

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4257626号
(P4257626)

(45) 発行日 平成21年4月22日 (2009. 4. 22)

(24) 登録日 平成21年2月13日 (2009. 2. 13)

(51) Int. Cl.

F 1

G 0 6 F 12/02 (2006. 01)

G 0 6 F 12/10 (2006. 01)

G 0 6 F 13/14 (2006. 01)

G 0 6 F 12/02 5 8 0 D

G 0 6 F 12/02 5 7 0 A

G 0 6 F 12/02 5 6 0 D

G 0 6 F 12/10 5 0 5 A

G 0 6 F 13/14 3 2 0 H

請求項の数 6 (全 15 頁)

(21) 出願番号 特願平11-277735

(22) 出願日 平成11年9月30日 (1999. 9. 30)

(65) 公開番号 特開2001-101072 (P2001-101072A)

(43) 公開日 平成13年4月13日 (2001. 4. 13)

審査請求日 平成17年8月30日 (2005. 8. 30)

(73) 特許権者 591044164

株式会社沖データ

東京都港区芝浦四丁目11番22号

(74) 代理人 100082050

弁理士 佐藤 幸男

(74) 代理人 100102923

弁理士 加藤 雄二

(72) 発明者 吉田 善彦

東京都港区芝浦四丁目11番地22号 株

式会社 沖データ内

審査官 多賀 実

(56) 参考文献 特開平09-152988 (JP, A)

特開平01-129345 (JP, A)

最終頁に続く

(54) 【発明の名称】 メモリアクセス装置

(57) 【特許請求の範囲】

【請求項1】

連続した論理アドレスを、アドレス変換テーブルを用いて主記憶装置のメモリ空間の物理アドレスに変換することにより主記憶装置のメモリ空間を連続エリアと仮想する仮想記憶方式を、主記憶装置における所定のメモリ空間をリング構造とみなして周回するようにアクセスする制御方式に適用して前記主記憶装置のメモリ空間をアクセスするメモリアクセス装置において、

前記アドレス変換テーブルには、複数のアドレスが格納され、

前記主記憶装置に対するアクセスを要求するデバイス毎に間接テーブルを備え、該間接テーブルは、アドレス変換テーブルのアドレスを指定する複数のアドレスデータを格納し、対応するデバイスからのアクセスが終了したときに格納されたアドレスデータが所定テーブルのテーブルサイズの中で周回するように更新され、

前記間接テーブルの前記テーブルサイズは、前記アドレス変換テーブルのテーブルサイズよりも小さく、

前記間接テーブルの複数のアドレスデータのアドレス総数は、前記アドレス変換テーブルのテーブルサイズと等しく、

前記アドレスデータが、アドレスの総数の周期で周回するように、前記間接テーブルが更新されるように構成されたことを特徴とするメモリアクセス装置。

【請求項2】

前記デバイスのアクセス終了後、間接テーブルに格納されたアドレス変換テーブルのア

10

20

ドレスデータが所定テーブルサイズの中で周回するように間接テーブルを更新する間接テーブル更新手段を備えたことを特徴とする請求項 1 に記載のメモリアクセス装置。

【請求項 3】

連続した論理アドレスを、アドレス変換テーブルを用いて主記憶装置のメモリ空間の物理アドレスに変換することにより主記憶装置のメモリ空間を連続エリアと仮想する仮想記憶方式を、主記憶装置における所定のメモリ空間をリング構造とみなして周回するようにアクセスする制御方式に適用して前記主記憶装置のメモリ空間をアクセスするメモリアクセス装置において、

前記主記憶装置に対するアクセスを要求するデバイス毎に間接テーブルを備え、該間接テーブルは、アドレス変換テーブルのアドレスを指定するアドレスデータを格納し、対応するデバイスからのアクセスが終了したときに格納されたアドレスデータが所定テーブルのテーブルサイズの中で周回するように更新されるように構成され、

前記デバイスのアクセス終了後、前記間接テーブルに格納されたアドレス変換テーブルのアドレスデータが所定テーブルサイズの中で周回するように、前記間接テーブルが更新される間接テーブル更新手段を備え、

前記間接テーブル更新手段は、間接テーブルに格納されたアドレス変換テーブルのアドレスデータに間接テーブルのテーブルサイズを加算する加算手段と、

該加算手段により加算された加算データをアドレス変換テーブルのテーブルサイズから減算する減算手段と、

該減算手段による減算の結果、減算データが零未満のときは、格納されたアドレス変換テーブルのアドレスデータを加算データに書き換え、減算データが零のときは、格納されたアドレス変換テーブルのアドレスデータを零に書き換える書き換え手段と、
を備えたことを特徴とするメモリアクセス装置。

【請求項 4】

主記憶装置にデータを書き込む書き込み要求が書き込み用デバイスからあり、主記憶装置からデータを読み出す読み出し要求が読み出し用デバイスからあったとき、主記憶装置へのアクセスの競合を回避するように、前記書き込み用デバイス及び読み出し用デバイスのアクセスを制御するアクセス制御手段を備えたことを特徴とする請求項 1 ～ 請求項 3 のいずれか 1 つに記載のメモリアクセス装置。

【請求項 5】

前記アクセス制御手段は、書き込み用デバイスによってデータが書き込まれた書き込みアドレスと読み出し用デバイスによってデータが読み出された読み出しアドレスとを比較して主記憶装置に書き込み可能な領域があるか読み出し可能なデータがあるかを判定する判定手段と、

該判定手段により、書き込み可能な領域があると判定されたときは、書き込み用デバイスに書き込みを許可する書き込み要求許可手段と、

前記判定手段により、読み出し可能なデータがあると判定され、書き込み用デバイスが書き込み要求をしていないときに読み出し用デバイスに読み出しを許可する読み出し要求許可手段と、

を備えたことを特徴とする請求項 4 に記載のメモリアクセス装置。

【請求項 6】

前記間接テーブルに格納されているアドレスデータの更新すべきデータを予め演算して格納するレジスタテーブルを備えたことを特徴とする請求項 1 ～ 請求項 5 のいずれか 1 つに記載のメモリアクセス装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、仮想記憶方式にメモリバッファ制御を適用したメモリアクセス装置に関する。

【0002】

【従来の技術】

10

20

30

40

50

あるデバイスがメモリ空間上にある開始アドレスと終了アドレスとにはさまれた空間を開始アドレスから終了アドレスまでをアクセスし、繰り返し、開始アドレスに戻るよう、リング状にアクセスするメモリアクセス装置が知られている。

このようなメモリ資源の使用方法は、データのフロー制御等で一時的にデータを保持しておくリングバッファとしてよく用いられている。

【0003】

このようなリングバッファを用いることにより、メモリ資源を効率よく使用することができ、しかもアドレス変換の制御が簡単になる。

また、計算機アーキテクチャにおいて、主記憶装置の一部を使い、プログラムが仮想的に連続エリアとしてメモリをアクセスする考え方がある。仮想記憶方式はこの考え方に基づいた方式であって、デバイスが論理アドレスを指定し、所定のアドレス変換テーブルを参照することにより、その論理アドレスを主記憶装置の物理アドレスに変換する方式である。その代表的な方式としてページング方式がある。

【0004】

この仮想記憶方式は一般的にソフト制御で行われるが、ハード制御によるリングバッファ制御にも応用でき、上位変換テーブルを動的に書き換えることによりリングバッファを実現することができる。

【0005】

【発明が解決しようとする課題】

ところで、従来のこのようなメモリアクセス装置では、2つのデバイスが同じメモリ空間をアクセスする場合、データ読み出しのための入力デバイスとデータ書き込み用の出力デバイスとでアクセスすべきメモリ空間が異なることが考えられる。このため、アドレス変換テーブルを複数備える必要がある。

【0006】

また、高速化のため、F/F（フリップフロップ）回路などでアドレス変換テーブルを構成した場合、アドレス変換テーブルを書き換える必要があることから複数のデバイスがF/F回路等を共通に使うことができず、デバイス毎に設ける必要がある。このため、回路規模が大きくなってしまう。

従って、できるだけ回路規模が大きくならないようにリングバッファ制御を行えるようにすることが好ましい。

【0007】

【課題を解決するための手段】

本発明は以上の点を解決するため次の構成を採用する。

構成1

請求項1の発明に係るメモリアクセス装置は、連続した論理アドレスを、アドレス変換テーブルを用いて主記憶装置のメモリ空間の物理アドレスに変換することにより主記憶装置のメモリ空間を連続エリアと仮想する仮想記憶方式を、主記憶装置における所定のメモリ空間をリング構造とみなして周回するようにアクセスする制御方式に適用して前記主記憶装置のメモリ空間をアクセスするものにおいて、前記アドレス変換テーブルには、複数のアドレスが格納され、前記主記憶装置に対するアクセスを要求するデバイス毎に間接テーブルを備え、該間接テーブルが、アドレス変換テーブルのアドレスを指定する複数のアドレスデータを格納し、対応するデバイスからのアクセスが終了したときに格納されたアドレスデータが所定テーブルのテーブルサイズの中で周回するように更新され、前記間接テーブルの前記テーブルサイズは、前記アドレス変換テーブルのテーブルサイズよりも小さく、前記間接テーブルの複数のアドレスデータのアドレス総数は、前記アドレス変換テーブルのテーブルサイズと等しく、前記アドレスデータが、アドレスの総数の周期で周回するように、前記間接テーブルが更新されるように構成されている。

【0008】

構成2

請求項2の発明に係るメモリアクセス装置では、前記デバイスのアクセス終了後、間接テ

10

20

30

40

50

ーブルに格納されたアドレス変換テーブルのアドレスデータが所定テーブルサイズの中で周回するように間接テーブルを更新する間接テーブル更新手段を備えている。

【 0 0 0 9 】

構成 3

請求項 3 の発明に係るメモリアクセス装置では、連続した論理アドレスを、アドレス変換テーブルを用いて主記憶装置のメモリ空間の物理アドレスに変換することにより主記憶装置のメモリ空間を連続エリアと仮想する仮想記憶方式を、主記憶装置における所定のメモリ空間をリング構造とみなして周回するようにアクセスする制御方式に適用して前記主記憶装置のメモリ空間をアクセスするメモリアクセス装置において、前記主記憶装置に対するアクセスを要求するデバイス毎に間接テーブルを備え、該間接テーブルは、アドレス変換テーブルのアドレスを指定するアドレスデータを格納し、対応するデバイスからのアクセスが終了したときに格納されたアドレスデータが所定テーブルのテーブルサイズの中で周回するように更新されるように構成され、前記デバイスのアクセス終了後、前記間接テーブルに格納されたアドレス変換テーブルのアドレスデータが所定テーブルサイズの中で周回するように前記間接テーブルを更新する間接テーブル更新手段を備え、前記間接テーブル更新手段が、間接テーブルに格納されたアドレス変換テーブルのアドレスデータに間接テーブルのテーブルサイズを加算する加算手段と、該加算手段により加算された加算データをアドレス変換テーブルのテーブルサイズから減算する減算手段と、該減算手段による減算の結果、減算データが零未満のときは、格納されたアドレス変換テーブルのアドレスデータを加算データに書き換え、減算データが零のときは、格納されたアドレス変換

10

20

【 0 0 1 0 】

構成 4

請求項 4 の発明に係るメモリアクセス装置では、主記憶装置にデータを書き込む書き込み要求が書き込み用デバイスからあり、主記憶装置からデータを読み出す読み出し要求が読み出し用デバイスからあったとき、主記憶装置へのアクセスの競合を回避するように、前記書き込み用デバイス及び読み出し用デバイスのアクセスを制御するアクセス制御手段を備えている。

【 0 0 1 1 】

構成 5

請求項 5 の発明に係るメモリアクセス装置では、前記アクセス制御手段が、書き込み用デバイスによってデータが書き込まれた書き込みアドレスと読み出し用デバイスによってデータが読み出された読み出しアドレスとを比較して主記憶装置に書き込み可能な領域があるか読み出し可能なデータがあるかを判定する判定手段と、該判定手段により、書き込み可能な領域があると判定されたときは、書き込み用デバイスに書き込みを許可する書き込み要求許可手段と、前記判定手段により、読み出し可能なデータがあると判定され、書き込み用デバイスが書き込み要求をしていないときに読み出し用デバイスに読み出しを許可する読み出し要求許可手段と、を備えている。

30

【 0 0 1 2 】

構成 6

請求項 6 の発明に係るメモリアクセス装置では、前記間接テーブルに格納されているアドレスデータの更新すべきデータを予め演算して格納するレジスタテーブルを備えている。

40

【 0 0 1 3 】

【発明の実施の形態】

以下、本発明の実施の形態を具体例を用いて説明する。

具体例 1

具体例 1 は、間接テーブルをデバイス毎に備え、デバイスがアドレス変換テーブルを間接的に参照することにより、アクセス空間が異なるデバイスが複数存在するときでもアドレス変換テーブルを 1 つにすることができるようにしたものである。

【 0 0 1 4 】

50

図 1 は、具体例 1 の構成を示すブロック図である。

主記憶装置 100 は、例えばプログラムを展開するための実記憶空間を有するメモリである。

【0015】

上位アドレス変換テーブル 200 は、ページング方式により連続した論理アドレス（仮想アドレス）を主記憶装置 100 の物理アドレス（実アドレス）に変換するためのテーブルであり、CPU 500 によってデータの読み書きが可能なレジスタを備えて構成されている。このレジスタに、主記憶装置 100 の使用可能エリアの上位アドレスが格納される。

【0016】

デバイス 300 A、B は、主記憶装置 100 に読み書きするためのデバイスであり、それぞれデータ転送部 301 A を備えている。

10

尚、具体例 1 では、デバイス 300 A は外部からデータを取り込んで主記憶装置 100 に書き込む入力専用の入力デバイス、デバイス 300 B は主記憶装置 100 に書き込まれたデータを読み出して外部に出力する出力専用の出力デバイスとして説明する。

【0017】

テーブル参照部 400 は、デバイス 300 A、B が上位アドレス変換テーブル 200 を参照できるように制御するブロックであり、間接テーブル 410 A、B と、間接テーブル更新部 420 A、B と、終了アドレス検出部 430 A、B と、アービタ 440 と、を備えて構成されている。

20

【0018】

間接テーブル 410 A、B は、それぞれデバイス 300 A、B に対応するように備えられ、CPU 500 によってデータの読み書きが可能なレジスタによって構成されたテーブルである。間接テーブル 410 A、B の各レジスタには上位アドレス変換テーブル 200 のアドレスが格納される。

【0019】

間接テーブル更新部 420 A、B は、それぞれ間接テーブル 410 A、B を更新してリングバッファ処理を行うための間接テーブル更新手段である。

間接テーブル 410 A、B 及び間接テーブル更新部 420 A、B の詳細な構成については後述する。

【0020】

30

アービタ 440 は、デバイス 300 A、B のメモリアクセスの競合を回避するためのブロックである。アービタ 440 の詳細な構成については後述する。

終了アドレス検出部 430 A、B は、それぞれデバイス 300 A、B の書き込み又は読み出しが終了したことを示す終了アドレスを検出するブロックであり、デバイス 300 A、B から出力されたデータの終了アドレスを検出し、間接テーブル更新部 420 A、B に終了タイミング信号を出力する。

【0021】

CPU 500 は、主記憶装置 100、上位アドレス変換テーブル 200 及び間接テーブル 410 A、B を制御するとともに、上位アドレス変換テーブル 200、間接テーブル 410 A、B に対してデータの読み書きを行う演算装置である。

40

【0022】

次に、間接テーブル更新部 420 A、B 及び間接テーブル 410 A、B の構成について説明する。

図 2 は間接テーブル 410 A、B 及び間接テーブル更新部 420 A、B の構成を示すブロック図である。

【0023】

具体例 1 の間接テーブル 410 A、B は、それぞれ 4 つのレジスタ 411 a ~ d を備えている。このレジスタ 411 a ~ d にそれぞれ上位アドレス変換テーブルのアドレスが格納される。

間接テーブル更新部 420 A、B は、書き込みタイミング発生部 421 と、セクタ 42

50

2 と、演算部 4 2 3 と、を備えている。

書き込みタイミング発生部 4 2 1 は、終了アドレス検出部 4 3 0 A 又は B から終了タイミング信号を入力して間接テーブル 4 1 0 A , B のデータを書き換える書き換えタイミング信号を作成し、この書き換えタイミング信号を各レジスタ 4 1 1 a ~ d に出力する。

【 0 0 2 4 】

セクタ 4 2 2 は、間接テーブル 4 1 0 A , B の各レジスタ 4 1 1 a ~ d のデータの演算部 4 2 3 への出力を切り換えるものである。

演算部 4 2 3 は、更新データを演算するものである。

【 0 0 2 5 】

図 3 は具体例 1 の演算部 4 2 3 の構成を示すブロック図である。

10

演算部 4 2 3 は、加算器 4 2 5 と、減算器 4 2 6 と、比較器 4 2 7 と、セクタ 4 2 8 と、を備えて構成されている。

【 0 0 2 6 】

加算器 4 2 5 は、各レジスタ 4 1 1 a ~ d のデータと間接テーブル長とを入力して両データの加算を行う加算手段である。尚、この間接テーブル長は、間接テーブル 4 1 0 A , B のサイズを示すデータであり、具体例 1 の間接テーブル長は「 4 」である。

減算器 4 2 6 は、加算器 4 2 5 の出力データが上位アドレス変換テーブル長を超えるかどうかを判定するために、上位アドレス変換テーブル長から加算器 4 2 5 の出力データを減算する減算手段である。尚、上位アドレス変換テーブル長は上位アドレス変換テーブル 2 0 0 のサイズを示すデータである。

20

【 0 0 2 7 】

セクタ 4 2 8 は、比較器 4 2 7 によって制御されて加算器 4 2 5 の出力データ及び減算器 4 2 6 の出力データのうち、いずれか一方を選択出力するものである。

比較器 4 2 7 は、減算器 4 2 6 によって減算された減算データを「 0 」と比較することにより加算器 4 2 5 の加算データが上位アドレス変換テーブル長と等しいかどうかを判定し、加算器 4 2 5 の出力データが上位アドレス変換テーブル長と等しくないときは、セクタ 4 2 8 から加算器 4 2 5 の出力データが出力され、等しいときは減算器 4 2 6 の出力データが出力されるようにセクタ 4 2 8 を制御する。

このセクタ 4 2 8 と比較器 4 2 7 とが書き換え手段に相当する。

【 0 0 2 8 】

30

次に、アービタ 4 4 0 の構成について説明する。

図 4 は、具体例 1 のアービタ 4 4 0 の構成を示すブロック図である。

書き込みアドレス記憶部 4 4 1 は、間接テーブル 4 1 0 A に最後に書き込んだアドレスに、1 を加えたアドレスを保持するブロックである。

【 0 0 2 9 】

読み出しアドレス記憶部 4 4 2 は、間接テーブル 4 1 0 B から最後に読み出したアドレスに、1 を加えたアドレスを保持するブロックである。

セクタ 4 4 3 は、デバイス 3 0 0 A から書き込み要求信号が入力されたとき、書き込み要求を行ったデバイス 3 0 0 A からのアドレス及び書き込みアドレス記憶部 4 4 1 により保持されたアドレスのうち、いずれか一方を書き込みアドレスとして選択出力するブロックである。

40

【 0 0 3 0 】

セクタ 4 4 4 は、デバイス 3 0 0 B から読み出し要求信号が入力されたとき、読み出し要求を行ったデバイス 3 0 0 B からのアドレス及び書き込みアドレス記憶部 4 4 1 により保持されたアドレスのうち、いずれか一方を読み出しアドレスとして選択出力するブロックである。

【 0 0 3 1 】

アドレス比較部 4 4 5 は、セクタ 4 4 3 から出力された書き込みアドレス及びセクタ 4 4 4 から出力された読み出しアドレスを比較することにより書き込み可能な領域があるか読み出し可能なデータがあるかを判定する判定手段であり、書き込み可能な領域がある

50

と判定したときは、書き込み要求受付信号作成部 4 4 6 に書き込み可能信号を出力し、読み出し可能なデータがあると判定したときは、読み出し要求受付信号作成部 4 4 7 に読みだし可能信号を出力する。

【 0 0 3 2 】

書き込み要求受付信号作成部 4 4 6 は、デバイス 3 0 0 A、アドレス比較部 4 4 5 から、それぞれ書き込み要求信号、書き込み可能信号が入力されたとき、デバイス 3 0 0 A にデータの書き込みを許可する書き込み受付信号を出力する書き込み要求許可手段である。また、書き込み要求受付信号作成部 4 4 6 は、データの書き込み中、読み出し要求受付信号作成部 4 4 7 に書き込み中信号を出力する。

【 0 0 3 3 】

読み出し要求受付信号作成部 4 4 7 は、デバイス 3 0 0 B、アドレス比較部 4 4 5 から、それぞれ読み出し要求信号、読み出し可能信号が入力され、さらに書き込み要求受付信号作成部 4 4 6 から書き込み中信号が出力されていないときに、デバイス 3 0 0 B にデータの読み出しを許可する読み出し受付信号を出力する読み出し要求許可手段である。

【 0 0 3 4 】

動作

ページング方式では、プログラムを主記憶装置 1 0 0 上に展開して動作させるとき、プログラムが一定の長さのいくつかのページに区切られて主記憶装置 1 0 0 上に配置される。従って、プログラムが格納されているエリアは、メモリ資源を確保するため、通常は不連続となる。

このため、上位アドレス変換テーブル 2 0 0 を使用して連続した論理アドレスが主記憶装置 1 0 0 上の物理アドレスに変換される。

【 0 0 3 5 】

図 5 は、具体例 1 の動作説明図である。

尚、具体例 1 では、分かりやすくするため、この図 5 に示すようにプログラムが 256×4 のサイズで収まるものとして説明する。

また、主記憶装置 1 0 0 は 32 ビットのアドレス空間を有し、1 エリアは、256 (8 ビット) 毎に区切られているものとし、デバイス 3 0 0 A, B がアクセスできるアドレス空間を 10 ビット、使用できるエリアをエリア 0 ~ 6 の 7 エリアとする。従って、上位アドレス変換テーブル長は、「7」となる。尚、ここでは、1 ワードのビット数は特に限定されない。

【 0 0 3 6 】

この場合、プログラムのアドレス空間は 256×4 なので $8 + 2 = 10$ ビットということになる。下位 8 ビットは主記憶装置 1 0 0 のアドレスとしてそのまま使用することができるので変換するのは上位 2 ビットとなる。

【 0 0 3 7 】

このプログラム空間の上位 2 ビットのアドレスが上位アドレス変換テーブル 2 0 0 のアドレスとなる。テーブルの大きさは、テーブル 1 アドレス当たりのビット数は主記憶装置 1 0 0 の上位 24 ビットであり、これが 4 つ分 (アドレス分) ということになる。また、上位アドレス変換テーブル 2 0 0 のレジスタ 2 1 0 ~ 2 1 6 には、それぞれ主記憶装置 1 0 0 のエリア 0 ~ 6 の上位 24 ビットのアドレスデータが入力される。

【 0 0 3 8 】

このように、上位 2 ビットのアドレスを上位アドレス変換テーブル 2 0 0 を通して変換し、下位のアドレスをそのまま使用することにより、あたかもプログラムがエリア 0 のアドレスから起動されたように動作させることができる。

具体的に、アドレス変換を行うには、まず、上位アドレス変換テーブル 2 0 0 の各レジスタ 2 1 0 ~ 2 1 6 に、それぞれ主記憶装置 1 0 0 のエリア 0 ~ 6 の上位アドレスを格納し、主記憶装置 1 0 0 上でのデバイスが使用可能なエリア 0 ~ 6 を上位アドレス変換テーブル 2 0 0 の各レジスタ 2 1 0 ~ 2 1 6 に割り当てる。これはプログラムによって実行される。

10

20

30

40

50

【 0 0 3 9 】

このように割り当てが行われた後、プログラムによってデバイス 3 0 0 A 用の間接テーブル 4 1 0 A の初期化が行われる。

間接テーブル 4 1 0 A の初期化を行うには、デバイス 3 0 0 A 用の間接テーブル 4 1 0 A のレジスタ 4 1 1 a に上位アドレス変換テーブル 2 0 0 のレジスタ 2 1 0 を指定するデータ「 0 」をセットする。同様にして間接テーブル 4 1 0 A のレジスタ 4 1 1 b ~ d に、それぞれ上位アドレス変換テーブル 2 0 0 のレジスタ 2 1 1 ~ 2 1 3 を指定するデータ「 1 」 ~ 「 3 」をセットする。

【 0 0 4 0 】

デバイス 3 0 0 B 用の間接テーブル 4 1 0 B についても同様に初期化が行われる。

10

間接テーブル 4 1 0 A の初期化後、プログラムによりデバイス 3 0 0 A に起動をかけて入力動作を開始させる。

デバイス 3 0 0 A が入力動作を開始して主記憶装置 1 0 0 にデータを書き込むとき、アドレス変換が行われる。

【 0 0 4 1 】

図 5 の例では、デバイス 3 0 0 A によって例えばヘクサ表現でアドレス「 0x2a5 」（「 0x 」はヘクサ表現であることを示す。）が指定される。このとき、上位 2 ビットは「 2 」であるので、間接テーブル 4 1 0 A のアドレスは「 2 」となり、レジスタ 4 1 1 c のデータ「 2 」が得られる。このデータ「 2 」が上位アドレス変換テーブル 2 0 0 のアドレスデータとなる。

20

【 0 0 4 2 】

このアドレスデータ「 2 」に基づいて上位アドレス変換テーブル 2 0 0 のレジスタに格納されているデータ「 0X000007 」が得られる。

このデータ「 0X000007 」は、主記憶装置 1 0 0 の上位 2 4 ビットのアドレスデータであるので、このデータ「 0X000007 」とデバイス 3 0 0 B の下位 8 ビットのデータ「 0xa5 」に基づいて主記憶装置 1 0 0 のアドレス「 0X000007a5 」が得られる。

【 0 0 4 3 】

そして、このアドレス「 0X000007a5 」を指定して主記憶装置 1 0 0 のエリア 6 に格納されているデータが読み出される。

主記憶装置 1 0 0 へのデータの書き込みが終了したとき、アクセスの終了タイミングが終了アドレス検出部 4 3 0 によって検出され、終了信号が間接テーブル 4 1 0 A の書き込みタイミング発生部 4 2 1 に入力される。

30

【 0 0 4 4 】

間接テーブル更新部 4 2 0 A はこの信号をトリガとして間接テーブル 4 1 0 A のデータを更新する。

更新は以下のようにして行われる。

まず、セクタ 4 2 2 が制御されて間接テーブル 4 1 0 A のレジスタ 4 1 1 a のデータが演算部 4 2 3 に入力される。

【 0 0 4 5 】

演算部 4 2 3 では、このデータに間接テーブル長が加算器 4 2 5 により加算される。

40

例えば、レジスタ 4 1 1 a に格納されているデータが「 0 」のときは、データ「 0 」に「 4 」が加算されて「 4 」となる。このときは、更新されたデータ「 4 」がレジスタ 4 1 1 a に格納される。

【 0 0 4 6 】

同じようにして、セクタ 4 2 2 が切り換えられ、間接テーブル 4 1 0 A のレジスタ 4 1 1 a ~ d に格納されているデータが順次更新される。

間接テーブル 4 1 0 A のレジスタに格納されているデータが「 3 」のときは、加算値は「 3 」 + 「 4 」 = 「 7 」となり、上位アドレス変換テーブル長と等しくなる。このときは、セクタ 4 2 2 が比較器 4 2 7 によって切り換えられてデータ「 0 」が出力される。

このようにして主記憶装置 1 0 0 のエリア 6 を越えないように、間接テーブル 4 1 0 A の

50

更新が行われる。

【 0 0 4 7 】

図 6 は間接テーブル 4 1 0 A の遷移を説明するための説明図である。

図 6 に示すように、状態 1 では、間接テーブル 4 1 0 A のレジスタ 4 1 1 a ~ d には、それぞれ上位アドレス変換テーブル 2 0 0 のアドレスデータ 0 ~ 3 がセットされている。間接テーブル 4 1 0 A が更新されたとき、間接テーブル 4 1 0 A の各レジスタ 4 1 1 a ~ d が書き換えられて状態 1 は状態 2 のように変化し、この状態 2 において、さらに間接テーブル 4 1 0 A が更新されて状態 3 のように変化する。

【 0 0 4 8 】

読み出しができる程度にデータが蓄積されたとき、今度は、デバイス 3 0 0 B を起動する。このときも同じようにしてアドレス変換が行われ、データの読み出しが行われる。

10

【 0 0 4 9 】

そして、デバイス 3 0 0 B による主記憶装置 1 0 0 からのデータの読み出しが終了したときは、書き込み終了時と同様に読み出しアクセスの終了タイミングが終了アドレス検出部 4 3 0 によって検出され、終了タイミング信号が間接テーブル 4 1 0 A の書き込みタイミング発生部 4 2 1 に入力される。この信号がトリガとなって書き込みタイミング発生部 4 2 1 から書き換えタイミング信号が出力され、間接テーブル 4 1 0 A のデータが更新される。

【 0 0 5 0 】

デバイス 3 0 0 A により書き込み要求が行われたとき、あるいはデバイス 3 0 0 B によって読み出し要求が行われたとき、アービタ 4 4 0 によってメモリアクセスがコントロールされ、メモリアクセスの競合が回避される。

20

【 0 0 5 1 】

例えば、デバイス 3 0 0 A からアービタ 4 4 0 に書き込み要求信号が入力されてデバイス 3 0 0 A により書き込み要求が行われたとき、アドレス比較部 4 4 5 によって書き込みアドレスと読み出しアドレスとが比較される。そして、書き込みアドレスが読み出しアドレスよりも小さいときは、書き込み領域があると判定されてアドレス比較部 4 4 5 から書き込み要求受付信号作成部 4 4 6 に書き込み可能信号が出力される。

【 0 0 5 2 】

この書き込み可能信号が書き込み要求受付信号作成部 4 4 6 に出力されたとき、書き込み受付信号が書き込み要求受付信号作成部 4 4 6 からデバイス 3 0 0 A に出力され、デバイス 3 0 0 A により主記憶装置 1 0 0 へのデータの書き込みが行われる。

30

【 0 0 5 3 】

同様に、デバイス 3 0 0 B からアービタ 4 4 0 に読み出し要求信号が入力されて読み出し要求が行われたとき、書き込みアドレスが読み出しアドレスよりも小さいときは、読み出すデータがあるとアドレス比較部 4 4 5 によって判定され、アドレス比較部 4 4 5 から書き込み要求受付信号作成部 4 4 6 に書き込み可能信号が出力される。そして、書き込み要求受付信号作成部 4 4 6 から書き込み中信号が出力されていないときは、読み出し要求受付信号作成部 4 4 7 からデバイス 3 0 0 B に読み出し受付信号が出力され、デバイス 3 0 0 B により主記憶装置 1 0 0 からデータの読み出しが行われる。

40

【 0 0 5 4 】

一方、書き込みアドレスが読み出しアドレスよりも大きいときは、デバイス 3 0 0 A から書き込み要求信号が入力されても書き込み領域はないとアドレス比較部 4 4 5 によって判定され、アドレス比較部 4 4 5 から書き込み可能信号は出力されない。また、デバイス 3 0 0 B から読み出し要求信号が入力されても読み出すデータはないとアドレス比較部 4 4 5 によって判定されて読み出し可能信号は出力されない。このようにしてメモリアクセスの競合が回避され、デバイス 3 0 0 A が主記憶装置 1 0 0 にデータを書き込む前にデバイス 3 0 0 B によってデータの読み出しが行われなくなる。

【 0 0 5 5 】

具体例 1 の効果

50

以上、説明したように具体例 1 によれば、間接テーブル 4 1 0 A , B をデバイス 3 0 0 A , B 毎に備え、上位アドレス変換テーブル 2 0 0 を間接的に参照するようにしたので、上位アドレス変換テーブル 2 0 0 を複数設ける必要がなく、回路規模を大幅に削減することができる。

【 0 0 5 6 】

ここで、アドレス変換を目的とするテーブルを上位アドレス変換テーブルのみで構成する場合と間接テーブルを追加して構成する場合とで、その回路規模を比較する。

【 0 0 5 7 】

尚、ここでも、主記憶装置のメモリ空間を 3 2 ビットのアドレス空間、1 エリアのアドレス空間を 8 ビット、デバイスのアクセスできるアドレス空間を 1 0 ビットとし、使用できるエリア数を 7 つとする。

10

【 0 0 5 8 】

各テーブルを F / F 回路で構成した場合、1 つの上位アドレス変換テーブルは 24×7 ビットとなる。デバイスが 2 つのとき、これを 2 つ使うので、F / F 回路 $24 \times 7 \times 2 = 336$ の数だけ必要となる。

これに対して、間接テーブルを設けた場合、上位アドレス変換テーブルは 1 つで済むため、F / F 回路は、 $24 \times 7 + 3 \times 4 \times 2 = 192$ の数で済み、回路規模を削減できることになる。

【 0 0 5 9 】

尚、具体例 1 では、デバイスを 2 つとしたので、2 つの間接テーブルを必要としたが、間接テーブルの数はこの具体例 1 には限定されない。また、テーブル長も各デバイスのアクセス空間に応じて増減することもできる。

20

【 0 0 6 0 】

また、上位アドレス変換テーブルも主記憶装置の使用可能エリアの数に応じて増減することができ、主記憶装置の使用可能エリア取得の範囲を制限する事でテーブルの各アドレスのビット数を減らすことが可能になる。

【 0 0 6 1 】

例えば、主記憶装置のメモリアドレス空間が 3 2 ビットで使用可能エリアが常に主記憶装置のアドレス空間の後半と制限すれば、最上位の 1 ビットは常に固定となるためテーブルの全てのアドレスで最上位の 1 ビットを省くことができる。

30

【 0 0 6 2 】

また、デバイス空間が使用可能エリアの空間より大きい場合でも間接テーブルをデバイス空間に合わせたテーブル長にすれば処理可能となる。

例えば、デバイスのアドレス空間が 4 エリア分で実際に使えるエリアが 3 つのときは、間接テーブルの内容を “ 0 ”、“ 1 ”、“ 2 ”、“ 0 ” というように設定することにより、リングバッファとして使用可能となる。

【 0 0 6 3 】

また、間接テーブル更新部の演算部も間接テーブルのアドレス分備えることもでき、このようにすれば間接テーブルの更新を全て同時に行える。一方、間接テーブル更新部のセレクタの入力数を増やし、書き込みタイミング発生部を全デバイス共通のものにすることにより、全デバイスの間接テーブルの更新用の演算部が 1 つで済むことになる。

40

【 0 0 6 4 】

また、具体例 1 では、間接テーブル更新部の演算部をハードウェア構成としたが、ソフトウェアによって構成することもできる。

図 7 はその動作を示すフローチャートである。

ステップ (図中、ステップを「 S 」と記す。) 1 では、間接テーブル 4 1 0 A (又は B) のデータを入力する。

【 0 0 6 5 】

ステップ 2 では、間接テーブル 4 1 0 A のデータと間接テーブル長を加算する。

ステップ 3 では、この加算値を上位アドレス変換テーブル長と比較する。

50

そして、加算値が上位アドレス変換テーブル長よりも大きいときは、ステップ 4 に進み、上位アドレス変換テーブル長から加算値を減算する。

【 0 0 6 6 】

ステップ 5 では、この値を出力する。

また、加算値が上位アドレス変換テーブル長を越えたときは、ステップ 6 に進み、変数 B に加算値を代入し、ステップ 5 に進んでこの変数 B を出力する。

このように、演算部 4 2 3 をソフトウェアによって構成することもできる。

【 0 0 6 7 】

具体例 2

具体例 2 は、更新時間を短縮するため、間接テーブルの各レジスタと対応するレジスタテーブルを備え、あらかじめ更新用のデータを演算させてレジスタテーブルに格納するようにしたものである。

【 0 0 6 8 】

図 8 は、具体例 2 の構成を示すブロック図である。

具体例 2 の間接テーブル更新部 4 2 0 A , B は、それぞれ書き込み可能なレジスタ 4 5 2 a ~ d を有するレジスタテーブル 4 5 0 を備えている。

尚、具体例 1 と同一要素については同一符号を付して説明を省略する。

【 0 0 6 9 】

動作

次に具体例 2 の更新動作を説明する。

具体例 2 では、間接テーブル 4 1 0 A (又は B) の更新前に、間接テーブル 4 1 0 A のレジスタ 4 1 1 a のデータを読み取り、演算部 4 2 3 により更新データが作成される。

尚、具体例 2 においても、演算部 4 2 3 をハードウェアにより構成してもよいし、ソフトウェアによって構成してもよい。

【 0 0 7 0 】

そして、この演算結果がレジスタテーブル 4 5 0 のレジスタ 4 5 2 a に書き込まれる。

同様にして間接テーブル 4 1 0 A のレジスタ 4 1 1 b ~ d についてもデータが読み取られ、演算結果が各レジスタテーブル 4 5 0 の 4 5 2 b ~ d に書き込まれる。

【 0 0 7 1 】

間接テーブル 4 1 0 A の更新時、書き込みタイミング発生部 4 2 1 から書き換えタイミング信号が出力されたとき、このレジスタテーブル 4 5 0 のレジスタ 4 5 2 a ~ d のデータが間接テーブル 4 1 0 A の各レジスタ 4 1 1 a ~ d に同時に出力され、コピーされる。

【 0 0 7 2 】

尚、間接テーブル 4 1 0 の書き換えタイミングは、割り込み信号等を用いて CPU 5 0 0 にも通知される。

レジスタテーブル 4 5 0 から間接テーブル 4 1 0 へのコピーは通常 1 クロックで終了する。

そして、コピー終了後、次の更新に備えて再び更新用データがレジスタテーブル 4 5 0 に書き込まれる。

このように、レジスタテーブル 4 5 0 の書き換え、レジスタテーブル 4 5 0 から間接テーブル 4 1 0 A へのコピーを繰り返すことによりリングバッファが実現される。

【 0 0 7 3 】

具体例 2 の効果

以上、説明したように具体例 2 によれば、レジスタが間接テーブル 4 1 0 A , B の各レジスタと対応するレジスタテーブル 4 5 0 を備え、あらかじめ演算した更新用データを間接テーブル 4 1 0 A , B の更新時、同時に更新するようにしたので、更新時間を大幅に短縮することができる。従って、デバイス 3 0 0 A , B は主記憶装置 1 0 0 を、高速かつ連続的にアクセスすることができる。

【図面の簡単な説明】

【図 1】具体例 1 の構成を示すブロック図である。

10

20

30

40

50

【図 2】具体例 1 の間接テーブル及び間接テーブル更新部の構成を示すブロック図である。

【図 3】具体例 1 の演算部（間接テーブル更新部）の構成を示すブロック図である。

【図 4】具体例 1 のアービタの構成を示すブロック図である。

【図 5】具体例 1 の動作説明図である。

【図 6】具体例 1 の間接テーブルの遷移を示す説明図である。

【図 7】具体例 1 の演算部をソフトウェアで構成した場合のその動作を示すフローチャートである。

【図 8】具体例 2 の間接テーブル及び間接テーブル更新部の構成を示すブロック図である。

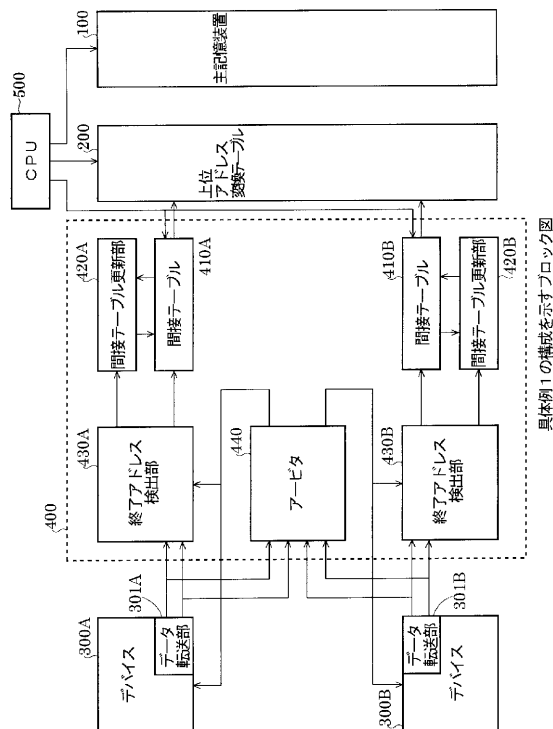
10

【符号の説明】

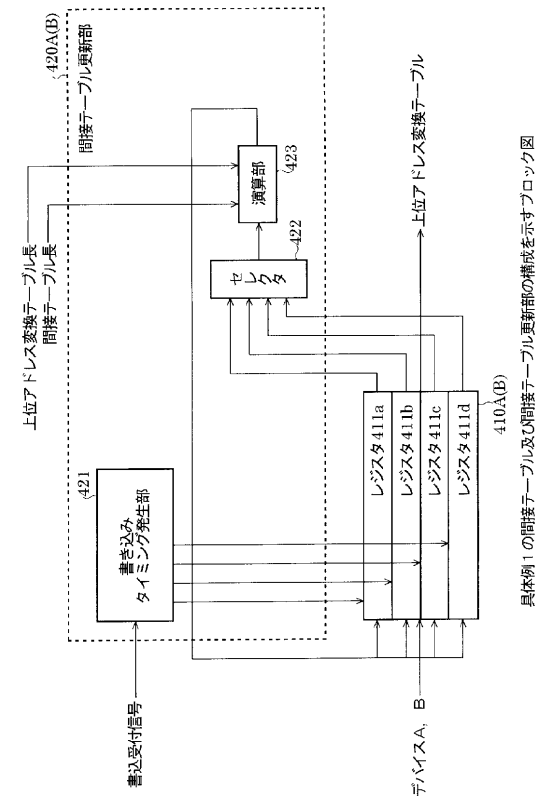
- 1 0 0 主記憶装置
- 2 0 0 上位アドレス変換テーブル
- 3 0 0 A, B デバイス
- 4 0 0 テーブル参照部
- 4 1 0 間接テーブル
- 4 2 0 間接テーブル更新部
- 4 4 0 アービタ
- 4 5 0 レジスタテーブル
- 5 0 0 C P U

20

【図 1】

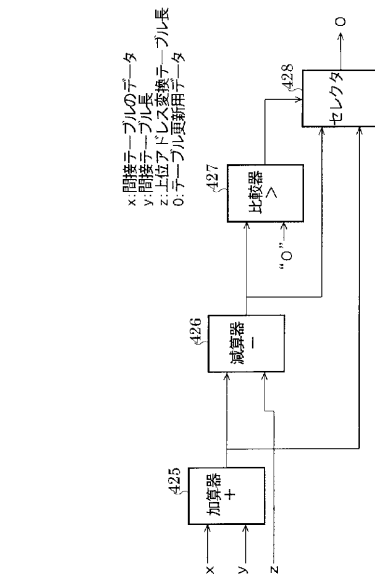


【図 2】



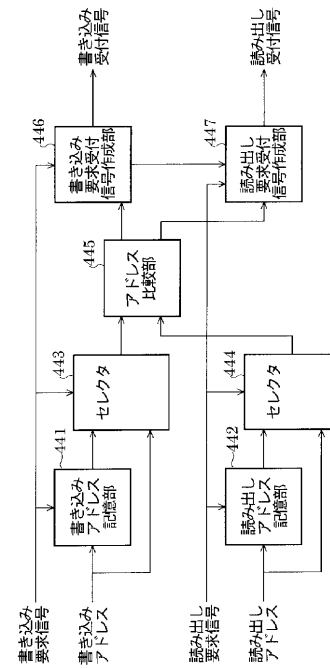
具体例 1 の間接テーブル及び間接テーブル更新部の構成を示すブロック図

【 図 3 】



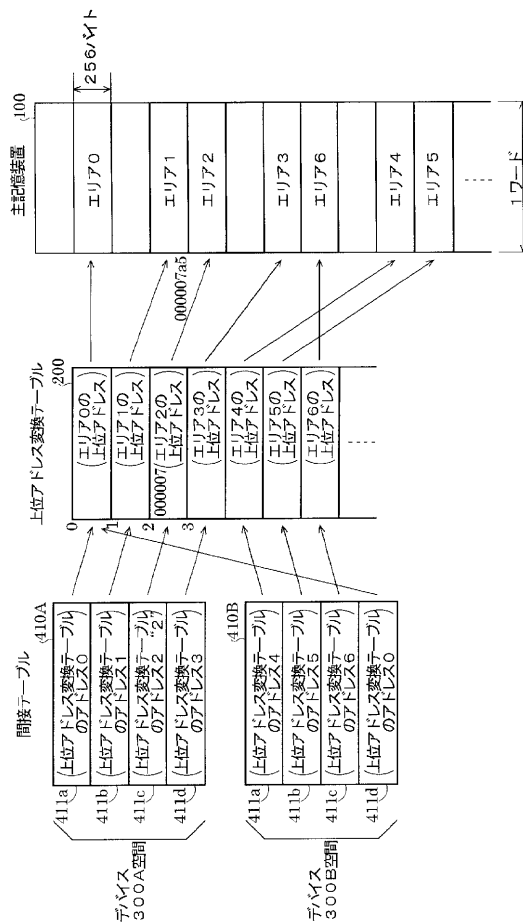
具体例 1 の間接テーブル更新部の演算部の構成を示すブロック図

【 図 4 】



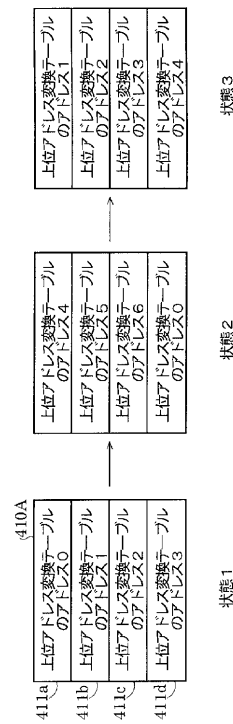
具体例1のアービタの構成を示すブロック図

【 図 5 】



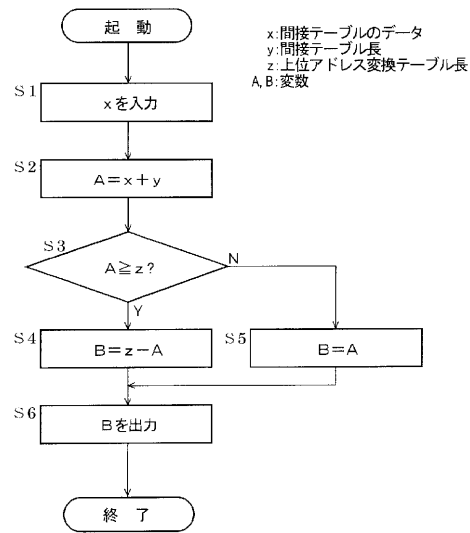
具体例 1 の動作説明図

【 図 6 】



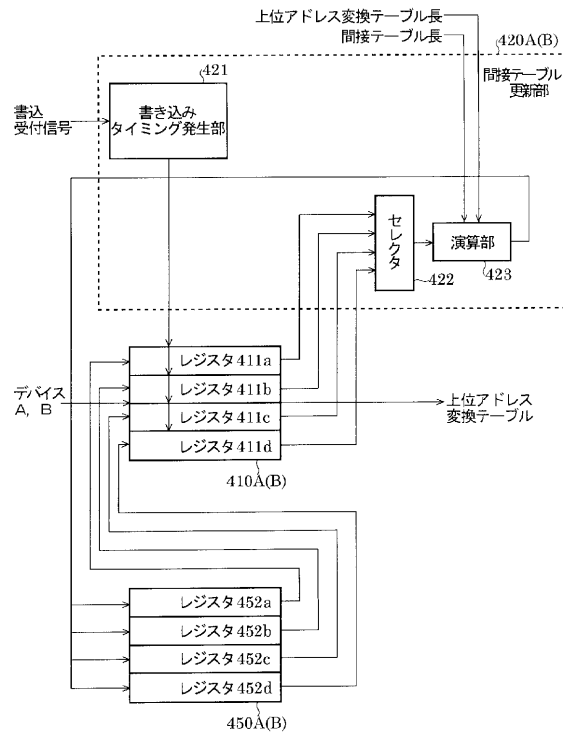
具体例1の間接テーブルの遷移を示す説明図

【図 7】



具体例 1 の演算部の動作を示すフローチャート

【図 8】



具体例 2 の間接テーブル及び間接テーブル更新部の構成を示すブロック図

フロントページの続き

(58)調査した分野(Int.Cl. , D B 名)

G06F12/00-12/06

G06F13/14