

(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2005-293609
(P2005-293609A)

(43) 公開日 平成17年10月20日(2005.10.20)

(51) Int. Cl. ⁷	F I	テーマコード (参考)
G06F 9/445	G06F 9/06 610J	5B054
G06F 1/24	G06F 1/00 350A	5B076

審査請求 有 請求項の数 3 O L (全 24 頁)

(21) 出願番号	特願2005-139979 (P2005-139979)	(71) 出願人	000005223 富士通株式会社
(22) 出願日	平成17年5月12日 (2005.5.12)		神奈川県川崎市中原区上小田中4丁目1番1号
(62) 分割の表示	特願2000-6823 (P2000-6823)の分割	(74) 代理人	100090273 弁理士 國分 孝悦
原出願日	平成12年1月14日 (2000.1.14)	(72) 発明者	内海 祐之 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		(72) 発明者	廣瀬 佳生 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		Fターム(参考)	5B054 AA01 AA03 CC01 5B076 BB02 BB18

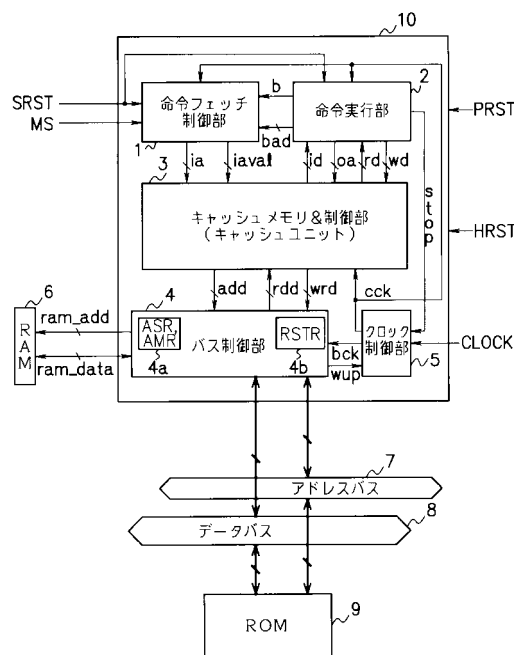
(54) 【発明の名称】 初期化方法

(57) 【要約】

【課題】 プロセッサのソフトウェア的な初期化処理をより高速に行うことができるようにする。

【解決手段】 マスタープロセッサとスレーブプロセッサとの間で通信を行うシステムにおいて、リセット信号に応じて、第1の初期化処理を行った後で初期化プログラムを使用してソフトウェア的な第2の初期化処理を行う初期化方法が提供される。マスタープロセッサが、リセット信号の種類に基づいて、初期化プログラムがローカルメモリ(6)又は外部メモリ(9)の何れにあるかを設定し、スレーブプロセッサは、選択されたローカルメモリ又は外部メモリから初期化プログラムを読み出す。

【選択図】 図1



本実施形態のプロセッサの構成例

【特許請求の範囲】

【請求項 1】

マスタープロセッサとスレーブプロセッサとの間で通信を行うシステムにおいて、リセット信号に応じて、第 1 の初期化処理を行った後で初期化プログラムを使用してソフトウェア的な第 2 の初期化処理を行う初期化方法であって、

前記マスタープロセッサが、前記リセット信号の種類に基づいて、初期化プログラムがローカルメモリ又は外部メモリの何れにあるかを設定し、

前記スレーブプロセッサは、前記選択されたローカルメモリ又は外部メモリから初期化プログラムを読み出すことを特徴とする初期化方法。

【請求項 2】

前記リセット信号の情報を格納するリセットレジスタを備えることを特徴とする請求項 1 に記載の初期化方法。

【請求項 3】

前記リセット信号は、初期起動リセット信号、再起動リセット信号又はプログラムリセット信号の何れかであることを特徴とする請求項 1 又は請求項 2 に記載の初期化方法。

10

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、初期化方法に関する。

20

【背景技術】

【0002】

ある装置に備えられたマイクロプロセッサは、装置の電源が投入されると、外部から与えられるリセット信号に応じてプロセッサ内部を初期化するリセット動作を行う。このリセット動作では、プロセッサ内部に備えられた記憶素子の記憶内容を初期化するハードウェア的な初期化処理の後、外部メモリから初期化処理用のプログラムを読み込んで実行することにより、種々のアプリケーションプログラムを実行可能な状態に設定するソフトウェア的な初期化処理が行われる。

【0003】

上記ハードウェア的な初期化処理は、リセット信号のアサートを契機として行われる。また、上記ソフトウェア的な初期化処理は、リセット信号のネゲートを契機として、リセットベクタで指定されるアドレスから初期化処理用プログラムの命令をフェッチすることによって行われる。通常、この初期化処理用プログラムは、プロセッサの外部にバスを介して接続されたリード・オンリ・メモリ（以下、ROM）に格納されている。

30

【0004】

すなわち、ソフトウェア的な初期化処理では、プロセッサがバスを介してROMに読み出し要求アドレスを与えることで、初期化処理用プログラムの命令列がROMから順次読み出されてバスを介してプロセッサに供給される。これを受け取ったプロセッサは、供給された命令に従って、プロセッサ内部に備えられている各種設定用レジスタにシステムに応じた適当な値を設定することにより、アプリケーションプログラムを実行可能な環境を整備する。

40

【0005】

ところで、このようなリセット動作を行うプロセッサを共有バスに複数個接続し、そのうちの 1 つをマスタープロセッサ、その他をスレーブプロセッサとして機能させるようにしたマルチプロセッサを構成するシステムが存在する。このようなマルチプロセッサシステム全体をリセットするためには、全てのプロセッサにおいてリセット動作を行う必要がある。

【0006】

そのために、まず全てのプロセッサに対してリセット信号がアサートされ、全てのプロセッサにおいてハードウェア的な初期化処理が行われる。そして、プロセッサ内部の記憶

50

素子の記憶内容を初期化するために必要十分な時間だけリセット信号がアサートされた後、このリセット信号は各プロセッサに対してネゲートされる。

【0007】

このリセット信号のネゲートを契機として、各プロセッサは続いてソフトウェア的な初期化処理を行うために、リセットベクタで指示されるアドレスに従ってROMに対するアクセスを一斉に開始しようとする。しかし、共有バス上に接続されたROMに対して複数のプロセッサが同時にアクセスすることはできない。そのため、各プロセッサは、共有バスのアービトレーション（バス調停）を介してROMに順番にアクセスし、ソフトウェア的な初期化処理のために必要なプログラムの命令列を順に読み出すことになる。

【0008】

また、上述のようなリセット動作を行うプロセッサを組み込んだシステムにおいて、電源投入後においてもプロセッサが動作していない状態では、当該プロセッサでのクロック供給を停止することにより、消費電力の低減化を図ったシステムも提案されている。例えば、上述のようなマルチプロセッサにより構成されるシステムでは、マスタープロセッサが動作してスレーブプロセッサが動作していない状態が存在するが、この場合にはスレーブプロセッサでのクロック供給を停止することにより、低消費電力化を図っている。

10

【0009】

プロセッサのクロック供給が停止されている状態で、その後そのプロセッサを使用する状態が発生したときは、再びクロックが供給されてプロセッサの起動が行われる。このとき、再起動されるプロセッサでは、リセット信号のアサートに応じてハードウェア的な初期化処理を行った後、リセット信号のネゲートに応じてバスを介してROMにアクセスし、必要な初期化処理用プログラムの命令列を読み込んでソフトウェア的な初期化処理を実行することになる。

20

【発明の開示】

【発明が解決しようとする課題】

【0010】

しかしながら、上記従来のシステムにおいては、ハードウェア的な初期化処理の後に続いて行われるソフトウェア的な初期化処理では、その初期化処理に必要なプログラムを、バスを介して低速なメモリデバイスであるROMにアクセスして取得しなければならず、そのリセット動作に多くの時間がかかっていた。

30

【0011】

特に、低消費電力化のためにプロセッサのクロック供給を一時的に停止するようにしたシステムにおいて、プロセッサの停止と起動とが頻繁に行われると、低速なROMへのアクセスがプロセッサ起動の度に実行されるため、システムの起動待ちという無駄な時間が増大するという問題があった。

【0012】

また、複数のプロセッサが共有バス上に接続されたマルチプロセッサシステムにおいては、それぞれのプロセッサが実行する初期化処理用のプログラムを記憶したROMが共有バスに接続されているため、各プロセッサからROMへのアクセスが競合した場合は、バス調停が行われて各プロセッサが低速なROMに順次アクセスすることになる。そのため、全てのプロセッサの初期化に要する時間が大幅に増大してしまうという問題があった。

40

【0013】

さらに、マルチプロセッサシステムでは、それぞれのプロセッサには異なる役割が与えられるのが通常であり、ソフトウェア的な初期化処理の内容もそれぞれ異なるものとなる。そのため、初期化処理のために使用するプログラムもそれぞれのプロセッサ毎に異なるものとなる。したがって、ROM内に格納された異なるプログラムのうち、どのプログラムをどのプロセッサが実行するのかを判別する機構が必要となり、システムの構成が煩雑になるという問題があった。

【0014】

本発明は、このような問題を解決するために成されたものであり、プロセッサのソフト

50

ウェア的な初期化処理をより高速に行うことができるようにすることを目的とする。

また、本発明は、プロセッサのソフトウェア的な初期化処理を行うための機構をより簡素化することができるようにすることをも目的とする。

【課題を解決するための手段】

【0015】

本発明の初期化方法は、マスタープロセッサとスレーブプロセッサとの間で通信を行うシステムにおいて、リセット信号に応じて、第1の初期化処理を行った後で初期化プログラムを使用してソフトウェア的な第2の初期化処理を行う初期化方法であって、前記マスタープロセッサが、前記リセット信号の種類に基づいて、初期化プログラムがローカルメモリ又は外部メモリの何れにあるかを設定し、前記スレーブプロセッサは、前記選択されたローカルメモリ又は外部メモリから初期化プログラムを読み出すことを特徴とする。

10

【発明の効果】

【0016】

電源投入後において内部クロックの供給が停止された後の再起動時には、高速なローカルメモリから再起動用の初期化プログラムを読み出してプロセッサの起動を行うことができ、その再起動処理を高速に行うことができる。これにより、システムの消費電力を抑制するとともにクロック停止状態からの高速な復帰を実現することができるようになる。また、マルチプロセッサシステムを構成した場合に、どのプロセッサがどのプログラムを実行するのかを判別するための機構も不要となり、システムの構成を簡素化することができる。

20

【発明を実施するための最良の形態】

【0017】

以下、本発明の一実施形態を図面に基づいて説明する。

図1は、本発明によるデータ処理システムの第1の実施形態に係るプロセッサの構成例を示すブロック図である。

【0018】

本実施形態のプロセッサ10は、メモリからプログラムの命令を読み出してフェッチする動作を制御する命令フェッチ制御部1、フェッチされた命令を解釈して実行する命令実行部2、メモリの内容のコピーを保持して高速なメモリアクセスをサポートするキャッシュメモリおよびその制御部（以下、キャッシュユニット3と略す）、ローカルなメモリであるランダム・アクセス・メモリ（RAM）6およびアドレスバス7やデータバス8などの外部バスの制御を行うバス制御部4、与えられる外部クロックCLOCKに基づいて内部クロックbc k, cc kの生成および分配を行うクロック制御部5を備えて構成されている。

30

【0019】

外部バス7, 8には、外部メモリであるROM9が接続されており、プロセッサ10のソフトウェア的な初期化処理を行うために必要な初期化プログラムがここにあらかじめ記憶されている。なお、ここでは外部メモリとしてROM9を想定しているが、これに限られない。また、RAM6は、通常はプロセッサ10のワークメモリとして使用されるものであり、必要に応じてROM9から外部バス7, 8を介して初期化プログラムが転送され、格納される。

40

【0020】

なお、キャッシュユニット3内のキャッシュメモリは、プロセッサ10に接続された外部メモリの内容のコピーを保持する、高速ではあるが容量の小さなバッファメモリである。外部メモリの内容のコピーがキャッシュメモリに存在していた場合、命令実行部2に対するデータはこのキャッシュメモリから供給される。本発明に関する限り、キャッシュユニット3の有無は本質的な意味を持たないため、詳細な説明は割愛する。本実施形態では、キャッシュメモリとしては、例えばストアスルー型のものを用いるものとする。

【0021】

上記命令フェッチ制御部1は、プロセッサ10が実行すべき初期化プログラムの命令が

50

格納されているメモリのアドレスを生成し、それを要求命令アドレス $i a$ として命令フェッチ要求 $i a v a l$ と共にキャッシュユニット 3 に発行する。このとき、命令フェッチ制御部 1 は、外部から与えられるアドレス選択信号 $M S$ に基づいて、ローカルメモリである $R A M 6$ または外部メモリである $R O M 9$ の何れかの開始アドレスを選択して要求命令アドレス $i a$ とする。つまり、この命令フェッチ制御部 1 は、本発明のアドレス選択手段を備えている。

【0022】

命令フェッチ制御部 1 から命令フェッチ要求 $i a v a l$ を受け取ったキャッシュユニット 3 は、それに伴って与えられた要求命令アドレス $i a$ を用いて、キャッシュメモリ内に有効なエントリが存在するかどうかを検索する。ここでキャッシュヒットした場合には、キャッシュメモリから該当するデータを読み出して、それを命令データ $i d$ として命令実行部 2 に供給する。一方、キャッシュメモリ内に有効なエントリが存在せず、キャッシュミスが発生した場合には、命令フェッチ要求と要求アドレス $a d d$ とをバス制御部 4 に供給する。

10

【0023】

バス制御部 4 は、受け取った要求アドレス $a d d$ に従って、 $R A M 6$ または $R O M 9$ の何れかに対して選択的にデータの読み出しを要求する。すなわち、このバス制御部 4 は、 $R A M 6$ のアドレス領域を指定する情報を保持するための領域指定レジスタ 4 a を備えており、ここには領域の先頭アドレス $A S R$ および領域サイズ $A M R$ の情報が保持されている。そして、キャッシュユニット 3 から受け取った要求アドレス $a d d$ が、この先頭アドレス $A S R$ および領域サイズ $A M R$ の対によって指定される $R A M 6$ のアドレス領域に含まれているか否かに応じて、 $R A M 6$ または $R O M 9$ の何れかに対してデータの読み出しを要求する。

20

【0024】

図 2 は、上記 $R A M 6$ や $R O M 9$ の他に、プロセッサ 10 内の各種レジスタを含むシステム全体のメモリマップを示す図である。図 2 の例では、アドレス番号が $0x0000_0000 \sim 0x01FF_FFFF$ までは $R A M 6$ に割り当てられ、 $0xFE00_0000 \sim 0xFF00_0000$ まではプロセッサ 10 内の各構成（命令フェッチ制御部 1、命令実行部 2、キャッシュメモリ 3、バス制御部 4 およびクロック制御部 5）に備えられる各種レジスタに割り当てられ、 $0xFF00_0000 \sim 0xFFFF_FFFF$ までは $R O M 9$ に割り当てられている。なお、 $0x01FF_FFFF \sim 0xFE00_0000$ までのアドレス番号は未使用である。

30

【0025】

この図 2 の例に従えば、領域指定レジスタ 4 a の先頭アドレス $A S R$ の情報として $0x0000_0000$ のアドレス番号が保持され、領域サイズ $A M R$ の情報として $01FF_FFFF$ の領域サイズが保持されている。したがって、バス制御部 4 は、キャッシュユニット 3 から受け取った要求アドレス $a d d$ が $0x0000_0000 \sim 0x01FF_FFFF$ の範囲内に含まれていれば、アドレス線 ram_add に要求アドレス $a d d$ を載せて $R A M 6$ に対してデータの読み出しを要求する。

【0026】

また、バス制御部 4 は、受け取った要求アドレス $a d d$ が、領域指定レジスタ 4 a 内の先頭アドレス $A S R$ および領域サイズ $A M R$ の対によって指定される上記 $0x0000_0000 \sim 0x01FF_FFFF$ のアドレス領域に含まれておらず、かつ、メモリマップされたプロセッサ 10 内の各種レジスタの領域を表す $0xFE00_0000 \sim 0xFF00_0000$ のアドレス領域にも含まれていなかった場合は、アドレスバス 7 に要求アドレス $a d d$ を載せて $R O M 9$ に対してデータの読み出しを要求する。

40

【0027】

$R A M 6$ または $R O M 9$ に対して要求されたデータは、それぞれのメモリから読み出され、データ線 ram_data またはデータバス 8 を介してバス制御部 4 に送られてくる。このデータを受け取ったバス制御部 4 は、それをリードデータ $r d d$ としてキャッシュユニット 3 に供給する。リードデータ $r d d$ を受けたキャッシュユニット 3 は、そのデータを命

50

令データ *id* として命令実行部 2 に供給し、必要に応じてキャッシュメモリにこの命令データを登録する。

【0028】

以上のようにして命令データ *id* の供給を受けた命令実行部 2 は、その命令を解釈して実行する。この実行の過程で RAM 6、ROM 9 あるいはプロセッサ 10 内の各種レジスタなどに格納されたデータを参照する必要がある場合、命令実行部 2 は、要求データアドレス *oa* を伴ってデータフェッチ要求をキャッシュユニット 3 に発行する。

【0029】

このときのデータフェッチに関して、これ以降の動作は命令フェッチの場合とほぼ同様である。ただし、メモリマップされたレジスタのアドレス領域 (0xFE00__0000 ~ 0xFF00__0000) を要求アドレス *add* が指定していた場合、バス制御部 4 はそのレジスタの値 (領域指定レジスタ 4 *a* や後述するリセットレジスタ (RSTR) 4 *b* の値) を読み出してリードデータ *rd* としてキャッシュユニット 3 に返す。また、RAM 6、ROM 9 あるいはレジスタからバス制御部 4 を介してキャッシュユニット 3 に返されたデータは、最終的にリードデータ *rd* として命令実行部 2 に供給される。

10

【0030】

また、命令実行部 2 における命令実行の過程で、RAM 6 あるいはレジスタなどのメモリの内容を更新する必要がある場合には、要求データアドレス *oa* とライトデータ *wd* とを伴ってデータライト要求をキャッシュユニット 3 に発行する。キャッシュユニット 3 は、キャッシュヒットした場合にはキャッシュメモリにライトデータ *wd* を書き込むとともに、バス制御部 4 に対して要求アドレス *add* を伴うデータライト要求とライトデータ *wrd* とを発行する。

20

【0031】

このデータライト要求を受け取ったバス制御部 4 は、受け取った要求アドレス *add* が、領域指定レジスタ 4 *a* 内の先頭アドレス ASR および領域サイズ AMR の対によって指定される RAM 6 のアドレス領域に含まれていた場合には、アドレス線 *ram__add* に要求アドレス *add*、データ線 *ram__data* にライトデータ *wrd* を載せて RAM 6 に対してデータの書き込みを要求する。また、受け取った要求アドレス *add* が、領域指定レジスタ 4 *a* やリセットレジスタ 4 *b* などのメモリマップされたレジスタのアドレス領域を指定していた場合は、バス制御部 4 は、その要求アドレス *add* で示されるレジスタに対してライトデータ *wrd* の書き込みを行う。

30

【0032】

また、バス制御部 4 は、受け取った要求アドレス *add* が、領域指定レジスタ 4 *a* 内の先頭アドレス ASR および領域サイズ AMR の対によって指定される RAM 6 のアドレス領域に含まれておらず、かつ、メモリマップされたレジスタのアドレス領域にも含まれていなかった場合は、アドレスバス 7 に要求アドレス *add*、データバス 8 にライトデータ *wrd* を載せて ROM 9 に対してデータの書き込みを要求する。

【0033】

本実施形態のプロセッサ 10 は、外部から与えられるクロック CLOCK によって駆動されている。クロック制御部 5 は、この外部クロック CLOCK をもとに第 1、第 2 の内部クロック *bc k*, *cc k* を発生する。そして、バス制御部 4 に対しては第 1 の内部クロック *bc k* を供給し、それ以外の命令フェッチ制御部 1、命令実行部 2 およびキャッシュユニット 3 には第 2 の内部クロック *cc k* を供給している。

40

【0034】

このクロック制御部 5 に対しては、内部クロック *bc k*, *cc k* の供給を停止させることを指示するクロック停止信号 *stop* が命令実行部 2 から必要に応じて与えられる。すなわち、このプロセッサ 10 は、命令実行部 2 においてクロック停止用の命令を実行することにより、内部クロック *bc k*, *cc k* の供給を自ら停止させることができるようになっていいる。クロックの停止モードとして、2 つの内部クロック *bc k*, *cc k* を共に停止させる第 1 のモードと、第 2 の内部クロック *cc k* のみを停止させる第 2 のモードとの 2

50

つがある。

【0035】

一旦停止させたクロック供給を再開するためには、初期起動リセット信号P R S T（本発明の第1のリセット信号）あるいは再起動リセット信号H R S T（本発明の第2のリセット信号）の何れかをアサートする方法か、バス制御部4が備えるリセットレジスタ4bに図示しない外部バスマスタから書き込みを行い、上記再起動リセット信号H R S Tあるいは次に述べるプログラムリセット信号S R S Tに相当するリセット信号（起動信号w u p）をバス制御部4の内部で生成してアサートする方法がある。後者の方法を採用する場合には、第1の内部クロックb c kはバス制御部4に供給され続けている必要があるため、この場合は上述の第2のモードが採用される。

10

【0036】

なお、図1中には詳細に示していないが、上記初期起動リセット信号P R S Tおよび再起動リセット信号H R S Tは、プロセッサ10内の各構成（命令フェッチ制御部1、命令実行部2、キャッシュユニット3、バス制御部4およびクロック制御部5）に与えられるようになっている。本実施形態では、これらのリセット信号の他に、内部クロックb c k, c c kの供給停止および再開とは関係なく独自にリセット動作を行うプログラムリセット信号S R S T（本発明の第3のリセット信号）も用意されている。このプログラムリセット信号S R S Tは、例えばデバッグを行うときなどに利用され、命令フェッチ制御部1および命令実行部2に対してのみ供給されるようになっている。

【0037】

図3は、上記命令フェッチ制御部1の詳細な構成例を示す図である。図3において、要求命令アドレスi aは、命令アドレスレジスタ11に保持され、キャッシュユニット3へと供給される。プロセッサ10が順次命令を実行している状態では、次の要求命令アドレスi aとして、命令アドレスレジスタ11に保持されている現在のアドレスに対して値"16"が加算器12で加算され、その結果がセクタ13, 14にて選択されて命令アドレスレジスタ11に保持される。このインクリメント値"16"は、プロセッサ10の命令データバスのバス幅が指定されている。

20

【0038】

また、プロセッサ10が分岐命令を実行した場合には、命令実行部2から分岐したことを通知する信号bがアサートされる。この場合は、次の要求命令アドレスi aとして、命令実行部2から送られてくる分岐先アドレスb a dがセクタ13にて選択されて、セクタ14を介して命令アドレスレジスタ11に保持される。

30

【0039】

また、初期起動リセット信号P R S T、再起動リセット信号H R S T、プログラムリセット信号S R S Tおよびリセットレジスタ4bからの起動信号w u p（以下、これら4つの信号を総称して単にリセット信号と呼ぶ）によりプロセッサ10の初期化が行われている状態では、外部から与えられるアドレス選択信号M Sに応じてセクタ15により選択された何れかのアドレス（0x0000__0000または0xFF00__0000）が、上記リセット信号に応じてセクタ14にて選択されて命令アドレスレジスタ11に保持される。

【0040】

例えば、初期起動リセット信号P R S Tが入力される時、つまり、装置の電源が投入されたときには、アドレス選択信号M Sの値が"1"に設定され、R O M 9のアドレス領域の先頭アドレスを表す固定値"0xFF00__0000"が選択されて命令アドレスレジスタ11に保持される。一方、再起動リセット信号H R S Tが入力される時、つまり、内部クロックb c k, c c kの供給停止状態からクロック供給を再開するときには、アドレス選択信号M Sの値が"0"に設定され、R A M 6のアドレス領域の先頭アドレスを表す固定値"0x0000__0000"が選択されて命令アドレスレジスタ11に保持される。

40

プログラムリセット信号S R S Tや起動信号w u pが入力されたときも、再起動リセット信号H R S Tが入力されたときと同様である。

【0041】

50

これにより、リセット信号がネゲートされて初期化プログラムの命令フェッチを最初に行おうとした際の要求命令アドレス $i a$ は、ROM 9 の先頭アドレス "0xFF00_0000" あるいは RAM 6 の先頭アドレス "0x0000_0000" の何れかとなる。このように、アドレス選択信号 MS により選択動作を行うセクタ 14, 15 によって本発明のアドレス選択手段が構成される。

【0042】

また、命令フェッチ制御ロジック 16 は、与えられるリセット信号に応じて、プロセッサ 10 の命令実行状況やキャッシュ制御部の状況等に従って命令フェッチ要求 $i a v a l$ の発行の可否を決定する。命令アドレスレジスタ 11 内に保持された要求命令アドレス $i a$ は、キャッシュユニット 3 に対して命令フェッチ要求 $i a v a l$ がアサートされてこれが受け付けられるまでは、更新されることはない。

10

【0043】

図 4 は、バス制御部 4 の領域指定レジスタ 4 a を含む部分の詳細な構成例を示す図であり、アドレスおよびデータの流れを示している。図 4 に示すように、バス制御部 4 は、バス I/F 21 と、メモリ制御部 22 と、外部バス制御部 23 とを備えている。上記バス I/F 21 は、上述の領域指定レジスタ 4 a を有するバスコマンド制御部 24 と、リードデータバス 25 と、ライトデータバス 26 とを備えている。

【0044】

また、上記メモリ制御部 22 は、メモリ - バス I/F (MBI) 27 と、メモリ I/F (MI) 28 と、メモリ - 外部バス I/F (MEI) 29 とを備えている。また、上記外部バス制御部 23 は、外部バス - バス I/F (EBI) 30 と、外部バス - メモリ I/F (EMI) 31 と、外部バス I/F 32 とを備えている。

20

【0045】

以下に、命令フェッチ時におけるバス制御部 4 の動作を説明する。キャッシュユニット 3 からの命令フェッチ要求を受けたバス制御部 4 は、バスコマンド制御部 24 において、当該命令フェッチ要求と共に与えられた要求アドレス $a d d$ と、領域指定レジスタ 4 a にあらかじめ設定されている先頭アドレス ASR および領域サイズ AMR の値とを比較し、要求アドレス $a d d$ が RAM 6 のアドレス領域内のものかどうかを判定する。

【0046】

このアドレス比較の結果、命令フェッチ要求が RAM 6 に対する要求であった場合、バスコマンド制御部 24 は、メモリ制御部 22 に対して命令フェッチ要求を発行する。この命令フェッチ要求をメモリ - バス I/F 27 で受けたメモリ制御部 22 は、メモリ I/F 28 を介して接続された RAM 6 をアクセスして、上記要求アドレス $a d d$ に対応する命令データを読み出す。読み出された命令データは、メモリ - バス I/F 27 およびリードデータバス 25 を介してキャッシュユニット 3 へと送り返され、最終的に命令実行部 2 へと送られる。

30

【0047】

一方、上記のアドレス比較の結果、命令フェッチ要求が外部バス 7, 8 上に接続された ROM 9 に対する要求であった場合、バスコマンド制御部 24 は、外部バス制御部 23 に対して命令フェッチ要求を発行する。この命令フェッチ要求を外部バス - バス I/F 30 で受けた外部バス制御部 23 は、バスの調停を行い、外部バス 7, 8 上にメモリトランザクションを発行してその応答を待つ。

40

【0048】

そして、外部バス 7, 8 上の ROM 9 から読み出された命令データが外部バス 7, 8 に出力されると、外部バス制御部 23 はその命令データを外部バス I/F 32 を介して取り込む。こうして取り込まれた命令データは、外部バス - バス I/F 30 およびリードデータバス 25 を介してキャッシュユニット 3 へと送り返され、最終的に命令実行部 2 へと送られる。

【0049】

図 5 は、上記図 4 に示したバスコマンド制御部 24 内でアドレス判定を行う部分の詳細

50

な構成例を示す図である。図5に示すように、バスコマンド制御部24は、上述の領域指定レジスタ4aと、2つの比較器41, 42と、2つのバッファ43, 45と、NANDゲート44とを備えている。

【0050】

一方の比較器41は、メモリマップされたレジスタのアドレス領域と、キャッシュユニット3から与えられる要求アドレスaddとを比較する。この比較の結果、メモリマップされたレジスタのアドレス領域内に要求アドレスaddが含まれている場合(信号match1がアサートされた場合)には、バッファ43を介して命令フェッチ要求のコマンドをレジスタに対して発行する。

【0051】

また、他方の比較器42は、キャッシュユニット3から与えられる要求アドレスaddと、領域指定レジスタ4a内の先頭アドレスASRおよび領域サイズAMRの対により指定されるRAM6のアドレス領域とを比較する。この比較の結果、先頭アドレスASRに設定された開始アドレスから領域サイズAMRで指定されるサイズ内に要求アドレスaddが含まれている場合(信号match2がアサートされた場合)には、バッファ45を介して命令フェッチ要求のコマンドをRAM6に対して発行する。

【0052】

さらに、上記2つの比較器41, 42における比較の結果、キャッシュユニット3から与えられる要求アドレスaddが、メモリマップされたレジスタのアドレス領域にもRAM6のアドレス領域にも含まれていない場合(信号match1, 2が両方ともアサートされない場合)には、NANDゲート44を介して命令フェッチ要求のコマンドを外部バス7, 8に対して発行する。

【0053】

図6は、バス制御部4のリセットレジスタ4bを含む部分の詳細な構成例を示す図であり、当該リセットレジスタ4bに対するデータ書き込みの流れを示している。図6に示すように、リセットレジスタ4bは、PD, HD, SD, HS, SSという5ビット分のレジスタから構成されている。

【0054】

このうちPD, HD, SDの各ビットは、直前のリセット要因を示すためのレジスタであり、PDビットは電源投入時における初期起動リセット、HDビットはクロック供給再開時における再起動リセット、SDビットはデバッグ時等におけるプログラム再起動リセットであることを示す。これらの3ビットは、リセット要因の識別情報を保持する本発明の情報保持手段に相当する。

【0055】

また、残りのHS, SSの2ビットは、図示しない外部バスマスタからここに書き込みを行うことにより、再起動リセット信号HRSTあるいはプログラムリセット信号SRSTに相当するリセット信号(HRST', SRST')をバス制御部4の内部で発生するためのものであり、本発明のリセット内部発生手段を構成する。すなわち、このHSビットおよびSSビットの出力は、それぞれ再起動リセット信号HRSTおよびプログラムリセット信号SRSTと共にORゲート51, 52に供給され、何れが有効な方が後段に供給される。

【0056】

上記リセットレジスタ4b内のPD, HD, SDの各ビットと、初期起動リセット信号PRST、再起動リセット信号HRSTおよびプログラムリセット信号SRSTとの間には、数個の論理ゲート51~57が適当に配置されており、PD, HD, SDの3つのビットのうち何れか1つのビットのみが"1"に設定されるようになっている。

【0057】

すなわち、初期起動リセット信号PRSTがアサートされた場合にはPDビットが"1"にセットされる。また、外部からの再起動リセット信号HRSTがアサートされた場合、あるいはHSビットに書き込みが行われて内部での再起動リセット信号HRST'がアサ

10

20

30

40

50

ートされた場合には、HDビットが"1"にセットされる。また、外部からのプログラムリセット信号SRSTがアサートされた場合、あるいはSDビットに書き込みが行われて内部でのプログラムリセット信号SRST'がアサートされた場合には、SDビットが"1"にセットされる。

【0058】

また、上記数個の論理ゲート51～57を適当に配置することにより、3種類のリセット間に、

$$PRST > HRST > SRST$$

の優先順位を持たせている。そして、新たにリセット要因が発生するまでは、直前のリセット要因をPD, HD, SDの各ビットに保持し続けるようになっている。

10

【0059】

図7は、クロック制御部5の構成例を示す図である。クロック制御部5は、外部から供給された外部クロックCLOCKから2つの内部クロックbck, cckを発生し、これをプロセッサ10の内部へ供給している。すなわち、本実施形態では、図1に示したように、バス制御部4とその他の部分への内部クロックはbck, cckの別系統で供給されている。

【0060】

また、本実施形態では、命令実行部2からの指示により、プロセッサ10内部へのクロック供給を一時的に停止させることを可能としている。例えば、消費電力を低減させるために、プロセッサ10のアイドル状態が一定時間続くと、命令実行部2からクロック制御部5に対してクロック停止信号stopがアサートされる。

20

【0061】

このとき、バス制御部4も含めたプロセッサ10全体に対する内部クロックbck, cckを共に停止させる第1のモードと、プロセッサ10内のバス制御部4を除く部分に対する第2の内部クロックcckのみを停止させる第2のモードとの何れかを選択することができる。選択されたモードに応じて、命令実行部2からクロック停止信号stopbcあるいはstopcがアサートされる。

【0062】

プロセッサ10の全体を停止させるためのクロック停止信号stopbcは、2つのORゲート61, 65に供給される。ORゲート61を通過したクロック停止信号stopbcは、レジスタ62に保持された後、ANDゲート64の負論理端子に供給され、ANDゲート64の正論理端子に供給される外部クロックCLOCKとの論理積がとられて内部クロックcckとして出力される。

30

【0063】

一方、ORゲート65を通過したクロック停止信号stopbcは、レジスタ66に保持された後、ANDゲート68の負論理端子に供給され、ANDゲート68の正論理端子に供給される外部クロックCLOCKとの論理積がとられて内部クロックbckとして出力される。

【0064】

このような構成により、クロック停止信号stopbcがアサートされたときには、ANDゲート64, 68の負論理端子には共に"1"の信号が入力されるので、2つの内部クロックbck, cckの供給が停止されることとなる。

40

【0065】

また、プロセッサ10内のバス制御部4以外を停止させるためのクロック停止信号stopcは、ORゲート61にのみ供給される。このORゲート61を通過したクロック停止信号stopcは、レジスタ62に保持された後、ANDゲート64の負論理端子に供給され、ANDゲート64の正論理端子に供給される外部クロックCLOCKとの論理積がとられて内部クロックcckとして出力される。

【0066】

このような構成により、クロック停止信号stopcがアサートされたときには、AN

50

Dゲート64の負論理端子には"1"の信号が入力されるので、内部クロックc c kの供給が停止されることとなる。一方、クロック停止信号s t o p cはORゲート65には供給されないので、当該クロック停止信号s t o p cのアサートによってANDゲート68の負論理端子に"1"の信号が入力されることはなく、内部クロックb c kの供給は継続して行われることとなる。

【0067】

なお、上記ORゲート61, 65には、それぞれANDゲート63, 67の出力も供給されている。ANDゲート63は、その正論理端子に供給されるレジスタ62の出力と、負論理端子に供給される起動信号w u pとの論理積をとってORゲート61に供給する。また、ANDゲート67は、その正論理端子に供給されるレジスタ66の出力と、負論理端子に供給される起動信号w u pとの論理積をとってORゲート65に供給する。

10

【0068】

また、上記レジスタ62およびレジスタ66の記憶内容は、初期起動リセット信号P R S T、再起動リセット信号H R S T、あるいは外部バスマスタ等からリセットレジスタ4 b (図6のH Sビット)への書き込みにより発生する内部再起動リセット信号H R S T'をアサートすることによって書き換えられるようになっている。

【0069】

例えば、バス制御部4に対してのみ内部クロックb c kが供給されている状態では、内部再起動リセット信号H R S T'のアサートにより、バス制御部4以外の部分に対する内部クロックc c kの供給を再開できるように構成されている。また、プロセッサ10の全体に対して内部クロックb c k, c c kの供給が停止された状態では、初期起動リセット信号P R S Tあるいは再起動リセット信号H R S Tをアサートすることにより、当該内部クロックb c k, c c kの供給を再開できるように構成されている。

20

【0070】

次に、以上のように構成した本実施形態によるプロセッサ10を単独で用いた場合の起動シーケンスについて、以下に説明する。

まず、装置の電源投入時における初期起動リセット信号P R S Tによる起動処理シーケンスについて説明する。

【0071】

装置の電源投入時には、まず、初期起動リセット信号P R S Tがアサートされてハードウェア的な初期化処理が行われる。このとき、バス制御部4内の領域指定レジスタ4 a やリセットレジスタ4 bを含むプロセッサ10内の各種レジスタの値が、例えば全て"0"に初期化される。

30

【0072】

そして、このハードウェア的な初期化処理を行うのに必要十分な時間だけ初期起動リセット信号P R S Tがアサートされ、電源が安定すると、上記初期起動リセット信号P R S Tがネゲートされる。初期起動リセット信号P R S Tがネゲートされると、続いてソフトウェア的な初期化処理を行うために、リセットベクタで指示されるアドレスに従って初期化プログラムの読み出しが行われる。

【0073】

電源投入直後では、R A M 6に有効な初期化プログラムのデータが保持されていないため、アドレス選択信号M Sの値は"1"に設定されている。これにより命令フェッチ制御部1は、初期アドレス"0xFF00_0000"からの命令フェッチ要求を発行する。

40

【0074】

これに応じてバス制御部4では、与えられた"0xFF00_0000"の要求アドレスa d dが、領域指定レジスタ4 a内の先頭アドレスA S Rおよび領域サイズA M Rにより指定されるR A M 6のアドレス領域に含まれているか否かを判定し、その結果に応じてR A M 6あるいはR O M 9の何れかにアクセスする。この時点では、R A M 6のアドレス領域は初期化されており、与えられた要求アドレスa d dもR O M 9のアドレス領域を指定しているため、バス制御部4は外部バス7, 8上に接続されたR O M 9に対してアクセスするこ

50

とになる。

【0075】

ROM9に格納されている初期化プログラムは、その最初のシーケンスにおいてリセットレジスタ4bの値を参照するように規定されている。電源投入直後においては、リセットレジスタ4b内のPDビットが"1"にセットされていることを確認して、初期起動リセット信号PRSTがネゲートされた場合に必要な処理を含むルーチンを実行する。

【0076】

このようにしてプロセッサ10が起動した後、RAM6をアクセスするために必要なアドレス情報(先頭アドレスASRおよび領域サイズAMR)を領域指定レジスタ4aに設定する。図2の例においては、先頭アドレスASRに"0x0000__0000"、領域サイズAMRに"0x01FF__FFFF"の値を設定し、RAM6のアドレス領域として32Mバイトを実装する。そして、領域指定レジスタ4aに必要な値を設定した後は、アドレス選択信号MSの値を"0"に切り替えるとともに、外部バス7,8上に接続されたROM9あるいはその他のデバイスから再起動用のプログラムをRAM6にロードしておく。

10

【0077】

次に、内部クロックの供給再開時における再起動リセット信号HRSTによる起動処理シーケンスについて説明する。

ここでは一例として、プロセッサ10の起動後、必要とされる処理が一定時間行われなかったことを契機としてプロセッサ10全体に内部クロックbc k, cc kの供給を停止させ、その後必要性が生じた時点で外部からの再起動リセット信号HRSTにより再起動を行う場合の再起動シーケンスを説明する。

20

【0078】

外部より再起動リセット信号HRSTがアサートされると、クロック制御部5内で内部クロックbc k, cc kの供給を停止させているレジスタ(図7のCレジスタ62およびBレジスタ66)がリセットされ、内部クロックbc k, cc kの供給が再開される。これと同時に、メモリアクセスに必要な情報(領域指定レジスタ4a内の先頭アドレスASR、領域サイズAMRなど)を除いてハードウェア的な初期化処理が行われる。

【0079】

なお、プロセッサ10に接続されるローカルメモリとしてSDRAMを用いた場合、プロセッサ10は、上述の先頭アドレスASRや領域サイズAMRの情報を格納する領域指定レジスタ4aの他に、メモリアクセスに必要な情報として、DRAMの品種毎に異なるDRAMのオペレーション間の最小サイクル数を設定するコントロールレジスタ、DRAMに対するアクセスのモードを設定するアクセスモード制御レジスタ、DRAMのモード設定時にDRAMに対して書き込む情報を設定するモードセットレジスタ、DRAMのタイプ、バス幅、直結/DIMM使用などの情報を設定するコンフィギュレーションレジスタ、DRAMのRASアドレスビット数、CASアドレスビット数、バンクアドレスビット数を設定するアドレスナンバレジスタ、DRAMコントローラの状態を表すステータスレジスタ、DRAMのオートリフレッシュまたはセルフリフレッシュを設定するリフレッシュコントロールレジスタ、オートリフレッシュ時のリフレッシュ間隔を設定するリフレッシュタイマレジスタなどを備える。これらのレジスタは何れも、初期起動リセット信号PRSTによっては初期化されるが、再起動リセット信号HRSTやプログラムリセット信号SRSTによっては初期化されない。

30

40

【0080】

上記ハードウェア的な初期化処理の後、再起動リセット信号HRSTがネゲートされると、アドレス選択信号MS(=0)により選択されるアドレス"0x0000__0000"からの命令フェッチが開始される。このとき、領域指定レジスタ4a内の先頭アドレスASRおよび領域サイズAMRの値は、再起動リセット信号HRSTのアサートによっては初期化されていないので、このときの命令フェッチ要求はRAM6へと発行される。これにより、低速なROM9へのアクセスを不要とし、プロセッサ10の再起動処理を高速に行うことができる。

50

【 0 0 8 1 】

なお、ここでアドレス選択信号MSの値として"1"を設定することにより、ROM9に対して命令フェッチ要求を発行するように指示することも可能である。プロセッサ10の再起動時にROM9からの命令フェッチが実行される場合には、ROM9に格納されている初期化プログラムは、その最初のシーケンスにおいてリセットレジスタ4bの値を参照し、リセットレジスタ4b内のHDビットがセットされていることを確認することで、既に電源投入時に領域指定レジスタ4aに設定されている情報の再設定処理を省くことができ、その分だけプロセッサ10の再起動処理を高速化することができる。

【 0 0 8 2 】

なお、内部クロックclkの供給停止後に外部からの再起動リセット信号HRSTによりプロセッサ10の再起動を行う場合や、内部クロックbck, cckあるいは内部クロックclkの供給停止後に、内部発生した再起動リセット信号HRST'によりプロセッサ10の再起動を行う場合の再起動シーケンスについても、その動作は以上と同様である。

【 0 0 8 3 】

次に、プログラムリセット信号SRSTによる起動処理シーケンスについて説明する。

外部よりプログラムリセット信号SRSTがアサートされると、命令実行部2において現在実行中のプログラムが強制的に中断され、命令フェッチ制御部1に対してハードウェア的な初期化処理が行われる。

【 0 0 8 4 】

その後、プログラムリセット信号SRSTがネゲートされると、アドレス選択信号MS(=0)により選択されるアドレス"0x0000__0000"からの命令フェッチが開始される。このとき、領域指定レジスタ4a内の先頭アドレスASRおよび領域サイズAMRの値は、プログラムリセット信号SRSTのアサートによっては初期化されていないので、このときの命令フェッチ要求はRAM6へと発行される。これにより、低速なROM9へのアクセスを不要とし、プロセッサ10の再起動処理を高速に行うことができる。

【 0 0 8 5 】

なお、ここでアドレス選択信号MSの値として"1"を設定することにより、ROM9に対して命令フェッチ要求を発行するように指示することも可能である。プログラムリセット時にROM9からの命令フェッチが実行される場合には、ROM9に格納されている初期化プログラムは、その最初のシーケンスにおいてリセットレジスタ4bの値を参照し、リセットレジスタ4b内のSDビットがセットされていることを確認することで、命令フェッチ制御部1のレジスタに対してのみ必要な再設定処理を行い、その他の不要な処理を省くことができ、その分だけプロセッサ10の再起動処理を高速化することができる。

【 0 0 8 6 】

以上詳しく説明したように、本実施形態のプロセッサ10によれば、ソフトウェア的な初期化処理を行う際に、初期起動リセット信号PRST、再起動リセット信号HRST、プログラムリセット信号SRSTの各リセット信号とアドレス選択信号MSとを用いた指示に応じて、外部バス7, 8に接続されたROM9からの起動か、プロセッサ10にローカルメモリとして接続されたRAM6からの起動かを選択することができる。

【 0 0 8 7 】

これにより、電源投入後において内部クロックbck, cckの供給が停止された後の再起動時には、高速なRAM6から再起動用の初期化プログラムを読み出してプロセッサ10の起動を行うことができ、その再起動処理を高速に行うことができる。また、再起動時にROM9に対してアクセスする場合でも、リセットレジスタ4bの値を読むことで、ROM9に格納されている初期化プログラム内で分岐して、リセット内容に応じた起動処理のみを選択的に実行することができる。したがって、低消費電力化のためにプロセッサ10でクロック供給の停止と起動が頻繁に行われるようなシステムでも、システムの起動待ちという無駄な時間を格段に短くすることができる。

【 0 0 8 8 】

10

20

30

40

50

また、上記のようなプロセッサ 10 を共有バスに複数接続してマルチプロセッサシステムを構成した場合においても、例えばスレーブプロセッサで内部クロック bck , clk の供給が一時的に停止された後の再起動時において、当該スレーブプロセッサは高速な RAM 6 から再起動用の初期化プログラムを読み出すことができ、その再起動処理を高速に行うことができる。その際、スレーブプロセッサからは外部バス 7 , 8 に対して命令フェッチ要求を発行することがないので、複数のプロセッサから ROM 9 へのアクセスが競合することがなく、バス調停の処理も不要とすることができる。

【0089】

さらに、上述のようにマルチプロセッサシステムを構成した場合でも、それぞれのプロセッサにおいて実行すべき初期化プログラムを格納したメモリの先頭アドレスがアドレス選択信号 MS により選択される。したがって、どのプロセッサが ROM 9 内のどのプログラムを実行するのかを判別するための機構も不要となり、システムの構成を簡素化することができる。

10

【0090】

次に、本発明によるデータ処理システムの第 2 の実施形態について説明する。

図 8 は、第 2 の実施形態によるプロセッサをスレーブプロセッサとして搭載したマルチプロセッサシステムの構成例を示すブロック図である。

【0091】

図 8 において、71 はマスタープロセッサ、72 はスレーブプロセッサであり、それぞれに対してローカルメモリである RAM 73 , 74 が接続されている。共有バスである外部バス 7 , 8 には、これらのマスタープロセッサ 71、スレーブプロセッサ 72 の他に、ROM 9、共有メモリ 75、メモリ - メモリ間のダイレクトデータ転送を行う DMAC (Direct Memory Access Controller) 76、スレーブプロセッサ 72 を制御する信号を生成する ASIC (Application Specific Intergrated Ciucuit) 77 が接続されている。

20

【0092】

上記マスタープロセッサ 71 は、電源投入後は常時動作しているものであり、通常のプロセッサを用いることが可能である。一方、スレーブプロセッサ 72 は、電源投入後において、低消費電力化のために内部クロックの供給停止および起動を行うことができるようになされたものであり、本実施形態に特有のプロセッサが用いられる。

【0093】

図 9 は、本実施形態に係るスレーブプロセッサ 72 の構成例を示すブロック図である。

30

本実施形態のスレーブプロセッサ 72 は、図 1 に示したプロセッサ 10 とほぼ同様の構成を有するが、命令フェッチ制御部 81 およびバス制御部 84 の構成が図 1 と異なっている。また、命令フェッチ制御部 81 には、プログラムリセット信号 S R S T やアドレス選択信号 MS に加えて、起動抑止信号 W A I T が入力されている。

【0094】

上記命令フェッチ制御部 81 は、図 10 のように構成されている。図 10 に示すように、本実施形態の命令フェッチ制御部 81 では、図 3 に示した命令フェッチ制御部 1 と比べて、起動抑止信号 W A I T の論理を反転するインバータ 85、このインバータ 85 の出力と命令フェッチ制御ロジック 16 の出力との論理積をとる AND ゲート 86 とが追加されている。

40

【0095】

このように構成することにより、命令フェッチ制御ロジック 16 がプロセッサ 72 の命令実行状況やキャッシュ制御部の状況等を判断して命令フェッチ要求 $ival$ の発行が可能な状態となっても、さらに外部からの起動抑止信号 W A I T がネゲートされない限り、命令フェッチ要求 $ival$ はキャッシュユニット 3 に対して発行されることはない。このように、インバータ 85 および AND ゲート 86 によって本発明の起動抑止手段が構成される。

【0096】

また、上記バス制御部 84 は、図 4 に示したのとほぼ同様に構成されている。ただし、

50

本実施形態において外部バス制御部 2 3 は、規定されたプロトコルに従って外部バス 7 , 8 上に発生したトランザクションが他のデバイスから自プロセッサに対する要求であった場合、そのコマンドを取り込んで該当する処理を開始する機構を有している。

【 0 0 9 7 】

具体的には、外部バス制御部 2 3 は、外部バス 7 , 8 上に発生したトランザクションを常に監視し、有効なトランザクションが生じた場合に、そのアドレス情報を取り込んで自プロセッサ内の資源に対してあらかじめ割り当てられたアドレス情報との比較を行う。この比較の結果、自プロセッサ内の資源に対する要求と認識された場合は、そのトランザクションのコマンドを取り込み、上記アドレス情報によって指定される内部資源へのアクセスを開始する。

10

【 0 0 9 8 】

図 1 1 は、この外部バス制御部 2 3 の構成例を示す図である。図 1 1 に示すように、本実施形態の外部バス制御部 2 3 は、キャッシュユニット 3 からのコマンドとライトデータとを保持する 4 段のキャッシュコマンドバッファ 9 1、外部バス 7 , 8 からの応答データをキャッシュユニット 3 へ送り出すための応答データバッファ 9 2、外部バス 7 , 8 からのコマンドとライトデータとを保持する 4 段のバスコマンドバッファ 9 3、ローカルメモリである R A M 7 4 からの読み出しデータを受ける読出データバッファ 9 4 を備えている。

【 0 0 9 9 】

外部バス制御部 2 3 は更に、上記 4 段のキャッシュコマンドバッファ 9 1 に格納されたデータの何れかを選択するセクタ 9 5、上記 4 段のバスコマンドバッファ 9 3 に格納されたデータの何れかを選択するセクタ 9 6、アドレスバス 7 の調停を行うコマンド調停部 9 7、データバス 8 の調停を行うデータ調停部 9 8 およびメモリマップされたレジスタ 9 9 を備えている。

20

【 0 1 0 0 】

上記レジスタ 9 9 は、図 9 に示した領域指定レジスタ 4 a およびリセットレジスタ 4 b に相当するものであり、R A M 7 4 の先頭アドレス A S R および領域サイズ A M R の情報、図 6 に示したリセット種識別情報 P D , H D , S D および内部リセット書込情報 H S , S S を保持している。このレジスタ 9 9 には、キャッシュユニット 3 からのコマンド、外部バス 7 , 8 からのコマンドの両方がアクセスできるように構成されている。

30

【 0 1 0 1 】

このように、上述した第 1 の実施形態では先頭アドレス A S R および領域サイズ A M R を格納する領域指定レジスタ 4 a はバス I / F 2 1 内に備えられていたが (図 4 参照)、第 2 の実施形態では、これらの情報を格納するレジスタ 9 9 は、外部バス制御部 2 3 内に備えられている。この場合、バス I / F 2 1 は、外部バス制御部 2 3 からこのレジスタ 9 9 に設定された値を受け取っている。

【 0 1 0 2 】

次に、以上のように本実施形態によるプロセッサ 7 2 をスレーブプロセッサとして用いたマルチプロセッサシステムの起動シーケンスについて、以下に説明する。

まず、装置の電源投入時における初期起動リセット信号 P R S T による起動処理シーケンスについて説明する。

40

【 0 1 0 3 】

装置の電源投入時には、マスタープロセッサ 7 1 に対しては初期起動リセット信号 P R S T がアサートされ、スレーブプロセッサ 7 2 に対しては初期起動リセット信号 P R S T と起動抑止信号 W A I T とがアサートされる。初期起動リセット信号 P R S T がアサートされると、マスタープロセッサ 7 1 およびスレーブプロセッサ 7 2 の両方にてハードウェア的な初期化処理が行われる。このとき、先頭アドレス A S R や領域サイズ A M R の情報を含むプロセッサ 7 1 , 7 2 内のレジスタの値が、例えば全て " 0 " に初期化される。

【 0 1 0 4 】

そして、ハードウェア的な初期化処理が行われ、電源が安定すると、上記初期起動リセ

50

ット信号 P R S T がネゲートされる。マスタープロセッサ 7 1 では、初期起動リセット信号 P R S T がネゲートされると、続いてソフトウェア的な初期化処理を行うために、リセットベクタで指示されるアドレスに従って、外部バス 7 , 8 に接続された R O M 9 から初期化プログラムの読み出しが行われる。このとき、起動抑止信号 W A I T はアサートされたままである。そのため、スレーブプロセッサ 7 2 では、命令フェッチ要求 i a v a l は発行されず、初期化プログラムの読み出しは行われない。

【 0 1 0 5 】

マスタープロセッサ 7 1 における初期化処理が完了すると、マスタープロセッサ 7 1 は、スレーブプロセッサ 7 2 の内部資源（先頭アドレス A S R、領域サイズ A M R の情報など）を外部バス 7 , 8 を介して適切な値に設定する。つまり、マスタープロセッサ 7 1 から図 1 1 のデータ調停部 9 8 を介してレジスタ 9 9 に書き込みを行うことにより、スレーブプロセッサ 7 2 に接続された R A M 7 4 のアドレス領域を適切に設定する。そして、マスタープロセッサ 7 1 は、D M A C 7 6 を起動して、スレーブプロセッサ 7 2 が必要とする初期化プログラムを、例えば R O M 9 から R A M 7 4 に転送する。

10

【 0 1 0 6 】

このとき、スレーブプロセッサ 7 2 が備える図 1 1 の外部バス制御部 2 3 においては、メモリ - メモリ間のデータダイレクト転送のために D M A C 7 6 よりアドレスバス 7 を介して与えられたアドレスが、レジスタ 9 9 に設定された先頭アドレス A S R および領域サイズ A M R の対で指定される R A M 7 4 のアドレス領域内のものかどうかを判定する。

【 0 1 0 7 】

この場合、レジスタ 9 9 内の先頭アドレス A S R および領域サイズ A M R の各情報には、事前に R A M 7 4 のアドレス領域がセットされているので、外部バス制御部 2 3 は、R O M 9 からデータバス 8 を介して送られてきた初期化プログラムのデータをメモリ制御部 2 2 に送り、R A M 7 4 の該当するアドレス領域に対して初期化プログラムを格納することになる。

20

【 0 1 0 8 】

本実施形態では、先頭アドレス A S R および領域サイズ A M R の情報がバス I / F 2 1 ではなく外部バス制御部 2 3 内に保持されているので、バス I / F 2 1 を用いることなく、メモリ制御部 2 2 と外部バス制御部 2 3 だけで初期化プログラムの転送を行うことができる。また、D M A C 7 6 はマスタープロセッサ 7 1 と独立して転送処理を実行するため、マスタープロセッサ 7 1 の処理部を煩わせることなく、スレーブプロセッサ 7 2 の R A M 7 4 に必要なソフトウェアの転送処理を行うことができる。

30

【 0 1 0 9 】

D M A C 7 6 は、転送処理の完了を割り込み信号を用いてマスタープロセッサ 7 1 に通知する。D M A C 7 6 からの転送完了割り込み信号を検出したマスタープロセッサ 7 1 は、その後必要が生じた時点で A S I C 7 7 にアクセスし、起動抑止信号 W A I T をネゲートさせる。この起動抑止信号 W A I T がネゲートされることにより、スレーブプロセッサ 7 2 では、命令フェッチ制御部 8 1 が、" 0 " の値が設定されたアドレス選択信号 M S によって選択される初期アドレス " 0 x 0 0 0 0 _ 0 0 0 0 " からの命令フェッチ要求 i a v a l をキャッシュユニット 3 を介してバス制御部 8 4 に発行する。

40

【 0 1 1 0 】

このとき、バス制御部 8 4 内のバスコマンド制御部 2 4 では、当該命令フェッチ要求 i a v a l と共に与えられる要求アドレス a d d と、外部バス制御部 2 3 内のレジスタ 9 9 から受け取った先頭アドレス A S R および領域サイズ A M R の値とを比較し、要求アドレス a d d が R A M 7 4 のアドレス領域内のものかどうかを判定する。

【 0 1 1 1 】

ここでは、先頭アドレス A S R および領域サイズ A M R には R A M 7 4 のアドレス領域がセットされているので、バスコマンド制御部 2 4 は、命令フェッチ要求 i a v a l が R A M 7 4 に対する要求であると判断し、メモリ制御部 2 2 に対して命令フェッチ要求を発行する。この命令フェッチ要求を受けたメモリ制御部 2 2 は、メモリ I / F 2 8 を介して

50

接続された R A M 7 4 をアクセスして上記要求アドレス a d d に対応する命令データを読み出し、それを命令実行部 2 に供給して実行する。

【 0 1 1 2 】

その後、必要とされる処理がスレーブプロセッサ 7 2 において一定時間行われなかったとすると、そのことを契機として、スレーブプロセッサ 7 2 の命令実行部 2 がクロック制御部 5 に対してクロック停止信号 s t o p c を供給することにより、バス制御部 8 4 以外に供給されている内部クロック c c k の供給を停止させる。

【 0 1 1 3 】

マスタープロセッサ 7 1 は、その後必要性が生じた時点で、スレーブプロセッサ 7 2 が備えるレジスタ 9 9 の H S ビットに書き込みを行うことにより、内部クロック c c k の供給を再開させるとともに、ハードウェア的な初期化処理およびソフトウェア的な初期化処理を実行する。このとき、ソフトウェア的な初期化処理を行うために必要な初期化プログラムの起動シーケンスは、第 1 の実施形態で述べたのと同様であり、R A M 7 4 からの高速な再起動が実行される。

10

【 0 1 1 4 】

このように、第 2 の実施形態では、電源投入時における初期起動リセット信号 P R S T による起動処理シーケンスにおいて、スレーブプロセッサ 7 2 では、起動抑止信号 W A I T のアサートにより命令フェッチ要求 i a v a l の発行を停止している間に R A M 7 4 に初期化プログラムを転送しておき、その後起動抑止信号 W A I T をネゲートして R A M 7 4 に対して命令フェッチ要求 i a v a l を発行するようにしている。

20

【 0 1 1 5 】

したがって、電源投入時における起動処理シーケンスにおいても、スレーブプロセッサ 7 2 は共有バスを介して R O M 9 にアクセスしなくても済む。これにより、高速な R A M 7 4 から初期化プログラムを読み出してスレーブプロセッサ 7 2 の起動を行うことができる。また、共有バス上に接続された複数のプロセッサが R O M 9 に対するアクセスを一斉に開始することがなく、共有バスのアービトレーションを不要とすることもでき、起動処理を高速に行うことができる。

【 0 1 1 6 】

また、スレーブプロセッサ 7 2 において実行すべき初期化プログラムは R A M 7 4 に格納され、スレーブプロセッサ 7 2 の初期化処理を行う際には、アドレス選択信号 M S により選択したこの R A M 7 4 の先頭アドレスから初期化プログラムを読み出せば良いので、どのプロセッサがどのプログラムを実行するのかを判別するための機構も不要となり、システムの構成を簡素化することができる。

30

【 0 1 1 7 】

なお、以上に説明した各実施形態は、何れも本発明を実施するにあたっての具体化の一例を示したものに過ぎず、これらによって本発明の技術的範囲が限定的に解釈されてはならないものである。すなわち、本発明はその精神、またはその主要な特徴から逸脱することなく、様々な形で実施することができる。

【 0 1 1 8 】

例えば、以上の各実施形態では、外部から与えられるアドレス選択信号 M S に基づいて初期化プログラムの開始アドレスを選択していたが、開始アドレスの選択が内部で行われるようにしても良い。図 1 2 は、その場合の命令フェッチ制御部の構成例を示す図である。この例では、L S I 内部のレジスタに設定した値に従って、再起動アドレスの選択を行うようにしている。なお、図 1 2 において、図 1 0 に示したブロックと同じブロックには同一の符号を付している。

40

【 0 1 1 9 】

図 1 2 に示す命令フェッチ制御部 1 0 1 では、図 1 0 に示した命令フェッチ制御部 8 1 に対して、数個の論理ゲート 1 0 2 ~ 1 0 5 とレジスタ 1 0 6 とが追加されている。A N D ゲート 1 0 2 は、アドレス選択信号 M S と初期起動リセット信号 P R S T との論理積をとってその結果を O R ゲート 1 0 5 に出力する。A N D ゲート 1 0 3 は、信号 b t d i r と、

50

信号 `set_btmdir` と、初期起動リセット信号 `PRST` の論理反転信号との論理積をとってその結果を `OR` ゲート 105 に出力する。また、`AND` ゲート 104 は、信号 `set_btmdir` の論理反転信号と、初期起動リセット信号 `PRST` の論理反転信号と、レジスタ 106 の出力信号との論理積をとってその結果を `OR` ゲート 105 に出力する。

【0120】

上記 `OR` ゲート 105 は、各 `AND` ゲート 102 ~ 104 の出力結果の論理和をとってその結果をレジスタ 106 に出力する。これによりレジスタ 106 は、初期起動リセット信号 `PRST` のアサート時には、外部端子からアドレス選択信号 `MS` の値を取り込む。また、初期起動リセット信号 `PRST` がネゲートされ、信号 `set_btmdir` がアサートされている時は、信号 `btmdir` の値を取り込む。また、これ以外の状態の時は、記憶されている値を保持する。ここで、信号 `btmdir` , `set_btmdir` は、命令実行部 2 がレジスタ 106 への書き込み命令を実行することによりアサートされるものとする。

10

【0121】

セクタ 15 は、レジスタ 106 に保持されている値に応じて、"0x0000__0000"または"0xFF00__0000"の何れかのアドレスを選択する。例えば、レジスタ 106 に保持されている値が "1" のときはアドレス "0xFF00__0000" を選択し、レジスタ 106 に保持されている値が "0" のときはアドレス "0x0000__0000" を選択する。

【0122】

以上のように構成することにより、電源投入時には初期起動リセット信号 `PRST` がアサートされ、外部端子からアドレス選択信号 `MS` の値を取り込んで初期化プログラムの開始アドレスを選択する。また、クロック供給停止前に命令実行部 2 がレジスタ 106 への書き込み命令を実行することで当該レジスタ 106 の値を "0" に設定しておくことで、クロック供給の停止後における再起動リセット信号 `HRS` T による再起動時には、レジスタ 106 内の "0" の値を参照して、初期化プログラムの開始アドレスとして "0x0000__0000" (RAM の先頭アドレス) を選択することができる。

20

【0123】

なお、ここではアドレス選択信号 `MS` の外部端子とレジスタ設定との混合型を示しているが、レジスタ設定のみの構成とすることも可能である。また、レジスタ 106 を他のユニットに移動してメモリマップとすることで、スレーブプロセッサの動作時にマスタープロセッサからレジスタ 106 への書き込みを可能とすることができ、初期化プログラムの開始アドレスの選択をマスタープロセッサから制御可能とする構成も可能である。

30

【0124】

上述したように、アドレス選択信号に基づいて、ローカルメモリまたはバスを介して接続される外部メモリに格納された初期化プログラムの開始アドレスを選択するアドレス選択手段を設けたので、電源投入後において内部クロックの供給が停止された後の再起動時には、高速なローカルメモリから再起動用の初期化プログラムを読み出してプロセッサの起動を行うことができ、その再起動処理を高速に行うことができる。これにより、システムの消費電力を抑制するとともにクロック停止状態からの高速な復帰を実現することができるようになる。

【0125】

また、起動抑止信号に基づいて、初期化プログラムが格納されたメモリへのアクセス要求を抑止するアクセス抑止手段を備え、メモリへのアクセス要求を抑止している間に、外部バスを介して接続される外部メモリに格納された初期化プログラムをローカルメモリに転送しておき、アクセス要求の抑止の解除後には、ローカルメモリに格納された初期化プログラムの開始アドレスを選択するようにしたので、電源投入時における起動処理シーケンスにおいても外部メモリにアクセスすることなく、高速なローカルメモリから初期化プログラムを読み出して起動を行うことができる。また、外部バス上に接続された複数のプロセッサが外部メモリに対するアクセスを一斉に開始することがなくなり、共有バスのアービトレーションを不要とすることもできるので、起動処理を高速に行うことができる。

40

また、マルチプロセッサシステムを構成した場合に、どのプロセッサがどのプログラム

50

を実行するのかを判別するための機構も不要となり、システムの構成を簡素化することができる。

【0126】

本発明の様々な形態をまとめると、以下のようになる。

(1) リセット信号に応じて、ハードウェア的な初期化処理を行った後でメモリから必要な初期化プログラムを読み込んでソフトウェア的な初期化処理を行うデータ処理システムであって、

アドレス選択信号に基づいて、ローカルメモリまたはバスを介して接続される外部メモリに格納された上記初期化プログラムの開始アドレスを選択するアドレス選択手段を備えたことを特徴とするデータ処理システム。

10

【0127】

(2) 上記アドレス選択手段は、電源投入時に第1のリセット信号が外部より与えられるときは、上記アドレス選択信号に基づいて、上記外部メモリに格納された上記初期化プログラムの開始アドレスを選択することを特徴とする上記(1)に記載のデータ処理システム。

【0128】

(3) 上記アドレス選択手段は、クロック供給の停止後における再起動時に第2のリセット信号が外部より与えられるときは、上記アドレス選択信号に基づいて、上記ローカルメモリに格納された上記初期化プログラムの開始アドレスを選択することを特徴とする上記(1)に記載のデータ処理システム。

20

【0129】

(4) 上記第2のリセット信号が外部より与えられるときは、上記ハードウェア的な初期化処理では、上記ローカルメモリのアドレス領域を含むメモリアクセスに必要な情報以外を初期化することを特徴とする上記(3)に記載のデータ処理システム。

【0130】

(5) 上記アドレス選択手段は、強制的な初期化のために第3のリセット信号が外部より与えられるときは、上記アドレス選択信号に基づいて、上記ローカルメモリに格納された上記初期化プログラムの開始アドレスを選択することを特徴とする上記(1)に記載のデータ処理システム。

【0131】

(6) 上記第3のリセット信号が外部より与えられるときは、上記ハードウェア的な初期化処理では、プログラムの命令フェッチを制御するフェッチ制御部を含む最小限の情報だけを初期化することを特徴とする上記(5)に記載のデータ処理システム。

30

【0132】

(7) 上記ローカルメモリ内の初期化プログラムは、電源投入時に第1のリセット信号が外部より与えられたときに、上記外部メモリに格納された上記初期化プログラムを転送して格納することを特徴とする上記(1)に記載のデータ処理システム。

【0133】

(8) 上記第2のリセット信号と等価なりセット信号を装置内部で発生させるリセット内部発生手段を備えることを特徴とする上記(3)に記載のデータ処理システム。

40

【0134】

(9) 上記第3のリセット信号と等価なりセット信号を装置内部で発生させるリセット内部発生手段を備えることを特徴とする上記(5)に記載のデータ処理システム。

【0135】

(10) データ処理システムに対して行われたリセット動作が第1のリセット、第2のリセットおよび第3のリセットの何れであるかを識別するための情報を保持する情報保持手段を備え、

上記初期化プログラムの命令に基づいて上記情報保持手段内の識別情報を読み取り、その結果に応じた処理を行うことを特徴とする上記(1)に記載のデータ処理システム。

【0136】

50

(11) リセット信号に応じて、ハードウェア的な初期化処理を行った後でメモリから必要な初期化プログラムを読み込んでソフトウェア的な初期化処理を行うデータ処理システムであって、

起動抑止信号に基づいて、上記初期化プログラムが格納されたメモリへのアクセス要求を抑止するアクセス抑止手段を備えたことを特徴とするデータ処理システム。

【0137】

(12) アドレス選択信号に基づいて、ローカルメモリまたはバスを介して接続される外部メモリに格納された上記初期化プログラムの開始アドレスを選択するアドレス選択手段と、

上記アクセス抑止手段によって上記メモリへのアクセス要求を抑止している間に、上記外部メモリに格納された上記初期化プログラムを上記ローカルメモリに転送するデータ転送手段とを備え、

上記アクセス要求の抑止の解除後に上記アドレス選択手段は、上記アドレス選択信号に基づいて、上記ローカルメモリに格納された上記初期化プログラムの開始アドレスを選択することを特徴とする上記(11)に記載のデータ処理システム。

【0138】

(13) 上記ローカルメモリのアドレス領域を表す情報を保持するレジスタおよび、外部より与えられるリセット信号と等価なリセット信号を装置内部で発生させるための情報を保持するレジスタを備え、上記外部バスに接続された他のバスマスタデバイスからこれらのレジスタに対してアクセス可能であることを特徴とする上記(12)に記載のデータ処理システム。

【0139】

(14) 共有バス上に複数のデータ処理装置が接続され、それぞれのデータ処理装置において、リセット信号に応じて、ハードウェア的な初期化処理を行った後でメモリから必要な初期化プログラムを読み込んでソフトウェア的な初期化処理を行うデータ処理システムであって、当該データ処理システムに含まれる少なくとも1つのデータ処理装置に、

アドレス選択信号に基づいて、ローカルメモリまたはバスを介して接続される外部メモリに格納された上記初期化プログラムの開始アドレスを選択するアドレス選択手段と、

起動抑止信号に基づいて、上記初期化プログラムが格納されたメモリへのアクセス要求を抑止するアクセス抑止手段と、

上記アクセス抑止手段によって上記メモリへのアクセス要求を抑止している間に、上記外部メモリに格納された上記初期化プログラムを上記ローカルメモリに転送するデータ転送手段とを備え、

上記アクセス要求の抑止の解除後に上記アドレス選択手段は、上記アドレス選択信号に基づいて、上記ローカルメモリに格納された上記初期化プログラムの開始アドレスを選択することを特徴とするデータ処理システム。

【0140】

(15) 上記データ転送手段によるデータ転送は、メモリ-メモリ間のダイレクト転送を行うDMACを用いて行うことを特徴とする上記(14)に記載のデータ処理システム。

【図面の簡単な説明】

【0141】

【図1】本発明によるデータ処理システムの第1の実施形態に係るプロセッサの構成例を示すブロック図である。

【図2】RAM、ROMおよびレジスタを含むシステム全体のメモリマップを示す図である。

【図3】命令フェッチ制御部の詳細な構成例を示す図である。

【図4】バス制御部の領域指定レジスタを含む部分の詳細な構成例を示す図であり、アドレスおよびデータの流れを示す図である。

【図5】バスコマンド制御部内でアドレス判定を行う部分の詳細な構成例を示す図である。

【図6】バス制御部のリセットレジスタを含む部分の詳細な構成例を示す図であり、当該リセットレジスタに対するデータ書き込みの流れを示す図である。

【図7】クロック制御部の構成例を示す図である。

【図8】第2の実施形態によるプロセッサをスレーブプロセッサとして搭載したマルチプロセッサシステムの構成例を示すブロック図である。

【図9】第2の実施形態によるプロセッサの構成例を示すブロック図である。

【図10】第2の実施形態による命令フェッチ制御部の構成例を示す図である。

【図11】第2の実施形態による外部バス制御部の構成例を示す図である。

【図12】命令フェッチ制御部の他の構成例を示す図である。

【符号の説明】

10

【0142】

1 命令フェッチ制御部

2 命令実行部

3 キャッシュユニット

4 バス制御部

4 a 領域指定レジスタ

4 b リセットレジスタ

5 クロック制御部

6 R A M (ローカルメモリ)

7 アドレスバス

20

8 データバス

9 R O M (外部メモリ)

10 プロセッサ

14, 15 セレクタ (アドレス選択手段)

23 外部バス制御部

24 バスコマンド制御部

71 マスタープロセッサ

72 スレーブプロセッサ

73, 74 R A M (ローカルメモリ)

76 D M A C

30

77 A S I C

81 命令フェッチ制御部

84 バス制御部

85 インバータ

86 A N Dゲート

99 レジスタ

101 命令フェッチ制御部

106 レジスタ

P R S T 初期起動リセット信号 (第1のリセット信号)

H R S T 再起動リセット信号 (第2のリセット信号)

40

S R S T プログラムリセット信号 (第3のリセット信号)

M S アドレス選択信号

A S R 先頭アドレス情報

A M R 領域サイズ情報

P D, H D, S D リセット種識別ビット

H S, S S 内部リセット書込みビット

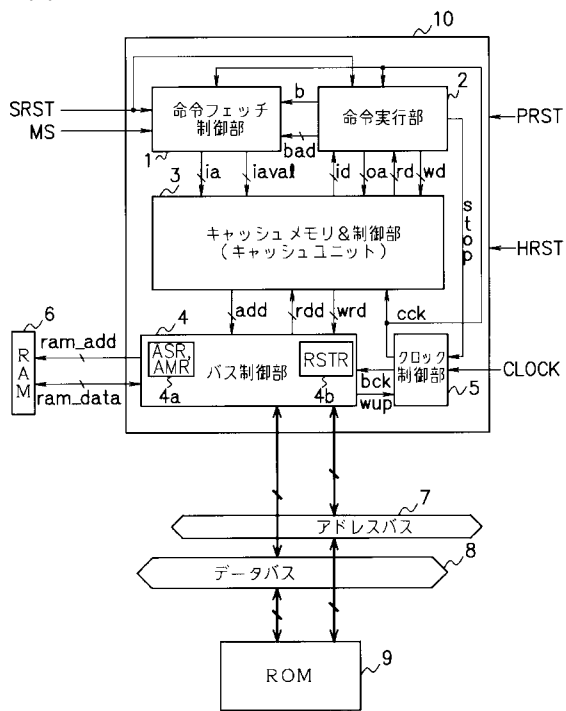
b c k, c c k 内部クロック

s t o p c, s t o p b c クロック停止信号

W A I T 起動抑止信号

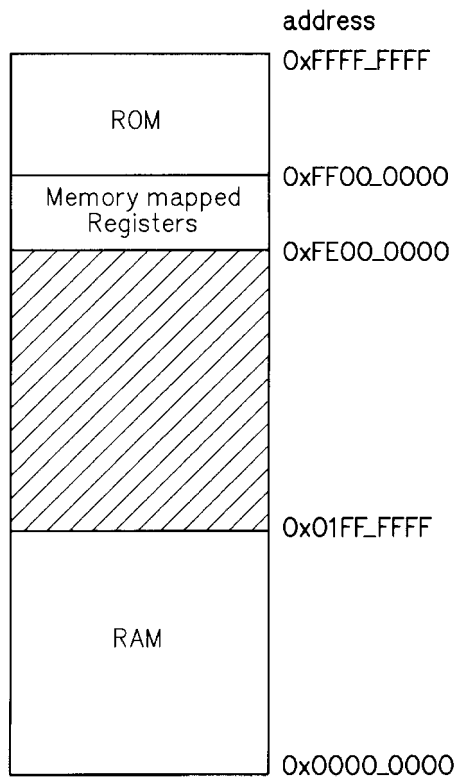
50

【 図 1 】



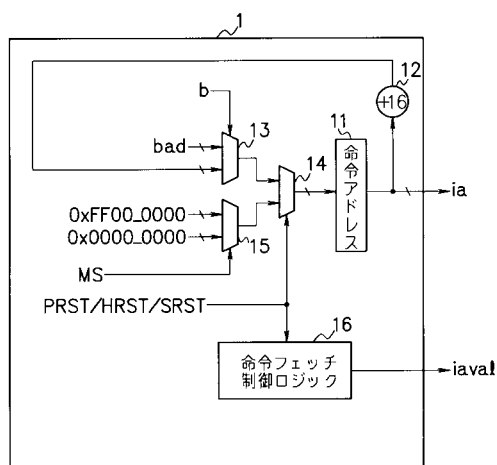
本実施形態のプロセッサの構成例

【 図 2 】



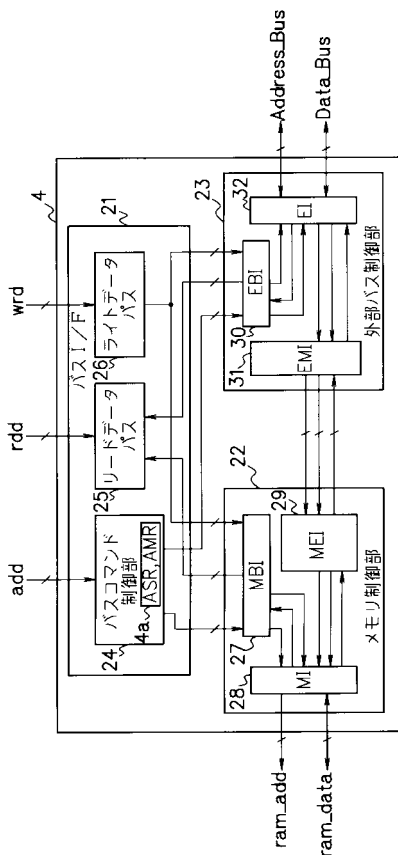
メモリマップの例

【 図 3 】



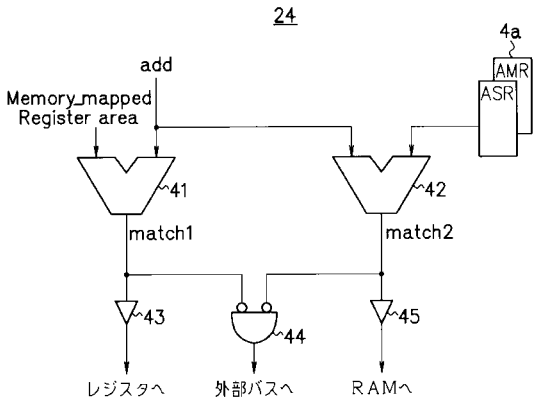
命令フェッチ制御部の構成例

【 図 4 】



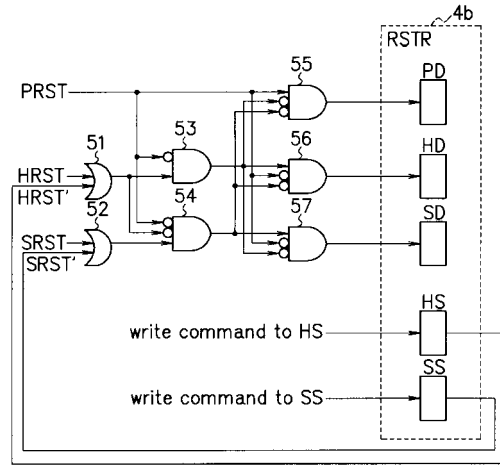
バス制御部の一部構成例

【 図 5 】



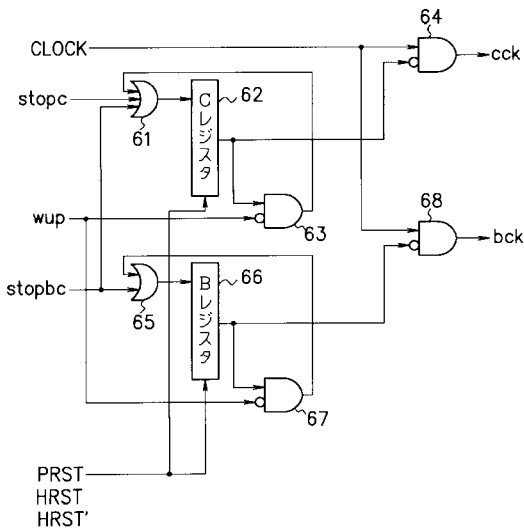
バスコマンド制御部の構成例

【 図 6 】



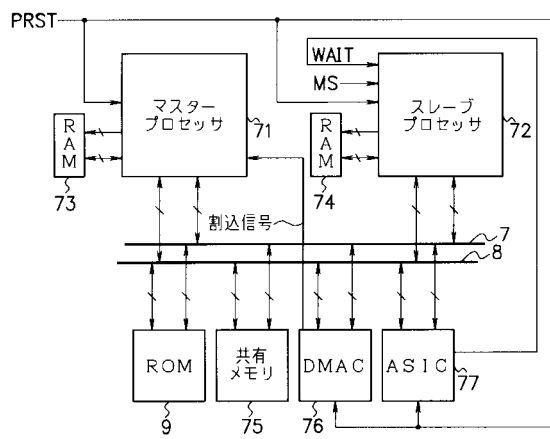
バス制御部の一部構成例

【 図 7 】



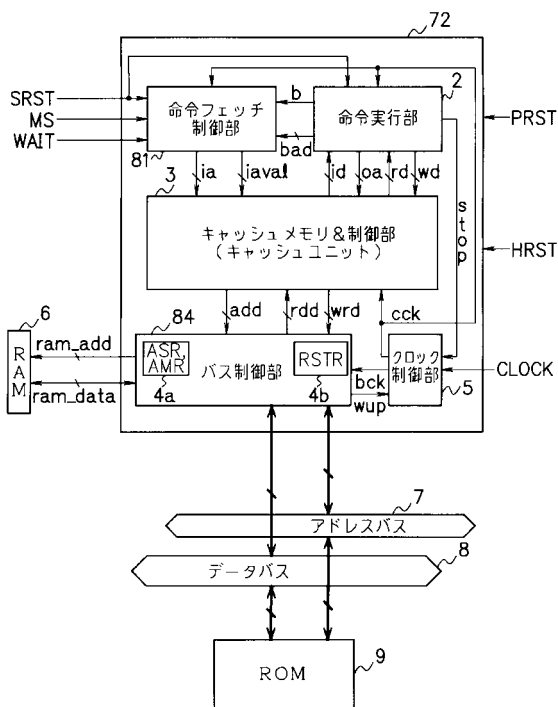
クロック制御部の構成例

【 図 8 】



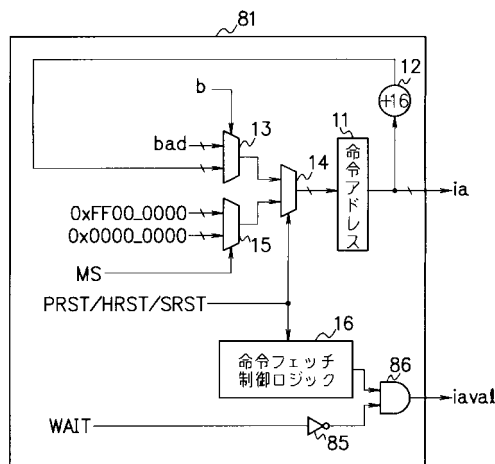
マルチプロセッサシステムの構成例

【 図 9 】



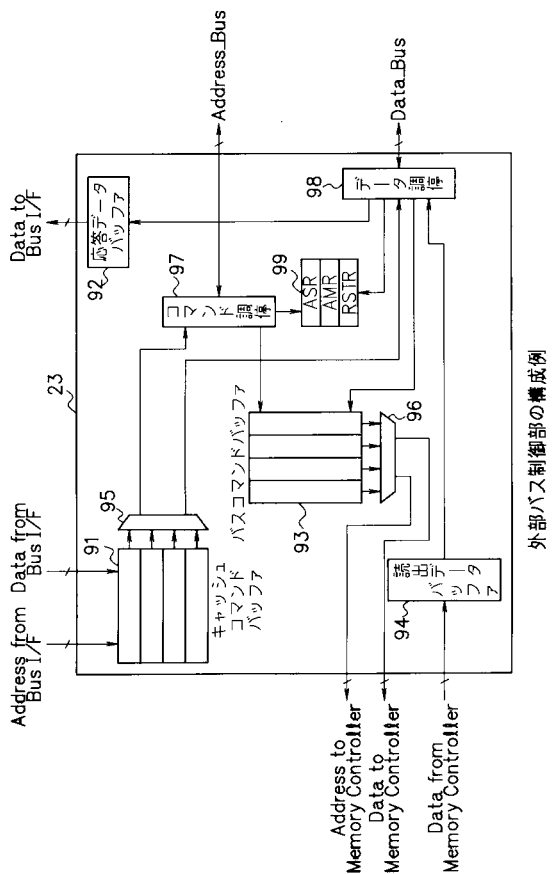
本実施形態のプロセッサの構成例

【 図 10 】



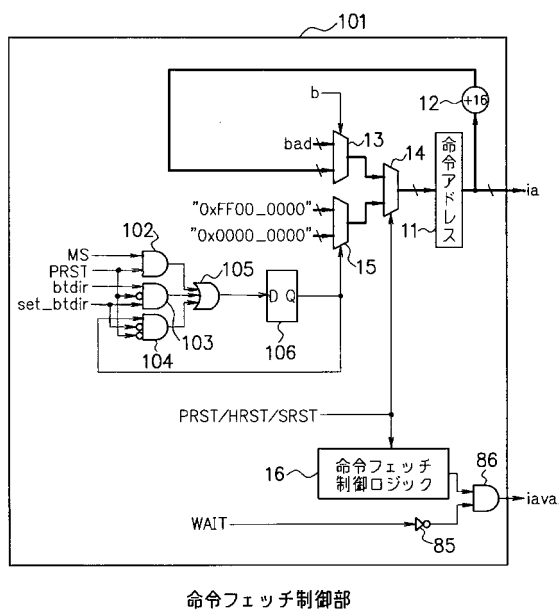
命令フェッチ制御部

【 図 11 】



外部バス制御部の構成例

【 図 12 】



命令フェッチ制御部