

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

**特許第3970786号**  
**(P3970786)**

(45) 発行日 平成19年9月5日(2007.9.5)

(24) 登録日 平成19年6月15日(2007.6.15)

(51) Int. Cl. F I  
**G06F 13/38 (2006.01)** G O 6 F 13/38 3 5 0  
 G O 6 F 13/38 3 4 0 A

請求項の数 11 (全 26 頁)

(21) 出願番号	特願2003-59060 (P2003-59060)	(73) 特許権者	000005108
(22) 出願日	平成15年3月5日(2003.3.5)		株式会社日立製作所
(65) 公開番号	特開2004-272386 (P2004-272386A)		東京都千代田区丸の内一丁目6番6号
(43) 公開日	平成16年9月30日(2004.9.30)	(74) 代理人	110000198
審査請求日	平成16年5月18日(2004.5.18)		特許業務法人湘洋内外特許事務所
		(74) 代理人	100084032
			弁理士 三品 岩男
		(72) 発明者	恒富 邦彦
			茨城県日立市大みか町七丁目1番1号 株
			式会社日立製作所 日立研究所内
		(72) 発明者	吉村 健太郎
			茨城県日立市大みか町七丁目1番1号 株
			式会社日立製作所 日立研究所内

最終頁に続く

(54) 【発明の名称】 マルチプロセッサシステム

(57) 【特許請求の範囲】

【請求項1】

マスタにしたがうスレーブと定められた第1シングルマスタ型双方向通信インタフェースユニットと、前記第1シングルマスタ型双方向通信インタフェースユニットよりも通信速度の小さな第1マルチマスタ型双方向通信インタフェースユニットとを有するスレーブプロセッサと、

前記スレーブとの通信を開始するマスタと定められた第2シングルマスタ型双方向通信インタフェースユニットと、前記第2シングルマスタ型双方向通信インタフェースユニットよりも通信速度の小さな第2マルチマスタ型双方向通信インタフェースユニットとを有するマスタプロセッサと、

前記第1シングルマスタ型双方向通信インタフェースユニットと前記第2シングルマスタ型双方向通信インタフェースユニットとをつなぐ第1通信ケーブルと、

前記第1マルチマスタ型双方向通信インタフェースユニットと前記第2マルチマスタ型双方向通信インタフェースユニットとをつなぐ第2通信ケーブルと、

を有し、

前記スレーブプロセッサは、

データの通信開始を前記マスタプロセッサに要求する通信要求コマンドを、前記第1マルチマスタ型双方向通信インタフェースユニットから、前記第2通信ケーブルを介して前記マスタプロセッサに送信する第1コマンド通信手段を有し、

前記マスタプロセッサは、

前記第 2 マルチマスタ型双方向通信インタフェースが受信した前記通信要求コマンドに応じて、前記第 1 シングルマスタ型双方向通信インタフェースユニットとの通信を前記第 2 シングルマスタ型双方向通信インタフェースに開始させ、前記第 1 通信ケーブルを介して前記スレーブプロセッサに前記データを送信するデータ通信手段を有することを特徴とする

マルチプロセッサシステム。

【請求項 2】

請求項 1 記載のマルチプロセッサシステムであって、

前記第 1 シングルマスタ型双方向通信インタフェースユニット及び前記第 2 シングルマスタ型双方向通信インタフェースユニットは、SPI (Serial Peripheral

10

Interface) ユニットであることを特徴とするマルチプロセッサシステム。

【請求項 3】

請求項 1 記載のマルチプロセッサシステムであって、

前記第 1 マルチマスタ型双方向通信インタフェースユニット及び前記第 2 マルチマスタ型双方向通信インタフェースユニットは、シリアル通信インタフェースユニット及び IEEE 1394 インタフェースユニットのうちのいずれか一方であることを特徴とするマルチプロセッサシステム。

【請求項 4】

請求項 1 記載のマルチプロセッサシステムであって、

20

前記マスタプロセッサは、

入力装置が接続され、前記入力装置の出力データを受け付ける入力手段を有し、

前記スレーブプロセッサは、

前記入力装置の出力データを用いた演算を行うプログラムを有し、

前記第 1 コマンド通信手段は、前記プログラムの要求に応じて、前記入力装置の出力データの送信を要求する第 1 通信要求コマンドを、前記第 1 マルチマスタ型双方向通信インタフェースユニットから前記マスタプロセッサに送信し、

前記データ通信手段は、前記第 2 マルチマスタ型双方向通信インタフェースユニットが前記第 1 通信要求コマンドを受信した場合には、当該第 1 通信要求コマンドに応じて、前記第 1 シングルマスタ型双方向通信インタフェースユニットからの、前記入力装置の出力データの受信を前記第 2 シングルマスタ型双方向通信インタフェースユニットに開始させることを特徴とするマルチプロセッサシステム。

30

【請求項 5】

請求項 4 記載のマルチプロセッサシステムであって、

前記マスタプロセッサは、

前記入力装置に対応付けられ、当該入力装置の出力データが格納される第 1 入力ページを含む記憶手段を有し、

前記スレーブプロセッサは、

前記第 1 入力ページに対応付けられた第 2 入力ページを含む記憶手段を有し、

前記第 1 コマンド通信手段は、前記第 1 入力ページの指定を含む前記第 1 通信要求コマンドを送信し、

40

前記データ通信手段は、前記第 1 通信要求コマンドに含まれる指定が示す前記第 1 入力ページに格納されたデータの受信を前記第 2 シングルマスタ型双方向通信インタフェースユニットに開始させ、当該データを前記第 2 入力ページに格納することを特徴とするマルチプロセッサシステム。

【請求項 6】

請求項 1 記載のマルチプロセッサシステムであって、

前記マスタプロセッサは、

出力装置が接続され、前記出力装置に入力データを入力する出力手段を有し、

前記スレーブプロセッサは、

50

前記入力データを算出するプログラムを有し、

前記第1コマンド通信手段は、前記プログラムの要求に応じて、前記入力データの受信を要求する第2通信要求コマンドを、前記第1マルチマスタ型双方向通信インタフェースユニットから前記マスタプロセッサに送信し、

前記データ通信手段は、前記第2マルチマスタ型双方向通信インタフェースユニットが前記第2通信要求コマンドを受信した場合には、当該第2通信要求コマンドに応じて、前記第1シングルマスタ型双方向通信インタフェースユニットへの、前記出力装置の入力データの送信を前記第2シングルマスタ型双方向通信インタフェースユニットに開始させることを特徴とするマルチプロセッサシステム。

【請求項7】

請求項6記載のマルチプロセッサシステムであって、

前記スレーブプロセッサは、

前記出力装置に対応付けられ、当該出力装置の入力データが前記プログラムにより格納される第1出力ページを含む記憶手段を有し、

前記第1コマンド通信手段は、前記第1出力ページの指定を含む前記第2通信要求コマンドを送信し、

前記データ通信手段は、前記第2通信要求コマンドに含まれる指定が示す前記第1出力ページに格納された前記入力データの送信を前記第2シングルマスタ型双方向通信インタフェースユニットに開始させ、

前記マスタプロセッサは、

前記第1出力ページに対応付けられた第2出力ページを含む記憶手段と、

前記第2シングルマスタ型双方向通信インタフェースユニットが前記第2通信要求コマンドに応じて送信したデータを、前記第2出力ページに格納するデータ通信手段と、

を有することを特徴とするマルチプロセッサシステム。

【請求項8】

請求項1記載のマルチプロセッサシステムであって、

前記マスタプロセッサは、第1プログラムを有し、

前記スレーブプロセッサは、前記第1プログラムと協働する第2プログラムを有し、

前記第1コマンド通信手段は、前記第2プログラムの要求に応じて、前記第1プログラムの出力データの送信を要求する第1通信要求コマンド及び前記第2プログラムの出力データの受信を要求する第2通信要求コマンドを、前記第1マルチマスタ型双方向通信インタフェースユニットから前記マスタプロセッサに送信し、

前記データ通信手段は、前記第2マルチマスタ型双方向通信インタフェースユニットが前記第1通信要求コマンドを受信した場合には、当該第1通信要求コマンドに応じて、前記第2シングルマスタ型双方向通信インタフェースユニットに、前記第1シングルマスタ型双方向通信インタフェースユニットへの、前記第1プログラムの出力データの送信を開始させ、前記第2マルチマスタ型双方向通信インタフェースユニットが前記第2通信要求コマンドを受信した場合には、当該第2通信要求コマンドに応じて、前記第1シングルマスタ型双方向通信インタフェースユニットへの、前記第2プログラムの出力データの受信を前記第2シングルマスタ型双方向通信インタフェースユニットに開始させることを特徴とするマルチプロセッサシステム。

【請求項9】

請求項1記載のマルチプロセッサシステムであって、

前記第1マルチマスタ型双方向通信インタフェースユニットと前記第2マルチマスタ型双方向通信インタフェースユニットとに接続された第3通信ケーブルを有し、

前記マスタプロセッサは、

前記スレーブプロセッサに対するコマンドを、前記第2マルチマスタ型双方向通信インタフェースユニットから、前記第3通信ケーブルを介して前記スレーブプロセッサに送信する第2コマンド通信手段を有し、

当該第2コマンド通信手段は、

10

20

30

40

50

前記第 2 マルチマスタ型双方向通信インタフェースユニットが、前記第 2 通信ケーブルを介して前記通信要求コマンドを受信した場合には、前記第 2 マルチマスタ型双方向通信インタフェースユニットから、前記第 2 通信ケーブルを介して確認メッセージを前記スレーブプロセッサに送信し、

前記第 1 コマンド通信手段は、前記第 2 マルチマスタ型双方向通信インタフェースユニットが、前記第 3 通信ケーブルを介して前記コマンドを受信した場合には、前記第 3 通信ケーブルを介して確認メッセージを前記マスタプロセッサに送信することを特徴とするマルチプロセッサシステム。

【請求項 10】

請求項 1 記載のマルチプロセッサシステムであって、

前記第 1 マルチマスタ型双方向通信インタフェースユニットは、前記スレーブプロセッサの内部または外部に設けられ、前記第 2 マルチマスタ型双方向通信インタフェースユニットは、前記マスタプロセッサの内部または外部に設けられることを特徴とするマルチプロセッサシステム。

【請求項 11】

請求項 1 記載のマルチプロセッサシステムであって、

前記マスタプロセッサは、前記スレーブプロセッサから前記通信要求コマンドを受信した後、サーバの使用権または前記第 1 通信ケーブルの使用権を確保することを特徴とするマルチプロセッサシステム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、マルチプロセッサシステムにおけるプロセッサ間通信技術に関する。

【0002】

【従来の技術】

処理能力向上のため、自動車制御用のコントローラ等にはマルチプロセッサシステムが採用されることがある。コストパフォーマンス向上等の観点から、通常のマルチプロセッサシステムでは、複数のプロセッサが、それぞれのプロセッサに内蔵されたシリアル通信インタフェース(S C I : Serial Communication Interface)で接続されている。S C I の通信速度は、プロセッサの機種にも依存するが最大毎秒 5 0 0 K ~ 1 M ビット程度である。

【0003】

一方、S C I よりも高速なシリアル通信インタフェースとして、周辺装置用通信インタフェース(S P I : Serial Peripheral Interface)が知られている。この S P I は、プロセッサの機種にも依存するが最大毎秒 4 M ~ 1 0 M ビット程度の通信速度でデータを伝送可能であるが、周辺回路との接続を行うためのシングルマスタ型の全二重双方向通信であるため、送信を開始できるマスタをシリアル線上に 1 つしか存在させることができない。

【0004】

例えば、このような S P I を利用したシステムとして、マイクロコンピュータとマイクロコンピュータのメモリデータをモニタする外部装置との間を S C I ケーブル及び S P I ケーブルで接続したシステムが特開平 9 - 2 9 3 0 4 7 号公報(特許文献 1)に記載されている。このシステムにおいて、S C I ケーブルは、外部装置がマイクロコンピュータにメモリデータ上位アドレスを送信するために利用され、S P I ケーブルは、外部装置がマイクロコンピュータにメモリデータ下位アドレスを送信するため及びマイクロコンピュータからメモリデータを受信するために利用される。

【特許文献 1】

特開平 9 - 2 9 3 0 4 7 号公報

【0005】

【発明が解決しようとする課題】

ところで、マルチプロセッサシステムには、一層の処理能力の向上が望まれている。例えば、電気自動車、ハイブリッド自動車、電磁駆動弁を用いたエンジン等の登場に伴い、自

10

20

30

40

50

動車制御用のマルチプロセッサシステムには、より多くの制御対象及びセンサを接続する必要が生じ、プロセッサ間通信速度をより高速化する必要も生じている。

【0006】

そこで、本発明は、マルチプロセッサシステムの処理能力を向上させることを目的とする。

【0007】

【課題を解決するための手段】

本発明においては、

スレーブプロセッサと、

前記スレーブプロセッサにつながれた第一通信路と、

前記第一通信路につながれ、当該第一通信路を介した通信を開始する第二プロセッサと、  
前記スレーブプロセッサと前記マスタプロセッサとの間をつなぐ、前記第一通信路よりも  
通信速度が低速な第二通信路と、

を有し、

前記スレーブプロセッサは、

データの送信及び受信のうち少なくとも一方を要求するコマンドを、前記第二通信路を介して前記マスタプロセッサに送信するコマンド通信手段を有し、

前記マスタプロセッサは、

前記スレーブプロセッサからの前記コマンドに応じて、前記第一通信路を介した、前記スレーブプロセッサとの、前記データの通信を開始するデータ通信手段を有することを特徴とするマルチプロセッサシステムを提供する。

【0008】

【発明の実施の形態】

以下、添付図面を参照しながら、本発明に係る実施の形態について説明する。

【0009】

まず、図1により、本実施の形態に係るマルチプロセッサシステムのハードウェア構成について説明する。なお、ここでは、マルチプロセッサシステムの一例として、自動車制御用のコントロールユニット1の内部システムを挙げることにする。

【0010】

本実施の形態に係るコントロールユニット1は、互いに異なる通信速度でデータを伝送する2種類の通信ケーブル4, 5、2種類の通信ケーブル4, 5で接続された2つのプロセッサ2, 3を有する。

【0011】

2種類の通信ケーブル4, 5のうち、一方の通信ケーブル4は、2本の信号線を含むS C Iケーブルであり、他方の通信ケーブル5は、3本の信号線を含むS P Iケーブルである。S C Iケーブル4の2本の信号線は、一方のプロセッサ2が有するS C Iユニット(後述)のT Xポートと他方のプロセッサ3が有するS C Iユニット(後述)のR Xポートとの間、及び、一方のプロセッサ2が有するS C IユニットのR Xポートと他方のプロセッサ3が有するS C IユニットのT Xポートとの間をつないでいる。一方、S P Iケーブル5の信号線のうち、1本の信号線は、2つのプロセッサ2, 3が有するS P Iユニット(後述)のC L Kポート間をつなぎ、他の2本の信号線は、一方のプロセッサ2が有するS P IユニットのM Iポートと他方のプロセッサ3が有するS P IユニットのS Oポートとの間、及び、一方のプロセッサ2が有するS P IユニットのM Oポートと他方のプロセッサ3が有するS P IユニットのS Iポートとの間をつないでいる。

【0012】

一方、各プロセッサ2, 3は、自動車の制御に関する処理が定義されたプログラム等の各種データを保持する内蔵メモリ(RAM、ROM、EEPROM、Flashメモリ等)22, 32、内蔵メモリ22, 32等からフェッチした命令を実行する演算ユニット21, 31、S C Iケーブル4が接続されるとともにクロック同期式または調歩同期式の全二重シリアル通信を実現するS C Iユニット23, 33、S P Iケーブル5が接続されるとともにクロック同期式の

10

20

30

40

50

全二重シリアル通信を実現するS P Iユニット2 4, 3 4、制御対象となる1つ以上の出力装置(例えば、アクチュエータ、リレー)8 a, 8 bにデータを入力する出力ユニット(例えば、I / Oポート、A Dコンバータ)2 6, 3 6、制御対象の状態に相関する情報を検知する1つ以上の入力装置(例えば、センサ類)9 a, 9 bの出力を受け付ける入力ユニット(例えば、I / Oポート、A Dコンバータ)2 5, 3 5、これらのユニットを相互に接続する内部バス2 8, 3 8、を有している。なお、図1には、2つのプロセッサ2, 3の入力ユニット2 5, 3 5及び出力ユニット2 6, 3 6に入力装置9 a, 9 b及び出力装置8 a, 8 bが接続された構成を例示しているが、いずれか一方のプロセッサの入力ユニット及び出力ユニットだけに入力装置及び出力装置が接続されることもある。

**【0013】**

S C Iユニット2 3, 3 3は、2つのプロセッサ2, 3のいずれからでも通信(以下、S C I通信)を開始できるマルチマスタ型通信インタフェースである。S C Iは基本的には1 b y t e単位のデジタル入出力であるが、例えば8 b y t e程度の送受信バッファをS C Iユニット内に設ければ、連続して数b y t eのデータを送受信可能である。

**【0014】**

S P Iユニット2 4, 3 4は、マスタプロセッサと定められた1つのプロセッサ(本実施の形態ではプロセッサ3)からのみ他のプロセッサ(スレーブプロセッサ)との通信(以下、S P I通信)を開始できるシングルマスタ型の通信インタフェースである。これらのS P Iユニット2 4, 3 4は、S P Iケーブル5を介して他方のS P Iユニットから転送されたデータを保持するための数十byte程度の受信バッファ2 4 A, 3 4 A、及び、S P Iケーブル5を介して他方のS P Iユニットに転送されるデータを保持するための数十byte程度の送信バッファ2 4 B, 3 4 Bを有している。マスタプロセッサ3がC S (Chip Select)信号によって宛先スレーブプロセッサ2を指定すると、マスタプロセッサ3のS P Iユニット(マスタ)3 4の送信バッファ3 4 B内のデータが、マスタの生成するクロック信号と同期して、スレーブプロセッサ2のS P Iユニット(スレーブ)2 4の受信バッファ2 4 Aに転送されるとともに、スレーブの送信バッファ2 4 B内のデータがマスタの受信バッファ3 4 Aへ転送される。

**【0015】**

なお、以上述べたハードウェア構成は、一例であり、必要に応じて、適宜の変更を加えることができる。例えば、内蔵メモリのみではメモリ容量が不足する場合、または、書換え回数の少ないF l a s hメモリが内蔵メモリとして用いられている場合には、内蔵メモリのメモリ容量または書換え回数を補うため、図1に破線で示したように、外部バス6 a, 6 bを介してプロセッサ2, 3に外部メモリ(R A M、F l a s hメモリ等)7 a, 7 bを接続することが望ましい。また、図1に示した構成では、S C Iユニット及びS P Iユニットがプロセッサに内蔵されているが、例えばプロセッサ2, 3として比較的低機能な小型のものを用いる場合には、S C Iユニット及びS P Iユニットをプロセッサ外部に設けてもよい。

**【0016】**

このようなハードウェア構成を有するマルチプロセッサシステムによれば、自動車制御のための機能を2つのプロセッサ2, 3に分散する機能分散システム、及び、自動車制御に伴う負荷を2つのプロセッサ2, 3に分散する負荷分散システムのいずれも実現することができる。ここでは、入出力プロセッサ3の入力ユニット3 5及び出力ユニット3 6にだけ入力装置及び出力装置を1つ以上ずつ接続して、出力装置に関する制御処理(制御プログラム)をスレーブプロセッサ(以下、制御プロセッサ)2に割り当て、各外部装置(入力装置、出力装置)との入出力処理(入出力ドライバ)及び制御プロセッサ2上の制御プログラムによる演算結果の判定処理(アプリケーションプログラム)をマスタプロセッサ(以下、入出力プロセッサ)3に割り当てることによって実現した機能分散システムを例として、以下、2つのプロセッサ2, 3の内蔵メモリ2 2, 2 3のバッファ領域及びソフトウェア構成を説明する。

**【0017】**

2つのプロセッサ2,3の内蔵メモリ2,3(外部メモリ6,7が設けられている場合には、内蔵メモリ2,3または外部メモリ6,7)の記憶領域には、SPIユニットを介して送信するデータ及びSPIユニットを介して受信したデータを保持するためのバッファ領域が含まれる。それらのバッファ領域は、図2に示すように、固定長の固定記憶領域であるページ単位で管理され、それぞれのページには、ページIDとしてページ番号 $P_1$ ,  $P_N$ が割り当てられている。

#### 【0018】

入出力プロセッサ3のページ群には、複数の入力ページ3C、複数の出力ページ3Dが含まれ、制御プロセッサ2のページ群には、入出力プロセッサ3の各入力ページ3Cと同じページ番号が与えられた複数の入力ページ2C、入出力プロセッサ3の各出力ページ3Dと同じページ番号が与えられた複数の出力ページ2Dが含まれる。入出力プロセッサ3の各入力ページ3Cには、自身に対応付けられた入力装置の出力データが入出力ドライバによって格納され、制御プロセッサ2の各入力ページ2Cには、自身と同じページ番号の入力ページ3Cのデータがコピーされる。一方、制御プロセッサ2の各出力ページ2Dには、自身に対応付けられた出力装置への入力が制御プログラムによって格納され、入出力プロセッサ3の各出力ページ3Dには、自身と同じページ番号の出力ページ2Dのデータがコピーされる。このため、制御プロセッサ2上の制御プログラムは、制御プロセッサ2の入力ページ2Cからデータをリードすることによって、入出力プロセッサ2側の入力装置の出力を取得することができ、入出力プロセッサ3上の入出力ドライバは、入出力プロセッサ3の出力ページ3Dからデータをリードすることによって、制御プロセッサ2上の制御プログラムの出力(出力装置への入力データ)を取得することができる。

#### 【0019】

さらに、入出力プロセッサ3のページ群には、複数のライトオンリページ3Aと複数のリードオンリページ3Bとを有する共有メモリページが含まれ、制御プロセッサ2のページ群には、入出力プロセッサ3の各リードオンリページ3Bと同じページ番号が与えられた複数のライトオンリページ2Bと入出力プロセッサ3の各ライトオンリページ3Aと同じページ番号が与えられた複数のリードオンリページ2Aとを有する共有メモリページが含まれる。

#### 【0020】

入出力プロセッサ3の各ライトオンリページ3Aには、自身に対応付けられたアプリケーションプログラムの出力が格納され、制御プロセッサ2の各リードオンリページ2Aには、自身と同じページ番号のライトオンリページ3Aのデータがコピーされる。一方、制御プロセッサ2の各ライトオンリページ2Bには、自身に対応付けられた制御プログラムの出力が格納され、入出力プロセッサ3の各リードオンリページ3Bには、自身と同じページ番号のライトオンリページ2Bのデータがコピーされる。このため、制御プロセッサ2上の制御プログラムは、制御プロセッサ2のリードオンリページ2Aのデータをリードすることによって、入出力プロセッサ3上のアプリケーションプログラムの出力データを取得することができ、入出力プロセッサ3上のアプリケーションプログラムは、入出力プロセッサ3のリードオンリページ3Bのデータをリードすることによって、制御プロセッサ2上の制御プログラムの出力データを取得することができる。

#### 【0021】

なお、SPIユニットを介して送受信される通信データには、図3に示すように、これらのページに格納されるデータの格納領域 $D_1$ の他、ページ番号の格納領域 $D_2$ 、パッドの格納領域(本実施の形態では未使用) $D_3$ 、チェックサムの格納領域 $D_4$ 、チェックサムの1の補数の格納領域 $D_5$ が設けられる。このため、これらのページのサイズは、SPIユニット2,3の受信バッファ及び送信バッファの容量よりも4バイト小さくしてある。ただし、これらの格納領域 $D_2 \sim D_5$ を通信データに設けない場合には、ページのサイズを、SPIユニット2,3の受信バッファ及び送信バッファの容量と同じにしてもよい。

#### 【0022】

図4に、2つのプロセッサ2,3のソフトウェア構成を示す。

10

20

30

40

50

## 【 0 0 2 3 】

制御プロセッサ 2 の内蔵メモリ 2 2 に格納されたソフトウェアのなかには、出力装置に関する制御処理(エンジン制御、トルク制御、パワートレイン系制御、ボディ制御、ハイブリッド自動車及び電気自動車の場合にはさらにバッテリー制御等)ごとに準備された制御プログラム 2 0 0 A ~ 2 0 0 M、S C I 通信を行うコマンド通信部 2 2 0 を実現するプログラム群、S P I 通信を行うデータ通信部 2 1 0 を実現するプログラム群が含まれている。これらの概要は、以下の通りである。

## 【 0 0 2 4 】

各制御プログラム 2 0 0 A ~ 2 0 0 M は、自身に対応付けられた入力装置の出力データを用いて制御処理を実行し、それにより、自身に対応付けられた出力装置に与えるための制御データを生成する。これら各制御プログラムは、コマンド通信部 2 2 0 が提供する機能により、自身に対応付けられた入力装置 9 b の出力を入出力プロセッサ 3 から取得し、かつ、自身に対応付けられた出力装置 8 b に入出力プロセッサ 3 から制御データを入力させる。

10

## 【 0 0 2 5 】

コマンド通信部 2 2 0 のプログラム群には、入出力プロセッサ 3 に対するコマンドを制御プログラム 2 0 0 A ~ 2 0 0 M の要求に応じて生成する通信クライアント群、制御プログラム 2 0 0 A ~ 2 0 0 M と通信クライアント群とのアプリケーションインタフェースを実現するインタフェースプログラム 2 2 1、S C I 通信を制御する通信マネージャ 2 2 3、入出力プロセッサ 3 からのコマンドを実行する通信サーバ(共有メモリサーバ) 2 2 4、S C I ユニット 2 3 を制御する S C I ドライバ 2 2 5 が含まれている。

20

## 【 0 0 2 6 】

通信クライアント群には、出力ページ 2 D のデータ受信を入出力プロセッサ 3 に要求する受信要求コマンド(RECV\_OUT)を生成する出力クライアント 2 2 2 B、入力ページ 3 C のデータ送信を入出力プロセッサ 3 に要求する送信要求コマンド(XMIT\_IN)を生成する入力クライアント 2 2 2 C、ライトオンリページ 2 B、3 A のデータ交換を入出力プロセッサ 3 に要求する送受信要求コマンド(SYNC\_PAGE)を生成する共有メモリクライアント 2 2 2 A が含まれる。これらの通信クライアント 2 2 2 A ~ 2 2 2 C が生成したコマンドを入出力プロセッサ 3 に送信するためのコマンドメッセージには、図 5 に示すように、S C I コネクションの確立及び切断を要求する S C I コマンド 7 1、コマンドメッセージの識別に用いられる 1 ビット以上のシーケンシャルビットデータ 7 2、通信クライアントが生成したコマンドに応じて提供されるサービスの I D 7 3、通信クライアントが生成したコマンド(コマンド名、サービスごとにコマンドに割りふられたコマンド I D) 7 4、通信クライアントが生成したコマンドの引数 7 5 が含まれている。以下においてコマンドメッセージに格納する、サービス I D、コマンド名、コマンド I D 及び引数の例を、図 6 の項目(1)~(3)に示す。

30

## 【 0 0 2 7 】

インタフェースプログラム 2 2 1 には、制御プログラム 2 0 0 A ~ 2 0 0 M の要求に応じて通信クライアント 2 2 2 A ~ 2 2 2 C をコールする下記関数が含まれている。

(A) input\_sync(top,num)

40

第一引数 top: 受信ページの先頭ページ番号、第二引数 num: 受信ページ数

この入力関数は、入出力プロセッサ 3 の入力ページ 3 C のデータをリードするためのインタフェースを、制御プロセッサ 2 上の制御プログラム群 2 0 0 に提供する。具体的には、第一引数 top が示す入力ページ 3 C を先頭とする、第二引数 num に相当するページ数分の一連の入力ページ 3 C から、それらと同じページ番号の入力ページ 3 C にデータが転送されるように、入力クライアント 2 2 2 C をコールする。

(B) output\_sync(top,num)

第一引数 top: 送信ページの先頭ページ番号、第二引数 num: 送信ページ数

この出力関数は、入出力プロセッサ 3 の出力ページ 3 D にデータをライトするためのインタフェースを、制御プロセッサ 2 上の制御プログラム群 2 0 0 に提供する。具体的には、

50



第一引数topが示す出力ページ2 Dを先頭とする、第二引数numに相当するページ数分の一連の出力ページから、それらと同じページ番号の出力ページ3 Dにデータが転送されるように、出力クライアント2 2 2 Bをコールする。

(C) read\_write\_sync(r\_page, w\_page)

第一引数r\_page:受信ページ番号、第二引数w\_page:送信ページ番号

この共有メモリ関数は、制御プロセッサ2のライトオンリページ2 Bのデータ及び入出力プロセッサ3のライトオンリページ3 Aのデータを交換するためのインタフェースを、制御プロセッサ2上の制御プログラム群2 0 0及び入出力プロセッサ3上のアプリケーションプログラム群3 0 0に提供する。具体的には、第二引数w\_pageが示すライトオンリページ2 Bから、そのライトオンリページと同じページ番号のリードオンリページ3 Bにデータが転送され、かつ、第一引数r\_pageが示すライトオンリページ3 Aから、そのライトオンリページと同じページ番号のリードオンリページ2 Aにデータが転送されるように、共有メモリクライアント2 2 2 Aをコールする。

10

【0 0 2 8】

共有メモリサーバ2 2 4は、入出力プロセッサ3からのコマンドのうち、送受信要求コマンド(SYNC\_PAGE)及び後述のリード要求コマンド(READ\_PAGE)を解釈及び実行し、それぞれのコマンドの実行結果を表す応答コマンド(正常終了を表すACK、異常終了を表すNAK)を生成する。この応答コマンドを入出力プロセッサ3に送信するための応答メッセージのフォーマットは、通信クライアントが生成するコマンドメッセージのフォーマット(図5参照)と同じである。但し、データ領域7 5には、サービスIDの代わりに、コマンドに対する応答であることを示す応答IDが格納される。以下において応答メッセージに格納する、応答ID、コマンド名、コマンドID及び引数の一例を、図6の項目(4)に示す。

20

【0 0 2 9】

通信マネージャ2 2 3は、通信クライアント2 2 2 A ~ 2 2 2 C及び通信サーバ2 2 4が生成したメッセージの送信をSCIDライバ2 2 5に依頼するとともに、SCIDライバ2 2 5が受信した、入出力プロセッサ3からのメッセージを、通信サーバ2 2 4またはコマンド発行元通信クライアントに渡す。また、通信マネージャ2 2 3は、複数のサービスが通信マネージャ2 2 3を同時使用しないように、通信マネージャを使用中のサービスのサービスIDを保持するクライアント使用権及びサーバ使用権を管理する。これにより通信マネージャ2 2 3の競合使用が防止されるため、例えば通信マネージャ2 2 3のコマンド保持用バッファが1コマンド分のサイズでよくなる。このため、通信マネージャ2 2 3が簡素化される。また、後述するように、通信マネージャ2 2 3は、クライアント使用権の参照により、入出力プロセッサ3からの応答メッセージの宛先クライアントを、通信クライアント2 2 2 A ~ 2 2 2 Cのなかから特定することができるため、上述したように、応答メッセージにはサービスIDが加えられている必要がない。このため、低速なSCI通信のオーバーヘッドを低減することができる。

30

【0 0 3 0】

SCIDライバ2 2 5は、初期化处理(SCIユニット2 3の割込み設定、ボーレート設定等)、通信マネージャ2 2 3から渡されたメッセージをSCIユニット2 3から送信する送信処理、SCIユニット2 3でメッセージを受信してそれを通信マネージャ2 2 3に渡す受信処理を実行する。なお、SCIDライバ2 2 5は、SCIレベルのエラー(パリティエラー、フレーミングエラー)を送信処理で検出したときにはメッセージ再送処理を実行し、SCIレベルのエラーを受信処理で検出したときには受信パケットを破棄する。また、必要に応じて、通信マネージャ、制御プログラム等にエラーの種別を通知する。

40

【0 0 3 1】

データ通信部2 1 0のソフトウェア群には、ページ管理プログラム2 1 1、SPIユニット2 4を制御するSPIDライバ2 1 2が含まれている。

【0 0 3 2】

ページ管理プログラム2 1 1は、外部からの通信クライアント2 2 2 A ~ 2 2 2 C及び共

50

有メモリサーバ224からの要求に応じて、出力ページ2Dまたはライトオンリページ2BからSPIユニット24の送信バッファ24Bに、または、SPIユニット24の受信バッファ24Aから入力ページ2Cまたはリードオンリページ2Aにデータをページ単位で転送する。

**【0033】**

SPIドライバ225は、SPIユニット24の初期化処理(スレーブモード設定、SPI通信完了後の割込み禁止設定)、コマンドメッセージ送信直前の通信クライアントの要求に応じてSPIユニット24をイネーブルにする送受信開始処理を実行する。また、SPIドライバ225は、送受信中のメッセージ化けを検出するため、送信前には、SPIユニット24の送信バッファ24Bのチェックサムを計算して、チェックサムと1の補数をコマンドメッセージにライトし、受信後には、SPIユニット24の受信バッファ24Aのチェックサムを計算し、チェックサムと1の補数が正しいか否かをチェックする。なお、本実施の形態では、応答メッセージによってSPI通信の終了が通知されるようにするため、初期化処理で、SPI通信完了後の割込みを禁止に設定しているが、応答メッセージの代わりに、SPI通信完了時に割込みを発生させるようにする場合には、初期化処理で割込みを設定すればよい。

10

**【0034】**

一方、入出力プロセッサ3の内蔵メモリ32に格納されたソフトウェアのなかには、制御プロセッサ2の制御プログラムに対応付けて準備されたアプリケーションプログラム300A~300M、SCI通信を行うコマンド通信部320を実現するプログラム群、SPI通信を行うデータ通信部310を実現するプログラム群、入力装置9bの出力データ取込み及び出力装置8bへのデータ入力を実行する入出力ドライバ350が含まれる。これらの概要は、以下の通りである。なお、入出力プロセッサ2上のソフトウェアのうち、制御プロセッサ3上の同一名のソフトウェアと同様な処理を実行するものについては、詳細な説明を省略する。

20

**【0035】**

各アプリケーションプログラム300A~300Mは、制御プロセッサ2上の制御プログラム200A~200Mの演算結果を判定するため、制御プロセッサ2上の制御プログラム200A~200Mと同様の処理を実行し、その結果と、制御プロセッサ2上の制御プログラム200A~200Mの演算結果とを比較する。これらの各アプリケーションプログラムは、コマンド通信部320が提供する機能によって、制御プロセッサ2上の制御プログラムの演算結果の判定のために必要なデータを、制御プロセッサ2上の制御プログラムと交換する。

30

**【0036】**

コマンド通信部320のプログラム群には、アプリケーションプログラム300A~300Mの要求に応じたコマンドを生成する通信クライアント(共有メモリクライアント)322、アプリケーションプログラム群300と共有メモリクライアント322とのアプリケーションインタフェースを実現するインタフェースプログラム321、制御プロセッサ2上の通信マネージャと同様なSCI通信を制御する通信マネージャ323、制御プロセッサ2からのコマンドに応じたサービスを提供する通信サーバ群324、制御プロセッサ2上のSCIDライバ225と同様な処理によってSCIユニット33を制御するSCIDライバ325が含まれている。

40

**【0037】**

通信クライアントには、共有メモリクライアント322が少なくとも含まれる。共有メモリクライアント322が、ライトオンリページ2B,3Aのデータ交換を制御プロセッサ2に要求する送受信要求コマンド(SYNC\_PAGE)を生成するとともに受信バッファ24Aからのデータリードを制御プロセッサ2に要求する送受信要求コマンド(SYNC\_PAGE)を生成する要求する。

**【0038】**

インタフェースプログラム321には、制御プロセッサ2上のインタフェースプログラム

50

2 2 2 の共有メモリ関数 `read_write_sync()` と同じ共有メモリ関数 `read_write_sync()` が少なくとも含まれる。

【 0 0 3 9 】

データ通信部 3 1 0 のプログラム群には、制御プロセッサ 2 上のページ管理プログラム 2 1 1 と同様な処理を実行するページ管理部プログラム 3 1 1、SPI マネージャ 3 1 2、SPI ユニット 2 4 を制御する SPI ドライバ 3 1 3 が含まれている。

【 0 0 4 0 】

SPI マネージャ 3 1 2 は、複数のサービスが SPI ユニットの排他的に使用するよう、SPI ユニットの使用中のサービスのサービス ID を記録する SPI 使用権を管理する。なお、本実施の形態では、サーバ使用権の管理によって、複数のサービスに SPI ユニットの排他的に使用させることができているため、必ずしも SPI マネージャ 3 1 2 を設ける必要はない。

10

【 0 0 4 1 】

SPI ドライバ 3 2 5 は、SPI ユニット 3 4 の初期化処理(マスタモード設定、SPI 通信完了後の割込み許可設定)、通信サーバ群 3 2 4 の指示に応じて SPI 通信を開始させる送受信開始処理を実行する。送受信開始処理では、SPI ドライバ 3 2 5 は、SPI ユニット 3 4 のレジスタを用いて SPI 通信を開始させる。SPI 通信では、SPI ユニット 3 4 が、CLK を発信しながら、その CLK に同期させて送信バッファ 3 4 B のデータを制御プロセッサ 2 の SPI ユニット 2 4 の受信バッファ 2 4 A に転送させるとともに、制御プロセッサ 2 の SPI ユニット 2 4 の送信バッファ 2 4 B のデータを受信バッファ 3 4 A に転送させる。そして、SPI 通信完了後の割込みに応じて、SPI ドライバ 3 2 5 は、通信マネージャ 3 2 3 を起動した通信サーバを通信マネージャ 3 2 3 にコールさせる。

20

【 0 0 4 2 】

通信サーバ群 3 2 4 には、共有メモリサーバ 3 2 4 A、入力サーバ 3 2 4 C、出力サーバ 3 2 4 B が含まれている。共有メモリサーバ 3 2 4 A は、制御プロセッサ 2 からのコマンドのうち、送受信要求コマンド(SYNC\_PAGE)を解釈及び実行し、その実行結果を表す応答コマンド(A C K、N A K)を生成する。入力サーバ 3 2 4 C は、制御プロセッサ 2 からのコマンドのうち、送信要求コマンド(XMIT\_IN)を解釈及び実行し、その実行結果を示す応答コマンド(A C K、N A K)を生成する。出力サーバ 3 2 4 B は、制御プロセッサ 2 からのコマンドのうち、受信要求コマンド(RECV\_OUT)を解釈及び実行し、その実行結果を示す応答コマンド(A C K、N A K)を生成する。これらの応答コマンドを制御プロセッサ 2 に送信するための応答メッセージのフォーマットは、制御プロセッサ 2 の共有メモリサーバ 2 2 4 が生成する応答メッセージと同じである。

30

【 0 0 4 3 】

入出力ドライバ 3 5 0 は、通信サーバ 3 2 4 A ~ 3 2 4 C の要求に応じて、入力装置 9 からのデータ取込み及び出力装置 8 へデータ書込みを実行する。

【 0 0 4 4 】

なお、本実施の形態では、入出力プロセッサ 3 の入力ユニット 3 5 及び出力ユニット 3 6 にだけ入力装置 9 b 及び出力装置 8 b が接続しているため、入出力プロセッサのインタフェースプログラム 3 2 1 には入力関数 `input_sync()` 及び出力関数 `output_sync()` が含まれず、入出力プロセッサ 3 の通信クライアントには入力クライアント及び出力クライアントが含まれていない。また、制御プロセッサの通信サーバには入力サーバ及び出力サーバが含まれていない。しかし、制御プロセッサ 2 の入力ユニット 2 5 及び出力ユニット 2 6 にも入力装置 9 a 及び出力装置 8 b が接続され、かつ、入出力プロセッサ 3 上のアプリケーションプログラムが、その入力装置 9 a 及び出力装置 8 a とデータをやりとりする必要がある場合には、入出力プロセッサ 3 の通信クライアントに入力クライアント及び出力クライアントを含め、かつ、入出力プロセッサ 3 のインタフェースプログラム 3 2 1 に入力関数 `input_sync()` 及び出力関数 `output_sync()` を含める必要がある。また、制御プロセッサの通信サーバには入力サーバ及び出力サーバを含める必要がある。

40

50

## 【 0 0 4 5 】

また、本実施の形態に係る機能分散システムでは、出力装置に関する制御処理を制御プロセッサにのみ割り当てているため、制御プログラムが制御プロセッサにだけインストールされているが、例えば、出力装置に関する制御処理の一部が入出力プロセッサ3に割り当てられてもよい。また、上記の処理以外の処理が各プロセッサに割り当てられてもよい。

## 【 0 0 4 6 】

つぎに、以上のようなソフトウェア構成を有するマルチプロセッサシステムにおいて実行される4種類の処理について説明する。

(1)制御プロセッサ2上の制御プログラム200A~200Mのうち、いずれかの制御プログラム(ここでは、制御プログラム200Kとする)が入出力プロセッサ3側の入力装置9bの出力を取得する場合には、図7に示した処理が実行される。具体的には、以下の通りである。

## 【 0 0 4 7 】

制御プログラム200Kは、自身に対応付けられた複数の入力装置の出力が格納される一連の入力ページ3Cの先頭ページ番号及びページ数numを第一引数top及び第二引数として、インタフェースプログラム221の入力関数input\_sync()をコールする。なお、制御プログラム200Kに対応付けられた入力装置の出力が格納される入力ページ3Cが1ページである場合には、その入力ページのページ番号及びページ数「1」を第一引数top及び第二引数numとして入力関数input\_sync()がコールされる。

## 【 0 0 4 8 】

その後、入力関数input\_sync()によりコールされた入力クライアント222Cは、入出力プロセッサ3の入力サーバ324Cに要求する入力サービスのサービスID「0」の登録を通信マネージャ223に要求する。これにより、入力サービスのサービスID「0」がクライアント使用権に登録され、入力クライアント222Cは、クライアント使用権を獲得する(1101)。

## 【 0 0 4 9 】

さらに、入力クライアント222Cは、起動時に渡された第一引数top及び終了フラグ(numが1の場合、最終ページの送信要求コマンドであることを示す値「1」、numが複数の場合、「0」)を引数とする送信要求コマンドメッセージ(XMIT\_IN)を生成し、それを通信マネージャ223に渡す。通信マネージャ223が、その送信要求コマンドメッセージの送信をSCIDライバ225に依頼すると、その送信要求コマンドメッセージが、SCIケーブル4を介して入出力プロセッサ3に送信される(1102)。

## 【 0 0 5 0 】

入出力プロセッサ3では、SCIDライバ325が、制御プロセッサ2からのメッセージを受信すると、それが通信マネージャ323に渡される。通信マネージャ323は、それがコマンドメッセージであることを確認すると、そのコマンドメッセージを、そのコマンドメッセージに含まれていたサービスIDにより定まる入力サーバ324Cに渡す。入力サーバ324Cは、そのコマンドメッセージを解釈し、その解釈結果に基づき以下の処理を実行させる(1103)。

## 【 0 0 5 1 】

入力サーバ324Cは、コマンドメッセージに含まれていたサービスIDの登録を通信マネージャ323及びSPIマネージャ312に要求する。これにより、入力サーバ324Cが提供する入力サービスのサービスID「0」がサーバ使用権及びSPI使用権に登録され、入力サーバ324Cは、サーバ使用権及びSPI使用権を獲得する。さらに、入力サーバ324Cは、入力ページ3CからSPIユニット34へのデータ転送をページ管理プログラム311に依頼する。このとき、ページ管理プログラム311には、コマンドメッセージが引数として含むページ番号(転送元ページのページ番号)、入力ページに対する処理内容(リード)、転送先アドレス(送信バッファ34Bのアドレス)が渡される。これに応じて、ページ管理プログラム311は、入力ドライバ350を介して、すべての入力装置の出力をリードし、すべての入力ページ3Cのデータを、それぞれのページに対応付け

10

20

30

40

50

られた入力装置の出力で更新する(1104)。さらに、ページ管理プログラム311は、転送元ページからデータをリードし、それを、転送先アドレスが示す、送信バッファ34Bの記憶領域にライトする(1105)。送信バッファ34Bへのデータ転送が終了すると、入力サーバ324Cの指示に応じて、通信マネージャ323が、SPIドライバ313に送信開始を指示する。これにより、メッセージコマンドに引数として含まれていたページ番号の入力ページ3Cのデータ、即ち、入力装置の出力が、SPIケーブル5を介して制御プロセッサ2に送信される(1106)。

【0052】

1ページ分の入力ページのデータ送信が正常終了し、入出力プロセッサ3で割込みが発生すると、入力サーバ324Cは、応答ID「7」及び実行結果ACK等を含む応答メッセージを作成し、それを通信マネージャ323に渡す。通信マネージャ323が、その応答メッセージの送信をSCIドライバ325に依頼すると、その応答メッセージが、SCIケーブル4を介して制御プロセッサ2に送信される。

10

【0053】

また、入力サーバ324Cは、コマンドメッセージに引数として含まれていた終了フラグの値が「1」であるか否かを判断する。その結果、入力サーバ324Cは、終了フラグの値が「1」であれば、通信マネージャ323及びSPIマネージャ312にサーバ使用権及SPI使用権を開放させ、終了フラグの値が「1」でなければ、次の入力ページの送信要求コマンドメッセージ(XMIT\_IN)を待つ(1107)。

【0054】

20

制御プロセッサ2では、SCIドライバ225が、入出力プロセッサ3からのメッセージを受信すると、それを通信マネージャ323に渡す。通信マネージャ323は、そのメッセージが応答ID「7」を含めば、それを、クライアント使用権に登録されたサービスIDに対応する入力クライアント222Cに送信メッセージとして渡す。入力クライアント222Cは、送信メッセージがACKを含むこと、即ちSPI通信の正常終了を確認すると、SPIユニット24Aから入力ページ2Cへのデータ転送をページ管理プログラム211に依頼する。このとき、ページ管理プログラム211には、処理1102で送信要求コマンドメッセージの引数としたページ番号(転送先ページ番号)、転送ページに対する処理内容(ライト)、転送元アドレス(受信バッファ24Aのアドレス)が渡される。

【0055】

30

これに応じて、ページ管理プログラム211は、転送元アドレスが示す、受信バッファ24Aの記憶領域からデータをリードし、それを、転送先ページにライトする。受信バッファ24Aからのデータ転送が終了すると、入力クライアント222Cは、起動時にinput\_sync()から渡された第二引数numの示すページ数分のデータの受信が終了したか否かを判断する。

【0056】

その結果、起動時に渡された第二引数numが示すページ数分のデータの受信が終了していれば、入力クライアント222Cは、通信マネージャ223にクライアント使用権を開放させる(1108)。一方、起動時に渡された第二引数numが示すページ数分のデータの受信が終了していなければ、そのページ数分のデータの受信が終了するまで、以下の処理1109~1113が繰り返し実行される。

40

【0057】

制御プロセッサ2では、次の入力ページのページ番号及び終了フラグ(次の入力ページが最終ページであれば「1」、次の入力ページが最終ページでなければ「0」)を引数として、処理1102と同様な処理が実行される。これにより、次の入力ページの送信要求コマンドメッセージ(XMIT\_IN)が入出力プロセッサ3に送信される(1109)。

【0058】

入出力プロセッサ3では、その送信要求コマンドメッセージに応じて、処理1103及び処理1105と同様な処理が実行される(1110)。さらに、処理1106及び処理1107と同様な処理が実行される(1111,1113)。これにより、処理1110で新た

50

に受信した送信要求コマンドメッセージの引数が示すページ番号の入力ページ3Cのデータが、SPIケーブル5を介して制御プロセッサ2に送信され、さらに、応答メッセージが、SCIケーブル4を介して制御プロセッサ2に送信される。

【0059】

その後、制御プロセッサ2では、処理1108と同様な処理が実行され、SPIユニット24の受信バッファ24Aの記憶領域のデータが、つぎのページ番号の入力ページ2Cにライトされる(1112)。

【0060】

以上の処理により、制御プロセッサ2では、制御プログラム200Kが入力関数input\_sync()に渡した第一引数topが示す入力ページ3Cから、制御プログラム200Kが入力関数input\_sync()に渡した第二引数numが示すページ数分の入力ページまでの各ページに、それぞれ、データがライトされる。その結果、制御プログラム200Kは、それらの入力ページ3Cからデータをリードすることによって、自身に対応付けられた入力装置の出力データを取得することができる。(2)制御プロセッサ2上の制御プログラム200A~200Mのうち、いずれかの制御プログラム(ここでは、制御プログラム200Kとする)が入出力プロセッサ3側の出力装置8bにデータを入力する場合には、図8に示した処理が実行される。具体的には、以下の通りである。

【0061】

制御プログラム200Kは、自身に対応付けられた一連の出力ページ2Dにデータをライトしてから、それら一連の出力ページ2Dの先頭ページ番号top及びページ数numを第一引数及び第二引数として、インタフェースプログラム221の出力関数output\_sync()をコールする。なお、制御プログラム200Kに対応付けられた出力ページが1ページである場合には、その出力ページのページ番号及びページ数「1」を第一引数top及び第二引数numとして入力関数input\_sync()がコールされる。

【0062】

その後、output\_sync()によりコールされた出力クライアント222Bは、入出力プロセッサ3の出力サーバ234Bに要求する出力サービスのサービスID「1」の登録を通信マネージャ223に要求し、クライアント使用権を獲得する(1201)。

【0063】

さらに、出力クライアント222Bは、起動時に渡された第一引数topが示す出力ページ2DからSPIユニット24へのデータ転送をページ管理プログラム211に依頼する。このとき、ページ管理プログラム211には、転送元ページのページ番号、転送元ページに対する処理内容(リード)、転送先アドレス(送信バッファ34Bのアドレス)が渡される。これに応じて、ページ管理プログラム211は、転送元ページのデータをリードし、それを、転送先アドレスが示す、送信バッファ24Bの記憶領域にライトする。送信バッファ24Bへのデータ転送が終了すると、出力クライアント222Bは、起動時に渡された第一引数top及び終了フラグ(numが1の場合は「1」、numが複数の場合は「0」)を引数とする受信要求コマンドメッセージ(RECV\_OUT)を生成し、それを通信マネージャ223に渡す。通信マネージャ223が、その受信要求コマンドメッセージの送信をSCIDライバ225に依頼すると、その受信要求コマンドメッセージが、SCIケーブル4を介して入出力プロセッサ3に送信される(1202)。

【0064】

入出力プロセッサ3では、SCIDライバ325が、制御プロセッサ2からのメッセージを受信すると、それが通信マネージャ323に渡される。通信マネージャ323は、それがコマンドメッセージであることを確認すると、そのコマンドメッセージを、そのコマンドメッセージに含まれていたサービスIDにより定まる出力サーバ324Bに渡す。出力サーバ324Bは、そのコマンドメッセージを解釈して、その解釈結果に基づき以下の処理を実行させる(1203)。

【0065】

出力サーバ324Bは、コマンドメッセージに含まれていたサービスIDの登録を通信マ

10

20

30

40

50

ネージャ323及びSPIマネージャ312に要求することにより、サーバ使用権及びSPI使用権を獲得する(1204)。その後、通信マネージャ323は、出力サーバ324Bの指示に応じてSPIドライバ313に受信開始を指示する。これにより、メッセージコマンドに引数として含まれていたページ番号の出力ページ2Dのデータ、即ち、出力装置への入力データが、SPIケーブル5を介して入出力プロセッサ3に送信される(1205)。

【0066】

1ページ分の出力ページのデータ送信が終了し、入出力プロセッサ3で割込みが発生すると、出力サーバ324Bは、SPIユニット34からのデータ転送をページ管理プログラム311に依頼する。このとき、ページ管理プログラム311には、コマンドメッセージが引数として含むページ番号(転送先ページ番号)、転送先ページに対する処理内容(ライト)、転送元アドレス(受信バッファ34Aのアドレス)が渡される。これに応じて、ページ管理プログラム311は、転送元アドレスが示す、受信バッファ34Aの記憶領域からデータをリードし、それを転送先ページ3Dにライトする。受信バッファ34Aからのデータ転送が正常終了すると、出力サーバ324Bは、応答ID「7」及び実行結果ACK等を含む応答メッセージを作成する。通信マネージャ223が、その応答メッセージの送信をSCIドライバ325に依頼すると、その応答メッセージが、SCIケーブル4を介して制御プロセッサ2に送信される。

10

【0067】

また、出力サーバ324Bは、コマンドメッセージに引数として含まれていた終了フラグの値が「1」であるか否かを判断する。その結果、出力サーバ324Bは、終了フラグの値が「1」であれば、通信マネージャ及びSPIマネージャにサーバ使用権及びSPI使用権を開放させ、終了フラグの値が「1」でなければ、次の出力ページの受信要求コマンドメッセージ(RECV\_OUT)を待つ(1206)。

20

【0068】

制御プロセッサ2では、SCIドライバ225が、入出力プロセッサ3からのメッセージを受信すると、それが通信マネージャ223に渡される。通信マネージャ223は、そのメッセージに応答ID「7」が含まれれば、それを、クライアント使用権に登録されたサービスID「1」に対応する出力クライアント222Bに応答メッセージとして渡す。出力クライアント222Bは、応答メッセージがACKを含むこと、即ち、SPI通信の正常終了を確認すると、起動時にoutput\_sync()から渡された第二引数numの示すページ数分のデータの送信が終了したか否かを判断する。

30

【0069】

その結果、起動時に渡された第二引数numが示すページ数分のデータの送信が終了していれば、出力クライアント222Bは、通信マネージャ223にクライアント使用権を開放させる(1207)。一方、起動時に渡された第二引数numが示すページ数分のデータの送信が終了していなければ、そのページ数分のデータ送信が終了するまで、以下の処理1208~1213が繰り返し実行される。

【0070】

制御プロセッサ2では、次の出力ページのページ番号及び終了フラグ(最終ページであれば「1」、最終ページでなければ「0」)を引数として、処理1202と同様な処理が実行される。これにより、受信要求コマンドメッセージ(RECV\_OUT)が入出力プロセッサに送信される(1208)。

40

【0071】

入出力プロセッサ3では、その受信要求コマンドメッセージに応じて、処理1203と同様な処理が実行され(1209)、さらに、処理1205及び処理1206と同様な処理が実行される(1210,1211)。これにより、受信要求コマンドメッセージの引数が示すページ番号の出力ページ2Dから、それと同じページ番号の出力ページ3Dへ、SPIケーブル5を介してデータが転送され、さらに、SCIケーブル4を介して制御プロセッサ2に応答メッセージが送信される。

50

## 【 0 0 7 2 】

制御プロセッサ 2 では、入出力プロセッサ 3 からの応答メッセージを受信すると、処理 1 2 1 1 と同様な処理が実行される (1 2 1 2)。

## 【 0 0 7 3 】

以上の処理により、入出力プロセッサ 3 では、制御プログラム 2 0 0 K が出力関数 output\_sync() に渡した第一引数 top が示すページ番号の出力ページ 3 D から、制御プログラム 2 0 0 K が出力関数 output\_sync() に渡した第二引数 num が示すページ数分の出力ページ 3 D までの各ページに、それぞれ、データがライトされる。その結果、入出力ドライバ 3 5 0 は、それらの出力ページ 3 D のデータをリードすることにより、それらの出力ページ 3 D に対応付けられた出力装置への入力を取得することができる。

10

(3) 制御プロセッサ 2 上の制御プログラム 2 0 0 A ~ 2 0 0 M のうち、いずれかの制御プログラム (ここでは、制御プログラム 2 0 0 K とする) が、入出力プロセッサ 3 上のアプリケーションプログラム 3 0 0 K にデータ交換を要求する場合には、図 9 に示した処理が実行される。具体的には、以下の通りである。

## 【 0 0 7 4 】

制御プログラム 2 0 0 K は、アプリケーションプログラム 3 0 0 K に対応付けられたライトオンリページ 2 B にデータをライトしてから、アプリケーションプログラム 3 0 0 K に対応付けられたライトオンリページ 2 B 及びリードオンリページ 2 A の各ページ番号を第一引数 r\_page 及び第二引数 w\_page として、インタフェースプログラム 2 2 1 のリードライト関数 read\_write\_sync() をコールする。その後、リードライト関数 read\_write\_sync() に

20

## 【 0 0 7 5 】

さらに、共有メモリクライアント 2 2 2 A は、起動時に渡された第二引数 w\_page が示すライトオンリページ 2 B から S P I ユニット 2 4 へのデータ転送をページ管理プログラム 2 1 1 に依頼する。このとき、ページ管理プログラム 2 1 1 には、転送元ページのページ番号 w\_page、ライトオンリページに対する処理内容 (リード)、転送先アドレス (送信バッファ 2 4 B のアドレス) が渡される。これに応じて、ページ管理プログラム 2 1 1 は、転送元ページのデータをリードし、それを、転送先アドレスが示す、送信バッファ 2 4 B の記

30

## 【 0 0 7 6 】

入出力プロセッサ 3 では、S C I ドライバ 3 2 5 が、制御プロセッサ 2 からのメッセージを受信すると、それが通信マネージャ 3 2 3 に渡される。通信マネージャ 3 2 3 は、それがコマンドメッセージであることを確認すると、そのコマンドメッセージを、そのコマンドメッセージに含まれていたサービス ID により定まる共有メモリサーバ 3 2 4 A に渡す。共有メモリサーバ 3 2 4 A は、そのコマンドメッセージを解釈して、その解釈結果に基づき以下の処理を実行させる。

40

## 【 0 0 7 7 】

共有メモリサーバ 3 2 4 A は、コマンドメッセージに含まれているサービス ID 「 2 」の登録を通信マネージャ 3 2 3 及び S P I マネージャ 3 1 2 に要求することによりサーバ使用権及び S P I 使用権を獲得すると、さらに、起動時に渡された引数 r\_page が示すライトオンリページ 3 A から S P I ユニット 3 4 へのデータ転送をページ管理プログラム 3 1 1 に依頼する。このとき、ページ管理プログラム 3 1 1 には、転送元ページのページ番号 r\_

50



page、転送元ページに対する処理内容(リード)、送信バッファ34Bのアドレス(転送先アドレス)が渡される。これに応じて、ページ管理プログラム311は、転送元ページのデータをリードし、それを、転送先アドレスが示す、送信バッファ34Bの記憶領域に転送する(1303)。

**【0078】**

送信バッファ34Bへのデータ転送が終了すると、通信マネージャ323は、共有メモリサーバ324Aの指示に応じて、SPIドライバ313に送受信開始を指示する。これにより、制御プロセッサ2上の制御プログラム200Kの出力が、SPIケーブル5を介して入出力プロセッサ3に送信されるとともに、入出力プロセッサ3上のアプリケーションプログラム300Kの出力が、SPIケーブル5を介して制御プロセッサ2に送信される(1304)。

10

**【0079】**

共有メモリページのデータ交換が終了し、入出力プロセッサ3で割込みが発生すると、共有メモリサーバ324Aは、受信バッファ34Aの61バイト目のデータをリードし、そのデータが示すページ番号のリードオンリページ3Bへのデータ転送をページ管理プログラム311に依頼する。このとき、ページ管理プログラム311には、転送先ページ番号(61バイト目のデータ)、転送先ページに対する処理内容(ライト)、転送元アドレス(受信バッファ34Aのアドレス)が渡される。これに応じて、ページ管理プログラム311は、転送元アドレスが示す、受信バッファ34Aの記憶領域からデータをリードし、それを転送先ページ3Bにライトする。転送先ページへのデータ転送が正常終了すると、共有メモリサーバ324Aは、応答ID「7」及び実行結果ACK等を含む応答メッセージを作成し、それを通信マネージャ323に渡す。通信マネージャ223が、その応答メッセージの送信をSCIドライバ325に依頼すると、その応答メッセージが、SCIケーブル4を介して制御プロセッサ2に送信される。さらに、共有メモリサーバ324Aは、通信マネージャ及びSPIマネージャにサーバ使用権及SPI使用権を開放させる(1305)。

20

**【0080】**

制御プロセッサ2では、SCIドライバ225が入出力プロセッサ3からのメッセージを受信すると、それが通信マネージャ223に渡される。通信マネージャ223は、そのメッセージが応答ID「7」を含めば、それを、クライアント使用権に登録されたサービスIDに対応する共有メモリクライアントに応答メッセージとして渡す。共有メモリクライアント222Aは、応答メッセージがACKを含むこと、即ちSPI通信の正常終了を確認すると、起動時に渡された第一引数r\_pageが示すリードオンリページ2Aへのデータ転送をページ管理プログラム211に依頼する。このとき、ページ管理プログラム211には、転送先ページ番号r\_page、転送先ページに対する処理内容(ライト)、SPIユニット24の受信バッファ24Aのアドレス(転送元アドレス)が渡される。これに応じ、ページ管理プログラム211は、転送元アドレスが示す、受信バッファ24Aの記憶領域からデータをリードし、それを転送先ページ2Aにライトする。転送先ページへのデータ転送が終了すると、共有メモリクライアント222Aは、通信マネージャ223にクライアント使用権を開放させる(1306)。

30

40

**【0081】**

以上の処理により、入出力プロセッサ3のライトオンリページ3Aから、そのライトオンリページと同じページ番号の、制御プロセッサ2のリードオンリページ2Aにデータが転送される。その結果、制御プロセッサ2上の制御プログラムは、それらのリードオンリページ2Aのデータをリードすることによって、自身に対応付けられた、入出力プロセッサ3上のアプリケーションプログラムの出力を取得することができる。その一方で、制御プロセッサ2のライトオンリページ2Bから、そのライトオンリページと同じページ番号の、入出力プロセッサ3のリードオンリページ3Bに転送される。その結果、入出力プロセッサ3上のアプリケーションプログラムは、それらのリードオンリページ3Bのデータをリードすることによって、自身に対応付けられた、制御プロセッサ2上の制御プログラム

50

の出力を取得することができる。これにより、制御プロセッサ 2 上の制御プログラムと入出力プロセッサ 3 上のアプリケーションプログラムとの協調動作が可能となる。

(4) 入出力プロセッサ 3 のアプリケーションプログラム 3 0 0 A ~ 3 0 0 M のうち、いずれかのアプリケーションプログラム(ここでは、アプリケーションプログラム 3 0 0 K とする)が、制御プロセッサ 2 上の制御プログラム 2 0 0 K にデータ交換を要求する場合には、図 1 0 に示した処理が実行される。具体的には、以下の通りである。

【 0 0 8 2 】

アプリケーションプログラム 3 0 0 K は、制御プログラム 2 0 0 K に対応付けられたライトオンリページ 3 A にデータをライトしてから、その制御プログラム 2 0 0 K に対応付けられたライトオンリページ 3 A 及びリードオンリページ 3 B の各ページ番号を第一引数  $r\_page$  及び第二引数  $w\_page$  として、インタフェースプログラム 3 2 1 のリードライト関数  $read\_write\_sync()$  をコールする。その後、リードライト関数  $read\_write\_sync()$  によってコールされた共有メモリアダプタ 3 2 2 は、制御プロセッサ 2 の共有メモリサーバ 2 2 4 に要求する共有メモリアダプタのサービス ID 「 2 」の登録を通信マネージャ 3 2 3 及び S P I マネージャ 3 1 2 に要求し、クライアント使用権及び S P I 使用権を獲得する。

【 0 0 8 3 】

さらに、共有メモリアダプタ 3 2 2 は、起動時に渡された第二引数  $w\_page$  が示すライトオンリページ 3 A から S P I ユニット 3 4 へのデータ転送をページ管理プログラム 3 1 1 に依頼する。このとき、ページ管理プログラム 3 1 1 には、転送元ページ番号  $w\_page$ 、転送元ページに対する処理内容(リード)、転送先アドレス(送信バッファ 3 4 B のアドレス)が渡される。これに応じ、ページ管理プログラム 3 1 1 は、転送元ページのデータをリードし、それを、転送先アドレスが示す、送信バッファ 3 4 B の記憶領域に転送するとともに、送信バッファの 6 1 バイト目(図 3 の  $D_2$ )に転送元ページ番号  $w\_page$  をライトする。送信バッファ 3 4 B へのデータ転送が終了すると、共有メモリアダプタ 3 2 2 は、起動時に渡された第一引数  $r\_page$  を引数とする送受信要求コマンドメッセージ(SYNC\_PAGE)を生成し、それを通信マネージャ 2 2 3 に渡す(1 4 0 1)。通信マネージャ 3 2 3 が、その送受信要求コマンドメッセージの送信を S C I ドライバ 3 2 5 に依頼すると、その送受信要求コマンドメッセージが、S C I ケーブル 4 を介して制御プロセッサ 2 に送信される(1 4 0 2)。

【 0 0 8 4 】

制御プロセッサ 2 では、S C I ドライバ 2 2 5 が、入出力プロセッサ 3 からのメッセージを受信すると、それが通信マネージャ 2 2 3 に渡される。通信マネージャ 2 2 3 は、それがコマンドメッセージであることを確認すると、そのコマンドメッセージを、そのコマンドメッセージに含まれていたサービス ID により定まる共有メモリサーバ 2 2 4 に渡す。共有メモリサーバ 2 2 4 は、そのコマンドメッセージを解釈して、その解釈の結果に基づき以下の処理を実行させる。

【 0 0 8 5 】

共有メモリサーバ 2 2 4 は、コマンドメッセージに含まれているサービス ID 「 2 」の登録を通信マネージャ 3 2 3 及び S P I マネージャ 3 1 2 に要求することによりサーバ使用権及び S P I 使用権を獲得すると、起動時に渡された引数  $r\_page$  が示すライトオンリページ 2 B から S P I ユニット 2 4 へのデータ転送をページ管理プログラム 2 1 1 に依頼する。このとき、ページ管理プログラム 2 1 1 には、転送元ページ番号  $r\_page$ 、転送元ページに対する処理内容(リード)、転送先アドレス(送信バッファ 2 4 B のアドレス)が渡される。これに応じて、ページ管理プログラム 2 1 1 は、転送元ページのデータをリードし、それを、転送先アドレスが示す、送信バッファ 2 4 B の記憶領域にライトする。送信バッファ 2 4 B へのデータ転送が正常終了すると、共有メモリサーバ 2 2 4 は、応答 ID 「 7 」及び実行結果 A C K 等を含む応答メッセージを作成し、それを通信マネージャ 2 2 3 に渡す。通信マネージャ 2 2 3 が、その応答メッセージの送信を S C I ドライバ 2 2 5 に依頼すると、その応答メッセージが、S C I ケーブル 4 を介して入出力プロセッサ 3 に送信される(1 4 0 3)。

10

20

30

40

50

## 【 0 0 8 6 】

入出力プロセッサ3では、S C Iドライバ3 2 5が、制御プロセッサ2からのメッセージを受信すると、それが通信マネージャ3 2 3に渡される。通信マネージャ3 2 3は、そのメッセージが応答ID「7」を含めば、そのメッセージを、クライアント使用権に登録されたサービスIDに対応する共有メモリクライアント3 2 2に返信メッセージとして渡す。返信メッセージがACKを含むこと、即ち、制御プロセッサ2側においてS P Iユニット2 4へのデータ転送が正常終了していることが、共有メモリクライアント3 2 2により確認されると(1 4 0 4)、通信マネージャ3 2 3は、共有メモリクライアント3 2 2の指示に応じ、S P Iドライバ3 1 3に送受信開始を指示する。これにより、入出力プロセッサ3のライトオンリページ3 Aのデータ(即ち、入出力プロセッサ3上のアプリケーションプログラム3 0 0 Kの出力)が、S P Iケーブル5を介して制御プロセッサ2に送信されるとともに、制御プロセッサ2のライトオンリページ2 Bのデータ(即ち、制御プロセッサ2上の制御プログラム2 0 0 Kの出力)が、S P Iケーブル5を介して入出力プロセッサ3に送信される(1 4 0 5)。

10

## 【 0 0 8 7 】

共有メモリページのデータ交換が正常終了し、入出力プロセッサ3で割り込みが発生すると、共有メモリクライアント3 2 2は、起動に渡された第一引数r\_pageが示すページ番号のリードオンリページ3 Bへのデータ転送をページ管理プログラム3 1 1に依頼する。このとき、ページ管理プログラム3 1 1には、転送先ページ番号r\_page、転送先ページに対する処理内容(ライト)、転送元アドレス(S P Iユニット3 4の受信バッファ3 4 Aのアドレス)が渡される。これに応じて、ページ管理プログラム3 1 1は、転送元アドレスが示す、受信バッファ3 4 Aの記憶領域からデータをリードし、そのデータを転送先ページ3 Bにライトする。

20

## 【 0 0 8 8 】

転送先ページへのデータ転送が終了すると、共有メモリサーバ3 2 4 Aは、リード要求コマンドメッセージ(READ\_PAGE)を作成し、それを通信マネージャ3 2 3に渡す。通信マネージャ2 2 3が、そのリード要求コマンドメッセージの送信をS C Iドライバ3 2 5に依頼すると、そのリード要求コマンドメッセージが、S C Iケーブル4を介して制御プロセッサ2に送信される(1 4 0 6)。

## 【 0 0 8 9 】

制御プロセッサ2では、S C Iドライバ2 2 5が、入出力プロセッサ3からのメッセージを受信すると、それが通信マネージャ2 2 3に渡される。通信マネージャ2 2 3は、それがコマンドメッセージであることを確認すると、そのコマンドメッセージを、そのコマンドメッセージに含まれるサービスIDにより定まる共有メモリサーバ2 2 4に渡す。共有メモリサーバ2 2 4は、そのコマンドメッセージを解釈し、受信バッファ2 4 Aの6 1バイト目のデータ(w\_page)が示すリードオンリページ2 Aへのデータ転送をページ管理プログラム2 1 1に依頼する。このとき、ページ管理プログラム2 1 1には、転送先ページ番号(6 1バイト目のデータ)、転送先ページに対する処理内容(ライト)、転送元アドレス(受信バッファ3 4 Aのアドレス)が渡される。

30

## 【 0 0 9 0 】

これに応じて、ページ管理プログラム2 1 1は、転送元アドレスが示す、受信バッファ2 4 Aの記憶領域からデータをリードし、そのデータを転送先ページ2 Aにライトする。受信バッファ2 4 Aからのデータ転送が正常終了すると、共有メモリサーバ2 2 4は、応答ID「7」及び実行結果ACK等を含む返信メッセージを作成し、それを通信マネージャ2 2 3に渡す。通信マネージャ2 2 3は、その返信メッセージの送信をS C Iドライバ2 2 5に依頼すると、その返信メッセージが、S C Iケーブル4を介して入出力プロセッサ3に送信される。さらに、共有メモリサーバ2 2 4は、通信マネージャ2 2 3にサーバ使用権を開放させる(1 4 0 7)。

40

## 【 0 0 9 1 】

入出力プロセッサ3では、制御プロセッサ2からのメッセージをS C Iドライバ3 2 5が

50

受信すると、それが通信マネージャ323に渡される。通信マネージャ323は、そのメッセージが応答ID「7」を含めば、それを、クライアント使用権に登録されたサービスIDに対応する共有メモリクライアント322に応答メッセージとして渡す。応答メッセージがACKを含むこと、即ち、制御プロセッサ2とのデータ交換の正常終了が共有メモリクライアント322により確認されると、通信マネージャ323及びSPIマネージャは、共有メモリクライアント322の指示に応じて、メモリクライアント使用権及びSPI使用権を解放する(1408)。

**【0092】**

以上の処理により、制御プロセッサ2のライトオンリページ2Bから、そのライトオンリページと同じページ番号の、入出力プロセッサ3のリードオンリページ3Bにデータが転送される。その結果、入出力プロセッサ3上のアプリケーションプログラムは、それらのリードオンリページ3Bのデータをリードすることによって、自身に対応付けられた、制御プロセッサ2上の制御プログラムの出力を取得することができる。その一方で、入出力プロセッサ3のライトオンリページ3Aから、そのライトオンリページと同じページ番号の、制御プロセッサ2のリードオンリページ2Aにデータが転送される。その結果、制御プロセッサ2上の制御プログラムは、それらのリードオンリページ2Aのデータをリードすることによって、自身に対応付けられた、入出力プロセッサ3上のアプリケーションプログラムの出力を取得することができる。これにより、制御プロセッサ2上の制御プログラムと入出力プロセッサ3上のアプリケーションプログラムとの協調動作が可能となる。

**【0093】**

なお、本実施の形態では、入力プロセッサのライトオンリページから制御プロセッサへのリードオンリページへのデータ転送と、制御プロセッサのライトオンリページから入力プロセッサへのリードオンリページへのデータ転送とが、制御プログラムからの1の要求に応じて実行されるようにしているが、制御プログラムからの個別の要求に応じて個別のタイミングで実行されるようにしてもよい。

**【0094】**

以上述べたように、本実施の形態に係るソフトウェア構成によれば、2つのプロセッサのSPIユニットをあたかもマルチマスタ型通信インタフェースであるかのように利用して、2つのプロセッサ間でデータを双方向にやりとりすることができる。このため、SCIのみを利用したプロセッサ間通信より高速なプロセッサ間通信を実現することができる。そして、このようにプロセッサ間通信の高速化が図られたことにより、外部装置との入出力を専用プロセッサに実行させることが可能となり、制御対象及びセンサの増加にも対応可能となる。これらのことから、本実施の形態によれば、マルチプロセッサシステムの処理能力の向上を図ることができる。

**【0095】**

また、プロセッサの外部バスを共有するバス結合型マルチプロセッサシステムとは異なり、プロセッサ間通信のための外付け調停回路が不要であるため、安価なマルチプロセッサシステムを実現することができる。

**【0096】**

なお、本実施の形態では、入力プロセッサの入力ページから制御プロセッサへの入力ページへのデータ転送(input\_sync()による処理)と、制御プロセッサの出力ページから入力プロセッサへの出力ページへのデータ転送(output\_sync()による処理)とが、制御プログラムからの個別の要求に応じて個別のタイミングで実行されるようにしているが、制御プログラムからの1の要求に応じて両データ転送が実行されるようにしてもよい。このようにする場合には、例えば、入力クライアント222C及び出力クライアント222Bの代わりに入出力クライアントを制御プロセッサ2にインストールしておき、入出力クライアントをコールする関数がコールされたら、入出力クライアントが、制御プログラムの指定した入力ページのデータを送信バッファにコピーし、制御プログラムの指定した出力ページ番号を引数とするコマンドメッセージを送信するようにすればよい。一方、入出力プロセッサ3には、入力サーバ324C及び出力サーバ324Bの代わりに入出力サーバをイン

10

20

30

40

50

ストールしておき、入出力サーバが、入出力クライアントからのコマンドメッセージを受信したら、そのコマンドメッセージで指定された出力ページのデータを送信バッファにコピーし、S P I通信開始を通信マネージャ3 2 3に指示するようにすればよい。そして、S P I通信によるデータ交換が終了したら、S P Iユニットから出力ページへのデータ転送が入出力プロセッサで行われ、かつ、S P Iユニットから入力ページへのデータ転送が制御プロセッサで行われるようにすればよい。このようにすれば、入力プロセッサの入力ページから制御プロセッサの入力ページへのデータ転送と制御プロセッサの出力ページから入力プロセッサの出力ページへのデータ転送とが個別に実行される上述の場合と比較して、入力装置からのデータ取得及び出力装置へのデータ入力に要する時間をほぼ半減させることができる。

10

**【0097】**

また、本実施の形態では、制御プロセッサのS P Iユニットをスレーブ、入出力プロセッサのS P Iユニットをマスタとしたが、制御プロセッサのS P Iユニットをマスタ、入出力プロセッサのS P Iユニットをスレーブとしてもよい。このようにする場合、S P Iマネージャを制御プロセッサにおくとともに、入力サーバ及び出力サーバにおけるS P I通信開始タイミングを以下のように変更する必要がある。

**【0098】**

入力サーバでは、例えば、入出力プロセッサ上の入力サーバは、制御プロセッサからのXMIT\_INコマンドメッセージに応じて入力ページを送信バッファにコピーした段階で応答メッセージを返し、制御プロセッサ上の入力クライアントは、入出力プロセッサからの応答メッセージに応じてS P I通信を開始して、S P I通信が終了したら、受信バッファから入力ページにデータを転送させる。

20

**【0099】**

出力サーバでは、例えば、制御プロセッサ上の出力クライアントは、RECV\_OUTコマンドメッセージの送信前にS P I通信を開始し、入出力プロセッサ上の出力サーバは、制御プロセッサからのRECV\_OUTコマンドメッセージに応じて、受信バッファのデータを出力ページに転送させる。

**【0100】**

また、本実施の形態では、1本のS C Iケーブルと1本のS P Iケーブルとで2つのプロセッサ間を接続しているが、2つのプロセッサ間の2種類のケーブルの本数の組み合わせは、必ずしも、この通りである必要はない。例えば、2つのプロセッサ間を接続するS P Iケーブルを複数本設けることによって、複数のサーバが、それぞれ、個別のS P Iケーブルを介して提供されるようにしてもよい。また、2つのプロセッサ間を接続するS C Iケーブルを複数本設けてもよい。図12に、2つのプロセッサ間を2本のS C Iケーブル4, 4'と1本のS P Iケーブル5とで接続した場合の構成例を示す。S C Iドライバは、メッセージを受信すると、そのメッセージの送信完了確認のために、S C Iコマンド領域(図5の71)を用いてACKまたはNAKを返すことがあるが、S C Iケーブル上でメッセージが化けると、そのメッセージが、プロセッサのメッセージ(コマンド、応答)なのか、S C Iドライバの確認メッセージ(ACK、NAK)なのか判定できない。上述のように、2つのプロセッサ2, 3間のS C Iケーブルを2本にすれば、一方のS C Iケーブル4を、制御プロセッサのメッセージ送信及び確認受信用、他方のS C Iケーブル4'を、入出力プロセッサのメッセージ送信及び確認受信用に割り当てることができる。このようにすることにより、コマンドメッセージが化けてしまっても、S C Iドライバのエラーなのか通信サーバ及び通信クライアントのエラーなのかを判定し、エラーを切り分けることができる。このため、速やかにエラー処理を行うことができる。

30

40

**【0101】**

また、本実施の形態ではS P Iを用いているが、他のシングルマスタ型シリアル通信インタフェース、マルチマスタ型シリアル通信インタフェースを用いてもよい。

**【0102】**

また、本実施の形態ではS C Iを用いているが、他のマルチマスタ型双方向シリアル通信

50

インタフェース、パラレル通信インタフェースを用いてもよい。例えば、IEEE1394を用いる場合には、メッセージをIEEE1394データパケットに入れて送受信する。また、図14に示したようにDI, DOを用いる場合には、1本以上のDIを用いて、それぞれの信号線にコマンドを割り当て、8本ずつのDI及びDOを用いて、それぞれの信号線に引数の1バイトのデータを割り当てる。

【0103】

以上においては、マルチプロセッサシステムにより機能分散システムを実現する場合を説明したが、前述したように、同様なハードウェアを有するマルチプロセッサシステムによって負荷分散システムを実現することもできる。負荷分散システムを実現する場合には、2つのプロセッサにほぼ均等な負荷がかかるように、2つのプロセッサにジョブを分担させるようにすればよい。

10

【0104】

また、以上においては、自動車制御用のコントローラを例に挙げたが、本実施の形態に係るマルチプロセッサシステムは、自動車制御用のコントローラに限らず、例えば、ファクトリオートメーションのコントローラ等として用いることもできる。

【0105】

【発明の効果】

本発明によれば、マルチプロセッサシステムの処理能力の向上が図られる。

【図面の簡単な説明】

【図1】本発明の一実施形態に係るマルチプロセッサシステムのハードウェア構成図である。

20

【図2】本発明の実施の一形態に係る2つのプロセッサのバッファのページ構成を概念的に示した図である。

【図3】SPI通信データのフォーマットを概念的に示した図である。

【図4】本発明の一実施形態に係るマルチプロセッサシステムのソフトウェア構成図である。

【図5】コマンドメッセージのフォーマットを概念的に示した図である。

【図6】コマンドメッセージに含ませるデータの一例を示した図である。

【図7】本発明の一実施形態に係る入力サービスを説明するための図である。

【図8】本発明の一実施形態に係る出力サービスを説明するための図である。

30

【図9】本発明の一実施形態に係る共有メモリサービスを説明するための図である。

【図10】本発明の一実施形態に係る共有メモリサービスを説明するための図である。

【図11】本発明の一実施形態に係るマルチプロセッサシステムのハードウェアの概略構成図である。

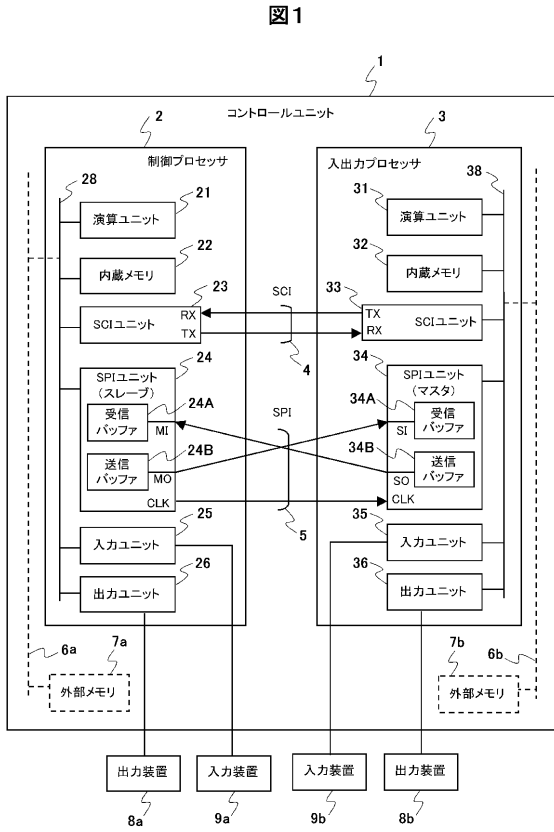
【図12】本発明の一実施形態に係るマルチプロセッサシステムのハードウェアの概略構成図である。

【符号の説明】

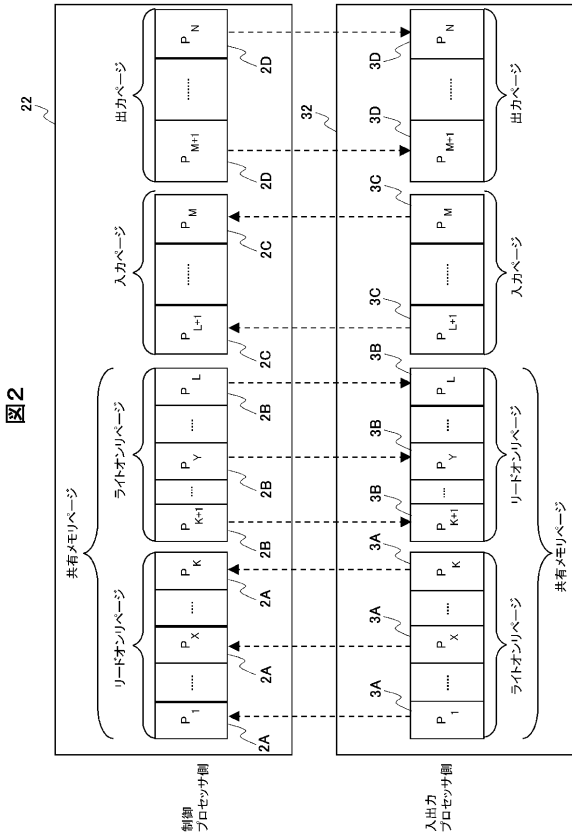
1...自動車制御用のコントローラ、2,3...プロセッサ、4...SCIケーブル、5...SPIケーブル、8...出力装置、9...入力装置、21,31...演算ユニット、22,32...内蔵メモリ、23,33...SCIユニット、24,34...SPIユニット、25,35...入力ユニット、26,36...出力ユニット、200A~200M...制御プログラム、210...データ通信部、220...コマンド通信部、300A~300M...アプリケーションプログラム、310...データ通信部、320...コマンド通信部、212,313...SPIDライバ、225,325...SCIDライバ、350...入出力ドライバ

40

【 図 1 】



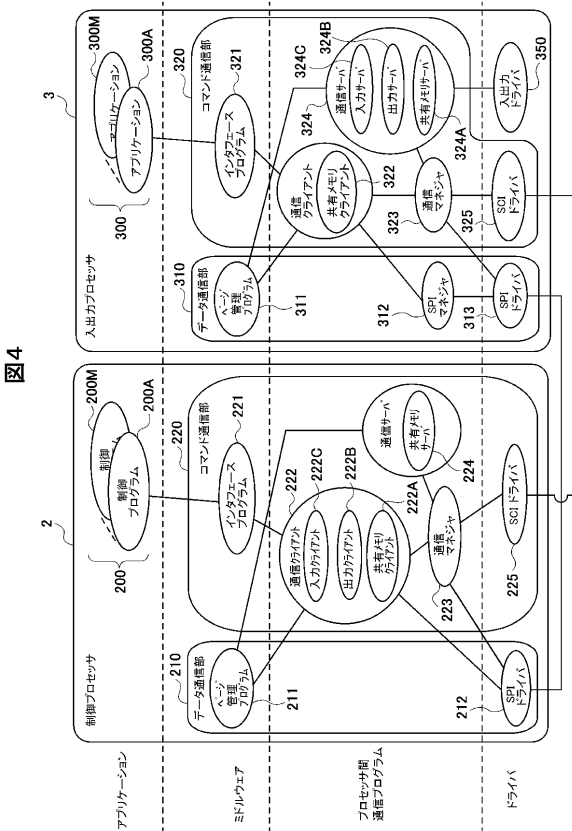
【 図 2 】



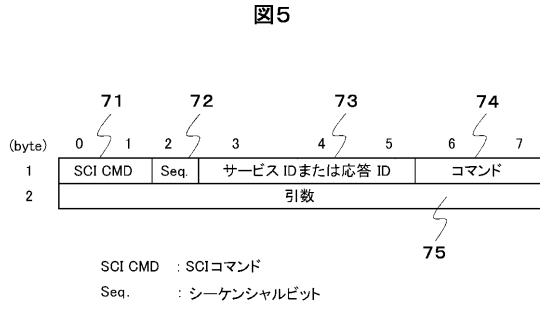
【 図 3 】



【 図 4 】



【 図 5 】

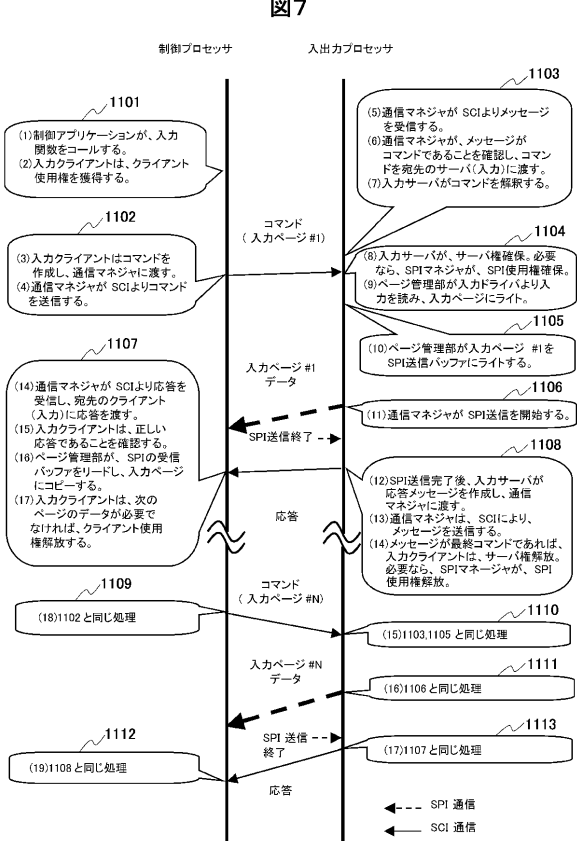


【 図 6 】

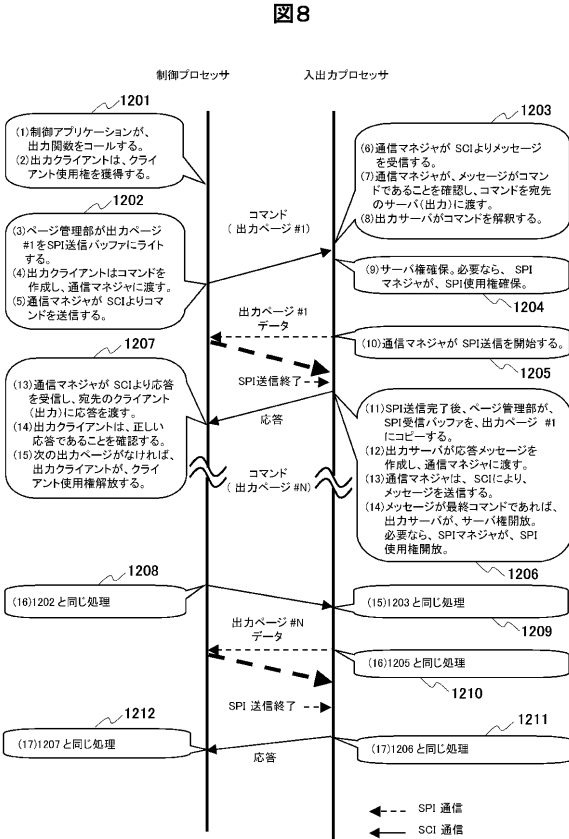
図6

	サービス ID または 応答 ID	コマンド		引数
		ID	名前	
(1)	0x0 (入力)	0x0	XMIT_IN	bit1 - bit7 : ページ ID bit 0 : 終了フラグ
(2)	0x1 (出力)	0x0	RECV_OUT	終了フラグ
(3)	0x2 (共有メモリ)	0x0	SYNC_PAGE	ページ ID
		0x1	READ_PAGE	なし
(4)	0x7 (応答)	0x0	ACK	リターンコード
		0x1	NAK	エラーコード

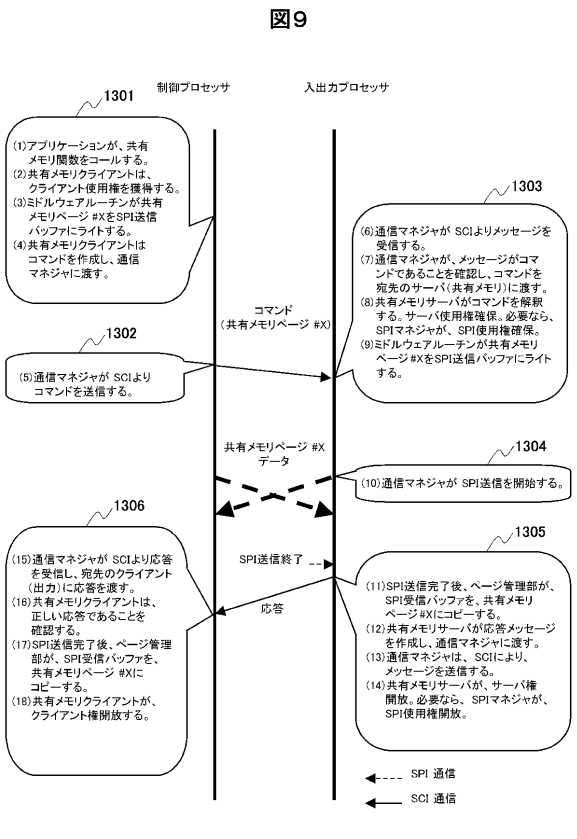
【 図 7 】



【 図 8 】

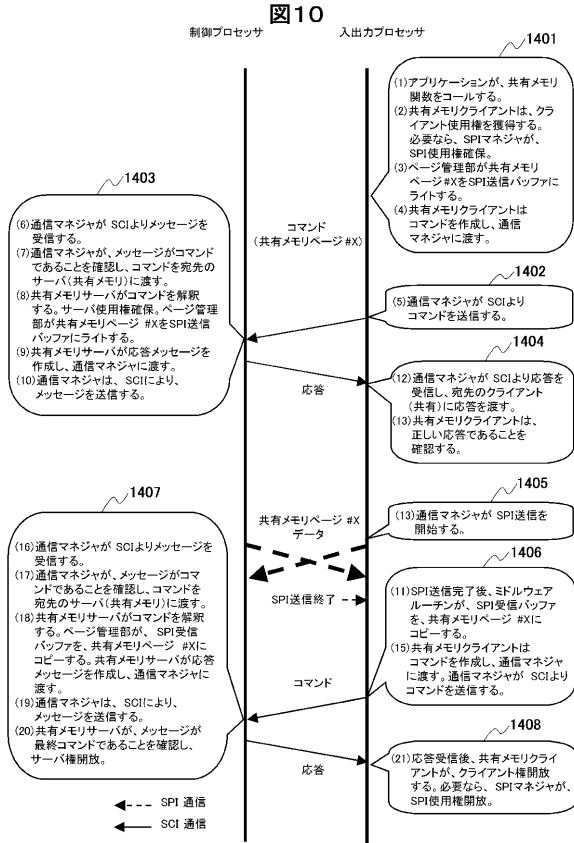


【 図 9 】

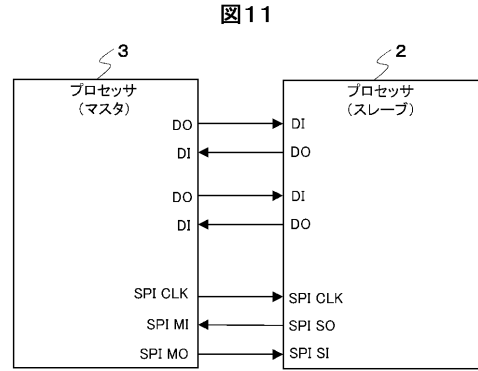




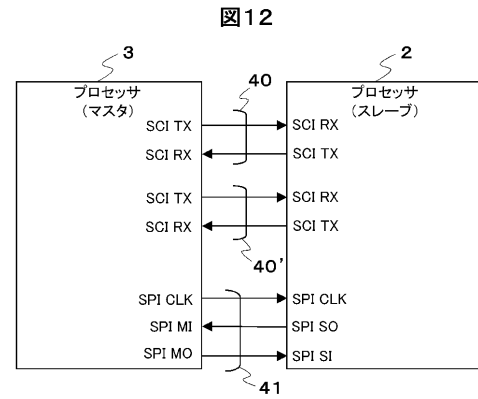
【 図 1 0 】



【 図 1 1 】



【 図 1 2 】



---

フロントページの続き

(72)発明者 金川 信康

茨城県日立市大みか町七丁目1番1号 株式会社日立製作所 日立研究所内

(72)発明者 横山 孝典

茨城県ひたちなか市大字高場2520番地 株式会社日立製作所 自動車機器グループ内

(72)発明者 渡部 満

茨城県ひたちなか市大字高場2520番地 株式会社日立製作所 自動車機器グループ内

審査官 石井 茂和

(56)参考文献 特開平07-084962(JP,A)

特開昭63-024741(JP,A)

特開2000-173173(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 13/38

WPI(DIALOG)