

(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2004-199187

(P2004-199187A)

(43) 公開日 平成16年7月15日(2004.7.15)

(51) Int.Cl.⁷

G06F 13/362

F I

G06F 13/362 510D

テーマコード (参考)

5B061

審査請求 有 請求項の数 11 O L (全 15 頁)

(21) 出願番号 特願2002-364231 (P2002-364231)

(22) 出願日 平成14年12月16日 (2002.12.16)

(71) 出願人 000005821

松下電器産業株式会社

大阪府門真市大字門真1006番地

(74) 代理人 100105647

弁理士 小栗 昌平

(74) 代理人 100105474

弁理士 本多 弘徳

(74) 代理人 100108589

弁理士 市川 利光

(74) 代理人 100115107

弁理士 高松 猛

(74) 代理人 100090343

弁理士 濱田 百合子

最終頁に続く

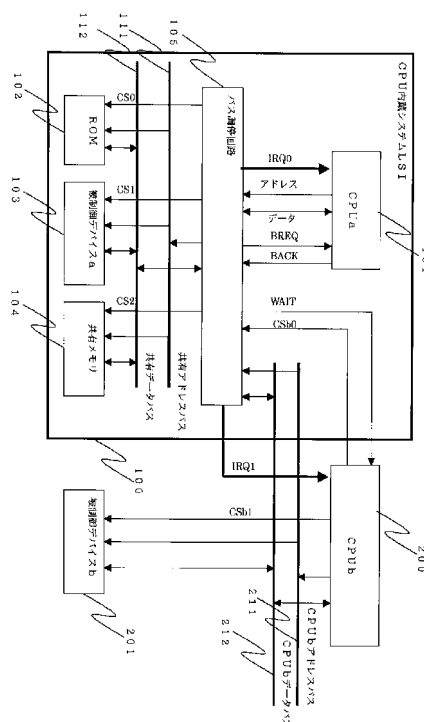
(54) 【発明の名称】 CPU内蔵LSI

(57) 【要約】

【課題】 CPU内蔵LSIと外部CPUがバスを共有するシステムにおいて、外部CPUがバスに接続された被制御デバイスにアクセスする際に、CPU内蔵LSIにおいて共有バス上に実装されたデバイスのアクセスを妨げないようにする。

【解決手段】 CPUaと、CPUaに接続された共有アドレス／データバス111、112と、CPUbに接続されたCPUbアドレス／データバス211、212と、共有アドレス／データバスとCPUbアドレス／データバスとの間に配置され、共有アドレス／データバスに接続されたデバイスに対するCPUbおよびCPUaからのアクセスを排他的に制御し、CPUbが共有アドレス／データバスに接続されたデバイスに対するアクセスを許可されたときのみCPUbアドレス／データバスを共有アドレス／データバスに結合するバス調停回路105とを備える。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

内蔵 CPU と、前記内蔵 CPU に接続された第 1 のバスと、外部 CPU に接続された第 2 のバスと、前記第 1 のバスと前記第 2 のバスとの間に配置され、前記第 1 のバスに接続されたデバイスに対する前記外部 CPU および前記内蔵 CPU からのアクセスを排他的に制御し、前記外部 CPU が前記第 1 のバスに接続されたデバイスに対するアクセスを許可されたときにのみ前記第 2 のバスを前記第 1 のバスに結合するバス調停回路とを備えたことを特徴とする CPU 内蔵 LSI。

【請求項 2】

前記バス調整回路は、前記内蔵 CPU による前記第 1 のバスに接続されたデバイスへのアクセス中に、前記外部 CPU から前記第 1 のバスに接続されたデバイスに対するアクセス要求が発生すると、前記内蔵 CPU へバス解放要求信号を伝えるとともに前記外部 CPU へウェイト信号を伝え、前記内蔵 CPU からバス解放完了信号を受け取ると前記ウェイト信号を解除し、前記第 1 のバスに接続されたデバイスに対する前記外部 CPU からのアクセスを許可することを特徴とする請求項 1 記載の CPU 内蔵 LSI。 10

【請求項 3】

前記バス調停回路は、前記内蔵 CPU の動作停止が設定されると、前記バス解放要求信号を前記内蔵 CPU に伝えることなく、前記第 1 のバスに接続されたデバイスに対する前記外部 CPU からのアクセスを許可することを特徴とする請求項 2 記載の CPU 内蔵 LSI。 20

【請求項 4】

前記第 1 のバスに接続された共有メモリを備えることを特徴とする請求項 1 から 3 のいずれか一項記載の CPU 内蔵 LSI。

【請求項 5】

前記第 1 のバスに接続され、前記 CPU 内蔵 LSI を動作させるプログラムを格納するメモリデバイスを備えることを特徴とする請求項 1 から 4 のいずれか一項記載の CPU 内蔵 LSI。

【請求項 6】

前記バス調停回路は、前記内蔵 CPU と前記外部 CPU との間における割り込み通知を行うための割り込み制御回路を備えることを特徴とする請求項 1 から 5 のいずれか一項記載の CPU 内蔵 LSI。 30

【請求項 7】

前記割り込み制御回路は、割り込み要因のビット割り当て設定がプログラム可能な複数ビットを有する割り込み要因レジスタと、前記割り込み要因レジスタのビット論理和を割り込み信号として出力する回路とを備えることを特徴とする請求項 6 記載の CPU 内蔵 LSI。

【請求項 8】

前記第 1 のバスに接続されたメモリデバイスは RAM であり、前記 CPU 内蔵 LSI の起動時に、前記外部 CPU が前記第 2 のバスに接続された外部メモリから前記 RAM へ前記内蔵 CPU を動作させるプログラムをロードすることを特徴とする請求項 5 記載の CPU 内蔵 LSI。 40

【請求項 9】

前記バス調停回路は、書き込みアドレスレジスタと書き込みデータレジスタとを備え、前記外部 CPU は、前記内蔵 CPU を動作させるプログラムを前記 RAM へロードする際に、前記 RAM のアドレスを前記書き込みアドレスレジスタに設定し、前記 RAM に書き込むデータを前記書き込みデータレジスタに書き込むことを特徴とする請求項 8 記載の CPU 内蔵 LSI。

【請求項 10】

前記書き込みアドレスレジスタは、前記書き込みデータレジスタへの書き込みが発生する度にインクリメントされることを特徴とする請求項 9 記載の CPU 内蔵 LSI。 50

【請求項 11】

第1の内蔵CPUと、第2の内蔵CPUと、前記第1の内蔵CPUに接続された第1のバスと、前記第2の内蔵CPUに接続された第2のバスと、前記第1のバスと前記第2のバスとの間に配置され、前記バス調停回路は、前記第1のバスに接続されたデバイスに対する前記第2の内蔵CPUおよび前記第1の内蔵CPUからのアクセスを排他的に制御し、前記第2の内蔵CPUが前記第1のバスに接続されたデバイスに対するアクセスを許可されたときにのみ前記第2のバスを前記第1のバスに結合するバス調停回路とを備えたことを特徴とするCPU内蔵LSI。

【発明の詳細な説明】

【0001】

10

【発明の属する技術分野】

本発明は、内蔵CPUと外部CPUとが共有バスをアクセスするシステム構成におけるバス制御機能を備えたCPU内蔵LSIに関するものである。

【0002】

【従来の技術】

近年、システムLSIが複雑化／大規模化する中で、システムLSIの制御を行うソフトウェアにおいても複雑化／大規模化が進んでいる。これに伴い、システムLSIが実現する機能をセット商品に実装する際に、ソフトウェア実装工数の増加が最大の課題となってきた。

【0003】

20

そこで、システムLSIにCPUを内蔵し、システムLSIの制御ソフトウェアを内蔵CPUに実行させることにより、複雑なソフトウェア処理をシステムLSI内に隠蔽する方法がトレンドになりつつある。この方法は、USB、IEEE1394等の後付け機能となることが多いインターフェース系を有するシステムLSIに多用されている。

【0004】

前述のCPUを内蔵したシステムLSIをセット商品のシステムに実装する場合、セット商品システムに既存するCPUと内蔵CPUとの間の通信が必要となる。通信方法は、シリアルバス接続による方法と、パラレルバスによるバス共有の方法に大別されるが、大量のデータを通信する場合は、通信速度の面からパラレルバスによるバス共有を採用しメモリ共有を行うことが多い。

30

【0005】

ここで図6を用いて、2つのCPUが外部拡張バスを共有してメモリ共有を行うことによりデータ通信を実現する従来のシステム構成を説明する。図6に示すように、システムはCPU内蔵システムLSI100、CPUb200、CPUbが制御する被制御デバイスb201から構成される。さらに、CPU内蔵システムLSI100は、CPUa101、CPUaの処理プログラムを格納するROM102、CPUaが制御する被制御デバイスa103、CPUaとCPUbがデータ通信を行うための共有メモリ104から構成される。

【0006】

CPUb200は、外部拡張されたアドレスバス／データバスを介して、被制御デバイスb201とCPU内蔵システムLSI100の2つのデバイスを制御する構成となり、CPU内蔵システムLSI100を制御する際は共有メモリ104を介して制御を行う。

40

【0007】

CPUa101は外部拡張されたアドレスバス／データバスをCPUb200と共有しており、この共有バスを介してROM102から処理プログラムを読み出し、これを実行しながら被制御デバイスa103の制御を行うとともに、共有メモリ104を介してCPUb200からの制御を受ける。

【0008】

また、CPUa101とCPUb200はそれぞれバス調停機能を備えており、CPUb200が共有バスを使用して被制御デバイスb201や共有メモリ104にアクセスする

50

際は、以下の手順でバスの使用权を獲得してアクセスを行う。

【0009】

(1) CPU b 200がCPU a 101に対してバス開放要求信号BREQをアサートする。

(2) CPU a 101は共有バスへのアクセスを停止して、CPU b 200に対してバス開放完了信号BACKをアサートする。

(3) CPU b 200は被制御デバイスb 201あるいは共有メモリ104へのアクセスを完了すると、CPU a 101に対するBREQ信号をネゲートする。

(4) CPU a 101はBREQ信号のネゲートを確認してからBACK信号をネゲートする。

10

【0010】

同様に、CPU aが共有バスを使用してROM102、被制御デバイスa 103および共有メモリ104にアクセスする場合に関しても、CPU b 200にバス開放要求信号BREQを出してバスの使用权を獲得し、上記と同様の処理を行う。

【0011】

このような複数のCPUがバスを共有するシステムにおける、共有バスの使用权を獲得するためのバス調停に関しては優先度制御の課題があり、従来各種方式が提案されている。例えば、特許文献1に記載のものは、複数のCPUのバス使用权の要求／許可信号をリング状に接続し、固定的な優先度制御を排除するものである。

【0012】

20

【特許文献1】

特開平5-282246号公報

【0013】

【発明が解決しようとする課題】

しかしながら、上記従来のシステム構成では、通常、CPU a 101はROM102から処理プログラムを読み出し実行しているが、CPU b 200が被制御デバイスb 201や共有メモリ104にアクセスする度に、CPU b 200が共有バスを使用するためにCPU a 101がROM102にアクセスできなくなり、CPU a 101は処理を停止する必要がある。その結果、CPU a 101の処理効率が著しく低下するという問題が生じていた。

30

【0014】

また、CPU b 200がCPU内蔵システムLSI100を制御するためには必ず上記BREQ信号とBACK信号を用いたバス調停機能を備えている必要があり、CPU b 200の品種の選択範囲が制限される点も問題であった。

【0015】

さらに、CPU内蔵システムLSI100が後付けのインターフェース系を有するシステムLSIである場合など、CPU b 200がシステム全体の制御を行うマイコンである場合では、全体の制御を司るCPU b 200の動作をCPU a 101に対して優先させたいことが多い。

【0016】

40

しかしながら、従来の構成においては、CPU a 101とCPU b 200がバスに対して対等の権限をもっているため、CPU b 200の動作中にCPU a 101が共有バスにアクセスすると割り込みが生じ、CPU b 200がシステム全体の制御を行えなくなり、セットとしても処理効率が低下するという問題もあった。

【0017】

本願発明は上記事情に鑑みてなされたものであり、CPU内蔵LSIが他のCPUとバスを共有するシステムにおいて、それぞれのCPUが常時使用するデバイスがバスに接続されている場合に、それぞれのCPUがそれぞれ常時使用するデバイスを同時にアクセスしあるいは制御することを可能にする機構を備えたCPU内蔵LSIを提供し、システムの処理効率を上げることを第1の課題とする。

50

【0018】

また、CPU内蔵LSIがバス調停回路を備えていない他のCPUとバスを共有するシステムにおいても、共有バスを介して他のCPUを制御することが可能なCPU内蔵システムLSIを提供することを第2の課題とする。さらに、特定のCPUを優先動作させることを第3の課題とする。

【0019】

【課題を解決するための手段】

この課題を解決するために、請求項1に係る発明は、内蔵CPUと、前記内蔵CPUと接続された第1のバスと、外部CPUと接続された第2のバスと、前記第1のバスと前記第2のバスとの間に配置され、前記バス調停回路は、前記第1のバスに接続されたデバイスに対する前記外部CPUおよび前記内蔵CPUからのアクセスを排他的に制御し、前記外部CPUが前記第1のバスに接続されたデバイスに対するアクセスを許可されたときにのみ前記第2のバスを前記第1のバスに結合するバス調停回路とを備える。 10

【0020】

上記構成によれば、内蔵CPUと接続された第1のバスと、外部CPUと接続された第2のバスが両者の間に配置されたバス調停回路により分離され、バス調停回路の排他的アクセス制御により、第1のバスに接続されたデバイスを内蔵CPUと外部CPUとが共有するとともに、外部CPUが第2のバスに接続されたデバイスをアクセス中であっても、内蔵CPUは第1のバスに接続されたデバイスをアクセスすることができるため、システムの処理効率を低下させることがない。 20

【0021】

請求項2に係る発明は、請求項1記載のCPU内蔵LSIにおいて、前記バス調停回路は、内蔵CPUによる第1のバスに接続されたデバイスへのアクセス中に、外部CPUから第1のバスに接続されたデバイスに対するアクセス要求が発生すると、内蔵CPUへバス解放要求信号を伝えとともに外部CPUへウェイト信号を伝え、内蔵CPUからバス解放完了信号を受け取ると前記ウェイト信号を解除し、第1のバスに接続されたデバイスに対する外部CPUからのアクセスを許可する。

【0022】

上記構成によれば、バス調停回路にバス開放要求信号とバス開放完了信号を用いた調停制御だけでなくウェイト信号を用いた調停機能を備えることにより、外部CPUがバス調停機能を備えていなくてもバス調停を実現できるとともに、外部CPUが内蔵CPUに優先して動作することが可能になるため、特に外部CPUがシステム全体の制御を司る場合にはシステム全体のパフォーマンス低下に与える影響を最小限に抑えることができる。 30

【0023】

請求項3に係る発明は、請求項2記載のCPU内蔵LSIにおいて、前記バス調停回路は、内蔵CPUの動作停止が設定されると、バス解放要求信号を内蔵CPUに伝えることなく、第1のバスに接続されたデバイスに対する外部CPUからのアクセスを許可する。

【0024】

上記構成によれば、内蔵CPUの動作停止が設定されたときには、バス調停回路は内蔵CPUからのバス解放完了信号を待つステップをスキップして外部CPUからのアクセスを許可することができる。これにより、例えばCPUを内蔵しないシステムLSIに対して作成された処理プログラムにおいても、外部CPUを用いてLSIを動作させることが可能となるので、処理プログラムの互換性を高めることができる。 40

【0025】

請求項4に係る発明は、請求項1から3のいずれか一項記載のCPU内蔵LSIにおいて、第1のバスに接続された共有メモリを備える。

【0026】

上記構成によれば、バス調停回路の排他的アクセス制御により外部CPUからはバス調停回路の存在を意識することなく共有メモリへのアクセスを行いCPU内蔵システムLSIを制御することができ、システム全体のパフォーマンス低下に与える影響を最小限に抑え 50

ながら、内蔵CPUと外部CPUとが共有メモリを使用してデータ通信を行うことができる。

【0027】

請求項5に係る発明は、請求項1から4のいずれか一項記載のCPU内蔵LSIにおいて、第1のバスに接続され、CPU内蔵LSIを動作させるプログラムを格納するメモリデバイスを備える。

【0028】

上記構成によれば、バス調停回路の排他的アクセス制御により、外部CPUが第2のバスに接続されたデバイスをアクセス中であっても、内蔵CPUは第1のバスに接続されたメモリデバイスからプログラムを読み出し実行することができるため、内蔵CPUの処理効率10
率が低下するという問題が生じない。

【0029】

請求項6に係る発明は、請求項1から5のいずれか一項記載のCPU内蔵LSIにおいて、前記バス調停回路は、前記内蔵CPUと前記外部CPUとの間における割り込み通知を行うための割り込み制御回路を備える。

【0030】

上記構成によれば、バス調停回路に割り込み制御回路を備えることにより、内蔵CPUと外部CPUとが共有する第1のバスの使用に関する割り込み通知を行うことができ、例えば、共有メモリへの書き込み完了等の通知に使用することができる。

【0031】

請求項7に係る発明は、請求項6記載のCPU内蔵LSIにおいて、前記割り込み制御回路は、割り込み要因のビット割り当て設定がプログラム可能な複数ビットを有する割り込み要因レジスタと、この割り込み要因レジスタのビット論理和を割り込み信号として出力する回路とを備える。20

【0032】

上記構成によれば、割り込みの要因のビット割り当て設定がプログラム可能な割り込み要因レジスタを備えることにより、そのビットをプログラマが自由に割り込み要因に割り付けることができ、処理プログラムの自由度を高めることができる。

【0033】

請求項8に係る発明は、請求項5記載のCPU内蔵LSIにおいて、前記第1のバスに接続されたメモリデバイスはRAMであり、CPU内蔵LSIの起動時に、外部CPUが第2のバスに接続された外部メモリから前記RAMへ内蔵CPUを動作させるプログラムをロードするものである。30

【0034】

上記構成によれば、CPU内蔵LSIを動作させるプログラムを格納するメモリデバイスをRAMにすることにより、プログラムをROM化するリスクを避けることができ、かつ内蔵CPUの処理速度を向上させることができる。また、フラッシュROMを外部CPUと内蔵CPUとで共有することにより、システム全体のコストを削減することができる。

【0035】

請求項9に係る発明は、請求項8記載のCPU内蔵LSIにおいて、前記バス調停回路は書き込みアドレスレジスタと書き込みデータレジスタとを備え、前記外部CPUは、内蔵CPUを動作させるプログラムをRAMへロードする際に、RAMのアドレスを前記書き込みアドレスレジスタに設定し、RAMに書き込むデータを前記書き込みデータレジスタに書き込む。40

【0036】

上記構成によれば、外部CPUがRAMへプログラムをロードする際に、アドレス指定レジスタおよびデータレジスタを介してRAMへの書き込みを行うことにより、外部CPUからアクセス指定するアドレスとしてレジスタのアドレスを指定すれば良いだけになるため、外部CPUからアクセスするメモリ空間を削減することができ、CPU内蔵LSIと外部CPU間のアドレスバスの端子数を削減することができる。50

【0037】

請求項10に係る発明は、請求項9記載のCPU内蔵LSIにおいて、前記書き込みアドレスレジスタは、前記書き込みデータレジスタへの書き込みが発生する度にインクリメントされる。

【0038】

上記構成によれば、書き込みアドレスレジスタが書き込みデータレジスタへの書き込みが発生する度にインクリメントされるため、プログラムをRAMの連続したアドレスに書き込む場合はアドレス指定レジスタを毎回書き換える必要がなくなる。

【0039】

請求項11に係る発明は、第1の内蔵CPUと、第2の内蔵CPUと、前記第1の内蔵CPUに接続された第1のバスと、前記第2の内蔵CPUに接続された第2のバスと、前記第1のバスと前記第2のバスとの間に配置され、前記バス調停回路は、前記第1のバスに接続されたデバイスに対する前記第2の内蔵CPUおよび前記第1の内蔵CPUからのアクセスを排他的に制御し、前記第2の内蔵CPUが前記第1のバスに接続されたデバイスに対するアクセスを許可されたときにのみ前記第2のバスを前記第1のバスに結合するバス調停回路とを備える。

【0040】

上記構成によれば、第1の内蔵CPUに接続された第1のバスと、第2の内蔵CPUに接続された第2のバスが両者の間に配置されたバス調停回路により分離され、バス調停回路の排他的アクセス制御により、第1のバスに接続されたデバイスを第1の内蔵CPUと第2の内蔵CPUとが共有するとともに、第2の内蔵CPUが第2のバスに接続されたデバイスをアクセス中であっても、第1の内蔵CPUは第1のバスに接続されたデバイスをアクセスすることができる。これにより、システムの処理効率を低下させることのないCPU内蔵LSIが得られる。

【0041】

【発明の実施の形態】

以下、図面を参照して本発明の実施の形態を詳細に説明するが、これによって本発明の技術的範囲が制限されるものではない。

【0042】

図1は本発明の第1の実施形態に係るCPU内蔵LSIを用いたシステム構成を示すブロック図である。図1において、システムはCPU内蔵システムLSI100、CPUb200、CPUbが制御する被制御デバイスb201から構成される。さらに、CPU内蔵システムLSI100は、CPUa101、CPUaの処理プログラムを格納するROM102、CPUaが制御する被制御デバイスa103、CPUaとCPUbがデータ通信を行うための共有メモリ104、バス調停回路105から構成される。

【0043】

CPUb200は、CPUbアドレスバス211およびCPUbデータバス212を介して、CPU内蔵システムLSI100と被制御デバイスb201の2個のデバイスを制御する。またCPUb200は、CPU内蔵システムLSI100と被制御デバイスb201を、2本のチップセレクト信号CSb0、CSb1を用いてそれぞれアクセスする。

【0044】

CPUa101は、バス調停回路105と共有アドレスバス111および共有データバス112を介して、ROM102、被制御デバイスa103、共有メモリ104にアクセスする。ROM102はCPUa101の処理プログラムを格納しており、CPUa101はROM102に格納された処理プログラムを読み出しながら処理を実行する。共有メモリ104は、CPUa101とCPUb200の両CPUからのアクセスを受け、CPUb200は共有メモリ104を介してCPU内蔵システムLSI100を制御する。

【0045】

バス調停回路105は、CPUa101とCPUb200の両CPUからのアクセスを調停し、共有アドレスバス111および共有データバス112を介して、ROM102、被

制御デバイス a 1 0 3、共有メモリ 1 0 4 へのアクセスを行う。

【0046】

このとき、バス調停回路 1 0 5 は、CPU a 1 0 0、CPU b 2 0 0 からのアクセスにおいて指定されたアドレスにより、特定のアドレスにマッピングされた ROM 1 0 2、被制御デバイス a 1 0 3、共有メモリ 1 0 4 に対して、それぞれにチップセレクト信号 CS 0、CS 1、CS 2 を出力してアクセスを行う。

【0047】

図 2 は CPU b 2 0 0 から共有メモリ 1 0 4 へのアクセスにおける制御動作を説明するタイミングチャートである。以下、図 1 と図 2 を用いて、CPU b 2 0 0 が共有メモリ 1 0 4 にアクセスする際の動作を説明する。

10

【0048】

(1) CPU b 2 0 0 が共有メモリ 1 0 4 にアクセスしていないときは、WAIT 信号は“H”となっている。CPU b 2 0 0 は、共有メモリ 1 0 4 にアクセスするために、CS b 0 を“L”にアサートする(図 2 の▲1▼)。

【0049】

(2) バス調停回路 1 0 5 は、CS b 0 の“L”極性を検出し、CPU a 1 0 1 に対してバス開放要求信号 BREQ を“H”にアサートする(図 2 の▲2▼)。この時点では、WAIT 信号を“H”に保ち CPU b 2 0 0 をウェイト状態にさせておく。ウェイト状態のときは、バス調停回路 1 0 5 は CPU b アドレスバス 2 1 1 からのアドレス入力を共有アドレスバス 1 1 1 へ出力しない。

20

【0050】

(3) CPU a 1 0 1 は、BREQ 信号を受けて共有バスへのアクセスを停止し、CPU b に共有バスを開放し、開放が完了するとバス解放完了信号 BACK を“H”にアサートする(図 2 の▲3▼)。

【0051】

(4) バス調停回路 1 0 5 は、BACK 信号を確認すると、CPU b アドレスバス 2 1 1 からのアドレス入力を共有アドレスバス 1 1 1 に出力し、CPU b 2 0 0 からのアクセスを実行させ、WAIT 信号を“L”にアサートして CPU b 2 0 0 へのウェイト状態を解除する(図 2 の▲4▼)。

【0052】

(5) CPU b 2 0 0 は、WAIT 信号の“L”極性を検出してウェイト状態が解除されたことを知ると共有メモリ 1 0 4 へのアクセスを行う。アクセスが完了すると CS b 0 信号をネゲートする(図 2 の▲5▼)。

30

【0053】

(6) バス調停回路 1 0 5 は CS b 0 信号のネゲートを確認すると、WAIT 信号をネゲートするとともに、BREQ 信号をネゲートしてバス使用权を CPU a 1 0 1 に戻す。バス使用权が戻ると CPU a 1 0 1 は BACK 信号をネゲートする。

【0054】

なお、上述の動作と平行して、CPU a 1 0 1 と CPU b 2 0 0 とは割り込み信号 IRQ 0、IRQ 1 を用いて互いに割り込み制御をする。例えば、割り込みによって CPU a 1 0 1 あるいは CPU b 2 0 0 は共有バスへのアクセスを停止する。割り込み信号 IRQ については後でさらに詳述する。

40

【0055】

CPU b 2 0 0 が共有メモリ 1 0 4 にアクセス中に CPU a 1 0 1 が共有バスにアクセスしようとしたときは、BREQ 信号が“H”であるために CPU a 1 0 1 のアクセスは許可されない。

【0056】

CPU b 2 0 0 が被制御デバイス b 2 0 1 をアクセスしているときには CS b 0 信号はアサートされないので、CPU a 1 0 1 は共有バスを開放する必要は無く、CPU a 1 0 1 は継続して ROM 1 0 2 から処理プログラムを読み出し実行することができる。

50

【0057】

以上のように、CPUb200がWAIT信号を用いたハンドシェーク機能を使用することにより、CPUb200がCPUa101に優先して動作することができる。これにより、特にCPUb200がシステム全体の制御を司る場合などは、システム全体のパフォーマンス低下に与える影響を最小限に抑えることができる。

【0058】

なお、図1の構成において、バス調停回路105はCPUa101の動作を完全に停止させる機能を有していても良い。CPUa101が停止している場合は、バス調停回路105はBACK信号を内部で“H”に固定することにより、前記(2)、(3)のフェーズをスキップしてCPUb200からのアクセスを実行することができる。これにより、例えばCPUを内蔵しないシステムLSIに対して作成された処理プログラムにおいても、CPUb200を用いてLSIを動作させることが可能となるので、処理プログラムの互換性を高めることができる。

【0059】

次に、本発明の第2の実施形態について説明する。第2の実施形態においては、CPU内蔵システムLSI100の中のROM部分をRAMとしている。近年のソフトウェアはその複雑さから潜在バグが多く、処理プログラムをROM化することは極めてリスクが高い。

【0060】

そこで、近年ではフラッシュROMのような不揮発性メモリをCPUに外付けし、このフラッシュROMに処理プログラムを格納して書き換え可能にしておくことが多い。また、フラッシュROMはアクセス速度が遅いためCPUの処理速度が低下する。そこで、LSIに内蔵RAMを設け、ブート時に処理プログラムをフラッシュROMからRAMにロードし、RAM上の処理プログラムでCPUを動作させる手法をとる場合がある。

【0061】

しかしながら、図1のようにCPUが複数存在するシステムでは、CPU毎にフラッシュROMを設けることはコスト面から望ましくない。そこで、第2の実施形態においては、CPU内蔵システムLSI100に処理プログラム格納用のRAMを設け、CPUb200は外付けのフラッシュROMで動作する環境にするとともに、CPUa101とCPUb200の両CPUの処理プログラムを1個のフラッシュROMに格納して、ブート時にCPUb200がCPUa101用の処理プログラムをRAMにロードする形態とする。

【0062】

図3は、上記のように構成された、本発明の第2の実施形態に係るCPU内蔵LSIを用いたシステム構成を示すブロック図である。図3において、図1と同じ構成要素には同一符号を付し、それぞれの説明を省略する。図3においては、新たにCPUbが制御するフラッシュROM202が用意され、また、ROM102の代わりに処理プログラムを格納するRAMとしてIRAM106が共有バスに接続されている。

【0063】

CPUb200は、CPUbアドレスバス211およびCPUbデータバス212を介して、CPU内蔵システムLSI100、被制御デバイスb201、フラッシュROM202の3つのデバイスを制御する。このとき、CPUb200は、これらの3つのデバイスを3本チップセレクト信号CSb0、CSb1、CSb2を用いてそれぞれアクセスする。

【0064】

フラッシュROM202はCPUb200の処理プログラムおよびCPUa101の処理プログラムを格納しており、CPUb200はフラッシュROM202に格納された処理プログラムを読み出しながら処理を実行する。

【0065】

IRAM106には、起動時にCPUb200によりCPUa101の処理プログラムが格納され、CPUa101はIRAM106に格納された処理プログラムを読み出しなが

ら処理を実行する。

【0066】

バス調停回路105には、IRAM106への読み出し／書き込みアクセス窓口としてアドレス指定レジスタ107とデータレジスタ108を設けており、CPUb200からIRAM106へアクセスする際は、アドレス指定レジスタ107とデータレジスタ108を介してアクセスを行う。

【0067】

第2の実施形態において、通常動作は第1の実施形態と同様である。以下、第2の実施形態における新たな機能であるブート時の動作を説明する。

【0068】

(1)電源オン時、CPUa101は処理プログラムの実行を行わずに、CPUb200からの通知を待つ。

【0069】

(2)CPUb200は、フラッシュROM202からCPUb200用の処理プログラムを読み出しながら処理実行を開始する。

【0070】

(3)CPUb200は、フラッシュROM202に格納されたCPUa101用の処理プログラムを読み出し、これをバス調停回路105に設けられたアドレス指定レジスタ107とデータレジスタ108を介してIRAM106へ書き込む。

【0071】

(4)具体的には、CPUb200はフラッシュROM202からCPUa101用の処理プログラムの先頭データを読み出し、アドレス指定レジスタ107にIRAMの先頭アドレスを書き込み、データレジスタ108に先頭データを書き込む。バス調停回路105はアドレス指定レジスタ107に書き込まれたIRAMのアドレスに、データレジスタ108に書き込まれたデータを書き込む。

【0072】

(5)次に、CPUb200はフラッシュROM202から次のデータを読み出し、アドレス指定レジスタ107にIRAMの前回アドレス+1を書き込み、データレジスタ108に次のデータを書き込む。バス調停回路105はアドレス指定レジスタ107に書き込まれたIRAMのアドレスに、データレジスタ108に書き込まれたデータを書き込む。

【0073】

(6)CPUb200は(5)を繰り返すことにより、すべてのCPUa101用の処理プログラムを読み出しIRAM106に書き込む。

【0074】

(7)CPUb200は、CPUa101用の処理プログラムのIRAM106への書き込みが完了すると、CPUa101に完了通知を行う。

【0075】

(8)CPUa101は、CPUb200からの完了通知を受けると、IRAM106に格納された処理プログラムを読み出しながら処理の実行を開始する。

【0076】

以上のように、処理プログラムをIRAM106にロードすることにより、CPUa101の処理速度を向上させることができる。また、フラッシュROM202をCPUA101とCPUb200とで共有にすることによりフラッシュROMの個数を削減でき、ひいてはシステム全体のコストを削減することができる。

【0077】

また、アドレス指定レジスタ107およびデータレジスタ108を設け、これらを介してCPUb200がIRAM106への処理プログラムの書き込みを行うことにより、CPUb200からアクセス指定するアドレスとして、IRAM106全体ではなくレジスタのアドレスを指定すれば良いだけになるため、CPUb200からアクセスするメモリ空間を削減することができる。これにより、CPU内蔵システムLSI100とCPUb2

10

20

30

40

50

00間のアドレスバスの端子数を削減することができる。

【0078】

また、アドレス指定レジスタ107は、データレジスタ108への書き込みが行われる度にインクリメントされるようにすることができる。これにより、上述したようにCPUa101用の処理プログラムをIRAM106の連続したアドレスに書き込む場合は、アドレス指定レジスタ107を毎回書き換える必要が無くなる。

【0079】

次に、本発明における第3の実施形態について説明する。第3の実施の形態は、CPUa101とCPUb200が割り込み信号によって互いに通知を行う方法に関するものである。

10

【0080】

図4は、本発明の第3の実施形態に係るCPU内蔵LSIを用いたシステム構成を示すブロック図である。図4において、図1と同じ構成要素には同一符号を付し、それぞれの説明を省略する。また、共有バス、ROM、被制御デバイス、共有メモリについては、本実施形態の説明に直接関係無いので省略している。

【0081】

図4において、バス調停回路105にはCPUa用割り込み要因レジスタ109とCPUb用割り込み要因レジスタ110が設けられている。CPUa用割り込み要因レジスタ109およびCPUb用割り込み要因レジスタ110は、それぞれ図5に示すようにnビットのレジスタ構成となっており、全ビットの論理和がそれぞれ図4における割り込み信号IRQ0、IRQ1としてアサートされる。

20

【0082】

すなわち、割り込み要因レジスタのいずれかのビットに“1”が書き込まれると、該当する割り込み信号IRQ0またはIRQ1がアサートされる。CPUa用割り込み要因レジスタ109またはCPUb用割り込み要因レジスタ110は、各ビットを特定の割り込み要因に固定することなく、プログラマが自由にビット割り当てできる構造になっている。

【0083】

CPUa101からCPUb200へ、あるいはCPUb200からCPUa101への割り込みによる通知内容はいろいろな種類が考えられる。例えば、図2における共有メモリ104への書き込み完了やIRAMへの処理プログラム書き込み完了の通知に使用することができ、その他にも処理プログラムの内容によって多くの通知内容が発生すると考えられ、本実施の形態においては割り込み要因レジスタを設けることにより、そのビットをプログラマが自由に割り込み要因に割り付けることができ、処理プログラムの自由度を高めている。

30

【0084】

ここで動作例として、CPUa101がCPUb用割り込み要因レジスタ110を使用してCPUb200へ割り込み通知を行う処理の流れについて、図4を用いて説明する。

【0085】

(1) CPUa101はCPUb用割り込み要因レジスタ110のビット0に“1”を書き込む。

40

【0086】

(2) バス調停回路105は、CPUb用割り込み要因レジスタ110の全ビットの論理和が“1”になったので、割り込み信号IRQ1をアサートする。

【0087】

(3) CPUb200は、割り込み信号IRQ1がアサートされると、割り込みを受け付けるとともにCPUb用割り込み要因レジスタ110を読み出し、ビット0が示す割り込み要因が通知されたことを認識する。

【0088】

(4) CPUb200は、CPUb用割り込み要因レジスタ110のビット0に“0”を書き込むことによりビット0をクリアする。

50

【0089】

なお、本発明は上記実施形態1から実施形態3に限定されるものではない。例えば、上記実施形態ではCPUが2つの場合で説明したが、CPUが2つ以上であっても同様の効果を得ることができる。また、CPUaの共有バスが外部に拡張されているかどうかは本発明においては問わない。また、信号の“H”と“L”をどの状態に割り振るかが問われないことは言うまでもない。

【0090】

また、1つのチップ内に複数のCPUを備えるマルチCPUシステムLSIにおいても、バスを複数に分け本発明のバス調停回路を設けることにより、複数のCPUが複数のバスに接続されたデバイスをそれぞれ制御可能となるので、マルチCPUシステムの処理効率を向上させることができる。 10

【0091】

【発明の効果】

以上説明したように、本発明によれば、システムLSIに内蔵されたCPUと外部CPUがバスを共有する環境において、CPU内蔵システムLSIにバス調停回路を設けることにより、CPU内蔵システムLSI内の共有バスと、外部CPUが接続される外部拡張バスを分離させ、外部CPUがその被制御デバイスへのアクセス中であっても、内蔵CPUは共有バスに接続されたROMデバイスをアクセスして処理プログラムを読み出し実行することができ、内蔵CPUの性能低下を回避できるという優れた効果が得られる。

【0092】

さらに本発明によれば、バス調停回路はバス開放要求信号とバス開放完了信号を用いた調停制御だけでなくチップセレクト信号とウェイト信号を用いた調停機能を備えることにより、外部CPUがバス調停機能を備えていなくてもバス調停を実現することができる。また、前記ウェイト信号を用いたハンドシェイク機能を使用することにより、外部のCPUを優先動作させることができ、外部CPUが制御するシステムのパフォーマンス低下に与える影響を最小限に抑えることができる。 20

【0093】

さらに本発明によれば、CPU内蔵システムLSIは内蔵CPU用のRAMを備え、外部CPUからRAMへ処理プログラムをロードすることにより、処理プログラムをROM化するリスクを避けながら低コストのシステムを実現することができる。 30

【0094】

さらに本発明によれば、バス調停回路に割り込みの要因のビット割り当て設定がプログラム可能な割り込み要因レジスタを備えることにより、内蔵CPUと外部CPU間の割り込み通知の要因をプログラマブルに実装することができる。

【図面の簡単な説明】

【図1】本発明の第1の実施形態に係るCPU内蔵LSIを用いたシステム構成を示すブロック図。

【図2】共有メモリへのアクセスにおける制御動作を説明するタイミングチャート。

【図3】本発明の第2の実施形態に係るCPU内蔵LSIを用いたシステム構成を示すブロック図。 40

【図4】本発明の第3の実施形態に係るCPU内蔵LSIを用いたシステム構成を示すブロック図。

【図5】割り込み要因レジスタの構成図。

【図6】2つのCPUがバスを共有する従来のシステム構成を示すブロック図。

【符号の説明】

100 CPU内蔵システムLSI

101 CPUa

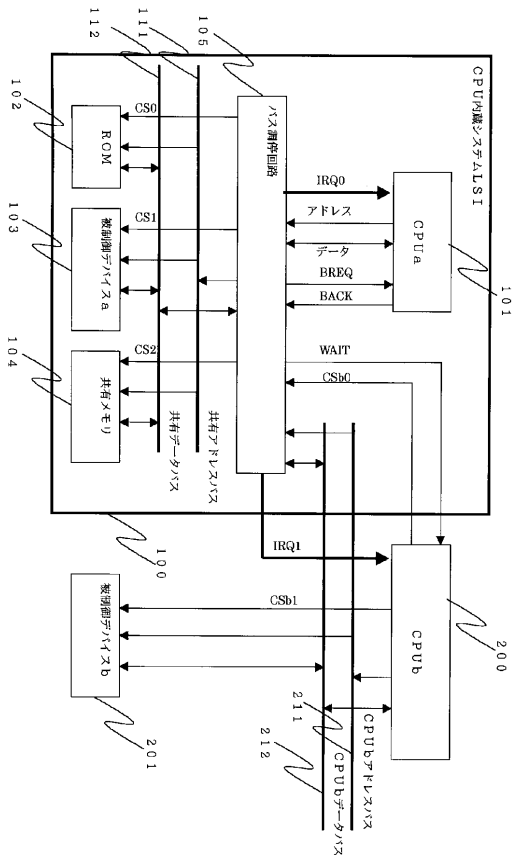
102 ROM

103 被制御デバイスa

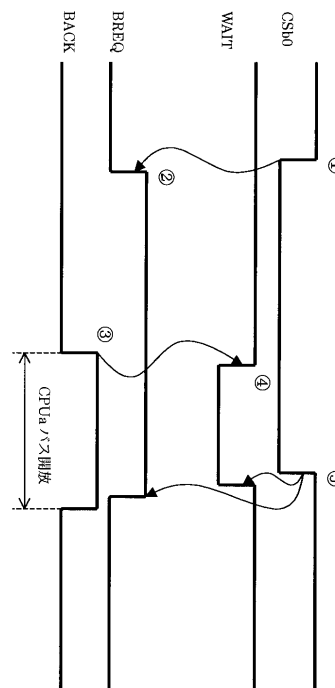
104 共有メモリ 50

- 105 バス調停回路
- 106 RAM
- 107 アドレス指定レジスタ
- 108 データレジスタ
- 109 CPUa用割り込み要因レジスタ
- 110 CPUb用割り込み要因レジスタ
- 111 共有アドレスバス
- 112 共有データバス
- 200 CPUb
- 201 被制御デバイスb
- 202 フラッシュROM
- 211 CPUbアドレスバス
- 212 CPUbデータバス

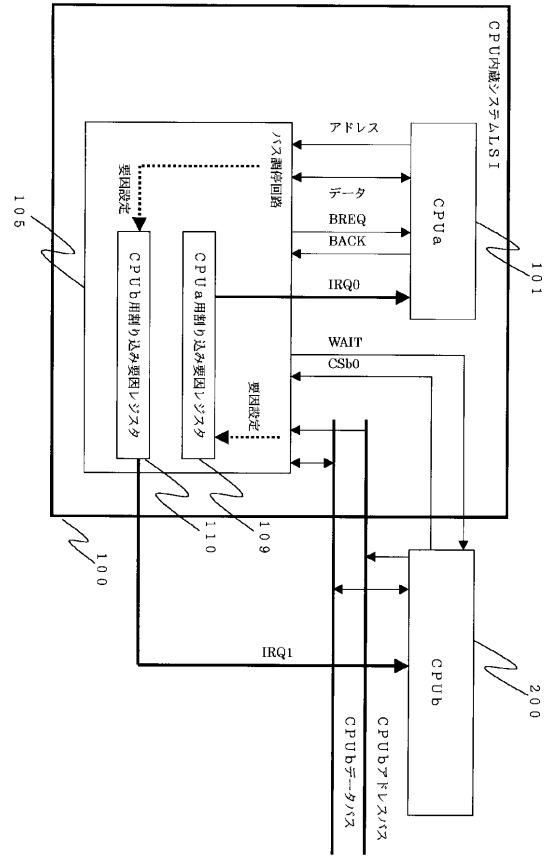
【図1】



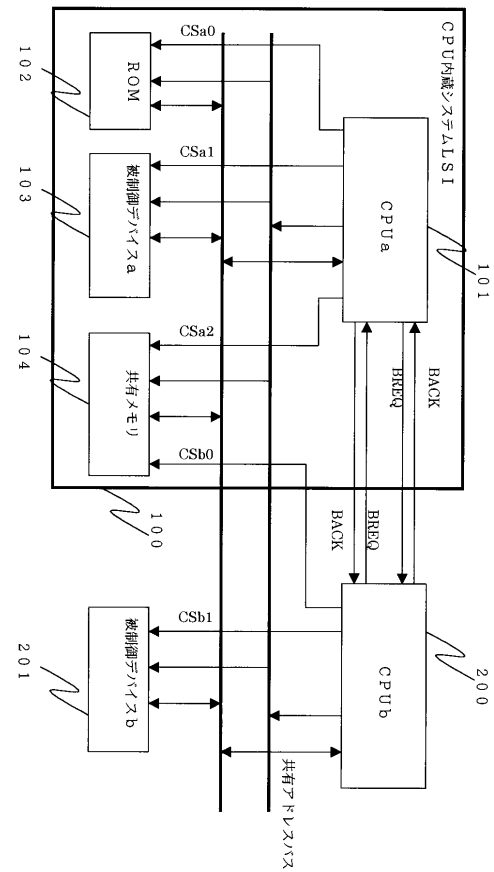
【図2】



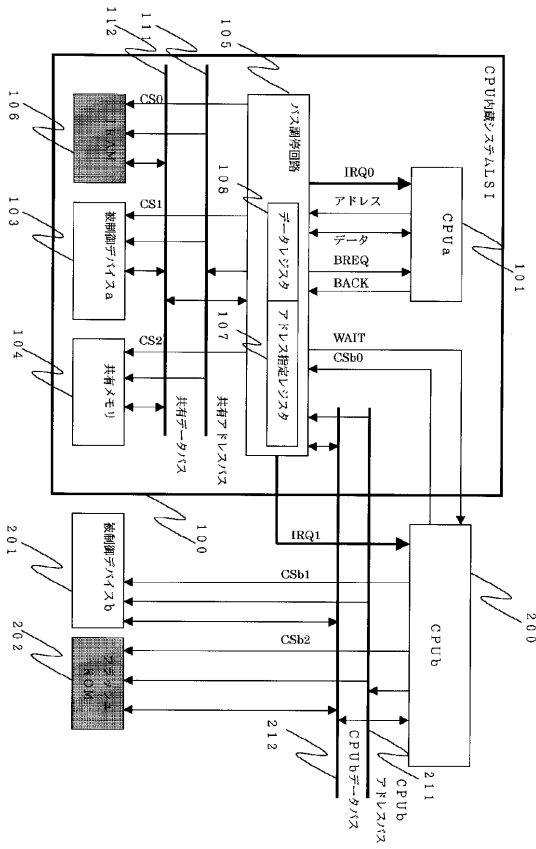
【図 4】



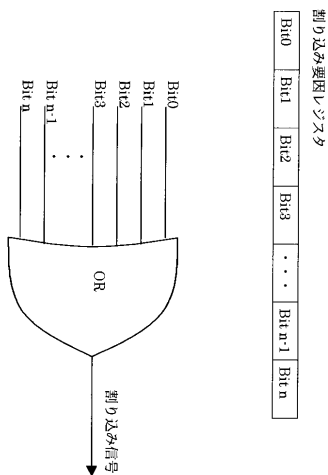
【図 6】



【図 3】



【図 5】



フロントページの続き

(72)発明者 石村 勇

大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

(72)発明者 町田 忍

大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

Fターム(参考) 5B061 BB04