトコル制御部(115)に対しフレームペイロードの送信準備が出来た事を通知する(136)。

#### 【0049】

##### [ペイロード分割部のチェックサム]

ペイロード分割部(103)ではフレーム個数フレーム長算出部(102)から与えられたフレーム個数とフレーム長に関する入力(109)を基にアプリケーションデバイス(101)から与えられた送信データ(110)を分割しながらフレームペイロードとしてペイロードTXメモリ(104)に書き込み保存を行う(111)際、フレームペイロードのチェックサムをフレーム毎に求め、UDPヘッダ生成部(117)及び、TCPヘッダ生成部(116)にチェックサム値を伝える(135)。

#### 【0050】

##### [TXメモリ]

TXメモリ(104)では1つの送信要求に伴う一連の送信データを、送信要求毎、フレームペイロードFIFOにフレームペイロードの列として保存している。図5に示す様にTXメモリ(104)内部には1つの送信要求に伴い1つのFIFOが割り当てられ(340)、例えばここでは1つの送信要求による送信データが5つのフレームペイロードから成ることを示している。本図ではフレームペイロード1(先頭フレーム)からフレームペイロード5(最後尾フレーム)がFIFOに蓄えられているがフレーム単位での送信が行われる場合はこの様に5つのフレームが溜まることはない。各々のフレームペイロードはFIFOに挿入された直後にペイロード読み出し部(105)により抜き出され送信される事も考えられる。

#### 【0051】

##### [ペイロード読み出し部]

ペイロード読み出し部(105)ではヘッダ/ペイロード合成部(120)からの、1つのペイロード読み出し指示(114)に対し、送信すべきフレームペイロードをTXメモリ(104)から1つ取り出し、ヘッダ/ペイロード合成部(120)へ出力する(113)。

#### 【0052】

##### [プロトコル制御部]

プロトコル制御部(115)ではフレーム個数フレーム長算出部(102)からの指示

10

20

30

40

50

により、フレームごとに必要なヘッダを特定してヘッダに必要な情報を収集し、各プロトコルヘッダ生成部（１１６～１１９）にヘッダ生成指示及びヘッダ生成情報を通知する（１２２～１２５）。図１２にプロトコル制御部（１１５）のフローチャートを示す。ステップＳ１２０１においてフレーム個数フレーム長算出部（１０２）よりフレーム個数、先頭フレームのペイロード長、中間フレームのペイロード長、最終フレームのペイロード長の４つの値（１０８）を受信したかの判断を行う。もし受信すれば（ステップＳ１２０１：Ｙｅｓ）、ステップＳ１２０２において送信フレームがＴＣＰなのかＵＤＰなのかを判断し、フレーム送信に必要なプロトコルヘッダを生成するための情報を収集する（ステップＳ１２０３）。この時、収集する情報は１つのフレームに付加されるヘッダについてのみ行われる。つまり最初は先頭フレームに付加されるヘッダのみの情報収集を行う。また、収集する各種情報は静的に設定されている場合と動的にテーブル検索を行う場合がある。動的なテーブルとしては、コネクションに付随する宛先ＩＰアドレス、送信元ＩＰアドレス、宛先ポート、送信元ポートを割り出すアドレステーブルや、ＩＰのルーティングテーブルなどがあげられる。次にペイロード分割部（１０３）からフレームペイロード送信準備完了通知を受信したかの判断を行う（ステップＳ１２０４）。もし、フレームペイロード送信準備完了通知を受信したら（ステップＳ１２０４：Ｙｅｓ）、ステップＳ１２０５にてヘッダ情報の収集が終了しているかの判断を行う。ヘッダ情報の収集が終了した場合（ステップＳ１２０５：Ｙｅｓ）は、ステップＳ１２０６にて送信フレームに必要なヘッダを生成するヘッダ生成部にヘッダ生成指示及びヘッダ生成情報を出力する。ヘッダ生成指示を出力後はステップＳ１２０７においてフレーム個数分のヘッダ生成指示を行ったかの判断を行う。もし、フレーム個数分のヘッダ生成指示を行った（ステップＳ１２０７：Ｙｅｓ）のなら、ステップＳ１２０８においてステップＳ１２０６で作成を指示したヘッダのヘッダ送信完了通知をヘッダ／ペイロード合成部（１２０）から受信するのを待ち、ヘッダ送信完了通知を受信したら１つのデータグラムに対する全てのフレームのヘッダが送信されたとしてプロトコル制御を終了する。ステップＳ１２０７においてもしフレーム個数分のヘッダ生成指示を行っていないのならば（Ｓ１２０７：Ｎｏ）、まだ送信すべきフレームを生成しなければならないとして、ステップＳ１２０９においてステップＳ１２０６で作成を指示したヘッダのヘッダ送信完了通知をヘッダ／ペイロード合成部（１２０）から受信するのを待ち、ヘッダ送信完了通知を受信したら次のフレームのヘッダ作成を開始する（ステップＳ１２１０）。そしてステップＳ１２０３に戻る。

#### 【００５３】

##### [ＴＣＰヘッダ生成部]

ＴＣＰヘッダ生成部（１１６）ではプロトコル制御部（１１５）からのヘッダ生成指示及びヘッダ生成情報に基づきヘッダ生成を開始し、ヘッダ生成完了後にヘッダ／ペイロード合成部（１２０）にＴＣＰヘッダ生成完了通知（１２６）を行う。ＴＣＰヘッダ生成指示をプロトコル制御部（１１５）から受信したら、プロトコル制御部（１１５）からＴＣＰヘッダ情報を取得する。ＴＣＰヘッダ情報は送信元ポート番号、宛先ポート番号、シーケンス番号、確認応答番号、ヘッダ長、フラグビット、ウィンドウサイズ、緊急ポインタ及び擬似ヘッダ用の宛先ＩＰアドレス、送信元ＩＰアドレス、セグメント長（ＴＣＰヘッダ長＋ペイロード長）である。取得したヘッダ生成情報から擬似ヘッダとチェックサム以外のＴＣＰヘッダを作成する。次にフレームごとのペイロードサムをペイロード分割部（１０３）より受信し、擬似ヘッダとチェックサム以外のＴＣＰヘッダとペイロードサムからＴＣＰのチェックサムを算出し、チェックサムをＴＣＰヘッダのチェックサムフィールドに設定し、ＴＣＰヘッダの生成を完了する。ＴＣＰヘッダの生成が完了すると、ヘッダ／ペイロード合成部（１２０）にＴＣＰヘッダ生成完了通知を行い、ＴＣＰヘッダ生成を終了し、プロトコル制御部（１１５）より次のヘッダ生成指示があるまで待機する。ＴＣＰヘッダ生成部（１１６）で作成するヘッダのフォーマットを図１４に、擬似ヘッダフォーマットを図１５に示す。

#### 【００５４】

また、ＴＣＰヘッダ生成部（１１６）はヘッダ／ペイロード合成部（１２０）からの出

10

20

30

40

50

力要求(126)を受信すると、生成したTCPヘッダをヘッダ/ペイロード合成部(120)に出力(130)する。

【0055】

[UDPヘッダ生成部]

UDPヘッダ生成部(117)ではプロトコル制御部(115)からのヘッダ生成指示及びヘッダ生成情報に基づきヘッダ生成を開始し、ヘッダ生成完了後にヘッダ/ペイロード合成部(120)にUDPヘッダ生成完了通知(127)を行う。UDPヘッダ生成指示をプロトコル制御部(115)から受信したら、プロトコル制御部(115)からUDPヘッダ情報を取得する。UDPヘッダ情報は、送信元ポート番号、宛先ポート番号、UDPデータ長である。取得したヘッダ生成情報から擬似ヘッダとチェックサム以外のUDPヘッダを作成する。次にペイロードサムをペイロード分割部(103)より受信したかの判断を行い、もし受信していれば擬似ヘッダとチェックサム以外のUDPヘッダとペイロードサムの合計からUDPチェックサムを算出し、チェックサムをUDPヘッダのチェックサムフィールドに設定し、UDPヘッダの生成を完了する。UDPヘッダの生成が完了すると、ヘッダ/ペイロード合成部(120)にUDPヘッダ生成完了通知を行い、UDPヘッダ生成を終了する。

10

【0056】

ただし、UDPヘッダのチェックサムはオプションでの対応となっているため、UDPチェックサムを行わない設定にした場合は、プロトコル制御部(115)よりUDPヘッダ情報(送信元ポート番号、宛先ポート番号、UDPデータ長)を取得した時点で、UDPチェックサムフィールドに0を設定し、UDPヘッダ生成完了通知をヘッダ/ペイロード合成部(120)に出力する。

20

【0057】

UDPヘッダ生成部(117)で作成するヘッダのフォーマットを図16に、擬似ヘッダフォーマットを図17に示す。

【0058】

また、UDPヘッダ生成部(117)はヘッダ/ペイロード合成部(120)からの出力要求(127)を受信すると、生成したUDPヘッダをヘッダ/ペイロード合成部(120)に出力(131)する。

【0059】

30

[IPヘッダ生成部]

IPヘッダ生成部(118)ではプロトコル制御部(115)からのヘッダ生成指示及びヘッダ生成情報に基づきヘッダ生成を開始し、ヘッダ生成完了後にヘッダ/ペイロード合成部(120)にIPヘッダ生成完了通知(128)を行う。IPヘッダ生成指示をプロトコル制御部(115)から受信したら、プロトコル制御部(115)からIPヘッダ情報を取得する。IPヘッダ情報はバージョン、ヘッダ長、TOS、全長、識別子、フラグ、フラグメントオフセット、TTL、プロトコル、送信元IPアドレス、宛先IPアドレスである。これらの情報からIPヘッダのヘッダチェックサムを算出してIPヘッダを完成させ、ヘッダ/ペイロード合成部(120)にIPヘッダ生成完了通知を行う。IPヘッダ生成部(118)で作成するヘッダのフォーマットを図20に示す。

40

【0060】

また、IPヘッダ生成部(118)はヘッダ/ペイロード合成部(120)からの出力要求(128)を受信すると、生成した順番にIPヘッダをヘッダ/ペイロード合成部(120)に出力(132)する。

【0061】

[MACヘッダ生成部]

MACヘッダ生成部(119)ではプロトコル制御部(115)からのヘッダ生成指示及びヘッダ生成情報に基づきヘッダ生成を開始し、ヘッダ生成完了後にヘッダ/ペイロード合成部(120)にMACヘッダ生成完了通知(129)を行う。MACヘッダ生成部(119)はプロトコル制御部(115)からMACヘッダ生成指示を受信したら、MA

50

Cヘッダ情報を取得する。MACヘッダ生成情報は、ヘッダ生成数、送信元MACアドレス、宛先MACアドレス、タイプレングスである。これらの情報からMACヘッダを完成させ、ヘッダ/ペイロード合成部(120)にMACヘッダ生成完了通知を行う。MACヘッダ生成部(119)で作成するMACヘッダのフォーマットを図21に示す。

【0062】

また、MACヘッダ生成部(119)はヘッダ/ペイロード合成部(120)からの出力要求(129)を受信すると、生成したMACヘッダをヘッダ/ペイロード合成部(120)に出力(133)する。

【0063】

[ヘッダ/ペイロード合成部]

ヘッダ/ペイロード合成部(120)はフレーム個数フレーム長算出部(102)からのフレーム個数から生成すべきヘッダの個数を割り出す。次に、各プロトコルヘッダ生成部からのヘッダ完了通知により、送信フレームに必要なヘッダが完了していることを認識し、TCP/IPの場合はMACヘッダ生成部(119)、IPヘッダ生成部(118)、TCPヘッダ生成部(116)、ペイロード読み出し部(105)の順に出力要求を行い、ヘッダ及びペイロードを読み出してフレームとして合成し、フレーム送出部(121)に出力する。UDP/IPの場合は先頭フレームはMACヘッダ生成部(119)、IPヘッダ生成部(118)、UDPヘッダ生成部(117)、ペイロード読み出し部(105)の順に出力要求を行い、中間、最終フレームはMACヘッダ生成部(119)、IPヘッダ生成部(118)、ペイロード読み出し部(105)の順に出力要求を行い、ヘッダ及びペイロードを読み出してフレームとして合成しフレーム送出部(121)に出力する。ペイロードを持たないフレームを送信する場合はヘッダのみ読み出して、フレームとして合成し、フレーム送出部(121)に出力する。ヘッダ/ペイロード合成部(120)は、ヘッダをフレーム送出部(121)に出力した時点で、プロトコル制御部(115)にヘッダ送信完了(137)を出力する。

【0064】

[フレーム送出部]

フレーム送出部(121)では、ヘッダ/ペイロード合成部(120)から出力されるフレームを通信制御装置外に送信する機能を有する。IEEE802.3に準拠し、ブリアンプルの付加、フレームパディング、CRCの生成等を行う。

【0065】

<TCPセグメントの送信例>

TCP/IPのフレーム送信時に送信データグラムが3つのセグメントに分割された場合の例を図13に示す。アプリケーションデバイス(101)からの送信指示及び送信データ長(1301)を受けてフレーム個数フレーム長算出部(102)は、フレーム個数(3個)、先頭フレームのペイロード長(PL0の長さ)、中間フレームのペイロード長(PL1の長さ)、最終フレームのペイロード長(PL2の長さ)をプロトコル制御部(115)及びヘッダ/ペイロード合成部(120)に出力(1302)する。プロトコル制御部(115)ではプロトコルタイプをTCPであると判断し、TCP、IP、MACのヘッダ情報の収集を行う。またペイロード分割部(103)においてペイロードをTxメモリ(104)に書き込んでいき、1フレーム分(1TCPセグメント分)書き込みが完了すると、プロトコル制御部(115)に対してフレームペイロード送信準備完了(1303)を出力する。プロトコル制御部(115)はフレームペイロード送信準備完了(1303)を受信するとデータグラムをセグメント化した先頭フレームのTCP、IP、MACのヘッダを生成するためのヘッダ情報収集が終了しているかを判断し、終了していればTCPヘッダ生成部(116)にヘッダ生成指示及びヘッダ情報(1304)を、IPヘッダ生成部(118)にヘッダ生成指示及びヘッダ情報(1305)を、MACヘッダ生成部(119)にヘッダ生成指示及びヘッダ情報(1306)を同時に出力する。ヘッダ生成指示を受信したTCPヘッダ生成部(116)はプロトコル制御部(115)からのヘッダ情報とペイロード分割部(103)からのペイロードサム(1307)よりT

10

20

30

40

50



ＣＰヘッダを生成し、ヘッダ／ペイロード合成部（１２０）にＴＣＰヘッダ生成完了（１３０８）を出力する。ヘッダ生成指示を受信したＩＰヘッダ生成部（１１８）はプロトコル制御部（１１５）からのヘッダ情報からＩＰヘッダを生成し、ヘッダ／ペイロード合成部（１２０）にＩＰヘッダ生成完了（１３０９）を出力する。ヘッダ生成指示を受信したＭＡＣヘッダ生成部（１１９）はプロトコル制御部（１１５）からのヘッダ情報からＭＡＣヘッダを生成し、ヘッダ／ペイロード合成部（１２０）にＭＡＣヘッダ生成完了（１３１０）をフレーム個数分出力する。ヘッダ／ペイロード合成部（１２０）はフレーム個数フレーム長算出部（１０２）から受信したフレーム個数（１３０２）から３つにセグメント化されたフレームの先頭フレームにはＴＣＰヘッダ、ＩＰヘッダ、ＭＡＣヘッダが必要であることを認識しており、ＴＣＰヘッダ生成完了（１３０８）、ＩＰヘッダ生成完了（１３０９）、ＭＡＣヘッダ生成完了（１３１０）の全てを受信するとフレームを送信可能な状態であると判断する。ヘッダ／ペイロード合成部（１２０）はＭＡＣヘッダ、ＩＰヘッダ、ＴＣＰヘッダ、ペイロードと読み出していきフレーム送出部（１２１）に先頭フレームを出力する。また、ＭＡＣ、ＩＰ、ＴＣＰヘッダをフレーム送出部（１２１）に出力した時点でプロトコル制御部（１１５）にヘッダ送信完了（１３１１）を出力する。

【００６６】

プロトコル制御部（１１５）はヘッダ送信完了（１３１１）を受信すると次のフレーム、つまりここでは３つにセグメント化されたフレームの中間フレームのヘッダ情報収集を行う。中間フレームに必要なヘッダは先頭と同様にＴＣＰ、ＩＰ、ＭＡＣヘッダの情報収集を行う。ペイロード分割部（１０３）からの次のフレームペイロード送信準備完了（１３１２）を受信すると、中間フレームのＴＣＰ、ＩＰ、ＭＡＣのヘッダを生成するためのヘッダ情報収集が終了しているかを判断し、終了していればＴＣＰヘッダ生成部（１１６）にヘッダ生成指示及びヘッダ情報（１３１３）を、ＩＰヘッダ生成部（１１８）にヘッダ生成指示及びヘッダ情報（１３１４）を同時に出力する。ヘッダ生成指示を受信したＴＣＰヘッダ生成部（１１６）はプロトコル制御部（１１５）からのヘッダ情報とペイロード分割部（１０３）からのペイロードサム（１３１５）よりＴＣＰヘッダを生成し、ヘッダ／ペイロード合成部（１２０）にＴＣＰヘッダ生成完了（１３１６）を出力する。ヘッダ生成指示を受信したＩＰヘッダ生成部（１１８）はプロトコル制御部（１１５）からのヘッダ情報からＩＰヘッダを生成し、ヘッダ／ペイロード合成部（１２０）にＩＰヘッダ生成完了（１３１７）を出力する。

【００６７】

ヘッダ／ペイロード合成部（１２０）はフレーム個数フレーム長算出部（１０２）から受信したフレーム個数（１３０２）から３つにセグメント化されたフレームの中間フレームにはＴＣＰヘッダ、ＩＰヘッダ、ＭＡＣヘッダが必要であることを認識しており、ＴＣＰヘッダ生成完了（１３１６）、ＩＰヘッダ生成完了（１３１７）を受信するとフレームを送信可能な状態であると判断する。ＭＡＣヘッダ生成完了については先頭フレームの時にフレーム個数分出力されているため、ヘッダ／ペイロード合成部（１２０）はこの時点で中間フレームのＭＡＣヘッダ生成が完了していると判断している。フレームが送信可能な状態であると判断したヘッダ／ペイロード合成部（１２０）はＭＡＣヘッダ、ＩＰヘッダ、ＴＣＰヘッダ、ペイロードと読み出していきフレーム送出部（１２１）に中間フレームを出力する。また、ＭＡＣ、ＩＰ、ＴＣＰヘッダをフレーム送出部（１２１）に出力した時点でプロトコル制御部（１１５）にヘッダ送信完了（１３２０）を出力する。

【００６８】

プロトコル制御部（１１５）はヘッダ送信完了（１３２０）を受信すると次のフレーム、つまりここでは３つにセグメント化されたフレームの最終フレームのヘッダ情報収集を行う。最終フレームに必要なヘッダは先頭、中間と同様にＴＣＰ、ＩＰ、ＭＡＣヘッダの情報収集を行う。ペイロード分割部（１０３）からの次のフレームペイロード送信準備完了（１３１８）をすでに受信しているので、ＴＣＰ、ＩＰ、ＭＡＣのヘッダを生成するためのヘッダ情報収集が終了し次第、ＴＣＰヘッダ生成部（１１６）にヘッダ生成指示及びヘッダ情報（１３２１）を、ＩＰヘッダ生成部（１１８）にヘッダ生成指示及びヘッダ情

報(1322)を同時に出力する。ヘッダ生成指示を受信したTCPヘッダ生成部(116)はプロトコル制御部(115)からのヘッダ情報とペイロード分割部(103)からのペイロードサム(1319)よりTCPヘッダを生成し、ヘッダ/ペイロード合成部(120)にTCPヘッダ生成完了(1323)を出力する。ヘッダ生成指示を受信したIPヘッダ生成部(118)はプロトコル制御部(115)からのヘッダ情報からIPヘッダを生成し、ヘッダ/ペイロード合成部(120)にIPヘッダ生成完了(1324)を出力する。

#### 【0069】

ヘッダ/ペイロード合成部(120)はフレーム個数フレーム長算出部(102)から受信したフレーム個数(1302)から3つにセグメント化されたフレームの最終フレームにはTCPヘッダ、IPヘッダ、MACヘッダが必要であることを認識しており、TCPヘッダ生成完了(1323)、IPヘッダ生成完了(1324)を受信するとフレームを送信可能な状態であると判断する。MACヘッダ生成完了については先頭フレームの時にフレーム個数分出力されているため、ヘッダ/ペイロード合成部(120)はこの時点で中間フレームのMACヘッダ生成が完了していると判断している。フレームが送信可能な状態であると判断したヘッダ/ペイロード合成部(120)はMACヘッダ、IPヘッダ、TCPヘッダ、ペイロードと読み出していきフレーム送出部(121)に最終フレームを出力する。また、MAC、IP、TCPヘッダをフレーム送出部に出力した時点でプロトコル制御部(115)にヘッダ送信完了(1325)を出力する。

#### 【0070】

以上のように、第1及び第2の実施形態は、ネットワークに接続し、低価格のCPUもしくはCPUレスで、比較的高速なデータを送受信する機器における、ネットワークインタフェースの通信制御装置であって、アプリケーションデバイスから任意の長さの送信データを受信し、サポートするデータリンク層に適切なフレームサイズでフレーム出力する。

#### 【0071】

フレーム個数フレーム長算出部は、データ送信要求に基づき送信データを送信するためのフレーム個数及びフレーム長を算出する。ペイロード分割部は、フレーム個数フレーム長算出部により算出されたフレーム個数及びフレーム長を基に送信データをフレーム単位に分割する。ヘッダ生成部は、フレーム個数フレーム長算出部により算出されたフレーム個数及びフレーム長を基にフレーム単位の送信データに付加すべきヘッダを生成する。その際、ヘッダ生成部は、ペイロード分割部が送信データを入力している間に送信データに付加すべきヘッダを生成することができる。

#### 【0072】

ペイロード分割部は、フレーム単位の送信データを加算してフレーム毎のチェックサムを算出する。ヘッダ生成部は、ヘッダ内にチェックサムを付加する。これにより、送信データを受信してから短期間でヘッダつきフレームを出力することができる。

#### 【0073】

ペイロード分割部は、複数フレームの送信データを入力している間でも1フレーム分の送信データを入力して分割する毎にフレーム単位の分割完了通知を出力する。ヘッダ生成部は、フレーム単位の分割完了通知を基にフレーム単位のヘッダを生成する。また、ヘッダ生成部は、ペイロード分割部が複数フレームの送信データを入力している間でも各フレームのヘッダを生成する毎にフレーム単位のヘッダ生成完了通知を出力する。ヘッダ/ペイロード合成部は、フレーム単位のヘッダ生成完了通知を基にフレーム単位の送信データ及びヘッダを合成する。これにより、アプリケーションデバイスから複数フレームに相当する長い送信データを受信している間でも、その送信データを構成する各フレームの送信を開始することができる。

#### 【0074】

以上のように、第1及び第2の実施形態をまとめると、以下ようになる。

(1) アプリケーションデバイスからの送信指示を基に、送信データをフレームとして

どう分割すべきかを初めに算出する。即ち 1 つの送信指示に伴う送信データを幾つのフレームとして出力すべきか、先頭フレームのペイロード長、中間フレームのペイロード長、最終フレームのペイロード長を算出する。

【0075】

(2)(1)の結果を基に、送信データをフレームとして送出する為のヘッダ生成と、フレームペイロード生成(分割)の処理を同時に開始する。

【0076】

(3)フレーム個数とペイロード長が判明しているので各々のプロトコルスタック(TCP/UDP/IP/MAC)のヘッダ生成を同時に開始する。

【0077】

(4)ペイロード長が判明しているのでアプリケーションデバイスから送信データを受け取りメモリに一時退避するときにペイロード単位で分割しながらペイロード単位で保存する。

【0078】

(5)フレーム単位でのヘッダ生成、ペイロード保存が成されるので、フレーム単位でヘッダとペイロードを合成してフレーム出力を行う。このフレーム出力はアプリケーションデバイスからの送信データを全て受けてからの場合もあるが、アプリケーションデバイスからの送信でデータのうちの1フレームを受信したらすぐにはじめの1フレームを出力する場合もある。

【0079】

本実施形態によれば、ハードウェアが持つ並列処理能力を発揮でき、メモリ管理方法も単純化される。スループットの大きい(帯域の広い)通信制御装置(TOE又はプロトコル制御装置)を実現できる。アプリケーションからの送信データが本通信制御装置(TOE又はプロトコル制御装置)に入力完了すると殆ど直ぐにインタフェースにフレームとして送信可能となる。即ち、アプリケーションデバイスからの送信データがフレームとしてネットワークインタフェースに出力されるまでの遅延時間が殆どない。

【0080】

なお、上記実施形態は、何れも本発明を実施するにあたっての具体化の例を示したものに過ぎず、これらによって本発明の技術的範囲が限定的に解釈されてはならないものである。すなわち、本発明はその技術思想、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。

【図面の簡単な説明】

【0081】

【図1】本発明の第1の実施形態による通信制御装置の構成図である。

【図2】シーケンス1を示す図である。

【図3】シーケンス2を示す図である。

【図4】シーケンス3を示す図である。

【図5】TXメモリを示す図である。

【図6】シーケンス4を示す図である。

【図7】シーケンス5を示す図である。

【図8】TCPヘッダ生成部の処理を示すフローチャートである。

【図9】UDPヘッダ生成部の処理を示すフローチャートである。

【図10】IPヘッダ生成部の処理を示すフローチャートである。

【図11】本発明の第2の実施形態による通信制御装置の構成図である。

【図12】プロトコル制御部の処理を示すフローチャートである。

【図13】シーケンス6を示すフローチャートである。

【図14】TCPヘッダフォーマットを示す図である。

【図15】TCP擬似ヘッダフォーマットを示す図である。

【図16】UDPヘッダフォーマットを示す図である。

【図17】UDP擬似ヘッダフォーマットを示す図である。

10

20

30

40

50

【図 18】IP ヘッダフォーマット（フラグメント）を示す図である。

【図 19】IP ヘッダフォーマット（セグメント）を示す図である。

【図 20】IP ヘッダフォーマットを示す図である。

【図 21】MAC ヘッダフォーマットを示す図である。

【図 22】ペイロード長とヘッダ数算出の流れを示すフローチャートである。

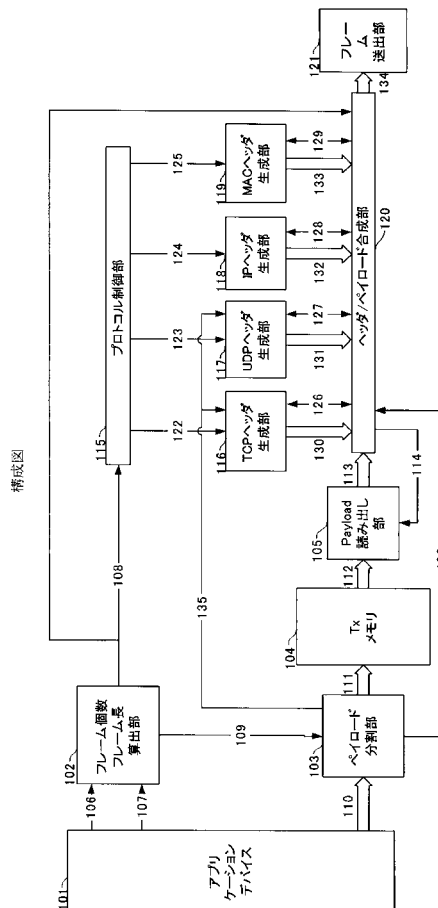
【符号の説明】

【 0 0 8 2 】

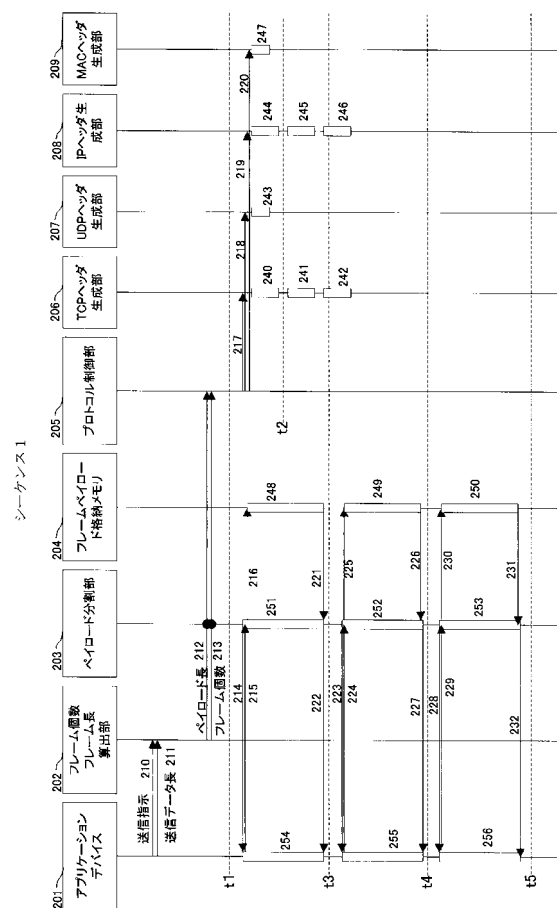
- 1 0 1 アプリケーションデバイス
- 1 0 2 フレーム個数フレーム長算出部
- 1 0 3 ペイロード分割部
- 1 0 4 TXメモリ
- 1 0 5 ペイロード読み出し部
- 1 1 5 プロトコル制御部
- 1 1 6 TCPヘッダ生成部
- 1 1 7 UDPヘッダ生成部
- 1 1 8 IPヘッダ生成部
- 1 1 9 MACヘッダ生成部
- 1 2 0 ヘッダ/ペイロード合成部
- 1 2 1 フレーム送出部

10

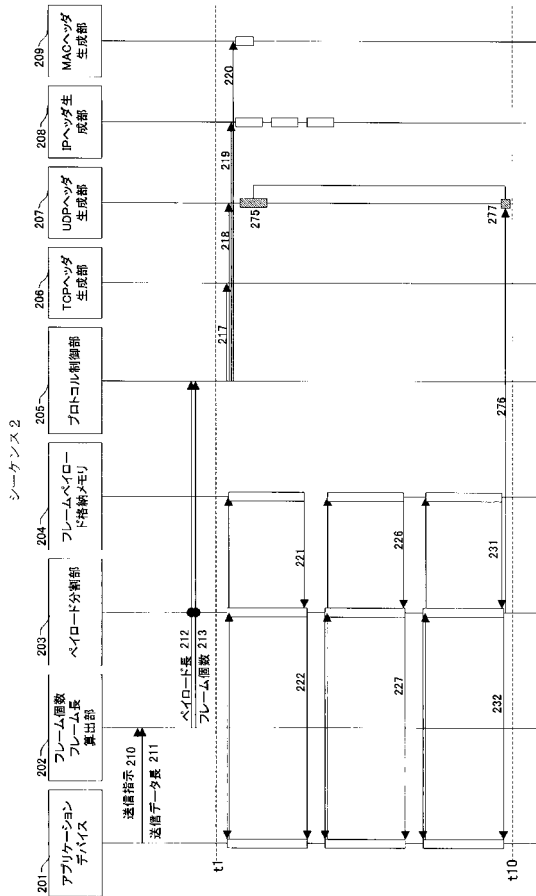
【図 1】



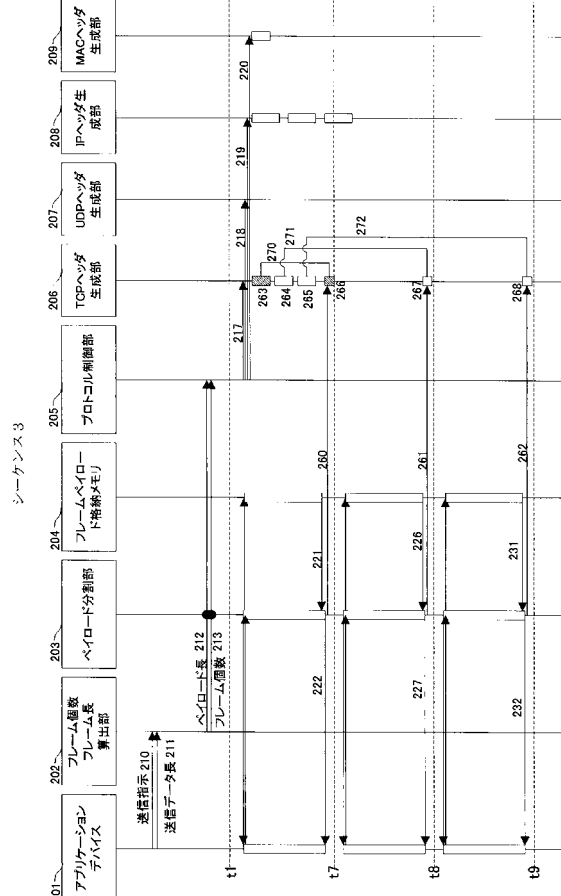
【図 2】



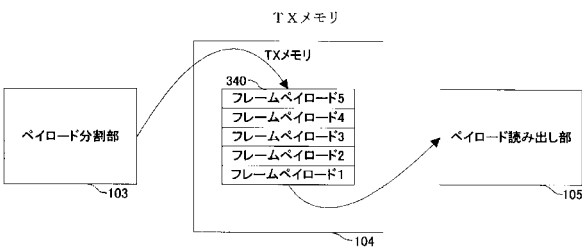
【図 3】



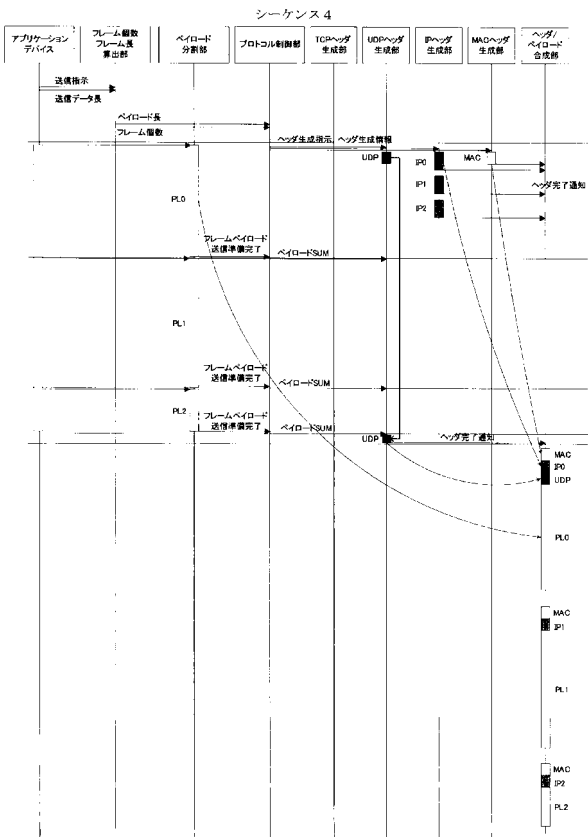
【図 4】



【図 5】

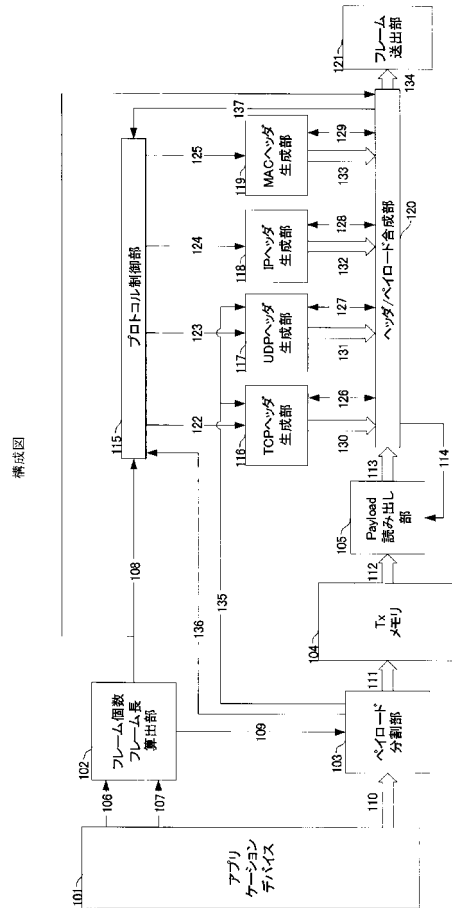


【図 6】

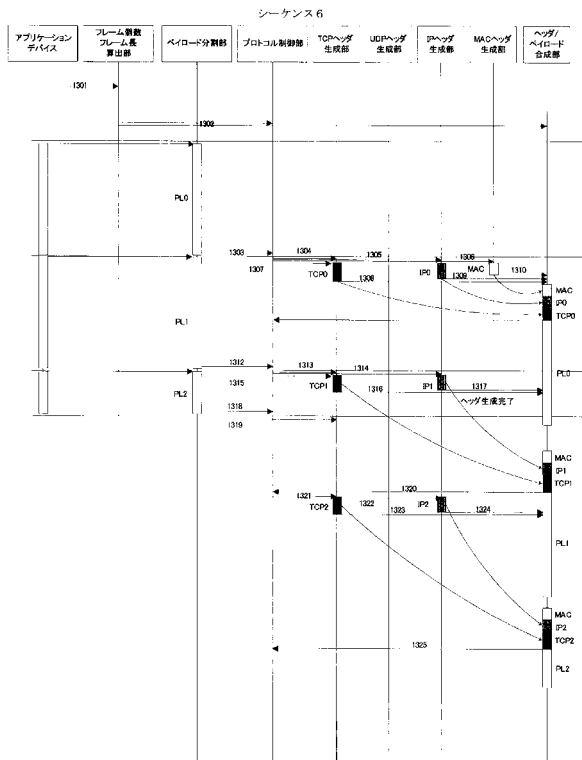




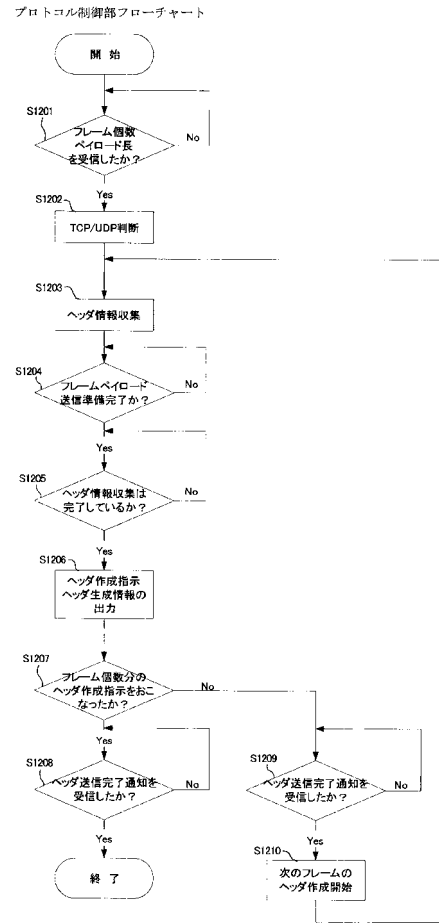
【 図 1 1 】



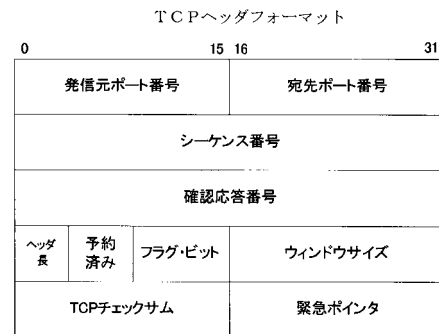
【 図 1 3 】



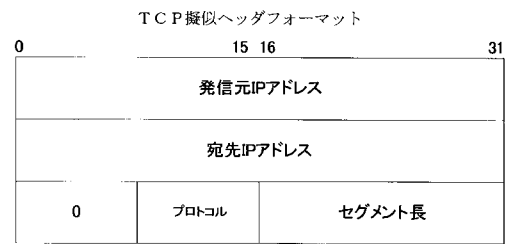
【 図 1 2 】



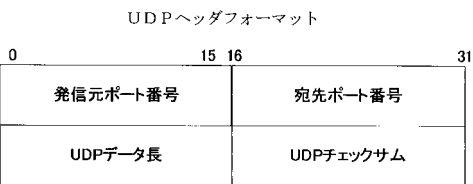
【 図 1 4 】



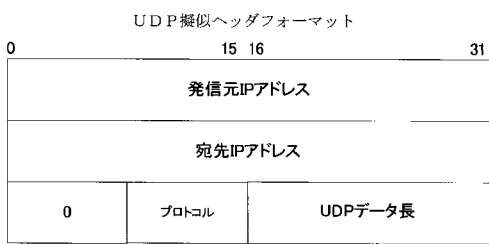
【図 1 5】



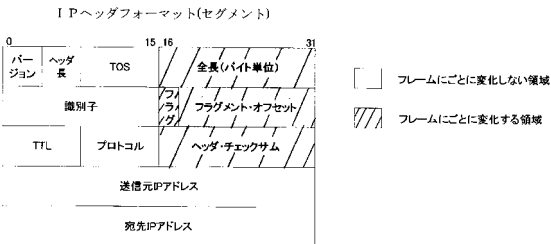
【図 1 6】



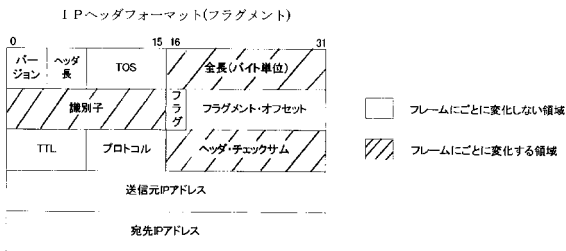
【図 1 7】



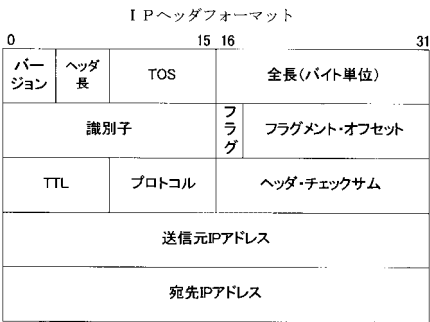
【図 1 9】



【図 1 8】

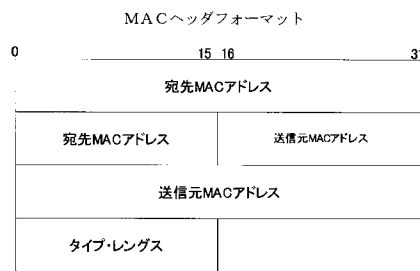


【図 2 0】

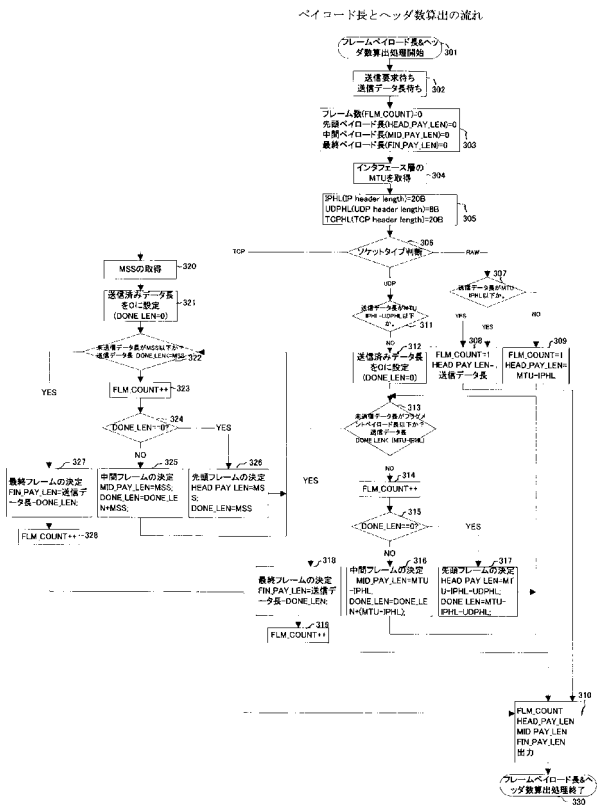




【 図 2 1 】



【 図 2 2 】



---

フロントページの続き

審査官 阿部 弘

- (56)参考文献 特開2000-253054(JP,A)  
特開2003-046566(JP,A)  
特開平06-069957(JP,A)  
特開2001-016226(JP,A)  
特開2004-215203(JP,A)  
特開平11-317762(JP,A)  
特開昭63-099651(JP,A)  
特開2004-241872(JP,A)  
特開2001-339462(JP,A)

- (58)調査した分野(Int.Cl., DB名)  
H04L 29/06  
H04L 12/56