

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2022年1月13日(13.01.2022)



(10) 国際公開番号

WO 2022/009741 A1

(51) 国際特許分類:

G06F 9/52 (2006.01)

G06F 12/02 (2006.01)

G06F 9/54 (2006.01)

G06F 15/167 (2006.01)

(21) 国際出願番号: PCT/JP2021/024626

(22) 国際出願日: 2021年6月29日(29.06.2021)

(25) 国際出願の言語: 日本語

(26) 国際公開の言語: 日本語

(30) 優先権データ:

特願 2020-117161 2020年7月7日(07.07.2020) JP

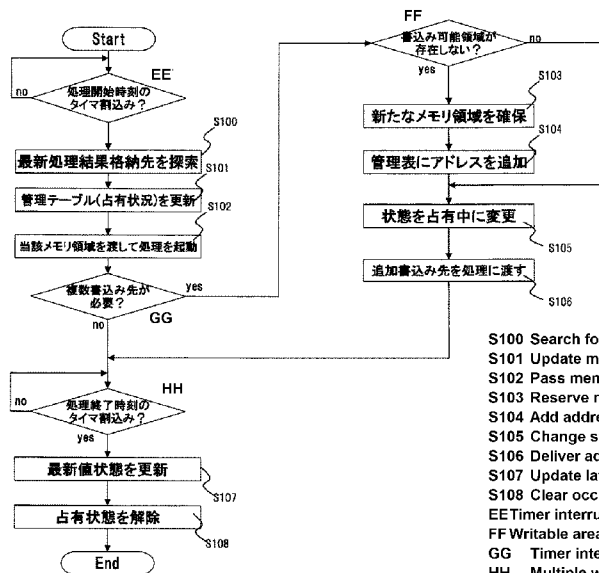
(71) 出願人: 日立 A s t e m o 株式会社(HITACHI
ASTEMO, LTD.) [JP/JP]; 〒3128503 茨城県ひ
たちなか市高場 2 5 2 0 番地 Ibaraki (JP).

(72) 発明者: 堀 口 辰 也 (HORIGUCHI Tatsuya);

〒1008280 東京都千代田区丸の内一丁目 6 番
6 号 株式会社日立製作所内 Tokyo (JP). 石郷
岡 祐(ISHIGOOKA Tasuku); 〒1008280 東京都
千代田区丸の内一丁目 6 番 6 号 株式会社日
立製作所内 Tokyo (JP). 大塚 敏史(OTSUKA
Satoshi); 〒1008280 東京都千代田区丸の内一
丁目 6 番 6 号 株式会社日立製作所内 Tokyo (JP).
芹沢 一(SERIZAWA Kazuyoshi); 〒3128503 茨
城県ひたちなか市高場 2 5 2 0 番地 日立
A s t e m o 株式会社内 Ibaraki (JP). 村上
隆(MURAKAMI Takashi); 〒3128503 茨城県ひ
たちなか市高場 2 5 2 0 番地 日立 A s t
e m o 株式会社内 Ibaraki (JP).

(54) Title: ELECTRONIC CONTROL DEVICE

(54) 発明の名称: 電子制御装置



S100 Search for destination in which latest processing result is stored
S101 Update management table (occupied state)
S102 Pass memory area and activate process
S103 Reserve new memory area
S104 Add address to management table
S105 Change state to being occupied
S106 Deliver additional write destination to process
S107 Update latest value state
S108 Clear occupied state
EE Timer interrupt for processing start time?
FF Writable area nonexistent?
GG Timer interrupt for processing end time?
HH Multiple write destinations needed?

(57) Abstract: Provided is an electronic control device with which it is possible to raise the efficiency of data delivery between applications in time-synchronized design. This electronic control device comprises: a processor including a plurality of cores operating on different cycles, said cores executing a plurality of processes; and a memory including a plurality of memory areas that can be individually accessed for each of the plurality of cores that execute a plurality of processes. The electronic control device changes, in accordance with the progress of the processes, the memory areas accessible by a preceding process that performs data writing and a subsequent process that performs data reading. The

(74) 代理人: 特許業務法人開知国際特許事務所
(KAICHI IP); 〒1030022 東京都中央区日本橋
室町四丁目3番16号 Tokyo (JP).

(81) 指定国(表示のない限り、全ての種類の国内保
護が可能): AE, AG, AL, AM, AO, AT, AU, AZ,
BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH,
CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,
HN, HR, HU, ID, IL, IN, IR, IS, IT, JO, KE, KG, KH,
KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY,
MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ,
NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT,
QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL,
ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG,
US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) 指定国(表示のない限り、全ての種類の広域保
護が可能): ARIPO (BW, GH, GM, KE, LR, LS,
MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM,
ZW), ユーラシア (AM, AZ, BY, KG, KZ, RU, TJ,
TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ,
DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT,
LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS,
SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM,
GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

添付公開書類:

一 国際調査報告(条約第21条(3))

processor searches for a writable memory area into which the latest result of processing has not been written yet and which is not in the middle of a read, reserves a new memory area when no writable memory areas exist, and sets the new memory area as the destination to which the result of processing is written, whereby the result of processing is delivered from the preceding process to the subsequent process.

(57) 要約: 時刻同期型設計におけるアプリケーション間におけるデータ受渡の効率を高めることが可能な電子制御装置が提供される。電子制御装置は、異なる周期で動作する複数の処理を実行する複数のコアを含むプロセッサと、複数の処理を実行する前記複数のコアごとにそれぞれアクセス可能な複数のメモリ領域を含むメモリと、を備え、データ書込みを行う先行処理とデータ読出しを行う後続処理とがそれぞれアクセス可能な前記メモリ領域を処理の進捗に応じて変更する。プロセッサは、最新の処理の結果が書き込まれておらず、かつ読出し中でない書込み可能なメモリ領域を探索し、書込み可能なメモリ領域が存在しない場合、新たなメモリ領域を確保し、新たなメモリ領域を処理の結果の書き込み先とすることで、先行処理から後続処理へ処理の結果を受け渡す。

明 細 書

発明の名称：電子制御装置

技術分野

[0001] 本発明は、電子制御装置に関する。

背景技術

[0002] 近年、産業用電子制御装置(ECU: Electronic Control Unit)を支える組み込みシステムの開発においては、制御性能の向上、厳格化する規制への対応、装置の自律化に伴う制御演算処理増大等による、搭載アプリケーションの複雑化・高負荷化が進んでいる。これに対応する形で、ECUに搭載されるプロセッサにおいてもマルチコア/メニーコア化によるハードウェアの高性能化が進んでいる。

[0003] 一方で、これら高性能化したハードウェア上に搭載されるソフトウェア開発においては、前述のマルチコア化、メニーコア化に対応したソフトウェア開発が求められ、設計、開発、検証の複雑さが増大する。この原因としてはマルチコアプロセッサの開発において複数のコアが同時に共有資源であるバスやメモリ、キャッシュにアクセスすることから、シングルコアプロセッサ開発と異なりソフトウェア挙動が実行時のプロセッサ状態に応じて容易に変化し得ることが挙げられる。

[0004] また、このような課題を解決して製品化を完了した後も、プロセッサ毎に共有資源に関するパラメタ(バス幅、メモリサイズ、キャッシュサイズ)が異なることから、あるプロセッサ上での開発結果を別プロセッサへと移植することも単純ではなく、同様に設計、開発、検証の工数が増大してしまう。

[0005] このようなプロセッサ構成の複雑化とそれに伴う工数増に対する方策として、車載ソフトウェアアーキテクチャに関する標準規格であるAUTOSARにおいてタイミング設計に関して時刻同期型のタイミング設計方式であるLET(Logical Execution Time)パラダイムが定義されている。同方式は上流工程でメモリアクセスタイミングの決定を中心とするタイミング設計を行い、このタイ

ミングを各処理が実行時に守ることを特徴としており、これにより各処理の開始/終了時刻におけるメモリ状態が一定となることから、プロセッサ状態の組合せによる開発、検証パターン増を防ぐことができ、工数低減に寄与することができる。

[0006] また、異なるプロセッサ間での移植においても、あるプロセッサに対するタイミング設計結果は、別のプロセッサにおいても性能上タイミングを満たすことができる限りにおいて、そのまま移植することが可能となる。

[0007] 一方で、同パラダイムの構成を実現するためには、データコピーが多発する。これは、同パラダイムが複数タスク間でのデータ共有を行う共有メモリ領域と、各処理が実行時に占有するローカルメモリ領域との間において、処理の開始/終了時にデータコピーを行うような構成を取ることによる。このため大規模なデータを処理間でやり取りするような処理においては、従来設計方式に対しデータコピーのための時間オーバーヘッドが大となることから、アプリケーション処理の効率低下が発生する要因となる。

[0008] 先行特許文献1では、このようなマルチコアプロセッサにおけるコア間通信オーバーヘッドを低減する方式として、バッファを用いた解決法が示されている。

先行技術文献

特許文献

[0009] 特許文献1：国際公開第2010/119932号

発明の概要

発明が解決しようとする課題

[0010] 一方で、同方式においては、通信機器を前提とした固定的なパケットサイズを単位とする個数管理に対する実装方式が定義されているが、産業用制御装置においては処理間でのデータ通信毎に、その通信量が異なることが特徴である。このように異なるメモリサイズでの処理に対しても、高速なデータ受渡方式が必要である。また、産業用電子制御装置においては、一定の制御

周期に沿って動作するアクチュエータを駆動する観点から、各処理、もしくは一連の複数処理が複数の異なる周期で動作することが想定されたため、同データ受渡し方式においてはデータ入出力関係のある複数の処理が異なる周期で動作するケースへの対応が求められる。

[0011] 本発明の目的は、時刻同期型設計におけるアプリケーション間におけるデータ受渡しの効率を高めることが可能な電子制御装置を提供することにある。

課題を解決するための手段

[0012] 上記目的を達成するために、本発明は、異なる周期で動作する複数の処理を実行する複数のコアを含むプロセッサと、前記複数の処理を実行する前記複数のコアごとにそれぞれアクセス可能な複数のメモリ領域を含むメモリと、を備え、データ書込みを行う先行処理とデータ読出しを行う後続処理とがそれぞれアクセス可能な前記メモリ領域を前記処理の進捗に応じて変更する電子制御装置であって、前記プロセッサは、最新の前記処理の結果が書き込まれておらず、かつ読出し中でない書込み可能な前記メモリ領域を探索し、書込み可能な前記メモリ領域が存在しない場合、新たな前記メモリ領域を確保し、新たな前記メモリ領域を前記処理の結果の書き込み先とすることで、前記先行処理から前記後続処理へ前記処理の結果を受け渡す。

発明の効果

[0013] 本発明によれば、時刻同期型設計におけるアプリケーション間におけるデータ受渡しの効率を高めることができる。上記した以外の課題、構成及び効果は、以下の実施形態の説明により明らかにされる。

図面の簡単な説明

[0014] [図1]本発明の第1の実施例における、マルチコアプロセッサ1の構成を示す図である。

[図2]本発明の第1の実施例における、時刻同期型のタイミング設計における処理の様子を示す図である。

[図3]本発明の第1の実施例における、共有メモリ上に複数の処理領域を定義することで低時間オーバーヘッドなデータ授受方式を実現する例を示す図であ

る。

[図4]本発明の第1の実施例における、周期の異なる複数の処理A/Bの周期間で共有メモリ上に複数の処理領域を定義することで低時間オーバーヘッドなデータ授受方式を示す図である。

[図5]本発明の第1の実施例における、共有メモリ上のデータ管理方法の例を示す図である。

[図6]本発明の第1の実施例における、時刻 T_1 における処理状態を示す図である。

[図7]本発明の第1の実施例における、時刻 T_1 におけるメモリ管理状態を示す図である。

[図8]本発明の第1の実施例における、時刻 T_2 における処理状態を示す図である。

[図9]本発明の第1の実施例における、時刻 T_2 におけるメモリ管理状態を示す図である。

[図10]本発明の第1の実施例における、データ授受に使うメモリ領域が不足する場合の例を示す図である。

[図11]本発明の第1の実施例における、データ授受に使うメモリ領域が不足する場合におけるメモリ管理表追加方式を示す図である。

[図12]本発明の第1の実施例における、ミドルウェア動作を示すフロー図である。

[図13]本発明の第2の実施例における、共有メモリ上に複数の処理領域を定義することで低時間オーバーヘッドなデータ授受方式を実現する例を示す図である。

[図14]本発明の第2の実施例における、共有メモリ上に複数の処理領域を定義することで低時間オーバーヘッドなデータ授受方式を実現する例の別例として、処理Bの周期が処理Aの周期の2倍以上長い際の例を示す図である。

[図15]本発明の第2の実施例における、共有メモリ上に複数の処理領域を定義することで低時間オーバーヘッドなデータ授受方式を実現する例の別例とし

て、処理Bの周期が処理Aの周期より短い例を示す図である。

[図16]本発明の第3の実施例における、共有メモリ上に複数の処理領域を定義することで低時間オーバーヘッドなデータ授受方式を実現する例を示す図である。

[図17]本発明の第3の実施例における、予備領域の追加および削除によりメモリ効率化を実現する際の例を示す図である。

[図18]本発明の第3の実施例における、時刻 T_3 直前におけるメモリ管理表追加方式を示す図である。

[図19]本発明の第3の実施例における、時刻 T_4 直前におけるメモリ管理表追加方式を示す図である。

[図20]本発明の第3の実施例における、ミドルウェア動作を示すフロー図である。

[図21]本発明の第4の実施例における、実行時間制約違反時の処理挙動を示すフロー図である。

[図22]本発明の第4の実施例における、実行時間制約違反時において処理を継続するための方式を示す図である。

[図23]本発明の第4の実施例における、実行時間制約違反時におけるミドルウェア動作を示すフロー図である。

発明を実施するための形態

[0015] (実施例1)

以下、本発明に係る第1の実施例について図面を用いて説明する。

[0016] 図1に、本発明が対象とする、車両制御向け電子制御装置に搭載されるマルチコアプロセッサ1、および周辺機器の構成を示す。このマルチコアプロセッサ1は複数のCPUコア2を有するマルチコアプロセッサである。このようなマルチコアプロセッサ1においては、各CPUコア2がそれぞれ有するレベル1キャッシュ(L1キャッシュ)21と、複数コアにより共有されるレベル2キャッシュ(L2キャッシュ)22を有する多段キャッシュ構成が主に用いられ、各コア上での命令実行効率を高める工夫が為されている。各CPUコアは内部バス

3を通じてプロセッサ外部にある外部メモリ4や、電子制御装置内外の各種センサ5に接続される。なお、簡単のためマルチコアプロセッサ1内部の詳細構成(各種機構や、詳細なバス構造)は省略する。

[0017] このようなマルチコアプロセッサ1上において、複数のコア上で同時にアプリケーション処理を行う場合、複数のCPUコア2が共有する外部メモリ4上のデータに対するアクセス競合、内部バス3にかかる負荷や複数プロセスからのメモリ利用に伴うL2キャッシュ22のキャッシュヒット率変化により、各アプリケーションの処理効率が変化する。

[0018] ここでの処理効率は、アプリケーション処理完了までに要する時間により測定される性能指標である。このため、事前に規定された各アプリケーションの実行順序だけでなく、各アプリケーションの動作に応じて動的に変化するプロセッサの内部状態変化(L2キャッシュ22状況、内部バス3負荷状況)に応じる形で、アプリケーション処理の状況や順序、メモリアクセスタイミングが変化する。

[0019] さらに、割込み処理など、動作タイミングを設計時に確定させることが難しいアプリケーションも存在することから、マルチコアプロセッサ上におけるアプリケーション設計、開発、デバッグ、および検証は、取り得るプロセッサ状態を鑑みてそれぞれ為される必要があり、これら各作業の複雑化と工数増大が懸念される。

[0020] また、あるマルチコアプロセッサ1上にて設計・検証されたアプリケーションを、別のマルチコアプロセッサに移植する場合は、CPUコア2の動作速度や数量、バス3の動作速度およびバス幅、キャッシュの構成(L1キャッシュ21、L2キャッシュ22、場合によっては更に下位のL3キャッシュなどのキャッシュ階層構造)やそのサイズといったマルチコアプロセッサ1の構成要素変化が発生するため、アプリケーション処理効率やそれに伴う実行順序、メモリアクセスタイミングが変化する。

[0021] そのため、あるプロセッサ1を対象とした設計結果を他のプロセッサ上に移植した際には、必ずしも同等の挙動を示すとは限らず、同様に設計、開発

、デバッグ、検証の工程が必要となってしまう。

[0022] このようなマルチコアプロセッサ1の利用、および他マルチコアプロセッサへの移植を意識した設計方式として、各アプリケーションの処理開始および完了タイミングを規定する時刻同期型の設計方式が知られている。同設計方式について、図2を用いて説明する。

[0023] 同設計方式においては、各アプリケーション毎に定められた周期の最初に、全ての処理からアクセス可能な共有メモリ領域から、自身のみがアクセス可能なローカルメモリ領域に演算に必要な値を読み込み(図中Copy-inと記載した部分)、処理区間の最後に共有メモリに演算結果を書き出す(図中Copy-out)ように構成され、演算はこの読み込みから書き出しまでの間の任意のタイミングで実行される。

[0024] 同設計方式では、このようにローカルメモリ空間上でアプリケーション処理を実行することにより他アプリケーション処理とのメモリ競合を避けると同時に、処理区間の最後における共有メモリへの書き出し順序を設計時に定義することで、CPUコアやバスの動作速度、キャッシュサイズ、コア数変更に伴うアプリケーション実行コア変化などの挙動変化要因に対しても、設計時に定められた処理時間内に処理が完了する限りにおいて、プロセッサ状態に依らずに共有メモリ状態を一致させられることが期待される。

[0025] これにより、プロセッサ状態に起因するアプリケーション処理の挙動変化を抑制すると同時に、また異なるプロセッサ間における高い移植性を得ることができる方式となる。

[0026] このような方式を実現する方法の1つとして、前述のようなデータコピーを用いた共有メモリ、ローカルメモリ構成が知られているが、同データコピー方式では特にアプリケーション処理が利用するメモリサイズが大となる場合において、データコピーに要する時間オーバーヘッドが大きくなる課題がある。

[0027] そこで、図3に示すような2つのバッファを用いた時刻同期型タイミング設計実現手法が知られている。なお、同図における処理区間と周期は、それ

ぞれ時刻同期型タイミング設計方式における処理開始(共有メモリ領域からローカルメモリ領域へデータを読み出し)から処理完了(演算完了時にデータを逆方向に書込み)までの時間の幅、およびこのような処理区間を繰り返す際の単位を示すものである。

[0028] また図面記載の都合からSwap処理の後に処理Aおよび処理Bがアクセスするメモリ領域が指し示されるまでに時間を要す記載となっているが、実際にはSwap処理の終了時点において各処理がアクセスするメモリ領域は決定されている。

[0029] 同手法においては、データ依存関係を有する2つの処理A、B(A、Bが互いに前周期の相手の演算結果を用いて、次周期に演算を行う)に対し、共有メモリ上に処理Aが書込みアクセスを行うメモリ領域と処理Bが読み出しアクセスを行うメモリ領域との2つのメモリ領域から構成される。ここで、 A_i 、 B_i はそれぞれ処理A、Bの*i*周期目の処理の処理を表す記号とする。また、上記のように記載される各処理中においては、A、B以外の図示しない処理によりプリエンプトされ、再開していてもよい。

[0030] このような構成をとると、*i*番目の処理区間の開始時に、処理 A_i および B_i はそれぞれ自身が直前に用いたバッファ領域と反対側のバッファ領域を指すことで、処理 A_i は B_{i-1} の結果である $Res(B_{i-1})$ を用いて自身の計算を行い、処理 A_i の演算結果である $Res(A_{i-1})$ を同領域に書込む動作を行うことができる。Bも同様に、処理Aの前周期における処理結果 $Res(A_{i-1})$ を読み込み、自身の演算結果 $Res(B_i)$ を書込むことで、演算処理が進んでいく。なお、本図において初回に実行される処理 A_1 および B_1 は、データの入力なしで演算を開始するものとする。

[0031] ここで、本動作における各メモリ領域の状態に着目すると、*i*番目の周期に処理Aにより書込まれた演算結果を、次の*i*+1番目の周期にBが、*i*番目の周期に処理Bにより書込まれた演算結果を、次の*i*+1番目の周期にAが、それぞれ参照して演算を継続するため、データをコピーすることなく演算を継続することが可能となる。

[0032] また、このメモリ領域は、データ依存かつ順序依存関係を有する2つの処

理A、Bの間でのみ保有され、かつ常に互いに異なるバッファ領域を有することから、処理A、Bの処理区間中はそれぞれの処理のみがアクセス可能なローカルメモリ領域と見做すことができる。

[0033] 以上から本構成を用いることで、時刻同期型タイミング設計方式における演算構造を実現しつつ、データコピーに要する時間オーバーヘッドを削減することが可能となる。

[0034] 一方で、処理Aと処理Bの周期は必ずしも同一であるとは限らない。例えば自動運転処理を想定すると、自転車周辺状況をマッピングする処理が存在するが、同処理の入力は例えば50ms毎に動作するレーザレンジセンサや数ms毎に動作する自転車速度センサに連動して処理が行われ、マッピング結果が更新される。

[0035] また、パワートレイン分野に目を向けると、例えばエンジン処理のようなケースにおいてはエンジン回転数に同期した処理が発生するため、動作中に制御周期が変化するケースもある。このように異なる周期で動作する処理間でデータ授受が発生する場合は、前述のようにメモリ領域を固定し、同時に更新するような形での方式実現は困難となる。

[0036] 特に制御周期変化に対応するためには、制御装置の状態に合わせた動的なメモリ領域の管理が必要となる。図4に、処理Aと処理Bの周期が異なる場合の、本動作の例を示す。なお、図面記載の簡単のため、本例では処理区間と周期とが一致する場合について、スワップタイミング記載を省略して説明するが、両者が一致しない場合においても同様の方式により演算処理が実現される。また処理A、Bの周期によっても、メモリアクセスのタイミングや必要となるバッファ領域数が異なる点に注意が必要である。

[0037] 同図においては、破線で示したAの周期(本例では処理区間に等しい)と、一点鎖線で示したBの周期とが異なっている例を示しており、A、Bは互いに極力相手の最新処理結果を用いて処理を進める例を示す。すなわち、処理A、Bが互いに、自身の周期が開始される以前に終了している相手の処理を検出し、当該メモリ領域を読み出す構成を取る。

[0038] 本例では、処理Aの周期が処理Bよりも長い例を示しているが、これにより、処理AおよびBの開始タイミングによっては、ある処理 A_i の結果 $\text{Res}(A_i)$ が、2つ以上の複数の処理 $\text{Res}(B_j)$ 、 $\text{Res}(B_k)$ 、…の演算過程において読み出されるケースが発生する。同図においては、処理Aの周期は処理Bの周期の2倍以下となっていることから、ある処理 A_i の結果 $\text{Res}(A_i)$ は、最大2つの処理Bの演算に用いられることとなる。

[0039] 上記の例では、ある処理 A_i の結果 $\text{Res}(A_i)$ が書き込まれたメモリ領域は、処理 B_j が演算結果 $\text{Res}(B_j)$ を演算する過程で破壊されてしまうことから、処理 B_k が演算結果 $\text{Res}(B_j)$ を演算するメモリ領域を別途生成する必要があるため、本実施例では処理 A_i の演算終了後に、新たに別のメモリ領域に演算結果をコピーする必要がある。演算結果をコピーする箇所数は、処理Aと処理Bの周期により定まる。

[0040] この領域数は、
(処理Aの周期)/(処理Bの周期)
の小数点以下を切り上げた数を最小値とする値が取られることとなる。

[0041] 図4に示したコピー処理 $\text{Cp}(\text{Res}(A_i))$ を行うことで、例えば図中処理 A_2 の結果 $\text{Res}(A_2)$ は、処理 B_4 および処理 B_5 に用いることが可能となる。なお、本図では処理 A_2 の演算結果を自身の保有するメモリ領域2に書込む際、同時に別メモリ領域4へ同時に書込む例を想定した。このため、 $\text{Res}(A_2)$ の領域2への書込みが開始される処理 A_2 の後半部分からコピー処理 $\text{Cp}(\text{Res}(A_2))$ が動作開始する例を示しているが、例えばメモリ領域2を処理 A_2 が確保すると同時に、コピー先のメモリ領域4も確保する構成を取っても構わない。

[0042] このようなメモリ管理構成をとることで、先述の同じ周期の処理A、B間において2つのメモリ領域を用いたデータ授受を行った場合と同様、処理Aと処理B間でのデータコピーオーバーヘッドは削減される。これは、またデータコピー処理 $\text{Cp}(\text{Res}(A_i))$ も処理 A_i 実行中に行われることから、例えば処理 A_2 の後続処理 B_5 の開始時には、既にデータがメモリ領域上に備わっているためである。

[0043] 以上のメモリ管理構成を実現する手段として、専用のミドルウェアを定義

する方法が考えられる。ミドルウェアを定義することで、アプリケーション開発者は細かなメモリ管理を意識せずアプリケーションの挙動に集中することができ、また産業用制御装置におけるOSやハードウェアといった変化要素からもアプリケーションを解放することができ、時刻同期型タイミング設計の目的であった異なるハードウェア(プロセッサ)間での移植性向上と最も整合する。

[0044] 同方式を、ミドルウェアを用いて実現する場合の実装方法の例としては、ミドルウェアが当該メモリ領域を管理し、アプリケーション側にはメモリ領域を示すポインタとして渡す方法などが想定される。本実施例においては、この方式に基づき説明する。

[0045] 上記のミドルウェアには、各処理における処理区間毎に更新される共有メモリ上のデータを管理し各処理の起動時に適切なメモリ領域を通知すること、および各処理間でデータ授受に使うメモリ領域が不足した際に共有メモリ上に新たなメモリ領域を作成することでデータ授受を維持すること、の2つの機能が求められる。それぞれについて、以下説明する。

[0046] まず、共有メモリ上のデータ管理方法の例を、図5に示す。

[0047] 本方式における必要事項は、ある処理が演算中のメモリ領域に、他の処理がアクセスすることを防ぐこと、および各処理が開始される際に適切な入力情報が書込まれているメモリ領域を渡すこと、の2点である。ここでの適切な入力情報とは、当該処理の開始時刻までに完了している、最新の相手処理の演算結果である。

[0048] 前者は例えばメモリ上のデータ管理にはメモリ領域の占有状況として、後者はメモリ領域に格納された最新演算結果の位置として、それぞれ管理される。本例では、複数の処理が相互にデータを授受する構成であるため、最新演算結果は双方向に定義される。

[0049] 図6および図7を用いて、本実施例における処理A₂開始時(時刻T₁)におけるミドルウェア挙動を説明する。時刻T₁においては、メモリ領域3が処理B₂により占有されている。また、処理A₁および処理B₁が、当該時刻の開始時刻までに

完了した最新の処理となるため、処理A₁に関してはメモリ領域1および4が、処理B₁に関してはメモリ領域2が、それぞれ最新処理結果であることが示される。本例では前述のように、処理Aの周期が処理Bより長いことから、処理Aは複数個所に演算結果を保存している。

[0050] 時刻T₁にて起動される処理A₂に対し、ミドルウェアは図7の管理テーブルにおける最新処理結果の列のうち、処理Bから処理Aの項目を検索する。結果、領域2に最新処理結果(処理B₁の演算結果)が記載されていることから、当該領域が処理A₂の利用すべき領域と分かる。そこで、管理テーブルにおける処理Aによる占有状況欄のメモリ領域2部分にチェック(図7の点線の丸)を入れ、当該メモリ領域のアドレスを処理Aに渡すことで処理A₂を起動する。

[0051] 次に、時刻T₂にて処理B₂が終了した際の挙動を図8および図9を用いて説明する。図8に示すように、同時刻において処理A₂は継続していること、および当該時刻での最新演算結果が処理B₂の結果となることから、処理B₂が占有していたメモリ領域3が最新の演算結果を保持している領域と分かる。

[0052] そこで、処理Bによる占有状況テーブルからメモリ領域3のチェックを外し、代わりに処理B→処理Aへの最新値テーブルの領域3の箇所にチェックを入れる。これにより、メモリ管理テーブルが図9のように更新される。このように管理テーブルが更新されることで、時刻T₂直後に別の処理Aが起動された場合には、メモリ領域3を用いることで処理Bの最新処理結果を用いた演算を行うことが可能となる。

[0053] 次に、各処理間でデータ授受に使うメモリ領域が不足した際に共有メモリ上に新たなメモリ領域を作成することでデータ授受を維持する方法について、図10および図11を用いて説明する。

[0054] 図10に示した時刻T₀は、処理Aにおいて演算結果の書込みが開始されたタイミングである。前述のように、処理Aは処理Bに対して長い周期と定義されており、処理Aから処理Bへのデータ授受については複数のメモリ領域を備える必要がある。そこで、処理Aは複数のメモリ領域を確保する必要があるが、同タイミングにおいて確保されているメモリ領域は領域1のみであるため、

ミドルウェアを通じて新たな書込み先を確保する必要がある。

[0055] そこで、ミドルウェアは図 1 1 の管理テーブルを参照するが、確保されているメモリ領域 1 およびメモリ領域 3 は処理A₁、処理B₂にそれぞれ占有されており、またメモリ領域 2 には処理Bの最新処理結果が格納されていることから、処理A₁が新たに確保することができるメモリ領域が存在しない。そこで、新たに共有メモリの一部(処理A₁の書込みサイズに相当)を処理A-処理B間のデータ授受用メモリ領域として確保および管理テーブル上の占有状態をAと更新し、メモリ領域 4 を作成する。このメモリ領域 4 を処理Aに渡すことで、処理A₁の終了時には処理Aの最新値が両メモリ領域に書込まれた状態となる。

[0056] 以上の構成によりミドルウェアを用いた上述のメモリ管理方式が提供され、先述の同じ周期の処理A、B間において2つのメモリ領域を用いたデータ授受を行った場合と同様、処理Aと処理B間でのデータコピーオーバーヘッドは削減される。

[0057] また、産業用電子制御装置においては、一定の制御周期に沿って動作するアクチュエータを駆動する観点から、各処理、もしくは一連の複数処理に対する実行時間制約(リアルタイム制約)が存在する。これに違反した場合、アクチュエータ制御指令値が出力されない、もしくはセンサ入力値に対して適切でない指令値が出力される可能性があるため、アクチュエータに対応した未完了時処理が必要となる。例えば、冗長系構成を用いた別システムを利用した制御継続方式や、縮退制御方式などが挙げられる。

[0058] 本構成においては、アプリケーション処理の終了予定時刻が設計時に規定されていることから、このタイミングまでに処理が完了しないケースに対して前述の未完了処理が実行される。未完了処理実施の要否をミドルウェアが判別するためには、例えばミドルウェアが処理を起動する際にフラグをセットし、処理の完了時にこのフラグを解除することで、処理終了時刻においてミドルウェアがチェックするような実装が考えられる。

[0059] 以上の構成を実現するミドルウェア挙動について、図 1 2 のフローチャートを用いて説明する。ミドルウェアは前述のように各処理の起動時刻に合わ

せる形で動作するため、例えば処理開始時刻のタイマ割込みにより起動される。換言すれば、複数の処理のそれぞれの開始時刻においてタイマ割込みが入る。その後、管理テーブルから最新処理結果格納先を探索し(S100)、当該領域を確保するため管理テーブルの占有状況箇所を更新する(S101)。

[0060] そして、当該メモリ領域を処理に渡して処理を起動することで、アプリケーション処理が開始される(S102)。ここで、アプリケーション処理の周期の長短により、複数のメモリ領域へデータを書込む必要があり、かつそのような書込み先領域が確保されていない場合には新たなメモリ領域を確保し(S103)、管理表にアドレスを追加(S104)、当該書込み先の状態を占有中に変更した上で(S105)、処理に渡す(S106)。

[0061] 書込み先領域がミドルウェアにより既に確保され管理テーブルに記載されている場合は、S103およびS104をスキップし、S105が実施される。そして、アプリケーション処理終了時刻には開始時と同様にタイマ割込みが入る。換言すれば、複数の処理のそれぞれの終了時刻においてタイマ割込みが入る。アプリケーション処理終了時においては、メモリ管理テーブルの最新値状態を更新し(S107)、メモリ占有状態を解除する(S108)。以上により、1つのアプリケーション処理に対するミドルウェア処理が完結する。

[0062] 換言すれば、本実施例の電子制御装置は、異なる周期で動作する複数の処理を実行する複数のコア（CPUコア2）を含むプロセッサ（マルチコアプロセッサ1）と、複数の処理を実行する前記複数のコア（CPUコア2）ごとにそれぞれアクセス可能な複数のメモリ領域を含むメモリ（共有メモリ）と、を備え、データ書込みを行う先行処理とデータ読出しを行う後続処理とがそれぞれアクセス可能な前記メモリ領域を処理の進捗に応じて変更する。プロセッサ（マルチコアプロセッサ1）は、最新の処理の結果が書き込まれておらず、かつ読出し中でない書込み可能なメモリ領域を探索し、書込み可能なメモリ領域が存在しない場合、新たなメモリ領域を確保し、新たなメモリ領域を処理の結果の書き込み先とすることで、先行処理から後続処理へ処理の結果を受け渡す。

[0063] 図5に示すように、電子制御装置は、メモリ領域ごとの占有状況と処理の結果の最新値の有無を格納するテーブルを備える。プロセッサは、このテーブルのすべてのメモリ領域について、占有状況が専有を示し、または最新値の有無が最新値を格納していることを示す場合、書込み可能な前記メモリ領域が存在しないと判定する。

[0064] 以上の構成により、マルチコアCPUにおける設計、開発、検証工数を低減する時刻同期型タイミング設計方式を低時間オーバーヘッドに実現し、かつ異なるハードウェア(プロセッサ)間での移植性向上を実現する方式が提供される。

[0065] (実施例2)

以下、本発明に係る第2の実施例について図面を用いて説明する。想定するCPU構成(図1)や適用する時刻同期型タイミング設計方式(図2)は実施例1に準拠するが、本実施例では複数の処理Aおよび処理Bの間に一方向のデータ依存関係があるケース、すなわち処理Aの実行結果を利用して処理Bが実行されるが、逆は存在しないケースにおける本発明の適用事例を、主に実施例1との差分に着目して説明する。

[0066] 本実施例においては、処理B実行に必要な処理Aの演算結果のみを共有メモリ上に展開することで、例えば処理Aへの入力データといった処理B実行に必要なデータを共有メモリ上に複数保持する必要がなく、処理間のメモリ利用効率の向上および、これに伴う制御装置内におけるメモリ利用効率向上、最終的には制御装置に搭載するメモリサイズ低減による低コスト化が期待される。

[0067] 図13に、本発明が例題とする処理構成を示す。本実施例では、メモリ領域は3つの異なるメモリ領域(領域1～3)から構成される。前述のように、本実施例においては、処理Aの演算結果を処理Bが読込む構成となっていることから、図中の共有メモリ領域記載においては、処理Aは自身の演算結果の書込みのみ、処理Bは処理Aの演算結果の読込みのみ、をそれぞれ記載している。

- [0068] 処理Aが自身の演算に用いる読込データや、処理Bが自身の演算結果を書込む領域については、図中の共有メモリ領域に書込む構成をとることも、それ以外のデータ依存関係を有する処理との間で同様の共有メモリ構成をとることも、可能である。
- [0069] このような共有メモリの利用方式においては、処理A、Bの2つの処理間でデータ共有する場合については、処理Aの書込み先が2箇所(処理Aの周期が処理Bの周期より短い場合)と、処理Bの読出し先が1ヶ所の合計3箇所以上あれば、実施例1に記載のデータ受渡し(1つの処理が書込みもしくは読出しのいずれかのみを担当する)が可能となる。
- [0070] なお、本実施例のような構成については、図14に示したように、処理Aの周期が処理Bの周期に対して更に小さくなる場合についても、同様のメモリ構成をとることが可能である。このように、処理Bの周期が、処理Aの周期の2倍以上となったケースについては、実施例1に示したメモリ管理表において、最新値ではないメモリ領域に処理結果が上書きされることで、共有メモリ領域の更なる追加が不要となる。
- [0071] 例えば、処理A₁の演算結果Res(A₁)が処理Bにより読出される前に処理A₂が演算完了するため、処理B₂には処理A₂の演算結果Res(A₂)が読出されればよい。そのため、同図に示したように、処理A₃の演算結果Res(A₃)は、メモリ領域2にある処理A₁の演算結果Res(A₁)を上書きすることが許容される。
- [0072] 逆に、処理Aの周期が処理Bの周期より長いケースにおいても、処理Bが自身が読出すメモリ領域に自身の演算結果を上書きしない限りにおいて、図15に示すように同様の構成をとることが可能となる。
- [0073] 以上の構成、すなわちデータ受け渡しが発生する2つの処理間において、1つの処理が書込みのみ、もう1つの処理は読出しのみ、を担当する場合については、本実施例のように3つの共有メモリ領域のみを用いることで、実施例1の事例に比べ、必要メモリ量の削減およびミドルウェア(もしくは、実装形態によりOS、各アプリケーション)におけるメモリ領域の追加判定処理が不要となり、実装の効率化が図られる。

[0074] (実施例3)

以下、本発明に係る第3の実施例について図面を用いて説明する。

[0075] 図16に、本発明が例題とする処理構成を示す。本例においては、実施例2と同様に処理Aは書込みのみを、処理Bは読出しのみを行う構成ではあるが、Aの処理区間とAの周期とが不一致な場合を例として説明する。時刻同期型のタイミング設計方式においては、前述のように必ずしも処理区間と周期とが一致する必要はないため、このような構成をとることが可能となる。

[0076] このようなケースにおいては、同図に示したように、処理A₃の演算結果Res(A₃)が読みだされる前に処理A₄が完了し、処理B₄には処理A₄の演算結果Res(A₄)を用いて演算を開始することができる。この場合、実施例1および2とは異なり、共有メモリ領域の利用率が低下し、メモリ効率の悪化が発生する。

[0077] そこで、共有メモリ上には(同一周期の場合と同様に)最低限必要な2つのメモリ領域を構成し、3つめのメモリ領域は必要に応じて共有メモリから動的に確保し、不要と判別された場合には解放することで、制御装置全体の処理構成の間で予備領域を融通し、メモリ効率を更に向上することが可能となる。

[0078] 換言すれば、本実施例では、メモリ領域は2つの異なるメモリ領域から構成され、プロセッサ(マルチコアプロセッサ1)は、書込み可能なメモリ領域が存在しない場合、新たなメモリ領域を確保する。その後、例えば、メモリ領域として3つ以上のメモリアドレスを有している条件において、プロセッサ(マルチコアプロセッサ1)は、1つのメモリアドレスが最新の処理の結果の書込み先ではなく、かつ読出し中でもない場合において、前記1つのメモリアドレスを解放する。

[0079] 以上の構成を、ミドルウェアによりメモリ管理により実現する方式を示す。本実施例では、メモリ領域の管理処理は、プロセッサ(マルチコアプロセッサ1)が専用のミドルウェアを実行することにより実現される。なお、実施例1と同様、本実施例で説明するミドルウェア処理はOSや各処理内部にて実施されても良い。

[0080] 予備領域の確保については、実施例 1 に説明した構成と同様のもので実現可能であるため、本実施例においては予備領域の解放処理について説明する。

[0081] 前述のように、予備領域が不要となるケースについて図 17 から図 19 を用いて説明する。図 17 における時刻 T_3 直前(処理 A_4 の処理区間終了直前)においては、図 18 のようにメモリ領域 1 は処理 A_4 がメモリ領域 2 を占有、また処理 A_3 の結果が最新値となっている。

[0082] 時刻 T_3 直後のメモリ状態は図 19 のようになっており、メモリ領域 2 がB占有中かつ最新処理結果が書き込まれており、予備領域は空欄となる。このように処理 A_4 の完了に伴い、予備領域が占有されておらず、かつ最新値が書き込まれていないケースにおいて、予備領域の解放が可能となる。処理Aの周期が処理Bより長い場合においては、最新値が書き込まれておらず、かつ処理Aが予備領域を占有する必要がない場合に、同領域が解放可能となる。

[0083] 以上のことから、ミドルウェア処理では、処理の開始時にメモリ解放可能かを判別する構成をとることで、全てのケースに対して適用が可能となる(処理Aの周期が処理Bより短い場合は、次の処理A起動時まではオーバーヘッドが継続する)。

[0084] 図 20 に、このようなミドルウェア挙動フローを示す。基本的な構成は実施例 1 にと同様だが、処理Aおよび処理Bは書込みもしくは読出し処理のいずれかを行うため、この判別機構によりフローが少々異なっている。そのため、処理起動時に、読出し処理か書込み処理化の条件分岐が入る。以降は実施例 1 のミドルウェア処理を踏襲する。本例において追加となるのは、書込み処理側において書込み可能領域が存在した場合である。この際に、管理メモリ領域数が3つ以上であり、かつ、予備領域が最新値を保持しておらず読出し中でもない場合に、解放可能となり、メモリ領域を管理表から削除する(S109)。

[0085] なお、本例では処理間でのデータ受渡し用に用いる固定的な2つのメモリ領域と、必要に応じて動的に確保/解放する1つの予備領域を用いる例につい

て説明したが、2つのメモリ領域が常に確保されるのであれば、前述の固定的なメモリ領域としての形態は不要である。

[0086] すなわち、常に管理テーブル上に確保される2つのメモリ領域と、さらに必要に応じて動的に確保/解放される1つのメモリ領域、の構成でも、本実施例は実現可能である。この場合には、管理メモリ領域の解放条件は、管理アドレスが3つ以上であり、かつ、最新値を保持せず読出し中でもない、場合において、管理表からアドレス削除を行うことが可能となる。この場合には、S109処理が、当該メモリ領域を探索し削除、と変化する。

[0087] (実施例4)

以下、本発明に係る第4の実施例について図面を用いて説明する。

[0088] 産業系電子制御装置はセンサ情報の入力に基づき制御指令値を演算しアクチュエータを制御する構成が取られる。このような構成においては、周期的に動作するアクチュエータを正しく制御するため、一般的にセンサ入力から制御指令値出力までの間に実行時間制約が存在する。

[0089] 一方で、マルチコア/メニーコアCPUにおける設計においては、前述の複数コア間におけるソフトウェア挙動は実行時のプロセッサ状態に応じて容易に変化し得ることから、例えば実行時間の変動が発生する。このような実行時間変動に対して、設計時におけるマージン設計などによる対処が取られるが、実行時間制約の厳しさや各種資源の競合により必ずしも処理完了が保証されるとは限らない。そのため、ミドルウェアに処理完了の確認と、未完了時の対処が必要となる。

[0090] 図21に、処理Aの図中2回目の実行である処理A₂が、実行時間変動の影響により規定の処理区間内に処理を完了できなかった例を示す。

[0091] 本時刻同期型タイミング設計方式における、処理未完了時の対処としては以下の2つの方式が想定される。1つは、処理A₂を強制的に中断もしくは実行終了させるケースである。対処法としては上位システムへ通知を行い、判断を仰ぐケースや、設計時にこのような違反発生ケースに対する処理が規定され、そちらに移行するケースも考えられるが、制御目的や実行時間制約違反

時の影響範囲が小さいケースにおいては処理を継続するケースも想定される。

[0092] このような場合、ミドルウェアでは処理 A_2 を中止し、かつメモリ管理テーブルを更新しないことで、処理 A_2 の演算結果 $\text{Res}(A_2)$ を用いるはずであった処理 B_4 および処理 B_5 が、演算未完了な値を用いないように制御する。これら処理は、処理 A_1 の演算結果 $\text{Res}(A_1)$ を用いて制御を継続する。

[0093] 換言すれば、ミドルウェアを実行するプロセッサ（マルチコアプロセッサ 1）は、複数の処理のそれぞれの終了時刻において、処理未完了を検出した場合に強制的に当該処理を停止させる未完了処理を行う。

[0094] もう1つの方式は、このようなケースにおいて、処理Aに対する処理Bのような、先行処理の結果を用いて処理を行う後続処理の設計時マージンを用いて処理Aの実行時間を延長することで、後続処理Bに最新の処理結果を渡す方式である。

[0095] 換言すれば、ミドルウェアを実行するプロセッサ（マルチコアプロセッサ 1）は、複数の処理のそれぞれの終了時刻において、処理未完了を検出した場合において、当該処理の演算結果を用いて演算を行う後続処理の開始時刻を、設計時に規定される猶予時間分だけ遅らせることにより、後続処理において最新の演算処理結果を用いて演算を継続する。

[0096] 図 2 1 において、処理 A_2 の後続処理である処理 B_4 、およびコア数によっては処理 A_2 の遅れの影響が自身の処理時間削減に繋がる処理 A_3 の両者は、共にマージン時間、すなわち処理区間と各処理の実行時間との差分が存在する。この時間は通常、前述のようなマルチコア上における実行時間変動を吸収するために用いられるものであるが、設計時における最悪実行時間が常に発生するわけではないことから、同マージン時間を利用することは可能である。

[0097] 本実施例においては、例えばマージン時間を最悪実行時間+20%と設計し、そのうち10%をこのような先行処理の遅れに対し適用可能であるとする。このような時間を、遅延吸収時間（図 2 2）と呼ぶと、このような条件において、処理 A_2 に遅延が発生した場合、影響を受ける処理（処理 B_4 、および例えば2コ

アで本処理を行う場合には処理A₃も含む)のうち、直近の遅延吸収時間終了時刻(同図における時刻T₄)を、ミドルウェアがタスク情報などから取得し、同時刻まで処理A₂を継続する。同時刻において処理A₂が完了した場合には実施例3に記載のメモリ管理と同様の終了処理を実施し、同時刻においても処理A₂が完了しない場合には、前述の1つめの方式と同様の強制終了処理を取る。

[0098] 図23に、本実施例におけるミドルウェア挙動に関するフロー図を示す。上述の未完了時処理(S110)を追加することで、産業用制御装置に求められる実行時間制約に関し、特に制約違反の検出および対処方式が提供され、より幅広い産業用制御装置への本方式適用が実現される。

[0099] 以上の構成により、産業用制御装置に求められる実行時間制約に対し、本発明が準拠することが出来る。また上記に示した2つめの対処法においては、実行時間制約に違反した場合においても、処理実行上のマージン時間を用いて対処可能となる場合において処理継続もしくは最新の処理結果を用いた処理継続を可能とすることで、産業用制御装置による制御演算継続およびその精度向上が実現される。

[0100] 以上説明したように、第1～第4の実施例によれば、時刻同期型設計におけるアプリケーション間におけるデータ受渡の効率を高めることができる。

[0101] なお、本発明は上記した実施例に限定されるものではなく、様々な変形例が含まれる。例えば、上述した実施例は本発明を分かりやすく説明するために詳細に説明したものであり、必ずしも説明した全ての構成を備えるものに限定されるものではない。また、ある実施例の構成の一部を他の実施例の構成に置き換えることが可能であり、また、ある実施例の構成に他の実施例の構成を加えることも可能である。また、各実施例の構成の一部について、他の構成の追加・削除・置換をすることが可能である。

[0102] また、上記の各構成、機能等は、それらの一部又は全部を、例えば集積回路で設計する等によりハードウェアで実現してもよい。また、上記の各構成、機能等は、プロセッサがそれぞれの機能を実現するプログラムを解釈し、実行することによりソフトウェアで実現してもよい。各機能を実現するプロ

グラム、テーブル、ファイル等の情報は、メモリや、ハードディスク、SSD (Solid State Drive) 等の記録装置、または、ICカード、SDカード、DVD等の記録媒体に置くことができる。

[0103] なお、本発明の実施例は、以下の態様であってもよい。

[0104] 複数のコアを含むプロセッサを有する電子制御装置であって、前記プロセッサは、異なる周期で動作する複数の処理を実行し、前記複数の処理を実行する前記複数のコアごとにそれぞれアクセス可能な複数のメモリ領域を備え、データ書込みを行う先行処理と、データ読出しを行う後続処理とがそれぞれアクセス可能な前記メモリ領域を、前記処理の進捗に応じて変更する電子制御装置において、最新の前記処理の結果が書き込まれておらず、かつ読出し中でない書込み可能な前記メモリ領域を探索し、書込み可能な前記メモリ領域が存在しない場合、新たな前記メモリ領域を確保し、新たな前記メモリ領域を前記処理の結果の書き込み先とすることで、前記先行処理から前記後続処理へ前記処理の結果を受け渡すこと、を特徴とする電子制御装置。

[0105] これにより、産業用制御装置に求められる時刻同期型設計におけるアプリケーション間におけるデータ受渡の効率を高めることが可能である。

符号の説明

[0106] 1 : マルチコアプロセッサ

2 : CPUコア

2 1 : L1キャッシュ

2 2 : L2キャッシュ

3 : 内部バス

4 : 外部メモリ

5 : センサ

請求の範囲

- [請求項1] 異なる周期で動作する複数の処理を実行する複数のコアを含むプロセッサと、
- 前記複数の処理を実行する前記複数のコアごとにそれぞれアクセス可能な複数のメモリ領域を含むメモリと、を備え、
- データ書込みを行う先行処理とデータ読出しを行う後続処理とがそれぞれアクセス可能な前記メモリ領域を前記処理の進捗に応じて変更する電子制御装置であって、
- 前記プロセッサは、
- 最新の前記処理の結果が書き込まれておらず、かつ読出し中でない書込み可能な前記メモリ領域を探索し、書込み可能な前記メモリ領域が存在しない場合、新たな前記メモリ領域を確保し、新たな前記メモリ領域を前記処理の結果の書き込み先とすることで、前記先行処理から前記後続処理へ前記処理の結果を受け渡すこと、を特徴とする電子制御装置。
- [請求項2] 請求項1に記載の電子制御装置であって、
- 前記複数の処理のそれぞれの開始時刻および終了時刻においてタイム割込みが入ること、を特徴とする電子制御装置。
- [請求項3] 請求項2に記載の電子演算装置であって、
- 前記メモリ領域は3つの異なるメモリ領域から構成されること、を特徴とする電子制御装置。
- [請求項4] 請求項2に記載の電子制御装置であって、
- 前記メモリ領域は2つの異なるメモリ領域から構成され、
- 前記プロセッサは、
- 書込み可能な前記メモリ領域が存在しない場合、新たな前記メモリ領域を確保すること、を特徴とする電子制御装置。
- [請求項5] 請求項4に記載の電子制御装置であって、
- 前記メモリ領域として3つ以上のメモリアドレスを有している条件

において、

前記プロセッサは、

1つのメモリアドレスが最新の前記処理の結果の書込み先ではなく、かつ読出し中でもない場合において、前記1つのメモリアドレスを解放すること、を特徴とする電子制御装置。

[請求項6]

請求項1に記載の電子制御装置であって、

前記メモリ領域の管理処理は、

前記プロセッサが専用のミドルウェアを実行することにより実現されること、を特徴とする電子制御装置。

[請求項7]

請求項6に記載の電子制御装置であって、

前記ミドルウェアを実行する前記プロセッサは、

前記複数の処理のそれぞれの終了時刻において、処理未完了を検出した場合に強制的に当該処理を停止させる未完了処理を行うこと、を特徴とする電子制御装置。

[請求項8]

請求項6に記載の電子制御装置であって、

前記ミドルウェアを実行する前記プロセッサは、

前記複数の処理のそれぞれの終了時刻において、処理未完了を検出した場合において、当該処理の演算結果を用いて演算を行う後続処理の開始時刻を、設計時に規定される猶予時間分だけ遅らせることにより、後続処理において最新の演算処理結果を用いて演算を継続すること、を特徴とする電子制御装置。

[請求項9]

請求項1に記載の電子制御装置であって、

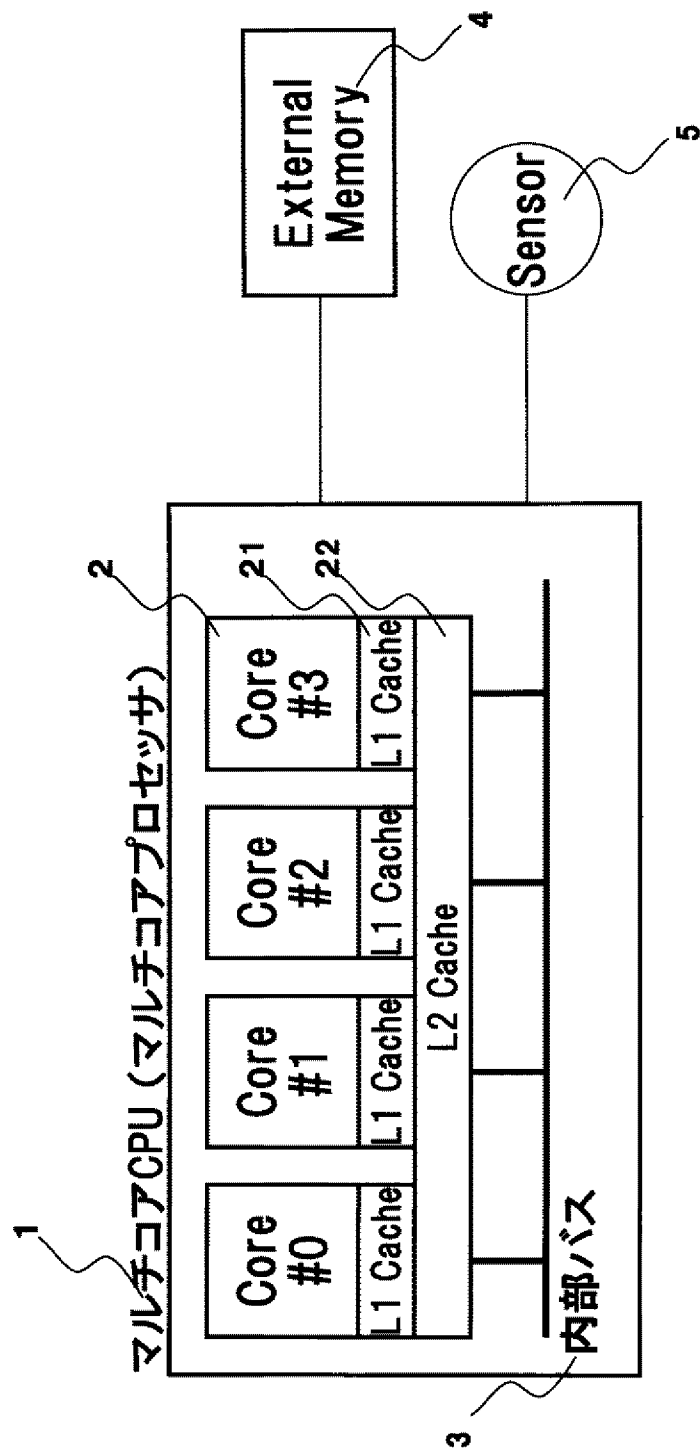
前記メモリ領域ごとの占有状況と前記処理の結果の最新値の有無を格納するテーブルを備え、

前記プロセッサは、

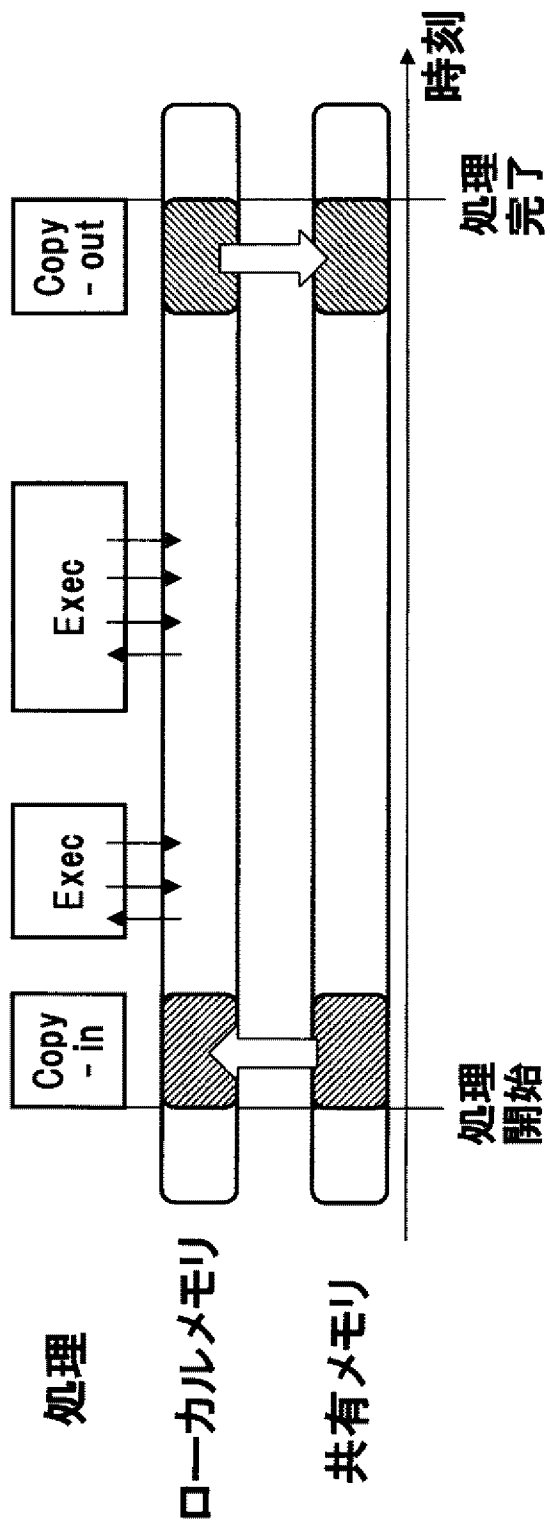
前記テーブルのすべての前記メモリ領域について、前記占有状況が専有を示し、または前記最新値の有無が最新値を格納していることを示す場合、書込み可能な前記メモリ領域が存在しないと判定すること

、を特徴とする電子制御装置。

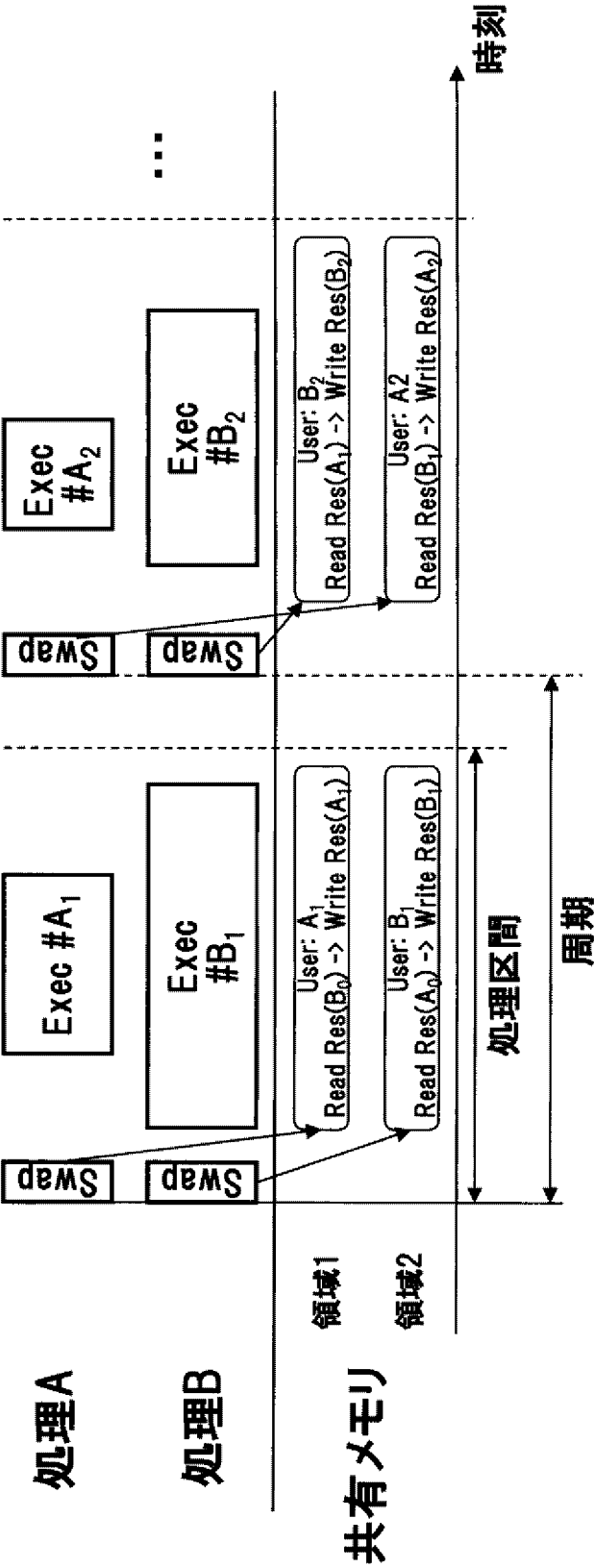
[図1]



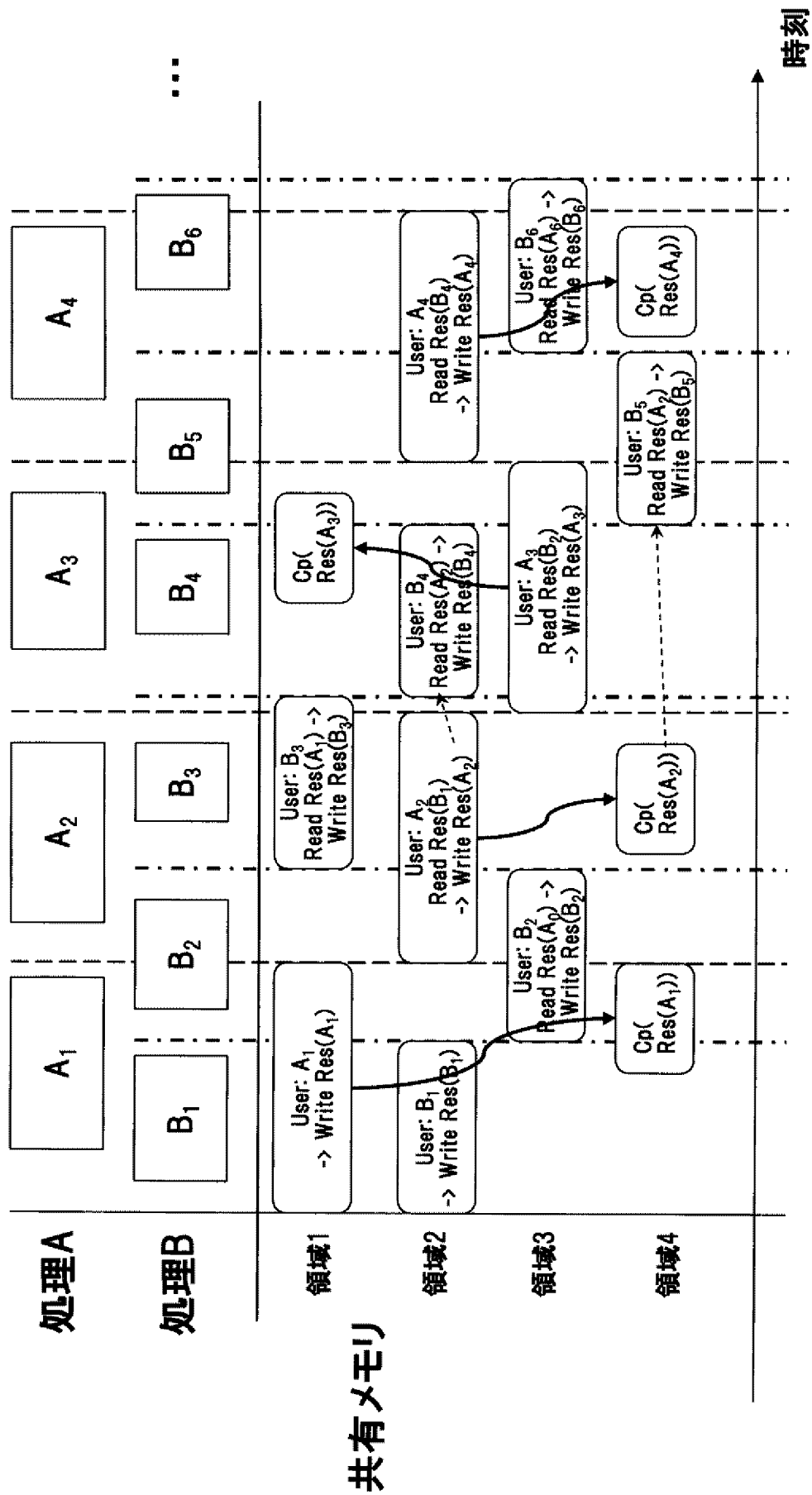
[図2]



[図3]



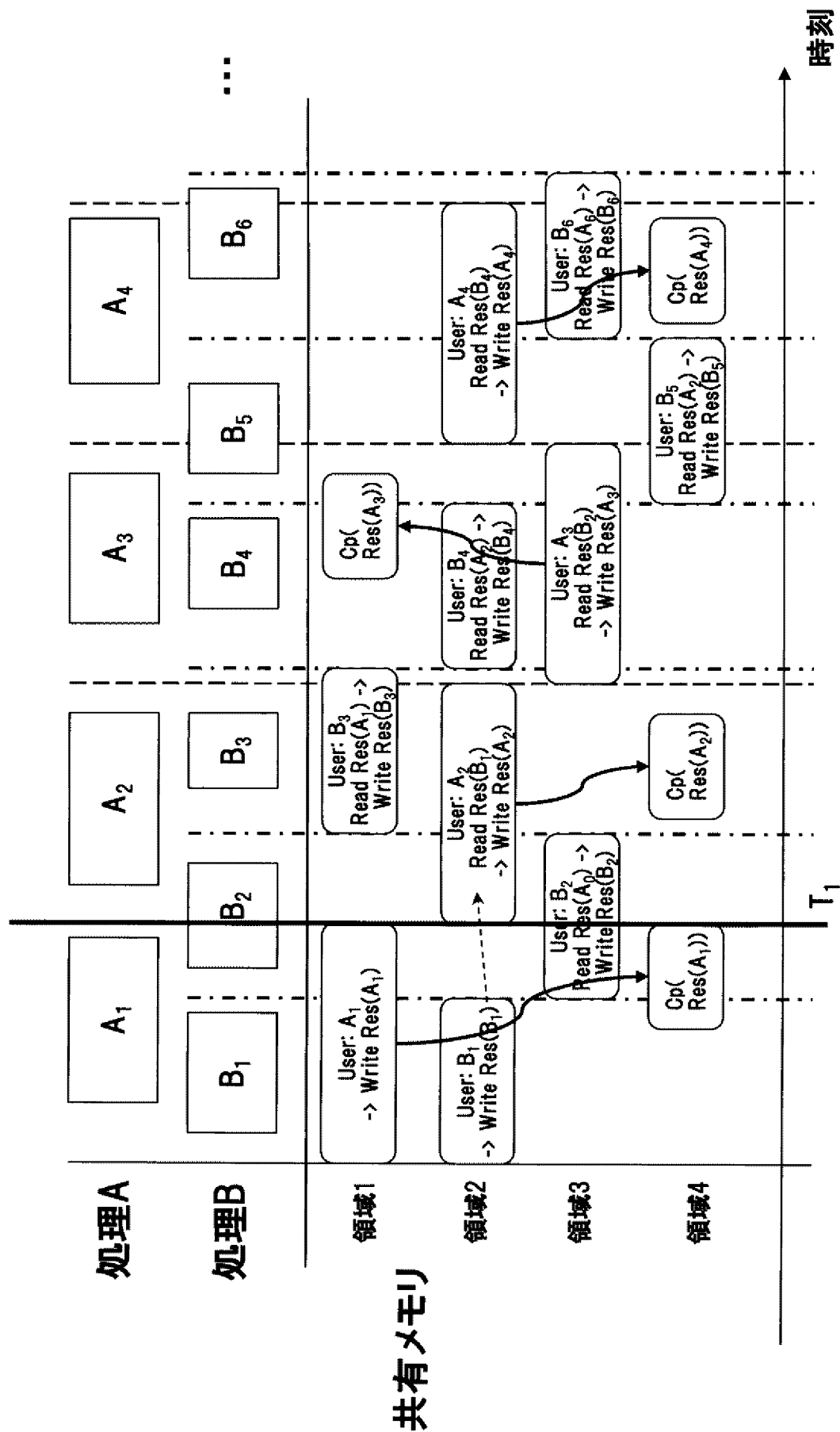
[図4]



[図5]

	占有状況		最新値	
	A占有中	B占有中	$A \rightarrow B$	$B \rightarrow A$
領域1			○	
領域2				○
領域3		○		
領域4			○	
		⋮		

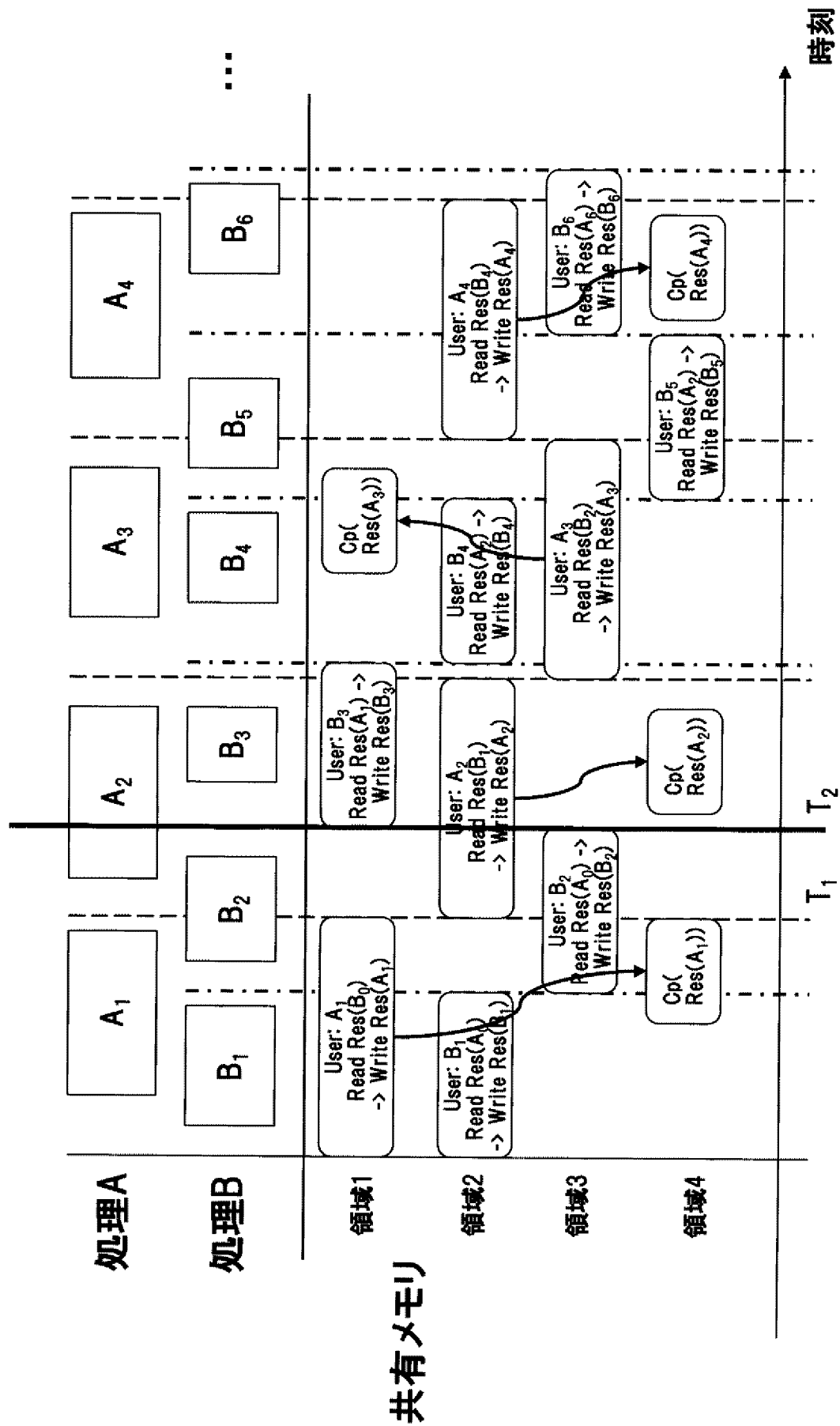
[図6]



[図7]

	占有状況		最新値	
	A占有中	B占有中	$A \rightarrow B$	$B \rightarrow A$
領域1			○	
領域2	○			○
領域3		○		
領域4			○	
		⋮		

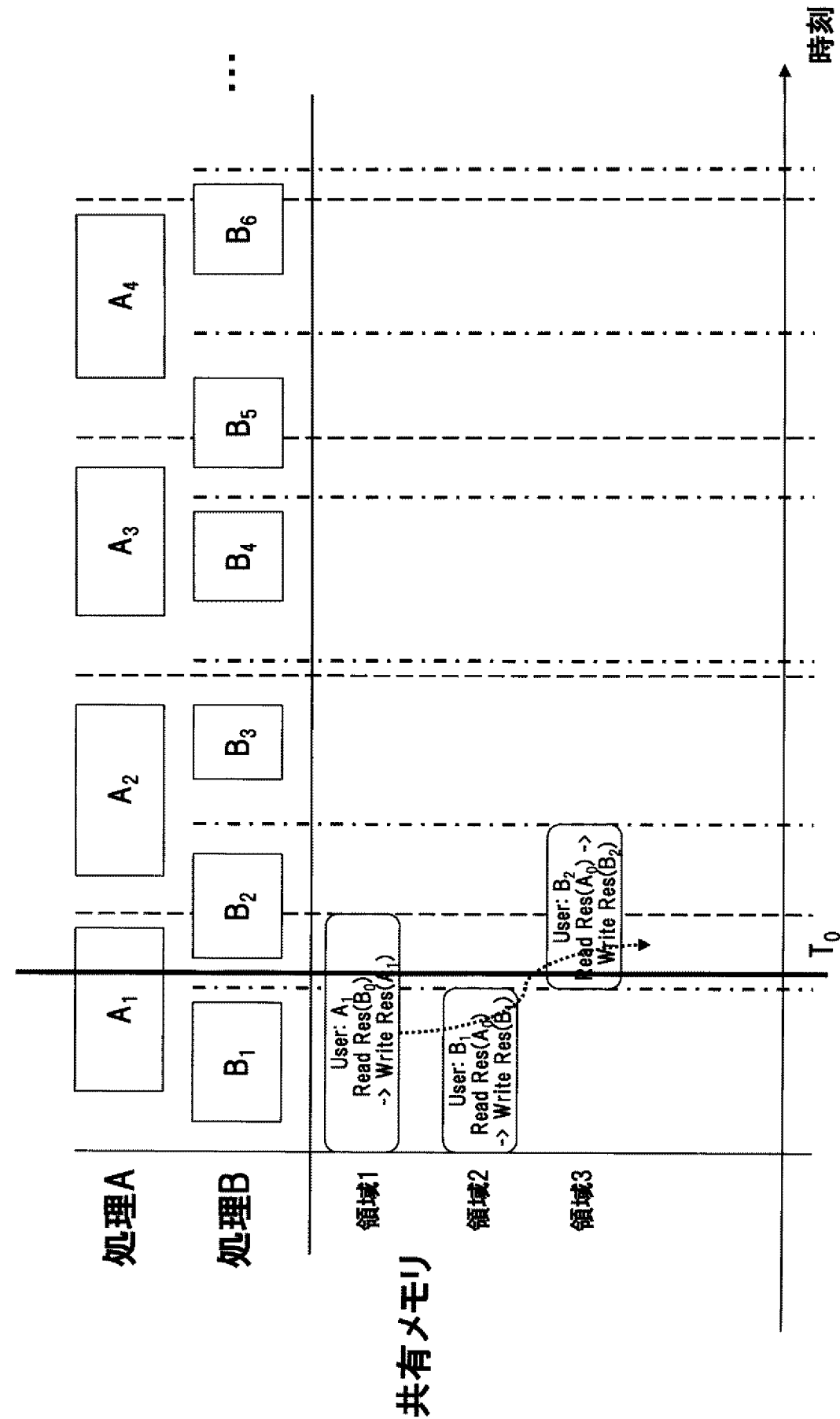
[図8]



[図9]

	占有状況		最新値	
	A占有中	B占有中	$A \rightarrow B$	$B \rightarrow A$
領域1			○	
領域2	○			
領域3				○
領域4			○	
		⋮		

[図10]

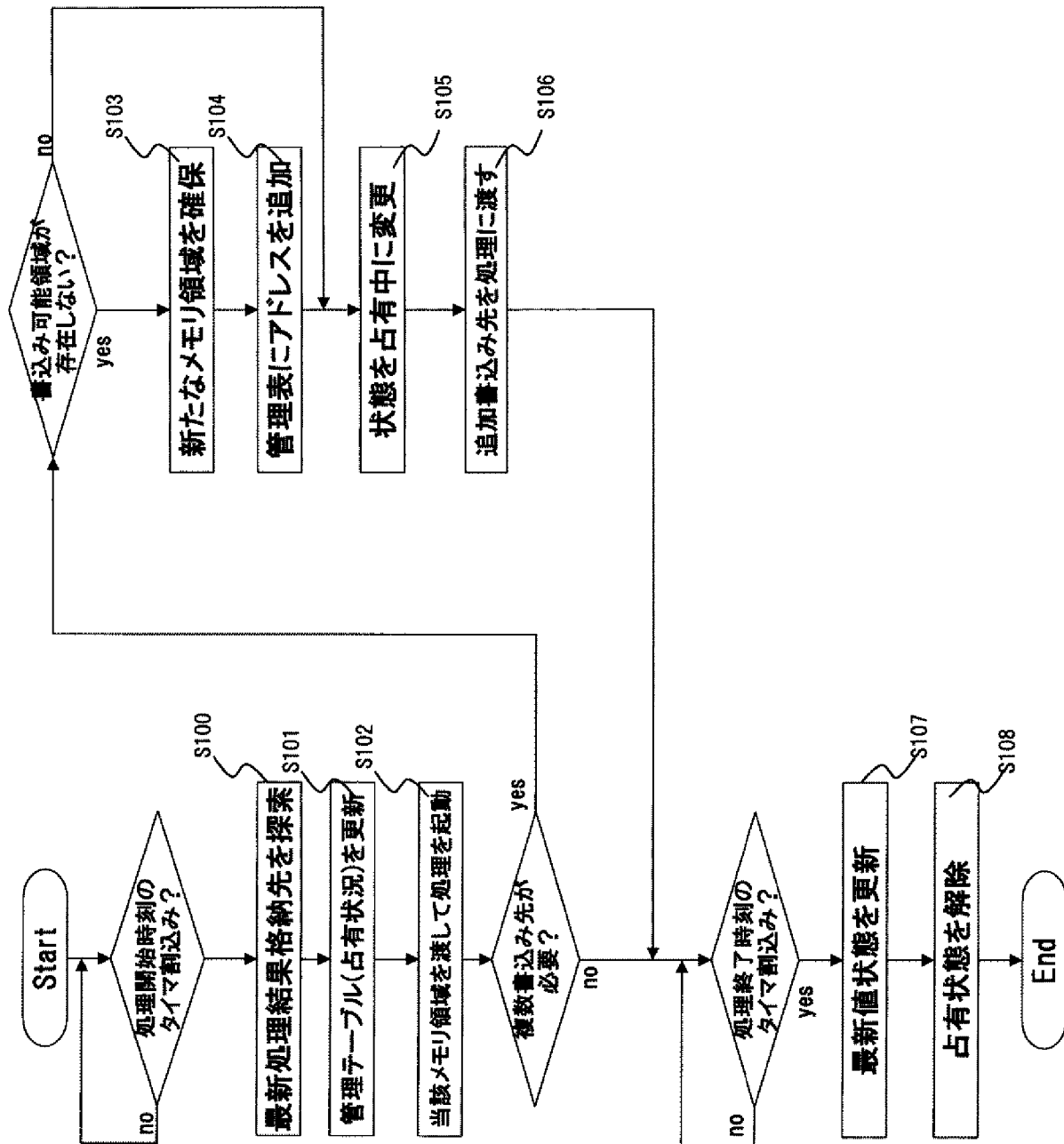


[図11]

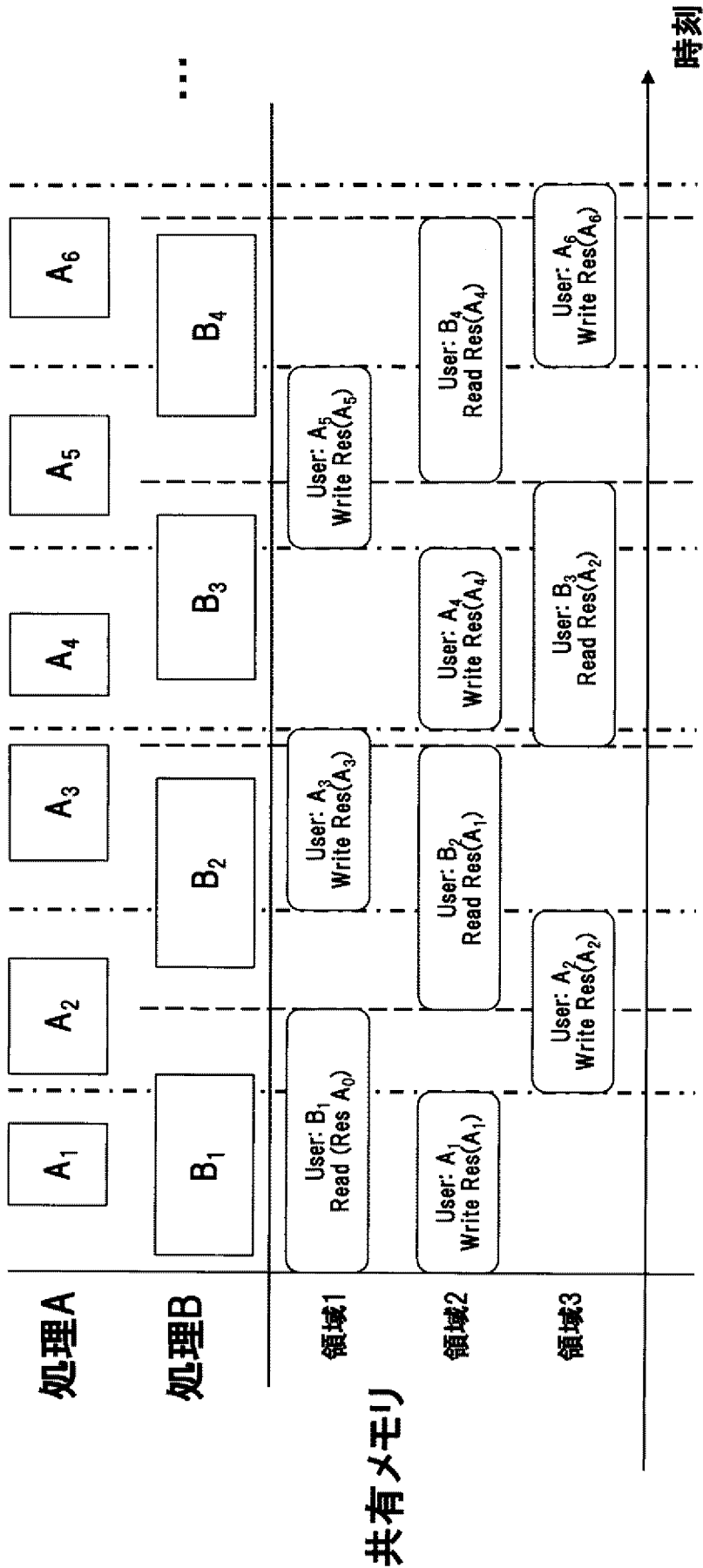
(作成) ----->

	占有状況		最新値	
	A占有中	B占有中	$A \rightarrow B$	$B \rightarrow A$
領域1	○		○	
領域2		○		
領域3				○
領域4	○		○	
		⋮		

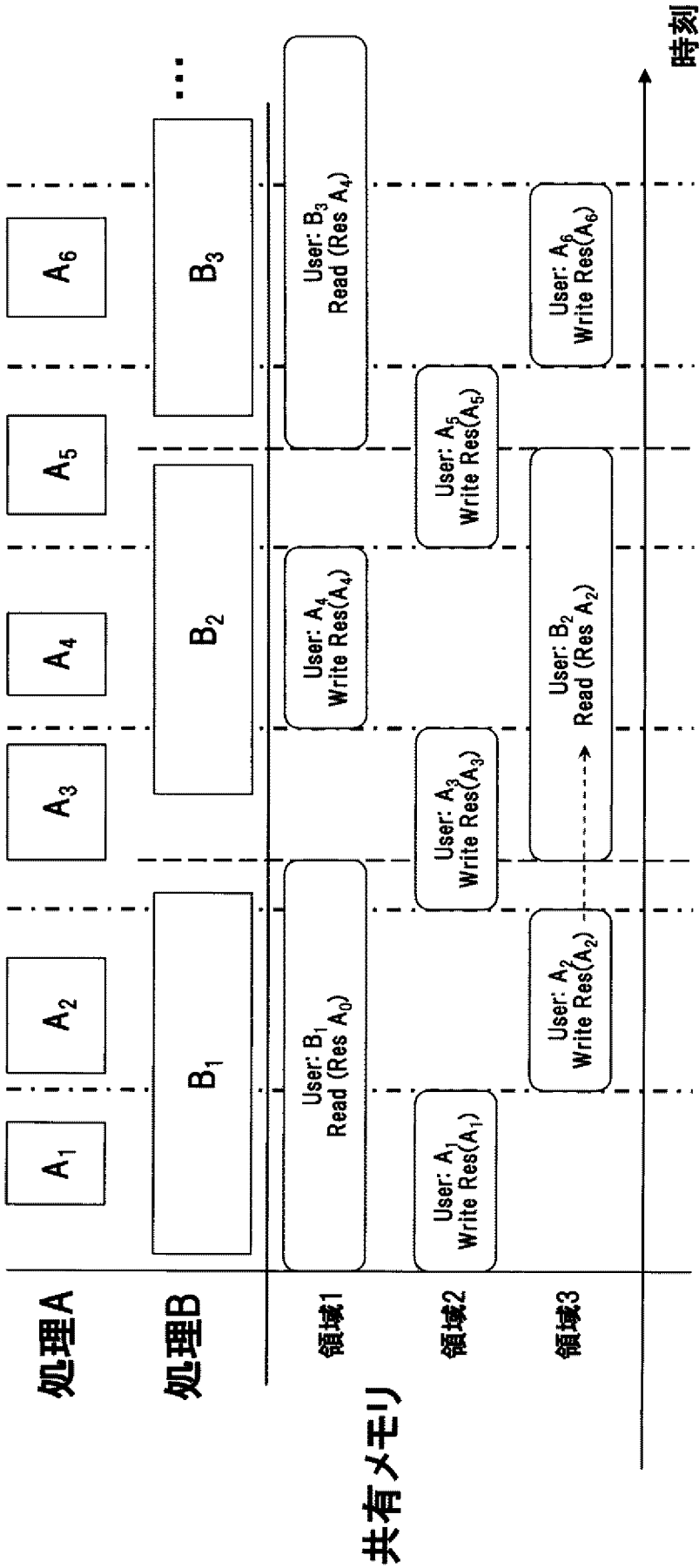
[図12]



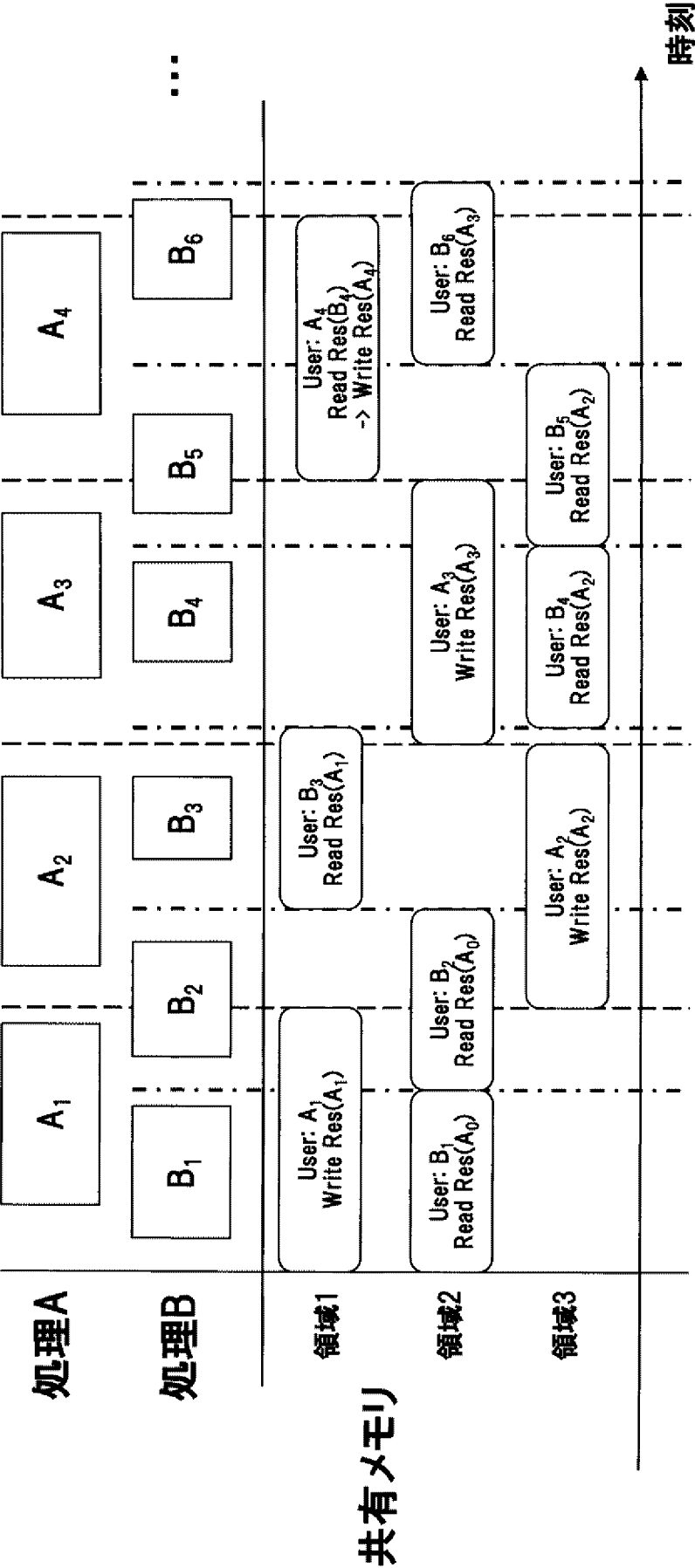
[図13]

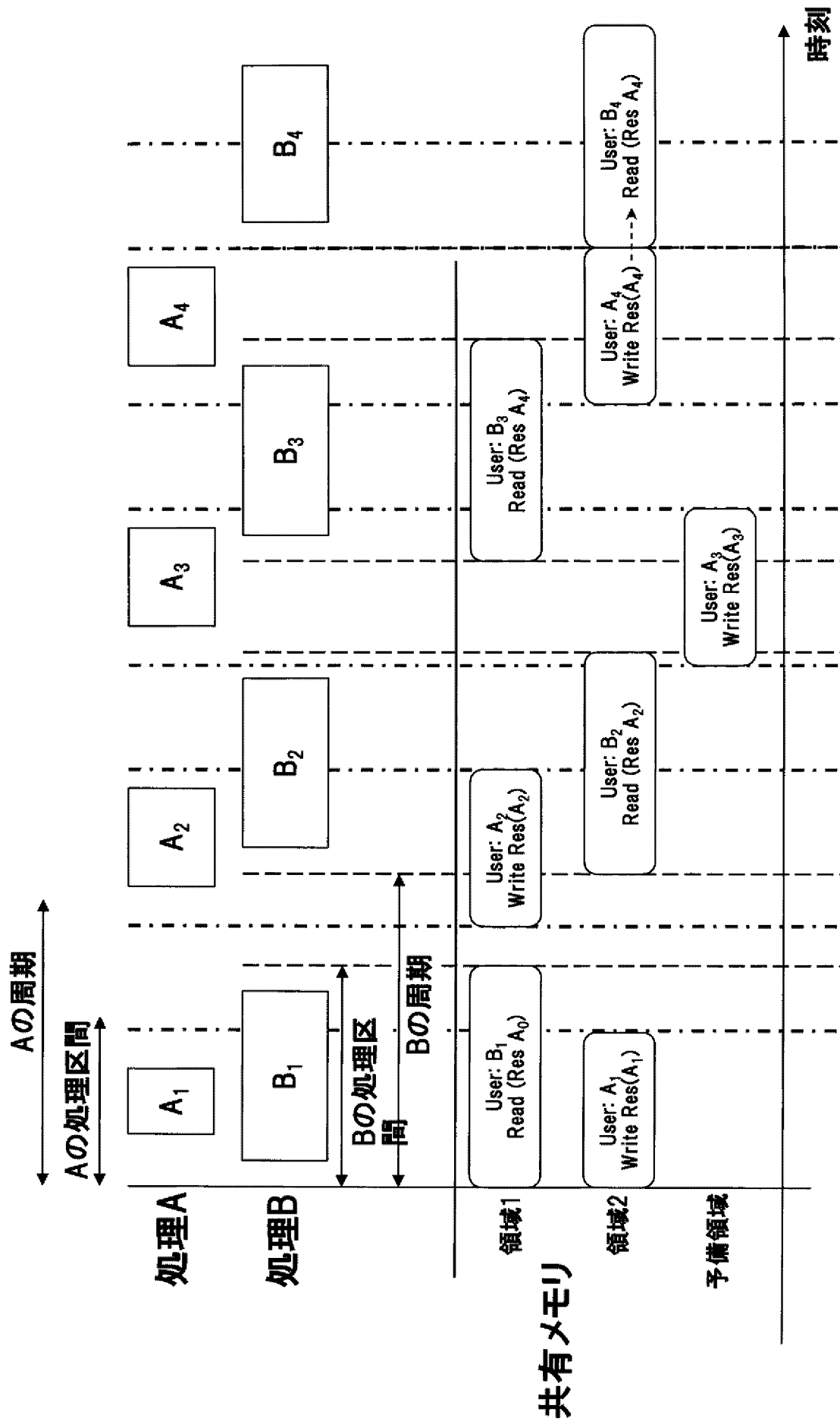


[図14]

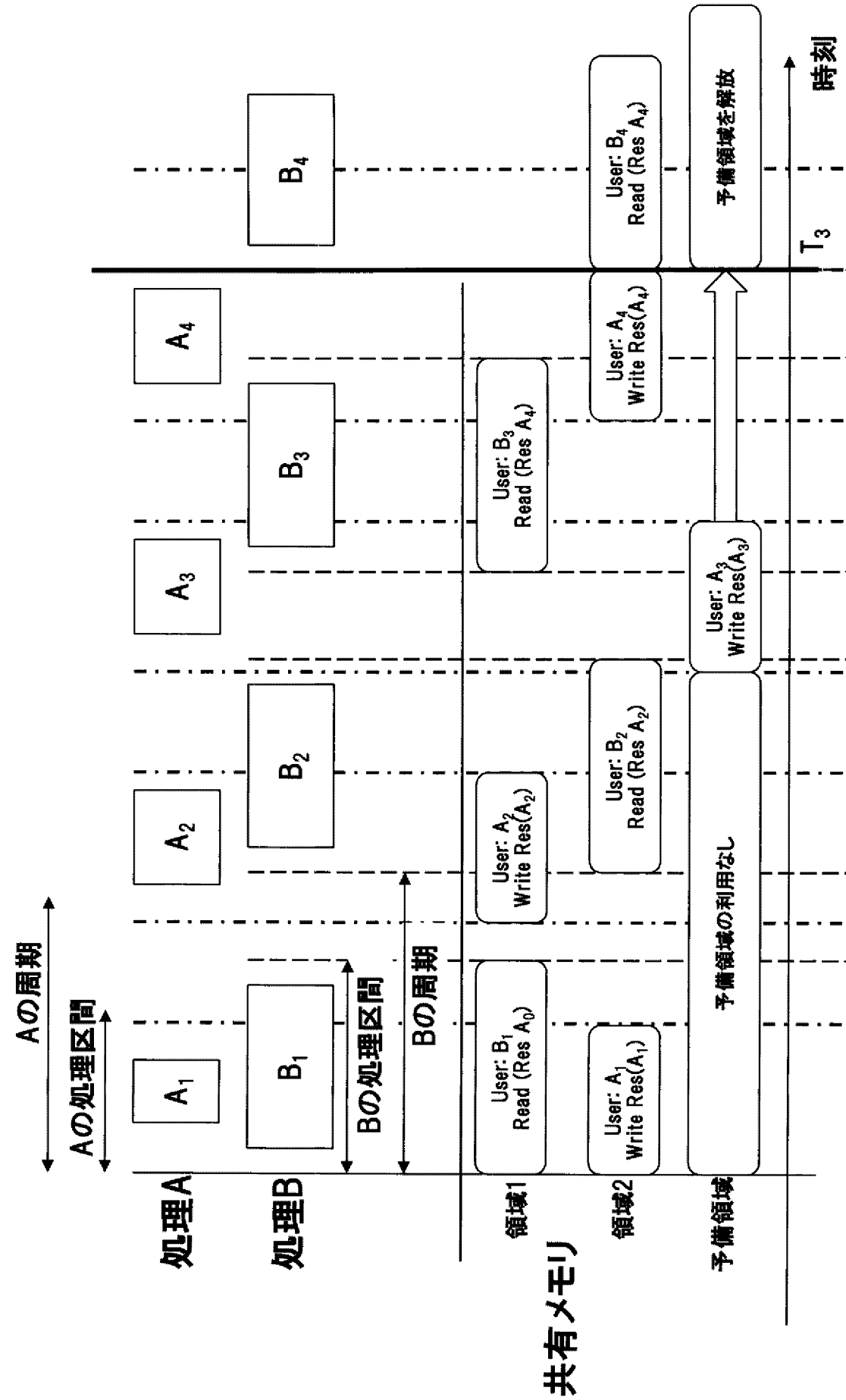


[図15]





[図17]



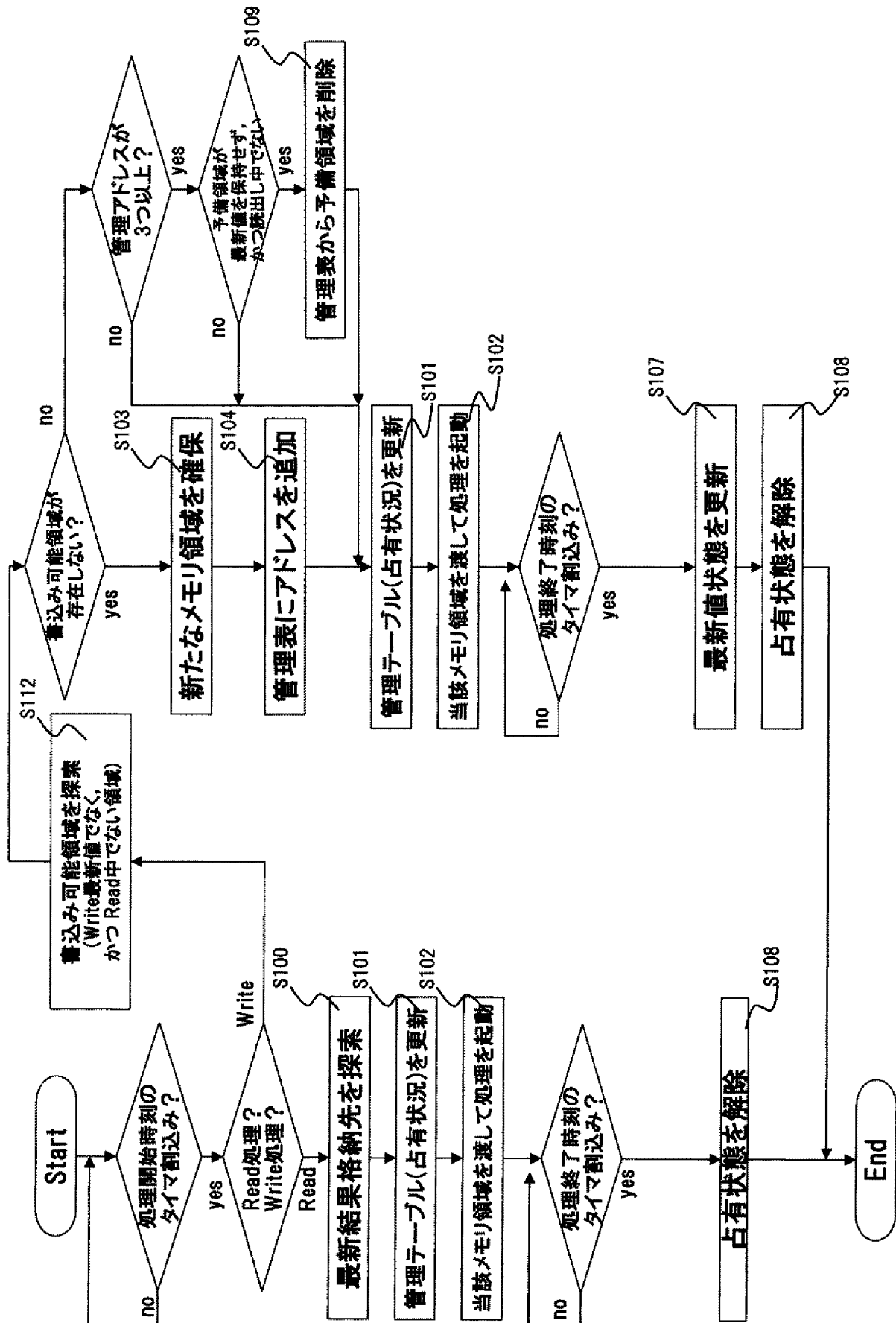
[図18]

	占有状況		最新値	
	A占有中	B占有中	$A \rightarrow B$	$B \rightarrow A$
領域1				
領域2	○			
予備領域			○	
		⋮		

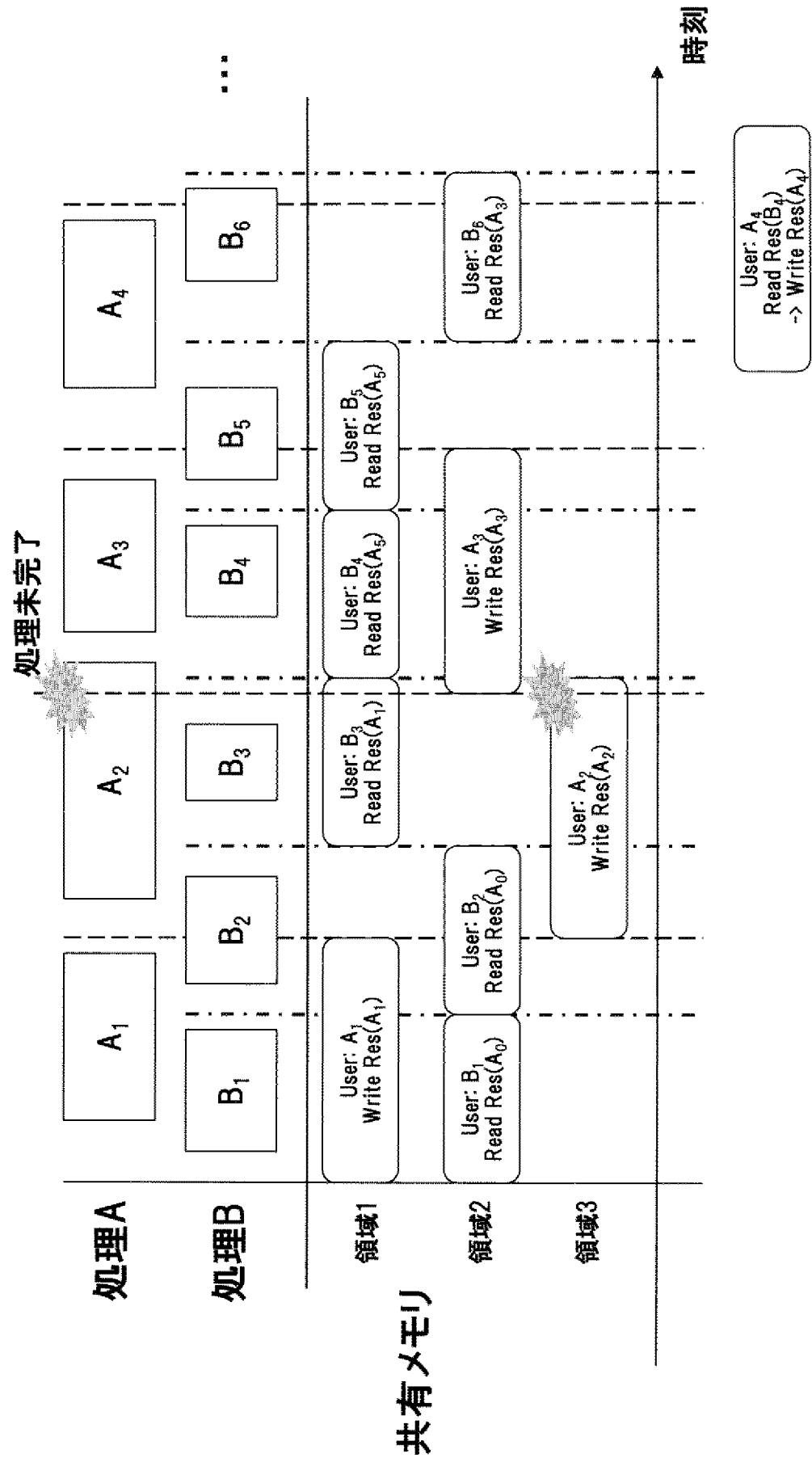
[図19]

	占有状況		最新値	
	A占有中	B占有中	$A \rightarrow B$	$B \rightarrow A$
領域1				
領域2		○	○	
予備領域				
		⋮		

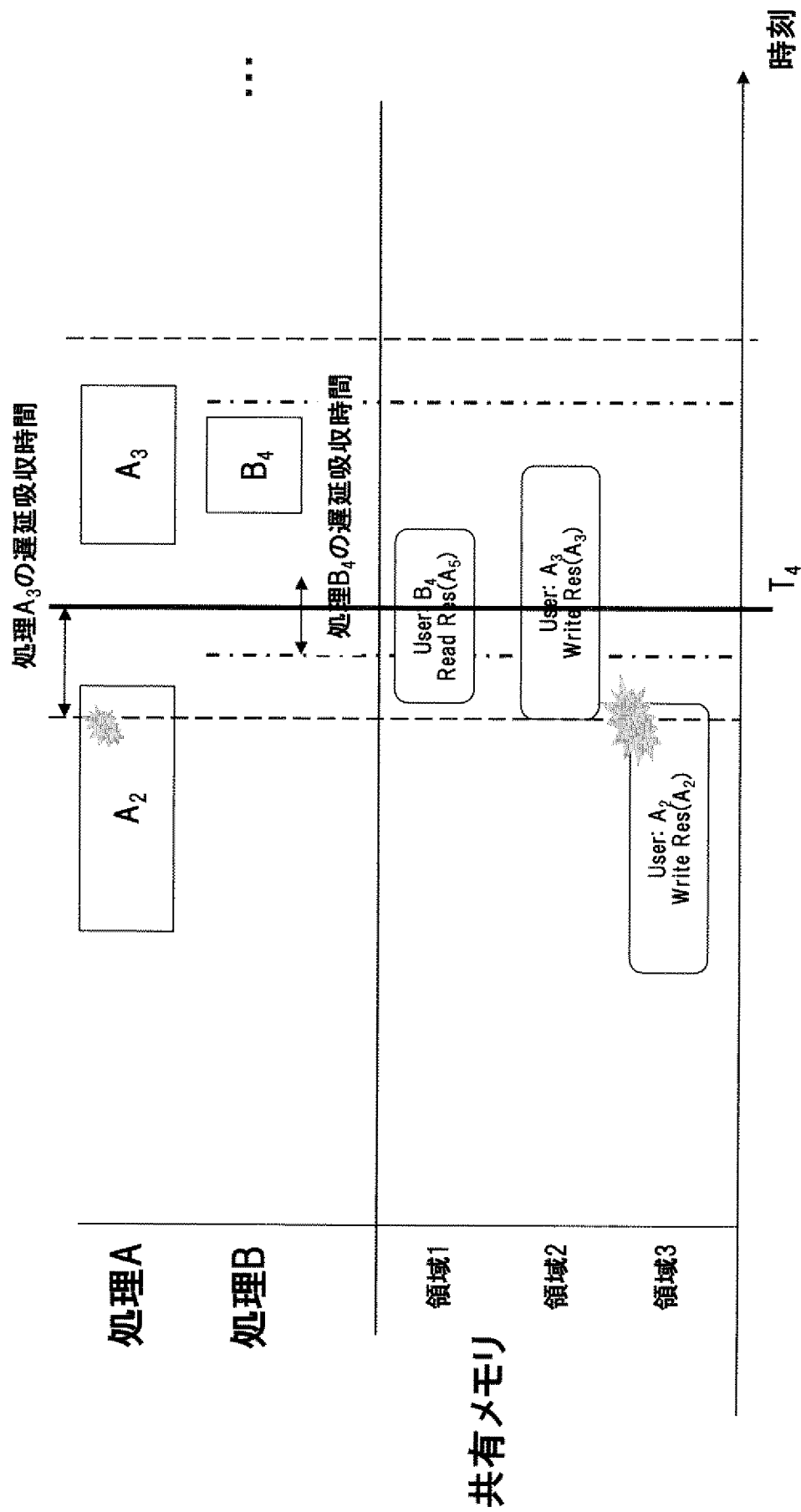
[図20]



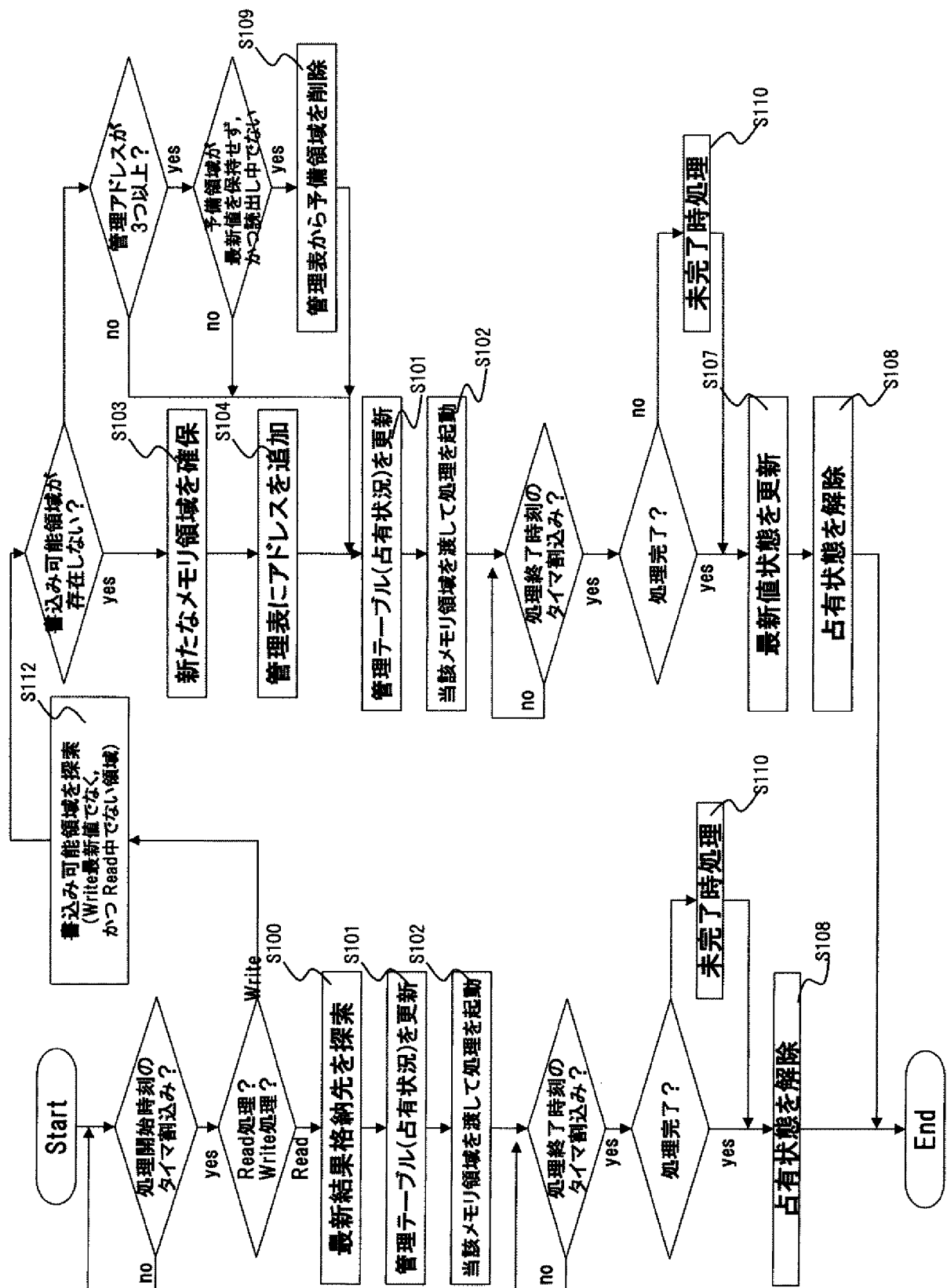
[図21]



[図22]



[図23]



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2021/024626

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/52(2006.01)i; G06F 9/54(2006.01)i; G06F 12/02(2006.01)i; G06F 15/167(2006.01)i

FI: G06F15/167 610B; G06F9/52 150Z; G06F12/02 510M; G06F9/54 A

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F9/52; G06F9/54; G06F12/02; G06F15/167

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Published examined utility model applications of Japan 1922-1996

Published unexamined utility model applications of Japan 1971-2021

Registered utility model specifications of Japan 1996-2021

Published registered utility model applications of Japan 1994-2021

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2010-244096 A (SEIKO EPSON CORP.) 28 October 2010 (2010-10-28) abstract	1-9
A	WO 2010/119932 A1 (NEC CORP.) 21 October 2010 (2010-10-21) abstract	1-9



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"I" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search
14 September 2021 (14.09.2021)Date of mailing of the international search report
21 September 2021 (21.09.2021)Name and mailing address of the ISA/
Japan Patent Office
3-4-3, Kasumigaseki, Chiyoda-ku,
Tokyo 100-8915, Japan

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/JP2021/024626

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
JP 2010-244096 A	28 Oct. 2010	(Family: none)	
WO 2010/119932 A1	21 Oct. 2010	(Family: none)	

A. 発明の属する分野の分類（国際特許分類（IPC）） G06F 9/52(2006.01)i; G06F 9/54(2006.01)i; G06F 12/02(2006.01)i; G06F 15/167(2006.01)i FI: G06F15/167 610B; G06F9/52 150Z; G06F12/02 510M; G06F9/54 A														
B. 調査を行った分野														
調査を行った最小限資料（国際特許分類（IPC）） G06F9/52; G06F9/54; G06F12/02; G06F15/167														
最小限資料以外の資料で調査を行った分野に含まれるもの														
<table border="0"> <tr> <td>日本国実用新案公報</td> <td>1922 - 1996年</td> </tr> <tr> <td>日本国公開実用新案公報</td> <td>1971 - 2021年</td> </tr> <tr> <td>日本国実用新案登録公報</td> <td>1996 - 2021年</td> </tr> <tr> <td>日本国登録実用新案公報</td> <td>1994 - 2021年</td> </tr> </table>			日本国実用新案公報	1922 - 1996年	日本国公開実用新案公報	1971 - 2021年	日本国実用新案登録公報	1996 - 2021年	日本国登録実用新案公報	1994 - 2021年				
日本国実用新案公報	1922 - 1996年													
日本国公開実用新案公報	1971 - 2021年													
日本国実用新案登録公報	1996 - 2021年													
日本国登録実用新案公報	1994 - 2021年													
国際調査で使用した電子データベース（データベースの名称、調査に使用した用語）														
C. 関連すると認められる文献														
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号												
A	JP 2010-244096 A（セイコーエプソン株式会社）28.10.2010（2010 - 10 - 28） 【要約】	1-9												
A	WO 2010/119932 A1（日本電気株式会社）21.10.2010（2010 - 10 - 21） 要約	1-9												
☐ C欄の続きにも文献が列挙されている。 ☒ パテントファミリーに関する別紙を参照。														
<table border="0"> <tr> <td>* 引用文献のカテゴリー</td> <td>“T” 国際出願日又は優先日後に公表された文献であって出願と抵触するものではなく、発明の原理又は理論の理解のために引用するもの</td> </tr> <tr> <td>“A” 特に関連のある文献ではなく、一般的な技術水準を示すもの</td> <td>“X” 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの</td> </tr> <tr> <td>“E” 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの</td> <td>“Y” 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの</td> </tr> <tr> <td>“L” 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）</td> <td>“&” 同一パテントファミリー文献</td> </tr> <tr> <td>“O” 口頭による開示、使用、展示等に言及する文献</td> <td></td> </tr> <tr> <td>“P” 国際出願日前で、かつ優先権の主張の基礎となる出願の日の後に公表された文献</td> <td></td> </tr> </table>			* 引用文献のカテゴリー	“T” 国際出願日又は優先日後に公表された文献であって出願と抵触するものではなく、発明の原理又は理論の理解のために引用するもの	“A” 特に関連のある文献ではなく、一般的な技術水準を示すもの	“X” 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの	“E” 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの	“Y” 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの	“L” 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）	“&” 同一パテントファミリー文献	“O” 口頭による開示、使用、展示等に言及する文献		“P” 国際出願日前で、かつ優先権の主張の基礎となる出願の日の後に公表された文献	
* 引用文献のカテゴリー	“T” 国際出願日又は優先日後に公表された文献であって出願と抵触するものではなく、発明の原理又は理論の理解のために引用するもの													
“A” 特に関連のある文献ではなく、一般的な技術水準を示すもの	“X” 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの													
“E” 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの	“Y” 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの													
“L” 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）	“&” 同一パテントファミリー文献													
“O” 口頭による開示、使用、展示等に言及する文献														
“P” 国際出願日前で、かつ優先権の主張の基礎となる出願の日の後に公表された文献														
国際調査を完了した日 14.09.2021	国際調査報告の発送日 21.09.2021													
名称及びあて先 日本国特許庁(ISA/JP) 〒100-8915 日本国 東京都千代田区霞が関三丁目4番3号	権限のある職員（特許庁審査官） 田中 幸雄 5B 9191 電話番号 03-3581-1101 内線 3545													

国際調査報告
パテントファミリーに関する情報

国際出願番号

PCT/JP2021/024626

引用文献			公表日	パテントファミリー文献	公表日
JP	2010-244096	A	28.10.2010	(ファミリーなし)	
WO	2010/119932	A1	21.10.2010	(ファミリーなし)	