

(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2008-193262

(P2008-193262A)

(43) 公開日 平成20年8月21日(2008.8.21)

(51) Int.Cl.
H04L 12/47 (2006.01)F I
H04L 12/47テーマコード (参考)
5K032

審査請求 未請求 請求項の数 8 O L (全 16 頁)

(21) 出願番号 特願2007-23463 (P2007-23463)
(22) 出願日 平成19年2月1日(2007.2.1)(71) 出願人 000005108
株式会社日立製作所
東京都千代田区丸の内一丁目6番6号
(74) 代理人 100122884
弁理士 角田 芳末
(72) 発明者 池 勝久
茨城県ひたちなか市堀口832番地2 株
式会社日立製作所機械研究所内
Fターム(参考) 5K032 AA01 BA08 CA17 CA20 CC01
CD01 DA14 EA07

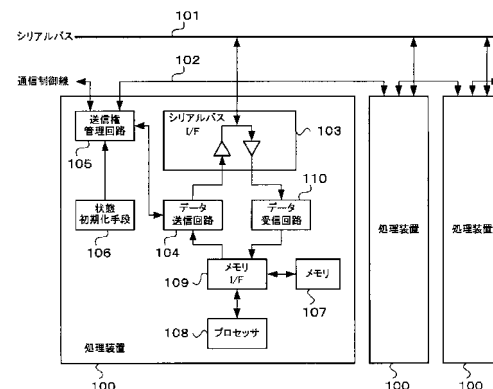
(54) 【発明の名称】 データ通信システム及びデータ通信方法

(57) 【要約】

【課題】 共有データ量によらず通信のワーストレイテンシを小さくし、ソフトウェア負担が小さいデータ通信システムを実現する。

【解決手段】 シリアルバスインタフェース103を介してシリアルバス101に接続する複数の処理装置100において、各処理装置間を順次接続してシリアルバスの送信権の受渡しを行う通信制御線102を設け、通信制御線102に接続して送信権の受け渡しを行うと共に、自処理装置内の送信権の状態を管理する送信権管理手段105と、送信権が有効の場合はデータ送信を開始し、送信権が無効の場合はデータ送信を保留するデータ送信手段104と、初期化時に1台の処理装置の送信権のみを有効とするように、送信権の初期状態を設定する状態初期化手段106を備えるデータ通信システム。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

シリアルバスインタフェースを備えた複数の処理装置を、シリアルバスを介して相互に接続したデータ通信システムであって、

前記処理装置間を順次接続して、シリアルバスへのデータ送信を許可する送信権の受渡しを行う通信制御線を設け、

前記処理装置は、前記通信制御線に接続し、送信権の受け渡しを行うと共に、自処理装置内の送信権の状態を管理する送信権管理手段と、

自処理装置の送信権が有効の場合はシリアルバスへのデータ送信を開始し、送信権が無効の場合はデータ送信を保留するデータ送信手段と、

初期化時に、シリアルバスに接続する複数の処理装置のうち、1 台の処理装置の送信権のみを有効とするように、自処理装置の初期状態を設定する状態初期化手段とを備えたことを特徴とするデータ通信システム。

10

【請求項 2】

請求項 1 記載のデータ通信システムにおいて、

前記送信権管理手段は、隣接する処理装置から前記通信制御線を介して送信権受渡し信号を受信し、自処理装置の送信権を有効化する処理と、

前記データ送信回路の状態を判定し、保留中の送信データが存在しない場合は、送信権受渡し信号を受信した処理装置とは反対側に隣接する処理装置に送信権受渡し信号を送信し、保留中の送信データが存在する場合は、前記データ送信手段のデータ送信を開始し、データの送信完了後に送信権受渡し信号を反対側に隣接する処理装置に送信する処理と、

20

送信権受渡し信号の送信と同時に、自処理装置の送信権を無効化する処理とを行うことを特徴とするデータ通信システム。

【請求項 3】

請求項 1 記載のデータ通信システムにおいて、

前記通信制御線をリング状ではなく、端部の処理装置では一方の通信制御線が開放状態となるように接続し、

前記状態初期化手段は、前記通信制御線が開放状態であることを検出する開放検出手段と、

2 つある通信制御線のうち、第 1 の通信制御線が開放状態であった場合は、初期化において自処理装置の送信権を有効化し、第 1 の通信制御線が開放状態でなかった場合は、初期化において自処理装置の送信権を無効化するように初期状態を設定する送信権初期化手段とから構成し、

30

前記送信権管理手段は、隣接する処理装置に送信権受渡し信号を送信する処理において、一方の通信制御線が開放状態の場合は、他方の通信制御線から送信権受渡し信号を受信した後、送信権受渡し信号を受信した側の通信制御線に折り返し送信権受渡し信号を送信する処理を行うことを特徴とするデータ通信システム。

【請求項 4】

請求項 1 記載のデータ通信システムにおいて、

前記処理装置は、データを格納する記憶手段と、

前記記憶手段へのアクセス要求を出す処理手段と、

前記処理手段から前記記憶手段へのライトアクセス要求があった場合は、前記記憶手段にデータを書き込むと同時に、書込み先のアドレスとデータとを付加したデータ送信要求を前記データ送信手段に送り、前記処理手段から前記記憶手段へのリードアクセス要求があった場合は、前記記憶手段から該当データを読み出して前記処理手段へ渡すインタフェース手段と、

40

シリアルバスに接続する他の処理装置から送信されたデータを受信し、前記記憶手段に格納するデータ受信手段を備えたことを特徴とするデータ通信システム。

【請求項 5】

請求項 4 記載のデータ通信システムにおいて、

50

前記データ受信手段により受信した受信データのアドレス情報を、複数回分保持する受信アドレスバッファと、

前記送信手段が送信するアドレスとデータに基づいて訂正符号を生成し、送信データに付加する符号生成手段と、

前記データ受信手段が受信したアドレスとデータと訂正符号に基づいて通信エラーの有無を判定し、通信エラーが発生しかつ訂正符号を用いたエラー訂正が可能である場合はエラー訂正を実施し、通信エラーが発生しかつエラー訂正が不可能である場合はエラー信号を生成するエラー訂正手段と、

前記エラー訂正手段が生成したエラー信号を受けて、何回前の受信データでエラーが発生したかの情報を含む再送要求コマンドを生成し、前記データ送信手段に送る再送要求生成手段と、

前記データ受信手段が再送要求コマンドを受信した場合、再送が必要なデータが何回前の受信データかを判定し、前記受信アドレスバッファを参照して再送が必要なデータの格納アドレスを取得する再送アドレス生成手段と、

前記再送アドレス生成手段により取得したアドレスを受けて、前記データ送信手段を参照し、取得したアドレスと同じアドレスの再送要求コマンドが送信保留中だった場合、前記データ送信手段に保留されている再送要求コマンドの送信キャンセル要求を出す再送要求キャンセル手段と、

前記再送アドレス生成手段が取得した再送が必要なデータの格納アドレスと、前記受信アドレスバッファの履歴とを参照し、再送要求コマンドを受信した後に、再送対象のアドレスと同一アドレスのデータを受信していない場合、前記記憶手段から該アドレスのデータを読み出し、そのアドレスとデータを再送要求に対する再送データとして生成し、前記データ送信手段に送るデータ再送手段とを備えたことを特徴とするデータ通信システム。

【請求項 6】

シリアルバスインタフェースとデータを格納する記憶手段と前記記憶手段へのアクセス要求を出す処理手段とを備えた複数の処理装置を、シリアルバスを介して接続したデータ通信システムにおいて、

前記処理装置間を順次接続して、シリアルバスへのデータ送信を許可する送信権の受渡しを行う通信制御線を設け、

前記通信制御線を介して接続する他の処理装置との間で、送信権の受け渡しを行うと共に、自処理装置内の送信権の状態を管理する送信権管理処理と、

前記処理手段から前記記憶手段にライトアクセス要求があった場合は、前記記憶手段にデータを書き込むと同時に、書込み先のアドレスとデータとを付加したデータ送信要求を生成する処理と、

自処理装置の送信権が有効の場合はシリアルバスへのデータ送信を開始し、送信権が無効の場合はデータ送信を保留するデータ送信処理と、

シリアルバスに接続する他の処理装置から送信されたデータを受信し、前記記憶手段に格納するデータ受信処理と、

初期化時に、シリアルバスに接続する複数の処理装置のうち、1台の処理装置の送信権のみを有効とするように、自処理装置の初期状態を設定する状態初期化処理とを行うことを特徴とするデータ通信方法。

【請求項 7】

請求項 6 記載のデータ通信方法において、

前記送信権管理処理は、隣接する処理装置から前記通信制御線を介して送信権受渡し信号を受信し、自処理装置の送信権を有効化する処理と、

前記データ送信処理において、保留中の送信データが存在しない場合は、送信権受渡し信号を受信した処理装置とは反対側に隣接する処理装置に送信権受渡し信号を送信し、保留中の送信データが存在する場合は、データ送信を開始し、データの送信完了後に送信権受渡し信号を反対側に隣接する処理装置に送信する処理と、

送信権受渡し信号の送信と同時に、自処理装置の送信権を無効化する処理とを行うこと

10

20

30

40

50

を特徴とするデータ通信方法。

【請求項 8】

請求項 6 記載のデータ通信方法において、

前記データ受信処理により受信した受信データのアドレス情報を、複数回分保持する受信アドレスバッファを設け、

データ送信時に、送信するアドレスとデータに基づいて訂正符号を生成し、送信データに付加する符号生成処理と、

データ受信時に、受信したアドレスとデータと訂正符号に基づいて通信エラーの有無を判定し、通信エラーが発生しかつ訂正符号を用いたエラー訂正が可能である場合はエラー訂正を実施し、通信エラーが発生しかつエラー訂正が不可能である場合はエラー信号を生成するエラー訂正処理と、

前記エラー訂正処理が生成したエラー信号を受けて、何回前の受信データでエラーが発生したかの情報を含む再送要求コマンドを生成し、データ送信要求を行う再送要求生成処理と、

データ受信時に、再送要求コマンドを受信した場合、再送が必要なデータが何回前の受信データかを判定し、前記受信アドレスバッファを参照して再送が必要なデータの格納アドレスを取得する再送アドレス生成処理と、

前記再送アドレス生成処理により取得したアドレスを受け、取得したアドレスと同じアドレスの再送要求コマンドが送信保留中だった場合、保留されている再送要求コマンドの送信キャンセルを行う再送要求キャンセル処理と、

前記再送アドレス生成処理にて取得した再送が必要なデータの格納アドレスを基に、前記受信アドレスバッファの履歴を参照し、再送要求コマンドを受信した後に、再送対象のアドレスと同一アドレスのデータを受信していない場合、前記記憶手段から該アドレスのデータを読み出し、そのアドレスとデータを再送要求に対する再送データとして生成し、データ送信要求を行うデータ再送処理とを行うことを特徴とするデータ通信方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、シリアルバスで接続された複数の分散コントローラ間において、リアルタイムにメモリデータを共有するためのデータ通信システム及びデータ通信方法に関する。

【背景技術】

【0002】

ロボットなどのように、センサやアクチュエータなど多数のデバイスが実装され、それらを使った複雑な制御を必要とする組込み制御機器では、それら複数のデバイスをいくつかの分散コントローラに分割して管理し、相互に通信しながら協調制御する構成が多く採用されている。こうした分散制御型の構成が採用される理由はいくつかあるが、組込み制御として特に重要なのは、配線集中の緩和による機器の小型化と、配線距離の短縮に伴うノイズ問題の改善及び信頼性向上である。

【0003】

多数のデバイスを単一のコントローラで制御する構成では、各デバイスと接続するための信号線をコントローラ部分に集中配線しなければならないため、コントローラの周辺に多数の信号線を集中して取り回すための広い空きスペースを確保しなければならない。更に、太く束ねた配線束を機器末端部から中央のコントローラ部まで引き回すための配線スペースが必要になる。そのため、集中制御型の構成では、機器の小型化が難しい。

【0004】

また、アクチュエータ駆動線のような大電流配線からは、周囲に大きなノイズを放射する場合がある。そのため、センサ等の微小アナログ信号線とアクチュエータ駆動線のような大電流配線を機器末端部からコントローラ部まで長距離配線すると、大電流配線から放射されたノイズがアナログ信号線に混入し誤動作の要因となる場合があり、機器の信頼性が低下する。

【0005】

こうした理由から、機器の部位毎にコントローラを分散し、特定の部位に属するデバイスは、デバイス近くに設置したコントローラに配線して管理する構成が採用されている。また、複数の部位にまたがったデバイスの情報を使って協調制御する際は、各コントローラ間を互いに接続したシリアルバスを介して相互にデータを受け渡す方法が多く採られている。このような分散コントローラシステムでは、コントローラ相互のデータ共有をリアルタイムで実現する必要があるため、どのような方法でデータ通信を行うかが重要である。

【0006】

現在、最も普及している通信方式の一つに、イーサネット（登録商標）（商標：Ethernet（登録商標）、IEEE802.3uとして100BASE-TXという規格が標準化されている）などを利用したLAN（Local Area Network）がある。イーサネット（登録商標）は、CSMA/CD（Carrier Sense Multiple Access/Collision Detection）方式を採用しているため、通信データ量が増えるとデータ衝突が多発してバスの利用効率が低下するほか、通信遅れ時間（レイテンシ）のワースト値が予想不能であるという問題がある。このため、イーサネット（登録商標）は、組込み制御用の通信手段としては不適當である。また、データ衝突の問題を解決したLANの通信制御方式の一つにトークンリング（Token Ring、IEEE 802.5で標準化された規格）がある。「トークン」と呼ばれる送信権を表すデータがネットワークを流れていて、データを送信したい端末は、トークンを取り込んでデータを送信し、送信終了後、再びトークンをネットワークに流す。これにより、一度に一台の端末のみネットワークを利用するように制御し、データ衝突の問題を解決している。ところが、トークンはある長さを持ったデータ列であり、通信データと同じ線に流れるため、通信路に負担がかかる。更に、データはネットワーク上を順番に伝送されるため、通信レイテンシが大きいという問題がある。

【0007】

また、イーサネット（登録商標）、トークンリング共に、データ受渡し先の指定やデータ送信処理、応答処理やエラー訂正処理などを全てソフトウェアで管理しているため、プロセッサの負担が大きい。そのため、組込み制御機器などのように、機器本体のハードウェアとコントローラなどに組み込むソフトウェアとを同時並行開発する場合は、障害発生時の原因の切り分けが難しい。

【0008】

特許文献1には、上記問題点を解決するために、メモリを搭載した複数のコントローラ（局）をシリアルバスで接続し、各局が保持するメモリデータを周期的に他局のメモリの該当領域に転写するハードウェアを実装することにより、複数局間でメモリデータを共有する複数局メモリデータ共有システムの例についての開示がある。非特許文献1には、複数局メモリデータ共有システムに関する製品実施例についての開示がある。

【特許文献1】特開2002-73584号公報

【非特許文献1】株式会社ステップテクニカ 技術資料（STD-CU40-V2.1J）

「C u n e t ユーザーズマニュアル(MKY40)」

【発明の開示】

【発明が解決しようとする課題】

【0009】

上記従来の方法では、シリアルバスの通信処理をハード化することでソフトウェアの負担が皆無になり、なおかつメモリデータの転写周期が定まることにより通信レイテンシが明確になる。ところが、複数局間で共有する全メモリデータを周期的にシリアルバス上に送信し、全ての局のメモリデータを一致させる方法のため、ある局のメモリに書き込んだデータを他局のメモリに反映させるための遅れ時間（レイテンシ）は、データを共有するメモリの容量に比例して増大するという問題がある。

【0010】

本発明はかかる点に鑑みてなされたものであり、（1）共有データ量の多少に影響を受けず通信のワーストレイテンシを小さく実現すること、（2）プロセッサのソフトウェア

負担が小さいこと、(3)配線量が少なく、省スペース実装に有利なシリアル接続方式とすること、を同時に実現したデータ通信システムを実現することを目的とする。

【課題を解決するための手段】

【0011】

本発明は、シリアルバスインタフェースを備えた複数の処理装置を、シリアルバスを介して相互に接続したデータ通信システムであって、各処理装置間を順次接続して、シリアルバスへのデータ送信を許可する送信権の受渡しを行う通信制御線を設け、処理装置には、通信制御線に接続し、送信権の受け渡しを行うと共に、自処理装置内の送信権の状態を管理する送信権管理手段と、自処理装置の送信権が有効の場合はシリアルバスへのデータ送信を開始し、送信権が無効の場合はデータ送信を保留するデータ送信手段と、初期化時に、シリアルバスに接続する複数の処理装置のうち、1つの処理装置の送信権のみを有効とするように、自処理装置の初期状態を設定する状態初期化手段とを備える。

10

【発明の効果】

【0012】

本発明によると、シリアルバスで繋がれた複数の処理装置の間に通信制御線を設けて送信権の受渡しを行うことにより、シリアルバスのデータ通信に負担をかけることなく、シリアルバスへのデータ送信タイミングをハード的に制御する手段を提供することができる。これにより、バス利用効率低下の原因となるデータ衝突を回避することができ、データ通信速度が向上すると同時に、通信のワーストレイテンシを小さくすることができる。

【0013】

20

また、処理装置にプロセッサおよびプロセッサが読み書きするメモリを追加し、メモリへのデータ書込み時に、シリアルバスを介して接続する全処理装置のメモリへデータ変更を反映させる手段を設けることにより、ソフトウェアを介することなくメモリデータを自動的に共有化することができる。更に、データを変更する時だけ変更対象のアドレスとデータを送信するため、共有メモリサイズの増加によるレイテンシへの影響が無い。またシリアルバスの利用効率が高く、転送速度が速くなるため、大容量データの転送に好適である。

【0014】

更に、本発明において、1つの処理装置が1回に変更できるデータサイズに制約を設けた(例えば1データ)とすると、シリアルバスに接続する各処理装置にはデータ変更の機会が均等に訪れる。1つの処理装置で大量のデータ変更を行う場合は、待ちが発生してレイテンシが増加する場合があるが、増加するレイテンシのワースト値は、その処理装置が行おうとしている処理に関する情報だけに基づいて予測できる。そのため、1データだけ変更するよう制約を設けた場合のワーストレイテンシは、シリアルバスに接続した装置台数だけに依存することになるため、算出可能である。

30

【0015】

また、本発明によると、通信エラーの検出と訂正、および再送機能をハードウェアで実現することができる。本発明では、データを変更した処理装置がシリアルバスにデータを送信すると、送信した処理装置以外の全ての処理装置が同時にデータを受信する。データを受信した処理装置は、個別にエラー検出／訂正を実施するが、訂正しきれないエラーを検出した場合は、送信権が巡ってきた時に再送要求コマンドを送信する。一方、正常にデータを受信できた処理装置は全て正常なデータを保持しているので、再送要求コマンドが送信された後、最初に送信権が巡ってきた処理装置(変更データを送信した装置でなくて良い)が再送を実施することができる。また、受信データが訂正不能の場合であっても、自処理装置に送信権が巡ってくる以前に、別の処理装置により該データの再送要求が送出された場合は、同じデータに対して重複して再送要求することはせず、他の処理装置から再送データが送られるのを待つ。これにより、本来であればソフトウェアの負担となるエラー検出／訂正処理や再送処理を、全てハードウェアに任せることができる。また、エラー検出／訂正処理や再送処理は処理装置が個別に行うが、シリアルバスに接続する全ての処理装置が連携して行うため、再送処理などの重複がなく、迅速に処理することができる。

40

50

。また、データ再送によるワーストレイテンシの増加量も明確に算出することが可能となる。

【発明を実施するための最良の形態】

【0016】

以下、本発明の一実施の形態を、添付図面を参照して説明する。

【0017】

図1に、本例のデータ通信システムの構成例を示す。本例は、複数の処理装置100が、シリアルバスインタフェース(I/F)103を介してシリアルバス101に接続し、各処理装置100の間は、シリアルバス101のデータ送信権の受け渡しを行う通信制御線102により順次つないで構成する。図1では、通信制御線102の端部は開放状態となっ

10

ているが、開放部の通信制御線を互いに接続し、リング状として構成することも可能である。

【0018】

次に、図1を参照して処理装置100の構成について説明する。なお、シリアルバス101に接続する処理装置はすべて同じ構成であるため、1つの処理装置の構成を説明し、他の処理装置の説明は省略する。処理装置100は、シリアルバス101に接続するためのシリアルバスI/F103と、通信制御線102に接続し、送信権の受け渡しを行うと共に、自処理装置内の送信権の状態を管理する送信権管理手段である送信権管理回路105と、初期化時に自処理装置の送信権の初期状態を設定する状態初期化手段106と、シリアルバスI/F103を介してデータを送信するデータ送信手段であるデータ送信回路104と、シリアルバスI/F103を介してデータを受信するデータ受信手段であるデータ受信回路110と、データを格納する記憶手段であるメモリ107と、メモリ107へのアクセス要求を出す処理手段であるプロセッサ108と、プロセッサ108からのアクセス要求を受け、メモリ107へのデータの入出力を行うメモリインタフェース(I/F)109から構成する。

20

【0019】

次に、本例により、分散型メモリデータ共有システムを構成した場合のメモリデータの共有処理の動作例について説明する。本例を適用して分散型メモリデータ共有システムを構成した場合、シリアルバスに接続する全ての処理装置に搭載するメモリは、すべて同じ内容になる。また、全ての処理装置は、メモリの全領域を読み書きすることが可能であり、メモリの内容を変更した場合は、本例の通信処理により他の処理装置内のメモリへ変更内容を反映する。

30

【0020】

まず、プロセッサ108が、メモリ107のデータを読み出す場合の処理について説明する。プロセッサ108が、メモリ107のデータを読み出す際は、メモリI/F109に対して読み出したいメモリのアドレスを指定し、データの読み出し要求を出す。メモリI/F109は、指定されたアドレスのデータをメモリ107から読み出し、読み出したデータをプロセッサ108に渡す。

【0021】

次に、図2を参照して、プロセッサ108が、メモリ107へデータを書き込む場合の処理について説明する。図2は、プロセッサ108がメモリ107へデータを書き込む場合の、メモリへのデータ書き込み処理例を示すフローチャートである。

40

【0022】

プロセッサ108が、メモリ107へデータを書き込む際は、まず、メモリI/F109に対して書き込みたいデータとそのアドレスを指定し、データの書き込み要求を出す(ステップS201)。メモリI/F109は、メモリ107の指定されたアドレスへデータを書き込み(ステップS202)、メモリ107へ書き込んだアドレスとデータをデータ送信回路104に渡し、シリアルバス101への送信を要求する(ステップS203)。データ送信回路104は、メモリI/F109からアドレスとデータを受け取ると、送信権管理回路105が保持する自処理装置のシリアルバス送信権の状態を判定し(ステッ

50

プ S 2 0 4)、送信権が有効である場合は、シリアルバス I / F 1 0 3 にアドレスとデータを渡し、データ送信を要求する(ステップ S 2 0 5)。シリアルバス I / F 1 0 3 は、データ送信回路 1 0 4 から送信要求を受けると、シリアルバス 1 0 1 ヘデータを送信する(ステップ S 2 0 6)。ステップ S 2 0 4 の判定の結果、送信権が無効である場合は、シリアルバス 1 0 1 への送信を一時保留する(ステップ S 2 0 7)。送信を保留した場合は、送信権が有効になった時点で、送信を開始する。なお、送信権の状態を管理する方法と、保留した送信データの送信方法については後述する。

【0023】

次に、データの受信処理について説明する。シリアルバス 1 0 1 に接続する処理装置 1 0 0 からデータが送信されると、シリアルバス 1 0 1 に接続する他の全ての処理装置がそのデータを受信する。ある処理装置から送信されたデータは、シリアルバス 1 0 1 を介して他の処理装置 1 0 0 のシリアルバス I / F 1 0 3 へ入力され、データ受信回路 1 1 0 へ受け渡される。データ受信回路 1 1 0 は、他の処理装置 1 0 0 が送信したアドレスとデータをシリアルバス I / F 1 0 3 から受け取り、そのデータを基にメモリ I / F 1 0 9 にメモリ 1 0 7 へのデータの書き込みを指示する。メモリ I / F 1 0 9 は、メモリ 1 0 7 の指定されたアドレスヘデータを書き込む。

【0024】

これらの動作により、ある処理装置のプロセッサ 1 0 8 が変更したメモリデータは、その都度シリアルバス 1 0 1 に接続された他の全処理装置のメモリに反映され、変更データを共有することができる。

【0025】

次に、送信権管理回路 1 0 5 による送信権の管理方法について説明する。送信権管理回路 1 0 5 は、隣接する両隣の処理装置内の送信権管理回路 1 0 5 とそれぞれ双方向の通信制御線 1 0 2 でつながっているため、送信権管理回路 1 0 5 には、通信制御線 1 0 2 が 2 本接続することになる。送信権管理回路 1 0 5 は、通信制御線 1 0 2 を介して送信権受け渡し信号のやりとりをすることにより、順番に送信権を受け渡していく。

【0026】

最初に、送信権管理回路 1 0 5 は、状態初期化手段 1 0 6 の働きにより、同じシリアルバス 1 0 1 に接続する複数の処理装置 1 0 0 の中で、1 台だけシリアルバス送信権が有効になるように初期化される。初期化後は、送信権が有効となった処理装置から順番に送信権受け渡し信号を受け渡していく。

【0027】

状態初期化手段 1 0 6 の処理としては、例えば通信制御線 1 0 2 をリング状に接続している場合、予め任意の 1 台の処理装置の送信権を有効とするような送信権の初期値をメモリなどの記憶装置へ設定しておき、状態初期化手段 1 0 6 はその初期値を参照して初期状態を設定するように処理してもよい。また、通信制御線 1 0 2 がリング状ではなく、端部の処理装置では一方の通信制御線が開放状態となっている場合、状態初期化手段 1 0 6 では、通信制御線が開放状態であるかを検出し、2 つある通信制御線のうち、第 1 の通信制御線が開放状態であった場合は、初期化時の送信権の状態を有効とし、第 1 の通信制御線が開放状態でなかった場合は、初期化時の送信権の状態を無効とするように初期状態を設定してもよい。このように設定することで、初期化時に 1 台の処理装置のみ送信権を有効にすることができる。

【0028】

次に、図 3 を参照して、初期化後の送信権の管理方法と、保留した送信データの送信方法について説明する。図 3 は、送信権管理回路 1 0 5 の送信権管理処理例を示すフローチャートである。

【0029】

まず、送信権管理回路 1 0 5 は、第 1 の通信制御線 1 0 2 に接続する隣接処理装置から送信権受け渡し信号を受け取り(ステップ S 3 0 1)、自処理装置の送信権の状態を有効とする(ステップ S 3 0 2)。送信権が有効となった送信権管理回路 1 0 5 は、データ送

信回路104の状態を参照して送信保留データがあるかを判定し（ステップS303）、送信が保留されているデータがある場合は、データ送信回路104へ保留中のデータの送信開始を指示し（ステップS304）、データの送信完了を待つ（ステップS305）。データ送信完了後、送信権を無効化し（ステップS306）、第2の通信制御線102に接続する隣接処理装置へ送信権受け渡し信号を送出する（ステップS307）。ステップS303の送信保留データ有無の判定の結果、送信保留データが無い場合は、すぐに自処理装置の送信権を無効化し（ステップS306）、送信権受け渡し信号を送出する（ステップS307）。

【0030】

本例によるデータ通信時、シリアルバス101と通信制御線102に流れる信号のタイミング例を図4に示す。ここでは、シリアルバス101上に3つの処理装置（A～C）が接続しているものとし、各処理装置のプロセッサ108をプロセッサA～Cと呼ぶ。また、各処理装置間を接続する通信制御線は、リング状ではなく、端部の処理装置、ここでは処理装置AとCは、通信制御線の片方が開放状態となっている例を示す。

【0031】

図中の波形（1）～（3）は、プロセッサA～Cのメモリアクセスタイミングを示し、“RD”はメモリデータの読み出し、“WT”はメモリへのデータ書き込みを表している。（4）、（5）は、処理装置A～B間とB～C間の通信制御線の信号の動きを示している。（4）の最初の信号は、処理装置AからBへ送信権受け渡し信号を送出したことを表し、（5）の最初の信号は、処理装置BからCへ送信権受け渡し信号を送出したことを表している。また、（6）は、シリアルバス送信権が有効となっている処理装置の遷移状況であり、（4）、（5）で示される送信権受け渡し信号の送出に伴って、送信権が処理装置A→B→Cの順番に受け渡されていることが分かる。（7）～（9）は、処理装置A～Cがそれぞれシリアルバス101に出力する信号を、（10）は（7）～（9）を合成したもの、すなわちシリアルバス101に流れるデータを示す。（11）は、時間を表す。

【0032】

時間t1～t2では、波形（1）～（3）の最初の部分に示すように、プロセッサAとプロセッサCのリードアクセスタイミングが重なっているが、リードアクセスはそれぞれの処理装置内で完結する構造となっているため、互いに干渉は起きない。またこの時、どの処理装置もシリアルバス101に送信すべきデータを持っていないため、バス送信権はA→B→C→A→B→Cと待ち時間無しで速やかに巡回している。

【0033】

なおこの実施例では、処理装置AがA～B間の送信権受け渡し信号を出力することでA→Bに、処理装置BがB～C間の送信権受け渡し信号を出力することでB→Cに送信権が移動するが、処理装置Cは通信制御線の片方が開放状態となっていて、隣接する処理装置が存在しないため、B～C間の送信権受け渡し信号を出力することで、逆方向（C→B方向）に送信権を移動する。送信権を逆方向へ移動する際、途中にある処理装置（B）は自処理装置の送信権は有効とせず、反対側に隣接する処理装置へ送信権受け渡し信号を送出し、他方の端部に位置する処理装置（A）のところまで送信権を戻す。その後、再び処理装置（A）の送信権を有効とし、順次送信権を受け渡す。このように処理することにより、A→B→C→A→B→Cの順に送信権が巡回し、端部に位置する処理装置とその他の処理装置について、送信権を有効化する機会を均等化することができる。

【0034】

図4において、最初にメモリ書き込み要求を出すのはプロセッサCである。プロセッサCが書き込み要求（C1）を出した時間t3は、バス送信権を処理装置BからAへ受け渡すために送信権受け渡し信号が送出されたタイミングであり、処理装置Cは送信権が有効ではないため、データ送信を保留する。処理装置Cは、次にバス送信権が巡ってくる時間t4にシリアルバス101にライトデータ（C1）を出力する。プロセッサCはこれに続いてC2、C3、C4と連続して書き込み要求を出す、送信権が有効の時に送信できるデータは1データだけに制限してあるため、C1データの送信終了後、ライトデータ（C

10

20

30

40

50

2) 以降の送信は保留し、処理装置 A にバス送信権を移す。

【0035】

一方、ライトデータ (C 1) がシリアルバス 1 0 1 に出力されている間、プロセッサ A と B もそれぞれメモリ書き込み要求 A 1 と B 1 を要求している。ライトデータ (C 1) が出力されている間は処理装置 C に送信権があるため、自処理装置に送信権が巡ってくるのを待って出力する。処理装置 A は、時間 t 6 に送信権が有効になると、ライトデータ (A 1) をシリアルバス 1 0 1 に送信し、処理装置 B は、時間 t 7 に送信権が有効になると、ライトデータ (B 1) を送信する。時間 t 8 にライトデータ (B 1) の出力が終了すると、処理装置 C の送信権が有効になり、送信保留していたライトデータ (C 2) を送信する。C 2 の送信終了後、ライトデータ (C 3) 以降の送信は保留し、バス送信権を移す。そして、再び送信権が巡ってきた時点で次のライトデータを送信する。

10

【0036】

このように、送信権が有効の時に送信できるデータの数制限することにより、1 台の処理装置が送信権を長時間独占し、他の処理装置の送信タイミングが遅れることを防いでいる。

【0037】

本例における通信遅れ時間 (レイテンシ) のワーストケースは、上記の処理装置 C におけるライトデータ (C 2) の送信のような場合で、自処理装置が書き込み要求を連続して発生し、同時に他の処理装置も書き込み要求を発生するため、自処理装置を含む全ての処理装置がデータ送信後に送信権を受け渡し、送信権が一巡した後にデータ送信するケースである。図 4 では、時間 t 5 で書き込み要求 (C 2) を発生してから t 9 でデータ送信が完了するまでの時間がレイテンシの最大値となる。その長さは、シリアルバス 1 0 1 に接続する処理装置の数と、一度に送信が許されるデータの長さによって定まる。

20

【0038】

例えば、シリアルバスの通信速度が 1 0 M b p s の場合で、処理装置が 1 回に送信する 1 データが (スタートビット + 8 b i t データ + ストップビット) の構成だった場合、1 b i t 周期が 1 0 0 n s となるため、

$$\begin{aligned} 1 \text{ データ周期} &= (\text{スタートビット} + 8 \text{ b i t データ} + \text{ストップビット}) \times 1 0 0 \text{ n s} \\ &= 1 \mu \text{ s} \end{aligned}$$

となる。図 4 では、シリアルバス 1 0 1 に接続する処理装置は 3 台のため、レイテンシの最大値は、 $(3 \mu \text{ s} + \alpha)$ となる。なお、 α は、本例では、処理装置 C から A へ送信権を受け渡す際の送信権受け渡し信号の通信時間であり、この時間はすべての処理装置の送信権が無効になる時間である。このような無効時間は、システムの構成により異なるが、各構成について固定となるため、それぞれのレイテンシの最大値は算出可能である。

30

【0039】

次に、本例の他の実施の形態による、通信エラーの検出／訂正手段を有するデータ通信システムについて説明する。図 5 に通信エラー検出／訂正機能を追加したデータ通信システムの構成例を示す。なお、図 1 に示したデータ通信システムと同じ構成には同じ符号を付け、説明を省略する。

【0040】

本例では、通信エラーの検出／訂正機能を実現するため、シリアルバス 1 0 1 に接続する処理装置 1 0 0 に、データ受信回路 1 1 0 により受信したデータとアドレス情報を、複数回分保持する受信アドレスバッファ 1 1 4 と、送信するデータを基に訂正符号を生成し、送信データに付加してデータ送信回路 1 0 4 へ渡す符号生成手段である符号生成回路 1 1 1 と、データ受信回路 1 1 0 が受信したデータに付加された訂正符号に基づいてエラーの有無を検査し、エラー訂正を行うエラー訂正手段であるエラー訂正回路 1 1 2 と、エラー訂正回路 1 1 2 にて訂正できない通信エラーに対し、再送要求コマンドを生成し、データ送信要求を行う再送要求生成手段である再送要求生成回路 1 1 3 を設ける。

40

【0041】

また、再送要求コマンドを受信したときのデータの再送機能を実現するため、データ受

50

信回路 110 が受信した再送要求コマンドを基に、再送が必要なデータの格納アドレスを取得する再送アドレス生成手段である再送アドレス生成回路 115 と、再送アドレス生成回路 115 から指示されたアドレスを基にメモリからデータを読み出し、再送データを生成して符号生成回路 111 に送り、データ送信要求を行うデータ再送手段である再送回路 116 とを設ける。更に、複数の処理装置において同じデータの通信エラーが発生した場合に、重複してデータを再送することを避けるため、再送要求キャンセル手段である再送要求キャンセル回路 117 を設ける。

【0042】

次に、本例を構成する各回路の動作例について説明する。

【0043】

符号生成回路 111 は、シリアルバス 101 へ送信されるデータを入力し、そのデータを基に訂正符号を生成し、データに付加してデータ送信回路 104 に送る。例えば、メモリへの書き込みデータを送信する場合、メモリ I/F 109 から送られたデータに訂正符号を付加してデータ送信回路 104 に送る。

【0044】

データ受信回路 110 は、受信したデータがメモリデータの場合、そのメモリアドレスを受信アドレスバッファ 114 に保存するとともに、受信データをエラー訂正回路 112 へ渡す。受信アドレスバッファ 114 は、リングバッファであり、過去数回分の受信データの受信履歴を保存し、何回前に受信したデータかという情報に基づいて、そのデータのメモリ上の格納アドレスを参照できるようにしたものである。また、データ受信回路 110 は、受信したデータが再送要求コマンドの場合、受信した再送要求コマンドのデータを再送アドレス生成回路 115 へ渡す。

【0045】

エラー訂正回路 112 は、データ受信回路 110 から受信したデータを入力し、受信データに付加された訂正符号に基づいて通信エラーの有無を検査する。検査の結果、通信エラーが発生しかつ訂正符号を用いたエラー訂正が可能である場合は、エラー訂正を実施し、メモリ I/F 109 にメモリ 107 へのデータ書き込みを指示する。訂正符号では訂正できない通信エラーが発生した場合は、再送要求生成回路 113 に再送要求コマンドの生成を指示する。

【0046】

再送要求生成回路 113 は、エラー訂正回路 112 から訂正失敗情報とその後のデータ受信情報を受け取り、訂正不可のデータが何回前に送信されたデータであるかの情報を付加した再送要求コマンドを生成し、データ送信回路 104 に送る。データ送信回路 104 は、送信権が有効となるのを待って再送要求コマンドを送信する。このとき、送信権が有効になる前に新たなデータを受信する場合があるため、データ受信の度にエラー訂正回路 112 からデータ受信情報を受け取り、送信保留中の再送要求コマンドに付加した何回前に送信されたデータであるかの情報を更新する。

【0047】

再送アドレス生成回路 115 は、データ受信回路 110 が受信した再送要求コマンドを入力し、再送すべきメモリデータのアドレスを生成する。再送要求コマンドには、再送要求コマンドが送信される時点より何回前にシリアルバスに送信されたデータが再送対象であるかという情報が付加されているため、その情報に基づいて受信アドレスバッファ 114 を参照し、再送対象であるデータのメモリ上の格納アドレスを取得する。取得した格納アドレスを再送回路 116 へ送り、再送を指示する。一方、受信した再送要求コマンドと同じデータを対象とした再送処理を既に実施していた場合、再送処理の重複を避ける必要がある。そのため、再送要求キャンセル回路 117 に対して、受信した再送要求コマンドとアドレスを渡し、キャンセル処理を指示する。

【0048】

再送回路 116 は、まず、再送アドレス生成回路 115 から指示された格納アドレスを基に受信アドレスバッファ 114 を参照し、再送要求コマンドを受信した後に再送要求を

10

20

30

40

50

受けたアドレスと同一アドレスのデータを受信していないか確認する。同じアドレスのデータを受信していない場合は、メモリ I / F 109 を介してメモリ 107 からデータを読み出し、再送データとして符号生成回路 111 に送る。符号生成回路 111 は、送られた再送データに対して、通常のメモリデータ送信と同様に訂正符号を付加し、データ送信回路 104 へ送る。データ送信回路 104 は、送信権の有効化を待って再送データを送信する。再送回路 116 が、再送要求コマンドの受信後に再送対象のデータと同じアドレスのデータを受信したことを検出した場合は、データの再送を中止する。これにより、同一データに対する再送データをシリアルバスへ複数送出することを避けることができる。

【0049】

再送要求キャンセル回路 117 は、再送アドレス生成回路 115 から再送要求コマンドとアドレスを受け取ると、再送要求生成回路 113 を参照し、同じデータを対象とした再送要求コマンドが送信保留状態になっているかを確認する。具体的には、再送要求生成回路 113 が生成した再送要求コマンドと、再送アドレス生成回路 115 が生成した再送アドレス情報を比較し、再送の対象となるメモリアドレスが一致した場合に同一データと判断する。その結果、同一データの再送要求コマンドが保留状態になっている場合は、再送要求生成回路 113 に再送要求コマンドのキャンセルを指示する。更に、再送回路 116 に対し、受信した再送要求コマンドに対応する再送処理をキャンセルするよう指示する。これにより、同一データに対する再送要求をシリアルバスへ複数送出することを避けることができる。また、自処理装置において、通信エラーにより正しく受信できなかったデータに関する再送処理を中止することで、誤ったデータを再送してしまうことを防ぐことができる。

【0050】

次に、本例による通信エラーの検出／訂正処理について、図 6、図 7 を参照して説明する。まず、図 6 を参照してデータ受信時の処理について説明する。図 6 は、本例のデータ受信処理例を示すフローチャートである。

【0051】

まず、データ受信回路 110 がデータを受信する（ステップ S 601）。受信したデータがメモリデータか否かを判定し（ステップ S 602）、メモリデータの場合は、受信アドレスバッファ 114 にメモリアドレスを格納する（ステップ S 603）。次に、エラー訂正回路 112 は、データ受信回路 110 が受信したデータを入力し、訂正符号に基づいて通信エラーの有無を検査する（ステップ S 604）。検査の結果、通信エラーがないかを判定し（ステップ S 605）、通信エラーなしの場合は、メモリ I / F 109 を介してメモリ 107 へデータを書き込み（ステップ S 606）、処理を終了する。ステップ S 605 の判定の結果、通信エラーがある場合は、訂正符号による訂正が可能かを判定し（ステップ S 607）、訂正可能な場合は、エラー訂正回路 112 にて訂正符号による訂正を実施し（ステップ S 608）、メモリへデータを書き込み（ステップ S 606）、処理を終了する。

【0052】

ステップ S 607 の判定の結果、訂正符号による訂正が不可能な場合は、エラー訂正回路 112 から訂正失敗情報とその後のデータ受信情報を再送要求生成回路 113 へ受け渡し、再送要求生成回路 113 にて再送要求コマンドを生成する（ステップ S 609）。再送要求生成回路 113 は、生成した再送要求コマンドをデータ送信回路 104 へ送り、再送要求コマンドを送信する（ステップ S 610）。

【0053】

ステップ S 602 の判定の結果、受信したデータがメモリデータでなかった場合は、受信データが再送要求コマンドか否かを判定し（ステップ S 611）、再送要求コマンドの場合は、再送処理を実行する（ステップ S 612）。ステップ S 611 の判定の結果、再送要求コマンドでなかった場合は、処理を終了する。

【0054】

次に、図 7 を参照してデータの再送処理について説明する。図 6 に示したフローチャー

トにおいて、データ受信回路 110 が受信したデータが再送要求コマンドだった場合、ステップ S 612 にて再送処理を実行する。図 7 は、ステップ S 612 の再送処理の詳細な処理例を示すフローチャートである。

【0055】

まず、データ受信回路 110 は、受信した再送要求コマンドを再送アドレス生成回路 115 へ入力する（ステップ S 701）。再送アドレス生成回路 115 は、再送要求コマンドに付加された、再送対象データは何回前にシリアルバスに送信されたデータか、の情報に基づいて受信アドレスバッファ 114 を参照し、再送対象であるデータのメモリ上の格納アドレスを取得する（ステップ S 702）。次に、データ再送回路 116 は、再送アドレス生成回路 115 から再送対象データの格納アドレスを入力し、そのアドレスと受信アドレスバッファ 114 の履歴を比較して、再送要求コマンドを受信した後に再送対象データのアドレスと同一アドレスのデータを受信していないかを確認する（ステップ S 703）。確認の結果、同一アドレスのデータを受信していない場合は、メモリ I/F 109 を介してメモリ 107 から当該アドレスのデータを読み出し（ステップ S 704）、再送データとして符号生成回路 111 に送り、データを送信する（ステップ S 705）。データの送信は、通常のメモリデータの送信と同様に訂正符号を付加し、送信権の有効化を待って送信する。ステップ S 703 の確認の結果、再送対象のデータと同じアドレスのデータを受信したことを検出した場合は、データは再送しない。

【0056】

次に、再送アドレス生成回路 115 から、再送要求コマンドとアドレスを再送要求キャンセル回路 117 へ受け渡し、キャンセル処理を行う。再送要求キャンセル回路 117 は、再送アドレス生成回路 115 から受信した再送要求コマンドと再送対象データの格納アドレスを受け取り、再送要求生成回路 113 を参照して、同じデータを対象とした再送要求コマンドが送信保留状態になっているかを確認する（ステップ S 706）。確認の結果、同じデータの再送要求コマンドが送信保留になっている場合は、再送要求生成回路 113 に再送要求コマンドのキャンセルを指示し、再送要求をキャンセルする（ステップ S 707）。次に、再送回路 116 に対し、受信した再送要求コマンドに対応した再送処理をキャンセルするよう指示する（ステップ S 708）。

【0057】

このように、本例のエラー検出／訂正機能は、データ送信時にエラー訂正符号を付加し、データを受信した処理装置がそれぞれエラー訂正符号を使ったエラーチェックを行うものである。エラー訂正符号により訂正可能なエラーであった場合は、それぞれ受信側でエラーを訂正するものとし、訂正ができなかった場合のみ再送要求コマンドを生成し、シリアルバス送信権が有効になった時点で再送要求コマンドを送信する。データの再送は、そのデータを送信した処理装置が再送するとは限らない。受信に失敗した処理装置を除けば、全ての処理装置が正しいデータを保持しているため、再送要求コマンドを受け取った後、最初に送信権が有効になった処理装置がデータの再送を実施することができる。これにより、データの再送を迅速に行うことができる。

【0058】

また、再送要求コマンドには、再送要求コマンドが送信される時点より何回前にシリアルバスに送信されたデータが再送対象であるかという情報を付加している。これは、通信エラーが発生したデータを特定するのに、通信時のデータを用いないようにするためである。再送要求コマンドを受信した処理装置は、内部に保持している受信履歴データと照らし合わせて再送すべきデータを判断し、再送する。但し、送信権が巡ってくる前に別の処理装置が再送データを送った場合や、自処理装置もその対象データの受信に失敗していた場合は、再送要求コマンドに対する処理は行わない。また、自処理装置が受信失敗していたデータに対する再送要求コマンドが、既に他の処理装置から送信されていた場合は、再送要求コマンドは送信しない。このように、シリアルバスに接続するすべての処理装置が連携してエラー検出／訂正することにより、1つのデータに対するデータ再送は1回のみ実施される。更に、同一データについて複数の処理装置の受信が失敗した場合でも、再送

処理を重複して行うことはなく、1回の再送で対処できる。

【図面の簡単な説明】

【0059】

【図1】本発明の一実施の形態による構成例を示すブロック図である。

【図2】本発明の一実施の形態によるメモリへのデータ書き込み処理例を示すフローチャートである。

【図3】本発明の一実施の形態による送信権管理処理例を示すフローチャートである。

【図4】本発明の一実施の形態によるデータ通信のアクセスタイミングを示す説明図である。

【図5】本発明の他の実施の形態による構成例を示すブロック図である。

10

【図6】本発明の他の実施の形態によるデータ受信処理例を示すフローチャートである。

【図7】本発明の他の実施の形態による再送処理例を示すフローチャートである。

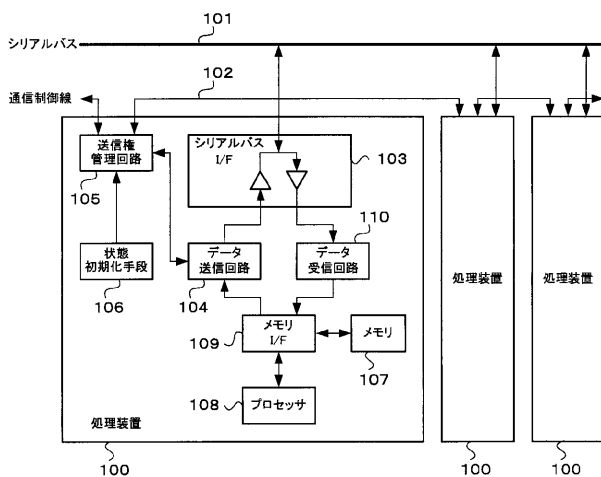
【符号の説明】

【0060】

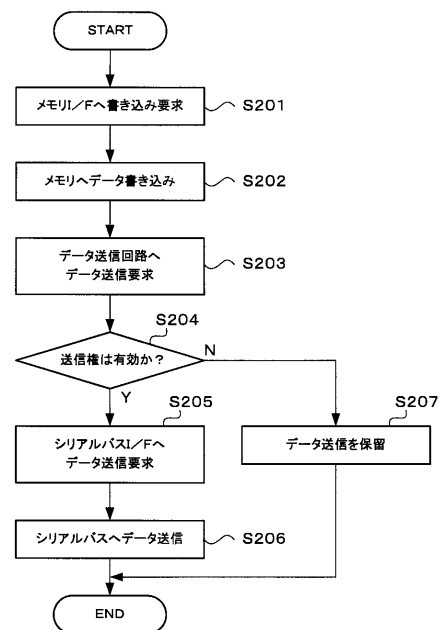
100…処理装置、101…シリアルバス、102…通信制御線、103…シリアルバスインタフェース(I/F)、104…データ送信回路、105…送信権管理回路、106…状態初期化手段、107…メモリ、108…プロセッサ、109…メモリインタフェース(I/F)、110…データ受信回路、111…符号生成回路、112…エラー訂正回路、113…再送要求生成回路、114…受信アドレスバッファ、115…再送アドレス生成回路、116…再送回路、117…再送要求キャンセル回路

20

【図1】

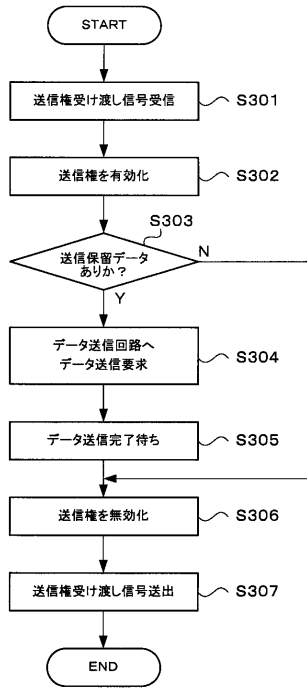


【図2】



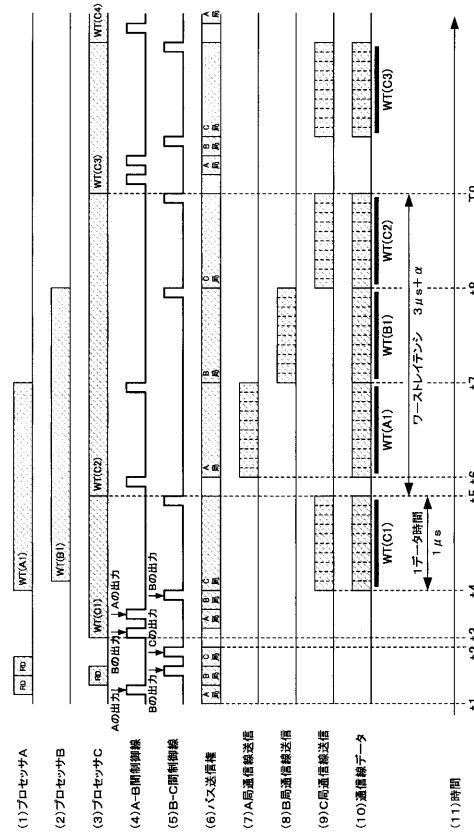
メモリへのデータ書き込み処理例

【図 3】

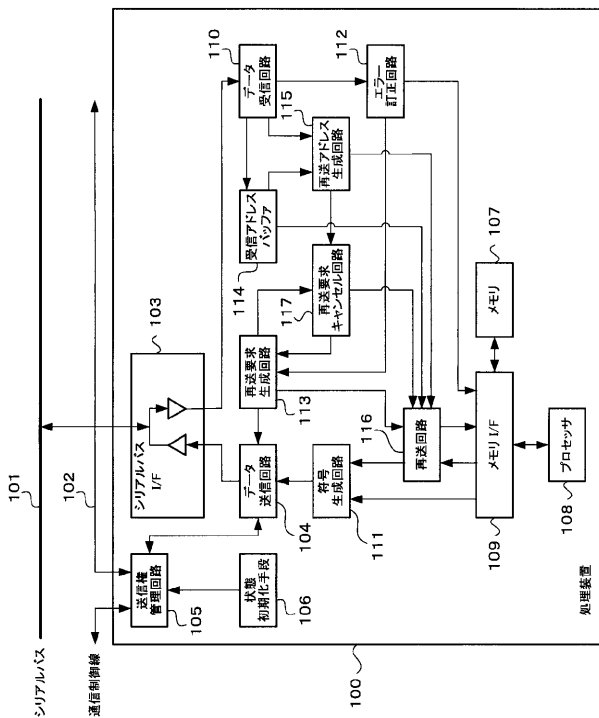


送信権管理処理例

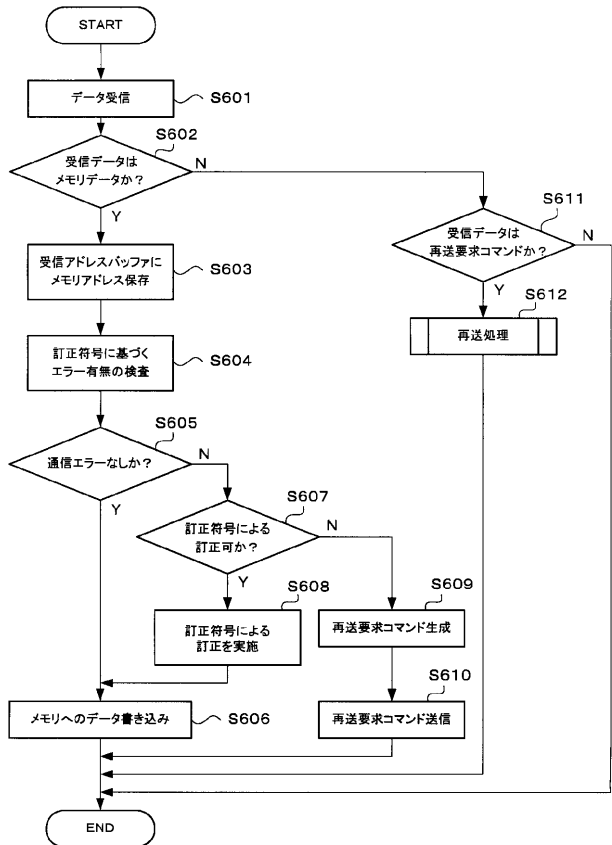
【図 4】



【図 5】



【図 6】



データ受信処理例

【図 7】

