

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第3780449号
(P3780449)

(45) 発行日 平成18年5月31日(2006.5.31)

(24) 登録日 平成18年3月17日(2006.3.17)

(51) Int. Cl. F I
G06F 9/38 (2006.01) G O 6 F 9/38 3 7 O X
G06F 9/46 (2006.01) G O 6 F 9/46 3 6 O B

請求項の数 12 (全 10 頁)

<p>(21) 出願番号 特願2001-94455 (P2001-94455) (22) 出願日 平成13年3月29日 (2001.3.29) (65) 公開番号 特開2001-306322 (P2001-306322A) (43) 公開日 平成13年11月2日 (2001.11.2) 審査請求日 平成15年4月18日 (2003.4.18) (31) 優先権主張番号 09/538669 (32) 優先日 平成12年3月30日 (2000.3.30) (33) 優先権主張国 米国 (US)</p> <p>前置審査</p>	<p>(73) 特許権者 500587067 アギア システムズ インコーポレーテッド アメリカ合衆国, 18109 ペンシルヴァニア, アレントウン, アメリカン パークウェイ エヌイー 1110</p> <p>(74) 代理人 100064447 弁理士 岡部 正夫</p> <p>(74) 代理人 100085176 弁理士 加藤 伸晃</p> <p>(74) 代理人 100096943 弁理士 臼井 伸一</p> <p>(74) 代理人 100101498 弁理士 越智 隆夫</p> <p style="text-align: right;">最終頁に続く</p>
--	--

(54) 【発明の名称】 マルチスレッド V L I W プロセッサにおいて機能単位を解放するための方法および装置

(57) 【特許請求の範囲】

【請求項 1】

マルチスレッド超長命令語プロセッサであって、
 マルチスレッド命令ストリームから命令を実行するための複数の機能単位と、
 前記機能単位のうちの少なくとも1つによって実行される現在実行されつつある命令が指示に応じて完了した時、該機能単位のうちの少なくとも1つを別のスレッドに割り当て直す機能単位解放機構であって、前記指示が、前記現在実行されつつある命令が完了するであろう時刻を示すものである機能単位解放機構を含むマルチスレッド超長命令語プロセッサ。

【請求項 2】

前記機能単位解放機構は、各機能単位が占有されるサイクルの数を監視する請求項 1 に記載のマルチスレッド超長命令語プロセッサ。

【請求項 3】

前記少なくとも1つの機能単位は状態マシンを含み、状態情報を維持する請求項 1 に記載のマルチスレッド超長命令語プロセッサ。

【請求項 4】

前記状態マシンは、前記少なくとも1つの機能単位が多数サイクルの命令によって占有されるサイクルの数を監視する請求項 3 に記載のマルチスレッド超長命令語プロセッサ。

【請求項 5】

前記機能単位解放機構は、前記少なくとも1つの機能単位がアイドルである時を検出す

る請求項 3 に記載のマルチスレッド超長命令語プロセッサ。

【請求項 6】

マルチスレッド超長命令語プロセッサにおいてマルチスレッド命令ストリームから命令を処理する方法であって、

複数の機能単位を使用して前記命令を実行するステップと、

前記機能単位のうちの少なくとも 1 つによって実行される現在実行されつつある命令が指示に応じて完了した時に前記機能単位のうちの少なくとも 1 つを別のスレッドに割り当て直すステップであって、前記指示が、前記現在実行されつつある命令が完了するであろう時刻を示すものであるステップとを含む方法。

【請求項 7】

前記割り当て直すステップはさらに、

各機能単位が占有されるサイクルの数を監視するステップを含む請求項 6 に記載の方法

【請求項 8】

前記少なくとも 1 つの機能単位に関する状態情報を維持するステップをさらに含む請求項 6 に記載の方法。

【請求項 9】

前記状態情報は、前記少なくとも 1 つの機能単位が多数サイクルの命令によって占有されるサイクルの数を含む請求項 8 に記載の方法。

【請求項 10】

前記割り当て直すステップは、前記少なくとも 1 つの機能単位がアイドルである時を検出する請求項 8 に記載の方法。

【請求項 11】

マルチスレッド超長命令語プロセッサ内で複数のスレッドを有する命令ストリームから命令を処理するための装置であって、

その上に記録されたコンピュータ読み取り可能プログラムを有するコンピュータ読み取り可能媒体を含み、該コンピュータ読み取り可能プログラムは、コンピュータに、

複数の機能単位を使用して前記命令を実行させる手順と、

前記少なくとも 1 つの機能単位によって実行される現在実行されつつある命令が指示に応じて完了した時に前記機能ユニットのうちの少なくとも 1 つを別のスレッドに割り当て直す手順であって、前記指示が、前記現在実行されつつある命令が完了するであろう時刻を示すものである手順とを実行させるようになっている装置。

【請求項 12】

前記コンピュータ読み取り可能プログラムが、さらに、コンピュータに、

前記少なくとも 1 つの機能単位が占有されるであろうサイクルの数を監視する手順を実行させるようになっている請求項 11 に記載の装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は一般にマルチスレッドプロセッサに関し、特に、このようなマルチスレッドプロセッサ内で機能単位を解放するための方法および装置に関する。

【0002】

【従来の技術】

コンピュータアーキテクチャの設計は、さらに早くワークロードを完了しようと試みている。プログラムの平行処理を使用するための、多くのアーキテクチャ設計が提案または提言されてきた。一般に、一度に複数の動作を開始できるアーキテクチャは、一度に 1 つの動作しか開始できないアーキテクチャよりも早くプログラムを実行できる。コンピュータアーキテクチャにおけるもっとも最近の発展は、一度に 1 つ以上の動作を開始し、それによってプログラム動作の速度を上げる方法に向けられてきた。図 1 は、従来のマイクロプロセッサのアーキテクチャ 100 を示す。具体的には、マイクロプロセッサ 100 はプロ

10

20

30

40

50

グラムカウンタ (P C) 1 1 0、レジスタセット 1 2 0、および多くの機能単位 (F U) 1 3 0 - N を含む。冗長機能単位 (F U) 1 3 0 1 ~ 1 3 0 N は、例としてのマイクロプロセッサのアーキテクチャ 1 0 0 に十分なハードウェアリソースを提供して、対応する数の動作を並列的に実行する。

【 0 0 0 3 】

プログラム内で平行処理を使用するアーキテクチャは、一度に複数の機能単位にオペランドを発行して、プログラム実行の速度を上げる。スーパースカラープロセッサ、超長命令語 (V L I W) プロセッサ、およびマルチスレッドプロセッサを含む、並列アーキテクチャを伴う多くのアーキテクチャが提案または提言されており、各々はそれぞれ、図 2、図 4、および図 5 と共に下記に説明される。一般に、スーパースカラープロセッサは実行時にハードウェアを使用して、単一の命令ストリームからの多くの動作が独立しているか否かを動的に決定し、独立している場合は、プロセッサは並列演算論理機構 (A L U) を使用して命令を実行する。ソースオペランドがそれに先行する命令の宛先オペランドに依存していない場合、2 つの命令は独立していると言われる。超長命令語 (V L I W) プロセッサは、依存性の情報に基づいてコンパイル中に命令を評価し、動作を適切にグループ分けし、並列実行する。他方マルチスレッドプロセッサは、単一の命令ストリーム内で並列処理を使用しようと試みるのではなく、複数の命令ストリームを並列に実行する。

10

【 0 0 0 4 】

図 2 に示されたスーパースカラープロセッサのアーキテクチャ 2 0 0 は、各々に有効なデータが提供された場合、独立して動作する多くの機能単位を有する。たとえば図 2 に示されたように、スーパースカラープロセッサ 2 0 0 は演算論理機構 (A L U) 2 3 0 N として具現化された 3 つの機能単位を有し、その各々は同時に結果を計算できる。スーパースカラープロセッサ 2 0 0 は、命令フェッチブロック 2 1 0、命令復号ブロック 2 1 5、および命令順序付けユニット 2 2 0 (発行ブロック) を有するフロントエンド部 2 0 8 を含む。命令フェッチブロック 2 1 0 は単一スレッドの命令ストリームの入力待ち行列 2 0 5 から命令を得る。命令順序付けユニット 2 2 0 は、知られた方法で使用可能な演算論理機構 (A L U) 2 3 0 N の中で同時に実行できる独立した命令を識別する。詳細化ブロック 2 5 0 は命令が完了することを可能にし、バッファ化および再順序付けを提供して、結果をレジスタセット 2 4 0 に書き込んで戻す。

20

【 0 0 0 5 】

図 3 に示されたプログラムフラグメント 3 1 0 の中で、命令 L 2 および L 3 の中にあるソースオペランドのいずれもそれに先行する任意の命令の宛先オペランドに依存していないという点で、場所 L 1、L 2、および L 3 にある命令は独立している。プログラムカウンタ (P C) が場所 L 1 に設定されている時、命令順序付けユニット 2 2 0 は命令ストリームの先を見越して、L 2 および L 3 における命令が独立していることを検出するので、したがってこれら 3 つすべてが 3 つの使用可能な機能単位 2 3 0 N に同時に発行できる。スーパースカラープロセッサのさらに詳細な説明に関しては、たとえば、参照により本明細書に援用された、I E E E (1 9 9 5 年 1 2 月) の議事録の、J a m e s . E . S m i t h および G u r i n d a r . S . S o h i による「The Microarchitecture of Superscalar Processors」を参照されたい。

30

40

【 0 0 0 6 】

先に示したように、図 4 に示された超長命令語 (V L I W) プロセッサ 4 0 0 は、ハードウェアを使用して実行時に動的に並列処理を検出するのではなく、ソフトウェアに頼って単一の命令ストリームからコンパイル時にデータの並列処理を検出する。V L I W コンパイラは図 3 のコードフラグメント 3 1 0 を生成するために使用されたソースコードを与えられ、命令の独立性を検出し、3 つの動作すべてから成る単一の非常に長い命令を構築する。実行時には、プロセッサ 4 0 0 の発行論理はこの大きな命令を 1 つのサイクルで発行し、データをすべての使用可能な機能単位 4 3 0 N に向ける。図 4 に示したように、超長命令語 (V L I W) プロセッサ 4 0 0 は統合フェッチ / 復号ブロック 4 2 0 を含み、統合フェッチ / 復号ブロック 4 2 0 は前もってグループ分けされた命令 4 1 0 をメモリ

50

から得る。超長命令語 (V L I W) プロセッサのさらに詳細な説明に関しては、たとえば、参照により本明細書に援用されている、SPIE Real Time Signal Processing I V (1 9 8 1) の 2 4 1 ~ 2 4 8 ページにある、B u r t o n J . S m i t h による「Architecture and Applications of the HEP Multiprocessor Computer System」を参照されたい。

【 0 0 0 7 】

V L I W プロセッサの一変形例は、たとえば、I E E E T r a n s a c t i o n s o n C o m p u t e r s (1 9 9 8 年 8 月) の R o b e r t P . C o l w e l l らによる「A VLIW Architecture for a Trace Scheduling Compiler」で論じられているようなマルチフローアーキテクチャによって代表され、固定長命令を使用して、そこにおいて 10
あらかじめ定義されたフィールドがデータを一度にすべての機能ユニット 4 3 0 N に向ける。大きな命令の中で指定されたすべての動作が完了すると、プロセッサは新しい、多数の動作の命令を発行する。テキサス州ダラスの T e x a s I n s t r u m e n t s から市販されている C 6 x プロセッサや、カリフォルニア州サンタクララの I n t e l C o r p から市販されている E P I C I A - 6 4 などのいくつかのより最近の V L I W プロセッサは、その代わりに可変長命令パケットを使用し、可変長命令パケットは互いにバンドルされている 1 つまたは複数の動作を含む。

【 0 0 0 8 】

図 5 に示されたマルチスレッドプロセッサ 5 0 0 は単一の命令ストリーム内で平行処理を使用しようと試みるのではなく、複数の命令ストリームを並列に実行することによってパ 20
フォーマンスの向上を得る。図 5 に示されたマルチスレッドプロセッサ 5 0 0 は、プログラムカウンタ 5 1 0 N、レジスタセット 5 2 0 N、および機能単位 5 3 0 N を含み、各々は対応する命令ストリーム N に専用である。マルチスレッドプロセッサ 5 0 0 の代替の実装は、いくつかのレジスタセット 5 2 0 N およびプログラムカウンタ 5 1 0 N を伴う単一の機能単位 5 3 0 を使用してきた。このような代替のマルチスレッドプロセッサ 5 0 0 は、プロセッサ 5 0 0 が、1 つまたは 2 つのサイクル内で命令発行を 1 つのプログラムカウンタ / レジスタセット 5 1 0 N / 5 2 0 - N から、別のプログラムカウンタ / レジスタセット 5 1 0 - N / 5 2 0 - N に切り換えることができるように設計されている。したがって、L O A D 命令などの待ち時間が長い命令は、別の命令ストリームからの 30
より短い動作と重ねることができる。ワシントン州シアトルの T e r a C o m p u t e r C o m p a n y から市販されている T E R A M T A アーキテクチャはこのタイプの 1 つの例である。

【 0 0 0 9 】

同時マルチスレッド化と呼ばれる、マルチスレッドアーキテクチャ 5 0 0 の拡張は、図 2 と共に上記に説明されたスーパースカラーアーキテクチャと、図 5 と共に上記に説明されたマルチスレッド設計との両方を組み合わせている。同時マルチスレッド化技法の詳細な説明に関しては、たとえば、参照により本明細書に援用されている、第 2 2 回 A n n u a l I n t ' l S y m p o s i u m o n C o m p u t e r A r c h i t e c t u r e (1 9 9 5 年 6 月、イタリア、S a n t a M a r g h e r i t a L i g u r e) の議事録、3 9 2 ~ 4 0 3 ページにある、D e a n T u l l s e n らによる「Simult 40
aneous Multithreading: Maximizing On-Chip Parallelism」を参照されたい。一般に、同時マルチスレッド化アーキテクチャでは機能単位のプールがあり、そのうち任意の数が、多くのプログラムカウンタ / レジスタセット構造のうち任意の 1 つから発行できる命令に動的に割り当てられる。機能単位を多くのプログラムスレッドの間で共有することにより、同時マルチスレッド化アーキテクチャは図 5 に示されているより効率的にハードウェアを使用することができる。

【 0 0 1 0 】

【 発明が解決しようとする課題 】

同時マルチスレッド化アーキテクチャの組み合わせられた手法がスーパースカラーアーキテクチャまたはマルチスレッドアーキテクチャの個別の手法よりも改善された効率を提供す 50

る一方、同時マルチスレッド化アーキテクチャは依然として、動的に命令ストリームを検討して潜在的な並列処理を検出するために、精巧な発行論理を必要とする。したがって、2つの命令ストリームが独立しているか否かを動的に決定する必要のない、マルチスレッドプロセッサのアーキテクチャに対するニーズが存在する。さらに、同時マルチスレッド化を提供するマルチスレッドのアーキテクチャに対するニーズが存在する。

【0011】**【課題を解決するための手段】**

一般に、マルチスレッド超長命令語（VLIW）プロセッサにおいて、多数サイクルの命令によって失われた容量を検索できる、機能単位を解放するための方法および装置が開示される。本発明は従来のVLIWアーキテクチャおよび従来のマルチスレッドアーキテクチャの技法を組み合わせている。本発明の組み合わせられたアーキテクチャは、個別のプログラム内、およびワークロード全体で実行時間を軽減する。従来のマルチスレッドVLIWアーキテクチャでは、多数の命令パケット内の1つの多数サイクルの命令は、パケット内の他の命令が単一のサイクルのみをとる場合でも多数サイクルの命令の長さによってすべての割り当てられた機能単位を占有する。本発明はアイドルの機能単位を他のスレッドに割り当て直すことのできる機能単位の解放を提供し、これによってワークロードの効率を向上させる。

10

【0012】

本発明は、従来の機能単位を命令パケットに割り当てる手法ではなく、命令パケットを機能単位に割り当て、機能単位は発行論理と独立してその状態を維持することができる。本発明のマルチスレッドVLIWアーキテクチャでは、各機能単位は関連づけられた状態マシン（SM）を有し、状態マシンは機能単位が多数サイクルの命令によって占有されるサイクルの数を追跡する。したがって、機能単位がビジーである限り、機能単位はそれ自体を割り当て直さない。命令が完了すると、同じスレッドに割り当てられた別の機能単位が依然としてビジーであっても機能単位は機能単位の割り当てに関与できる。

20

【0013】

このようにして、本発明の機能単位解放手法を使用すると、多数サイクルの命令に関連づけられていない機能単位を、ブロックされたスレッドが待っている間に別のスレッドに割り当てることができ、これによってマルチスレッドVLIWプロセッサのスループットが改善する。状態は、命令発行単位とは別に各機能単位に関連づけられているので、機能単位は任意のスレッドおよびその構成要素である命令の状態からは独立してスレッドに割り当てられることができる。

30

【0014】

本発明は、コンパイラを使用してマルチスレッドプロセッサのアーキテクチャ内で並列処理を検出する。したがって動的な決定が必要ないため、単一スレッドVLIWプロセッサと同じ方法で単一のプログラムシーケンサから多数の命令を発行することによって、また、同時マルチスレッド化と同じように多数のプログラムシーケンサをサポートするが発行論理内で複雑さが軽減されてサポートすることによって、プログラムの並列処理を使用するマルチスレッドVLIWアーキテクチャが開示される。本発明は命令を機能単位に割り当て、多数のVLIW命令を同じサイクル内の多数の機能単位に発行する。本発明の割り当て機構は、引数が機能単位にディスパッチされる直前に、パイプライン段を占有する。一般に、割り当て段は、適切な命令を選択しその命令を機能単位に割り当てることによって、命令をどのようにグループ分けして効率を最大にするかを決定する。

40

【0015】

本発明のさらに完全な理解、また本発明のさらなる特徴および利点は、次の詳細な説明および図面を参照することによって得られる。

【0016】**【発明の実施の形態】**

（関連出願への相互参照）

本発明は、代理人整理番号（B e r e n b a u m 7 - 2 - 3 - 3）の、「Method and A

50

pparatus for Allocating Functional Units in a Multithreaded Very Large Instruction Word (VLIW) Processor」という名称の米国特許出願と、代理人整理番号 (B e r e n b a u m 9 - 4 - 5 - 5) の、「Method and Apparatus for Splitting Packets in a Multithreaded Very Large Instruction Word (VLIW) Processor」という名称の米国特許出願と、代理人整理番号 (B e r e n b a u m 10 - 5 - 6 - 6) で、「Method and Apparatus for Identifying Splittable Packets in a Multithreaded Very Large Instruction Word (VLIW) Processor」という名称の米国特許出願に関連し、各々は本明細書と同時に提出され、本発明の譲渡人に譲渡され、参照により本発明に援用されている。

【 0 0 1 7 】

図 6 は、本発明によるマルチスレッド V L I W プロセッサ 6 0 0 を示す図である。図 6 に示したように、3つの命令スレッド、すなわちスレッド A (T A)、スレッド B (T B) およびスレッド C (T C) があり、各々は命令番号 n で動作している。さらに、例としてのマルチスレッド V L I W プロセッサ 6 0 0 は、9つの機能単位 6 2 0 - 1 ~ 6 2 0 - 9 を含み、これは任意のスレッド T A T C に独立的に割り当てることができる。例としての3つのスレッド T A T C 上の命令の数は9で、例としての使用可能な機能ユニット 6 2 0 の数もまた9なので、3つのスレッド T A T C すべてからの命令の各々は1つのサイクルで命令パケットを発行でき、次のサイクルで命令 n + 1 に移動できる。

10

【 0 0 1 8 】

一般的に、命令とそれによって指定される動作の間には1対1の対応があることに留意されたい。したがって、この用語は本明細書内では相互に交換可能であるように使用される。さらに、命令が多数の動作を指定する状況では、マルチスレッド V L I W プロセッサ 6 0 0 は1つまたは複数の多数動作の機能単位 6 2 0 を含んで、多数動作を指定する命令を実行すると仮定されていることに留意されたい。多数動作を指定する命令が処理できるアーキテクチャの例は、複雑命令セットコンピュータ (C I S C) である。

20

【 0 0 1 9 】

従来の単一スレッド V L I W アーキテクチャでは、1つの命令パケット内のすべての動作は同時に発行される。常に、パケットを発行するのに使用可能な十分な機能単位がある。動作が多数のサイクルをとる時、動作の他のソースが使用可能ではないので、命令発行論理はストールする可能性がある。たとえば、キャッシュミスによって遅れた多数サイクルのメモリアクセス命令の間、命令発行論理は不定の期間ブロックされ、この時間はコンパイル時には決定できない。この待ち時間の間、命令はコンパイラによってスケジューリングできないので、命令は発行できない。他方、本発明によるマルチスレッド V L I W プロセッサでは、これらの制限はあてはまらない。多数サイクルの動作のために命令パケットがストールする場合には、別のスレッドの最初に、使用可能な別の動作がある。

30

【 0 0 2 0 】

図 7 は、従来のマルチスレッド実装 (本発明の利点がない) に関して、図 6 と共に上記に説明された3スレッド T A T C の命令 n + 1 における次のサイクルを示す。図 6 および図 7 に示されたように、図 6 および図 7 のスレッド A の M U L 動作が2サイクルをとり、スレッド A の別の3つの動作が1サイクルをとる場合、スレッド A に割り当てられた4つの機能単位すべてが2サイクルの間ビジーになり、別のスレッド T B T C に割り当てられることができない。図 7 は可能性のある結果を示す。スレッド A 内の命令 n における命令パケットは、図 6 に表されたサイクルおよび図 7 の次のサイクルの両方について4つの機能単位 7 2 0 を必要とする。スレッド B 内の場所 n + 1 からの命令パケットは2つの機能単位を必要とし、残りの機能単位 7 2 0 - 2 および 7 2 0 - 8 の2つに割り当てられる。しかし、スレッド C 内の場所 n + 1 にある命令パケットは4つの機能単位を必要とするが、使用可能な機能単位は3つしかない。したがってスレッド C はストールし、その結果、3つの機能単位は使用されない。

40

【 0 0 2 1 】

本発明は、機能単位を解放し多数サイクルの命令によって失われた容量を検索できる方法および装置を提供する。機能単位を命令パケットに割り当てる代わりに、命令パケットが

50

機能単位に割り当てられ、機能単位は発行論理から独立して状態を維持できる。図8に示されたように、各機能単位820-Nは関連づけられた状態マシン(SM)830-Nを有し、状態マシンは図9と共に次にさらに説明され、多数サイクルの動作によって占有された機能単位820-Nのサイクルの数を追跡する。このように、機能単位820-Nは機能単位820-Nがビジーである限り、それ自体を割り当てなおさない。動作が完了すると、機能単位820-Nは、同じスレッドに割り当てられた他の機能単位820が依然としてビジーであっても、機能単位の割り当てに関与できる。

【0022】

このように、本発明の機能単位解放手法を実装することにより、ブロックされたスレッドが待っている間に多数サイクルの命令に関連づけられていない機能単位を他のスレッドに割り当てることができ、これによって、マルチスレッドVLIWプロセッサ500のスループットを向上させることができる。状態は命令発行ユニットとは別に各機能単位に関連づけられているので、機能単位は任意の1つのスレッドおよびその構成単位の命令の状態とは独立してスレッドに割り当てることができる。

10

【0023】

図8は、本発明による、図6と共に上記に説明された3スレッドTA-TCの、命令n+1における次のサイクルを示す。図6と同じように、MUL動作は2つのサイクルをとり、スレッドCに関して場所n+1における命令パケットは4つの機能単位を必要とする。第1のサイクルの後、スレッドAに割り当てられた4つの機能単位(図6の機能単位620-1、620-3、620-4、および620-5)のうちの3つは解放されるので、サイクルn+1に関してはスレッドBおよびスレッドCへの割り当てのために使用可能な機能単位は8つある。スレッドTBおよびTCは6つの機能単位820しか必要としないので、スレッドTBもTCもストールすることはなく、図7の構成と比較してサイクルは節約される。

20

【0024】

図9は図8の状態マシン830-Nの実装を示す。図9に示されたように、状態マシン830-Nは継続的に多数サイクルの動作の実行を監視し、機能単位820-Nが占有されているサイクルの数を追跡する。状態マシン830-Nが、動作が完了したと決定すると、状態マシン830-Nは機能単位を解放し、別のスレッドによって再使用する。1つの実装では、状態マシン830-Nは、各動作に関して指定された最大の実行時間にしたがって動作が完了したことを決定する。

30

【0025】

本明細書に示され、説明された実施形態およびその変形例は、本発明の原理の例にすぎず、当業者によって本発明の範囲および精神から離れることなく種々の改変例が実装できることを理解されたい。

【図面の簡単な説明】

【図1】従来的一般化されたマイクロプロセッサのアーキテクチャを示す図である。

【図2】従来のスーパースカラープロセッサのアーキテクチャの概略構成図である。

【図3】動作の独立性を示すプログラムフラグメントを示す図である。

【図4】従来超長命令語(VLIW)プロセッサのアーキテクチャの概略構成図である。

40

【図5】従来マルチスレッドプロセッサの概略構成図である。

【図6】本発明によるマルチスレッドVLIWプロセッサの図である。

【図7】従来マルチスレッド実装に関する、図6に示された3つのスレッドTA-TCの命令n+1における次のサイクルを示す図である。

【図8】本発明によるマルチスレッド実装に関する、図6に示された3つのスレッドTA-TCの命令n+1における次のサイクルを示す図である。

【図9】図8に示された状態マシンの実装を示す図である。

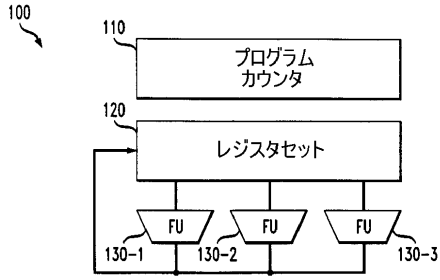
【符号の説明】

100 マイクロプロセッサ

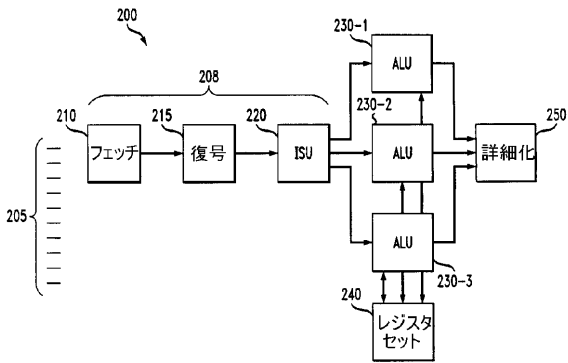
50

1 1 0	プログラムカウンタ	
1 2 0	レジスタセット	
1 3 0	- N 機能単位	
2 0 0	スーパースカラープロセッサ	
2 0 5	入力待ち行列	
2 0 8	フロントエンド部	
2 1 0	命令フェッチブロック	
2 1 5	命令復号ブロック	
2 2 0	命令順序づけユニット	
2 3 0	N 演算論理機構	10
2 4 0	レジスタセット	
2 5 0	詳細化ブロック	
3 1 0	プログラムフラグメント	
4 0 0	超長命令語 (V L I W) プロセッサ	
4 1 0	命令	
4 2 0	統合フェッチ / 復号ブロック	
4 3 0	- N 機能単位	
5 0 0	マルチスレッドプロセッサ	
5 1 0	- N プログラムカウンタ	
5 2 0	- N レジスタ	20
5 3 0	- N 機能単位	
6 0 0	V L I W プロセッサ	
6 2 0	- N 機能単位	
7 2 0	機能単位	
8 2 0	- N 機能単位	
8 3 0	- N 状態マシン	

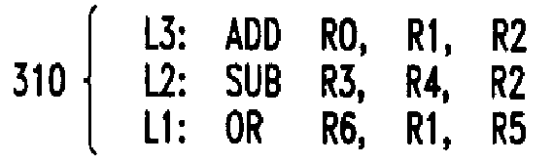
【図 1】



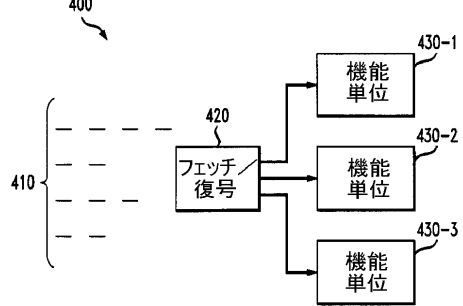
【図 2】



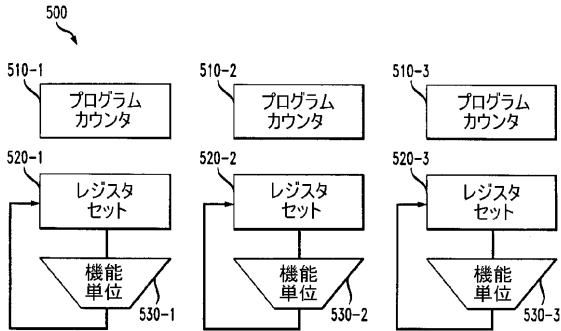
【図 3】



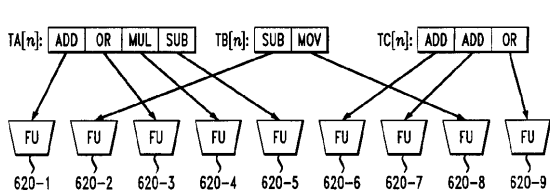
【図 4】



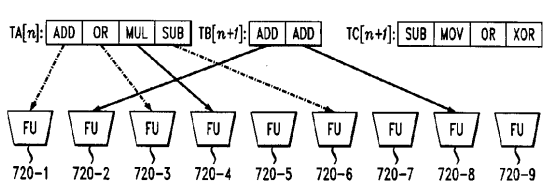
【図 5】



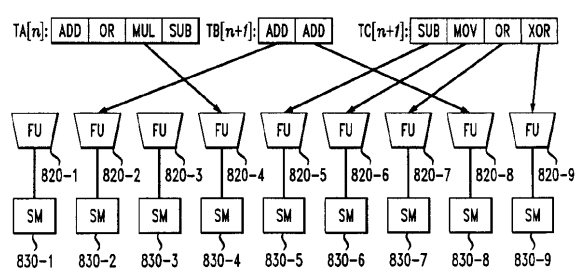
【図 6】



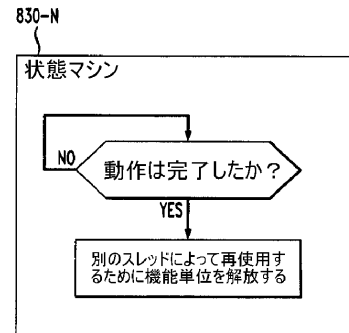
【図 7】



【図 8】



【図 9】



フロントページの続き

- (74)代理人 100096688
弁理士 本宮 照久
- (74)代理人 100104352
弁理士 朝日 伸光
- (72)発明者 アラン ディヴィッド ベレンバウム
アメリカ合衆国 10011 ニューヨーク, ニューヨーク, ウエスト ツエルヴス ストリート
37, アpartment 9ジェー
- (72)発明者 ネヴィン ヘインツ
アメリカ合衆国 07960 ニュージャージー, モリスタウン, マウント ケンブル アヴェニ
ュー 301
- (72)発明者 トア イー. ジェレミアッセン
アメリカ合衆国 08873 ニュージャージー, サマーセット, オスウエストリー ウエイ 8
1
- (72)発明者 ステファノス カックシラス
アメリカ合衆国 ニュージャージー, ジャーシィ シティ, ワシントン ブウルヴァード 444
, アpartment 2119

審査官 後藤 彰

- (56)参考文献 特開平7-281896(JP, A)
特開平3-154125(JP, A)
特開平5-197547(JP, A)

- (58)調査した分野(Int.Cl., DB名)
G06F 9/38
G06F 9/46