(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3599499号
(P3599499)
(45)発行日　平成16年12月8日(2004.12.8)　(24)登録日　平成16年9月24日(2004.9.24)

(51)Int.Cl.[7]　F I
G06F　9/38　G06F　9/38　３１０B

請求項の数 4　(全 9 頁)

(21)出願番号　特願平8-283657
(22)出願日　平成8年10月25日(1996.10.25)
(65)公開番号　特開平10-124312
(43)公開日　平成10年5月15日(1998.5.15)
審査請求日　平成13年8月22日(2001.8.22)

(73)特許権者　000006747
株式会社リコー
東京都大田区中馬込１丁目３番６号
(74)代理人　100062144
弁理士　青山　葆
(74)代理人　100086405
弁理士　河宮　治
(74)代理人　100104695
弁理士　島田　明宏
(72)発明者　原　和彦
東京都大田区中馬込１丁目３番６号　株式会社リコー内

審査官　後藤　彰

最終頁に続く

(54)【発明の名称】中央処理装置

(57)【特許請求の範囲】
【請求項１】
メモリから命令を読み出して解読し実行するまでの処理をパイプライン方式により行う中央処理装置において、
実行すべき命令のオペランドの演算のための演算器とは別個に設けられた加算器であって、該オペランドのアドレスを計算するための専用の加算器と、
内部の特定レジスタに格納されたデータを前記加算器へ入力データとして転送するための専用の第１データ経路と、
実行すべき命令の即値フィールドに設定された即値データを前記加算器へ入力データとして転送するための専用の第２データ経路と、
前記メモリから読み出された命令のオペランドのアドレスが前記特定レジスタに格納されたデータと前記即値データとの加算により算出できるか否かを判定する判定手段を有し、該判定手段において前記加算により算出できると判定された場合に、前記特定レジスタに格納されたデータが前記第１のデータ経路を経て前記加算器へ転送され、前記即値データが前記第２データ経路を経て前記加算器へ転送され、転送された両データから前記加算器により前記アドレスが算出されるように、前記第１および第２データ経路ならびに前記加算器を制御し、また、該判定手段において前記加算により算出できると判定されない場合には、前記演算器により前記アドレスが算出されるように前記演算器を制御する制御手段と、
を備えることを特徴とする中央処理装置。

【請求項２】
請求項１に記載の中央処理装置において、
前記特定レジスタがプログラムカウンタであることを特徴とする中央処理装置。
【請求項３】
請求項１に記載の中央処理装置において、
前記特定レジスタがスタックポインタレジスタであることを特徴とする中央処理装置。
【請求項４】
請求項１に記載の中央処理装置において、
前記特定レジスタがフレームポインタレジスタであることを特徴とする中央処理装置。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】
本発明は、パイプライン処理を行うマイクロプロセッサ等の中央処理装置（ＣＰＵ）に関する。
【０００２】
【従来の技術】
マイクロプロセッサ等の中央処理装置（ＣＰＵ）の性能を向上させるための手段の一つとして、パイプライン処理が利用されている。パイプライン処理は、命令を読み取って解読し実行するまでをステージと呼ばれるいくつかの段階に分け、複数の命令の各ステージを同時に実行することにより、次々に命令が実行されている状態において１命令当たりの処理に要する時間を短縮するものである。
【０００３】
ＣＰＵで命令を実行するには、その実行に使用されるデータすなわちオペランドのアドレスを指定する必要がある。このアドレスの指定方法（アドレス指定モード）としてどのようなものを用意するかは、そのＣＰＵを特徴づけるものであり、各命令では、そのＣＰＵに用意されたいくつかのアドレス指定モードのうちのいずれかのモードでその命令のオペランドのアドレスが指定される。そしてＣＰＵは、各命令を実行する際に、その命令のアドレス指定モードに基づきその命令で指定されたアドレスデータを用いてその命令のオペランドのアドレス（実効アドレス）を算出する。
【０００４】
この実効アドレスの算出には必要に応じて演算器が使用されるが、アドレス演算専用の演算器が設けられていないＣＰＵでは、オペランドの演算処理（データ処理）に使用されるＡＬＵがこのアドレス計算にも使用される。この場合、オペランドの演算とアドレス計算とはパイプライン上における同一ステージとして扱われることになる。その結果、或る命令のアドレス計算は、その前の命令のオペランドの演算が終了するまで行うことができない。この点につき、図３に示すタイミングチャートを参照しつつ説明する。
【０００５】
図３に示した例では、パイプラインのステージは命令フェッチのステージ、デコードのステージ、および演算処理のステージという三つのステージから成り、サイクル１では、或る命令Ａのデコード（解読）が行われると共に次の命令Ｂのフェッチが行われる。次のサイクル２では、命令Ａのデータ処理が行われると共に命令Ｂの解読が行われる。しかし、この例ではアドレス専用の演算器が設けられていないため、このサイクル２において命令Ｂのオペランドのアドレスを算出することはできない。その結果、その次のサイクル３でＡＬＵにおいて命令Ｂのアドレス計算が行われ、サイクル４で命令Ｂの演算処理が行われることになる。このように、１命令当たり２回以上実行されるステージ（図３の例におけるＡＬＵでの演算ステージ）が存在すると、そのステージよりも前段のステージ（命令フェッチおよびデコードのステージ）は待機させられ動作が停止する。
【０００６】
【発明が解決しようとする課題】
上記のような事態を回避するために、アドレス計算とオペランドの演算処理とを別のステ

ージにすることが考えられる。これにより、或る命令がオペランドの演算処理のステージにあるとき、別のステージであるアドレス計算のステージで次の命令のオペランドのアドレスを算出することができる。

【０００７】

しかし、この場合には、アドレス計算にＡＬＵを使用することができないため、アドレス計算専用の演算器（以下「アドレス用演算器」という）を別途設ける必要がある。このアドレス用演算器は、そのＣＰＵが用意する全てのアドレス指定モードでのアドレス計算を実行できなければならず、複雑なアドレス指定モードが用意されている場合にはそれに応じてアドレス用演算器やそれへのデータ経路等の構成が複雑なものとなる。例えば、或るレジスタの値と別のレジスタの値との加算結果を目的のアドレスとする場合には、それらのレジスタからアドレス用演算器へのデータ経路を確保しなければならない。加算する一方がレジスタの値ではなく命令コードに設定された即値の場合には、その即値をアドレス用演算器へ入力するデータ経路を確保しなければならない。目的のアドレスを得るのに複数回演算しなければならない場合は、演算結果を一時的に保持する手段と、そこから再びその演算結果をアドレス用演算器へ入力するデータ経路の確保も必要となる。

【０００８】

また、ＣＰＵ内部の特定のレジスタをベースレジスタとして使用するアドレス指定モードによりオペランドのアドレスを算出する場合には、ハザードの問題を考慮しなければならない。ここでのハザードの問題とは、或る命令がＣＰＵ内部のレジスタの値を更新した場合、その後続の命令は更新後の値を使用しなければならないが、パイプラインのステージがオーバラップしてくると、或る命令がレジスタの値を更新する前に後続の命令がそのレジスタの更新前の値を参照する可能性が生じる、という問題をいう。この問題を回避するためには、ハザードを検出して後続のステージをストール（停止）させるか、または更新する予定のデータをバイパスする（レジスタを介さずに横流しする）などの工夫が必要になる。

【０００９】

このように、アドレス計算とオペランドの演算処理とを別のステージとした場合には、ＣＰＵの用意する全てのアドレス指定モードに対応できるアドレス用演算器が必要となって回路規模が増大したり、パイプライン処理におけるハザードの問題に対処するためにパイプライン処理における制御が複雑化したりする。そして、これらはＣＰＵのコストを上昇させることになる。

【００１０】

本発明は上記問題を解決するためになされたものであり、その目的とするところは、回路規模の増大や制御の複雑化を抑えつつ、パイプライン処理によってアドレス計算をオペランドの演算処理と同時に行うことにより処理速度を向上させた中央処理装置を提供することにある。

【００１１】

【課題を解決するための手段】

上記課題を解決するために成された本発明では、メモリから命令を読み出して解読し実行するまでの処理をパイプライン方式により行う中央処理装置において、

実行すべき命令のオペランドの演算のための演算器とは別個に設けられた加算器であって、該オペランドのアドレスを計算するための専用の加算器と、

内部の特定レジスタに格納されたデータを前記加算器へ入力データとして転送するための専用の第１データ経路と、

実行すべき命令の即値フィールドに設定された即値データを前記加算器へ入力データとして転送するための専用の第２データ経路と、

前記メモリから読み出された命令のオペランドのアドレスが前記即値データと前記特定レジスタに格納されたデータとの加算により算出できるか否かを判定する判定手段を有し、

該判定手段により算出できると判定された場合に、前記特定レジスタに格納されたデータが前記第１のデータ経路を経て前記加算器へ転送され、前記即値データが前記第２データ

経路を経て前記加算器へ転送され、転送された両データから前記加算器により前記アドレスが算出されるように、前記第１および第２データ経路ならびに前記加算器を制御し、また、該判定手段において前記加算により算出できると判定されない場合には、前記演算器により前記アドレスが算出されるように前記演算器を制御する制御手段と、
を備えた構成としている。
【００１２】
なお、本発明の中央処理装置において、プログラムカウンタ、スタックポインタレジスタ、またはフレームポインタレジスタのうちのいずれかを上記特定のレジスタとするのが好ましい。
【００１３】
【発明の効果】
本発明によれば、命令のオペランドのアドレスが特定レジスタのデータと命令コード中の即値データとの加算により算出されるディスプレースメント付きレジスタ間接アドレス指定モードの場合には、第１および第２データ経路とアドレス計算専用の加算器を用いて、他の命令のオペランドの演算処理と並行してアドレス計算を行うことができ、これにより命令の実行に必要なサイクル数を１サイクルだけ低減することができる。また本発明では、アドレス計算をオペランドの演算処理と同時に行うための機構を、アドレス計算専用の加算器を中心として第１および第２データ経路を含む比較的少量のハードウェアで実現できる。したがって、比較的少ないハードウェア量の追加で中央処理装置の処理速度を向上させることができる。
【００１４】
本発明に係る好ましい中央処理装置では、プログラムカウンタ、スタックポインタレジスタ、またはフレームポインタレジスタのうちのいずれかが上記特定レジスタとして使用され、上記処理速度の向上において大きな効果が得られる。すなわち、プログラムカウンタの値に即値データを加算するアドレス指定モードは分岐命令で相対分岐として多用され、スタックポインタに即値データを加算するアドレス指定モードはスタックに蓄積された情報を参照する場合に多用され、フレームポインタに命令コード中の即値データを加算するアドレス指定モードは高級言語における関数コール（サブルーチンコール）によって形成されたスタックフレームの情報を参照する場合に多用され、これらのアドレス指定モードはいずれも使用頻度が高い。したがって、これらのアドレス指定モードの命令のサイクル数の低減は中央処理装置による処理の高速化に大きく寄与する。
【００１５】
なお本発明のように、アドレス計算専用の加算器での演算に使用されるレジスタをプログラムカウンタやスタックポインタレジスタ、フレームポインタレジスタなど、汎用レジスタ以外の特定のレジスタに限定すると、そのようなレジスタの値の更新をサイクルの前半に限定する等の方法により、アドレス計算においてそれらのレジスタの値が読み出される前に更新されるようにすることができる。これにより、パイプライン処理における制御の複雑化を抑えつつ、ハザードの発生を防止することができる。
【００１６】
以上のように本発明によれば、オペランドの演算処理と同時にアドレス計算を行えるアドレス指定モードを限定する（アドレス計算に使用するレジスタを限定する）ことによりアドレス計算専用のハードウェア量の増大やパイプライン処理における制御の複雑化を抑えつつ、その限定すべきアドレス指定モードとして使用頻度の高いものを選定することによりアドレス計算に要する時間の短縮化の効果を高めることができる。したがって、コスト上昇を抑えつつ処理速度を向上させた中央処理装置を提供することができる。
【００１７】
【発明の実施の形態】
＜実施形態の構成＞
図１は、本発明の一実施形態であるマイクロプロセッサの要部の構成を示すブロック図である。このマイクロプロセッサは、プリフェッチキュー１０を備え、外部のメモリ（図示

せず）に格納された命令を先読みしてプリフェッチキューに蓄える。そして命令を解読する部分として、命令実行のために各部を制御する信号を生成するメインデコーダ１６に加えて、メインデコーダ１６による命令の解読に先立ってアドレス計算の種類を判定するために命令の部分的な解読を行うプリデコーダ１４を備えている。演算器としては、オペランド用の演算器（データ処理用演算器）である算術論理演算器（ＡＬＵ）３５に加えて、オペランドのアドレスを算出するためのアドレス計算専用の加算器（以下「アドレス用加算器」という）２５を備えている。内部レジスタとしては、プログラムカウンタ２１、スタックポインタレジスタ（以下「ＳＰレジスタ」という）２２、アドレス格納レジスタ３０、および汎用レジスタ３６、３７などを備えている。データ経路としては、ＡＬＵ３５や内部のレジスタ等との間でデータを転送するための内部データバス４０およびアドレス格納レジスタ３０に格納されたアドレスを外部に転送するために使用される内部アドレスバス３２を備えると共に、プログラムカウンタ２１およびＳＰレジスタ２２に格納されたデータをアドレス用加算器２５へ入力するための第１アドレスデータ経路２３、およびプリフェッチキューに蓄えられた命令のコードの中に設定された即値データをアドレス用加算器２５に入力するための第２アドレスデータ経路２４を備えている。

【００１８】

＜実施形態の動作＞

次に、上記構成のマクロプロセッサの動作を図２を参照しつつ説明する。

図２は、本発明の実施形態である上記マイクロプロセッサの動作を示すタイミングチャートである。図２では、理解を容易にするために、命令フェッチ、デコード、アドレス計算、および演算処理の各ステージのタイミングチャートが示されているが、デコードのステージとアドレス計算のステージとは本実施形態におけるパイプラインのステージとしては一つのものである。

【００１９】

本実施形態では、主記憶である所定のメモリ（図示せず）に格納された命令が、プログラムカウンタ２１の値に基づいて先読みされ、外部バスを経てプリフェッチキュー１０に蓄えられる。プリフェッチキュー１０に蓄えられた命令は、順次、プリデコーダ１４およびメインデコーダ１６で解読される。図２に示すサイクル１では、命令Ｂは命令フェッチのステージにあって、命令Ｂが主記憶から読み出されてプリフェッチキュー１０の蓄えられる。このとき、その直前の命令Ａはデコードのステージにあって、命令Ａがメインデコーダ１６で解読される。前述のようにデコードのステージはアドレス計算のステージと同一のパイプライン・ステージであるため、このサイクル１では、解読される命令のオペランドのアドレス計算も必要に応じて行われる。ただし、図２に示した例における命令Ａはオペランドのアドレス計算が不要な命令である。

【００２０】

次のサイクル２では、命令Ａは演算処理のステージに進み、メインデコーダ１６での命令Ａの解読結果に基づきＡＬＵ３５においてオペランドの演算処理が行われる。また、命令Ｂはデコードのステージに進んでいて、命令Ｂがメインデコーダ１６で解読される。このときプリデコーダ１４においても命令Ｂの部分的な解読が行われ、この解読結果は同一サイクルでアドレス計算をする際に使用される。すなわち、プリデコーダ１４により、命令Ｂのオペランドのアドレスをアドレス用加算器２５で算出できるか否かの判定や、アドレス計算に使用するレジスタを示す情報の抽出が行われる。前述のように、アドレス用加算器２５には、プログラムカウンタ２１およびＳＰレジスタ２２からデータを入力するための第１アドレスデータ経路２３と、プリフェッチキュー１０から即値データを入力するための第２アドレスデータ経路２４とが接続されているため（図１参照）、プログラムカウンタ２１またはＳＰレジスタ２２に格納されたデータに命令コードに設定された即値データを加算するディスプレースメント付きレジスタ間接アドレス指定モードの場合には、アドレス用加算器２５を用いてアドレス計算を行うことができる。プリデコーダ１４は、サイクル２において、命令Ｂのアドレス指定モードがこのようなディスプレースメント付きレジスタ間接アドレス指定モードか否かを判定し、その判定結果に応じて第１および第２

アドレスデータ経路２３、２４や、アドレス用加算器２５、ＡＬＵ３５などを制御する。これにより、アドレス計算が必要な場合において、上記のディスプレースメント付きレジスタ間接アドレス指定モードのときはアドレス用加算器２５でアドレス計算が行われ、上記以外のアドレス指定モードのときはＡＬＵ３５でアドレス計算が行われる。

【００２１】

図２に示した例では、命令Ｂのアドレス指定モードは上記のディスプレースメント付きレジスタ間接アドレス指定モードであると判定され、サイクル２において命令Ｂのオペランドのアドレス計算がアドレス用加算器２５で行われる。図１からわかるように、このアドレス計算に使用されるアドレス用加算器２５や第１および第２アドレスデータ経路２３、２４は、演算処理のステージに使用されるＡＬＵ３５や、内部データバス４０、汎用レジスタ３６、３７等とは独立に動作できる。したがって、命令Ｂのアドレス計算は命令Ａのオペランドの演算処理と同一のサイクル２で実行される。これにより、図３に示した従来例に比べ、命令の実行に必要なサイクル数が一つだけ低減されることになる。なお、上記のディスプレースメント付きレジスタ間接アドレス指定モードのアドレス計算において必要な演算は１回の加算のみであるため、アドレス用加算器２５でのアドレス計算は常に１サイクルで終了する。

【００２２】

上記のようにして算出された命令Ｂのオペランドのアドレスは一旦アドレス格納レジスタ３０に格納され、このアドレスは内部アドレスバス３２を経て外部へ出力される。これにより、外部の所定のメモリに格納されたオペランドが読み出されて内部に取り込まれ、内部バス４０を経てＡＬＵ３５や汎用レジスタ３６、３７などに転送される。

【００２３】

その後のサイクル３では、命令Ｂは演算処理のステージへ進み、メインデコーダ１６での命令Ｂの解読結果に基づき、外部から取り込まれた上記オペランドを用いてＡＬＵ３５において演算処理が行われる。

【００２４】

なお、上記サイクル２においてプリデコーダ１４による判定の結果、アドレス用加算器２５ではなくＡＬＵ３５でアドレス計算が行われる場合は、図３に示した従来例のように、命令Ａについての演算処理のステージが終了してから命令Ｂについてのアドレス計算が行われることになる。

【００２５】

ところで、上記サイクル２のようにマイクロプロセッサ内部のレジスタの値を用いてアドレス計算を行う場合にはパイプライン処理においてハザードが生じる可能性がある。すなわち、或る命令が演算処理のステージで内部レジスタを変更する場合において、その変更の前に後続の命令がアドレス計算のステージでその内部レジスタを参照する可能性があり、このような参照がなされると誤ったアドレス計算が行われる。しかし、本実施形態では、アドレス計算に使用される内部レジスタはプログラムカウンタ２１とＳＰレジスタ２２に限定されており、汎用レジスタはアドレス計算に使用されないため、この点を利用して、アドレス計算に使用される内部レジスタのデータがアドレス用加算器２５への入力経路で読み出される前に更新されるようにすることができ、これにより上記ハザードが回避される。このハザード回避の具体的な方法としては、アドレス計算に使用されるレジスタ２１、２２への入力がサイクルの前半で終了するようにすると共にアドレス用加算器２５がサイクルの後半まで有効に動作するようにマイクロプロセッサのハードウェアを設計する、という入力タイミングによる回避方法を使用することができる。また、本実施形態のマイクロプロセッサがマイクロプログラム制御方式の場合には、アドレス計算に使用されるレジスタの値を命令の最終サイクルで更新しないようにマイクロプログラムを作成する（例えば図２における命令Ａについては、サイクル２の演算処理のステージではプログラムカウンタ２１およびＳＰレジスタのいずれの内容も変更しないようにする）、というマイクロプログラミングによる回避方法を使用することができる。

【００２６】

＜実施形態の効果＞
上記実施形態によれば、プログラムカウンタ２１またはＳＰレジスタ２２に格納されたデータに命令コード中の即値データを加算するディスプレースメント付きレジスタ間接アドレス指定モードの命令については、そのアドレス計算が直前の命令のオペランドの演算処理と同時に行われかつ１サイクルで終了するため（図２参照）、命令の実行に必要なサイクル数が低減される。また、プログラムカウンタ２１の値に即値データを加算するアドレス指定モードは分岐命令で相対分岐として多用されるアドレス指定モードであり、ＳＰレジスタ２２の値に即値データを加算するアドレス指定モードはスタックに蓄積された情報を参照する場合に多用されるアドレス指定モードである。このため、これらのアドレス指定モードの命令のサイクル数が低減されることは、本マイクロプロセッサの高速化に大きく寄与する。一方、アドレス用加算２５でアドレス計算が行われるアドレス指定モードは上記のように限定されているため、簡単な構成のアドレス用加算器２５を中心とする比較的少ないハードウェア量の追加で上記のようなマイクロプロセッサの高速化が実現される。

【００２７】
また上記実施形態によれば、アドレス用加算器２５でアドレス計算が行われる場合、すなわち他の命令のオペランドの演算処理と同時にアドレス計算が行われる場合に使用されるレジスタはプログラムカウンタ２１またはＳＰレジスタ２２であって汎用レジスタではないため、この点を利用して、アドレス計算に使用される内部レジスタのデータがアドレス用加算器２５への入力経路で読み出される前に更新されるようにすることができ、ハザードを検出して後続のステージをストールさせ、または更新する予定のデータをバイパス（横流し）する等という従来のハザード対策を必要としない。このため、パイプライン処理における制御の複雑化を抑えつつ、ハザードの発生を防止することができる。

【００２８】
＜変形例＞
本実施形態では、アドレス用加算器２５でのアドレス計算に使用可能な内部レジスタはプログラムカウンタ２１とＳＰレジスタ２２であるが、汎用レジスタ以外の他の内部レジスタをこのアドレス計算に使用できるようにしてもよい。例えばフレームポインタレジスタ（以下「ＦＰレジスタ」という）を備えるマイクロプロセッサにおいては、プログラムカウンタ２１およびＳＰレジスタ２２に加えて又はこれらに代えてＦＰレジスタを使用できるようにしてもよい。この場合には、ＦＰレジスタが第１アドレスデータ経路２３に接続されることになる。

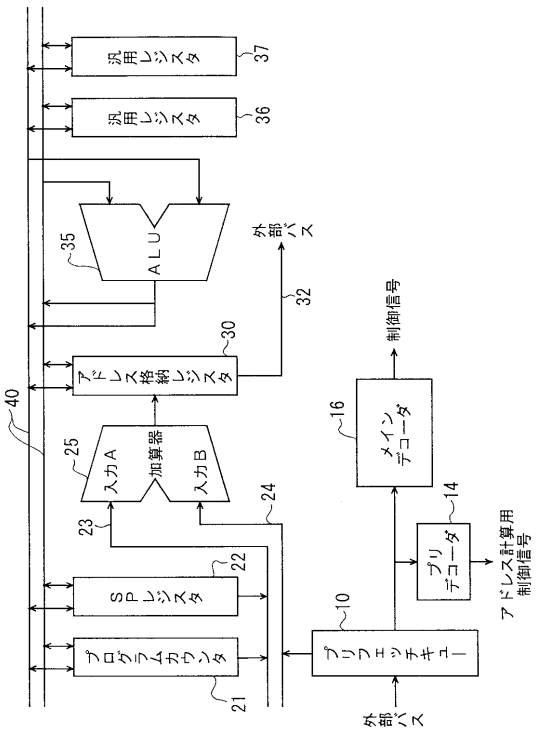
【００２９】
この場合、フレームポインタに命令コード中の即値データを加算するディスプレースメント付きレジスタ間接アドレス指定モードの命令について、そのアドレス計算が直前の命令のオペランドの演算処理と同時に行われかつ１サイクルで終了する。このため、命令の実行に必要なサイクル数が低減される。また、フレームポインタに命令コード中の即値データを加算するアドレス指定モードは、高級言語における関数コール（サブルーチンコール）によって形成されたスタックフレームの情報を参照する場合に多用されるアドレス指定モードである。このため、このアドレス指定モードの命令のサイクル数が低減されることも、本マイクロプロセッサの高速化に大きく寄与する。

【００３０】
なお、アドレス計算に使用されるフレームポインタが第１アドレスデータ経路２３で読み出される前に更新されるようにすることができるため、上記と同様の方法により、パイプライン処理における制御の複雑化を抑えつつハザードを回避することができる。

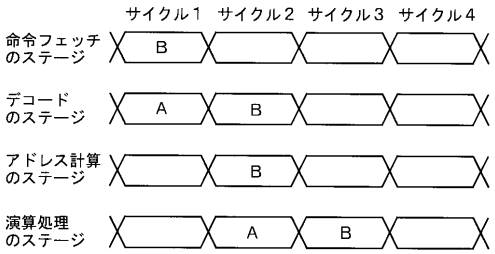
【図面の簡単な説明】
【図１】本発明の一実施形態であるマイクロプロセッサの要部の構成を示すブロック図。
【図２】上記実施形態のマイクロプロセッサの動作を示すタイミングチャート。
【図３】従来のマイクロプロセッサの動作を示すタイミングチャート。
【符号の説明】

１０　…プリフェッチキュー
１４　…プリデコーダ
１６　…メインデコーダ
２１　…プログラムカウンタ
２２　…スタックポインタレジスタ（ＳＰレジスタ）
２３　…第１アドレスデータ経路
２４　…第２アドレスデータ経路
２５　…アドレス用加算器
３０　…アドレス格納レジスタ
３５　…算術論理演算器（ＡＬＵ）

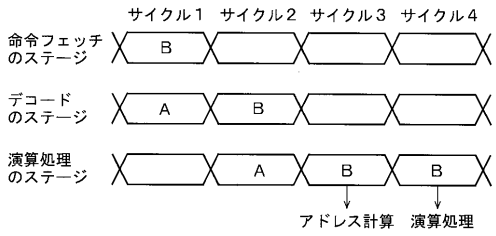
【図１】




【図２】




【図３】



フロントページの続き

(56)参考文献 特開平３－２２９３２４（ＪＰ，Ａ）
特開昭６３－５５６３８（ＪＰ，Ａ）
特開平２－５８１２８（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
G06F 9/38