(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)公開番号
特開2024-39643
(P2024-39643A)
(43)公開日　令和6年3月22日(2024.3.22)

(51)国際特許分類			FI			テーマコード（参考）
G06F	*12/00*	*(2006.01)*	G06F	12/00	570A	5B160
H04L	*49/90*	*(2022.01)*	H04L	49/90		5K030
G06F	*13/10*	*(2006.01)*	G06F	13/10	330A	
G06F	*13/12*	*(2006.01)*	G06F	13/12	330F	

審査請求　未請求　請求項の数　12　ＯＬ　外国語出願　（全11頁）

(21)出願番号　特願2023-145786(P2023-145786)
(22)出願日　令和5年9月8日(2023.9.8)
(31)優先権主張番号　10 2022 209 392.5
(32)優先日　令和4年9月9日(2022.9.9)
(33)優先権主張国・地域又は機関
ドイツ(DE)

(71)出願人　591245473
ロベルト・ボッシュ・ゲゼルシャフト・ミト・ベシュレンクテル・ハフツング
ＲＯＢＥＲＴ　ＢＯＳＣＨ　ＧＭＢＨ
ドイツ連邦共和国　７０４４２　シュトゥットガルト　ポストファッハ　３０　０２２０
(74)代理人　100118902
弁理士　山本 修
(74)代理人　100196508
弁理士　松尾 淳一
(72)発明者　オリバー・シューネマン
ドイツ国　３１１３９　ヒルデスハイム，バルニンガーオーダー・カンプ　２アー

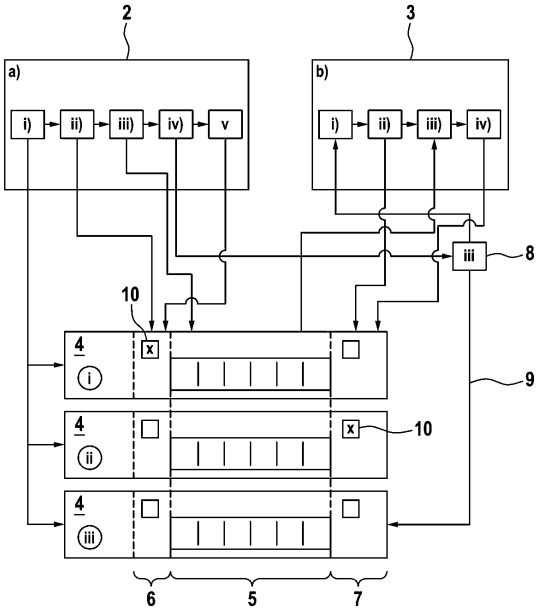
Ｆターム（参考）　5B160　CA17 KA06

最終頁に続く

(54)【発明の名称】　定義済みデータを繰り返し伝送するための方法

(57)【要約】
【課題】定義済みデータを繰り返し伝送するための方法を提供すること。
【解決手段】定義済みデータ（１）をデータ送信モジュール（２）からデータ受信モジュール（３）に繰り返し伝送するための方法であって、方法の実行のために、３つのデータ伝送メモリ（４）が使用され、データ伝送メモリ（４）がそれぞれ、データメモリブロック（５）と、データ送信モジュール（２）のアクセス予約（１０）を記憶するための第１のメモリビット（６）と、受信モジュール（３）のアクセス予約（１０）を記憶するための第２のメモリビット（７）とを有し、方法の実行のために、さらにポインタメモリ（８）が使用され、ポインタメモリ（８）に、現在のデータ伝送メモリ（４）に向けたポインタ（９）を保存することができる、方法。
【選択図】図１

Fig. 1



【特許請求の範囲】
【請求項１】
定義済みデータ（１）をデータ送信モジュール（２）からデータ受信モジュール（３）に繰り返し伝送するための方法であって、前記方法の実行のために、３つのデータ伝送メモリ（４）が使用され、前記データ伝送メモリ（４）がそれぞれ、データメモリブロック（５）と、前記データ送信モジュール（２）のアクセス予約（１０）を記憶するための第１のメモリビット（６）と、前記受信モジュール（３）のアクセス予約（１０）を記憶するための第２のメモリビット（７）とを有し、前記方法の実行のために、さらにポインタメモリ（８）が使用され、前記ポインタメモリ（８）に、現在のデータ伝送メモリ（４）に向けたポインタ（９）を保存することができ、
ａ）データ書込みプロセスを実行する、
ｉ）前記第１のメモリビット（６）と前記第２のメモリビット（７）とのどちらにもアクセス予約（１０）が登録されていないデータ伝送メモリ（４）を決定するステップと、
ｉｉ）前記データ送信モジュール（２）によって、前記それぞれのデータ伝送メモリ（４）の前記第１のメモリビット（６）へのアクセス予約（１０）の設定により、ステップａ－ｉ）で決定された前記データ伝送メモリ（４）を予約するステップと、
ｉｉｉ）前記定義済みデータ（１）を、前記データ伝送メモリ（４）の前記データメモリブロック（５）に書き込むステップと、
ｉｖ）前記ポインタメモリ（８）で、前記使用されているデータ伝送メモリに向けたポインタ（９）を設定するステップと、
ｖ）ステップａ－ｉｉ）で設定された前記アクセス予約（１０）の解除によって、前記データ伝送メモリ（４）を解放するステップと、
を含むステップと、
ｂ）データ読出しプロセスを実行する、
ｉ）前記ポインタメモリ（８）での前記ポインタ（９）に基づいて、前記読出し対象のデータ伝送メモリ（４）を決定するステップと、
ｉｉ）前記データ受信モジュール（３）によって、前記それぞれのデータ伝送メモリ（４）の前記第２メモリビット（７）へのアクセス予約（１０）の設定により、前記読出し対象のデータ伝送メモリ（４）を予約するステップと、
ｉｉｉ）前記定義済みデータ（１）を、前記データ伝送メモリ（４）の前記データメモリブロック（５）から読み出すステップと、
ｉｖ）ステップｂ－ｉ）で設定された前記アクセス予約（１０）の解除により、前記データ伝送メモリ（４）を解放するステップと、
を含むステップと、
を含む方法。
【請求項２】
最新の定義済みデータ（１）を前記データ送信モジュール（２）から前記データ受信モジュール（３）に恒久的に提供するためにループ式に恒久的に繰り返される、請求項１に記載の方法。
【請求項３】
前記ステップａ）と前記ステップｂ）とが時間をずらして互いに重複して恒久的に繰り返し実行され、前記ステップａ）およびｂ）内で前記ステップがそれぞれ順次実行される、請求項１または２に記載の方法。
【請求項４】
少なくとも前記データ送信モジュール（２）または前記データ受信モジュール（３）がどちらも、自動車の高度自動運転動作機能に使用するための生データを加工するためのネットワーク内のデータ処理モジュールであり、前記データ送信モジュール（２）と前記データ受信モジュール（３）との間で伝送される定義済みデータ（１）が、加工された生データであり、高度自動運転動作機能を提供するために前記データ受信モジュール（３）に

よってさらに処理される、請求項１～３のいずれか一項に記載の方法。
【請求項５】
生データが、
－　自動車のセンサからのセンサデータと、
－　高度自動運転動作のために車両が利用することができる、車両外部のデータソースからのデータと、
－　高度自動運転動作のために前記車両内のデータメモリに提供されるデータと、
を含む、請求項４に記載の方法。
【請求項６】
前記ステップａ）およびｂ）による定義済みデータ（１）の前記繰り返しの伝送が、前記ネットワーク内の複数のデータ処理モジュール間で行われる、請求項４または５に記載の方法。
【請求項７】
前記ステップａ－ｉ）およびａ－ｉｉ）での前記データ伝送メモリ（４）の前記決定および予約中、ステップｂ－ｉ）およびｂ－ｉｉ）の実行に関してすべてのデータ伝送メモリ（４）が遮断される、請求項１～６のいずれか一項に記載の方法。
【請求項８】
ステップａ－ｉ）およびａ－ｉｉ）での前記データ伝送メモリ（３）の前記決定および予約中、まず前記第１のメモリビット（６）および前記第２のメモリビット（７）に基づいて、データ伝送メモリ（４）が空いているかどうかがチェックされ、次いでまず一時予約、および前記第１のメモリビット（６）と前記第２のメモリビット（７）との後続の再チェックが行われる、請求項１～７のいずれか一項に記載の方法。
【請求項９】
ステップａ－ｉ）で、前記ポインタメモリ（８）に保存されているポインタ（９）が指し示しているデータ伝送メモリ（４）が考慮に入れられない、請求項１～８のいずれか一項に記載の方法。
【請求項１０】
データ処理用の装置であって、少なくとも１つのデータ送信モジュール（２）および１つのデータ受信モジュール（３）と、少なくとも３つのデータ伝送メモリ（４）と、ポインタメモリ（８）とを含み、請求項１～９のいずれか一項に記載の方法を実行するように構成されている装置。
【請求項１１】
コンピュータによって実行されるときに、前記コンピュータに請求項１～９のいずれか一項に記載の方法を実行させる命令を含むコンピュータプログラム製品。
【請求項１２】
コンピュータによって実行されるときに、前記コンピュータに請求項１～９のいずれか一項に記載の方法／ステップを実行させる命令を含むコンピュータ可読記憶媒体。
【発明の詳細な説明】
【技術分野】
【０００１】
運転支援または自動運転のためのシステムは、通常は相互にデータを交換する多数の個別のソフトウェアユニットからなる。好ましくは、これらのソフトウェアユニット（ランナブル（Ｒｕｎｎａｂｌｅ）、ノード、またはデータ処理コンポーネントもしくはデータ処理モジュールとも呼ばれることが多い）のそれぞれは、指定の入力データおよび指定の出力データを有する。
【背景技術】
【０００２】
上記のシステムでは、入力データ、例えばレーダやビデオなどのセンサからのデータが処理され、そこから高度自動運転機能のためのより正確なデータを取得する。例えば、ビデオデータから、データ処理コンポーネントからなる多段階処理カスケードで、交通標識

に関連するデータが抽出される。
【０００３】
したがって、様々なデータ処理コンポーネントまたはデータ処理モジュールは、センサデータに基づいてアクションを実行するために、生データまたはセンサデータが処理される複雑なデータ処理ネットワークを定期的に形成する。そのようなアクションは、例えば、車両の自律運転動作の枠組みでの制御タスクでよい。
【０００４】
異なるデータ処理コンポーネントおよびデータ処理モジュール間でデータを交換するための効率的な方法が必要である。そのようなデータ交換では、通常、データ送信モジュール（送信側とも呼ばれる）およびデータ受信モジュール（受信側とも呼ばれる）が存在する。そのような交換方法には、様々な要件が課されることがある。よくある要件は、常にできるだけ最新のデータがデータ送信モジュールからデータ受信モジュールに提供される必要があることである。これは、データ送信モジュールとデータ受信モジュールとの間で、同様の定義済みデータが定期的に交換される場合に特に当てはまる。この例としては、例えば、カメラデータを加工し、定義済みデータとしてエッジデータを提供するデータ送信モジュールと、この定義済みデータを処理して交通標識認識などの機能を実行するデータ受信モジュールとが挙げられる。そのような用途では、場合によっては、提供された定義済みデータの個々のセットがすべてデータ受信モジュールによって受信される必要はない。データ受信モジュールが各場合にできるだけ最新の定義済みデータを得ることがはるかに重要である。これには、データ送信モジュールが、データ読出し処理に関係なく、定義済みデータを途切れることなく提供することができる必要がある。
【０００５】
そのような用途には、データ伝送方法が必要である。そのようなデータ伝送方法は、メッセージキューと呼ばれることも多い。
さらなる要件は、そのようなデータ伝送方法を、いわゆる共有メモリ環境（共有されたメモリを有する環境）で定期的に実行しなければならないことである。共有メモリ環境では、データ処理モジュールは通常、オペレーティングシステムによるアクセス制御なしでデータ処理モジュールがアクセスすることができるメモリを共有する。オペレーティングシステムは、データ処理モジュールがメモリを利用できるようにし、アクセスを許可する。しかし、オペレーティングシステムは、データ処理モジュールがメモリにアクセスする方法を制御しない。そのような環境は、データ処理モジュール間でデータを交換するのに必要な計算量を大幅に削減する。しかし、この環境は、上位のインスタンスがアクセス間での衝突を妨げないので、そのような衝突が発生する危険性を伴う。
【発明の概要】
【発明が解決しようとする課題】
【０００６】
様々なミドルウェアでの共有メモリ通信に基づく既存のメッセージキューは、送信側と受信側とが互いにブロックする可能性があるという欠点を有している。さらに、送信側と受信側とが共に共有メモリへの読出しおよび書込みアクセスを行うことができることも必要である。
【課題を解決するための手段】
【０００７】
本明細書に記載の方法およびこの方法を実行するための記載の装置を用いて、どの時点においても送信側と受信側とが互いをブロックすることがないメッセージキューを述べる。さらに、この解決策は、共有の読出し／書込みメモリを必要としない。
【０００８】
本明細書では、定義済みデータをデータ送信モジュールからデータ受信モジュールに繰り返し伝送するための方法であって、方法の実行のために、３つのデータ伝送メモリが使用され、データ伝送メモリがそれぞれ、データメモリブロックと、データ送信モジュールのアクセス予約を記憶するための第１のメモリビットと、受信モジュールのアクセス予約

を記憶するための第２のメモリビットとを有し、方法の実行のために、さらにポインタメモリが使用され、ポインタメモリに、現在のデータ伝送メモリに向けたポインタを保存することができ、
ａ）データ書込みプロセスを実行する、
ｉ）第１のメモリビットと第２のメモリビットとのどちらにもアクセス予約が登録されていないデータ伝送メモリを決定するステップと、
ｉｉ）データ送信モジュールによって、それぞれのデータ伝送メモリの第１のメモリビットへのアクセス予約の設定により、ステップａ－ｉ）で決定されたデータ伝送メモリを予約するステップと、
ｉｉｉ）定義済みデータを、データ伝送メモリのデータメモリブロックに書き込むステップと、
ｉｖ）ポインタメモリで、使用されているデータ伝送メモリに向けたポインタを設定するステップと、
ｖ）ステップａ－ｉｉ）で設定されたアクセス予約の解除によって、データ伝送メモリを解放するステップと、
を含むステップと、
ｂ）データ読出しプロセスを実行する、
ｉ）ポインタメモリでのポインタに基づいて、読出し対象のデータ伝送メモリを決定するステップと、
ｉｉ）データ受信モジュールによって、それぞれのデータ伝送メモリの第２メモリビットへのアクセス予約の設定により、読出し対象のデータ伝送メモリを予約するステップと、
ｉｉｉ）データを、データ伝送メモリのデータメモリブロックから読み出すステップと、
ｉｖ）ステップｂ－ｉ）で設定されたアクセス予約の解除により、データメモリを解放するステップと、
を含むステップと、
を含む方法を述べる。

【０００９】

本明細書に記載の方法は、特別な利点として、データ送信モジュールとデータ受信モジュールとの間の１：１通信のために好ましくは正確に３つのデータ伝送メモリが提供されることを利用する。モジュールは、特に自動車に高度自動機能を提供するためのシステムの一部である任意のデータ処理モジュールまたはデータ処理コンポーネントでよい。

【００１０】

本明細書に記載の方法は、いわゆるメッセージキューを利用する。メッセージキューは、メッセージの送信側（データ送信モジュール）と受信側（データ受信モジュール）とが同時にメッセージキューと対話する必要がないことにより、非同期通信プロトコルを提供する。データ伝送メモリの読出しのためのアクセス予約を記憶するための第２のメモリビットにより、データ読出しモジュールは、読出し処理のためにデータを予約することが可能になる。同時に、データ書込みモジュールは、他の２つのデータ伝送メモリのうちの１つを使用して、最新のデータを送信することができる。

【００１１】

ステップａ）およびそのステップは、データ送信モジュールによって実行される。ステップｂ）およびそのステップは、データ受信モジュールによって実行される。ここで、どちらのモジュールも、３つのデータ伝送メモリが提供される共通のメモリ領域にアクセスする。

【００１２】

データ受信モジュールのデータ読出しアクセスにより、ほぼ常に、読出しアクセスのためにメモリの１つがブロックされる。データ受信モジュールが定義済みデータを読み出す速度に関係なく、データ送信モジュールは、残りの２つのデータ伝送メモリの一方または

他方に交互にデータを書き込むことができる。ステップａ－ｉ）およびａ－ｉｉ）による空いているデータ伝送メモリの決定および予約により、ならびにステップｂ－ｉ）およびｂ－ｉｉ）でのそれぞれのアクセス予約の設定により、３つのデータ伝送メモリから、現在アクセス予約済みとなっていないデータ伝送メモリが常に選択されることが保証される。ステップａ－ｉｖ）およびｂ－ｉ）に従って、ポインタメモリでのポインタを使用してそれぞれ最新のデータセットを指し示すことにより、データ受信モジュールに、最新の定義済みデータを含むデータ伝送メモリが常に通知される。

【００１３】

既存の解決策は、大抵は、データ伝送メモリを２つだけ使用していた。しかし、そのような解決策に対し、本明細書に記載の解決策は大きな利点を有する。

２つのデータ伝送メモリを使用する場合、データ送信モジュールが、最後に書き込まれたデータを、新たに定義されたデータの提供に伴って上書きしなければならない、または最後に書き込まれたデータがデータ受信モジュールによって完全に読み込まれるまで待機しなければならない場合が生じ得る。これにより、データ伝送の速度および適時性が大幅に低下する。これは、３つのデータ伝送メモリによって回避することができる。

【００１４】

記載の方法、または記載の方法によって提供されるメッセージキューを用いて、共有の読出し／書込みメモリなしでデータ送信モジュールとデータ受信モジュールとの間で通信を行うことができる。

【００１５】

データ送信モジュールは、データまたはフラグおよびポインタを第１のメモリビット、データメモリブロック、およびポインタ用のポインタメモリに単に書き込む。データ送信モジュールは、好ましくは、データ伝送メモリのこれらの部分に対してのみアクセス権限を有する。データ受信モジュールは、好ましくは、データ伝送メモリのこれらの部分に対して読出しアクセスのみを行うことができる。

【００１６】

データ受信モジュールは、データまたはフラグを第２のメモリビットに単に書き込む。また、データ受信モジュールは、好ましくは、データ伝送メモリのこれらの部分に対してのみアクセス権限を有する。データ送信モジュールは、好ましくは、データ伝送メモリのこれらの部分に対して読出しアクセスのみを行うことができる。

【００１７】

最新の定義済みデータをデータ送信モジュールからデータ受信モジュールに恒久的に提供するためにループ式に恒久的に繰り返されると特に有利である。

さらに、ステップａ）とステップｂ）とが時間をずらして互いに重複して恒久的に繰り返し実行され、ステップａ）およびｂ）内でステップがそれぞれ順次実行されると有利である。

【００１８】

好ましくは、データ受信モジュールとデータ送信モジュールとは、既に処理した最新のデータをデータ送信モジュールがデータ受信モジュールに常に提供するように、互いに独立して恒久的に動作する。

【００１９】

また、少なくともデータ送信モジュールまたはデータ受信モジュールがどちらも、自動車の高度自動運転動作機能に使用するための生データを加工するためのネットワーク内のデータ処理モジュールであり、データ送信モジュールとデータ受信モジュールとの間で伝送される定義済みデータが、加工された生データであり、高度自動運転動作機能を提供するためにデータ受信モジュールによってさらに処理されると有利である。

【００２０】

さらに、生データが、

－　自動車のセンサからのセンサデータと、

－　高度自動運転動作のために車両が利用することができる、車両外部のデータソース

からのデータと、
－　高度自動運転動作のために車両内のデータメモリに提供されるデータと
を含む。
【００２１】
同様に、ステップａ）およびｂ）による定義済みデータの繰り返しの伝送が、ネットワーク内の複数のデータ処理モジュール間で行われると有利である。
さらに、ステップａ－ｉ）およびａ－ｉｉ）でのデータ伝送メモリの決定および予約中、ステップｂ－ｉ）およびｂ－ｉｉ）の実行に関してすべてのデータ伝送メモリが遮断されると有利である。
【００２２】
さらに、ステップａ－ｉ）およびａ－ｉｉ）でのデータ伝送メモリの決定および予約中、まず第１のメモリビットおよび第２のメモリビットに基づいて、データ伝送メモリが空いているかどうかがチェックされ、次いでまず一時予約、および第１のメモリビットと第２のメモリビットとの後続の再チェックが行われると有利である。
【００２３】
一時予約を伴うこの手順により、データ受信モジュールとデータ受信モジュールとの衝突が起こらないことを保証することができる。一時予約が設定されるとすぐに、データ受信モジュールは、このデータ伝送メモリを読出しアクセスでブロックすることができなくなる。一時予約後の再チェックにより、場合により、それぞれのデータ伝送メモリの選択と一時予約との間で読出しアクセスが行われたかどうかを確認することができる。データ受信モジュールによるアクセス予約が確認された場合、一時予約は再び解除され、好ましくは、データ伝送メモリを決定して予約するための新たな試行が行われる。
【００２４】
さらに、ステップａ－ｉ）で、ポインタメモリに保存されているポインタが指し示しているデータ伝送メモリが考慮に入れられないと有利である。
各場合に最後に保存された定義済みデータのセットは、このデータ伝送メモリに保存され、データ読出しプロセスまたはデータ受信モジュールに利用できるようになる。このため、ステップａ－ｉ）においてこのデータ伝送メモリが考慮に入れられない場合、ステップａ－ｉ）を実行する速度を高めることができる。
【００２５】
本明細書には、データ処理用の装置であって、少なくとも１つのデータ送信モジュールおよび１つのデータ受信モジュールと、少なくとも３つのデータ伝送メモリと、ポインタメモリとを含み、上記の方法を実行するように構成されている装置も記載される。
【００２６】
上記の方法に関連して述べた特別な利点および設計上の特徴は、以下に述べる装置にも適用可能であり転用可能であることに留意されたい。
さらに、コンピュータによって実行されるときに、コンピュータに上記の方法を実行させる命令を含むコンピュータプログラム製品も記載される。
【００２７】
さらに、本明細書には、コンピュータによって実行されるときに、コンピュータに上記の方法を実行させる命令を含むコンピュータ可読記憶媒体も記載される。
以下、図面に基づいて、本明細書に記載の方法および記載の装置を説明する。
【図面の簡単な説明】
【００２８】
【図１】本明細書に記載の方法を参照した、記載の装置の概略図である。
【図２】記載の方法におけるデータ伝送メモリを予約するプロセスを示す図である。
【発明を実施するための形態】
【００２９】
図１に、データ送信モジュール２およびデータ受信モジュール３が示されており、データ送信モジュール２は、定義済みデータ１をデータ受信モジュール３に伝達する。この目

的のために、合計３つのデータ伝送メモリ４が設けられ、データ伝送メモリ４はそれぞれ、伝送すべき定義済みデータ１を記憶するためのデータメモリブロック５と、データ送信モジュール２のアクセス予約１０を設定するための第１のメモリビット６と、データ受信モジュール３のアクセス予約１０を設定するための第２のメモリビット７とを有する。

【００３０】

データ送信モジュール２では、ステップａ）がステップｉ）～ｖ）で実行される。データ受信モジュール３では、ステップｂ）がステップｉ）～ｉｖ）で実行される。

ステップａ－ｉ）で、まず、第１のメモリビット６も第２メモリビット７も設定されておらず、したがって現在使用されていないデータ伝送メモリ４が決定される。さらなる制限として、ポインタメモリに保存されているポインタが現在指し示していないデータ伝送メモリ４のみが決定される。ステップａ－ｉｉ）で、第１のメモリビット６が設定されることにより、それぞれのデータ伝送メモリ４の予約が行われる。

【００３１】

次いで、ステップａ－ｉｉｉ）に従って、定義済みデータ１が、それぞれのデータ伝送メモリ４のデータメモリブロック５に書き込まれる。これが完了するとすぐに、ステップａ－ｉｖ）で、ポインタメモリでのポインタ９がこのデータメモリブロック５に向けて設定される。図１では、例として、データ伝送メモリ４のポインタ９が、（ｉｉｉ）と記された第３のデータ伝送メモリ４に向けて設定されていることが示されている。ステップａ）の最後に、ステップａ－ｖ）で、それぞれのデータ伝送メモリ４の第１のメモリビット６に設定されたアクセス予約１０を解除することによって、データ伝送メモリ４の解放が行われる。

【００３２】

データ送信モジュール２におけるステップｉ）～ｖ）でのステップａ）の実行と並行して、しかし場合によっては時間をずらして、データ受信モジュール３におけるステップｉ）～ｉｖ）でのステップｂ）が実行される。まず、ステップｂ－ｉ）で、定義済みデータ１の現在のセットが保存されている、読出し対象のデータ伝送メモリ４が決定される。次に、ステップｂ－ｉｉ）で、データ受信モジュール３によって、第２のメモリビット７にアクセス予約１０を設定することによって、読出し対象のデータ伝送メモリ４の予約が行われる。次に、ステップｂ－ｉｉｉ）で、データメモリブロック４からの定義済みデータ１の読出しが行われる。定義済みデータ１の読出しが完了するとすぐに、ステップｂ－ｉ）で設定された第２のメモリビット１でのアクセス予約の解除により、データ伝送メモリの解放が行われる。

【００３３】

図２は、データ伝送メモリの決定および予約のために実行される、ステップａ－ｉ）およびａ－ｉｉ）、またはｂ－ｉ）およびｂ－ｉｉ）によるプロセスを示す。図２は、ステップａ－ｉ）およびａ－ｉｉ）、またはｂ－ｉ）およびｂ－ｉｉ）をどのように実施することができるかについての２つの詳細図も示す。ステップａ－ｉｉｉ）によるデータの書込み、またはステップｂ－ｉｉｉ）によるデータの読出しを妨害なく行うことができるように、それぞれ決定および予約が必要である。

【００３４】

ステップａ－ｉ）およびａ－ｉｉ）、またはｂ－ｉ）およびｂ－ｉｉ）は、データ送信モジュールの書込みアクセスとデータ受信モジュールの読出しアクセスとの衝突が妨げられるように実施される。この目的で、ステップａ－ｉ）およびａ－ｉｉ）、またはｂ－ｉ）およびｂ－ｉｉ）中、それぞれ他のモジュールによるデータ伝送メモリへのアクセスがブロックされる。しかし、このブロックは、３つのデータ伝送メモリのうちのそれぞれ１つに関してのみ、それぞれのデータ伝送メモリがステップａ－ｉｉ）またはｂ－ｉｉ）での予約の候補である限りにおいてのみ適用される。

【００３５】

図２は、データ伝送メモリを決定して予約するために、データ伝送メモリの予約が正常に行われるまで、各データ伝送メモリに対して実行されるプログラムシーケンスのフロー

チャートを示す。フローチャートの各ブロックにおいて、それぞれ２つのメモリビットが「（＿）（＿）」で表され、（ｘ）によって、メモリビットでアクセス予約が認識された場合が表され、（？）によって、メモリビットでアクセス予約をチェックすべき場合が表される。第１の位置「（ｘ）（＿）」は、それぞれ、データ送信モジュールのアクセス予約を記憶するための第１のメモリビットを表し、第２の位置「（＿）（ｘ）」は、データ受信モジュールのアクセス予約を記憶するための第２のメモリビットを表す。

【００３６】

図２でのステップａ－ｉ）およびａ－ｉｉ）、またはステップｂ－ｉ）およびｂ－ｉｉ）の実施は、基本的には同じであるが、チェックはそれぞれ逆に行われる。すなわち、ステップａ－ｉ）およびａ－ｉｉ）の実施時に、第２のメモリビットがチェックされて、データ受信モジュールによるアクセス予約があるかどうかが判断され、逆に、ステップｂ－ｉ）およびｂ－ｉｉ）の実施時に、第１のメモリビットがチェックされて、データ送信モジュールによるアクセス予約があるかどうかが判断される。以下の説明は、図２でのステップａ－ｉ）およびａ－ｉｉ）の表現を参照し、ステップｂ－ｉ）およびｂ－ｉｉ）の表現に転用することができ、その際、第１のメモリビットと第２のメモリビット、およびデータ送信モジュールとデータ受信モジュールがそれぞれ入れ替わることになる。

【００３７】

プロセス開始１１の後、まず、それぞれのデータ伝送メモリに対するデータ受信モジュールによるアクセス予約があるかどうかを判断するために、第２のメモリビットに関する予約チェック１２が開始される。これは、メモリビットの状態「（＿）（？）」で表される。次いで、プログラム分岐１３が行われる。予約がないときには、どのメモリビットも設定されておらず、これは、次に「（＿）（＿）」で示される。データ送信モジュールに関して仮予約１４が行われ、これは、ステップａ－ｉｉ）の第１の部分を成す。仮予約により、第１のメモリビットが設定され、状態は「（ｘ）（＿）」に変わる。予約があり、状態「（＿）（ｘ）」が確認された場合、それぞれのデータ伝送メモリは使用することができない。予約処理の中断１５が行われる。

【００３８】

仮予約１４の後、再度の予約チェック１２が導入される。これは、衝突を妨げるために必要である。仮予約１４の前に、データ送信モジュールとデータ受信モジュールとの間でのプロセス衝突の危険がある。プロセス開始１１と仮予約１４との間で、それぞれ他のプロセスが同様にアクセス予約を行った可能性もある。これを妨げるために、仮予約１４の後に予約チェック１２が再度導入される。ここでの状態は、次に「（ｘ）（？）」となる。次いで、再びプロセス分岐１３が行われる。このプロセス分岐１３の後に状態「（ｘ）（＿）」が確認された場合には、これが最終予約１６として受け入れられる。このプロセス分岐１３の後に状態「（ｘ）（ｘ）」が確認された場合には、第１のメモリビットの状態がリセットされ、次いで、第１の予約チェック前と同様の状態「（＿）（？）」が再び生じる。ここでも、予約処理の中断１５が行われる。

【００３９】

予約処理の中断１５の後、再開始１７が導入され、プロセス開始１１に戻る。これは、データ伝送メモリが決定されるまで行われ、その後、さらなるプロセスステップａ－ｉｉｉ）など、またはｂ－ｉｉｉ）などを実行することができる。

【符号の説明】

【００４０】

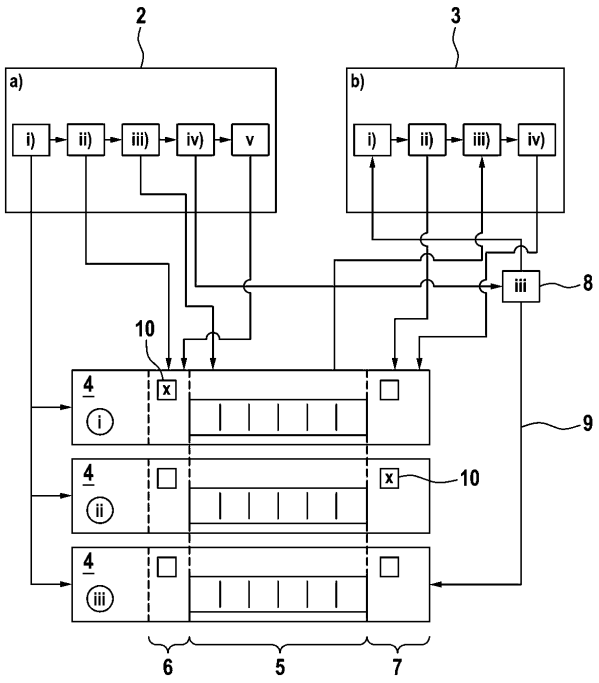
１　定義済みデータ、第２のメモリビット
２　データ送信モジュール
３　データ受信モジュール
４　データ伝送メモリ
５　データメモリブロック
６　第１のメモリビット
７　第２のメモリビット

８　ポインタメモリ
９　ポインタ
１０　アクセス予約
１１　プロセス開始
１２　予約チェック
１３　プログラム分岐、プロセス分岐
１４　仮予約
１５　中断
１６　最終予約
１７　再開始

【図面】

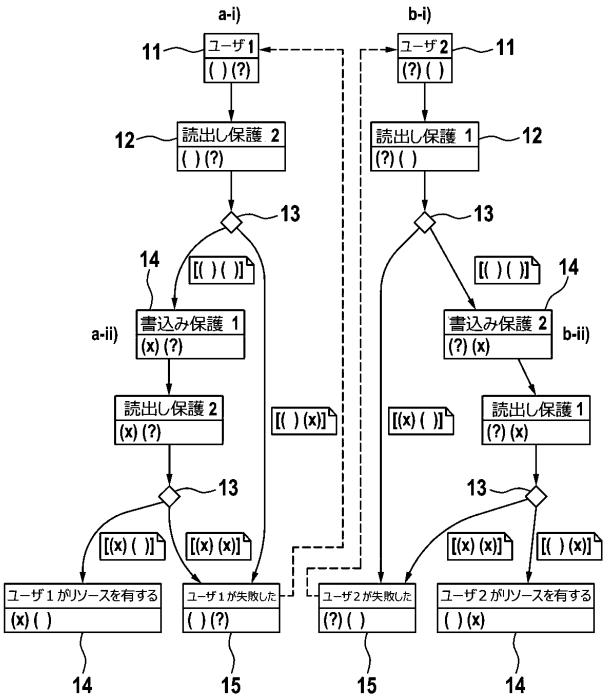
【図１】

Fig. 1



【図２】

Fig. 2



【外国語明細書】
202403964300000４.pdf

フロントページの続き

Ｆターム（参考）　5K030　GA13 HC14 KA03 LA03