(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2016-63421
(P2016-63421A)
(43)公開日　平成28年4月25日(2016.4.25)

(51) Int.Cl.		F I			テーマコード（参考）
H04L 13/08	(2006.01)	H04L	13/08		5B089
G06F 13/00	(2006.01)	G06F	13/00	353Q	5K034

審査請求 未請求　請求項の数 17　ＯＬ　（全 27 頁）

(21)出願番号　特願2014-190420 (P2014-190420)
(22)出願日　平成26年9月18日 (2014.9.18)

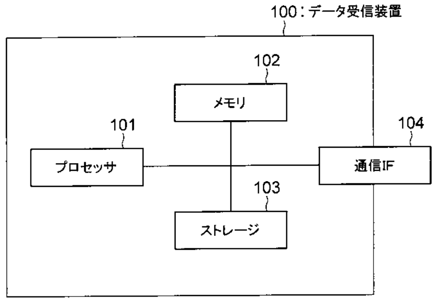
（特許庁注：以下のものは登録商標）
１．ＢＬＵＥＴＯＯＴＨ

(71)出願人　000003078
株式会社東芝
東京都港区芝浦一丁目１番１号
(74)代理人　100117787
弁理士　勝沼　宏仁
(74)代理人　100107582
弁理士　関根　毅
(74)代理人　100118876
弁理士　鈴木　順生
(72)発明者　西本　寛
東京都港区芝浦一丁目１番１号　株式会社東芝内
(72)発明者　石原　丈士
東京都港区芝浦一丁目１番１号　株式会社東芝内

最終頁に続く

(54)【発明の名称】データ受信装置およびデータ受信方法

(57)【要約】
【課題】データを高速に受信するとともに、当該データを効率よく記憶部に書き込む
【解決手段】本発明の一態様としてのデータ受信装置は、ネットワークを介して第１データおよび第２データを受信する通信部と、データの読み書きが行われる第１記憶部と、一定のブロックサイズを単位としてデータの読み書きが行われる第２記憶部と、プロセッサと、を備える。前記プロセッサは、第１バッファおよび第２バッファを第１記憶部に設定する。前記第１データのサイズから前記第１バッファの前記空き領域のサイズを減じた値を前記第１バッファのサイズで除算したときの剰余のサイズ分の末尾データを第２バッファに書き込み、前記剰余のサイズ分のデータに連続するように第２バッファへ前記第２データを書き込む。
【選択図】図１



【特許請求の範囲】
【請求項１】
ネットワークを介して第１データおよび第２データを受信する通信部と、
データの読み書きが行われる第１記憶部と、
一定のブロックサイズを単位としてデータの読み書きが行われる第２記憶部と、
プロセッサと、を備え、
前記プロセッサは、前記第１記憶部に前記ブロックサイズの整数倍のサイズを有するバッファを設定する手段と、前記通信部で受信される前記第１データのサイズを特定する手段とを有し、
前記通信部で受信された第１データを前記第１記憶部に予め設定した第１バッファの空き領域に書き込み、
前記第１記憶部に第２バッファを設定し、前記第１データのサイズから前記第１バッファにおける前記第１データの書き込み開始前の前記空き領域のサイズを減じた値を前記第１バッファのサイズで除算したときの剰余のサイズ分の領域を、前記第２バッファに確保し、
前記第１バッファへの前記第１データの書き込みの途中で前記第１バッファ内のデータ量が第１の所定値に達すると、前記第１バッファ内のデータを前記第２記憶部に書き出し、前記第１バッファを解放し、
前記第１データの末尾の前記剰余のサイズ分のデータを、前記第２バッファの前記確保した領域に書き込み、前記第２バッファにおける前記確保した領域から連続するアドレスの領域に前記第２データを書き込み、前記第２データの書き込みの途中で前記第２バッファ内のデータ量が第２の所定値に達すると、前記第２バッファ内のデータを前記第２記憶部に書き出す
データ受信装置。
【請求項２】
前記第１の所定値は、前記第１バッファのサイズに一致し、前記第２の所定値は、前記第２バッファのサイズに一致する
請求項１に記載のデータ受信装置。
【請求項３】
前記第１記憶部のデータの読み書き速度は、前記第２記憶部のデータの読み書き速度よりも高速である
請求項１または２に記載のデータ受信装置。
【請求項４】
前記プロセッサは、前記第１データのサイズから前記第１バッファの前記空き領域のサイズを減じた値を前記第１バッファのサイズで除算したときの商が０より大きいときに前記第２記憶部に前記第２バッファを設定し、前記商が０のときには、前記第２バッファの設定を行わず、
前記第２バッファの設定を行わない場合、前記第２データを、前記第１バッファにおいて前記第１データの末尾が格納されるアドレスから連続するアドレスの領域に書き込む
請求項１ないし３のいずれか一項に記載のデータ受信装置。
【請求項５】
前記第１バッファの前記空き領域より前のアドレスの領域は、前記第１データよりも前に受信されたデータが格納された領域、または前記第１データよりも前に受信されたデータを格納するために確保された領域である
請求項１ないし４のいずれか一項に記載のデータ受信装置。
【請求項６】
前記第１バッファは、第１記憶部に予め複数設定した空き領域の中で最大の空き領域であり、前記第１データを前記第１バッファの先頭から書き込む
請求項１ないし４のいずれか一項に記載のデータ受信装置。
【請求項７】

前記第２データの書き込みが完了した場合において、前記第１データおよび第２データ以外に受信する他のデータが存在しない場合は、前記第２バッファ内のデータ量が前記第２の所定値に達していなくても、前記第２バッファ内のデータを前記第２記憶部に書き出す

請求項１ないし６のいずれか一項に記載のデータ受信装置。

【請求項８】

前記通信部は、前記第１取得要求を第１装置に送信し、前記第１取得要求とは異なる第２取得要求を前記第１装置または前記第１装置と異なる第２装置に送信し、

前記第１データは、第１取得要求に応じて前記第１装置から送信されるデータであり、前記第２データは、前記第２取得要求に応じて前記第２装置から送信されるデータである

請求項１なしい７のいずれか一項に記載のデータ受信装置。

【請求項９】

前記第１取得要求および前記第２取得要求は、ＨＴＴＰに従って送信され、

前記プロセッサは、前記第１データのサイズを、前記第１データに含まれるＨＴＴＰヘッダのＣｏｎｔｅｎｔ－Ｌｅｎｇｔｈヘッダから特定する

請求項８に記載のデータ受信装置。

【請求項１０】

前記第１取得要求に応じて、１つまたは複数の第１応答パケットを順次受信し、前記第１データは、前記１つまたは複数の第１応答パケットのペイロード部分に分割して含まれており、

前記第２取得要求に応じて、１つまたは複数の第２応答パケットを順次受信し、前記第２データは、前記１つまたは複数の第２応答パケットのペイロード部分に分割して含まれている

請求項９に記載のデータ受信装置。

【請求項１１】

前記プロセッサは、前記第１応答パケットの先頭の応答パケットに含まれるデータからＨＴＴＰヘッダのＣｏｎｔｅｎｔ－Ｌｅｎｇｔｈヘッダを抽出する

請求項１０に記載のデータ受信装置。

【請求項１２】

前記プロセッサは、前記第１データを解析することにより前記第２取得要求を生成する

請求項９ないし１１のいずれか一項に記載のデータ受信装置。

【請求項１３】

前記第２取得要求は、前記第１データ内で参照されているリンク先からデータを取得することの要求である

請求項１２に記載のデータ受信装置。

【請求項１４】

前記プロセッサは、外部プロセッサから前記第１取得要求を取得し、前記第１取得要求を、前記通信部を介して送信し、

前記データ受信装置は、前記第１データおよび第２データが前記ストレージに保存された後、または前記第１データおよび第２データを前記外部のプロセッサに出力した後、低消費電力状態に移行する

請求項８ないし１３のいずれか一項に記載のデータ受信装置。

【請求項１５】

前記プロセッサは、前記第１データおよび前記第２データを前記ストレージに保存した後、前記外部プロセッサに前記第１取得要求に対する完了通知を送信する

請求項１４に記載のデータ受信装置。

【請求項１６】

前記第２記憶部に書き込まれた前記第１データおよび前記第２データは、同じ１つファイルで管理される

請求項１ないし１５のいずれか一項に記載のデータ受信装置。

【請求項１７】
ネットワークを介して第１データを受信するステップと、
前記第１データのサイズを特定するステップと、
前記ネットワークを介して第２データを受信するステップと、
前記受信された第１データを第１記憶部に予め設定した、前記ブロックサイズの整数倍のサイズを有する第１バッファの空き領域に書き込むステップと、
一定のブロックサイズを単位としてデータの読み書きが行われる第２記憶部に第２バッファを設定するステップと、
前記第１データのサイズから前記第１バッファにおける前記第１データの書き込み開始前の前記空き領域のサイズを減じた値を前記第１バッファのサイズで除算したときの剰余のサイズ分の領域を、前記第２バッファに確保するステップと、
前記第１バッファへの前記第１データの書き込みの途中で前記第１バッファ内のデータ量が第１の所定値に達すると、前記第１バッファ内のデータを前記第２記憶部に書き出し、前記第１バッファを解放するステップと、
前記第１データの末尾の前記剰余のサイズ分のデータを、前記第２バッファの前記確保した領域に書き込み、前記第２バッファにおける前記確保した領域から連続するアドレスの領域に前記第２データを書き込み、前記第２データの書き込みの途中で前記第２バッファ内のデータ量が第２の所定値に達すると、前記第２バッファ内のデータを前記第２記憶部に書き出すステップと
を備えたデータ受信方法。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明の実施形態は、データ受信装置およびデータ受信方法に関する。
【背景技術】
【０００２】
通信技術の発達により、通信インタフェースの伝送レートが向上している。しかし、通信路の不安定さは改善しておらず、エンド・ツー・エンドの安定した通信の実現には、到達確認および再送などの処理が、必要不可欠である。また、無線通信のように、帯域と遅延がともに大きい通信路が、常時使用されるようになりつつある。このような通信路では、送信側機器が受信側機器から到達確認を受け取るまでの間に、受信側機器に対し多くのデータをネットワーク上に送信することで、エンド・ツー・エンドの通信速度を向上させている。
【０００３】
しかしながら、ネットワーク上に多くのデータを送信するためには、それを受信する機器も、多くのデータを高速で処理できる能力を備えなければならない。そのためには通信処理の高速化だけでなく、データ処理の高速化、およびバッファやストレージの高速化が必要である。
【０００４】
バッファやストレージの高速化としては、例えば、新たに書き込むデータが、すでにバッファリング済みのデータに連続するデータか否かを判定し、バッファリング済みのデータに連続するデータであれば、連続領域へと追記し、そうでなければバッファを、フラッシュしてから、当該データを書き込む方法が知られている。また、ＨＴＴＰコネクションを使ってデータを受信する際に、任意長のバッファを動的に確保し、そこにデータを一時保存して、印刷処理と受信処理を並列化する、高速化プリンタが知られている。
【０００５】
しかしながら、これらの先行技術では、複数の接続を用いて、データを並列にダウンロードする場合に、性能が出ない可能性があるうえ、多くのメモリ容量を消費してしまうという問題がある。例えば、接続１と接続２という２つの接続を用いて、データを同時に取得する場合を考える。１番目に説明した先行技術では、２つの異なるデータを、互い違い

に受信すると、毎回バッファのフラッシュが発生して、効率が非常に悪くなってしまう。一方、２番目に説明した先行技術では、状況によっては大きなサイズのバッファも必要になり、搭載するメモリ容量が少ない装置での利用に問題がある。

【先行技術文献】

【特許文献】

【０００６】

【特許文献１】特開２００９－５９２１９号公報

【特許文献２】特開２００６－１８４１５号公報

【発明の概要】

【発明が解決しようとする課題】

【０００７】

本発明の実施形態は、データを高速に受信するとともに、当該データを効率よく記憶部に書き込むことを目的とする。

【課題を解決するための手段】

【０００８】

本発明の一態様としてのデータ受信装置は、ネットワークを介して第１データおよび第２データを受信する通信部と、データの読み書きが行われる第１記憶部と、一定のブロックサイズを単位としてデータの読み書きが行われる第２記憶部と、プロセッサと、を備える。

【０００９】

前記プロセッサは、前記第１記憶部に前記ブロックサイズの整数倍のサイズを有するバッファを設定する手段と、前記通信部で受信される前記第１データのサイズを特定する手段とを有する。

【００１０】

前記プロセッサは、前記通信部で受信された第１データを前記第１記憶部に予め設定した第１バッファの空き領域に書き込む。

【００１１】

前記プロセッサは、前記第１記憶部に第２バッファを設定し、前記第１データのサイズから前記第１バッファにおける前記第１データの書き込みの開始前の前記空き領域のサイズを減じた値を前記第１バッファのサイズで除算したときの剰余のサイズ分の領域を、前記第２バッファに確保する。

【００１２】

前記プロセッサは、前記第１バッファへの前記第１データの書き込みの途中で前記第１バッファ内のデータ量が第１の所定値に達すると、前記第１バッファ内のデータを前記第２記憶部に書き出し、前記第１バッファを解放する。

【００１３】

前記プロセッサは、前記第１データの末尾の前記剰余のサイズ分のデータを、前記第２バッファの前記確保した領域に書き込み、前記第２バッファにおける前記確保した領域から連続するアドレスの領域に前記第２データを書き込み、前記第２データの書き込みの途中で前記第２バッファ内のデータ量が第２の所定値に達すると、前記第２バッファ内のデータを前記第２記憶部に書き出す。

【図面の簡単な説明】

【００１４】

【図１】第１の実施形態に係るデータ受信装置の機能ブロック図。

【図２】データ受信装置を備えた通信ネットワークシステムの模式図。

【図３】データ受信装置が、サーバ装置からウェブページを取得する処理のシーケンスを示す図。

【図４Ａ】バッファにデータを格納する処理の説明図。

【図４Ｂ】バッファにデータを格納する処理の説明図。

【図４Ｃ】バッファにデータを格納する処理の説明図。

【図４Ｄ】バッファにデータを格納する処理の説明図。
【図４Ｅ】バッファにデータを格納する処理の説明図。
【図４Ｆ】バッファにデータを格納する処理の説明図。
【図４Ｇ】バッファにデータを格納する処理の説明図。
【図５】バッファ管理情報の例を示す図。
【図６】応答１および応答２の受信処理が完了したときのストレージの書き込み状態を示す図。
【図７】図１のデータ受信装置による処理の基本的な流れを示すフローチャート。
【図８】図７のフローチャートにおけるいくつかのステップの詳細なフローチャート。
【図９】第２の実施形態に係るモジュールとメインプロセッサとを備えたデータ受信装置のブロック図。
【図１０】第２の実施形態に係る動作のシーケンス図。
【図１１】第３の実施形態に係る他の動作のシーケンス図。
【図１２】第４の実施形態に係るバッファ管理情報の例を示す図。
【図１３】第４の実施形態に係るストレージ管理情報の例を示す図。
【図１４】第４の実施形態に係る動作のフローチャート。

【発明を実施するための形態】

【００１５】
以下、図面を参照しながら、本発明の実施形態について説明する。以下で示す各実施形態は一例であり、本発明は、必ずしもこれらと同一の形態で実施される必要はない。

【００１６】
＜第１の実施形態＞
図１に、第１の実施形態に係るデータ受信装置１００の機能ブロック図を示す。データ受信装置１００は、プロセッサ１０１、メモリ１０２、ストレージ１０３、通信インタフェース１０４を備えている。プロセッサ１０１、メモリ１０２、ストレージ１０３、通信インタフェース１０４は、バスを介して接続されている。

【００１７】
プロセッサ１０１は、アプリケーションプログラムおよびＯＳなどのプログラムを実行する。プロセッサ１０１は、本データ受信装置の動作を司る。

【００１８】
通信インタフェース１０４は、ネットワーク（後述する図２参照）に接続され、ネットワーク上の他の装置と通信を行う。通信インタフェース１０４に接続されるネットワークとして、ＩＥＥＥ８０２．３などの規格に従った有線ＬＡＮや、ＩＥＥＥ８０２．１１などの規格に従った無線ＬＡＮなどがある。通信インタフェースは、これらの例のネットワークに接続されるインタフェースに限定されず、１：１で接続するタイプの通信インタフェースでもよい。または、１：多で接続するタイプの通信インタフェースでもよい。通信インタフェース１０４は、電気的もしくは光学的な方法で、ネットワークに接続可能なものであれば、任意のインタフェースでよい。例えば、通信インタフェース１０４は、ｘＤＳＬ、ＷｉＭＡＸ、ＬＴＥ、Ｂｌｕｅｔｏｏｔｈ、赤外線、可視光通信などに対応したものでもよい。

【００１９】
メモリ１０２は、プロセッサ１０１で実行されるプログラムや、プログラムが使用するデータ（一時データも含む）を格納する記憶部（第１記憶部）である。メモリ１０２は、キャッシュやバッファとしても使用される。例えば、メモリ１０２は、プロセッサ１０１がストレージ１０３との間でデータの読み出しや書き込みを行ったり、通信インタフェース１０４を介して、他の装置との間でデータを交換したりするときに、キャッシュやバッファとして用いられる。メモリ１０２は、例えばＳＲＡＭやＤＲＡＭのような揮発性メモリでもよいし、ＭＲＡＭのような不揮発性メモリでもよい。

【００２０】
ストレージ１０３は、プロセッサ１０１で動作するプログラムやデータを永続的に保存

する記憶部（第２記憶部）である。ストレージ１０３は、メモリ１０２に格納しきれない情報を一時的に退避させたり、一時データを保存する場合にも使用される。ストレージ１０３は、データを永続的に保存できる限り、どのような装置でもよく、一例として、ＮＡＮＤ型フラッシュメモリや、ハードディスク、ＳＳＤがある。本実施形態では、ＮＡＮＤ型フラッシュメモリを想定する。ストレージ１０３のＩ／Ｏ処理は、メモリ１０２よりも低速である。つまり、ストレージ１０３のデータ読み書きの速度は、メモリ１０２よりも低速である。ＮＡＮＤ型フラッシュメモリでは、固定長のブロックを単位としてＩ／Ｏ処理が行われ、ブロック長の整数倍のデータを読み書きすることが効率的である。

【００２１】

図２は、データ受信装置１００を備えた通信ネットワークシステムの模式図である。データ受信装置１００は、インターネット等のネットワーク２００を介して、サーバ装置２０１に接続されている。データ受信装置１００は、サーバ装置２００と通信することにより、例えばウェブページのようなデータを取得する。ウェブページは、例えば動画データや音声データ、画像データなど多様なデータを一般的に含む。以下では、ウェブページを取得する場合を、例に説明する。

【００２２】

図３は、データ受信装置１００が、サーバ装置２０１からウェブページを取得する処理のシーケンスを示したものである。本シーケンスは、プロセッサ１０１で動作するプログラムが、ウェブページの取得要求を発生させることで開始する（Ｓ１０１）。ここではウェブページに含まれる情報を、２つのＴＣＰ接続を使って、並行して取得すると仮定する。

【００２３】

プロセッサ１０１は、２つのＴＣＰ接続を使って、取得要求１と取得要求２の送信指示をそれぞれ、例えばこの順で連続して通信インタフェース１０４に送る（Ｓ１０２、Ｓ１０３）。通信インタフェース１０４は、当該送信指示に従って、取得要求１と取得要求２をサーバ装置２０１に送信する（Ｓ１０４、Ｓ１０５）。取得要求１と取得要求２はそれぞれ別々のオブジェクトに対する取得要求であってもよいし、同一オブジェクトに対する別々のデータレンジの要求であってもよい。例えば、オブジェクト内のデータ範囲を指定できるHTTP Rangeフィールド等を使用してファイルの前半を取得要求１で要求し、ファイルの後半を取得要求２で要求してもよい。

【００２４】

通信インタフェース１０４は、サーバ装置２０１から送信される応答パケット１－１および応答パケット２－１を受信する（Ｓ１０６、Ｓ１０７）。応答パケット１－１は取得要求１に対する応答パケット、応答パケット２－１は取得要求２に対する応答パケットである。

【００２５】

ここで、サーバ装置２０１から送信されるデータは、１つの応答パケットに格納されているとは限らず、複数に分割されて、別々の応答パケットで送信される場合もある。例えば１つのパケットが１５００バイトのサイズを格納可能である場合、１つのパケットには格納できないサイズの画像データは、複数に分割され、それぞれ異なる応答パケットに含められる。この場合、取得要求１に対する１番目の応答パケット１－１に続いて，２番目の応答パケット１－２、３番目の応答パケット１－３、Ｘ番目の応答パケット１－Ｘというように、複数の応答パケットを受信する。取得要求２についても同様に、１番目の応答パケット２－１に続いて、２番目の応答パケット２－２、３番目の応答パケット２－３、Ｙ番目の応答２－Ｙというように、複数の応答パケットを受信する。

【００２６】

以下、応答パケット１－１～応答パケット１－Ｘの集合を、応答１と表記し、応答パケット２－１～応答２－Ｙの集合を応答２と表記することがある。

【００２７】

通信インタフェース１０４が、応答パケット１－１、２－１を受信すると、その応答パ

ケット１－１、２－１は、いったんメモリ１０２に格納される（Ｓ１０８、Ｓ１０９）。その後、通信インタフェース１０４からプロセッサ１０１に、応答パケット１－１、２－１のそれぞれについて、受信通知の割り込みが発行される（Ｓ１１０）。

【００２８】

プロセッサ１０１は受信通知を検出すると、受信処理（Ｓ１１１）を開始する。プロセッサ１０１は、受信処理において、メモリ１０２を参照しながら、応答パケット１－１、２－１内のペイロード部からデータを取り出し、取り出したデータを、メモリ１０２上に確保したバッファに蓄積していく（Ｓ１１２）。メモリ１０２内のバッファへのデータの書き込みには、後述する本実施形態の方式を使用する（バッファには応答１に属するデータの末尾と応答２に属するデータの先頭が連続して書き込まれ得る）。なお、メモリ１０２に確保されるバッファは１つとは限らない。後述するように、本実施形態の例では、３つのバッファが確保される。

【００２９】

ここで、通信インタフェース１０４からメモリ１０２に応答パケット１－１、２－１を保存する処理（図３のＳ１０８、Ｓ１０９の処理）の際に使用するメモリ１０２の領域と、受信処理（Ｓ１１１）とデータ保存（Ｓ１１２）の過程でデータを蓄積するバッファの領域とは、異なるとする。

【００３０】

ステップＳ１１２でメモリ１０２上のバッファに書き込んだデータ量が、所定の閾値（所定値）に達したかを判断する（Ｓ１１３）。所定の閾値（所定値）は、バッファのサイズに一致する。所定の閾値に達した場合には、メモリ１０２上のバッファに蓄積されたデータをすべて読み出して、ストレージ１０３に書き込む（Ｓ１１４）。所定の閾値に達しないが、最後の応答に属する最後のパケットの受信が完了した場合や、メモリ１０２上のバッファ領域が不足した場合や、データの受信が途中で中断された場合も、同様に、バッファに蓄積されたデータをすべて読み出して、ストレージ１０３に書き込む。バッファが、ストレージ１０３の読み書きの単位であるブロック長（例えば５１２バイト）の整数倍であり、ブロック長の整数倍で書き込みが行われるため、効率的なストレージへの書き込みが行うことができる。データ読み出しが完了したバッファは、いったん解放された後、応答１または応答２に属する後続のデータを書き込むために再利用される。

【００３１】

ステップＳ１０６～Ｓ１１４は、応答１（応答パケット１－１～応答パケット１－Ｘ）と応答２（応答パケット２－１～応答パケット２－Ｙ）の受信が完了するまで繰り返す。上述したように、ステップＳ１１２で、後述する本実施形態の方式を使用してメモリ１０２上のバッファへデータ書き込みを行うことで、ストレージ１０３に、応答１に属するデータの末尾と応答２に属するデータの先頭を連続したアドレスに書き込むことができる。すなわち、ストレージ１０３のあるブロック内では、応答１に属するデータの末尾がブロックの境界に一致しない限り、応答１に属するデータの末尾と、応答２に属するデータの先頭とが連続することになる。

【００３２】

以上が、データ受信装置１００の基本的な動作である。ここでは、２つのＴＣＰ接続を使って、ウェブページを取得する方法を述べたが、より多くの接続を使って取得するようにしてもよい。また、図３では、２つのＴＣＰ接続を使い、各接続で１つの取得要求を送信して、それぞれ応答を受信しているように図示しているが、この対応関係も変化しうる。例えば、１つのＴＣＰ接続を使って、複数の取得要求を送信し、複数の応答を受信するようにしてもよい。このような動作は、ＨＴＴＰの仕様に従って実行すればよいため、ここでは詳細に述べない。

【００３３】

以下、ステップＳ１１２の動作の詳細として、メモリ１０２上に確保したバッファに、データを格納していく処理について述べる。

【００３４】

図４Ａは、メモリ１０２に確保された１つのバッファＢ（１）を模式的に示している。図の左から右、上から下に向かうに従って、アドレスが増加するものとする。すなわち、先頭から隙間なくデータを格納していく場合には、左から右、上から下へと格納されることになる。また、バッファＢ（１）は、サイズＬを有しているとする。このサイズＬは、ストレージ１０３へのアクセス（読み出し、書き込み）が効率よく行われるサイズである。例えば、ブロックを単位としてアクセスするデバイスであれば、サイズＬは、ブロックサイズの整数倍に設定される。また、ストレージ１０３がキャッシュやバッファを有する場合には、キャッシュやバッファのサイズに合わせて設定してもよい。この際、各バッファで、サイズＬを統一する必要は必ずしもなく、ブロックサイズの整数倍であれば、複数のサイズを使い分けてもよい。例えば、ブロックサイズをL_bとしたときに、バッファサイズＬは、ｎ×L_b（ｎ＝自然数）であればよい。また、サイズＬはメモリ１０２のサイズを、コネクション数×２、またはコネクション数×２＋１で割ったサイズ以下で最も大きいｎ×L_b（ｎ＝自然数）と決定してもよい。コネクション数×２＋１とした場合には、ストレージ１０３にデータを書き出している間に受信したデータをバッファすることができる。また、Ｌを複数定め、サイズの異なるバッファを設定してもよい。優先度の高い応答にｎの大きいバッファを優先的に割り当ててもよい。優先度は取得要求によって決定してもよい。例えば、取得するデータの画像・テキストなどの種別、言語、更新頻度、サイズ、キャッシュの可否、キャッシュの有効期限などにより決定してもよい。

【００３５】

図３のステップＳ１０６で受信した応答パケット１－１の受信処理（Ｓ１１１）を、プロセッサ１０１が開始したとする。また、バッファＢ（１）には、何もデータが格納されていないとする。はじめにプロセッサ１０１は、受信した応答パケット１－１に基づき、応答１で受信するデータのサイズｄ１を求める。応答１で受信するデータのサイズｄ１とは、複数の応答パケット１－１～１－Ｘがある場合は、応答パケット１－１～１－Ｘで受信するデータの合計サイズを表す。ＨＴＴＰによる通信を例にすると、応答パケット１－１についてＴＣＰ層までの各層による受信処理を行った後、ＴＣＰのデータとして格納されているＨＴＴＰの情報を抽出し、ＨＴＴＰヘッダを解析することで（より具体的にはＣｏｎｔｅｎｔ－Ｌｅｎｇｔｈヘッダの値を取得することで）、応答１で受信するデータのサイズｄ１を求めることができる。

【００３６】

データサイズを求めたら、現在確保されているバッファの空きサイズを確認する。現在、バッファＢ（１）には何もデータが格納されていないので、空きサイズはＬとなる。この時、データサイズｄ１と、バッファサイズＬを比較して、現在のバッファに、当該データサイズｄ１のデータ（応答１に属する全データ）を格納可能か否か判断する。ｄ１≦Ｌであれば、現在のバッファＢ（１）に、データサイズｄ１のデータを格納可能なので、バッファＢ（１）の先頭から、応答パケット１のデータを格納する。その結果、図４Ｂのようになる。応答パケット１－１のデータがサイズｄ１に一致する場合は、応答１の受信はこれで完了である。応答パケット１－１のサイズがサイズｄ１未満であれば、後続の応答パケット１－２、・・・に含まれるデータを順次、連続したアドレスに格納していけばよい。

【００３７】

一方、応答１に属するデータのサイズｄ１＞バッファサイズＬであれば、確保したバッファＢ（１）には、応答１に属するデータをすべて格納できない。この場合、データサイズｄ１のバッファ長Ｌに対する商Ｎ１と剰余ｒ１を求める。そして、バッファＢ（１）の空き領域（この場合は先頭から）に、応答パケット１－１に属するデータを格納するとともに、バッファＢ（１）の残りの全領域を予約する。そして、図４Ｃに示すように、さらに新たなバッファＢ（２）を確保し、バッファＢ（２）の先頭からサイズｒ１の部分を、応答１に対する領域として予約しておく。なお、応答１に属するデータとこれまで受信した各応答（ここでは存在しない）に属するデータとの合計をバッファサイズＬで除算した値をＮＮ１と記載することがある（ここではＮＮ１はＮ１に一致する）。

【００３８】
データサイズｄ１≦バッファサイズＬであっても、データサイズｄ１＞バッファサイズＬであっても、複数のパケットに分割されて、応答１に属するデータが送信される場合は、各応答パケット１－１、応答パケット１－２・・・内のデータは、順次、バッファＢ（１）へ格納する。その際、直前の応答パケット１－１に属するデータに対して連続する位置に書き込んでいく。例えば、図４Ｃに示すように、応答パケット１－１で受信したデータと、応答パケット１－２で受信したデータを、順番に連続するアドレスに書き込んでいく。ｄ１－１は応答パケット１－１で受信したデータのサイズを表し、ｄ１－２は応答パケット１－２で受信したデータのサイズを表す。バッファＢ（１）が一杯になったら、バッファＢ（１）内の全データを、ストレージ１０３へ書き出し、バッファＢ（１）を解放し、バッファＢ（１）の先頭から、応答１についてさらに後続する応答パケットのデータを書き込んでいく（商Ｎ１が２以上の場合）。詳細は、後に図４Ｆで説明する。ただし、応答１に属するデータの末尾のサイズｒ１分のデータは、バッファＢ（２）の予約した先頭のサイズｒ１分の領域に書き込む。
【００３９】
なお、図４Ｃでは、バッファＢ（１）とバッファＢ（２）が連続しておらず、互いに離れた位置にあるが、バッファＢ（１）とバッファＢ（２）が連続してもかまわない。
【００４０】
図４Ａ、図４Ｂ、図４Ｃでは、応答１に対してバッファを確保する例を示したが、応答２に対しても同様にして、バッファを確保する。図４Ｃの状態で、応答２に属する最初の応答パケット（応答パケット２－１）を受信したと仮定する。応答１の場合と同様に、応答２に属するデータサイズｄ２を求める。応答１の場合と同様に、バッファの現在の空き領域に、当該サイズｄ２分のデータを格納できるかどうかを確認する。図４Ｃの状態で、バッファＢ（１）の一部は空いているが、応答１のために予約されているため、バッファＢ（２）の空き領域を使用する。バッファＢ（２）の空き領域のサイズは、Ｌ－ｒ１（＝Ｌ’）である。
【００４１】
ｄ２≦Ｌ’であれば、応答２に属するデータは、図４Ｄに示すように、バッファＢ（２）の空き領域にすべて格納できる。応答パケット２－１のデータがサイズｄ２に一致する場合、応答２の受信はこれで完了である。応答パケット２－１のデータサイズがｄ２未満であれば、後続の応答パケット２－２、・・・に含まれるデータを、応答パケット２－１のデータに対して順次、連続したアドレスに格納する。
【００４２】
一方、ｄ２＞Ｌ’であれば、データサイズｄ２から、バッファＢ（２）の空き領域サイズＬ’を除いた長さに対して、サイズＬに対する商（Ｎ２とする）と剰余（ｒ２とする）を計算する。すなわち、Ｎ２＝（ｄ２－Ｌ’）／Ｌ、ｒ２＝（ｄ２－Ｌ’） ｍｏｄ Ｌとする。Ｎ２が商、ｒ２が剰余である。そして、図４Ｅに示すように、新たにバッファＢ（３）を確保して、先頭からサイズｒ２を、応答２用に予約済みにする。図示のｄ２－１は、応答パケット２－１に含まれるデータのサイズを表している。応答パケット２－１に後続する応答パケット２－２・・・のデータは、直前の応答パケット２－１に属するデータに対して連続する位置に書き込んでいく。バッファＢ（２）が一杯になったら、バッファＢ（２）内の全データをストレージ１０３へ書き出し（応答１に属するデータの末尾のサイズｒ１のデータもバッファＢ（２）の先頭に書き込み済みとする）、バッファＢ（２）を解放し、バッファＢ（２）の先頭から、応答２についてさらに後続する応答パケットのデータを書き込んでいく（Ｎ２が１以上の場合）。ただし、応答２に属するデータの末尾のサイズｒ２分のデータは、バッファＢ（３）の予約した先頭のサイズｒ２分の領域に書き込む。なお、応答２に属するデータとこれまで受信した各応答（ここでは応答１のみ）に属するデータとの合計をバッファサイズＬで除算した値を、ＮＮ２と記載することがある。ここで、図４Ｃで応答１をバッファに保存中に使用中のバッファはＢ（１）とＢ（２）の２つである。この状態で応答２を受信開始すると、図４Ｇに示すように追加のバ

ッファ、バッファＢ（３）を必要とする。Ｎ２＞１の場合、応答２を保存中に必要となるバッファの数はさらに一つ増え、バッファＢ（４）を必要とし、合計４つのバッファが必要となる。なお、パケット２－６のデータはバッファＢ（２）とバッファＢ（４）にまたがって格納されている。このように、応答を保存するのに必要なバッファ数は応答数×２である。仮に応答数×２＋１のバッファを用意すれば、ストレージにデータを書き出し中にもバッファをすることが可能である。

【００４３】

ここでバッファ管理情報について説明する。図５は、バッファ管理情報の例を示す。メモリ１０２上に確保するバッファは、バッファ管理情報で管理される。図示のバッファ管理情報は、応答１および応答２に対するものであり、図４Ｅの状態に対応している。

【００４４】

応答１に対して、バッファＢ（１）とバッファＢ（２）の各々について、使用済みサイズ、予約済みサイズ、オフセット、メモリ１０２におけるバッファの位置を表すポインタを保持している。応答１に属するデータの先頭からの長さ（応答１に属するデータを格納するストレージの先頭位置からの長さ）を示すオフセットは、バッファサイズの整数倍（０以上）の単位で設定される。応答２についても同様に、バッファＢ（２）とバッファＢ（３）の各々について、使用済みサイズ、予約済みサイズ、応答１に属するデータの先頭からの長さ（応答１に属するデータを格納するストレージの先頭位置からの長さ）を示すオフセット、メモリ１０２におけるバッファの位置を表すポインタを保持している。バッファへアクセスする際は、ポインタを利用してバッファの位置を特定する。図５におけるＮＮ１は、前述したように、応答１に属するデータをバッファサイズＬで除算した値（ここではＮ１に一致）である。ＮＮ２は、前述したように、応答２に属するデータとこれまで受信した各応答（ここでは応答１のみ）に属するデータとの合計をバッファサイズＬで除算した値である。なお、バッファサイズがバッファによって異なる場合も、同様の考えでオフセットを計算すればよい。

【００４５】

バッファ管理情報における使用済みサイズは、実際にデータが書き込まれているサイズである。使用済みサイズは、バッファにデータが書き込まれると更新される。例えば、図５の状態において、バッファＢ（２）に応答１に属する末尾のサイズｒ１分のデータが書き込まれた場合、応答１に対するバッファＢ（２）の使用済みサイズはｒ１に更新され、応答２に対するバッファＢ（２）の使用済みサイズは、ｒ１＋ｄ２－１に更新される。予約済みサイズは、該当する応答によって予約されているサイズである。予約している領域は、より早い応答（先に予約した応答）ほど、バッファの先頭側の領域を予約しているとする。例えば、バッファＢ（２）では、応答１が先頭のサイズｒ１の領域、応答２が、当該サイズｒ１の領域に後続するサイズＬ’の領域を予約しているとする。オフセットは、上述したように、応答１に属するデータの先頭からの位置を、バッファサイズの整数倍（０以上）で示す。バッファからストレージ１０３への書き出しが行われると、オフセットの値は、バッファサイズＬ分、インクリメントされる。例えば図５の状態において、バッファＢ（１）内のデータがストレージ１０３へ書き出され、バッファＢ（１）が解放され、引き続きバッファＢ（１）の先頭から応答１に属する続きのデータを書き込む場合、オフセットは、０からＬに更新される。次も同様にして、バッファＢ（１）へのデータ書き込み、データ書き出しおよびバッファ解放が行われれば、オフセット値は、２×Ｌに更新される。

【００４６】

図５では、応答の個数が２つ（応答１および応答２）の場合だが、応答３、応答４・・・といったように、応答の個数が増えれば、それに応じて、バッファ管理情報が追加される。なお、バッファ管理情報の更新は、後述するように、バッファが新規に設定された際、バッファへのデータ書込が行われた際、バッファからストレージ１０３への書き出しが行われた際などに行われる。

【００４７】

ここではバッファ管理情報を、リストの形で保持しているが、テーブル形式で保持してもよい。または、バッファ管理情報におけるバッファ毎の情報を、メモリ１０２における各バッファの先頭に配置してもよく、この場合、ポインタの情報のみを、リストまたはテーブル等の形式で別途管理してもよい。また、ここで述べた以外の形式で、例えば、ビットマップなどを用いて使用済みメモリと空きメモリのバッファ管理情報を管理してもよい。例えば、先頭のアドレスから、アドレス毎、または任意のバイト数毎にバッファ領域が使用済みの場合１を、未使用の場合は０としてビットを並べて、ビットマップを構成したバッファ管理情報を管理してもよい。

【００４８】

図４Ｆは、図４Ｅの状態の後、応答１に関するデータの受信が進んだ状態を示している。応答パケット１－１、応答パケット１－２の後、応答パケット１－３、応答パケット１－４、応答パケット１－５、応答パケット１－６、応答パケット１－７、応答パケット１－８を受信したとする。応答パケット１－３、応答パケット１－４、応答パケット１－５、応答パケット１－６、応答パケット１－７、応答パケット１－８の各々のデータの長さは、ｄ１－３、ｄ１－４、ｄ１－５、ｄ１－６、ｄ１－７、ｄ１－８である。応答パケット１－３のデータ（データサイズｄ１－３）を、応答パケット１－２のデータに続いて格納し、応答パケット１－４のデータ（データサイズｄ１－４）を、応答パケット１－３のデータに続いて格納し、以降同様にして、応答パケット１－５以降のデータを格納していく。

【００４９】

ここで、最後に受信した応答パケット１－８のデータの末尾が、バッファＢ（１）の末尾に一致するとは限らない。すなわち、応答パケット１－７のデータを格納した後のバッファＢ（１）の空き領域サイズ、すなわち、バッファＢ（１）のサイズＬから、応答パケット１－１～応答パケット１－７のデータの合計サイズを減じたサイズ、である端数サイズが、応答パケット１－８のデータの長さｄ１－８に一致するとは限らない。上述したｒ１が０でない場合は、一致せず、本例では一致しない。この場合、バッファＢ（１）には、応答１－８のデータのうち、先頭から、上記の端数サイズ（ｄ１－８（１）とする）までのデータを、バッファＢ（１）の最後の端数サイズの領域に格納し、残りのサイズ（ｄ１－８（２））のデータを、バッファＢ（２）の予約領域ｒ１に格納する。なお、サイズｄ１－８（２）は、ｒ１に一致する。

【００５０】

図４Ｆの例では、応答１に属するデータサイズが、Ｌ＋ｒ１の例であるが、２Ｌ＋ｒ１の場合は、応答パケットが順序通り到着すれば、バッファＢ（１）へのサイズＬ分のデータ書き込みが２回行われた後（つまりバッファＢ（１）が２回使い回された後）、最後のサイズｒ１分のデータが、バッファＢ（２）の先頭のサイズｒ１の予約領域に書き込まれることとなる。

【００５１】

図３のステップＳ１１３、Ｓ１１４の説明で述べたとおり、データをバッファに書き込んだ際に、当該バッファに蓄積されたデータ量が、閾値（バッファサイズＬ）に達したかを判断する。すなわち、ひとつのバッファが全て埋まったかどうかを、当該バッファへデータを格納した際に判定する。閾値に達した場合、当該バッファに格納されている全データは、当該バッファから読み出されて、ストレージ１０３内の所定の位置へ書き込まれる。その後、当該バッファ（メモリ）は解放され、未使用の状態へと戻される。ここで、バッファの解放とは、使用していた領域を他の用途に転用できる状態に戻す処理としても良いし、バッファ管理情報に含まれる使用済みサイズをゼロに設定する処理としてもよい。また、ストレージ１０３内の「所定の位置」は、バッファ管理情報に含まれる“オフセット”に基づいて決定される。例えば、オフセットがゼロであれば、ストレージ１０３内の基準位置に、バッファから読み出した全データを書き込み、オフセットの値がＸであれば、基準位置からＸバイトだけ進んだ位置に、当該全データを書き込む。

【００５２】

図６は、応答１および応答２の受信処理が完了したときのストレージ１０３の書き込み状態を示す。図６の例は、応答１のデータサイズｄ１に対してｄ１＞Ｌが成立し、応答２のデータサイズｄ２に対してｄ２＞Ｌ’が成立する場合であり、Ｎ１＝３、Ｎ２＝１である。すなわち、データサイズｄ１＝３×Ｌ＋ｒ１、データサイズｄ２＝Ｌ’＋１×Ｌ＋ｒ２である。バッファＢ（１）に対する書き込み、読み出しおよび解放が３回繰り返され、各回に対応してストレージ１０３の領域Ａ～Ｃへ、応答１に属するデータが書き込まれる。また、バッファＢ（２）に対する書き込み、読み出しおよび解放が２回繰り返され、各回に対応してストレージ１０３における領域Ｄおよび領域Ｅへデータが書き込まれる。領域Ｄの先頭側の領域Ｄ（１）に格納されたデータは、応答１に属するデータの末尾のサイズｒ１分のデータである。領域Ｄの領域Ｄ（２）に格納されたデータは、応答２に属するデータの先頭側のデータである。ストレージ１０３の領域Ｅに書き込まれたデータは、応答２に属する続きのデータである。領域Ｆに書き込まれたデータは、応答２に属する末尾のサイズｒ２分のデータである。ここでは２つのＴＣＰ接続を同時に使って、２つの応答のデータを取得する場合を例に述べたが、取得する応答数が増えても同じように実行することができる。

【００５３】

図７に、図１のデータ受信装置による処理の基本的な流れを示すフローチャートである。なお、このフローチャートの動作は、一つの実装例であり、基本的な考え方が同じであれば、動作の細部が異なっていてもよい。

【００５４】

受信したパケットが、応答Ｐの最初のパケットＰ１かどうかを確認する（ステップＳ７０１）。図３の例であれば、応答１もしくは応答２の最初のパケット１－１、２－１かどうかを確認する。なお、最初のパケットという表現は便宜上のもので、より正確には、応答Ｐのデータサイズ情報が含まれるパケットであるかどうかを確認する。また、これまでは応答Ｐに属するパケットをＰ－１、Ｐ－２、・・・Ｐ－Ｘといった表記で表したが、本フローの説明では、「－」を削除して、Ｐ１、Ｐ２、・・・ＰＸといった表記を用いる。

【００５５】

受信したパケットが、最初のパケットであれば（ステップＳ７０１－ＹＥＳ）、現時点で管理しているバッファから、オフセットが最大のバッファ（バッファ$Ｂ_{Ｌ}$と記述する）を見つけ、そのバッファ$Ｂ_{Ｌ}$を、操作対象のバッファＢとして決定する（ステップＳ７０２）。図では、これを「Ｂ←オフセットが最大のバッファ$Ｂ_{Ｌ}$」と表現している。ステップＳ７０２の具体例として、応答Ｐが応答１の場合、バッファ$Ｂ_{Ｌ}$として、バッファＢ（１）が特定される（バッファＢ（１）のみ存在する）。応答Ｐが応答２の場合、バッファ$Ｂ_{Ｌ}$として、バッファＢ（２）（この時点でバッファＢ（１）とＢ（２）が管理されており、バッファＢ（２）のオフセットＮ１×Ｌが最大）が特定される。

【００５６】

一方、受信したパケットが、最初のパケットでなければ（ステップＳ７０１－ＮＯ）、応答Ｐに属するデータを格納中のバッファ（バッファ$Ｂ_{Ｃ}$と記述する）を見つけ、そのバッファ$Ｂ_{Ｃ}$を、操作対象のバッファＢとして決定する（ステップＳ７０３）。図では、これを「Ｂ←オフセットが最大のバッファ$Ｂ_{Ｃ}$」と表現している。

【００５７】

操作対象のバッファＢが確定すると、バッファＢの空き領域の有無を確認する。すなわち、バッファＢの空き領域のサイズ（Ｌｅｆｔ（Ｂ）と記述する）が０より大きいか否かを確認する。空き領域が無い場合、すなわち空き領域のサイズＬｅｆｔ（Ｂ）が０の場合（ステップＳ７０４－ＮＯ）、現在のバッファＢ内の全データをストレージ１０３に書き出して当該バッファＢを解放し、解放されたバッファＢを新たなバッファＢとする（ステップＳ７０５）。ステップＳ７０５の処理の詳細は、「バッファＢの書き出しと新規割り当て処理」として後に説明する。一方、バッファＢに空き領域がある場合、すなわち、空き領域のサイズＬｅｆｔ（Ｂ）が０より大きい場合には（ステップＳ７０４－ＹＥＳ）、ステップＳ７０５の処理を行わずに、ステップＳ７０６へ進む。

【００５８】
　ステップＳ７０６では、ステップＳ７０１で受信したパケット（処理中のパケットＰｎ。ｎは１以上の整数）に含まれるデータが、バッファＢの空き領域に格納できるかどうかを確認する。パケットＰｎに含まれるデータのサイズを、Ｄａｔａｌｅｎ（Ｐｎ）と表す。バッファＢの空き領域に、パケットＰｎに含まれるデータの全体を格納できる場合（ステップＳ７０６－ＹＥＳ）、すなわち、バッファＢの空き領域のサイズＬｅｆｔ（Ｂ）がＤａｔａｌｅｎ（Ｐｎ）以上の場合、後述するステップＳ７１１に進む。バッファＢの空き領域に、パケットＰｎに含まれるデータの一部しか格納できない場合（ステップＳ７０６－ＮＯ）、すなわち、Ｄａｔａｌｅｎ（Ｐｎ）がバッファＢの空き領域のサイズＬｅｆｔ（Ｂ）未満の場合、ステップＳ７０７に進む。
【００５９】
　ステップＳ７０７では、バッファＢの空き領域のサイズＬｅｆｔ（Ｂ）の値を、パラメータ（ｗｒｉｔｔｅｎＬｅｎ）に保存する。そして、パケットＰｎのデータサイズＤａｔａｌｅｎ（Ｐｎ）から、バッファＢの空き領域のサイズ（Ｌｅｆｔ（Ｂ））を減じることで、Ｄａｔａｌｅｎ（Ｐｎ）の値を更新する（ステップＳ７０８）。
【００６０】
　パケットＰｎのデータのうち、バッファＢの空き領域のサイズＬｅｆｔ（Ｂ）分のデータを先頭から特定して、当該データをバッファＢの空き領域に格納する（ステップＳ７０９）。そして、バッファＢ内の全データを、ストレージ２０４へ書き出し、当該バッファＢを解放し、解放されたバッファＢを新たなバッファＢとする（ステップＳ７１０）。これにより、パケットＰｎのデータのうち残りのデータ（ステップＳ７０８で更新されたサイズＤａｔａｌｅｎ（Ｐｎ）分のデータ）を格納するための新たなバッファの確保を行う。ステップＳ７１０の処理の詳細は、「バッファＢの書き出しと新規割り当て処理」として後に説明する。
【００６１】
　ステップＳ７１０の後、再度、ステップＳ７０１で受信した、応答Ｐに属するパケットが、応答Ｐの最初のパケットＰ１であったかを判断する（ステップＳ７１１）。最初のパケットＰ１でなかった場合には（ステップＳ７１１－ＮＯ）、後述するステップＳ７１６に進む。
【００６２】
　ステップＳ７０１で受信したパケットが、応答Ｐの最初のパケットＰ１であった場合には（ステップＳ７１１－ＹＥＳ）、応答Ｐに属するデータのサイズ（応答Ｐ（パケットＰ１～ＰＸ）で受信する予定の全データ長）に応じて、バッファを新たに確保する。具体的には、以下のステップＳ７１２～Ｓ７１５を行う。応答Ｐに属するデータのサイズを、パラメータＣｏｎｔｅｎｔＬｅｎ（Ｐ）によって表すとする。ＣｏｎｔｅｎｔＬｅｎ（Ｐ）から、ｗｒｉｔｔｅｎＬｅｎ（ステップＳ７０７で求めた値。最初のパケットＰ１のデータがバッファＢの空き領域より大きい場合に、当該バッファＢの空き領域に書き込んだデータサイズ。すなわち当該バッファＢの空き領域のサイズ）を減算し、バッファ長Ｌで除することで、商Ｎと剰余ｒを求める（ステップＳ７１２）。すなわち、（ＣｏｎｔｅｎｔＬｅｎ（Ｐ）－ｗｒｉｔｔｅｎＬｅｎ）／Ｌにより商Ｎを計算し、（ＣｏｎｔｅｎｔＬｅｎ（Ｐ）－ｗｒｉｔｔｅｎＬｅｎ）％Ｌにより、剰余ｒを計算する。
【００６３】
　そして、商Ｎが０でなければ（すなわち商Ｎが０より大きければ）（ステップＳ７１３－ＹＥＳ）、追加のバッファ（バッファＢｒとする）をメモリ１０２上に確保し（ステップＳ７１４）、バッファＢｒの管理情報（図５参照）を初期化し（ステップＳ７１５）、次のステップＳ７１６に進む。本処理は、剰余ｒが０であった場合も行ってよい。ステップＳ７１４の処理は、「バッファＢｒの新規割り当て処理」として、ステップＳ７１５の処理は、「バッファＢｒの管理情報の更新処理」として、後に詳述する。
【００６４】
　一方、ステップＳ７１２で計算した商Ｎが０であれば（ステップＳ７１３－ＮＯ）、ス

テップＳ７１４、Ｓ７１５の処理を行うことなく、ステップＳ７１６に進む。

【００６５】

ステップＳ７１６では、操作対象のバッファＢに、パケットＰｎのデータを格納する。そして、バッファＢのバッファ管理情報を更新し（ステップＳ７１７）、処理を終了する。ステップＳ７１７の処理の詳細は、「バッファＢの管理情報の更新」として後に説明する

【００６６】

以下、上述した一連の処理に含まれるいくつかのステップについて、さらに詳細に説明する。

【００６７】

図８（Ａ）は、ステップＳ７０５とステップＳ７１０で行う「バッファＢの書き出しと新規割り当て処理」のフローチャートである。操作対象のバッファＢの管理情報におけるオフセットＯｆｆｓｅｔ（Ｂ）に、バッファ長Ｌを加えた値を、ｏｆｆｓｅｔとして保存する（ステップＳ８０１）。図では、これを「ｏｆｆｓｅｔ←Ｏｆｆｓｅｔ（Ｂ）＋Ｌ」と表現している。

【００６８】

次に、バッファＢ内の全データを、ストレージ１０３に書き出して、当該バッファＢを解放する（ステップＳ８０２）。

【００６９】

その解放されたバッファＢを、新しいバッファ$Ｂ_{N}$として確保し（ステップＳ８０３）、当該バッファ$Ｂ_{N}$のバッファ管理情報を初期化する（ステップＳ８０４）。具体的に、まず更新用パラメータ（Ｐ、ｄ、ｒ、ｏ）を設定する。左から、応答の識別子（応答Ｐの場合はＰ）、使用済みサイズ、予約済みサイズ、オフセットである。ここでは、（Ｐ、０、＊、ｏｆｆｓｅｔ）と設定する。＊は、必要な値を設定すればよい。一例として、ステップＳ７１４でバッファＢｒが確保されており、現在対象とするバッファＢが当該バッファＢｒでない場合は、Ｌを設定することが考えられる。あるいは、応答に属する以降のデータを当該バッファＢ内にすべて書き込み可能な場合は、そのデータのサイズを設定することもある。次に、図８（Ｃ）の「バッファＢの更新情報の更新」フローを実行する。すなわち、ステップＳ８２１で、バッファＢの管理情報の使用済みサイズＵｓｅｄ（Ｂ）をｄ（ここでは０）に更新し、ステップＳ８２２で、バッファＢの管理情報の予約済みサイズＲｅｓｅｒｖｅｄ（Ｂ）をｒ（ここでは０）に更新し、ステップＳ８２３で、オフセットＯｆｆｓｅｔ（Ｂ）をｏ（ここではステップＳ８０１で保存したｏｆｆｓｅｔ）に更新する。図８（Ａ）に戻り、ステップＳ８０５では、ステップＳ８０３で確保したバッファ$Ｂ_{N}$を呼び出し元に返し（ステップＳ８０５）、処理を終了する。以降、確保したバッファ$Ｂ_{N}$が、操作対象のバッファＢとして扱われる。

【００７０】

図８（Ｂ）は、ステップＳ７１４で行う「バッファ$Ｂ_{r}$の新規割り当て処理」のフローチャートである。バッファ$Ｂ_{r}$は、ある応答Ｐの最初のパケットＰ１を受信した際に、前述した端数データ（剰余サイズ分のデータ）を格納するために、必要に応じて確保される。本処理では、新しいバッファＢｒを確保し（ステップＳ８１１）、そのバッファＢｒを、呼び出し元に返す（ステップＳ８１２）。バッファＢｒは、その先頭部分に、応答Ｐに属するデータのうち、末尾の端数データ（剰余サイズ分のデータ）を格納するための予約領域を持つバッファである。このため、ステップＳ７１５では、更新用パラメータ（Ｐ、ｄ、ｒ、ｏ）として（Ｐ、０、ｒ、ＮＮ）を設定して、図８（Ｃ）の「バッファＢの更新情報の更新」フローを実行する。ＮＮはオフセットであり、（今回の応答Ｐに属するデータとこれまで受信した各応答に属するデータとの合計をバッファサイズＬで除算した値ＮＮ）×Ｌである。これにより、バッファＢｒの管理情報の予約済みサイズはｒ、オフセットはＮＮ×Ｌに設定される。なお、ｒは、ステップＳ７１２で計算された剰余である。

【００７１】

図８（Ｃ）は、「バッファＢ（Ｂｒの場合も含む）の管理情報の更新」フローを示し、

前述したように、図７のステップＳ７１５や図８のステップＳ８０４で実行され、さらにステップＳ７１７で実行される。ステップＳ７１５、Ｓ８０４で行う場合の本フローの処理は既に説明したため、ここではステップＳ７１７で行う本フローの処理を説明する。

【００７２】

バッファＢの管理情報は、図５に示したように、応答Ｐに対して、「使用済みサイズ」、「予約済みサイズ」、「オフセット」、「バッファへのポインタ」を含む。なお、前述したように、「バッファへのポインタ」の更新の説明は省略する。バッファに関連づけられている応答、使用済みサイズ、予約済みサイズ、オフセットを、（Ｐ、ｄ、ｒ、ｏ）と表し、バッファ管理情報の応答「Ｐ」、使用済みサイズ「Ｕｓｅｄ（Ｂ）」、予約済みサイズ「Ｒｅｓｅｒｖｅｄ（Ｂ）」、オフセット「Ｏｆｆｓｅｔ（Ｂ）」を、Ｐ、ｄ、ｒ、ｏにより更新する。そして、図７のステップＳ７１７の処理では、Ｕｓｅｄ（Ｂ）にＤａｔａＬｅｎ（Ｐｎ）を加算することでＵｓｅｄ（Ｂ）を更新し、Ｐ，Ｒｅｓｅｒｖｅｄ（Ｂ），Ｏｆｆｓｅｔ（Ｂ）は、現状の値を維持する。

【００７３】

図３に示したシーケンスでは、２つの取得要求しか送信していないが、送信する取得要求の数は、プロセッサ１０１にて実行するアプリケーションによって変化する。その際、最後の取得要求に対する最後の応答パケットを受信した際には、プロセッサ１０１は、確保されているバッファ内のデータを、（バッファ内のデータが閾値に達していなくても）ストレージ１０３に書き出す処理を行い、バッファを解放する。

【００７４】

また、通信の過程において、ある応答の受信が正常に完了できなかった場合（例えば何らかの理由で、受信途中でサーバ装置２０１への到達性がなくなった場合など）には、受信が完了している部分までをストレージ１０３に書き出し、エラーが発生した部分は破棄するようにしてもよい。その際、ストレージ１０３への書き出しは、パケット単位ではなく、応答単位で行うようにしてもよい。例えば、応答Ｐと応答Ｑを同時に受信しているときに、応答Ｑの受信が正常に完了する前に、エラーが発生したとする。その場合、応答Ｐの完了を待って、応答Ｐのみを書き出すようにしてもよいし、エラーが発生した時点で受信完了していた（ＰでもＱでもない）他の応答だけを書き出すようにしてもよい。なお、パケット単位で書き出すようにしてもよい。ある応答が受信途中であることは、各バッファの管理情報を参照することで検出可能である。なお、ここで述べたような例外処理を行う場合は、ストレージへの書き出しは、必ずしもブロックサイズの整数倍のデータになるとは限らない。

【００７５】

また、前述した図６のように応答１および応答２を受信して、ストレージにデータを書き出す場合において、領域Ｄ（２）に格納される、応答２に属するデータの受信が完了した時点で、領域Ｄ（１）に格納される、応答１に属するデータの受信がまだ行われていない場合があるとする。このときは、領域Ｄ（１）に格納されるデータの受信が完了するまで、バッファの書き出しを待機するようにしてもよい。この場合、応答２に属する後続のデータは一時的に別の領域に退避しておき、バッファの書き出しおよび解放が完了したら、当該退避しておいたデータをバッファに書き込んでもよい。

【００７６】

以上、本実施形態によれば、ネットワークを介して受信する応答に属するデータをストレージに保存する際に、不規則に受信され得る複数の応答に属するデータを、ストレージへのアクセス効率が高くなるサイズ（ブロックサイズの整数倍のサイズ）のバッファに一度格納することで、データ受信およびストレージ保存の高速化を図ることができる。この際、保存するデータのサイズとバッファ長に基づいて、ある応答に対して、予約済み領域を持ったバッファを確保し、予約領域の直後から次の応答に属するデータを保存するようにしたことで、複数の応答を同時に受信したとしても、各々の応答に属するデータを正しいバッファに保存できる。さらに、応答に属するデータがバッファサイズよりも大きい場合であっても、小さなバッファだけでデータ受信とストレージへの保存を管理できるため

、サーバからの情報（ウェブページ等）の取得に際し、必要なメモリ量を削減することができる。

【００７７】

＜第２の実施形態＞

本実施形態では、第１の実施形態と同じ動作を実現する機能を、モジュールとして実装する場合である。本モジュールは、外部のメインプロセッサとの間で制御情報を交換する機能、並びにネットワークを介して送受信するデータを当該メインプロセッサと交換する機能とを追加で具備する。

【００７８】

図９に第２の実施形態に係るモジュールとメインプロセッサとを備えたデータ受信装置のブロック図を示す。モジュール９００は、本実施形態の主機能を具備する部分である。モジュール９００は、プロセッサ９０１、メモリ９０２、ストレージ９０３、通信インタフェース９０４、ホストインタフェース９０５を備える。モジュール９００は、ネットワークカード等の通信カードとして構成されてもよい。

【００７９】

プロセッサ９０１、メモリ９０２、ストレージ９０３、通信インタフェース９０４は、第１の実施形態におけるプロセッサ１０１、メモリ１０２、ストレージ１０３、通信インタフェース１０４と同じ機能を具備し、基本的に同じ動作をする。ただし、プロセッサ９０１については、メインプロセッサ９０６との間でデータを交換する機能が追加されている。ホストインタフェース９０５は、モジュール９００とメインプロセッサ９０６とを接続する機能を提供する。その実装にはＳＤＩＯやＵＳＢなど外部バスの仕様に従うものであっても、ＰＣＩ　Ｅｘｐｒｅｓｓなど内部バスの仕様に従うものであってもよい。メインプロセッサ９０６は、ＯＳやアプリケーションソフトが動作しており、モジュール９００を利用するためのデバイスドライバや通信アプリケーションなども動作する。

【００８０】

なお、図９では、メインプロセッサ９０６が、モジュール９００とバスを介して直接接続するように図示しているが、メインプロセッサ９０６側もホストインタフェースなどを介して、モジュール９００と接続するようにしてもよい。また、図９では、モジュール９００の周辺装置として、プロセッサ９０６以外の装置は示していないが、計算機として動作するために必要なメモリや表示装置など、必要な周辺装置は、適宜接続されているものとする。

【００８１】

モジュール９００は、メインプロセッサ９０６からの指示により動作を開始する。具体的には、メインプロセッサ９０６が、第１の実施形態で述べた「取得要求」に相当する指示を、ホストインタフェース部９０５経由で、プロセッサ９０１に指示する。プロセッサ９０１は、指示の内容を把握し、第１の実施形態と同様の動作を行う。すなわち、プロセッサ９０１は、メモリ９０２において適切なバッファ管理を行いながら、取得要求により要求されたデータを外部のサーバから取得して、ストレージ９０３にデータを蓄積していく。

【００８２】

また、モジュール９００は、メインプロセッサ９０６からの指示により、ストレージ９０３に蓄積したデータを、メインプロセッサ９０６に転送する機能を具備する。本実施形態のバッファ管理方式に従って、プロセッサ９０６からの連続する取得要求により取得されたデータは、ストレージ９０３上で連続する領域に記録されている。そのため、ストレージ９０３を適切に走査し、対象となる領域を連続して読み出すことで、メインプロセッサ９０６から指示されたデータを効率的に取得できる。なお、情報の局所性を活用し、プロセッサ９０６からの読み出し指示を受ける前に、ストレージ９０３上の連続する領域を複数読み出し、事前にプロセッサ９０６（もしくはプロセッサ９０６と接続するメモリやその他の記憶装置）に、読み出したデータを転送するようにしてもよい。

【００８３】

なお、モジュール９００とメインプロセッサ９０６との間で交換される指示は、ホストインタフェース９０５上で定義された制御コマンドを用いてもよいし（例えば、ＳＤインタフェース仕様で規定される各コマンド群）、制御コマンド上に新たにコマンド体系を構築してもよい（例えば、ＳＤインタフェース仕様などで規定されるベンダー独自コマンド、ベンダー独自フィールドを活用したもの）。また、メインプロセッサ９０６が、何らかの形で、ストレージ９０３を読み書きできるのであれば、メインプロセッサ９０６がプロセッサ９０１に対する指示をストレージ９０３に格納し、プロセッサ９０１がその格納された指示を読み出して、解釈し、実行するようにしてもよい。例えば、指示を含むファイルをストレージ９０３に保存し、その指示の実行結果を、同一もしくは別のファイルとしてストレージ９０３に保存し、メインプロセッサ９０１がこのファイルを読み出すようにしてもよい。

【００８４】

また、モジュール９００は、メインプロセッサ９０６からの指示を受けるに先立って、メインプロセッサ９０６の制御を受けて通電が開始され、もしくは低消費電力な状態から通常の動作が可能な状態へと遷移してもよい。そして、モジュール９００は、一連の処理が完了すると（つまりストレージ９０３へのデータ保存が完了すると）、自発的に電源をオフにしたり、低消費電力な状態へと遷移したりするようにしてもよい。低消費電力な状態とはモジュール９００内の一部のブロックの電力供給を停止してもよいし、プロセッサ９０１の動作クロックを低下させてもよいし、その他の方法でもよい。さらに、ストレージ９０３へのデータ保存の完了の通知をメインプロセッサ９０６に出すようにしてもよい。これらの仕組みにより、モジュール９００の消費エネルギーを、削減することも可能となる。

【００８５】

＜第３の実施形態＞

第１および第２の実施形態では、各取得要求間に、独立もしくは従属などの関係性は特定されていなかった。これに対して、本実施形態では、各取得要求に主従関係を追加する。すなわち、最初に発生した取得要求と、それから派生した取得要求とで取得要求グループを形成し、少なくとも取得要求グループ内の各取得要求に応じて取得されたデータは、ストレージ１０３もしくはストレージ９０３上で連続した領域に保存されるようにする。

最初に発生した取得要求とそこから派生した取得要求とは、最初の取得要求を処理した結果として取得が必然であるデータが存在することが確認された場合に、後者の情報の取得要求を、派生した取得要求とする。このような派生した取得要求は、例えばウェブページの取得要求に見ることができる。ウェブページの取得では、最初のＨＴＭＬを取得・解析するとＨＴＭＬファイルから参照されているスタイルシートやスクリプトファイル、画像ファイルなどのデータを新たに取得する必要が生じる。本実施形態では、これらのデータの取得要求を一つの取得要求グループとして捉え、ストレージ１０３もしくはストレージ９０３上で連続する領域に、これらのデータが保存されるようにする。

【００８６】

本実施形態の機能は、図１で示した第１の実施形態、および図９で示した第２の実施形態のいずれにも適用可能である。初めに第１の実施形態に適用する場合について述べ、次に第２の実施形態に適用する場合について述べる。

【００８７】

（第１の実施形態に適用する場合）

プロセッサ１０１は、情報（ＨＴＭＬ）等のデータの取得等の機能に加え、取得したデータ（ＨＴＭＬ）を解析する機能、解析の結果として当該データから参照されているＵＲＬを抽出する機能（例えば、スタイルシートやスクリプトファイル、画像ファイルなど）、抽出したＵＲＬで特定されるデータを取得する機能を具備している。これらの機能は、ウェブブラウザのようにユーザインタフェースを備えたアプリケーション（アプリ）として実現されていてもよいし、ユーザインタフェースを備えないバックエンドのアプリとして実現されていてもよい。バックエンドのアプリの場合、ウェブブラウザなどがフロント

エンドとして動作する。ここではバックエンドアプリとして実現する場合について述べるが、ユーザインタフェースを搭載するアプリとして実現してもよい。

【００８８】

図１０は、第３の実施形態に係る動作のシーケンス図である。フロントエンド（ウェブブラウザ）で取得要求が発生し（Ｓ２０１）、フロントエンドがバックエンドアプリに対して、取得要求を送信する（Ｓ２０２）。この要求は、通常、ＨＴＭＬファイルに対する取得要求である。本実施形態を実現するバックエンドアプリが、この取得要求を受信すると、当該取得要求の取得対象となるデータ（以下、独立情報と呼ぶ）をサーバから取得する（Ｓ２０２、Ｓ２０３、Ｓ２０４、Ｓ２０５、Ｓ２０６）。その後、バックエンドアプリが、取得した独立情報を解析し（Ｓ２０７、Ｓ２０８）、当該独立情報のバッファへの書き込み、当該バッファ内の書き込み合計サイズの閾値判定、閾値に達した場合のストレージへの書き込み処理等を行い（Ｓ２０９、Ｓ２１０）、さらにこれに並行して、当該独立情報に含まれるＵＲＬに参照されるデータ（以下、従属情報と呼ぶ）を自発的に取得する（Ｓ２１１）。従属情報の取得、バッファ書き込み、閾値判定、ストレージ書き込み等は、独立情報と同様である。すなわち、独立情報から参照される複数の従属情報を、サーバから取得し、取得した複数の従属情報を、バッファを介して、ストレージに格納する。これらの処理は、第１の実施形態と同様であり、詳細なシーケンスの図示は省略する。このようにして、これら独立情報とその従属情報は、第１の実施形態で述べたバッファ格納方法に従ってストレージに格納されていく。なお、独立情報から参照されるＵＲＬの把握は、独立情報がストレージに書き込まれる前にバッファから読み出して行ってもよいし、ストレージに書き込んだ後、ストレージから読み出して行ってもよいし、あるいはバッファに独立情報を書き込む前に行うことも考えられる。

【００８９】

最終的に、独立情報と対応する従属情報が全てストレージに格納されると、バックエンドアプリはフロントエンドに完了を通知する（Ｓ２１２）。この際、最初に取得要求された情報（すなわち独立情報）を合わせて返す。その後、完了を通知と独立情報を返されたフロントエンドは、従属情報の取得要求を生成してバックエンドに送る（Ｓ２１３）。バックエンドは、取得要求されたこれらの従属情報の読み出し要求をストレージに行い（Ｓ２１４）、ストレージにおける連続した領域からこれらの従属情報を読み出し（Ｓ２１５）、各従属情報を、フロントエンドに送り返す（Ｓ２１６）。ステップＳ２１３でフロントエンドが発行する従属情報取得要求は、ＨＴＭＬファイルを取得したウェブブラウザが通常行う要求である。この従属情報取得要求をバックエンドが取得し、ストレージから対応する情報（従属情報）を読み出して、フロントエンドに返す。

【００９０】

（第２の実施形態に適用する場合）

第２の実施形態に対して本実施形態の機能を適用する場合、フロントエンドをメインプロセッサ９０６で動作させて、バックエンドをモジュール９００におけるプロセッサ９０１で動作させればよい。この場合のシーケンス図を図１１に示す。フロントエンドをメインプロセッサ９０６で動作させて、バックエンドをモジュール９００におけるプロセッサ９０１で動作させること以外は、基本的に図１０と同様であるため、対応するシーケンスには同一の参照符号を付して、図１１の説明は省略する。

【００９１】

＜第４の実施形態＞

これまでの実施形態では、応答に対し、必要に応じてバッファに予約領域を設定し、バッファの管理情報には、予約済みサイズを含めた。本実施形態のバッファ管理方式では、バッファに予約領域の設定を行わず、バッファの管理情報にも、予約済みサイズの項目を設けない。本実施形態の機能は、第１～第３の実施形態のいずれにも適用可能であるが、以下では第３の実施形態に適用する場合を示す。基本的な動作はこれまでの実施形態と同様であるため、以下では差分を中心に説明する。図１２に、以下の説明で使用するバッファ管理情報の例を示す。予約済みサイズの項目は存在しない。

【００９２】
プロセッサ１０１は、最初の取得要求（独立情報の取得要求に相当）をサーバ装置２０１に送信する。この際、プロセッサ１０１は、図１２のバッファ管理情報の左列に示すように、Ｏｆｆｓｅｔを０とするバッファ（バッファＢとする）の割り当てを行っておく（この時点では使用済みサイズは０である）。プロセッサ１０１は、サーバ装置２０１から応答を受信すると、応答に含まれるＣｏｎｔｅｎｔ－Ｌｅｎｇｔｈヘッダの値を取得することで、当該応答で受信するデータのサイズｄ１を求める。なお、このサイズｄ１は、最初の取得要求に応じて取得される独立情報を格納するために必要なストレージ領域のサイズに相当する。プロセッサ１０１は、サイズｄ１を、バッファサイズ（バッファ長）Ｌで割ったときの剰余ｒのサイズ分のデータを保存するためのバッファ（バッファＢ１とする）を、図１２のバッファ管理情報の２番目の列に示すように、新規に割り当てる。この際、バッファＢ１のオフセットは、サイズｄ１を、バッファサイズ（バッファ長）Ｌで割ったときの商をＮとすれば、Ｌ×Ｎとなる。例えば、サイズｄ１が、バッファサイズＬ×４＋ｒであった場合、商は４であるため、オフセットは、Ｌ×４となる。

【００９３】
最初の取得要求で取得したウェブページ（独立情報）に関連する従属情報を取得してバッファへ書き込む際は、独立情報の次に取得する従属情報の場合は、先に新規に割り当てたバッファＢ１の先頭からｒバイトシフトしたアドレスから書き込みを行う。従って、バッファＢ１には、取得した独立情報の末尾が偶然バッファＢ１の末尾に一致する場合を除き、最初に取得したウェブページの末尾と、ウェブページに続けて最初に取得要求を行った従属情報の先頭がバッファＢ１内で連続することになる。これにより、ストレージ１０３上における当該独立情報の次に取得する従属情報の書き込み位置は、最初に取得したウェブページのデータ長Ｌ＊Ｎ＋ｒバイトに続く位置となる。

【００９４】
これまでの実施形態と同様、パケットのデータをバッファに書き込んだ後、当該バッファの管理情報の使用済みサイズを更新する（なお、使用済みサイズの代わりに、バッファのサイズから使用済みサイズを減算した未使用サイズを用いてもよい。）。使用済みサイズが閾値（バッファサイズ）に一致した場合、すなわち、未使用サイズが０になった場合には、バッファのデータをストレージ１０３に書き出し、バッファを解放する。そして、オフセットをバッファサイズＬだけ加算して、当該解放したバッファを新規バッファとする（図８（Ａ）参照）。

【００９５】
プロセッサ１０１は、取得要求を送信した場合、もしくは応答を受信した場合、図１４のフローチャートに示すように、当該取得要求に応じて受信するデータの書き込み位置を包含するバッファが割り当て済みかを判断する（ステップＳ１５０１）。具体的に、バッファＢの管理情報が示すオフセットをＯｆｆｓｅｔ（Ｂ）、バッファサイズをＬとすると、Ｏｆｆｓｅｔ（Ｂ）以上、かつ、Ｏｆｆｓｅｔ（Ｂ）＋Ｌ以下の範囲に、当該書き込み位置が含まれるバッファが存在するかを判断する。そのようなバッファが存在しない場合は、当該応答に属するデータの書き込みの開始位置を計算する（ステップＳ１５０２）。この場合、書き込み位置は、ストレージの保存領域先頭からの位置であり、例えば上述した独立情報の次に取得する従属情報の場合は、Ｌ＊Ｎ＋ｒバイトに続く位置である。より一般的に、当該ストレージの保存領域先頭から当該書き込み位置までのサイズをバッファサイズＬで除算して商NN_xと剰余r_xを求め（ステップＳ１５０３）、$L_x \times NN_x$をバッファ管理情報のオフセットに設定したバッファを新規に割り当てる（ステップＳ１５０４）（図１２の右列参照）。この後、受信された応答のデータを、当該バッファの先頭から当該剰余r_xだけ進めた位置（すなわちストレージの保存領域先頭からの位置に対応する、バッファ上の位置）からデータの書き込みを行う。このときのパケットのデータのサイズがｄｘ－１だとすると、使用済みサイズは、図１２の右列に示すようにｄｘ－１となる。図１２では３列のみバッファ管理情報が示されているが、必要に応じて列が追加される。これまでの実施形態と異なり、バッファ管理情報は、応答ごとに管理される必要は

ない。

【００９６】

なお、ストレージに書き込むデータについて、図１３のようなストレージ管理情報で管理してもよい。この例ではストレージ管理情報は、リスト形式を有するが、別の形式でも構わない。情報（独立情報または従属情報）の識別子（ＵＲＬまたはＩＤ）、書き込み位置（ストレージの保存領域先頭からの位置）、書き込みファイル名、データの長さ、データの種類、関連情報の有無（自身から参照している情報（従属情報）が存在するか否か）を含んでいる。図示の例では、独立情報の取得ＵＲＬは、http://example.com/index.htmlであり、当該ＵＲＬから取得されたデータは、ファイル名「file1.dat」で管理されている。また、独立情報から参照されているＵＲＬ（従属情報のＵＲＬ）はhttp://example.com/example.jpgであり、当該従属情報は、独立情報と同じファイル名「file1.dat」でファイルシステムにより管理されている。ストレージにおいては、それぞれの情報（独立情報、従属情報）の前に、データのサイズを含む管理情報を格納し、当該管理情報に連続して、情報（独立情報、従属情報）を格納してもよい。

【００９７】

以上、本実施形態によれば、バッファにデータを書き込む際に、ストレージの保存領域先頭からのデータの書き込み位置を計算し、バッファにおいて当該書き込み位置に対応する位置からデータを書き込むようにしたことにより、第１の実施形態における予約領域の長さ等を考慮する必要はなくなり、処理の軽量化が期待できる。

【００９８】

尚、各実施形態のデータ受信装置またはモジュールは、例えば、汎用のコンピュータ装置を基本ハードウェアとして用いることでも実現することが可能である。すなわち、上記のコンピュータ装置に搭載されたプロセッサにプログラムを実行させることにより実現することができる。このとき、データ受信装置またはモジュールは、上記のプログラムをコンピュータ装置にあらかじめインストールすることで実現してもよいし、ＣＤ－ＲＯＭなどの記憶媒体に記憶して、あるいは通信網を介して上記のプログラムを配布して、このプログラムをコンピュータ装置に配置することで実現出来る。また、上記のコンピュータ装置に内蔵あるいは外付けされたメモリ、ハードディスク又はＣＤ－Ｒ、ＣＤ－ＲＷ、ＤＶＤ－ＲＡＭ、ＤＶＤ－Ｒ等の記憶媒体などを利用することが出来る。

【００９９】

本発明のいくつかの実施形態を説明したが、これらの実施形態は、例として提示したものであり、発明の範囲を限定することは意図していない。これら新規な実施形態は、その他の様々な形態で実施されることが可能であり、発明の要旨を逸脱しない範囲で、種々の省略、置き換え、変更を行うことができる。これら実施形態やその変形は、発明の範囲や要旨に含まれるとともに、特許請求の範囲に記載された発明とその均等の範囲に含まれる。

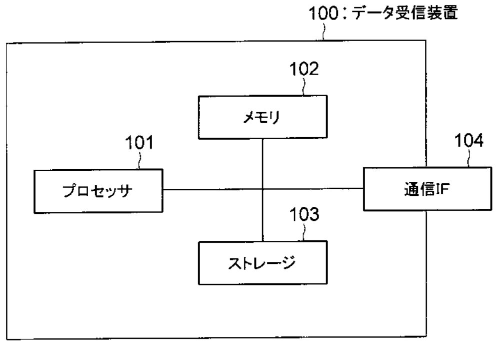
【符号の説明】

【０１００】

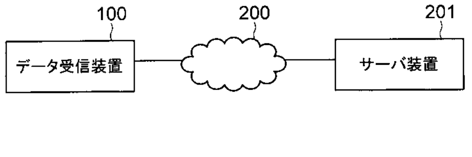
１００：データ受信装置
１０１：プロセッサ
１０２：メモリ
１０３：ストレージ
１０４：通信ＩＦ
１０６：データ受信装置
２００：ネットワーク
２０１：サーバ装置
９００：モジュール
９０１：プロセッサ
９０２：メモリ
９０３：ストレージ

９０４：通信ＩＦ
９０５：ホストインタフェース
９０６：メインプロセッサ

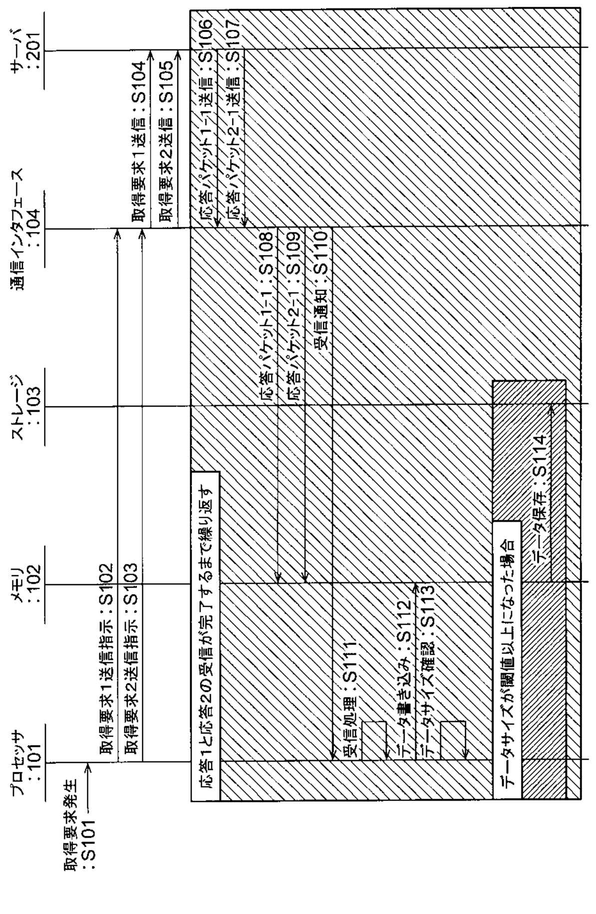
【図１】



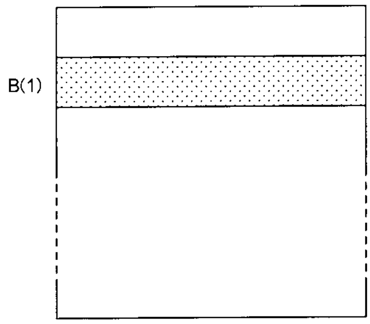
【図２】



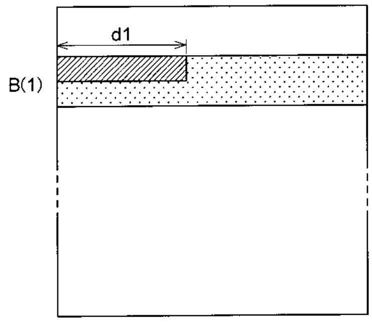
【図３】



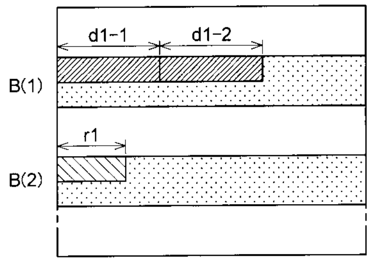
【図４Ａ】



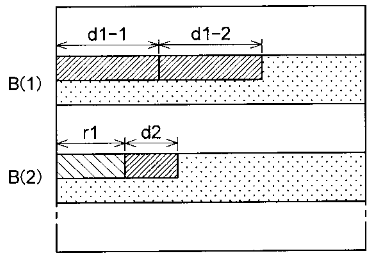
【図４Ｂ】



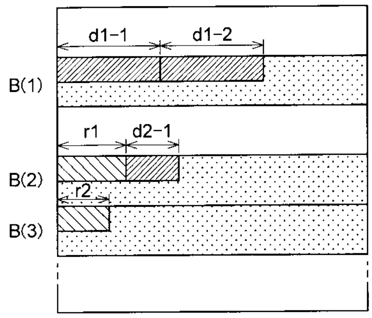
【図４Ｃ】



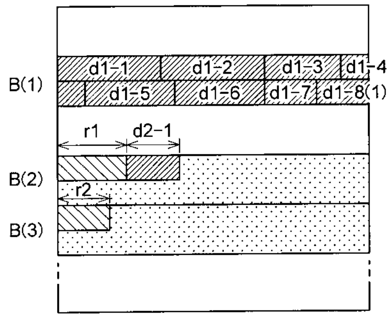
【図４Ｄ】



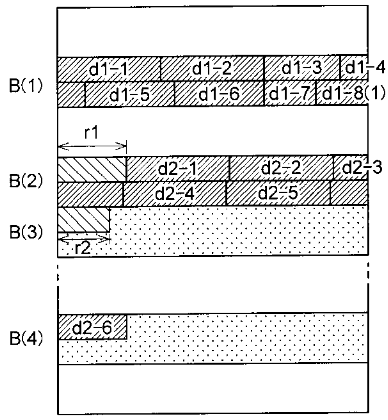
【図４Ｅ】



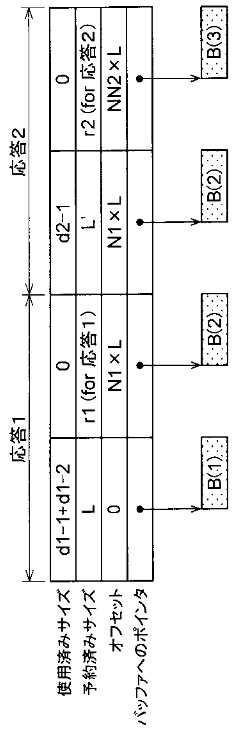
【図４Ｆ】



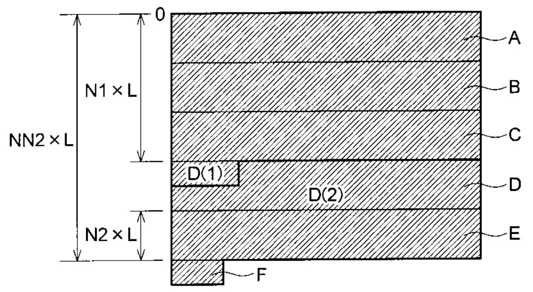
【図４Ｇ】



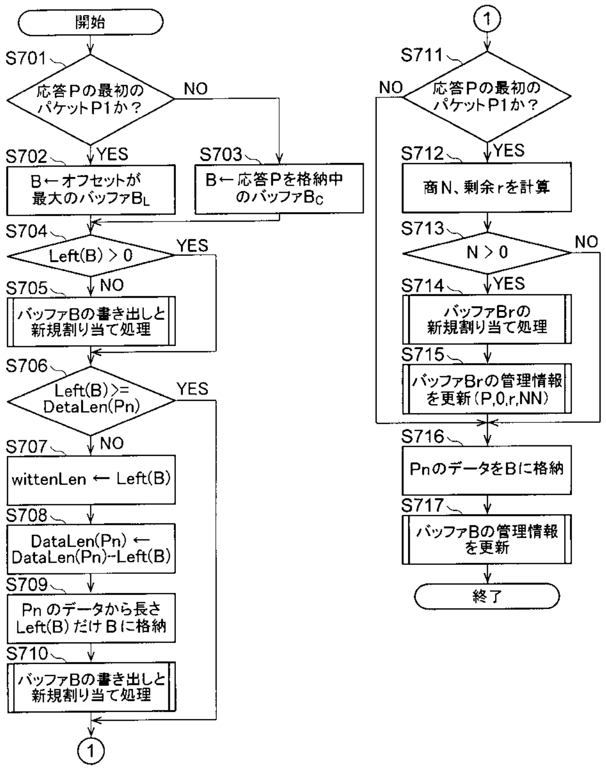
【図５】


応答1
応答2
d1-1+d1-2
L
0
B(1)
0
r1 (for 応答1)
N1×L
B(2)
d2-1
L'
N1×L
B(2)
0
r2 (for 応答2)
NN2×L
B(3)
使用済みサイズ
予約済みサイズ
オフセット
バッファへのポインタ

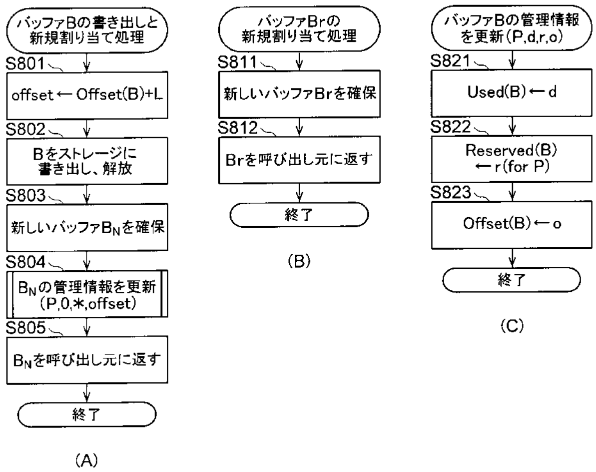
【図６】


0
N1×L
NN2×L
N2×L
A
B
C
D(1)
D(2)
D
E
F

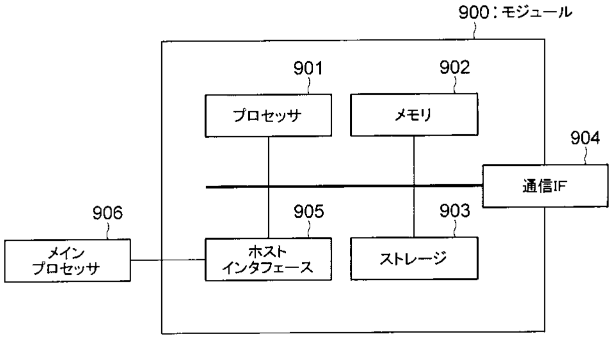
【図７】


開始
S701
応答Pの最初のパケットP1か？
NO
YES
S702
B←オフセットが最大のバッファB_L
S703
B←応答Pを格納中のバッファB_C
S704
Left(B) > 0
YES
S705
NO
バッファBの書き出しと新規割り当て処理
S706
Left(B)>= DetaLen(Pn)
YES
S707
NO
wittenLen ← Left(B)
S708
DataLen(Pn) ← DataLen(Pn)-Left(B)
S709
Pn のデータから長さLeft(B) だけ B に格納
S710
バッファBの書き出しと新規割り当て処理
1
1
S711
NO
応答Pの最初のパケットP1か？
S712
YES
商N、剰余rを計算
S713
N > 0
NO
S714
YES
バッファBrの新規割り当て処理
S715
バッファBrの管理情報を更新(P,0,r,NN)
S716
Pnのデータを Bに格納
S717
バッファBの管理情報を更新
終了

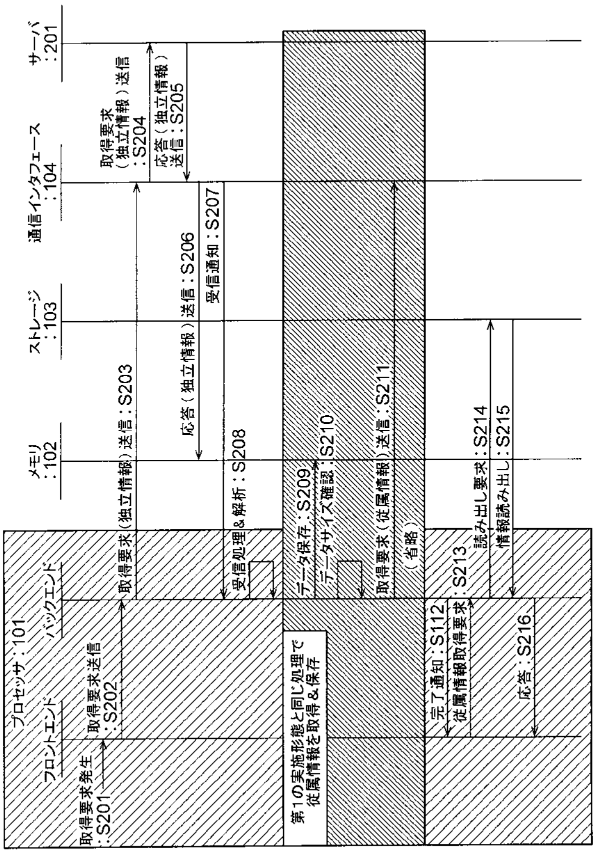
【図８】


バッファBの書き出しと新規割り当て処理
S801
offset ← Offset(B)+L
S802
Bをストレージに書き出し、解放
S803
新しいバッファB_Nを確保
S804
B_Nの管理情報を更新(P,0,*,offset)
S805
B_Nを呼び出し元に返す
終了
(A)
バッファBrの新規割り当て処理
S811
新しいバッファBrを確保
S812
Brを呼び出し元に返す
終了
(B)
バッファBの管理情報を更新(P,d,r,o)
S821
Used(B) ← d
S822
Reserved(B) ← r(for P)
S823
Offset(B) ← o
終了
(C)

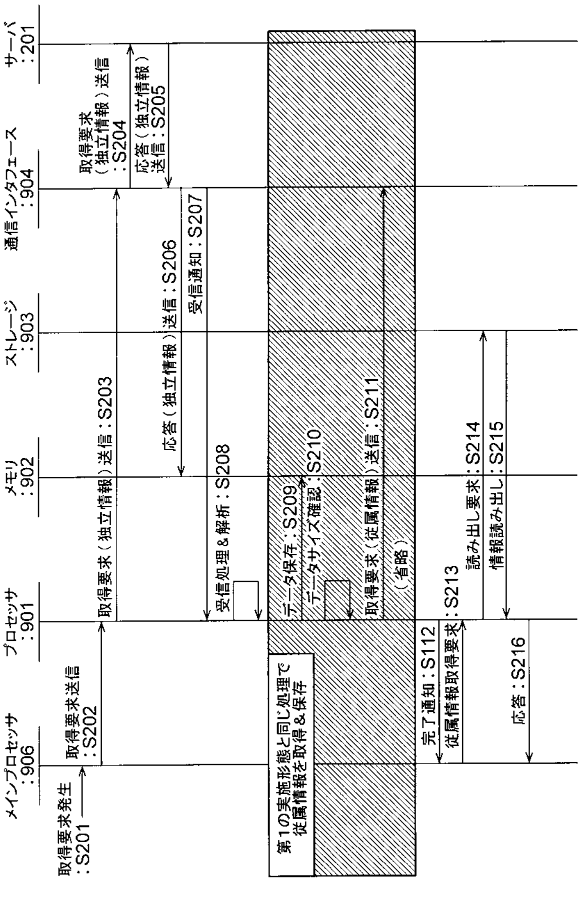
【図９】



【図１０】

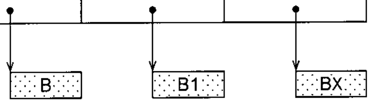


【図１１】



【図１２】

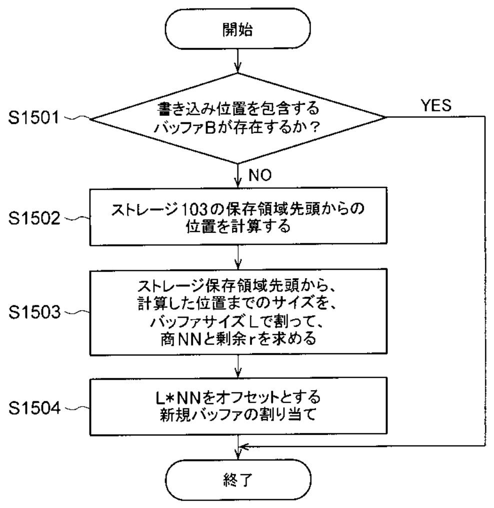
オフセット	0	L×N	L×NNx
使用済みサイズ	d1-1+d1-2	0	dx-1
バッファへのポインタ	●	●	●



【図１３】

取得URL、またはID	http://example.com /index.html	http://example.com /example.jpg
書き込み位置	r (1)	r (2)
書き込みファイル名	file1.dat	file1.dat
データサイズ	xx byte	yy byte
データ種類	html	jpg
関連情報の有無	1	0

【図１４】


開始
S1501
書き込み位置を包含する
バッファBが存在するか？
YES
NO
S1502
ストレージ103の保存領域先頭からの
位置を計算する
S1503
ストレージ保存領域先頭から、
計算した位置までのサイズを、
バッファサイズLで割って、
商NNと剰余rを求める
S1504
L*NNをオフセットとする
新規バッファの割り当て
終了

フロントページの続き

Ｆターム(参考) 5B089 GA21 HA04 KA05 KD01 KD07
5K034 AA07 AA11 HH21 HH27 HH28