

(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2004-355271

(P2004-355271A)

(43) 公開日 平成16年12月16日(2004.12.16)

(51) Int. Cl.⁷

G06F 12/06

F I

G06F 12/06 550A

テーマコード(参考)

5B060

審査請求 有 請求項の数 5 O L (全 16 頁)

(21) 出願番号 特願2003-151381 (P2003-151381)
 (22) 出願日 平成15年5月28日 (2003.5.28)

(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 100083806
 弁理士 三好 秀和
 (74) 代理人 100100712
 弁理士 岩▲崎▼ 幸邦
 (74) 代理人 100100929
 弁理士 川又 澄雄
 (74) 代理人 100108707
 弁理士 中村 友之
 (74) 代理人 100095500
 弁理士 伊藤 正和
 (74) 代理人 100101247
 弁理士 高橋 俊一

最終頁に続く

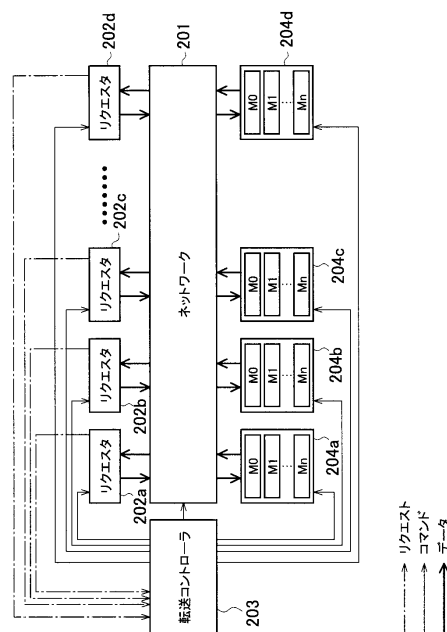
(54) 【発明の名称】 データ転送システム

(57) 【要約】

【課題】 ネットワークリソースを有効に利用することである。

【解決手段】 データ転送システムにおいて、データ転送要求を送信し、許可された場合には、データ転送を行う複数のリクエスト201aと、複数のリクエストからデータ転送要求を受信し、受信したデータ転送要求のアービトレーションを行って許可するデータ転送要求を決定し、転送指示を所定のタイミングで送信する転送コントローラ203と転送指示を受信し、転送指示に基づいて許可されたリクエストからのデータを転送するネットワーク201と、個々にデータの入出力部を有する複数のモジュールを有し、転送指示を受信し、送信指示に基づいてネットワークから転送データを受信するメモリ204aと、を有するようにしてある。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

データ転送システムにおいて、
データ転送要求を送信し、許可された場合には、データ転送を行う複数のリクエストと、
前記複数のリクエストからデータ転送要求を受信し、前記受信したデータ転送要求のアー
ビトレーションを行って許可するデータ転送要求を決定し、転送指示を所定のタイミング
で送信する転送コントローラと、
前記転送指示を受信し、前記転送指示に基づいて許可されたリクエストからのデータを転
送するネットワークと、
個々にデータの入出力部を有する複数のモジュールを有し、前記転送指示を受信し、前記
送信指示に基づいて前記ネットワークから転送データを受信するメモリと、を有すること
を特徴とするデータ転送システム。

【請求項 2】

前記転送コントローラは、前記メモリのアービトレーションを行った後に、前記ネットワ
ークのアービトレーションを行うことを特徴とする請求項 1 記載のデータ転送システム。

【請求項 3】

前記転送コントローラは、前記ネットワークのアービトレーションを行った後に、前記メ
モリのアービトレーションを行うことを特徴とする請求項 1 記載のデータ転送システム。

【請求項 4】

前記転送コントローラは、前記ネットワークのアービトレーションと前記メモリのアービ
トレーションを並列に行うことを特徴とする請求項 1 記載のデータ転送システム。

【請求項 5】

前記ネットワークのデータ幅と前記メモリの入出力データ幅が等しいことを特徴とする請
求項 1 乃至 4 のいずれか一項に記載のデータ転送システム。

【請求項 6】

前記転送コントローラは、前記リクエストからのデータ転送要求が読み込み処理か書き込み
処理かを判定して、この処理の種類によって転送指示の送信を制御することを特徴とする
請求項 1 乃至 4 のいずれか一項に記載のデータ転送システム。

【請求項 7】

前記複数のモジュールは、記憶領域であるマクロを複数有し、これらマクロに分散してデ
ータを記憶することを特徴とする請求項 1 乃至 4 のいずれか一項に記載のデータ転送シ
ステム。

【請求項 8】

前記転送コントローラは、2 以上のサイクルのメジャーサイクルを処理単位として転送制
御を行うことを特徴とする請求項 1 乃至 4 のいずれか一項に記載のデータ転送システム。

【請求項 9】

前記転送コントローラは、
前記複数のリクエストからのデータ転送要求を受け付けるリクエスト保持部と、
前記リクエスト保持部で保持されたデータ転送要求のアドレスをデコードするアドレスデ
コード部と、
前記デコードされたデータ転送要求について、前記メモリの各モジュール毎にアービト
レーションを行うモジュールアービトレーション部と、
前記デコードされたデータ転送要求について、ネットワークを割り当てるためのアービト
レーションを行うネットワークアービトレーション部と、
前記モジュールアービトレーション部及びネットワークアービトレーション部によって使
用権を割り当てられたリクエストに係る転送指示を生成する転送指示生成部と、を有す
ることを特徴とする請求項 1 乃至 4 のいずれか一項に記載のデータ転送システム。

【請求項 10】

前記モジュールアービトレーション部は、デコードされたリクエストをメモリモジュール
番号毎に保持するモジュールアービタを有することを特徴とする請求項 9 記載のデータ転

10

20

30

40

50

送システム。

【請求項 1 1】

前記モジュールアービターは、前記リクエストの中から、現在最も早くアクセス可能なマクロを選びだすことを特徴とする請求項 1 0 記載のデータ転送システム。

【請求項 1 2】

前記モジュールアービターは、前記マクロの応答速度に基づいて、その時点でアクセス可能なマクロへのデータ転送要求を選択することを特徴とする請求項 1 1 記載のデータ転送システム。

【請求項 1 3】

前記モジュールアービターは、前記リクエストのデータ転送の優先権情報を有し、前記選択されたデータ転送要求のうち、最も高い優先権に係るリクエストを決定することを特徴とする請求項 1 2 記載のデータ転送システム。

10

【請求項 1 4】

前記ネットワークアービトレーション部は、読み込み処理に係るサイクル及び書き込み処理に係るサイクルを考慮して決定されたメジャーサイクル毎に、前記ネットワークのバスの数に基づいてリクエストからの読み込み処理及び書き込み処理を割り当てることを特徴とする請求項 9 記載のデータ転送システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

20

本発明は、データ転送システムに関し、特に、複数のリクエストが共有するネットワークを介してメモリをアクセスする場合に用いられる技術に関する。

【0002】

【従来の技術】

図 1 1 は、第 1 の従来例に係るデータ転送システムを示すブロック図である。

図示の如く、複数のリクエスト 1 0 2 a , 1 0 2 b , 1 0 2 c , 及び、1 0 2 d はネットワーク 1 0 1 に接続される。これらリクエストがネットワーク 1 0 1 を経由してメモリ 1 0 5 a , 1 0 5 b , 1 0 5 c , 及び、1 0 5 d へアクセスする場合、リクエストはまず、ネットワークアービター 1 0 3 にネットワーク 1 0 1 を使用するリクエスト信号を送る。ネットワークアービター 1 0 3 はリクエストからのリクエストを、ラウンドロビン等のアルゴリズムに基づいてネットワーク 1 0 1 を割り当てる。割り当てに成功したリクエストに対してはネットワーク 1 0 1 の使用許可を示す許可信号が返される。ネットワーク 1 0 1 の使用許可を得られたリクエストは、メモリコントローラ 1 0 4 a , 1 0 4 b , 1 0 4 c , 及び、1 0 4 d のうち、所定のメモリコントローラにコマンドと書き込み処理の場合はデータを送る事が可能となる。その後、コマンドを受け取ったメモリコントローラはメモリを読み書きするための制御をし、データをメモリに書き込み、または読み出しを行う。

30

【0003】

図 1 2 は、第 2 の従来例に係るデータ転送システムを示すブロック図である。

この従来例は、メモリコントローラ部分にバッファを持たない構成例である。このデータ転送システムでは、リクエスト 1 0 7 a , 1 0 7 b , 1 0 7 c , 及び、1 0 7 d が読み出したいコマンドをメモリコントローラ 1 0 9 a , 1 0 9 b , 1 0 9 c , 及び、1 0 9 d のうち、所定のメモリコントローラに送る。コマンドを受け取ったメモリコントローラは、そのコマンドに従いメモリ 1 1 0 a , 1 1 0 b , 1 1 0 c , 及び、1 1 0 d のうちコマンドを受け取ったメモリコントローラに係るメモリに対してデータを読み出す制御をする。ここで、メモリコントローラは、メモリの状態を把握しているので、データが読み出せるタイミングに合わせてネットワークが使用できるようにネットワークアービター 1 0 8 に転送の要求を行う。期待したとおりにネットワーク 1 0 6 の転送許可がおりればよいが、そうでない場合でもメモリからデータを読み出さないようにすれば、中間バッファは不必要である。つまり、メモリ自身が中間バッファの役目を果たしている。書き込みの場合

40

50

も同様にリクエストがメモリコントローラにコマンドを送り、メモリコントローラはメモリにWriteコマンドを入れる。実際にデータを書き込むのはその後なので、そのタイミングに合わせてネットワークアービター108に転送許可を求め、許可が得られたらメモリコントローラがリクエストからデータを吸い出し、それをメモリへ書き込む。

【0004】

【発明が解決しようとする課題】

上記第1の従来例に係るデータ転送システムの問題点は、書き込み処理の場合メモリコントローラで実際にメモリへデータを書き込むタイミングまでデータをバッファでホールドしておく必要がある事である。また、読み込み処理の場合は、メモリコントローラがデータを読み出すまでネットワークを占有するという問題がある。このネットワークを占有する問題を解決するためにスプリットトランザクション法(Split Transaction)という手法があるが、この場合は書き込み処理と同様にリクエストが再アクセスに来るまで読み込みデータをメモリコントローラ側でデータ保持する必要が生じる。画像処理やストリーム処理のような一度に大量のデータ(バースト転送)を行う場合、メモリコントローラ側でデータを保持する大容量のバッファが面積増加を招き問題となる。

10

【0005】

また、第2の従来例に係るデータ転送システムにおいては、バッファを取り除く事はできるが、この場合の問題点は、メモリからデータは読み出せる状態になっているにもかかわらず、ネットワークの転送許可がおりるまでは、そのメモリへはアクセスできないことである。メモリが複数のバンクを持っていた場合には、アクセス効率が悪くなる。また、書き込みの場合は、メモリにコマンドを入れてから実際にデータを書き込むまでの時間がネットワークのレイテンシよりも短い場合、例えば、ネットワークのレイテンシが非常に長い場合には、メモリはデータの書き込み準備が出来ているにもかかわらずデータがこない状態に陥る。この場合は、読み出しの場合と同様にそのメモリへはアクセスできないので、読み出しの場合と同じ問題が発生する。

20

【0006】

本発明は上記事情に鑑みて成されたものであり、その目的とするところは、ネットワークリソースを有効に利用することを可能としたデータ転送システムを提供することにある。

【0007】

【課題を解決するための手段】

本願発明の一態様によれば、データ転送システムにおいて、データ転送要求を送信し、許可された場合には、データ転送を行う複数のリクエストと、前記複数のリクエストからデータ転送要求を受信し、前記受信したデータ転送要求のアービトレーションを行って許可するデータ転送要求を決定し、転送指示を所定のタイミングで送信する転送コントローラと、前記転送指示を受信し、前記転送指示に基づいて許可されたリクエストからのデータを転送するネットワークと、個々にデータの入出力部を有する複数のモジュールを有し、前記転送指示を受信し、前記送信指示に基づいて前記ネットワークから転送データを受信するメモリと、を有する。

30

【0008】

【発明の実施の形態】

本発明に係るデータ転送システムの実施形態について、図面を参照しながら詳細に説明する。図1は、本発明に係るデータ転送システムの実施形態を示すブロック図である。このデータ転送システムは、各種データ転送を実現するネットワーク201と、リクエスト202a, 202b, 202c, 及び、202dと、複数のマクロ(0乃至n(nは1以上の整数))を有するメモリモジュール204a, 204b, 204c, 及び、204dと、を有する。

40

【0009】

ここで、ネットワーク201は、リクエスト、転送コントローラ、及びメモリモジュールに接続され、リクエストからの要求に応じて所定のメモリモジュールへデータの転送を行う。ここで、このネットワーク201は、クロスバー方式(Cross Bar)または

50

複合バス方式 (Multiple Bus) を使用することができる。

【 0 0 1 0 】

各リクエストは、ネットワーク 2 0 1 及び転送コントローラ 2 0 3 に接続される。これらリクエストはデータ転送の要求を転送コントローラ 2 0 3 に行い、許可をされた場合には、ネットワークを介して所定のメモリモジュールとデータ転送を行う。

【 0 0 1 1 】

転送コントローラ 2 0 3 は、ネットワーク 2 0 1、各リクエスト、及び各メモリモジュールに接続され、各リクエストからのデータ転送の要求を受け取り、所定のリクエストにデータ転送の許可を行う。ネットワーク 2 0 1 のアービトレーションとマクロを考慮したメモリモジュールのアービトレーションを同時に行う。この例では、メモリモジュールには複数のマクロ (Macro) をもったメモリモジュールを使用する。

10

【 0 0 1 2 】

メモリモジュールはネットワーク 2 0 1 及び転送コントローラ 2 0 3 に接続され、ネットワーク 2 0 1 を介して、所定のリクエストとの間でデータ転送を行う。各メモリモジュールには複数の記憶領域が設けられ、それら記憶領域をマクロと呼ぶ。このメモリモジュールには、混載 D R A M (Embedded D R A M) と呼ばれる、I C 回路の中に埋め込まれた D R A M を用いることが出来る。

【 0 0 1 3 】

ここで、あるマクロへのアクセスした際のマクロの応答速度には所定サイクルを要するが (例えば、ランダム・サイクルタイム (t R C) = 3 2 サイクル) であるが、1つのメモリモジュールに複数のマクロを持ちインタリーブする事で、見かけのアクセスのためのサイクルを小さくする事ができる。例えばメモリモジュールのマクロ数が 4 (N = 4) の時、マクロのアクセスをうまく制御することで、見かけ上の t R C を 8 サイクルにする事が可能である。

20

【 0 0 1 4 】

次に、本実施形態に係るデータ転送システムの動作について説明する。まず、転送コントローラ 2 0 3 は、所定のリクエストからデータ転送要求を受けると、ネットワーク 2 0 1 の確保と、アクセスするメモリモジュールが有効利用されるようにマクロアクセスのアービトレーションを行い、マクロアクセス権の確保をする。その後、転送コントローラ 2 0 3 は、転送のタイミングに合わせてリクエストに転送指示を出し、ネットワークにスイッチの切り替え情報を送り、メモリモジュールへコマンドを送る。ここで、メモリモジュールへのコマンドは、アクセスの種類 (Read / Write)、及びアドレス (マクロ番号、 Row (Row) 番号、 Column (Column) 番号) である。これにより、これまで必要であったデータを一時的に保持するバッファを少なくする事が出来る。ここで、ネットワーク 2 0 1 の 1 チャンネルのバンド幅とメモリモジュールのバンド幅のバランスが取れていなくても実施は可能であるが、ネットワーク 1 チャンネル幅とメモリモジュールのバンド幅が等しい場合には、バッファを取り除くことが出来る。

30

【 0 0 1 5 】

図 2 は、メジャーサイクルを説明するための図面である。本実施形態に係るデータ転送システムにおいては、転送処理を簡単化、処理のパイプライン化を容易にするためメジャーサイクルと呼ばれるサイクルを単位として処理や指示を行う。図示の如く、本実施形態においては、通常の 8 サイクル分のクロックを 1 メジャーサイクルと定義した。このように、メジャーサイクル単位で処理を行う事により、転送コントローラ 2 0 3 の速度制約が緩くなり設計が容易になる。また、転送もメジャーサイクルを転送単位とする事で、転送コントローラ 2 0 3 のアービトレーションと実際の転送をパイプライン化する事が可能となる。例えば、図 2 の例ではバーストサイズは最大 8 で、バーストサイズが 8 以下の場合も転送に 8 サイクル消費する。

40

【 0 0 1 6 】

図 3 は、メジャーサイクルを用いて転送処理のパイプライン化を説明するための図面である。本実施形態で用いられるメモリモジュールは、読み込み命令 (Read コマンド) を入

50

れてから最初の読み込みデータが出てくるまで16サイクルかかる ($t_{RD} = 16$ サイクル)ものとする。また、書き込み処理の場合は、書き込み命令 (Write コマンド) を入れてから実際に書き込みデータを入れるまで12サイクル必要である ($t_{WR} = 12$ サイクル)ものとする。すなわち、図3において、転送コントローラによるアービトレーションにより命令0が読み込み命令の場合には、2メジャーサイクル後に当該命令に関して転送を許可するようにする。一方、命令0が書き込み命令の場合には、1メジャーサイクル後に当該命令に関して転送を許可するようにする。このようにして、パイプライン化することにより、ネットワークを効率的に使用することができるのである。

【0017】

図4は、本実施形態に係るデータ転送システムにおける各部の処理を説明するためのタイミングチャートである。ここで、 t_{TND} はネットワークの入口までの遅延を示し、 t_{FND} はネットワークの入口からの遅延を示し、 t_{RD} はコマンドを入れてから読み込みデータが出てくるまでの時間を示し、 t_{WR} はコマンドを入れてから書き込みデータが出てくるまでの時間を示す。転送コントローラは、メジャーサイクル0でリクエストからデータ転送要求を受け付ける。次のメジャーサイクル1でネットワークとメモリモジュールのアービトレーションを行っている。ネットワークとメモリモジュールのリソースが確保できた場合、以下のタイミングで各リソースに指示を出す。

10

【0018】

まず、メモリモジュールからのリクエストへの読み込み処理について説明する。

まず、転送コントローラはメジャーサイクル0にてデータ転送要求の受付を行う。このメジャーサイクル0の中でデータ転送要求があった場合には、転送コントローラは、メジャーサイクル1にてアービトレーションを行い、メジャーサイクル1の終わりでメモリモジュールに対して読み指示を行う (メジャーサイクル1のdの位置)。ここで、メモリモジュールの読み出しには2メジャーサイクル ($t_{RD} = 16$ サイクル) 掛かるので、メジャーサイクル4で読み込みデータがメモリモジュールから出てくる。従って、そのデータがネットワークまでに届く遅延を考慮し (t_{TND} : ここでは2サイクル)、メジャーサイクル4の2サイクル目でネットワークに指示を行う (メジャーサイクル4の2サイクル目のdの位置)。ネットワーク出口からのデータがリクエストまで届く遅延 ($t_{FND} = 2$ サイクル) を考慮し、4メジャーサイクル+4サイクルでリクエストにデータ受信の指示を行う (メジャーサイクル4の4サイクル目のdの位置)。

20

30

【0019】

次に、リクエストからメモリモジュールへの書き込み処理について説明する。

まず、転送コントローラはメジャーサイクル0にてデータ転送要求の受付を行う。このメジャーサイクル0の中でデータ転送要求があった場合には、転送コントローラは、メジャーサイクル1にてアービトレーションを行い、メジャーサイクル1の終わりでメモリモジュールに対して読み指示を行う (メジャーサイクル1のdの位置)。ここで、メモリモジュールの書き込みには1メジャーサイクル ($t_{RD} = 8$ サイクル) 掛かるので、メジャーサイクル3で読み込みデータがメモリモジュールから出てくる。従って、そのデータをメモリモジュールに入れるサイクルを考慮して、リクエストにメジャーサイクル2の終わりで転送指示を行う (メジャーサイクル2のdの位置)。そのデータがネットワークまでに届く遅延を考慮し (t_{TND} : ここでは2サイクル)、メジャーサイクル3の2サイクル目でネットワークにスイッチ情報を行う (メジャーサイクル3の2サイクル目のdの位置)。メモリモジュールは、届いたデータをメジャーサイクル3の4サイクル目で転送されたデータを受信し、メモリモジュールに書き込む。

40

【0020】

図5は、本実施形態の転送コントローラの詳細を説明するためのブロック図である。この転送コントローラ203は、リクエスト202からのデータ転送要求を受け付けるリクエスト保持部211と、リクエスト保持部で保持されたデータ転送要求のアドレスをデコードするアドレスデコード部212と、デコードされたデータ転送要求を各モジュール毎にアービトレーションを行うモジュールアービトレーション部213と、データ転送要求に

50

ネットワークを割り当てるためのアービトレーションを行うネットワークアービトレーション部 214 と、このネットワークアービトレーションによって使用権を割り当てられたデータ転送要求に係る転送指示を生成しする転送指示生成部 215 と、を有する。

【0021】

次に、この転送コントローラの動作について説明する。まず、リクエスト保持部 211 は、リクエストからデータ転送要求を受け付け、そのデータ転送要求をリクエストキューに保持する。ここで、データ転送要求には、そのデータ転送要求が書き込み命令であるか、又は、読み込み命令であるか、の命令の種類と、処理の目的のアドレス（ターゲットアドレス）に関する情報が含まれる。次に、アドレスデコード部 212 は、ターゲットアドレスをデコードし、リクエストに係る情報からメモリモジュールの番号とマクロ番号を取り出す。次に、モジュールアービトレーション部は、デコードされたデータ転送要求をメモリモジュール番号毎に保持するモジュールアービターを有する。このモジュールアービターはそのメモリモジュールが有するモジュールの個数に対応して設けられる。そして、送られて来たデータ転送要求の中から、現在最も早くアクセス可能なマクロを選び出す。そのマクロをアクセスするデータ転送要求をラウンドロビン等のアルゴリズムを用いて、1つを選択する。選択されたデータ転送要求は、ネットワークアービトレーション部 214 に送られる。次に、ネットワークアービトレーション部 214 は、ネットワークが複合バスの場合、モジュールアービターから送られて来たデータ転送要求がバスの本数以上であった場合、ここで再度アービトレーションを行い、所定のデータ転送要求に対してバスを割り当てる。メモリモジュールのアクセス、ネットワークの使用権を割り当てられたデータ転送要求は、転送指示生成部 215 (Transfer Directive Generator) で、図 4 に示したタイミングに指示が届くようなタイミングでデータ転送を取るような指示をリクエスト、ネットワーク、及びメモリモジュールへ送る。

10

20

【0022】

図 6 は、本実施形態のモジュールアービターのアービトレーション例を説明するための図面であり、(a) は、リクエストから投入されたデータ転送要求の図表であり、(b) は、モジュールアービターの処理を説明するための図面である。モジュールアービターは、モジュールアービトレーション部 213 に各モジュールに対応して設けられ、メモリモジュールを有効に利用するために tRC を意識してアービトレーションを行う。このモジュールアービターは、そのモジュール内のマクロの中でアクセス可能なマクロを選択し、選択されたマクロへのデータ転送に係るデータ転送要求を行ったリクエストであって、優先権の最も高いリクエストを選択し、その後、各種の情報の更新を行う。

30

【0023】

図 6 (a) の例では、reqid = 0 のリクエストはマクロ番号 3 への転送処理、reqid = 1 のリクエストはマクロ番号 0 への転送処理、reqid = 2 のリクエストはマクロ番号 1 への転送処理、reqid = 3 のリクエストはマクロ番号 0 への転送処理、reqid = 4 のリクエストはマクロ番号 2 への転送処理、また、reqid = 5 のリクエストはマクロ番号 3 への転送処理であるものとする。

【0024】

また、各モジュールアービターが保持している情報は、各マクロ毎に以前アクセス権を割り当てた時刻と最も優先順位の高いリクエスト Id (ReqStartId) である。図 6 (b) の例では、マクロ 0 はその時刻 N から 16 サイクル前にアクセスされたものであり、マクロ 1 はその時刻 N から 48 サイクル前にアクセスされたものであり、マクロ 2 はその時刻 N から 24 サイクル前にアクセスされたものであり、マクロ 3 はその時刻 N から 32 サイクル前にアクセスされたものであるものとする。また、最も優先順位の高いリクエストは 3 (ReqStartId = 3) であるものとする。

40

【0025】

ここで、tRC = 32 サイクルとすると、現在アクセス可能なマクロは、マクロ 1 とマクロ 3 である。モジュールアービターに投入されたデータ転送要求から、マクロ 1 とマクロ 3 へのアクセスを要求しているものは、図 6 (a) から reqid = 0, 2, 5 である

50

事がわかる。また、図6(b)からReqStartIdが示しているId=3であるため、Id=3のリクエストの優先順位が最も高く、続いて、Id=4のリクエスト、Id=5のリクエスト、Id=6のリクエスト、・・・Id=0のリクエスト、Id=1のリクエスト、Id=2のリクエスト、の順番で優先順位がつけられている。そのため、本実施形態においては、このメモリモジュールにアクセスできるリクエストは、reqid=5のリクエストである。次に、reqid=5のリクエストがアクセスするマクロ3の直前割り当て時刻をN-32サイクルからNに更新し、ReqStartIdを更新する。ReqStartIdを更新は、優先権が最も高いリクエストがメモリアクセスを許可された場合にのみインクリメントするようにしてもよい。例えば、優先権が最も高いリクエストが4(ReqStartId=4)の場合、そのId=4のリクエストのアクセスが許可された場合にのみReqStartId=4をインクリメントして5にするようにしてもよい。これらのデータの更新は、ネットワークのアクセスを得てスケジューリングが完了した際に更新される。

10

【0026】

図7は、本実施形態のネットワークアービタ―のアービトレーション例を説明するための図面であり、(a)は、リクエストから投入されたデータ転送要求の図表であり、(b)は、ネットワークアービタ―の処理を説明するための図面である。ここでは、ネットワークが複合バス式(Multiple Bus)の場合のネットワークアービタ―の例を示す。この例ではバスの数は4とする。ネットワークアービタ―には、各モジュールアービタ―にてメモリアクセス権を得たデータ転送要求が送られてくる。これらメモリアクセス権を得たデータ転送要求は図7(a)に示した通り、reqid=0のリクエストは書き込みアクセス要求、reqid=1のリクエストは書き込みアクセス要求、reqid=2のリクエストは読み込みアクセス要求、reqid=4のリクエストは書き込みアクセス要求、reqid=5のリクエストは書き込みアクセス要求、reqid=7のリクエストは読み込みアクセス要求、reqid=8のリクエストは読み込みアクセス要求、また、reqid=9のリクエストは書き込みアクセス要求であるものとする。

20

【0027】

ネットワークの使用するタイミングは、前述の如く読み込み処理の場合と書き込み処理の場合で異なっており、本実施形態においては、書き込み処理の場合は2メジャーサイクル先、読み込み処理の場合は3メジャーサイクル先に実際に使用される。そのため、ネットワークアービタ―は、2メジャーサイクル先と3メジャーサイクル先のバスの空き状況を把握しておく必要がある。この例では、現在2メジャー先の空きバス数は2である。この状況はすでに1メジャーサイクル前のRead要求がバスを2本使用する予約が入っている。

30

【0028】

また、ネットワークアービタ―は、読み込み要求と書き込み要求用にモジュールアービタ―と同様にそれぞれ優先順位をつけるための識別子ReadReqStartId及びWriteReqStartIdを保持している。本実施形態では、読み込み要求と書き込み要求は交互にネットワーク割り当てが行われる。まず読み込み要求を1つネットワークに割り当てる。ReadReqStartIdが1なので、もっとも優先順位が高い読み込み要求(reqid=2)を3メジャーサイクル先のネットワークに割り当て、3メジャーサイクル先のバスの空き数をデクリメントして3にする。次に書き込み要求を割り当てる。WriteReqStartId=2なので、もっとも優先順位の高い書き込み要求はreqid=4となり、それを2メジャーサイクル先のバスに割り当て、バス空き数をデクリメントして1にする。バスの割り当てはメジャーサイクル毎に行われる。

40

このように読み込み要求、書き込み要求を交互に割り当て、バスの空き数が0になるかデータ転送要求がなくなるまで行い、最後に各StartIdをインクリメントしておく。ここで、書き込み処理は2メジャーサイクル先、読み込み処理は3メジャーサイクル先に割り当てられるため、読み込み処理が進捗しないおそれもある。そこで、WriteReqStartIdが0になった場合には、読み込み要求を3メジャーサイクル先のバスの使用を割り

50

当てるようにしても良い。

【0029】

図8は、第2の実施形態の転送コントローラの詳細を説明するためのブロック図である。この転送コントローラは、リクエスタ202からのデータ転送要求を受け付けるリクエスト保持部211と、リクエスト保持部で保持されたデータ転送要求のアドレスをデコードするアドレスデコード部212と、データ転送要求にネットワークを割り当てるためのアービトレーションを行うネットワークアービトレーション部214と、ネットワークアービトレーション部214にて選択されたデータ転送要求を各モジュール毎にアービトレーションを行うモジュールアービトレーション部213と、このネットワークアービトレーションによって使用権を割り当てられたデータ転送要求に係る転送指示を生成する転送指示生成部215と、を有する。

10

【0030】

本実施形態においては、まずネットワークのアービトレーションを行い、ネットワークの割り当てが出来たデータ転送要求でメモリモジュールアービトレーションをさらに行っている。その他は、図7と同様である。

【0031】

図9は、第3の実施形態の転送コントローラの詳細を説明するためのブロック図である。この転送コントローラは、リクエスタ202からのデータ転送要求のアドレスをデコードするアドレスデコード部216と、デコードされたデータ転送要求を各モジュール毎に保持するリクエスト保持部217と、リクエスト保持部217に保持されたデータ転送要求を各モジュール毎にアービトレーションを行うモジュールアービトレーション部213と、データ転送要求にネットワークを割り当てるためのアービトレーションを行うネットワークアービトレーション部214と、このネットワークアービトレーションによって使用権を割り当てられたデータ転送要求に係る転送指示を生成する転送指示生成部215と、を有する。

20

【0032】

本実施形態においては、アドレスデコード部216は、リクエスタからのデータ転送要求を受け付けると共に、そのデータ転送要求のデコードを行う。リクエスト保持部217は、リクエストキューを各モジュール毎に有し、デコードされたデータ転送要求を各モジュール毎に保持する。モジュールアービトレーション部213は、リクエスト保持部217の

30

【0033】

ここで、第3の実施形態に係るデータ転送システムにおける各部の処理を説明するためのタイミングチャートはほぼ図3を用いることができるが、転送コントローラは、メジャーサイクル0でリクエスタからデータ転送要求を受け付ける際には、デコードの時間を考慮して、メジャーサイクル0が終了する2, 3サイクル前までにデータ転送要求の受付を終了させることが好ましい。

40

【0034】

図10は、この転送コントローラは、リクエスタ202からのデータ転送要求のアドレスをデコードするアドレスデコード部216と、デコードされたデータ転送要求を各モジュール毎に保持するリクエスト保持部217と、リクエスト保持部217に保持されたデータ転送要求を各モジュール毎にアービトレーションを行うモジュールアービトレーション部213と、デコードされたデータ転送要求にネットワークを割り当てるためのアービトレーションを行うネットワークアービトレーション部214と、モジュールアービトレーション部213及びネットワークアービトレーションによって使用権を割り当てられたデータ転送要求に係る転送指示を生成する転送指示生成部215と、を有する。

【0035】

50

本実施形態においては、アドレスデコード部 216 は、リクエストからのデータ転送要求を受け付けると共に、そのデータ転送要求のデコードを行う。リクエスト保持部 217 及びネットワークアービトレーション部 214 は、デコードされたデータ転送要求を受け取る。ネットワークのアービトレーションリクエスト保持部 217 は、リクエストキューを各モジュール毎に有し、デコードされたデータ転送要求を各モジュール毎に保持する。モジュールアービトレーション部 213 は、リクエスト保持部 217 のリクエストキューに保持された命令を入力して、各モジュール毎にアービトレーションを行う。デコードされたデータ転送要求について、ネットワークアービトレーション部 214 にてネットワークのアービトレーションを行う。モジュールアービトレーション部 213 にて選択されたデータ転送要求であって、ネットワークアービトレーション部 214 にて選択されたデータ転送要求に対して転送指示生成部 215 にて転送指示を生成してメモリモジュール、リクエスト、及び、ネットワークに転送指示を送信する。このように、モジュールアービトレーション部 213 の処理とネットワークアービトレーション部 214 の処理を並行して行うことにより、処理時間の短縮化を図ることが出来る。

【0036】

以上のように、本実施形態に係るデータ転送システムにおいては、ネットワークのアービトレーションとメモリモジュールのマクロの制御も考慮したメモリモジュールのアービトレーションを同時に行う事により、サイクルレベルでのスケジューリングを可能にし中間に必要な大量のバッファを少なくする、または取り除くことが出来る。また、メモリモジュールのアクセスタイミングにあわせたネットワークアービトレーションを行う事により、従来のようにネットワークを占有しつづけることがなくなり、ネットワークリソースを有効に利用できる。

【0037】

【発明の効果】

以上説明したように、本発明によれば、ネットワークリソースを有効に利用することを可能としたデータ転送システムを提供することができる。

【図面の簡単な説明】

【図 1】本発明に係るデータ転送システムの実施形態を示すブロック図である。

【図 2】メジャーサイクルを説明するための図面である

【図 3】メジャーサイクルを用いて転送処理のパイプライン化を説明するための図面である。

【図 4】本実施形態に係るデータ転送システムにおける各部の処理を説明するためのタイミングチャートである。

【図 5】本実施形態の転送コントローラの詳細を説明するためのブロック図である。

【図 6】本実施形態のマクロアービタのアービトレーション例を説明するための図面であり、(a) は、リクエストから投入されたデータ転送要求の図表であり、(b) は、モジュールアービタの処理を説明するための図面である。

【図 7】本実施形態のネットワークアービタのアービトレーション例を説明するための図面であり、(a) は、リクエストから投入されたデータ転送要求の図表であり、(b) は、ネットワークアービタの処理を説明するための図面である。

【図 8】第 2 の実施形態の転送コントローラの詳細を説明するためのブロック図である。

【図 9】第 3 の実施形態の転送コントローラの詳細を説明するためのブロック図である。

【図 10】第 4 の実施形態の転送コントローラの詳細を説明するためのブロック図である。

【図 11】第 1 の従来例に係るデータ転送システムを示すブロック図である。

【図 12】第 2 の従来例に係るデータ転送システムを示すブロック図である。

【符号の説明】

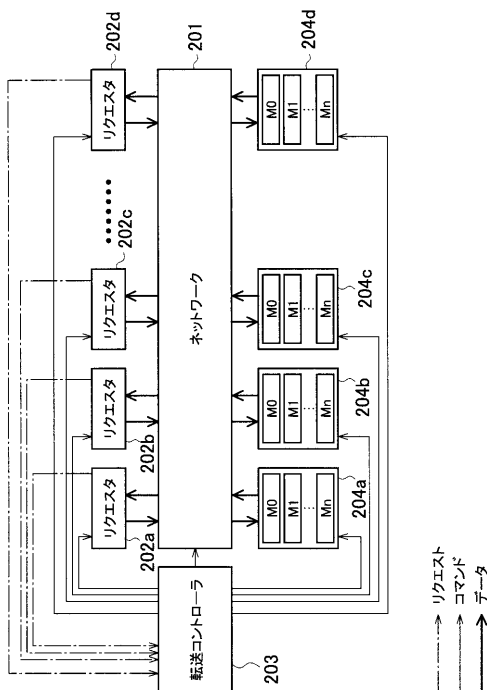
101, 106, 201 ネットワーク

102 a, 102 b, 102 c, 102 d, 107 a, 107 b, 107 c, 107 d,

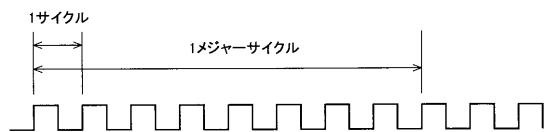
202 a, 202 b, 202 c, 202 d リクエスト

- 103, 108 ネットワークアービター
- 104a, 104b, 104c, 104d, 109a, 109b, 109c, 109d
- メモリコントローラ
- 105a, 105b, 105c, 105d, 110a, 110b, 110c, 110d
- メモリモジュール
- 203 転送コントローラ
- 204a, 204b, 204c, 204d モジュール
- 211, 217 リクエスト保持部
- 212 アドレスデコード部
- 213, 216 モジュールアービトレーション部
- 214 ネットワークアービトレーション部
- 215 転送指示部

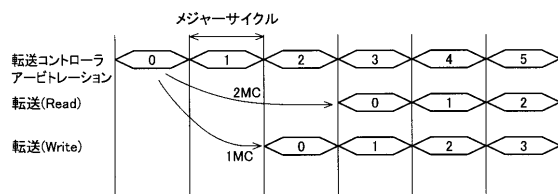
【図1】



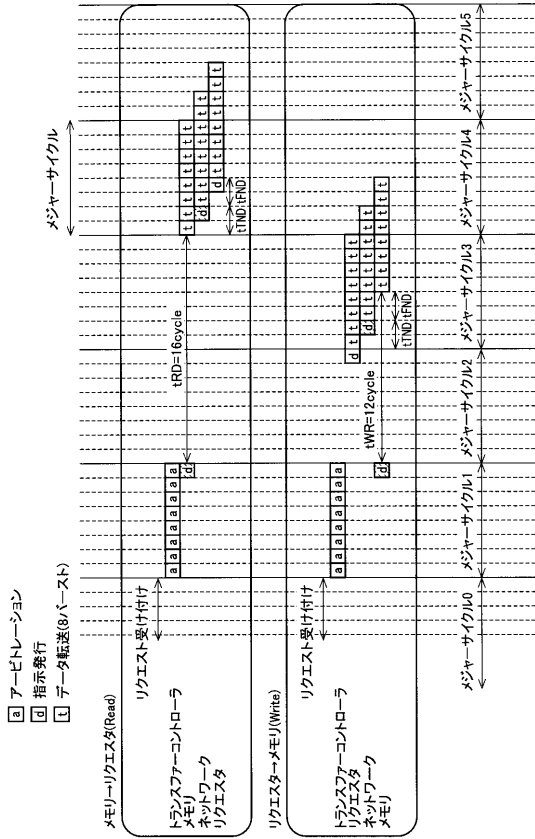
【図2】



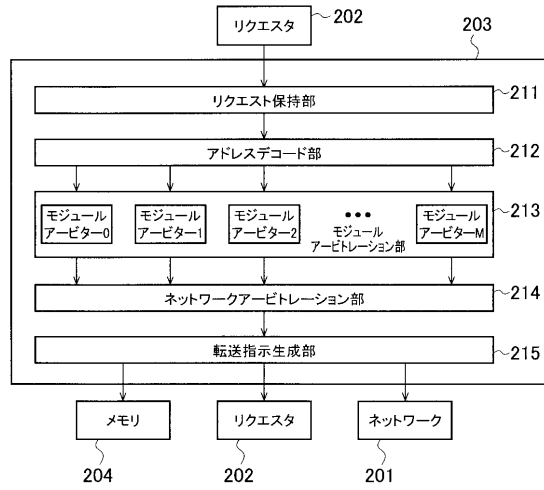
【図3】



【 図 4 】



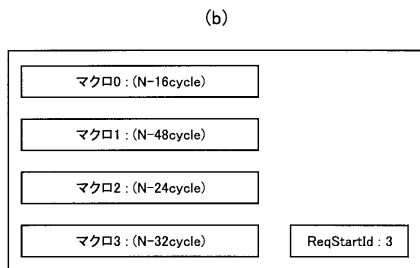
【 図 5 】



【 図 6 】

(a)

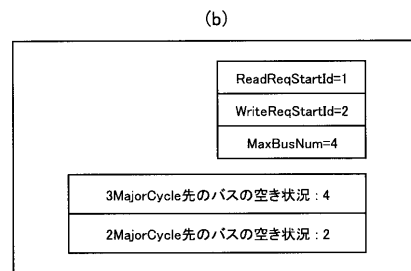
reqid	マクロ番号
0	3
1	0
2	1
3	0
4	2
5	3



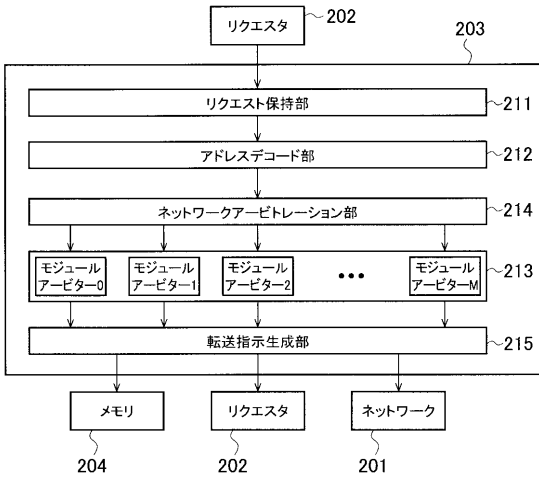
【 図 7 】

(a)

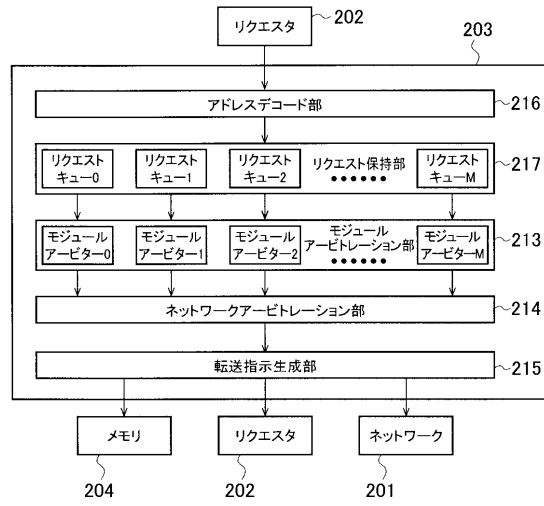
reqid	Read/Write
1	Write
2	Read
4	Write
5	Write
7	Read
8	Read
9	Write



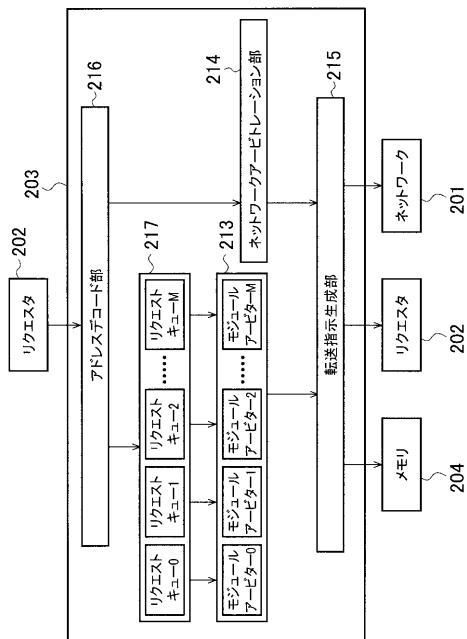
【 図 8 】



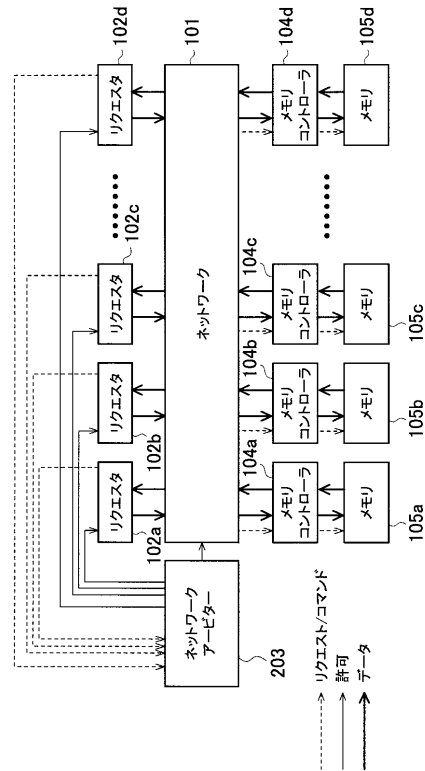
【 図 9 】



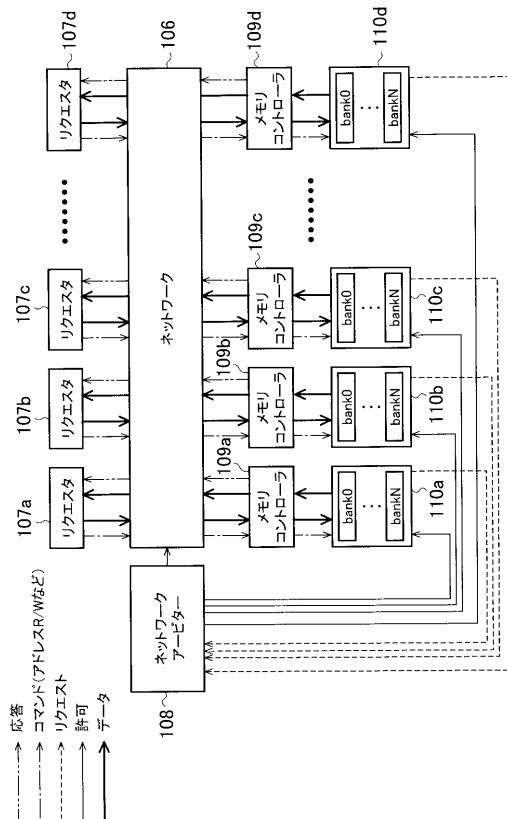
【 図 10 】



【 図 11 】



【図 12】



【手続補正書】

【提出日】平成15年11月18日(2003.11.18)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項1】

データ転送システムにおいて、

データ転送要求を送信し、許可された場合には、データ転送を行う複数のリクエストと、前記複数のリクエストからデータ転送要求を受信し、前記受信したデータ転送要求のアービトレーションを行って許可するデータ転送要求を決定し、転送指示を所定のタイミングで送信する転送コントローラと、

前記転送指示を受信し、前記転送指示に基づいて許可されたリクエストからのデータを転送するネットワークと、

個々にデータの入出力部を有する複数のモジュールを有し、前記転送指示を受信し、前記転送指示に基づいて前記ネットワークから転送データを受信するメモリと、

を有することを特徴とするデータ転送システム。

【請求項2】

前記転送コントローラは、前記メモリのアービトレーションを行った後に、前記ネットワークのアービトレーションを行うことを特徴とする請求項1記載のデータ転送システム。

【請求項3】

前記転送コントローラは、前記ネットワークのアービトレーションを行った後に、前記メモリのアービトレーションを行うことを特徴とする請求項1記載のデータ転送システム。

【請求項 4】

前記転送コントローラは、前記ネットワークのアービトレーションと前記メモリのアービトレーションを並列に行うことを特徴とする請求項 1 記載のデータ転送システム。

【請求項 5】

前記転送コントローラは、

前記複数のリクエストからのデータ転送要求を受け付けるリクエスト保持部と、

前記リクエスト保持部で保持されたデータ転送要求のアドレスをデコードするアドレスデコード部と、

前記デコードされたデータ転送要求について、前記メモリの各モジュール毎にアービトレーションを行うモジュールアービトレーション部と、

前記デコードされたデータ転送要求について、ネットワークを割り当てるためのアービトレーションを行うネットワークアービトレーション部と、

前記モジュールアービトレーション部及びネットワークアービトレーション部によって使用権を割り当てられたリクエストに係る転送指示を生成する転送指示生成部と、

を有することを特徴とする請求項 1 乃至 4 のいずれか一項に記載のデータ転送システム。

フロントページの続き

(74)代理人 100098327

弁理士 高松 俊雄

(72)発明者 藤原 崇

神奈川県川崎市幸区小向東芝町1番地 株式会社東芝マイクロエレクトロニクスセンター内

Fターム(参考) 5B060 CD01