(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第5059609号
(P5059609)

(45)発行日　平成24年10月24日(2012.10.24)　(24)登録日　平成24年8月10日(2012.8.10)

(51)Int.Cl.　　F I
G06F 12/08 (2006.01)　　G06F 12/08 505C
G06F 9/38 (2006.01)　　G06F 9/38 330B
　　G06F 9/38 330G

請求項の数 13　(全 55 頁)

(21)出願番号　特願2007-527950 (P2007-527950)
(86)(22)出願日　平成17年8月16日 (2005.8.16)
(65)公表番号　特表2008-510258 (P2008-510258A)
(43)公表日　平成20年4月3日 (2008.4.3)
(86)国際出願番号　PCT/US2005/029135
(87)国際公開番号　WO2006/038991
(87)国際公開日　平成18年4月13日 (2006.4.13)
審査請求日　平成20年7月31日 (2008.7.31)
(31)優先権主張番号　10/920,610
(32)優先日　平成16年8月17日 (2004.8.17)
(33)優先権主張国　米国 (US)
(31)優先権主張番号　10/920,682
(32)優先日　平成16年8月17日 (2004.8.17)
(33)優先権主張国　米国 (US)

(73)特許権者　501261300
エヌヴィディア　コーポレイション
アメリカ合衆国, カリフォルニア　９５０５０, サンタ　クララ, サン　トーマス　エクスプレスウェイ　２７０１
(74)代理人　100094318
弁理士　山田　行一
(74)代理人　100123995
弁理士　野田　雅一
(72)発明者　ハクラ,　ジヤド,　エス.
アメリカ合衆国,　カリフォルニア州,　マウンテン　ヴュー,　ナンバー４０１,　ウェスト　エル　キャミノ　リアル　２４００

最終頁に続く

(54)【発明の名称】メモリへの様々なタイプのアクセスを予測するため、およびキャッシュメモリに関連付けられた予測を管理するための、システム、装置、および方法

(57)【特許請求の範囲】
【請求項１】
メモリへのアクセスを予測するためのプリフェッチャであって、
ターゲットアドレスのグループをトリガアドレスに関連付けるように構成され、および
前記ターゲットアドレスのグループのうちの少なくとも１つのターゲットアドレスに基づき、予測アドレスのグループを発行するように構成された、
第１のアドレス予測器を備え、
前記ターゲットアドレスのグループのうちの少なくとも１つの前記ターゲットアドレスは、前記トリガアドレスに対して非順次であり、
前記第１のアドレス予測器は、前記トリガアドレスが検出された場合に、非順次予測として予測アドレスのグループを発行する非順次予測器を更に備え、
前記非順次予測器は、
前記ターゲットアドレスのグループと前記トリガアドレスとの関連を格納すると共に、前記ターゲットアドレスのグループ間の相対的優先順位を格納するためのリポジトリと,

要求されたアドレスのストリーム内の前記トリガアドレスを検出するように構成され、さらに、検出された前記トリガアドレスとの関連と、最高優先順位であるその優先順位とに基づき、前記少なくとも１つのターゲットアドレスを、非順次予測として選択するよ

うに構成された、非順次予測エンジンと、をさらに備え、
当該プリフェッチャは、
前記トリガアドレスが、要求されたアドレスの順次ストリーム内にある場合、および前記非順次予測が、前記トリガアドレスとして、前記順次ストリーム内の次のアドレスの指定と比較して早く生成される場合、前記順次ストリームの第１のアドレスを前記トリガアドレスとして指定するための促進器と、
少なくとも１つの予測アドレスの生成を抑制するように構成された抑制器と、
をさらに備える、プリフェッチャ。
【請求項２】
前記予測アドレスのグループは、順次アドレスのグループの基本アドレスとしての前記少なくとも１つのターゲットアドレスと共に順次アドレスのグループを含む、請求項１に記載のプリフェッチャ。
【請求項３】
前記最高優先順位が、プロセッサが前記少なくとも１つのターゲットアドレスを、前記ターゲットアドレスのグループのうちの他のアドレスに対して最も新しく要求したことを少なくとも示す、請求項１に記載のプリフェッチャ。
【請求項４】
前記第１のアドレス予測器がトリガアドレスからインデックスおよびタグを生成するように構成され、前記リポジトリがトリガアドレスとターゲットアドレスとの関連を格納するためのメモリロケーションをそれぞれ有するいくつかのｗａｙを含み、第１のｗａｙに格納されたトリガアドレスとターゲットアドレスとの関連が、第２のｗａｙに格納された他のトリガアドレスとターゲットアドレスとの関連よりも高位の優先順位に関連付けられる、請求項１に記載のプリフェッチャ。
【請求項５】
第２のアドレスとマッチするターゲットアドレスを含む前記トリガアドレスとターゲットアドレスとの関連のうちの１つに関する優先順位を修正するように構成された優先順位調整器をさらに備える、請求項４に記載のプリフェッチャ。
【請求項６】
前記抑制器が、前記トリガアドレスが、データに関する要求またはプリフェッチ要求のいずれか、あるいはその両方に関する場合、前記ターゲットアドレスのグループに関するアドレスの数を削減するように構成され、それによって前記少なくとも１つの予測アドレスの生成が抑制される、請求項１に記載のプリフェッチャ。
【請求項７】
さらに前記抑制器が、
前記トリガアドレスの検出から、前記非順次予測としての前記予測アドレスのグループの生成までの時間間隔が、予め設定された、時間に関するしきい値未満である場合、非順次予測としての前記予測アドレスのグループの生成を抑制するように構成される、
請求項１に記載のプリフェッチャ。
【請求項８】
さらに前記抑制器が、
複数のインタリーブされた、要求されたアドレスの順次ストリームの検出に基づいて、少なくとも前記予測アドレスの生成を抑制するように構成される、請求項１に記載のプリフェッチャ。
【請求項９】
前記複数のインタリーブされた順次ストリームのそれぞれが、スレッドに関連付けられている、請求項８に記載のプリフェッチャ。
【請求項１０】
少なくとも１つの他のトリガアドレスに基づいて、順次予測を生成するための順次予測器を含む、第２のアドレス予測器をさらに備える、請求項１に記載のプリフェッチャ。
【請求項１１】

前記順次予測が、
前記少なくとも１つの他のトリガアドレスから昇順に並べられた第１の数のアドレス、または、
前記少なくとも１つの他のトリガアドレスから降順に並べられた第２の数のアドレス、のいずれか、あるいは、
前記第１および第２の数のアドレスの両方
を含み、
さらに前記抑制器が、
前記少なくとも１つの他のアドレスが昇順の第１のアドレスストリームの一部であることを検出し、前記降順に並べられた前記第２の数のアドレスに基づく、前記数の追加の予測アドレスを抑制するように、および
前記少なくとも１つの他のアドレスが降順の第２のアドレスストリームの一部であることを検出し、前記昇順に並べられた前記第１の数のアドレスに基づく、前記数の追加の予測アドレスを抑制するように、
構成された、請求項１０に記載のプリフェッチャ。
【請求項１２】
前記順次予測が、
前記少なくとも１つの他のトリガアドレスから１つだけ降順のバックアドレス、または
前記少なくとも１つの他のトリガアドレスのバックセクタアドレス、のいずれか、あるいはその両方、を含み、
さらに前記抑制器が、
前記順次予測が前記バックアドレス又は前記バックセクタアドレスのいずれかを含む場合、前記順次予測の数を減少させるように構成された、請求項１０に記載のプリフェッチャ。
【請求項１３】
予測を維持するようにそれぞれ構成された、複数のキューを備える、予測インベントリと、
フィルタリング済みのアドレスのサブセットを生成するためのインベントリフィルタであって、
前記予測インベントリ、または
前記予測アドレスのグループおよび前記順次予測の、いずれかにおいて、
冗長アドレスを除去するように構成された、前記インベントリフィルタと、
をさらに備え、
前記プリフェッチャが、前記フィルタリング済みアドレスのサブセットのうちの少なくとも１つを提供するように構成された、請求項１０に記載のプリフェッチャ。
【発明の詳細な説明】
【発明の簡単な説明】
【０００１】
本発明は、一般にコンピューティングシステムに関し、より具体的には、たとえば、構成可能な量の予測を生成すること、ならびに、たとえば予測インベントリおよび／またはマルチレベルキャッシュに対して、予測を抑制およびフィルタリングすることによる、メモリへの順次および非順次アクセスの予測に関する。
【発明の背景】
【０００２】
プログラム命令およびプログラムデータをフェッチするためにプリフェッチャが使用されるため、プロセッサは取り出された情報それ自体を必要に応じて容易に利用することができる。プリフェッチャは、将来プロセッサがどの命令およびデータを使用するかを予測するため、プロセッサは、典型的にはプロセッサよりも低速で動作するシステムメモリから、命令またはデータにアクセスするのを待機する必要がない。プロセッサとシステムメモリとの間にプリフェッチャが実装されると、プロセッサがメモリからの要求データを待

ちながらアイドル状態を続ける可能性が低くなる。したがって、プリフェッチャは一般に、プロセッサの性能を向上させる。

【０００３】

一般に、プリフェッチャによってより多くの予測が生成されるほど、プロセッサが利用できる必要な命令およびデータを有するようにプリフェッチャが調整できる可能性が高くなり、それによってプロセッサの待ち時間が減少する。しかし、従来のプリフェッチャは、典型的には予測プロセスを十分に管理することができない。このように管理されないと、これらのプリフェッチャは、予測されたアドレスの量がプリフェッチャの処理能力を超えた場合にメモリリソースに負荷をかけすぎる傾向がある。このようなリソースの過負荷を防止するために、従来のプリフェッチャは、プリフェッチャまたはメモリリソースのいずれかを過負荷にする可能性のある量の予測を生成しないように、予測の生成を控えめにする傾向がある。加えて、従来のプリフェッチャは、通常、こうした予測プロセスを実施するコストを考慮せずに予測を生成するため、予測プロセスおよびこれをサポートするために必要なリソースの量を合理化することの特典が実現できない。特に、従来タイプのプリフェッチャは、主に、本来は逐次的である予測を生成するための標準技術に依拠しており、計算に拠るまたは拠らないにかかわらず、リソースを節約するように予測を格納しない。また、従来のプリフェッチャは、通常、予測プロセスを十分に管理していないため、予測アドレスの量がプリフェッチャの処理能力を超える場合、計算およびメモリリソースに負荷をかけすぎる傾向がある。そこで、リソースの過負荷を防止するために、これらのプリフェッチャは、プリフェッチャを過負荷にする可能性のある量の予測を生成しないように、予測の生成を控えめにする傾向がある。さらに、多くの従来のプリフェッチャは、予測が生成された後、およびプロセッサがそれらの予測を要求する前に、予測を管理するための機能が欠如している。通常、これらのプリフェッチャはプリフェッチデータを単一のキャッシュメモリに格納するが、このキャッシュメモリは、典型的にはすでにキャッシュに格納された予測に対して過剰な予測を制限するための機能が欠如している。従来のプリフェッチャで使用されるキャッシュメモリは、単にデータを格納するためのものであり、そこに格納された予測されたアドレスを効果的に管理するように十分な設計がなされていない。

【０００４】

前述の内容に鑑み、メモリへのアクセスを効果的に予測するためのシステム、装置、および方法を提供することが望ましい。理論的に言えば、例示的なシステム、装置、または方法は、少なくとも前述の欠点を最小にするかまたは除去することになる。

【発明の概要】

【０００５】

メモリへのアクセスを予測するためのシステム、装置、および方法が開示される。一実施形態では、例示的装置は、プログラム命令を実行しプログラムデータを処理するように構成されたプロセッサと、プログラム命令およびプログラムデータを含むメモリと、メモリプロセッサとを備える。メモリプロセッサは、プログラム命令またはプログラムデータを含むアドレスを受け取るように構成された、スペキュレータを含むことができる。こうしたスペキュレータは、構成可能な数の順次アドレスを生成するための順次予測器を備えることができる。スペキュレータは、アドレスのサブセットをアドレスに関連付けるように構成された、非順次予測器を含むこともできる。非順次予測器は、サブセットのうちの少なくとも１つのアドレスに基づいて、アドレスのグループを予測するように構成することも可能であり、サブセットのうちの少なくとも１つのアドレスは、アドレスに対してパターン化できない。一実施形態では、例示的な非順次予測器が、メモリへのアクセスを予想する。非順次予測器は、アドレスからインデックスおよびタグを生成するように構成された、予測生成器を含む。また、非順次予測器は、予測生成器に結合されたターゲットキャッシュも含む。ターゲットキャッシュは、それぞれがトリガーターゲット関連を格納するためのメモリロケーションを有する、メモリのいくつかの部分を含む。メモリの第１の部分に格納されたトリガーターゲット関連は、メモリの第２の部分に格納された他のトリ

ガーターゲット関連よりも高い優先順位に関連付けられる。

【０００６】

本発明の一実施形態では、この装置は、それぞれが項目のグループを維持するように構成されたキューを含む、予測インベントリを含む。項目のグループは、典型的には、項目のグループに対応するトリガアドレスを含む。グループの各項目は、予測の１つのタイプである。またこの装置は、予測の数を、予測の数と同じ予測タイプを有するキューのうちの少なくとも１つと比較するように構成された、インベントリフィルタも含む。いくつかのケースでは、インベントリフィルタは、予測の数を、異なる予測タイプを有するキューのうちの他の少なくとも１つと比較するように構成される。たとえば、いくつかの前方順次予測は、バックキュー、または同様のものに対してフィルタリング可能である。少なくとも１つの実施形態では、装置が、メモリへの予測アクセスを管理するための戻りデータキャッシュメモリを含む。戻りデータキャッシュメモリは、たとえばしきい値より短い経過時間を有する予測を格納するように構成された短期キャッシュメモリと、たとえばしきい値より長いかまたは等しい経過時間を有する予測を格納するように構成された長期キャッシュメモリとを、含むことができる。長期キャッシュメモリは、通常、短期キャッシュよりも多くのメモリ容量を有する。プリフェッチャは、複数の予測が短期キャッシュメモリまたは長期キャッシュメモリ、あるいはその両方の、いずれに格納されるかにかかわらず、１サイクルの操作中、または２サイクルにわたってなど、並行して検出するように構成されたインターフェースを含むことも可能であり、インターフェースは、短期キャッシュメモリおよび長期キャッシュメモリを検査する場合、複数の予測それぞれのうちの少なくとも２つの表現を使用する。

【０００７】

本発明は、添付の図面と共に説明される以下の詳細な記述に関連して、より完全に理解されよう。

【０００８】

同じ参照番号は、図面のいくつかの図全体を通じて対応する部分を表す。

【例示的実施形態の詳細な説明】

【０００９】

本発明は、プロセッサが必要とすると予測される可能性のあるプログラム命令およびプログラムデータを取り出すために、メモリへのアクセスを効果的に予測するためのシステム、装置、および方法を提供する。メモリへのアクセスを効果的に予測することにより、１つまたは複数のプロセッサに必要なデータを提供する待ち時間を最小限にすることができる。本発明の特定の実施形態によれば、装置は、メモリアクセスを予測するように構成されたスペキュレータを含む。例示的スペキュレータは、予測生成レートを変化させるために構成可能な量の予測を生成するように構成することができる。他の実施形態では、スペキュレータは、そうでなければプリフェッチャが管理しなければならなくなる可能性のある、冗長予測などの不必要な予測の量を制限するために、一定の予測の生成を抑制することができる。特定の実施形態では、スペキュレータは、予測を含むキャッシュメモリまたはインベントリが、プロセッサに提示するためのより好適な予測を含むかどうかを探査することによって、不必要な予測をフィルタリングすることもできる。一実施形態では、キャッシュメモリは、短期キャッシュおよび長期キャッシュ内に予測を格納し、その両方が冗長予測をフィルタリング除去するために並行して検査される。

【００１０】

順次および非順次予測を生成するためのプリフェッチャおよびスペキュレータに関する例示的実施形態

図１は、本発明の特定の実施形態に従った、例示的スペキュレータを示すブロック図である。この例でスペキュレータ１０８は、プリフェッチャ１０６内に常駐するように示される。さらにプリフェッチャ１０６は、１つまたは複数のプロセッサによるメモリアクセスを少なくとも制御するように設計された、メモリプロセッサ１０４内に常駐するように示される。プリフェッチャ１０６は、メモリ１１２からのプログラム命令およびプログラ

ムデータの両方を、要求される前に「フェッチ」し、その後フェッチされたプログラム命令およびプログラムデータを、プロセッサ１０２による要求に応じてプロセッサ１０２に提供するように動作する。使用に先立ってフェッチすること（すなわち「プリフェッチすること」）により、プロセッサのアイドル時間（たとえば、プロセッサ１０２がデータ不足である間の時間）が最小化される。プリフェッチャ１０６は、プリフェッチされたデータのプロセッサ１０２への提示を格納および管理するための、キャッシュメモリ１１０も含む。キャッシュメモリ１１０は、命令の実行およびデータ取り出しをスピードアップするためのデータストアとして働く。特に、キャッシュメモリ１１０はプリフェッチャ１０６内に常駐し、一般に、メモリコントローラ１０４とは別に何らかの待ち時間を減少させるために採用される、「Ｌ１」および「Ｌ２」キャッシュなどの他のメモリキャッシュを補足するように動作する。

【００１１】

動作時に、スペキュレータ１０８は、メモリ１１２にアクセスするためのプロセッサ１０２による要求（「読み取り要求」）について、システムバス１０３を監視する。特に、プロセッサ１０２がプログラム命令を実行する場合、スペキュレータ１０８は、プロセッサ１０２によってまだ使用されていないプログラム命令およびプログラムデータを含むアドレスに関する読み取り要求を検出する。考察のために、「アドレス」は、一般にメモリ１１２とキャッシュメモリ１１０との間で転送されるメモリのキャッシュラインまたは単位に関連付けられる。キャッシュラインの「アドレス」はメモリロケーションを表すことが可能であり、キャッシュラインはメモリ１１２の複数のアドレスからのデータを含むことができる。「データ」という用語は、プリフェッチ可能な情報の単位を表すのに対して、「プログラム命令」および「プログラムデータ」という用語は、それぞれ、プロセッサ１０２によってその処理中に使用される命令およびデータを表す。したがって、データ（たとえば任意のビット数）は、プログラム命令および／またはプログラムデータを構成する予測情報を表すことができる。また「予測」という用語は、「予測アドレス」という用語と同じ意味で使用することもできる。予測アドレスがメモリ１１２へのアクセスに使用される場合、典型的には、その予測アドレスならびに他の（予測されるかまたはされない）アドレスを含む１つまたは複数のキャッシュラインがフェッチされる。

【００１２】

検出された読み取り要求に基づいて、スペキュレータ１０８は、プロセッサ１０２によって次に要求される可能性のある、構成可能な数の予測アドレスを生成することができる。スペキュレータ１０８は、本発明の少なくとも１つの実施形態に従って、１つまたは複数の推測技法を使用することによってこれを実行する。スペキュレータ１０８は、これらの推測技法を予測器として実施するが、その実施については以下で説明する。さらにスペキュレータ１０８は、いくつかの予測の生成を抑制し、他の予測をフィルタリングする。一定の予測を抑制またはフィルタリングすることにより、あるいはそれらの両方を実行することにより、冗長予測の数が減少し、それによってリソースが保存される。保存されるリソースの例には、キャッシュメモリ１１０などのメモリリソース、およびメモリバス１１１などのバスリソース（たとえば帯域幅に関して）が含まれる。

【００１３】

スペキュレータ１０８の予測がオプションでフィルタリングされた後、メモリプロセッサ１０４は残った（すなわちフィルタリング除去されなかった）予測を、メモリバス１１１を介してメモリ１１２へ移送する。これに応答して、メモリ１１２はプリフェッチされたデータを予測アドレスと共に戻す。キャッシュメモリ１１０は、戻されたデータを、メモリプロセッサ１０４がそのデータをプロセッサ１０２に送るまでなど、一時的に格納する。メモリプロセッサ１０４は、適切な時点で、とりわけ待ち時間が最小になることを保証するために、プリフェッチされたデータをプロセッサ１０２へとシステムバス１０３を介して移送する。

【００１４】

図２は、本発明の一実施形態に従った、例示的スペキュレータを示す図である。スペキ

ュレータ１０８は、予測２０３の生成元である読み取り要求２０１を受け取るように構成される。図に示されるように、スペキュレータ１０８は、制御情報およびアドレス情報を順次予測器（「ＳＥＱ．予測器」）２０６および非順次予測器（「ＮＯＮＳＥＱ．予測器」）２１６へと提供するように構成された、予測コントローラ２０２を含み、それらはどちらも予測２０３を生成する。予測コントローラ２０２は、全体としてまたは部分的に、最適な量およびタイプの予測を提供するように予測生成プロセスを管理する働きをする。たとえば、予測コントローラ２０２は、読み取り要求２０１で指定された特定のキャッシュラインまたはキャッシュラインのグループに対して、生成される予測の数およびタイプを変えることができる。他の例として、予測コントローラ２０２は、ターゲットキャッシュ２１８内の使用可能なメモリなどのリソースを保存するように、または予測アドレスの重複によるメモリ１１２への不必要なアクセスを最小にするように、一定の予測の生成を抑制するための抑制器２０４を含む。予測コントローラ２０２は、非順次予測の生成を早めるために、オプションで促進器２０５を含むことができる。促進器２０８は、図８に示されるように、非順次予測が関係する非線形アドレスストリームの直前のアドレスの検出に先立って、非順次予測の生成をトリガするように動作する。予測コントローラ２０２については、順次予測器２０６および非順次予測器２１６の以下の説明の後で、より詳細に論じる。

【００１５】

順次予測器２０６は、ある程度の見込みを有する予測（すなわち予測アドレス）を生成するように構成される。すなわち、順次予測器２０６は、１つまたは複数のパターンの定期的な読み取り要求２０１に経時的に従うことが予測される予測を生成する。これらのパターンは、メモリ参照がそれらの間に空間的な局所性を有するという事実から生じる。たとえば、プロセッサ１０２がプログラム命令を実行する場合、読み取り要求２０１のストリームは、システムバス１０３をトラバースする際に事実上順次的とすることができる。順次パターンに従ってアドレスを予測するために、「前方順次予測」として以下で説明するあるタイプの推測技法が、順次アドレスを予測することができる。次に、このタイプの推測技法について説明する。

【００１６】

前方順次予測器２０８は、いくつかの順次アドレスを昇順で生成するように構成される。したがって、プロセッサ１０２が、昇順アドレスのストリームを含む一連の読み取り要求２０１をシステムバス１０３に伝送すると、前方順次予測器２０８は、追加の昇順アドレスをプリフェッチするためにいくつかの予測を生成することになる。前方順次予測器（「ＦＳＰ」）２０８の一例が、図３Ａに示される。図３Ａに示されるように、ＦＳＰ　２０８は、アドレスＡ０などのアドレスを受け取り、Ａ０アドレスから前方（すなわち昇順）順に１つまたは複数のアドレスを生成する。Ａ０の表記法は、１つまたは複数の予測の形成元である基本アドレス（すなわちＡ＋０）を識別する。したがって、表記法Ａ１、Ａ２、Ａ３などはＡ＋１、Ａ＋２、Ａ＋３などのアドレスを表し、表記法Ａ（－１）、Ａ（－２）、Ａ（－３）などはＡ－１、Ａ－２、Ａ－３などのアドレスを表す。これらの表記法は１アドレスごとに昇順または降順の一連のアドレスを表すが、任意のパターン可能なアドレスセットを順次的と呼ぶことが可能である。全体を通じて使用される場合、順次アドレスを単一の文字によって表し、単一の文字とみなすことができる。たとえば、「Ａ」はＡ０、Ａ１、Ａ２、Ａ３などを表し、「Ｂ」はＢ０、Ｂ１、Ｂ２、Ｂ３などを表す。したがって、「Ａ」および「Ｂ」はそれぞれ順次アドレスストリームを表すが、「Ｂ」のアドレスストリームは「Ａ」のそれに対して非順次的である。

【００１７】

さらに図３Ａでは、ＦＳＰ　２０８は少なくともイネーブル信号およびバッチ信号を受け取るように示され、その両方が予測コントローラ２０２によって提供される。イネーブル信号は、前方順次予測が生成されるかどうかを制御し、生成される場合、バッチ信号はＦＳＰ　２０８が生成する順次アドレスの数を制御する。この例では、バッチ信号は、基本アドレスの他に「７つ」のアドレスが予測されることを示す。したがって、ＦＳＰ　２

０８は、前方順のアドレスＡ１からＡ７を生成する。したがって、スペキュレータ１０８が読み取り要求２０１の一部としてＡ０などのアドレスを受け取る場合、順次予測器２０６は、予測２０３の一部としてアドレスＡ１、Ａ２、Ａ３、．．．、Ａｂを提供することが可能であり、ここでｂは「バッチ」の数である。

【００１８】

図２のブラインドバック順次予測器２１０は、１つの順次アドレスを生成するように構成されるが、基本アドレスからは降順である。ブラインドバック順次予測器（「ブラインドバック」）２１０の一例が図３Ｂに示され、ここでは、ブラインドバック順次予測器２１０がアドレスＡ０などの１つまたは複数のアドレスを受け取り、Ａ０アドレスから後方（すなわち降順）順にアドレスＡ（－１）などの予測を１つだけ生成する。ＦＳＰ２０８の場合と同様に、ブラインドバック順次予測器２１０は、後方予測を生成するかどうかを制御するためのイネーブル信号も受け取る。

【００１９】

図２のバックセクタ順次予測器２１４は、システムバス１０３から他の特定のキャッシュラインを検出した後に、特定のキャッシュラインを予測として生成するように構成される。とりわけ、バックセクタ順次予測器２１４は、ある一定の読み取り要求２０１が高位キャッシュライン用であることを検出した場合、関連付けられた低位キャッシュラインが予測として生成される。高位キャッシュラインは、奇数アドレスを含む上位（「フロント」）セクタと呼ぶことが可能であり、低位キャッシュラインは、偶数アドレスを含む下位（「バック」）セクタと呼ぶことが可能である。例示のために、キャッシュラインは１２８バイトを含み、６４バイトの高位キャッシュライン（すなわち１２８バイトの上位半分）および６４バイトの低位キャッシュライン（すなわち１２８バイトの下位半分）からなるものと考えてみる。

【００２０】

バックセクタ順次予測器２１４の一例が図３Ｃに示されており、ここでは、１つまたは複数のアドレスを受け取るバックセクタ順次予測器（「バックセクタ」）２１４が示されている。アドレスＡＵなどのキャッシュラインの上位またはフロントセクタに関する読み取り要求２０１を受け取ると、バックセクタ順次予測器２１４は１つだけの予測、アドレスＡＬを生成する。このタイプの推測技法は、プロセッサ１０２が、典型的にはキャッシュラインの上位またはフロントセクタを要求した後に、下位またはバックセクタを要求するという現象を活用する。また、バックセクタ順次予測器２１４は、バックセクタ予測を生成するかどうかを制御するためのイネーブル信号も受け取る。

【００２１】

図２の逆順次予測器２１２は、いくつかの順次アドレスを降順で生成するように構成される。したがって、プロセッサ１０２が、降順アドレスのストリームを含む一連の読み取り要求をシステムバス１０３に伝送すると、逆順次予測器２１２は、追加の降順アドレス用にいくつかの予測を生成することになる。逆順次予測器（「ＲＳＰ」）２１２の一例が図３Ｄに示される。図３Ｄに示されるように、ＲＳＰ　２１２は、アドレスＡ０、Ａ（－１）、およびＡ（－２）などのアドレスのストリームを検出し、これに応答して、基本アドレスＡ０から逆（すなわち降順）順で１つまたは複数のアドレスを生成する。図３Ｄは、ＲＳＰ　２１２が少なくともイネーブル信号、バッチ信号、および信頼レベル（「Ｃｏｎｆ．」）信号を受け取ることも示し、これら信号のすべてが予測コントローラ２０２によって提供される。イネーブル信号およびバッチ信号はＦＳＰ　２０８で使用される場合と同様に動作するが、信頼レベル（「Ｃｏｎｆ．」）信号は、逆順の予測の生成をトリガする時点を定義するしきい値を制御する。

【００２２】

さらに図３Ｄは、本発明の特定の実施形態に従った例示的ＲＳＰ　２１２の挙動を示す図３１０を示す。ここでは、信頼レベル「２」がトリガレベル３１２を設定し、バッチ信号は、トリガアドレス以外に「５つ」のアドレスが予測されることを示す。トリガアドレスとは、予測器に予測を生成させるアドレスのことである。間隔Ｉ１中にＡ（０）を検出

した後、ＲＳＰ　２１２が、続く間隔Ｉ２中にアドレスＡ（－１）も検出すると考えてみる。次に、間隔Ｉ３中にアドレスＡ（－２）を検出すると、検出されたストリームが一連の降順アドレスであるというある一定の信頼レベルに達する。トリガレベル３１２を上回った場合にこの信頼レベルに達し、これによって、ＲＳＰ　２１２が逆順のアドレスＡ（－３）からＡ（－７）を生成する。したがって、スペキュレータ１０８が一連の読み取り要求２０１としてＡ０、Ａ（－１）、およびＡ（－２）などの一定数のアドレスを受け取る場合、その後順次予測器２０６は、予測２０３の一部としてアドレスＡ（－３）、Ａ(－４)、Ａ(－５)、．．．、Ａｂを提供することが可能であり、ここでｂは「バッチ」の数である。いくつかの実施形態では、ＲＳＰ　２１２は信頼レベルを採用せず、基本アドレス後に始まる予測を生成することに留意されたい。本発明の他の実施形態では、本明細書で説明される他の予測器において信頼レベルの概念が採用される。ＲＳＰ　２１２および順次予測器２０６の他の構成予測器の制御については、以下でより詳細に論じるが、次に、図２の非順次予測器２１６について説明する。

【００２３】

非順次予測器２１６は、たとえアドレスが読み取り要求２０１の非線形ストリーム内にある場合でも、スペキュレータ１０８によって検出されたアドレスに続いて１つまたは複数の予測（すなわち予測アドレス）を生成するように構成される。典型的には、次のアドレスを予測する要求アドレスの顕著なパターンがない場合、先行アドレスのみに基づく予測は困難である。しかしながら本発明の実施形態によれば、非順次予測器２１６は、１つまたは複数の先行アドレスからパターン化不可能な予測アドレスを含む非順次予測を生成する。「パターン化不可能」な予測とは、先行アドレスによってパターン化できないか、または先行アドレスに対して不規則な、予測のことである。パターン化不可能な予測のタイプの１つが、非順次予測である。非順次予測が基づく先行アドレスは、即値アドレス、またはトリガアドレスとして構成された任意のアドレスの、いずれかとすることができる。とりわけ、読み取り要求２０１のストリーム内の２つまたはそれ以上のアドレスにわたって１つまたは複数のパターンが欠如していることは、プロセッサ１０２が、メモリロケーションの様々な空間的位置からの命令およびデータのフェッチに関してやや散漫な様式でプログラム命令を実行していることを示す。

【００２４】

非順次予測器２１６は、先行アドレスから非順次予測として分類可能な１つまたは複数の可能な非順次アドレスへの関連を格納するためのリポジトリとして、ターゲットキャッシュ２１８を含む。ターゲットキャッシュ２１８は、タイムリーな形で非順次予測を生成するために、そのコンテンツと着信する検出されたアドレスとを容易に比較するように設計される。非順次予測の生成元である検出されたアドレスは、「トリガ」アドレスと呼ばれ、結果として生じる予測は、この２つの間のパターン化不可能な関連の「ターゲット」である。次に、例示的非順次予測器２１６について説明する。

【００２５】

図４は、本発明の一実施形態に従った、例示的非順次予測器２１６を示す図である。非順次予測器２１６は、ターゲットキャッシュ４２２であるリポジトリに動作可能に結合された、非順次予測エンジン（「ＮｏｎＳｅｑ．予測エンジン」）４２０を含む。ターゲットキャッシュ４２２は、各トリガアドレスと１つまたは複数の対応するターゲットアドレスとの間の関連を維持する。図４が、非順次アドレスを関連付ける多くの方法のうちの１つを示すことに留意されたい。ここでは、ツリー構造が特定のトリガアドレスをその対応するターゲットアドレスに関係付けている。この例では、ターゲットキャッシュ４２２は、アドレス「Ｂ」、「Ｘ」、および「Ｌ」などの可能な非順次予測のアドレスへの関連を形成する元となる、トリガアドレスとしてのアドレス「Ａ」を含む。これら３つのターゲットアドレスは、それぞれアドレス「Ｃ」および「Ｇ」、「Ｙ」、ならびに「Ｍ」に対するトリガアドレスでもある。ターゲットキャッシュ４２２の形成および動作について、以下でより詳細に考察する。アドレス「Ａ」が、図４に示されていないトリガアドレスに対するターゲットアドレスともなり得ることに留意されたい。さらに、図示されていないア

ドレス間の多くの他の関連も可能である。

【００２６】

非順次予測エンジン４２０は、少なくとも４つの信号および任意数のアドレス４０２を受け取るように構成される。非順次予測エンジン４２０の動作を制御するために、予測コントローラ２０２は「バッチ」信号および「イネーブル」信号を提供し、これらはどちらも前述の信号と事実上同様である。予測コントローラ２０２は、２つの他の信号、幅（「Ｗ」）信号および深さ（「Ｄ」）信号も提供する。これらの信号はターゲットキャッシュ４２２の形成を制御するものであり、幅信号Ｗはトリガアドレスの予測元となり得る可能なターゲットの数を設定し、深さ信号Ｄはトリガアドレスに関連付けられるレベルの数を設定する。後者の例は、Ｄが深さ「４」を示す場合である。これは、アドレスＡが第１のレベルであり、アドレスＢが第２のレベルであり、アドレスＣおよびＧが第３のレベルであり、アドレスＤが第４のレベルであることを意味する。前者の例は、Ｗが「２」に設定された場合である。これは、３つのアドレス「Ｂ」、「Ｘ」、および「Ｌ」のうちの２つのみが非順次予測に使用されることを意味する。

【００２７】

図４は、それぞれが以前に検出されたアドレスにパターン化不可能なアドレスを含む、概念上、非順次アドレスストリーム４０４、４０６、４０８、４１０、および４１２で示されたアドレスなどの、例示的アドレス４０２を、予測コントローラ２０２から受け取るように構成された、非順次予測エンジン４２０も示す。たとえば、ストリーム４０４はアドレス「Ａ」を含み、その後にアドレス「Ｂ」が続き、さらにその後にアドレス「Ｃ」が続く。非順次アドレスの場合と同様に、「Ａ」から「Ｂ」を予測するため、および「Ｂ」から「Ｃ」を予測するためのパターンを検出することは、プロセッサ１０２からの読み取り要求２０１を監視するだけでは困難である。このため、非順次予測器２１６は、特定のトリガアドレスとそのターゲットアドレスとの間のパターン化不可能な関連の予測を実行可能にするために、ターゲットキャッシュ４２２を形成する。非順次予測エンジン４２０は、非順次予測を形成すると、関連付けられたターゲットアドレスから予測のグループを生成する。したがって、トリガアドレス「Ａ」がアドレス「Ｂ」（すなわち、基本アドレスとしてのＢ０）の非順次予測につながる場合、予測アドレスはＢ０、Ｂ１、Ｂ２、．．．、Ｂｂを含むことになり、ここでｂはバッチ信号によって設定される数である。

【００２８】

本発明の一実施形態では、非順次予測エンジン４２０は、アドレス４０２のそれぞれから後続のアドレスへの関連を格納しながら、ターゲットキャッシュ４２２を形成する。たとえば、ストリーム４０４のアドレスＡを検出すると、非順次予測エンジン４２０は、ＡからＢへの関連、ＢからＣへの関連、ＣからＤへの関連などの関連を、ターゲットキャッシュ４２２に追加する。非順次予測エンジン４２０は、他のストリーム４０６、４０８などのアドレスを検出した場合も、同様に実行する。

【００２９】

特定の実施形態によれば、ターゲットキャッシュ４２０は、これらの関連を表４３０、４４０、および４５０などの表形式で格納する。これらの表は、それぞれトリガアドレスとターゲットアドレスとの間の関連を格納するための、トリガ列４２６およびターゲット列４２８を含む。次に、すべてのストリームのアドレス４０２が、ターゲットキャッシュ４２２の表４３０、４４０、および４５０に格納されるものと考えてみる。表４３０に示されるように、トリガ－ターゲット関連４３２、４３４、および４３６は、それぞれＡからＢ、ＢからＣ、およびＧからＱへの関連を記述する。他のトリガ－ターゲット関連４３８は、ＣからＤなどの関連を含む。同様に、表４４０はＡからＸへの関連を記述するためのトリガ－ターゲット関連４４２を含み、表４５０はＡからＬへの関連を記述するためのトリガ－ターゲット関連４５２を含む。

【００３０】

図４は、表４３０、４４０、および４５０がそれぞれ、同じトリガアドレスに関する複数のトリガ－ターゲット関連の相対的優先順位を記述する、「Ｗａｙ　０」、「Ｗａｙ

１」、および「Ｗａｙ　２」として識別されることを示す。この場合、Ｗａｙ　０は最高優先順位に、Ｗａｙ　１は第２位優先順位に、という具合に関連付けられる。この例では、表４３０のトリガーターゲット関連４３２は、ＡからＢへの関連が、表４４０のトリガーターゲット関連４４２であるＡからＸへの関連よりも優先順位が高いことを示す。したがって、ターゲットキャッシュ４２２がこれらの関連を含んだ後、次に非順次予測エンジン４２０がアドレスＡを検出した場合（予測コントローラ２０２が非順次予測エンジン４２０を動作させられる限り）、表の相対的優先順位により、アドレスＢは最高優先順位、続いてアドレスＸは第２位優先順位、として予測されることになる。

【００３１】

本発明の一実施形態によれば、相対的優先順位は少なくとも２つの方法で決定される。第１に、トリガーターゲット関連が最初に検出され、ターゲットキャッシュ４２２内に配置された場合、これには最高優先順位が関連付けられる。第２に、非順次予測エンジン４２０が、トリガーターゲット関連が成功である（たとえば、その特定の関連に基づく非順次予測の結果として生じた、最新のキャッシュヒットが存在する）と判定した場合、そのトリガーターゲット関連には最高優先順位が関連付けられる。「最新の」キャッシュヒットとは、特定のトリガアドレスに関連付けられたターゲットアドレスのうちの少なくとも１つの新しいキャッシュヒットのことである。さらに、以前の「最高優先順位」（ｌｅｇ　０としても指定される）は、対応する関連をＷａｙ　１の表に移動させることによって、第２位優先順位（ｌｅｇ　１としても指定される）にシャッフルされる。一例として、ＡからＸへの関連が第１のトリガーターゲット関連としてターゲットキャッシュ４２２に導入される場合の、第１の時点について考えてみる。結果として、表４３０内（すなわちｗａｙ　０）に配置されることにより、最高優先順位（すなわち初期にはｌｅｇ　０）が関連付けられることになる。何らかのその後の時点で、ターゲットキャッシュ４２２はＡからＢへの関連を表４３０に挿入する（最高優先順位、ｌｅｇ　０）。また、ＡからＸへの関連も表４４０に移動される（第２位優先順位、ｌｅｇ　１）。本発明の特定の実施形態では、トリガーターゲット関連が格納される表は、インデックスを構成するアドレスビットの一部に依存する。

【００３２】

再度図２を参照すると、予測コントローラ２０２は、順次予測器２０６および非順次予測器２１６の両方を制御するように構成される。予測コントローラ２０２は、順次予測器２０６または非順次予測器２１６のいずれか、あるいはその両方によって生成される予測の量ならびにタイプを制御する。また、予測コントローラ２０２は、冗長または重複予測などの、そうでなければ不要な予測２０３の生成も抑制する。予測器２０８、２１０、２１２、２１４、および２１６のそれぞれが同時に動作可能な場合、予測２０３の数はプリフェッチャリソースに負荷をかけすぎないように管理しなければならない。予測コントローラ２０２は、この操作および他の同様の操作を実行するために、抑制器２０４を採用する。

【００３３】

本発明の一実施形態では、抑制器２０４が生成される予測の量を制御する。これは、第１に読み取り要求２０１の一定の属性を確認することによって実行される。具体的に言えば、抑制器２０４は、読み取り要求２０１がプログラム命令（すなわち「コード」）またはプログラムデータ（すなわち「非コード」）のどちらに関連するかを判定する。通常、プログラムデータ以外のコードを取り出すための読み取り要求２０１は、事実上より順次的であるか、または少なくともパターン化可能である傾向がある。これは、プロセッサ１０２が一般に、プログラムデータに対する要求よりも線形的な方法で命令を実行するためである。したがって、抑制器２０４は、順次予測器２０６および非順次予測器２１６に対して、読み取り要求２０１がプログラムデータに関係する場合は予測生成を抑制するように指示することができる。これにより、擬似予測の生成防止に役立つ。

【００３４】

抑制器２０４は、読み取り要求２０１が非プリフェッチ「デマンド」またはプリフェッ

チのいずれであるかを確認することによって、順次予測器２０６および非順次予測器２１６が生成する予測の量を調整することもできる。プロセッサ１０２は、通常、確実に必要な何らかのケースにおいて、プログラム命令またはプログラムデータをメモリ１１２から取り出すように要求する（非プリフェッチデマンドとして）が、プロセッサ１０２は、後で必要となることを見込んで、プログラム命令またはプログラムデータをプリフェッチするように要求するだけでも可能である。確実に必要である方が必要となる見込みであるよりも重要な処理である可能性が高いため、抑制器２０４は特定の予測器に対し、デマンド読み取り要求２０１に基づく予測を優先して、プリフェッチ読み取り要求２０１に基づく予測を抑制するように指示することができる。

【００３５】

表１は、生成される予測の数を抑制するための例示的技法を示す。すなわち、読み取り要求２０１がコードおよびデマンドの両方に関連する場合、抑制器２０４は最も抑制的でなくなる。すなわち、予測コントローラ２０２は「バッチ」を表１でバッチサイズ（４）と示されるような大規模サイズに設定することになる。特定の例では、バッチサイズ（４）は７に設定することができる。しかしながら、前述の理由では、読み取り要求２０１がプログラムデータ（すなわち非コード）およびプロセッサ生成プリフェッチの両方に関係する場合、抑制器２０４は最も抑制的となる。したがって、予測コントローラ２０２は「バッチ」を表１でバッチサイズ（１）と示されるような小規模サイズに設定することになる。一例として、バッチサイズ（１）は１に設定することができる。他のケースでは、予測コントローラ２０２は、バッチサイズ（２）およびバッチサイズ（３）などの他のバッチサイズを使用して、予測抑制のレベルを変更することができる。本発明の一実施形態に従った抑制器は、プロセッサ要求がデータまたはプリフェッチ要求、あるいはその両方に対するものである場合、「バッチ」量を減少させることによって、少なくとも１つの予測アドレスの生成を抑制するように構成されるが、表Ｉはこれに限定されない。たとえば、コードまたは命令に関するプロセッサ要求は、「バッチ」サイズを増加させるのではなく減少させることができる。他の例として、デマンドに関する要求も、「バッチ」サイズを増加させるのではなく減少させることができる。当業者であれば、多くの変形形態が本発明の範囲内であることを理解されよう。

【表１】

コードまたはデータ	デマンドまたはプリフェッチ	バッチ
非コード（すなわちデータ）	プリフェッチ	バッチサイズ (1)
非コード（すなわちデータ）	デマンド	バッチサイズ (2)
コード	プリフェッチ	バッチサイズ (3)
コード	デマンド	バッチサイズ (4)

表1. 読み取り要求のタイプ

【００３６】

抑制器２０４は、順次予測器２０６および非順次予測器２１６が生成する予測のタイプも調整することができる。第１に、予測コントローラ２０２が前方順次予測器２０８および逆順次予測器２１２の両方を同時に実行可能にできるものと考えてみる。したがって、抑制器２０４は、プロセッサ１０２が降順でアドレス読み取りを要求している場合、昇順でのアドレスの予測を最小限にするために、逆順次予測器２１２がトリガする（すなわち信頼レベルを超える）場合、少なくとも前方順次予測器２０８を実行不可にするよう、予

測コントローラ２０２に指示する。

【００３７】

第２に、予測コントローラ２０２が順次予測（すなわち、前方順次予測器２０８または逆順次予測器２１２のいずれか）を動作可能にする場合、あるアドレスが、バック予測（すなわち、ブラインドバック順次予測器２１０またはバックセクタ順次予測器２１４のいずれか）をトリガすると考えてみる。この場合、抑制器２０４は、前方順次予測器２０８または逆順次予測器２１２のいずれかについて、その初期量から１ずつバッチを抑制する。すなわち、「バッチ」が初期に７に設定されていた場合、ブラインドバック順次予測器２１０またはバックセクタ順次予測器２１４のいずれかがトリガまたは活動化されると同時に、「バッチ」は１つだけ減少することになる。たとえば、前方順次予測器２０８に関するバッチがアドレスＡ０、Ａ１、Ａ２、．．．、Ａ７を生成するように設定されている場合、およびブラインドバック順次予測器２１０が１つまたは複数の読み取り要求２０１に対して実行可能である場合、前方順次予測器２０８は予測Ａ１、Ａ２、．．．、Ａ６のみを生成する。最終結果は、それらの読み取り要求２０１に対して、予測Ａ（－１）、Ａ（０）、Ａ１、Ａ２、．．．、Ａ６のセットとなり、バック予測は予測Ａ（－１）を提供する。

【００３８】

第３に、予測コントローラ２０２はオプションで、プロセッサからのアドレス２０１の順次ストリームで最初に予測が生成された後に、ブラインドバック順次予測器２１０またはバックセクタ順次予測器２１４のいずれかに対して、それらの予測を抑制不可にすることができる。これは、シーケンスの基本アドレスが確立された後、後続の前方または逆順次予測も後方タイプの推測（たとえ１アドレス後方であっても）を予測するためである。たとえば、前方順次予測Ａ２、Ａ３、およびＡ４は、すべてがすでに予測されている（基本アドレスがＡ０の場合）後方タイプの予測Ａ１、Ａ２、およびＡ３もカバーしている。抑制器２０４は、他のタイプの予測を抑制するように構成することが可能であり、次にその例について説明する。

【００３９】

図５は、本発明の一実施形態に従った、非順次予測を抑制する例示的技法を示す図である。この技法によれば、抑制器２０４は、ターゲットキャッシュ４２２へのトリガ－ターゲット関連の格納を必要とする、そうでなければ非順次とみなされる可能性のあるインタリーブされた順次ストリームを検出する。リソース、特にターゲットキャッシュ４２２内の使用可能なメモリを保存するために、抑制器２０４はストリーム５０２内などの非順次アドレスを解析し、それらの非順次アドレスをインタリーブされた順次ストリームとしてモデル化する。図に示されるように、ストリーム５０２は、それぞれが各間隔Ｉ１、Ｉ２、Ｉ３、Ｉ４、Ｉ５、Ｉ６、Ｉ８、およびＩ９中に検出された、アドレスＡ０、Ｂ０、Ｃ０、Ａ１、Ｂ１、Ｃ１、Ａ２、Ｂ２、およびＣ２からなる。抑制器２０４は、非順次アドレスを順序どおりにモデル化するための、表５０４などのデータ構造を含む。表５０４は、ストリーム５０２を分解するための任意数のストリームトラッカを含むことができる。具体的に言えば、ストリームトラッカ５２０、５２２、および５２４は、それぞれ順次ストリームＢ０、Ｂ１、およびＢ２と、Ａ０、Ａ１、およびＡ２と、Ｃ０およびＣ１とをモデル化するように設計される。Ａ７（図示せず）などの、ストリーム５０２から後に検出される読み取りアドレスがこれらのストリームと比較され、追跡されているストリームに対して非順次予測が依然として抑制可能であるかどうかが調べられる。

【００４０】

抑制器２０４は動作時に、シーケンスの第１のアドレスなどの基本アドレス５１０を格納することによって、順次ストリームを追跡する。その後抑制器２０４は、最新検出アドレス５１４を維持する。各新しい最新検出アドレスについて（たとえば、ストリームトラッカ５２０のＢ２）、前の最新検出アドレス（たとえばストリームトラッカ５２０のＢ１）は、オプション列である列５１２内に配置されることによって無効（「無効」）とされる。抑制器２０４は、この例示的技法を使用して、他のタイプの予測が使用可能な場合に

不必要な非順次予測の生成を抑制する。したがって図５に示された例では、前方順次予測器２０８がストリーム５０２に関する予測を適切に生成することができる。

【００４１】

図６は、本発明の一実施形態に従った、非順次予測を抑制する他の例示的技法を示す図である。この技法によれば、抑制器２０４は、図５に記載されたプロセスと同様のインタリーブされた順次ストリームとして非順次アドレスをモデル化する。しかしながら、図６の技法は、任意数のスレッドにわたる順次ストリームを検出するためにそれぞれが使用される、複数のデータ構造を実施する。この例では、表６０４、６０６、および６０８は、それぞれスレッド（０）（「Ｔ」）、スレッド（１）（「Ｔ’」）、およびスレッド（２）（「Ｔ’’」）用のストリームトラッカを含む。したがってストリーム６０２の非順次アドレスは、この技法を使用して、非順次予測を抑制するように、複数のスレッドにわたる複数の順次ストリームとしてモデル化することができる。この技法は、逆順次ストリームまたは他のタイプの予測に適用可能であることに留意されたい。

【００４２】

図７は、本発明の特定の実施形態に従った、非順次予測を抑制するための他の技法を示す図である。アドレスストリーム７０２の場合、アドレスＡ４とＢ０との間に非順次性が存在する。しかしながらいくつかのケースでは、これらの要求された読み取りアドレス間の時間差が非常に短い場合、非順次予測を採用する十分な時間がないことになる。抑制器２０４のマッチャ７０６は、アドレスＡ４とＢ０との間の時間差ｄを比較するように動作する。ｄがしきい値ＴＨに等しいかまたはこれより大きい場合、マッチャ７０６は非順次予測器２１６を動作可能にする（すなわち「抑制しない」）信号を送る。しかしながら、ｄがＴＨよりも小さい場合、マッチャ７０６は非順次予測器２１６を使用不可にする信号を送り、それによって予測を抑制する。

【００４３】

抑制器２０４によって実施可能な他の抑制メカニズムは、以下のとおりである。一般に、プロセッサ１０２がフロントセクタアドレスを要求した後、バックセクタアドレスに関して要求を出すまでに経過する時間量は有限である。この時間量が十分に長い場合、バックセクタアドレスの読み取り要求は不規則（すなわちフロントセクタに対してパターン化不可能）に見える可能性がある。これを防止するために、抑制器２０４は、プロセッサ１０２によるフロントセクタ読み取りのリストを維持するように構成される。フロントセクタアドレスの検出に続いて、そのフロントセクタアドレスとアドレスが比較される。対応するバックセクタに達すると、そのように認識されることになる。したがって、その他の形の非順次性ならびにその予測を抑制することができる。

【００４４】

図８は、本発明の特定の実施形態に従った、予測の生成を促進するための例示的技法を示す図である。具体的に言えば、促進器２０５（図２）は、この技法に従って非順次予測の生成を早めるように動作する。この例では、ストリーム８０２は２つの隣接する順次ストリームＡ０からＡ４およびＢ０からＢ３を含む。非順次予測器２１６は、通常、アドレスＡ４をトリガアドレス８０８として指定し、アドレスＢ０をターゲットアドレス８１０として指定する。しかしながら、非順次予測を生成するための時間を減少させるために、トリガアドレス８０８を新しいトリガアドレス８０４（すなわちＡ０）に変更することができる。したがって、ターゲットアドレスに新しいトリガアドレスを指定することで、次にプロセッサ１０２がストリーム８０２のアドレスを要求した場合、非順次予測器２１６は最近のアドレスよりも前のアドレスを検出すると同時に直ちにその予測を生成することができる（すなわち、Ａ４ではなくＡ０が「新しい」トリガアドレスとして検出された場合、予測を生成する）。これにより、最も適切な時点での非順次予測の生成が保証される。

【００４５】

図９は、本発明の一実施形態に従った、他の例示的スペキュレータを示す図である。この例では、プリフェッチャ９００は、不必要な予測生成を最小限に維持するように冗長ア

ドレスをフィルタリングするためのフィルタ９１４を備えたスペキュレータ９０８を含む。図９のプリフェッチャ９００は、マルチレベルキャッシュ９２０および予測インベントリ９１６も含む。ここで、マルチレベルキャッシュ９２０は、第１レベルの戻りデータキャッシュ（「ＤＲＣ１」）９２２および第２レベルの戻りデータキャッシュ（「ＤＲＣ２」）９２４からなる。第１レベルの戻りデータキャッシュ９２２は、一般に、短期データストアとして説明可能であり、第２レベルの戻りデータキャッシュ９２４は、一般に、長期データストアとして説明可能である。マルチレベルキャッシュ９２０は、メモリ１１２からプリフェッチされたプログラム命令およびプログラムデータを、プロセッサ１０２が必要とするまで格納する。同様に、予測インベントリ９１６は、メモリ１１２にアクセスするためにアービタ９１８によって選択されるまで、生成された予測に一時ストレージを提供する。アービタ９１８は、アービトレーション規則に従って、メモリ１１２にアクセスして命令およびデータをプリフェッチするために、生成されたどの予測が発行されるかを決定するように構成される。

【００４６】

フィルタ９１４は、キャッシュフィルタ９１０およびインベントリフィルタ９１２の、少なくとも２つのフィルタを含む。キャッシュフィルタ９１０は、新しく生成された予測と、マルチレベルキャッシュ９２０内にすでに格納された命令およびデータをプリフェッチした以前の予測とを、比較するように構成される。したがって、マルチレベルキャッシュ９２０に関して、１つまたは複数の新しく生成された予測が、任意の以前に生成された予測と重複する場合、予測の数を最小にするようにそれらの冗長予測は無効とされる。さらにインベントリフィルタ９１２は、新しく生成された予測と、すでに生成され、予測インベントリ９１６に格納された予測とを、比較するように構成される。したがって、１つまたは複数の新しく生成された予測が予測インベントリ９１６に格納された予測と重複する場合、予測の数を最小にするように任意の冗長予測は無効とされ、それによってプリフェッチャリソースが解放される。

【００４７】

非順次予測器に関する例示的実施形態

図１０は、本発明の特定の実施形態に従った、例示的非順次（「ＮＯＮＳＥＱ」）予測器１０１０を示すブロック図である。この例では、非順次予測器１０１０は、順次予測を生成するための順次予測器１０１２も含むスペキュレータ１００８内に常駐するように示される。プリフェッチャ１００６はスペキュレータ１００８を含み、要求される前に（図示せず）メモリからプログラム命令およびプログラムデータの両方を「フェッチ」し、その後プロセッサ（図示せず）によって要求されると、フェッチされたプログラム命令およびプログラムデータをそのプロセッサに提供するように動作する。それらを使用する前にフェッチする（すなわち「プリフェッチ」）ことによって、プロセッサのアイドル時間（たとえば、プロセッサがデータ不足である間の時間）が最小となる。非順次予測器１０１０は、予測を生成するための非順次予測エンジン（「予測エンジン」）１０２０と、予測を格納および優先順位付けするためのターゲットキャッシュ１０３０とを含む。

【００４８】

プリフェッチャ１００６は、フィルタ１０１４、オプションの予測インベントリ１０１６、オプションのアービタ１０１８、およびマルチレベルキャッシュ１０４０も含む。ここで、フィルタ１０１４は、新しく生成された予測と、プログラム命令およびプログラムデータをすでにマルチレベルキャッシュ１０４０にプリフェッチされた状態にする以前の予測とを比較するように構成された、キャッシュフィルタ（図示せず）を含む。したがって、任意の新しく生成された予測が、マルチレベルキャッシュ１０４０に格納された任意の以前に生成された予測と重複する場合、予測の数を最小にするようにその冗長予測は無効とされ、それによってプリフェッチャリソースが解放される。予測インベントリ１０１６は、メモリにアクセスするためにアービタ１０１８によって選択されるまで、生成された予測を格納するための一時ストレージを提供する。アービタ１０１８は、メモリにアクセスして命令およびデータをプリフェッチするために、生成されたどの予測が発行される

かを決定するように構成される。

【００４９】

マルチレベルキャッシュ１０４０は、第１レベルの戻りデータキャッシュ（「ＤＲＣ１」）１０４２および第２レベルの戻りデータキャッシュ（「ＤＲＣ２」）１０４４からなる。第１レベルの戻りデータキャッシュ１０４２は、一般に、短期データストアとして説明可能であり、第２レベルの戻りデータキャッシュ１０４４は、一般に、長期データストアとして説明可能である。本発明の実施形態によれば、第１レベルの戻りデータキャッシュ１０４２または第２レベルの戻りデータキャッシュ１０４４、あるいはその両方が、予測アドレス（すなわちターゲットアドレス）に基づいてプリフェッチされた、プロフェッチ済みプログラム命令およびプログラムデータを格納することができる。図に示されるように、マルチレベルキャッシュ１０４０に格納されたプリフェッチ済み予測情報は、データ（ＴＲＴ１）およびデータ（ＴＲＴ２）として表される。この表記法は、ターゲットアドレスＴＲＴ１およびＴＲＴ２が、予測情報を表すデータのプリフェッチに寄与したことを意味する。図に示されるように、ならびに以下で論じるように、データ（ＴＲＴ１）およびデータ（ＴＲＴ２）は、それぞれ、予測識別子（「ＰＩＤ」）１および２と共にマルチレベルキャッシュ１０４０内に格納される。データ（ＴＲＴ１）またはデータ（ＴＲＴ２）のいずれかがプロセッサによって要求された場合、対応するターゲットアドレス（たとえばＴＲＴ１）および予測識別子が非順次予測器１０１０に送られることになる。

【００５０】

スペキュレータ１００８は、動作時に、プロセッサがメモリへのアクセスを要求する（「読み取り要求」）場合、システムバスを監視する。プロセッサがプログラム命令を実行する場合、スペキュレータ１００８は、まだプロセッサによって使用されていないプログラム命令およびプログラムデータを含むアドレスに関する読み取り要求を検出する。考察のために、「アドレス」は、一般にメモリと、マルチレベルキャッシュ１０４０などのキャッシュメモリとの間で転送される、メモリのキャッシュラインまたは単位に関連付けられる。キャッシュメモリは、ターゲットキャッシュ１０３０の外部リポジトリの一例であることに留意されたい。

【００５１】

検出された読み取り要求に基づいて、非順次予測器１０１０は、プロセッサによって次に要求される可能性のある構成可能な数の予測アドレスを生成することができる。具体的に言えば、非順次予測器１０１０は、たとえアドレスが読み取り要求の非線形ストリーム内にある場合でも、そのアドレスの検出に続いて、１つまたは複数の予測（すなわち予測アドレス）を生成するように構成される。典型的には、次のアドレスを予測する要求アドレスの顕著なパターンがない場合、先行アドレスのみに基づく予測は困難である。しかしながら本発明の実施形態によれば、非順次予測エンジン１０２０は、１つまたは複数の先行アドレスからパターン化不可能な予測アドレスを含む、非順次予測を生成する。「パターン化不可能」な予測とは、先行アドレスによってパターン化できないか、または先行アドレスに対して不規則な、予測のことである。パターン化不可能な予測のタイプの１つが、非順次予測である。非順次予測が基づく先行アドレスは、即値アドレス、またはトリガアドレスとして構成された任意のアドレスの、いずれかとすることができる。とりわけ、読み取り要求のストリーム内の２つまたはそれ以上のアドレスにわたって１つまたは複数のパターンが欠如していることは、プロセッサが、メモリロケーションの様々な空間的位置からの命令およびデータのフェッチに関してやや散漫な様式でプログラム命令を実行していることを示す。

【００５２】

非順次予測器１０１０は、先行アドレスから非順次予測としてそれぞれ分類可能な１つまたは複数の潜在的な非順次アドレスへの関連を格納するためのリポジトリとして、ターゲットキャッシュ１０３０を含む。ターゲットキャッシュ１０３０は、迅速な形で非順次予測を生成するために、そのコンテンツと着信する検出されたアドレスとを比較するように設計される。さらにターゲットキャッシュ１０３０は、たとえばキャッシュメモリ内の

ヒットに応答して、それらの非順次予測を優先順位付けするように構成される。あるいは、非順次予測器１０１０は、新しい非順次予測と特定のトリガアドレスとの間の関連を確立する第１のインスタンスを優先順位付けすることができる。「トリガ」アドレスとは、非順次予測器１０１０が非順次予測を生成する元となる検出されたアドレスのことであり、２つの間のパターン化不可能な関連の「ターゲット」と呼ばれる、結果として生じる予測を伴う。本発明の少なくとも１つの実施形態によれば、ターゲットキャッシュ１０３０は、それ以外の場合には複数ポートメモリによって使用される、リソースを節約するための単一ポートメモリとすることができることに留意されたい。

【００５３】

プリフェッチャ１００６が非順次予測器１０１０から予測を発行した後に、非順次予測を使用してメモリにアクセスする。これに応答して、メモリは、予測アドレスに関する参照と共にプリフェッチされたデータを戻し、ここで参照は、予測識別子（「ＰＩＤ」）および対応するターゲットアドレスを含むことができる。その後、マルチレベルキャッシュメモリ１０４０は、プロセッサが要求する時点などまで、戻されたデータを一時的に格納する。以下で説明するように、プロセッサがプリフェッチ済みデータ（すなわち予測情報）を要求する場合、必要であれば非順次予測の優先順位を再調整するために参照が非順次予測器１０１０に送信される。

【００５４】

図１１は、本発明の一実施形態に従った、例示的非順次予測器１０１０を示す図である。非順次予測器１０１０は、ターゲットキャッシュ１１３０によって例示されたリポジトリに動作可能に結合された、非順次予測エンジン（「ＮｏｎＳｅｑ．予測エンジン」）１１２０を含む。さらに非順次予測エンジン１１２０は、予測生成器１１２２および優先順位調整器１１２４を含む。予測生成器１１２２は予測を生成し、ターゲットキャッシュ１１３０に格納されたトリガ－ターゲット関連を管理する。優先順位調整器１３２４は、たとえば、最も新しい正常なターゲットアドレスから、最も古いかまたは正常でないターゲットアドレスへと、トリガ－ターゲット関連を優先順位付けするように動作する。予測生成器１１２２および優先順位調整器１１２４については、それぞれ図１２および１３でより詳細に説明する。

【００５５】

ターゲットキャッシュ１１３０は、各トリガアドレス（「ＴＧＲ」）と１つまたは複数の対応するターゲットアドレス（「ＴＲＴ」）との間の関連を維持する。図１１は、非順次アドレスを関連付けるために使用する多くの方法のうちの１つを示すことに留意されたい。ここでは、ツリー構造が特定のトリガアドレスをその対応するターゲットアドレスに関係付けている。この例では、ターゲットキャッシュ１１３０は、アドレス「Ｂ」、「Ｘ」、および「Ｌ」などの可能な非順次予測のアドレスへの関連を形成する元となる、トリガアドレスとしてのアドレス「Ａ」を含む。これら３つのターゲットアドレスは、それぞれアドレス「Ｃ」および「Ｇ」、「Ｙ」、ならびに「Ｍ」に対するトリガアドレスでもある。特に、予測生成器１１２２が新しいトリガ－ターゲット関連を発見し、その関連をターゲットキャッシュ１１３０に挿入する場合の、ターゲットキャッシュ１１３０の形成および動作について、以下でより詳細に考察する。アドレス「Ａ」が、図１１に示されていないトリガアドレスに対するターゲットアドレスともなり得ることに留意されたい。さらに、図示されていないアドレス間の多くの他の関連も可能である。

【００５６】

図に示されるように、ターゲットキャッシュは、本発明の一実施形態により、幅（「ｗ」）、深さ（「ｄ」）、および高さ（「ｈ」）の少なくとも３つの変数に従って、たとえば非順次予測エンジン１１２０によって構築することができる。幅ｗは、トリガアドレスの予測元とすることができる可能なターゲットの数を設定し、深さｄは、トリガアドレスに関連付けられるレベルの数を設定する。高さｈは、非順次予測を生成するために使用される連続するトリガアドレスの数を設定する。一例として、ｄが深さ「４」を示すものと考えてみる。これは、アドレスＡが第１のレベルであり、アドレスＢが第２のレベルであ

り、アドレスＣおよびＧが第３のレベルであり、アドレスＤが第４のレベルであることを意味する。他の例として、ｗが「２」に設定されるものと考えてみる。これは、３つのアドレス「Ｂ」、「Ｘ」、および「Ｌ」のうちの２つのみがｌｅｇ　０およびｌｅｇ　１として非順次予測に使用され、３つのアドレスすべてが第２のレベルにあることを意味する。特定の実施形態では、変数ｈは、マルチレベルの予測生成を達成するために第１のレベルを超えるレベル数を設定する。

【００５７】

図１１に示されるように、ｈが２に設定されるものと考えてみる。これは、第１のレベルのトリガアドレス（たとえばアドレスＡ）および次に続く第２のレベルのトリガアドレス（たとえばアドレスＢ）という、２つのレベルのトリガアドレスが存在することを意味する。したがって、ｈを２に設定すると、トリガアドレスＡに応答して第１の予測グループが形成される。すなわち、第２のレベルのターゲットアドレスのうちのいずれかが非順次アドレスの１つまたは複数のグループを生成することができる。たとえば、アドレス「Ｂ」、「Ｘ」、および「Ｌ」のうちのいずれかが非順次予測を生成するための基準となることが可能であり、これらのアドレスの数は、非順次予測エンジン１１２０によって定義されたアクティブｌｅｇの数（たとえばｌｅｇ　０からｌｅｇ　２まで）によって選択される。しかしながら、マルチレベル予測生成（およびｈを２に設定すること）に従って、アドレス「Ｂ」、「Ｘ」、および「Ｌ」のそれぞれを、次の下位レベルのターゲットアドレスに基づいて予測の第２グループを生成するための連続するトリガアドレスとすることが可能である。したがって、第３レベルのターゲットアドレスＣおよびＧを使用して、連続するトリガアドレスＢに基づいて追加の非順次予測を生成することができる。同様に、ターゲットアドレスＹおよびＭを使用して、それぞれ連続するトリガアドレスＸおよびＬに基づいて非順次予測を生成することもできる。当業者であれば、前述の３つの変数のうちの１つまたは複数を変更することによって、多くの実施が可能であることを理解されよう。

【００５８】

非順次予測エンジン１１２０は、読み取り要求の例示的アドレス１１０１を受け取るように構成される。図１１は、それぞれが以前に検出されたアドレスにパターン化不可能なアドレスを含む、非順次アドレスストリーム１１０２、１１０４、１１０６、１１０８、および１１１０を概念的に示す。たとえば、ストリーム１１０２は、アドレス「Ａ」を含み、その後にアドレス「Ｂ」が続き、さらにその後にアドレス「Ｃ」が続く。非順次アドレスの場合と同様に、「Ａ」から「Ｂ」を予測するため、および「Ｂ」から「Ｃ」を予測するためのパターンを検出することは、読み取り要求１１０１を監視するだけでは困難である。このため、予測生成器１１２２は、特定のトリガアドレスとそのターゲットアドレスとの間のパターン化不可能な関連の予測を実行可能にするために、ターゲットキャッシュ１１３０のコンテンツを確立する。たとえば、ストリーム１１０２のアドレスＡ（ならびに後続のアドレス）を検出すると、予測生成器１１２２は、ＡからＢへの関連、ＢからＣへの関連、ＣからＤへの関連などの関連を、ターゲットキャッシュ１１３０に追加する。非順次予測エンジン１１２０は、他のストリーム１１０４、１１０６などのアドレスを検出する場合も、同様に実行する。

【００５９】

特定の実施形態によれば、ターゲットキャッシュ１１３０はこれらの関連を、表１１４０、１１５０、および１１６０などの表形式で格納する。これらの表は、それぞれトリガアドレスとターゲットアドレスとを格納するための、トリガ列（「ＴＧＲ」）およびターゲット列（「ＴＧＴ」）を含む。次に、すべてのストリームのアドレス１１０１が、表１１４０、１１５０、および１１６０に格納されるものと考えてみる。表１１４０に示されるように、トリガ－ターゲット関連１１４２、１１４４、および１１４６は、それぞれＡからＢ、ＢからＣ、およびＧからＱへの関連を記述する。他のトリガ－ターゲット関連１１４８は、ＣからＤなどの関連を含む。同様に、表１１５０はＡからＸへの関連を記述するためのトリガ－ターゲット関連１１５２を含み、表１１６０はＡからＬへの関連を記述

するためのトリガーターゲット関連１１６２を含む。

【００６０】

図１１は、表１１４０、１１５０、および１１６０がそれぞれ、同じトリガアドレスに関するターゲットキャッシュ１１３０内の複数のトリガーターゲット関連の相対的位置を記述する、「ｗａｙ　０」、「ｗａｙ　１」、および「ｗａｙ　２」として識別されることを示す。優先順位調整器１１２４は、通常はメモリロケーションに優先順位を割り当てることによって、トリガーターゲット関連に優先順位、すなわち予測を割り当てる。この場合、ｗａｙ　０は最高優先順位に、ｗａｙ　１は第２位優先順位に、という具合に関連付けられる。この例では、表１１４０のトリガーターゲット関連１１４２は、ＡからＢへの関連が、表１１５０のトリガーターゲット関連１１５２であるＡからＸへの関連よりも優先順位が高いことを示す。したがって、ターゲットキャッシュ１１３０がこれらの関連を含んだ後、次に非順次予測エンジン１１２０がアドレスＡを検出した場合、非順次予測エンジン１１２０が１つまたは複数の予測を提供することができる。通常、非順次予測エンジン１１２０は、優先順に生成される非順次予測を生成する。具体的に言えば、非順次予測エンジン１１２０は、優先順位が下位の予測を生成する前に最高優先順位を有する予測を生成する。したがって、非順次予測エンジン１１２０は、優先順位に基づいて構成可能な数の予測を生成することができる。たとえば、非順次予測エンジン１１２０は、予測の数を、ｌｅｇ　０およびｌｅｇ　１（すなわち、トリガーターゲット関連の上位２つ）の２つに制限することができる。これは、何らかのケースで、非順次予測エンジン１１２０が、表の相対的優先順位により、アドレスＸではなくアドレスＢを提供する傾向がより高くなることを意味する。トリガーターゲット関連間での相対的優先順位がまさにそれ、すなわち相対的であることに留意されたい。これは、ターゲットキャッシュ１１３０が、特定のトリガアドレスの最高優先順位関連をたとえばｗａｙ　４に位置付け、第２位の優先順位関連をｗａｙ　９に位置付けることができることを意味する。しかしながら、ターゲットキャッシュ１１３０は、１つのアドレスから、単なるｌｅｇ　０およびｌｅｇ　１を超える任意の量の「ｌｅｇ」を含むことができることに留意されたい。

【００６１】

図１２は、本発明の実施形態に従った、例示的予測生成器１２２２を示す図である。この例では、予測生成器１２２２は予測を生成するため、ならびにその中に格納されたトリガーターゲット関連を管理するために、ターゲットキャッシュ１２３０に結合される。予測生成器１２２２は、インデックス生成器１２０４、タグ生成器１２０６、ターゲット特定器１２０８、およびコンバイナ１２１０を含む。また、予測生成器１２２２は、発見されたトリガーターゲット関連をターゲットキャッシュ１２３０に挿入するための挿入器１２０２も含む。

【００６２】

予測を生成する場合、インデックス生成器１２０４およびタグ生成器１２０６は、他のアドレスに先行するアドレスとすることが可能な第１のアドレス「ａｄｄｒ＿１」を表すために、それぞれインデックスおよびタグを作成するように動作する。インデックス生成器１２０４は、ａｄｄｒ＿１からターゲットキャッシュ１２３０内のメモリロケーションのサブセットにアクセスするための、インデックス「ｉｎｄｅｘ（ａｄｄｒ＿１）」を形成する。典型的には、ｉｎｄｅｘ（ａｄｄｒ＿１）の値は、選択される各ｗａｙの対応する各メモリロケーションを選択する。さらに、タグ生成器１２０６はタグ「ｔａｇ（ａｄｄｒ＿１）」を形成するため、予測生成器１２２２は、ａｄｄｒ＿１に関連付けられたターゲットキャッシュ１２３０内の特定のトリガーターゲット関連にアクセスすることができる。

【００６３】

一例として、ａｄｄｒ＿１が「Ｇ」であると考えてみる。このアドレスの場合、予測生成器１２２２は、そのインデックスに関連付けられたメモリロケーションを選択するためにｉｎｄｅｘ（Ｇ）を生成する。このインスタンスでは、ｉｎｄｅｘ（Ｇ）は値（Ｉ）を有し、これは３である（すなわちＩ＝３）。これは、ｉｎｄｅｘ（Ｇ）を使用して、ｗａ

ｙ（「ｗａｙ　０」）１２４０、ｗａｙ（「ｗａｙ　１」）１２５０から、ｗａｙ（「ｗａｙ　Ｎ」）１２６０について、Ｉ＝３によって識別された各メモリロケーションを選択できることを意味し、ここでＮは、ターゲットキャッシュ１２３０で使用可能なｗａｙの数を表す構成可能な数である。同じアドレスＧについて、タグ生成器１２０６は、Ｇに関連付けられた特定のメモリロケーションを識別するために、アドレスＧのタグをｔａｇ（Ｇ）として作成することになる。したがって、ｉｎｄｅｘ（Ｇ）のインデックスおよびｔａｇ（Ｇ）のタグが与えられた場合、図１２に示されるように、ターゲットアドレスＱおよびＰ（またはその代替表現）は、ｗａｙ　１２４０およびｗａｙ　１２５０内のそれぞれのメモリロケーションから取り出すか、そこに格納することができる。特定の実施形態では、各アドレスは３６ビットからなる。ビット２８：１８はアドレスのタグを表すことが可能であり、ビット１９：９、１８：８、１７：７、またはビット１６：６の任意のグループは、そのアドレスの構成可能なインデックスを表すことが可能である。一実施形態では、アドレスの一部がターゲットアドレスを交互に表す。たとえば、３６ビットターゲットアドレスのビット３０：６はターゲットキャッシュ１２３０のＴＲＴ列内で維持される。ターゲットおよびトリガアドレスの両方の表現が減少すると、必要なハードウェアが減少し、それによって材料、リソースなどに関するコストが削減される。

【００６４】

ターゲット特定器１２０８は、トリガーターゲット関連が特定のトリガに対して存在するかどうかを特定し、存在する場合、そのトリガの各ターゲットアドレスを特定する。引き続き前の例を見ると、ターゲット特定器１２０８は、ｔａｇ（Ｇ）が他のトリガアドレスを表すｉｎｄｅｘ（Ｇ）のタグとマッチするのに応答して、ターゲットアドレスＱおよびＰを取り出す。当業者であれば、よく知られた比較回路（図示せず）が、マッチングタグを識別するために、予測生成器１２２２またはターゲットキャッシュ１２３０いずれかでの実施に好適であることを理解されよう。１つまたは複数のターゲットアドレスが見つかった場合、それらのアドレスがコンバイナ１２１０に渡される。コンバイナ１２１０は各ターゲットアドレス１２１４を、トリガアドレスのインデックスおよびタグからなる予測識別子（「ＰＩＤ」）１２１２に関連付ける。ＰＩＤ　１２１２は、ターゲットアドレスＱおよびＰを予測させるトリガアドレスを識別する。したがって、ＰＩＤ　１２１２を［ｉｎｄｅｘ（Ｇ），ｔａｇ（Ｇ）］として表すことができる場合、予測生成器１２２２によって生成される非順次予測は、基準として［［ｉｎｄｅｘ（Ｇ），ｔａｇ（Ｇ）］，Ｑ］の形を有することになる。予測としてのＱは、［ｉｎｄｅｘ（Ｇ），ｔａｇ（Ｇ）］が関連付けられる場合、「基準予測」とみなされることに留意されたい。したがって、キャッシュメモリにプリフェッチされた予測情報は、ｄａｔａ（Ｑ）＋［［ｉｎｄｅｘ（Ｇ），ｔａｇ（Ｇ）］，Ｑ］として表すことができる。

【００６５】

コンバイナ１２１０は、トリガアドレスに対して非順次的ないくつかの追加予測を生成するための「バッチ」信号１２２６を受け取るように構成することができる。たとえば、バッチ信号１２２６がコンバイナ１２１０に対して、マッチしたターゲットアドレスを含む領域を有する予測のグループとして、「ｎ」個の予測を生成するように指示すると考えてみる。そこで、トリガアドレス「Ｇ」がアドレス「Ｑ」（すなわち、基本アドレスとしてのＱ０）の非順次予測を生成した場合、予測アドレスはＱ０、Ｑ１、Ｑ２、．．．Ｑｂを含むことが可能であり、ここでｂはバッチ信号によって設定される数である。バックセクタまたはブラインドバック順次予測が同時に生成されるいくつかのケースでは、バッチｂがｂ－１に設定できることに留意されたい。したがって、予測アドレスのグループは、Ｑ（－１）、Ｑ０、Ｑ１、Ｑ２、．．．Ｑ（ｂ－１）を含むことになる。予測アドレスのグループ内のそれぞれをＰＩＤ　１２１２に関連付けることもできることに留意されたい。特定の実施形態では、ターゲットアドレス１２１４はトリガアドレスの属性を継承し、ここでこうした属性は、トリガアドレスがコードまたはプログラムデータに関連付けられているかどうか、およびトリガアドレスがプロセッサデマンドアドレスであるか否かを示す。他の特定の実施形態では、グループ内の予測アドレスよりも少ない数をＰＩＤ

１２１２に関連付けることもできる。一例では、ターゲットアドレスＱ０のみがＰＩＤ１２１２に関連付けられ、グループ内の他の１つまたは複数（たとえば、Ｑ（－１）、Ｑ２、Ｑ３など）をＰＩＤ　１２１２に関連付ける必要がない。したがって、トリガアドレスＧに遭遇し、その後にターゲットアドレスＱ０が続く場合、ＰＩＤ　１２１２は非順次予測器に報告される。その後、Ｑ２またはグループのうちの任意の他のアドレスに遭遇した場合、ＰＩＤ　１２１２は報告されない。これにより、ターゲットキャッシュ内の冗長エントリの数が削減される。したがって、関連「Ｇ－＞Ｑ０」のみが格納され、その予測のヒットの結果として再度優先順位付けされる。アドレスストリーム内でアドレスＱ１が検出された場合、非順次予測器は関連「Ｇ－＞Ｑ１」を挿入する必要がなくなる。

【００６６】

次に、ターゲット特定器１２０８が、ａｄｄｒ＿１に関するターゲットアドレスを検出しないと考えてみる。次にターゲット特定器１２０８は、ａｄｄｒ＿１に関するトリガーターゲット関連が存在しない旨を挿入器１２０２に伝える。これに応答して挿入器１２０２は、ａｄｄｒ＿１に関するトリガーターゲット関連を形成し、その関連をターゲットキャッシュ１２３０に挿入する。このように実行するために、挿入器１２０２は第１に、ｔａｇ（ａｄｄｒ＿１）を格納するために使用されるｉｎｄｅｘ（ａｄｄｒ＿１）を使用してメモリロケーションを識別する。挿入器１２０２は、トリガアドレスａｄｄｒ＿１へのターゲットアドレスとして格納するために後続のアドレス「ａｄｄｒ＿２」を受け取るようにも構成される。新しく形成されたトリガーターゲット関連に先立って存在するトリガーターゲット関連がない場合、挿入器１２０２はｔａｇ（ａｄｄｒ＿１）およびａｄｄｒ＿２を、最高優先順位のｗａｙ（すなわちｗａｙ　０）であるｗａｙ　１２４０のＴＲＧ列およびＴＧＴ列にそれぞれ格納する。たとえば、図１１のアドレスストリーム１１０４について考えてみると、このストリームは「Ｙ」の後に「Ｚ」が続く第１のインスタンスを示す。「ｔａｇ（Ｙ）　ｔｏ　Ｚ」のトリガーターゲット関連が存在しないことが特定された後、図１２の挿入器１２０２はｉｎｄｅｘ（Ｙ）で新しいトリガーターゲット関連を格納する。したがって、「ｔａｇ（Ｙ）　ｔｏ　Ｚ」がトリガーターゲット関連１２４２としてｗａｙ　１２４０に格納される。特定の実施形態では、挿入器１２０２は優先順位調整器１３２４から挿入信号（「ＩＮＳ」）１２２４を受け取るが、これについて次に説明する。

【００６７】

図１３は、本発明の実施形態に従った、例示的優先順位調整器１３２４を示す図である。一般に、優先順位調整器１３２４は、最も新しい正常なターゲットアドレスから最も古いかまたは正常でないターゲットアドレスへと、トリガーターゲット関連を優先順位付けするように動作する。たとえば、項目に以前のターゲットが存在しない場合、トリガーターゲット関連には最高優先順位が割り当てられる（すなわち、ｗａｙ　０に格納される）ことになる。さらに、予測されたターゲットアドレスが正常に提供された場合（たとえば、プロセッサによりデータが読み取られ、そのデータが非順次予測に基づいてプリフェッチされた場合）、トリガーターゲット関連に最高優先順位を割り当てることができる。この例では、優先順位調整器１３２４は、とりわけその中に格納されたトリガーターゲット関連を優先順位付けするために、ターゲットキャッシュ１２３０に結合される。優先順位調整器１３２４は、レジスタ１３０２、インデックス復号器１３０８、タグ復号器１３１０、ターゲット特定器１３１８、マッチャ１３１４、および優先順位変更器１３１６を含む。

【００６８】

一般に、優先順位調整器１３２４は、特定アドレスがプロセッサの要求したデータの提供に成功したことを示す、非順次予測器１０１０外部の情報を受け取る。こうした情報は、図１０で説明したマルチレベルキャッシュ１０４０などのキャッシュメモリによって生成することができる。優先順位調整器１３２４は、この情報を「Ｈｉｔ　Ｉｎｆｏ」としてレジスタ１３０２内で受け取る。Ｈｉｔ　Ｉｎｆｏは、少なくともデータ（たとえば、プロセッサによって実際に要求されたプログラム命令および／またはプログラムデータ）

のアドレス１３０４を含む参照である。アドレス１３０４はａｄｄｒ＿２としてラベル付けされている。この参照は、アドレス１３０４に関連付けられたＰＩＤ　１３０６も含む。

【００６９】

インデックス復号器１３０８およびタグ復号器１３１０は、それぞれ、ａｄｄｒ＿２が適切な優先順位レベルを有するかどうかを特定するために、ＰＩＤ　１３０６からｉｎｄｅｘ（ａｄｄｒ＿１）およびｔａｇ（ａｄｄｒ＿１）を抽出する。このように実行するために、優先順位調整器１３２４は、ａｄｄｒ＿２が、ターゲットキャッシュ１２３０内の既存のトリガーターゲット関連のターゲットアドレスであるかどうかを識別する。優先順位調整器１３２４がターゲットキャッシュ１２３０にｔａｇ（ａｄｄｒ＿１）およびｉｎｄｅｘ（ａｄｄｒ＿１）を適用した後、ターゲットキャッシュ１２３０のＴＲＧ列にある任意のマッチングトリガアドレスがターゲット特定器１３１８によって受け取られることになる。ａｄｄｒ＿１に関連付けられた１つまたは複数のターゲットアドレスを検出すると、ターゲット特定器１３１８はそれらのターゲットアドレスをマッチャ１３１４に提供する。

【００７０】

しかしながら、ターゲット特定器１３１８が、トリガーターゲット関連内にターゲットアドレスが存在しない（すなわち、アドレスａｄｄｒ＿１に関連付けられたいかなるａｄｄｒ＿２も存在しない）ことを特定した場合、新しいトリガーターゲット関連を挿入するために、図１２の挿入器１２０２に挿入信号（「ＩＮＳ」）１２２４を送ることになる。挿入信号１２２４は、通常、ａｄｄｒ＿１およびａｄｄｒ＿２などのアドレス情報を含む。典型的には、Ｈｉｔ　ＩｎｆｏのＰＩＤ　１３０６に関するマッチングターゲットアドレスが存在しない状況は、プロセッサが以前に発行された非順次予測でヒットしたことを意味する。しかしながら、その後ターゲットキャッシュ１２３０は、その以前に発行された非順次予測に関する基準を形成したトリガーターゲット関連をパージした。したがって、非順次予測エンジン１０１０は、プロセッサによって正常に使用された非順次アドレスを予測するために再度使用可能なトリガーターゲット関連を挿入、または再挿入することになる。

【００７１】

ターゲット特定器１３１８は、１つまたは複数のターゲットアドレスを検出した場合、検出されたターゲットアドレスをマッチャ１３１４に提供する。マッチャ１３１４は、検出された各ターゲットアドレスを、ａｄｄｒ＿２（すなわちアドレス１３０４）と比較して、ａｄｄｒ＿１に関していくつの関連付けられたターゲットアドレスが存在するか、および、既存の各ターゲットアドレスに関して、対応するトリガーターゲット関連が常駐するｗａｙを特定する。マッチャ１３１４は、その比較の結果を優先順位変更器１３１６に提供し、必要であれば優先順位を修正する。

【００７２】

第１に、１つまたは複数のターゲットアドレスが、トリガアドレスとしてａｄｄｒ＿１を表すＰＩＤ　１３０６（すなわちａｄｄｒ＿１）に関連付けられているとして検出されているが、ａｄｄｒ＿２を含むトリガーターゲット関連は存在しない、というインスタンスを考えてみる。したがって、優先順位変更器１３１６は、最高優先順位を表す位置（たとえばｗａｙ　０）に新しいトリガーターゲット関連を挿入し、同じトリガの既存のトリガーターゲット関連の優先順位を降格することになる。たとえば、図１２に示されるように、「ｔａｇ（Ａ）－ｔｏ－Ｘ」トリガーターゲット関連が最高優先順位を表すメモリロケーションにあり、「ｔａｇ（Ａ）－ｔｏ－Ｌ」関連がより低位の優先順位を有すると考えてみる。次に、ＰＩＤ　１３０６がアドレスＡをａｄｄｒ＿１として表し、ａｄｄｒ＿２はアドレスＢであると想定する。優先順位変更器１３１６は図１３に示されるように、「ｔａｇ（Ａ）－ｔｏ－Ｂ」関連をｗａｙ　０に格納し、他の以前の関連はより優先順位の低い他のｗａｙに格納されるように動作することになる。

【００７３】

第２に、２つのターゲットアドレスがＰＩＤ　１３０６（すなわちａｄｄｒ＿１）に関連付けられているとして検出されているが、２つのトリガーターゲット関連がそれらの優先順位を不適切にスワップされている、というインスタンスを考えてみる。このケースでは、優先順位変更器１３１６は、最高優先順位を表す位置（たとえばｗａｙ　０）に最高優先順位トリガーターゲット関連を挿入し、第２位の優先順位を表す他の位置（たとえばｗａｙ　１）に以前の最高優先順位トリガーターゲット関連を挿入することになる。たとえば、図１２に示されるように、「ｔａｇ（Ｂ）－ｔｏ－Ｇ」トリガーターゲット関連が最高優先順位を表すメモリロケーションにあり、「ｔａｇ（Ｂ）－ｔｏ－Ｃ」関連がより低位の優先順位を有すると考えてみる。次に、ＰＩＤ　１３０６がアドレスＢをａｄｄｒ＿１として表し、アドレスＣはａｄｄｒ＿２であると想定する。優先順位変更器１３１６は図１３に示されるように、「ｔａｇ（Ｂ）－ｔｏ－Ｃ」関連をｗａｙ　０に格納し、他の関連はより優先順位の低いｗａｙ　１に格納されるように動作することになる。この優先順位付けの技法は、少なくとも上位２つの優先順位が、それぞれ、最高優先順位および第２位優先順位としての「ｌｅｇ　０」および「ｌｅｇ　１」として維持される場合に有用であることに留意されたい。

【００７４】

次に、２つのターゲットアドレスがＰＩＤ　１３０６（すなわちａｄｄｒ＿１）に関連付けられているとして検出されており、２つのトリガーターゲット関連にはそれらの優先順位が適切に割り当てられている、というインスタンスを考えてみる。このケースでは、優先順位変更器１３１６は、対応するトリガーターゲット関連が正しいとして、何のアクションも実行しない。

【００７５】

図１４は、本発明の特定の実施形態に従った、非順次予測を形成するために予測器生成器を動作させるための例示的パイプライン１４００を示す図である。図１４では、実線の四角形がステージ中またはステージ間のストレージを表し、破線の四角形が非順次予測器によって実行されるアクションを表す。ステージ０では、図１３のインデックス復号器１３０８およびタグ復号器１３１０の融合とすることが可能なタグ／インデックス組み合わせ生成器１４０２によって、読み取り要求のａｄｄｒ＿１が復号される。一実施形態では、タグ／インデックス組み合わせ生成器１４０２は、ａｄｄｒ＿１をアドレスの第１部分およびアドレスの第２部分に分離するように構成されたマルチプレクサである。第１の部分はｔａｇ（ａｄｄｒ＿１）として１４０６で保持され、第２の部分はｉｎｄｅｘ（ａｄｄｒ＿１）として１４０８で保持される。また、このステージでは、トリガーターゲット関連を記述するデータを取り出すために、ｉｎｄｅｘ（ａｄｄｒ＿１）が１４１０のターゲットキャッシュに印加される。オプションで、ターゲットキャッシュが書き込まれる間、読み取り要求のａｄｄｒ＿１を一時的にバッファ１４０４に格納することができる。

【００７６】

ステージ１では、ｔａｇ（ａｄｄｒ＿１）およびｉｎｄｅｘ（ａｄｄｒ＿１）がそれぞれ１４１２および１４１４で保持されたままである。１４１６で、ターゲットアドレスがターゲットキャッシュから読み取られる。ステージ２では、第１に１４１８でｔａｇ（ａｄｄｒ＿１）とｉｎｄｅｘ（ａｄｄｒ＿１）に関連付けられたタグとをマッチングすることによって、非順次予測エンジンが好適な非順次予測を選択する。１４２０で、非順次予測エンジンは、たとえば最高優先順位のターゲットアドレスを（すなわち最高優先順位のトリガーターゲット関連を格納しているｗａｙから）１４２２でｌｅｇ　０予測キューへ転送するように、および、第２位優先順位のターゲットアドレスを（すなわち第２位優先順位のトリガーターゲット関連を格納しているｗａｙから）１４２４でｌｅｇ　１予測キューへ転送するように、マルチプレクサを構成する。ステージ３では、１４３０で、これら２つの非順次予測がたとえばコンバイナへと出力される。図１４では非順次予測を４つのステージで生成するが、他の実施形態の他の非順次予測パイプラインは、これよりも多いかまたは少ないステージを有することができることに留意されたい。

【００７７】

図１５は、本発明の特定の実施形態に従った、非順次予測を優先順位付けするように優先順位調整器を動作させるための例示的パイプライン１５００を示す図である。実線の四角形がステージ中またはステージ間のストレージを表し、破線の四角形が優先順位調整器によって実行可能なアクションを表す。パイプライン１５００は、トリガーターゲット関連をターゲットキャッシュに挿入し、ターゲットキャッシュ関連の優先順位を変更する、例示的方法を示す。ステージ１は、優先順位調整器が挿入するか優先順位付けするかを決定する。優先順位調整器が挿入を実行しようとする場合、１５０２の読み取り要求のアドレスａｄｄｒ＿１が、このステージ中に１５０６で格納される。このアドレスは、ターゲットアドレス用のトリガアドレスとなる可能性を有する。優先順位調整器が優先順位付けを実行しようとする場合、１５０４で、優先順位調整器は外部ソース（たとえばキャッシュメモリ）からａｄｄｒ＿１アドレスを表すＰＩＤ　１５０８を受け取り、このステージ中に１５１０でアドレスａｄｄｒ＿２も受け取る。

【００７８】

図１４および１５は、１つのレベルの予測を使用して非順次予測を例示することに留意されたい。マルチレベルの予測生成を達成するためには、それぞれのパイプライン１４００および１５００の終わりに、生成された予測を入力アドレスとしてパイプライン１４００および１５００にフィードバックするように、例示的パイプライン１４００および１５００を修正することができる。その後、これらの予測は、他のレベルの予測生成用にキューに入れられる。たとえばＡが検出された場合、ターゲットキャッシュ１１３０はターゲットアドレスＢおよびＸを（たとえば、２つの最高優先順位ｗａｙとして）生成する。その後、連続するトリガアドレスとして、アドレスＢがパイプラインのトップに再入力され、これによってターゲットキャッシュ１１３０はアドレスＣおよびＧを生成する。端的に言えば、複数レベルの予測を実施するために、例示的パイプライン１４００および１５００にフィードバックループを追加することができる。

【００７９】

第１に、ステージ０で、優先順位調整器がトリガーターゲット関連挿入を実行していると考えてみる。このインスタンスでは、ａｄｄｒ＿１がタグ／インデックス組み合わせ生成器１５１４によって復号され、ａｄｄｒ＿２が１５１２からマルチプレクサ１５１６を介して選択される。タグ／インデックス組み合わせ生成器１５１４は、インデックス生成器およびタグ生成器の集合機能を実行する。一実施形態では、タグ／インデックス組み合わせ生成器１５１４は、１５０６または１５０８のいずれかからアドレスを選択するように構成されたマルチプレクサである。このケースでは、タグ／インデックス組み合わせ生成器１５１４は、１５２０でｔａｇ（ａｄｄｒ＿１）として保持される第１のアドレス部分を形成し、１５２２でｉｎｄｅｘ（ａｄｄｒ＿１）として保持される第２の部分を形成する。また、このステージでは、ｉｎｄｅｘ（ａｄｄｒ＿１）が、トリガーターゲット関連を記述するデータを取り出すために、マルチプレクサ１５１８を介して１５２４でターゲットキャッシュに印加される。次に、ステージ０で、優先順位調整器がターゲットキャッシュの優先順位付けを実行していると考えてみる。このインスタンスでは、ａｄｄｒ＿１（またはその代替表現）が１５０８から受け取られ、ａｄｄｒ＿２がマルチプレクサ１５１６を介して１５１０から選択される。その後、タグ／インデックス組み合わせ生成器１５１４は、ＰＩＤ　１５０８から第１および第２の部分を形成する。その後、ＰＩＤ　１５０８から形成されたＩｎｄｅｘ（ａｄｄｒ＿１）が、トリガーターゲット関連を記述するデータを取り出すために、マルチプレクサ１５１８を介して、１５２４でターゲットキャッシュに印加される。ステージ１からステージ３では、優先順位調整器が挿入または優先順位付けを実行しているかどうかにかかわらず、パイプライン１５００は同様に挙動する。

【００８０】

ステージ１では、ｔａｇ（ａｄｄｒ＿１）およびｉｎｄｅｘ（ａｄｄｒ＿１）は、１５３０および１５３２でそれぞれ保持されたままである。１５３４では、ターゲットアドレスがターゲットキャッシュから読み取られる。ステージ２では、優先順位調整器が第１に

ｔａｇ（ａｄｄｒ＿１）をタグとマッチングさせる。１５４０でマッチするタグがない場合、１５４２でマルチプレクサがトリガーターゲット関連を挿入する準備をするように構成される。しかしながら、ターゲットキャッシュのｗａｙからの少なくとも１つのタグが１５４４でマッチした場合、および最高優先順位のトリガーターゲット関連が最高優先順位に対応するｗａｙに常駐していない場合、１５５４でトリガーターゲット関連の優先順位が変更される。これを実行するために、１５５２で、新しいトリガーターゲット関連を優先順位変更または挿入するためのマルチプレクサが選択される。ステージ３では、完全に接続された優先順位変更マルチプレクサが、１５５６からのａｄｄｒ＿２を格納するように構成される。このアドレスは、１５５０で保持されたｉｎｄｅｘ（ａｄｄｒ＿１）によって特定された場合、ステージ０でｗａｙ　０のターゲットアドレスとして書き込まれることになる。図に示されるように、完全に接続された優先順位変更マルチプレクサによって１５６０で特定された他のトリガーターゲット関連も、１５５０で保持されるｉｎｄｅｘ（ａｄｄｒ＿１）を使用して、キャッシュ書き込みデータとして１５２４でターゲットキャッシュに書き込まれる。パイプライン１５００がステージ０に戻った後、優先順位調整器は適宜動作を続行する。

【００８１】

インベントリから予測を発行するための例示的実施形態

図１６は、本発明の特定の実施形態に従った、例示的予測インベントリ１６２０を示すブロック図である。この例で、予測インベントリ１６２０は、プリフェッチャ１６０６内に常駐するように示される。さらにプリフェッチャ１６０６は、１つまたは複数のプロセッサによって少なくともメモリアクセスを制御するように設計された、メモリプロセッサ１６０４内で動作するように示される。プリフェッチャ１６０６は、必要とされる前にプログラム命令およびプログラムデータの両方をメモリ１６１２から「フェッチ」し、その後、プロセッサの要求時に、フェッチしたプログラム命令およびプログラムデータをそのプロセッサ１６０２に提供するように動作する。使用に先立ってフェッチすること（すなわち「プリフェッチ」）によって、プロセッサのアイドル時間（たとえばプロセッサ１６０２がデータ不足である間の時間）が最小化される。プリフェッチャ１６０６は、予測を生成するためのスペキュレータ１６０８および不必要な予測を除去するためのフィルタ１６２２も含む。

【００８２】

フィルタ１６２２は、インベントリフィルタまたはインベントリ後フィルタ、あるいはその両方の表現である。不必要な予測を除去することにより、プリフェッチャ１６０６は、そうでなければ重複する予測を不必要に管理するために使用されることになる計算リソースおよびメモリリソースを保存することができる。インベントリフィルタ（インベントリ前フィルタとしての）が、予測インベントリ１６２０への挿入に先立って不必要な予測を除去するように動作するのに対して、インベントリ後フィルタは、メモリ１６１２への発行に先立って不必要な予測を除去する。インベントリ後フィルタの一例は、図２０に示されている。次に、プリフェッチャ１６０６の動作およびその構成要素について説明する。

【００８３】

動作時に、スペキュレータ１６０８は、メモリ１６１２にアクセスするためのプロセッサ１６０２による要求（「読み取り要求」）について、システムバス１６０３を監視する。プロセッサ１６０２がプログラム命令を実行する場合、スペキュレータ１６０８は、プロセッサ１６０２によってまだ使用されていないプログラム命令およびプログラムデータを含むアドレスに関する読み取り要求を検出する。考察のために、「アドレス」は、一般にメモリ１６１２とキャッシュメモリ（図示せず）との間で転送されるメモリのキャッシュラインまたは単位に関連付けられる。キャッシュメモリとは、予測インベントリの外部にある予測のリポジトリの一例である。キャッシュラインの「アドレス」はメモリロケーションを表すことが可能であり、キャッシュラインはメモリ１６１２の複数のアドレスからのデータを含むことができる。「データ」という用語は、プリフェッチ可能な情報の単

位を表すのに対して、「プログラム命令」および「プログラムデータ」という用語は、それぞれ、プロセッサ１６０２によってその処理中に使用される命令およびデータを表す。したがって、データ（たとえば任意のビット数）は、プログラム命令および／またはプログラムデータを構成する予測情報を表すことができる。

【００８４】

検出された読み取り要求に基づいて、スペキュレータ１６０８は、プロセッサ１６０２によるメモリ１６１２へのアクセスを正確に予測する機会を改善するために多数の予測を生成することが可能であり、それらの多数の予測は冗長予測を含む可能性がある。こうした予測の例には、前方順次予測、逆順次予測、バックブラインド順次予測、バックセクタ順次予測、非順次予測などが含まれる。こうした冗長を除去するために、インベントリフィルタ１６２２は重複予測をフィルタリング除去して、存続している予測を生成し、その後それらが予測インベントリ１６２０に格納される。冗長を除去するために、インベントリフィルタ１６２２は生成された予測をキャッシュ（図示せず）のコンテンツと比較した後、それら予測を予測インベントリ１６２０に挿入する。予測と予測インベントリ１６２０内に常駐する予測との間にマッチが見つかった場合、インベントリフィルタ１６２２はその予測を無効にする。しかしながらマッチが見つからない場合、インベントリフィルタ１６２２は存続している予測を予測インベントリ１６２０に挿入する。新しい予測グループ（すなわち、１つのイベントまたは同じトリガアドレスによって生成された予測）内のいくつかの予測はキャッシュのコンテンツとマッチするが、他の予測はマッチしないケースがあることに留意されたい。この場合、インベントリフィルタ１６２２は、キャッシュ内の予測とマッチする個々の予測を無効にし、マッチしなかった（たとえば「無効」とマーク付けされていない）予測を予測インベントリ１６２０に挿入する。

【００８５】

いったん予測インベントリ１６２０に入ると、予測はインベントリの「アイテム」として維持される。「アイテム」という用語は、予測インベントリ１６２０内に格納される「予測」または「トリガアドレス」（予測を生成する）のいずれかを表す。これらのアイテムは、フィルタリングの目的で後に生成される予測と比較することができる。プリフェッチャ１６０６は、インベントリ内のこれらのアイテムを様々なレートでメモリ１６１２に発行しながら、管理する。発行のレートは予測のタイプ（たとえば、前方順次予測、非順次予測など）、各予測のタイプの優先順位、および以下で説明する他の要因に依存する。

【００８６】

予測が冗長になる可能性のある場合の１つが、プロセッサ１６０２が特定のアドレスに関する実際の読み取り要求を発行し、そのアドレスに関する予測が予測インベントリ１６２０内にすでに存在する場合である。このケースでは、予測はフィルタリング除去（すなわち無効化）され、プロセッサ１６０２の実際の読み取り要求が維持される。これは、順次タイプおよびバックタイプの予測などの場合、特にあてはまる。また、いくつかの予測は、予測インベントリ１６２０がそれらの予測を受け取り、プリフェッチャ１６０６がそれらをメモリ１６１２に発行するまでの時間に冗長となり、プリフェッチャ１６０６はアイテムを発行するのに先立って予測をフィルタリング除去することもできる。これにより、重複時間中に発生する冗長予測の数が再度削減されるが、後に生成される予測は予測インベントリ１６２０内に挿入される。また、冗長予測の数が減少すればするほど、保存されるリソースは多くなる。

【００８７】

プリフェッチャ１６０６が予測インベントリ１６２０から予測を発行した後、メモリプロセッサ１６０４は、メモリバス１６１１を介して残りの（少なくともインベントリ後フィルタによってフィルタリング除去されなかった）予測をメモリ１６１２に移送する。これに応答して、メモリ１６１２は、予測アドレスを参照しながらプリフェッチされたデータを戻す。プリフェッチャ１６０６内に常駐するかまたは常駐しないキャッシュメモリ（図示せず）は、メモリプロセッサ１６０４がそのデータをプロセッサ１６０２に送信するまでなどの間、戻されたデータを一時的に格納する。適切な時点で、メモリプロセッサ１

６０４は、プリフェッチされたデータを、とりわけ待ち時間が最小になるのを保証するために、システムバス１６０３を介してプロセッサ１６０２に移送する。

【００８８】

図１７は、本発明の一実施形態に従った、例示的予測インベントリ１６２０を示す図である。予測インベントリ１６２０は、予測を格納するためのいくつかのキュー１７１０、１７１２、１７１４、および１７１６を含み、キューは、それぞれの予測が発行またはフィルタリング除去されるまで格納するためのバッファまたは任意の同様の構成要素とすることができる。予測インベントリ１６２０は、インベントリマネージャ１７０４および１つまたは複数のキュー属性１７０６も含み、これによってインベントリマネージャ１７０４は、対応するキュー属性１７０６に従ってキューそれぞれの構造および／または動作を構成する。

【００８９】

個々のキューは、予測をアイテムとして維持し、そのすべてが一般に、前方順次予測などの同じ特定タイプの予測のものである。図に示されるように、予測インベントリ１６２０は４つのキュー、すなわち順次キュー（「Ｓキュー」）１７１０、バックキュー（「Ｂキュー」）１７１２、非順次ゼロキュー（「ＮＳ０キュー」）１７１４、および非順次１キュー（「ＮＳ１キュー」）１７１６を含む。順次キュー１７１０は、前方順次予測または逆順次予測のいずれかを含むように構成可能であり、バックキュー１７１２はブラインドバック順次予測またはバックセクタ順次予測のいずれかを含むことができる。考察のために、前方順次予測、逆順次予測などは、まとめて「シリーズタイプ」予測と呼ぶことが可能であり、ブラインドバック順次予測、バックセクタ順次予測などは、まとめて「バックタイプ」予測と呼ぶことが可能であることに留意されたい。

【００９０】

予測インベントリ１６２０は、「０番目」の非順次キューおよび「１番目」の非順次キューを含む。非順次（「０」）キュー１７１４および非順次（「１」）キュー１７１６は、それぞれ「最高」および「第２位」の優先順位を有する非順次予測を含む。特に、非順次０キュー１７１４は、対応するトリガアドレスによって生成可能な（任意数のターゲットアドレスのうちの）最高優先順位のターゲットアドレスを含む、非順次予測を維持する。「トリガ」アドレスとは、スペキュレータ１６０８の予測生成元となる検出されたアドレスのことである。こうした予測（すなわち予測アドレス）は、ターゲットを生成するトリガでパターン化不可能（たとえば非順次）な、「ターゲット」アドレスである。同様に、非順次１キュー１７１６は非順次予測を維持しないが、代わりに対応するトリガアドレスによって生成可能な第２位の優先順位のターゲットアドレスを含む。

【００９１】

各キューは、グループ０、１、２、および３などの任意数のグループ１７２０からなるものとすることができる。各グループ１７２０は、トリガアドレス、およびトリガアドレスが生成する対応する予測などの、構成可能な数のアイテムを含む。たとえば、順次キュー１７１０のグループ１７２０は、それぞれトリガアドレスおよび７つの順次予測を含むことが可能であり、バックキュー１７１２のグループ１７２０は、それぞれトリガアドレスおよび１つのバックタイプ予測を含む（または、場合によってはこれらのキューはアイテムとしての予測のみを含む）ことが可能である。さらに、非順次０キュー１７１４または非順次１キュー１７１６、あるいはその両方の、グループ１７２０は、トリガアドレスと、４つの非順次予測のグループとを含む（または、場合によってはアイテムとしての予測のみを含む）ことが可能である。特定の実施形態では、スペキュレータ１６０８は、特定数の予測を生成するためにその「バッチ」数を設定することによって、予測インベントリ１６２０に格納されたグループ１７２０あたりのアイテム数を決定する。予測をグループ化されたアイテムとして予測インベントリ１６２０に格納することによって、グループ１７２０は、通常は各予測を個別に管理するために使用される情報の量を削減し、これによって予測を発行する場合のアービトレーションが容易になる。

【００９２】

インベントリマネージャ１７０４は、各キュー内のアイテムのインベントリを管理するように、ならびにキューの構造および／または動作を制御するように、構成される。予測インベントリ１６２０を管理するために、インベントリマネージャ１７０４は全体としてまたは部分的に、１つまたは複数のキュー属性１７０６を使用してこれを実行する。キュー属性の第１の例は、キューのタイプである。たとえば、キュー１７１０から１７１６のうちのいずれかを、先入れ先出し（「ＦＩＦＯ」）バッファ、後入れ先出し（「ＬＩＦＯ」）バッファ、または任意の他のタイプのバッファとなるように構成可能である。ＦＩＦＯまたはＬＩＦＯなどのキューのタイプは、キューに関してアイテムを挿入および除去する方法に影響を与える。一実施形態では、順次キュー１７１０はＬＩＦＯとして構成され、非順次０キュー１７１４および非順次１キュー１７１６はそれぞれＦＩＦＯとして構成される。

【００９３】

キュー属性の第２の例は、キュー、グループ、またはアイテムに割り当て可能な満了時間または存続時間である。この属性は、予測に関する陳腐化の程度を制御する。任意のグループ１７２０またはキュー内の予測が古くなる、または陳腐になるにつれて、正確な予測を反映する可能性が次第に少なくなる。したがって、古くなったアイテムを最小にするために、インベントリマネージャ１７０４は、グループがその現在のインベントリをある満了時間まで維持できるようにし、その時間が過ぎると、インベントリマネージャ１７０４は古くなったグループ全体またはまだ発行されていない任意の残りのアイテムのいずれかをパージする。本発明の一実施形態では、キュー、グループ、またはアイテムの存続時間を、それらを無制限に保持するように構成することができる。すなわち、それらを「不滅」として設定することが可能であり、これは、不滅の予測が発行されるまでまたは不滅性が撤回されるまで、キュー内に常駐することを意味する。特定の実施形態では、グループがキューに挿入される場合、満了時間はそのグループに関連付けられる。その後、ゼロに達した場合、そのグループの残りのアイテムが無効化されるように、タイマが満了時間からカウントダウンする。他の実施形態では、非順次予測が発行され、その結果としてデータキャッシュ内でヒットすることになる確率を上げるために、非順次０キュー１７１４または非順次１キュー１７１６のいずれかのグループ１７２０に関する満了時間が、順次キュー１７１０のグループ１７２０よりも長く設定される。

【００９４】

キュー属性の第３の例は、キューが満杯の場合、インベントリマネージャ１７０４が予測をどのようにキューに挿入するかを示すために、キューに関連付けられた挿入インジケータである。１つのインスタンスでは、挿入インジケータは、インベントリマネージャ１７０４が新しく生成された予測を挿入されないようにするか、または特定のキューに常駐する古いアイテムを上書きするかを示す。挿入インジケータが「ドロップ」された場合、インベントリマネージャ１７０４は、そうでなければ挿入されることになる任意の新しい予測を廃棄する。挿入インジケータが「上書き」された場合、インベントリマネージャ１７０４は特定のキューが対応するキューのタイプに応じて、２通りのアクションのうちの１つを実行する。キューがＬＩＦＯとして構成された場合、インベントリマネージャ１７０４は事実上新しい予測をスタックとしてＬＩＦＯに押し入れ、これが最も古いアイテムおよび／またはグループをＬＩＦＯの一番下から押し出す。しかしながら、キューがＦＩＦＯとして構成された場合、新しい予測はＦＩＦＯ内の最も古いアイテムを上書きする。

【００９５】

キュー属性の第４の例は、次のアイテムの発行元となる特定のキューを決定するために、それぞれのキューに関連付けられた優先順位である。一実施形態では、優先順位は、次の予測を選択するためにキュー間でアービトレーションするためのキュー１７１０、１７１２、１７１４、および１７１６のそれぞれに関して設定される。シリーズタイプの予測がより多量に生成される応用例では、順次キュー１７１０を処理することが重要である。したがって、このキューは通常比較的高い優先順位に関連付けられる。たとえば、これは非順次０キュー（「ＮＳ０キュー」）１７１４および非順次１キュー（「ＮＳ１キュー」

）１７１６が、順次キュー１７１０に比べて低い優先順位に設定される確率が最も高いことを意味する。キュー属性の他の例は、どれだけの予測を一時的に格納できるかを特定するために各キューに関連付けられるキューサイズである。たとえば、順次キューは２つのグループのサイズまたは深さを有することが可能であり、バックキューは１つのグループの深さを有することが可能であり、非順次キューは４つのグループの深さを有することが可能である。キューサイズは、異なるタイプの予測にどれだけのインベントリメモリが割り当てられるかを制御することによって、プリフェッチャ１６０６によって発行される予測の数を制御することができることに留意されたい。

【００９６】

本発明の一実施形態によれば、バックキュー１７１２の優先順位は、順次キュー１７１０のそれよりも高くなるように、動的に促進または修正することが可能である。この特徴は、スペキュレータ１６０８が上位または「フロント」セクタを検出した後に、メモリ１６１２から予測情報を取り出す際のものである。これは、プロセッサ１６０２が、キャッシュラインの上位またはフロントセクタを要求した直後に、より下位または「バック」セクタを要求する可能性が高いためである。したがって、バックキュー１７１２の優先順位を上げることによって、特に、バックセクタ順次予測を維持している場合、プリフェッチャ１６０６が適切なバックセクタ順次予測をメモリ１６１２に発行することになる確率が増加する。特定の実施形態では、バックキューカウンタ（図示せず）が、バックキュー１７１２以外のキューから発行されるアイテムの数をカウントする。このカウンタがしきい値に達すると、バックキュー１７１２は少なくとも順次キュー１７１０よりも高い優先順位へと促進される。その後、アイテム（たとえばバックセクタアイテム）をバックキュー１７１２から発行することができる。少なくとも１つのバックタイプアイテムを発行するかまたはバックキュー１７１２が（たとえば、古くなることまたはすべてのアイテムを発行することによって）空になった後、バックキュー１７１２の優先順位はその初期の優先順位に戻り（または逆戻りし）、バックキューカウンタがリセットされる。

【００９７】

一般に、予測の非順次グループのうちの任意のグループ１７２０について、非順次予測に関するターゲットアドレスとして、シリーズタイプおよびバックタイプの予測の混合が存在する可能性がある。特に、非順次アドレスのグループは、シリーズタイプ（すなわち、前方または逆のいずれか）の予測のみを含むことができる。しかしながらそれらのグループは、バックタイプと混合されたいくつかのシリーズタイプの予測を含むこともできる。前者の一例として、スペキュレータ１６０８が、トリガアドレス「Ａ」がターゲットアドレス「Ｂ」および他のターゲットアドレス「Ｃ」に関連付けられていることを特定すると考えてみる。ターゲットアドレスＢがＣより高い優先順位の場合、Ｂは非順次０キュー１７１４内に維持されると共に、予測のグループはトリガアドレスＡに対して非順次的である。その後、グループは予測Ｂ０（すなわちアドレスＢ）、Ｂ１、Ｂ２、およびＢ３を含むことが可能であり、そのすべてがアドレスＡに対して非順次的であるが、すべて前方シリーズタイプである。後者の一例として、グループ１７２０は非順次予測Ｂ（－１）（すなわちアドレスＢ－１）、Ｂ０、Ｂ１、およびＢ２を含むことが可能であり、ここで予測Ｂ（－１）は他のシリーズタイプ予測と混合されたバックタイプ予測である。あるいは、グループ１７２０は、本明細書では具体的に説明しない予測の任意の他の配置構成を含むことができる。ＣはＢよりも２番目に高位の優先順位を有するため、Ｃは非順次予測の同様のグループと共に非順次１キュー１７１６内で維持される。したがって、予測Ｂ０、Ｂ１、Ｂ２、およびＢ３を非順次０キュー１７１４のグループ３として挿入することが可能であり、予測Ｃ０、Ｃ１、Ｃ２、およびＣ３を非順次１キュー１７１６のグループ３として挿入することができる。

【００９８】

図１７は、一実施形態で、予測インベントリ１６２０が予測パスを存続させるインベントリフィルタ１７０２を介して予測１７０１を受け取るように構成されることも示す。その後、存続する予測が適切なキューに挿入され、前述のようにインベントリマネージャ１

７０４によって管理される。次に、例示的なインベントリフィルタ１７０２について説明する。

【００９９】

図１８は、本発明の特定の実施形態に従った、インベントリフィルタ１７０２の例を示す図である。この例は、図１７の順次キュー１７１０などの、順次キューに対する前方順次予測のフィルタリングに適用されるが、任意のタイプの予測をフィルタリングするために任意のキューと協働してインベントリフィルタ１７０２を使用することができる。すなわち、インベントリフィルタ１７０２は、任意の予測タイプの任意数の予測を、異なる予測タイプの予測を含む少なくとも１つの他のキューと比較するように構成することができる。たとえば、いくつかの前方順次予測をバックキューなどに対してフィルタリングすることができる。インベントリフィルタ１７０２は、グループ内のアイテム１８０６およびいくつかの予測１８０２をマッチングするために少なくともマッチャ１８０４を含む。グループ１８０６は、アイテムＡ１からＡ７を含み、そのそれぞれがアイテムＡ０に関連付けられる。Ａ０は、以前にアイテムＡ１からＡ７として識別された予測を生成したトリガアドレスである。また、グループ１８０６は、順次キュー１７１０内の任意のグループ１７２０として常駐することができる。予測１８０２の数に関しては、これらはトリガアドレスとしての「ＴＡ」および予測Ｂ１からＢ７を含み、そのすべてがアドレスＴＡの検出時にスペキュレータ１６０８によって生成されたものである。図１８は１つのグループ（すなわちグループ１８０６）のみを示しているが、同じキューの他のグループ１７２０が同じ方法および同時にフィルタリング可能であることに留意されたい。

【０１００】

特定の実施形態では、マッチャ１８０４は、ＣＭＰ０、ＣＭＰ１、ＣＭＰ２、．．．ＣＭＰＭ（図示せず）として識別されるいくつかのコンパレータからなる。コンパレータＣＭＰ０は、ＴＡとグループ１８０６内のＮ個のアイテムとを比較し、コンパレータＣＭＰ１、ＣＭＰ２、．．．ＣＭＰＭはそれぞれ予測１８０２からの予測とグループ１８０６内のＮ個のアイテムのうちのいくつかとを比較するように構成され、ここでＭは生成された最大数の予測を収容するように設定される。一例として、Ｍは７であり、それによって７つのコンパレータが必要であり、Ｎは３であり、その結果各コンパレータは１８０２内の１つの要素と１８０６内の３つのアイテムとを比較すると考える。さらに、予測１８０２の各要素が、同じ位置を有する対応するアイテムとマッチングされる（たとえば、１番目と１番目、２番目と２番目など）と考える。したがって、ＣＭＰ０はＴＡとＡ０、アイテムＡ１、およびアイテムＡ２とを比較し、ＣＭＰ１は予測Ｂ１とアイテムＡ１、Ａ２、およびＡ３とを比較するという具合である。数字Ｎは、コンパレータハードウェアの量を最小にするように、ただし、連続するストリームと、システムバス１６０３上で検出されたアドレスのストリーム内の小さな（すなわちＮより大きくない）ジャンプから生じる可能性のあるそれらの予測とを十分にフィルタリング除去するように、設定可能である。

【０１０１】

一実施形態では、キューは、Ａ０を表すためのページアドレスを格納し、アイテムＡ１、アイテムＡ２などを表すそれぞれをオフセットする。このケースでマッチが存在するかどうかを判別するために、アドレスＴＡのページアドレスおよび予測１８０２からの特定の予測のオフセットは、それぞれＡ０のページアドレスおよび対応するオフセットと比較される。本発明の特定の実施形態では、インベントリフィルタ１７０２は非順次予測に対して順次予測をフィルタリングしないため、非順次０キュー１７１４または非順次１キュー１７１６とは協働しない。これは、順次予測に存在するほど多くの冗長が非順次スペキュレーションにはない可能性が高いためである。

【０１０２】

図１９Ａおよび１９Ｂは、本発明の特定の実施形態に従って冗長をフィルタリング除去する例示的技法を示す図である。マッチャ１８０４がマッチを特定すると、新しく生成された予測（すなわち、新しいアイテムＫ）または以前に生成されたアイテム（すなわち古いアイテムＫ）のいずれかが無効化される。図１９Ａは、新しいアイテムＫまたは古いア

イテムＫのいずれがフィルタリング除去または無効化されるのかを示す。このケースでは、キュー１９０２はＦＩＦＯである。したがって、新しいアイテムＫが無効化されることになり、これによって古いアイテムＫは維持される。これに対して、図１９Ｂは、キュー１９０４がＬＩＦＯの場合、古いアイテムＫが無効化されることになり、これによって新しいアイテムＫが維持されることを示す。一般に、新しいアイテムＫまたは古いアイテムＫのいずれかのうち、最も早く発行したものが維持され、最も新しく発行したものは無効化されることになる。当業者であれば、インベントリフィルタ１７０２は、本発明の範囲および精神を逸脱しない他の技法が採用できることを理解されたい。

【０１０３】

図２０は、本発明の一実施形態に従った、プリフェッチャ内に配置される他の例示的予測インベントリを示す図である。この例では、プリフェッチャ２０００はスペキュレータ１６０８およびフィルタ２０１４を含む。図２０のプリフェッチャ２０００は、マルチレベルキャッシュ２０２０および予測インベントリ１６２０も含む。ここでマルチレベルキャッシュ２０２０は、第１レベルの戻りデータキャッシュ（「ＤＲＣ１」）２０２２および第２レベルの戻りデータキャッシュ（「ＤＲＣ２」）２０２４からなる。第１レベルの戻りデータキャッシュ２０２２は一般に短期データストアとして説明し、第２レベルの戻りデータキャッシュ２０２４は一般に長期データストアとして説明することができる。マルチレベルキャッシュ２０２０は、メモリ１６１２からプリフェッチされたプログラム命令およびプログラムデータを、プロセッサ１６０２が必要とするまで格納する。マルチレベルキャッシュ２０２０のキャッシュは、新しく生成された予測がマルチレベルキャッシュ２０２０に対してフィルタリング可能なように、プリフェッチされた予測情報を生成した予測への参照も格納する。たとえば、ＤＲＣ１　２０２２およびＤＲＣ２　２０２４は、キャッシュラインまたはメモリの単位に関するデータに加えて、（１）新しい予測に対するフィルタリングに使用される格納済みキャッシュラインに関するアドレス、および（２）キャッシュラインが予測の結果としてキャッシュに入れられた場合のトリガアドレス、という２つのタイプの情報を、参照として格納する。特に、トリガアドレスは、スペキュレータ１６０８内の非順次予測の優先順位をシャッフルするために使用される。

【０１０４】

予測インベントリ１６２０は、生成された予測に対して、アービタ２０１８によって選択されるまでの一時ストレージを提供する。予測インベントリ１６２０内の格納済み予測は、そうでなければ発行されることになる冗長をフィルタリング除去するために使用される。アービタ２０１８は、アービトレーション規則に従って、命令およびデータをプリフェッチするためにどの生成された予測を発行するかを決定するように構成される。一般に、こうしたアービトレーション規則は、予測を発行するために特定のキューを選択する際の基準を提供する。たとえばアービタ２０１８は、部分的または全体として、キューおよび／またはグループ間の相対的優先順位に基づいて、予測を選択および発行する。

【０１０５】

フィルタ２０１４は、キャッシュフィルタ２０１０およびインベントリフィルタ１７０２の、少なくとも２つのフィルタを含む。キャッシュフィルタ２０１０は、新しく生成された予測と、プリフェッチされた命令およびデータをすでにマルチレベルキャッシュ２０２０内に格納させた以前の予測とを、比較するように構成される。したがって、１つまたは複数の新しく生成された予測が、マルチレベルキャッシュ２０２０に関して任意の以前に生成された予測と重複する場合、処理を必要とする予測の数を最小にするために冗長予測は無効とされる。冗長予測（すなわち、余分な不必要な予測）は新しく生成された予測の可能性もあることに留意されたい。インベントリフィルタ１７０２は、新しく生成された予測と、すでに生成されて予測インベントリ１６２０内に格納された予測とを比較するように構成される。一実施形態では、インベントリフィルタ１７０２は、図１８に示された構造および／または機能と同様である。ここでも、１つまたは複数の新しく生成された予測が、予測インベントリ１６２０内に以前に格納された予測と重複する場合、プリフェッチャリソースを解放するために任意の冗長予測を無効化することができる。

【０１０６】
冗長予測の数をさらに減少させるために、プリフェッチャ２０００内にインベントリ後フィルタ２０１６が含められる。プリフェッチャ１６０６が予測インベントリ１６２０から予測を発行した後、または発行する直前に、インベントリ後フィルタ２０１６は、予測インベントリ１６２０がそれらの予測を最初に受け取った時間からアービタ２０１８が発行する予測を選択する時間までの間に発生した、冗長予測をフィルタリング除去する。これらの冗長は、通常、予測インベントリ内のアイテムの同じ予測アドレスを表す予測が、予測インベントリ１６２０からメモリに発行された可能性があるが、いずれの予測情報もまだキャッシュ２０２０に戻されていない（すなわち、フィルタリングの対象となる参照がキャッシュ２０２０内にない）可能性があることによって発生する。一実施形態では、インベントリ後フィルタ２０１６は、図１８に示されたインベントリフィルタ１７０２またはキャッシュフィルタ２００２のいずれかと同じ構造および／または機能とすることができる。

【０１０７】
一実施形態では、インベントリ後フィルタ２０１６は、予測インベントリ１６２０内の各グループ１７２０の各アイテムに関する発行情報を維持する。特にこの発行情報は、特定グループのうちのどのアイテムが発行されるかを示す。しかしながら、インベントリ後フィルタ２０１６は、予測インベントリ１６２０から発行されたアイテムを除去しない。むしろ、入ってくる冗長予測をフィルタリング除去する場合に比較の対象とすることができるため、それらを残しておく。その特定グループ内の各アイテムを発行する場合、発行情報はこれを反映するように更新される。すべてのアイテムが発行されるとグループはパージされ、追加のアイテムを受け入れるためにキューが解放される。

【０１０８】
一実施形態では、アービタ２０１８は、予測インベントリ１６２０の予測の発行に関する何らかの側面を制御することができる。特にアービタ２０１８は、最も有利な予測を発行するために、キュー、グループ、またはアイテム間の相対的優先順位を修正することができる。特定の実施形態では、アービタ２０１８は、メモリ１６１２、キャッシュメモリ２０２０、またはメモリサブシステムの他の構成要素などの、メモリに過度に負荷をかける（すなわち、メモリの過剰利用）多数の予測の生成を抑制するために、相対的優先順位を効果的に修正するように構成される。たとえばアービタ２０１８は、構成可能な負荷しきい値を各キューに割り当てることができる。このしきい値は、特定のキューが予測を発行できる最高レートを示す。この負荷しきい値と、メモリ１６１２から要求された累積作業単位を維持する作業負荷アキュムレータ（図示せず）のコンテンツとが比較される。作業単位とは読み取り、書き込みなどの、メモリ１６１２の任意の要求アクションのことである。メモリ１６１２の追加の作業単位が要求されると、作業負荷アキュムレータ内の値が増加する。しかしながら、時間が経過するにつれて（たとえば、あらゆる一定数のクロックサイクルについて）、その値は減少する。動作時に、アービタ２０１８は各キューの負荷しきい値と作業負荷アキュムレータの値とを比較する。作業負荷値が負荷しきい値を超えると、アービタ２０１８は２つの例示的アクションのうちの１つを実行する。アービタ２０１８は、その特定のキューに関する予測の入手を停止するように、予測インベントリ１６２０に指示することが可能であり、その結果、その中のアイテムが発行されるかまたは古くなる。あるいはアービタ２０１８は、アイテムを上書きすることによって、キューのアイテムを取り除くことができる。作業負荷値が負荷しきい値よりも低くなったことをアービタ２０１８が検出すると、予測の発行にキューを再度使用できるようになる。

【０１０９】
キャッシュメモリ内の予測情報に関して先読みルックアップを実行するための例示的実施形態
図２１は、本発明の特定の実施形態に従った、例示的マルチレベルキャッシュ２１２０を含むプリフェッチャ２１００を示すブロック図である。この例で、マルチレベルキャッシュ２１２０は、キャッシュフィルタ２１１０、第１レベルの戻りデータキャッシュ（「

ＤＲＣ１」）２１２２、および第２レベルの戻りデータキャッシュ（「ＤＲＣ２」）２１２４を含む。キャッシュフィルタ２１１０は、第１レベルのＤＲＣ　２１２２および第２レベルのＤＲＣ　２１２４の両方について、迅速に検査するかまたは「先読みルックアップ」を実行し、どちらのキャッシュにおいても予測アドレスなどの入力アドレスの有無を検出するように構成される。先読みルックアップとは、たとえばマルチレベルキャッシュ２１２０内にいくつかの予測がすでに存在するかどうかを並行して特定するための、キャッシュメモリの検査である。

【０１１０】

予測の有無に応じて、マルチレベルキャッシュ２１２０は、以下で説明するキャッシュポリシー、例に従って、第１レベルのＤＲＣ　２１２２および第２レベルのＤＲＣ　２１２４の両方のコンテンツを管理する。第１レベルのＤＲＣ　２１２２は一般に短期データストアとして説明し、第２レベルのＤＲＣ　２１２４は一般に長期データストアとして説明することが可能であり、これによって第１レベルのＤＲＣ　２１２２内の予測は、プロセッサがそれらの予測を要求しない場合、最終的に第２レベルのＤＲＣ　２１２４に移行される。本発明の実施形態によれば、第１レベルのＤＲＣ　２１２２または第２レベルのＤＲＣ　２１２４のいずれか、あるいは両方が、予測アドレスならびにプロセッサ要求アドレスに基づいて、プリフェッチされたプログラム命令およびプログラムデータを格納することができる。また、キャッシュフィルタ２１１０、第１レベルのＤＲＣ　２１２２、および第２レベルのＤＲＣ　２１２４は、冗長予測を減少させることによって、ならびに、たとえば予測情報のプリフェッチ速度を上げること（たとえば、ページオープン動作を予想すること）によって、プリフェッチされたプログラム命令およびプログラムデータを提供する待ち時間を削減するように協働する。以下の考察はマルチレベルキャッシュメモリ（すなわち複数キャッシュ）に関するが、以下のいずれの例示的実施形態も単一のキャッシュメモリを含むことができることに留意されたい。

【０１１１】

キャッシュフィルタ２１１０は、入力アドレスの範囲と、複数キャッシュが本来階層状である、いくつかの複数キャッシュのそれぞれとを、並行して比較するように構成される。たとえば、第１キャッシュの方がサイズが小さく、予測を比較的短期間格納するように適合されるのに対し、第２のキャッシュはサイズが大きく、予測を第１のキャッシュよりも長い期間格納するように適合される。本発明の一実施形態によれば、さらに第２のキャッシュは、その予測アドレスおよび対応する予測データを第１のキャッシュのみから受け取る。両方のキャッシュを並行して検査するために、特に第２のキャッシュの方が第１のキャッシュより大きい場合、キャッシュフィルタは、キャッシュ内で「ルックアップ」または検査された各アドレスの２つの表現を生成する。第１の表現が第１のキャッシュに使用され、第２の表現が第２のキャッシュに使用される場合、両方のキャッシュを同時に検査することができる。その理由の１つは、小さいキャッシュよりも大きいキャッシュの方が検査を必要とするアドレスおよびエントリが多いことである可能性がある。したがって、両方とも同時に検査される場合、大きい方のキャッシュのアドレスを検査するためには、小さい方のキャッシュよりも効率の良い技法が必要である。以下で論じる照会インターフェースが、これらの機能を実行する。

【０１１２】

プリフェッチャ２１００は、予測を生成するためのスペキュレータ２１０８も含む。具体的に言えば、スペキュレータ２１０８は、前方順次予測、逆順次予測、バックブラインド順次予測、バックセクタ順次予測などの順次予測を生成するための、順次予測器（「ＳＥＱ．予測器」）２１０２を含む。またスペキュレータ２１０８は、非順次予測を形成するための非順次予測器（「ＮＯＮＳＥＱ．予測器」）２１０４も含む。プリフェッチャ２１００は、これらの予測を使用して、メモリ（図示せず）からのプログラム命令およびプログラムデータの両方を「フェッチ」し、その後、プロセッサ（図示せず）が命令またはデータを要求する前に、フェッチされたプログラム命令およびプログラムデータをマルチレベルキャッシュ２１２０内に格納する。使用に先立ってそれらをフェッチすること（す

なわち「プリフェッチ」）によって、プロセッサのアイドル時間（たとえば、プロセッサがデータ不足である間の時間）が最小化される。

【０１１３】

非順次予測器２１０４は、先行アドレスから、それぞれを非順次予測としてみなすことが可能な１つまたは複数の潜在的非順次アドレスへの関連を格納するためのリポジトリとして、ターゲットキャッシュ（図示せず）を含む。ターゲットキャッシュは、迅速な方法で非順次予測を生成するために、そのコンテンツと、入ってくる検出されたアドレスとを比較するように設計され、それによってターゲットキャッシュは、たとえばマルチレベルキャッシュ２１２０におけるヒットに応答して、格納済みの非順次予測を優先順位付けするように構成される。具体的に言えば、マルチレベルキャッシュ２１２０がその要求に応じてプロセッサに予測アドレスを提供する場合、そのアドレスが属する格納済みのトリガーターゲット関連の優先順位が上がる。「トリガ」アドレスとは、非順次予測器２１０４が非順次予測を生成する元となる検出されたアドレスのことであり、結果として、その２つの間に形成されるパターン化不可能な関連の「ターゲット」と呼ばれる予測が生じる。トリガアドレスは、ターゲットアドレスと呼ぶことも可能な、順次予測を生じさせるアドレスと呼ぶことも可能であることに留意されたい。

【０１１４】

プリフェッチャ２１００は、フィルタ２１１４、オプションの予測インベントリ２１１６、オプションのインベントリ後フィルタ２１１７、およびオプションのアービタ２１１８も含む。ここでフィルタ２１１４は、生成された予測と、予測インベントリ２１１６内に常駐する以前に生成された予測とを比較するための、インベントリフィルタ（図示せず）を含むように構成することができる。予測インベントリ２１１６は、アービタ２１１８がメモリにアクセスするための予測を選択するまで、生成された予測を格納するための一時ストレージを提供する。アービタ２１１８は、命令およびデータをプリフェッチする場合に、生成された予測のうちのどの予測がメモリにアクセスするために発行されるかを決定するように構成される。いくつかの実施形態では、フィルタ２１１４は、生成された予測と、プログラム命令およびプログラムデータをすでにマルチレベルキャッシュ２１２０内に「プリフェッチ」された状態にした以前に生成された予測とを比較するように構成可能な、キャッシュフィルタ２１１０を含むことが可能である。したがって、任意の生成された予測が、マルチレベルキャッシュ２１２０に格納された任意の以前に生成された予測と重複する場合、管理が必要な予測の数を最小にするようにその冗長予測は無効（または無効化）とすることが可能であり、それによってプリフェッチャリソースが解放される。

【０１１５】

動作時に、プロセッサがメモリへのアクセスを要求した（読み取り要求）場合、スペキュレータ２１０８はシステムバスを監視する。プロセッサがプログラム命令を実行すると、スペキュレータ２１０８は、プロセッサによってまだ使用されていないプログラム命令およびプログラムデータを含むアドレスに関して、読み取り要求を検出する。考察のために、「アドレス」は、一般にメモリと、マルチレベルキャッシュ２１２０などのキャッシュメモリとの間で転送される、メモリのキャッシュラインまたは単位に関連付けられる。キャッシュラインの「アドレス」はメモリロケーションを表すことが可能であり、キャッシュラインはメモリの複数のアドレスからのデータを含むことができる。「データ」という用語は、プリフェッチ可能な情報の単位を表すのに対して、「プログラム命令」および「プログラムデータ」という用語は、それぞれ、プロセッサによってその処理中に使用される命令およびデータを表す。したがって、データ（たとえば任意のビット数）は、プログラム命令またはプログラムデータあるいはその両方を構成する情報を示す「予測情報」を表すことができる。また「予測」という用語は、「予測アドレス」という用語と同じ意味で使用することもできる。予測アドレスがメモリへのアクセスに使用される場合、典型的には、その予測アドレスならびに他の（予測されるかまたはされない）アドレスを含む１つまたは複数のキャッシュラインがフェッチされる。

【０１１６】

プリフェッチャ２１００は予測を発行する場合、各予測への参照を添付または関連付けることができる。予測が非順次予測である場合、それらに関連付けられる参照は予測識別子（「ＰＩＤ」）および対応するターゲットアドレスを含むことができる。ＰＩＤ（図示せず）は、対応するターゲットアドレスを予測させるトリガアドレス（またはその表現）を識別する。この参照は、メモリがプリフェッチされたデータを戻す場合、マルチレベルキャッシュ２１２０によって受け取られる。その後、マルチレベルキャッシュ２１２０は、プロセッサが要求するまでなどの間、戻されたデータを一時的に格納する。マルチレベルキャッシュ２１２０は、プリフェッチされたデータを格納する間、生成された予測に対してフィルタリングするため、その中に格納されたデータの一貫性を確認するため、そのデータを短期または長期データとして分類するためなどに、データを管理する。しかしながら、プロセッサがプリフェッチされたデータ（すなわち予測情報）を要求すると、そのデータはプロセッサに送信される。マルチレベルキャッシュ２１２０内に配置されたデータが非順次予測の結果である場合、必要であればターゲットキャッシュ内に格納された非順次予測の優先順位を再調整するために、非順次予測器２１０４に参照を送信することが可能である。

【０１１７】

図２２は、本発明の一実施形態に従った、例示的マルチレベルキャッシュ２２２０を示す図である。マルチレベルキャッシュ２２２０は、キャッシュフィルタ２２１０、第１レベルの戻りデータキャッシュ（「ＤＲＣ１」）２２２２、および第２レベルの戻りデータキャッシュ（「ＤＲＣ２」）２２２４を含む。キャッシュフィルタ２２１０は、それぞれ、第１レベルＤＲＣ　２２２２および第２レベルＤＲＣ　２１２４を、プリフェッチャ２１００の構成要素ならびにメモリプロセッサ（図示せず）の構成要素などの他の構成要素とインターフェースさせるための、ＤＲＣ１照会インターフェース２２０４およびＤＲＣ２照会インターフェース２２１４を含む。こうしたメモリプロセッサ構成要素の１つが、図２１の書き戻しキャッシュ２２９０であり、これは、よく知られたキャッシュ方法に従って動作し、それによってキャッシュ内のデータへの修正は、必要となるまでキャッシュソース（たとえばシステムメモリ）にコピーされない。書き戻しキャッシュ２２９０の構造および機能は当分野で周知のものと同様であるため、詳細に論じる必要はない。さらに、ＤＲＣ１照会インターフェース２２０４はＤＲＣ１マッチャ２２０６およびＤＲＣ１ハンドラ２２０８を含み、ＤＲＣ２照会インターフェース２２１４はＤＲＣ２マッチャ２２１６およびＤＲＣ２ハンドラ２２１８を含む。

【０１１８】

第１レベルＤＲＣ　２２２２は、アドレス（たとえば予測アドレス）を格納するためのＤＲＣ１アドレスストア２２３０を含み、ＤＲＣ１アドレスストア２２３０は、データ（すなわち予測情報）およびＰＩＤを格納するＤＲＣ１データストア２２３２に結合される。たとえば、予測アドレス（「ＰＡ」）の結果として生じるプリフェッチされたデータは、ＰＩＤ　２２３２ｂと関連付けられたｄａｔａ（ＰＡ）２２３２ａとして格納することができる。この表記法は、予測アドレスＰＡが予測情報を表すデータのプリフェッチに寄与したことを示す。ｄａｔａ（ＰＡ）２２３２ａがプロセッサによって要求されると、対応する予測アドレスＰＡ、および予測識別子ＰＩＤ　２２３２ｂが、必要であればその予測アドレスの優先順位を修正するために非順次予測器２１０４に送られることになる。予測識別子ＰＩＤ　２２３２ｂは、一般に、ＰＡを生じさせるトリガアドレスを示す情報を含む。プロセッサ要求アドレス（および関連データ）をマルチレベルキャッシュ２２２０に格納することも可能であるため、非順次予測器２１０４によって生成されるＰＡもターゲットアドレスと呼ぶことが可能であることに留意されたい。さらに、ｄａｔａ（ＰＡ）２２３２ａは必ずしもＰＩＤ　２２３２ｂを伴う必要がないことにも留意されたい。

【０１１９】

さらに、ＤＲＣ１アドレスストア２２３０およびＤＲＣ１データストア２２３２はどちらも、その機能および／または構造を管理するＤＲＣ１マネージャ２２３４とも通信的に結合される。第２レベルＤＲＣ　２２２４は、データ２２３２ａおよびＰＩＤ　２２３２

ｂと同様の形でデータを格納するＤＲＣ２データストア２２４２に結合された、ＤＲＣ２アドレスストア２２４０を含む。ＤＲＣ２アドレスストア２２４０およびＤＲＣ２データストア２２４２はどちらも、その機能および／または構造を管理するＤＲＣ２マネージャ２２４６と通信的に結合される。

【０１２０】

本発明の特定の実施形態では、第２レベルＤＲＣ　２２２４は、ＤＲＣ２アドレスストア２２４０とは別の有効ビット２２４４を維持するための、「有効ビット」２２４４のリポジトリも含み、各有効ビットは、格納された予測が有効である（ならびにデータに関するプロセッサ要求を処理するために使用可能である）か、無効である（ならびに使用不可である）かを示す。無効予測を有するエントリは、空エントリとみなすことができる。有効ビット２２２４のビットをアドレスから分離して維持することにより、ＤＲＣ２アドレスストア２２４０が有効ビットを対応するアドレスと共に格納する場合よりも１つまたは複数の有効ビットのリセットまたはセットは計算的な負担が軽くなり、迅速になる。ほとんどの場合、ＤＲＣ１のアドレスに関する有効ビットは、通常、それらのアドレスと共に、またはその一部として格納されることに留意されたい。

【０１２１】

動作時に、ＤＲＣ１照会インターフェース２２０４およびＤＲＣ２照会インターフェース２２１４は、第１レベルのＤＲＣ　２２２２および第２レベルのＤＲＣ　２２２４のコンテンツが、「入力アドレス」として印加される１つまたは複数のアドレスのうちのいずれかを含むかどうかを判別するために、そのコンテンツをそれぞれ検査するように構成される。入力アドレスは、生成された予測としてスペキュレータ２１０８から、書き込みアドレスとして書き戻しキャッシュから、またはマルチレベルキャッシュ２２２０外部にある他の要素からのものとすることができる。本明細書で説明するように、一般に、入力アドレスは、冗長をフィルタリング除去するためにマルチレベルキャッシュ２２２０のコンテンツと比較される、生成された予測である。しかしながら、時に入力アドレスは、データが書き込まれるかまたは今後書き込まれることになるメモリのロケーションを識別する書き込みアドレスである。この場合、マルチレベルキャッシュ２２２０は、メモリ、ＤＲＣ１データストア２２２２、およびＤＲＣ２データストア２２２４の間の一貫性を維持するためのアクションが必要であるかどうかを特定するために、検査される。

【０１２２】

ＤＲＣ１マッチャ２２０６およびＤＲＣ２マッチャ２２１６は、入力／出力ポート（「Ｉ／Ｏ」）２２５０上の１つまたは複数の入力アドレスが、ＤＲＣ１アドレスストア２２３０およびＤＲＣ２アドレスストア２２４０内にそれぞれ常駐しているかどうかを特定するように構成される。ＤＲＣ１マッチャ２２０６またはＤＲＣ２マッチャ２２１６のいずれかが、入力アドレスと、第１レベルのＤＲＣ　２２２２および第２レベルのＤＲＣ　２２２４内の入力アドレスとがマッチすることを検出した場合、ＤＲＣ１　ハンドラ２２０８またはＤＲＣ２ハンドラ２２１８などの関連付けられたハンドラは、冗長予測をフィルタリング除去するか、またはマルチレベルキャッシュ２２２０内のデータがメモリと一貫していることを確認するように動作する。ＤＲＣ１マッチャ２２０６およびＤＲＣ２マッチャ２２１６は、入力アドレスの領域と、第１レベルのＤＲＣ　２２２２および第２レベルのＤＲＣ　２２２４のコンテンツとを並行して（すなわち、マルチレベルキャッシュ２２２０の構造に応じて、１または２サイクル内の動作（たとえばクロックサイクル）、あるいは他の最小数のサイクルなどで、同時に、またはほとんど同時に）比較するように構成可能であることに留意されたい。キャッシュと並行して比較可能な入力アドレスの領域の一例が、アドレスＡ０（トリガアドレス）および予測アドレスＡ１、Ａ２、Ａ３、Ａ４、Ａ５、Ａ６、およびＡ７であり、後者の７つは、順次予測器２１０２によって生成されることが可能である。

【０１２３】

同時に検査される場合、こうした比較を実行するマッチャ２２０６、２２１６は、「先読みルックアップ」を実行していると言われる。いくつかの実施形態では、先読みルック

アップは、プロセッサがアイドル状態の場合、またはプリフェッチャ２１００にデータを要求していない場合に、実行される。ＤＲＣ１マッチャ２２０６およびＤＲＣ２マッチャ２２１６は、機能は同様であるが、それぞれの構造は、ＤＲＣ１アドレスストア２２３０およびＤＲＣ２アドレスストア２２４０とそれぞれ動作するように適合されるため、必ずしも同様の構造ではないことにも留意されたい。ＤＲＣ１マッチャ２２０６およびＤＲＣマッチャ２２１６の例について、それぞれ、本発明の少なくとも１つの特定の実施形態に従った図２３Ａおよび図２４に関連して以下で論じる。

【０１２４】

次に、照会インターフェース２２０４、２２１４がフィルタリング操作を実行している場合の状況について考えてみる。いくつかの入力アドレスをマルチレベルキャッシュ２２２０のコンテンツと比較すること、および、マッチしない入力アドレスを検出することによって、ハンドラ２２０８、２２１８は、マッチしない入力アドレスが、フィルタリングが実行されなかった場合よりも早く、生成された予測として予測情報のフェッチを進行できるようにしながら、マッチした予測（すなわち冗長予測）をフィルタリング除去するための適切なアクションを実行することができる。したがって、マルチレベルキャッシュ２２２０およびそのキャッシュフィルタ２２１０は、どのキャッシュラインがフェッチを開始するかをより迅速に特定することによって、待ち時間を削減する。これにより、第１レベルのＤＲＣ　２２２２および第２レベルのＤＲＣ　２２２４のキャッシュが、一般に、予測が並行して比較されないかまたはフィルタリング除去されない場合、あるいはその両方の場合よりも早く、プリフェッチされた予測情報を含む可能性が高いことから、プロセッサが経験する待ち時間がさらに削減される可能性がある。

【０１２５】

ＤＲＣ１アドレスストア２２３０およびＤＲＣ２アドレスストア２２４０はそれぞれ、ＤＲＣ１データストア２２３２およびＤＲＣ２データストア２２４２に格納されたプリフェッチされたデータに関連付けられたアドレスをそれぞれ格納する。アドレスストア２２３０および２２４０はそれぞれ、アドレスまたはアドレスの代替表現のいずれかを格納する。本発明の一実施形態によれば、例示的ＤＲＣ１アドレスストア２２３０は、完全連想型であり、完全に固有のアドレスを格納するように構成される。たとえば、各アドレスについて、それらのアドレスを固有に識別するためにビット３５：６がＤＲＣ１に格納される。ＤＲＣ１アドレスストア２２３０に格納されたアドレスは、共通部分（たとえばタグ）およびデルタ部分（たとえばインデックス）を含むものとみなすことが可能であり、その両方が、少なくとも１つの実施形態に従ってＤＲＣ１の先読みルックアップ中にアドレスを表すために使用されることに留意されたい。さらに、ＤＲＣ１アドレスストア２２３０およびＤＲＣ１データストア２２３２は、それぞれ、データのアドレスエントリごとに、３２エントリのアドレスおよび６４バイトのキャッシュラインを格納するように構成される。プリフェッチされたデータは、一般に、動的ランダムアクセスメモリ（「ＤＲＡＭ」）などのメモリからのものであるが、ＤＲＣ１データストア２２３２内のデータが更新を必要とする場合は、書き戻しキャッシュからのものとすることが可能である。

【０１２６】

これとは対照的に、例示的ＤＲＣ２アドレスストア２２４０は、４ｗａｙセットの関連エントリからなり、アドレスを表すための基本部分（たとえばタグ）を格納するように構成することができる。さらに、ＤＲＣ２アドレスストア２２４０およびＤＲＣ２データストア２２４２は、それぞれ、データのアドレスエントリごとに、１０２４エントリのアドレスおよび６４バイトのキャッシュラインを格納するように構成される。ＤＲＣ２データストア２２４２は、ＤＲＣ１データストア２２３２からのプリフェッチデータを格納し、いくつかの実施では、任意数のメモリバンク（たとえば４つのバンク：０、１、２、および３）からなるものとすることができる。

【０１２７】

予測情報のプリフェッチ元であるメモリは、通常、ＤＲＡＭメモリ（たとえば、デュアルインラインメモリモジュール、すなわち「ＤＩＭＭ」内に配置構成される）であるが、

メモリは任意の他の知られたメモリ技術のものとすることができる。通常、メモリは、特定の行アドレス内で使用可能なメモリのセクションである「ページ」に細分される。特定のページにアクセスする、すなわち「オープン」すると、他のページがクローズされ、このページのオープンおよびクローズのプロセスは完了するまでに時間を要する。したがって、プロセッサがＤＲＡＭメモリの様々なメモリロケーションからの命令およびデータのフェッチに関して、やや散漫な様式でプログラム命令を実行している場合、メモリへのアクセスは非順次的である。したがって、読み取り要求のストリームはページ領域をまたがって延在する可能性がある。次のページの次のアドレスが使用できない場合、通常、プロセッサはプログラム命令およびプログラムデータをメモリから直接フェッチしなければならない。これにより、こうした命令およびデータの取り出し待ち時間が増加する。そこで、マルチレベルキャッシュ２２２０内の複数ページにまたがる予測情報をプリフェッチおよび格納することによって、ページのオープンに関する待ち時間が本発明に従って削減される。さらに、プリフェッチされているデータがキャッシュからのものであるため、アクセスされたページはオープンされたままで、プロセッサによって認識されるかまたはプロセッサに関する待ち時間が削減される。

【０１２８】

たとえば、非順次予測器２１０４が、アドレス「００１００」のプロセッサ読み取りに続いてアドレス「００２００」にアクセスすることになると、正しく予測するものと考えてみる。したがって、非順次予測器２１０４は、プロセッサが実際にアドレス「００２００」にアクセスするより前に、アドレス「００２００」（ならびに、バッチが４の場合、アドレス００２０１、００２０２、００２０３、および００２０４）で開始するラインの範囲（たとえば、１つのターゲットアドレスおよび４つの予測アドレスであり、生成する予測の数はバッチ「ｂ」によって構成可能かつ定義される）をフェッチさせる。プロセッサが実際にアドレス「００２００」に関する読み取りを実行する場合、マルチレベルキャッシュ２２２０の先読みルックアップは、アドレス「００２００」に続く指定された範囲内で、どのキャッシュラインがすでにプリフェッチされているかを即時に特定する。読み取りアドレスストリーム内の非順次移行は、ＤＲＡＭページオープン動作を伴うことが可能であるため、先読みルックアップは、プリフェッチャ２１００が、読み取り要求のストリーム内で即時に先読みすること、および、どのアドレスまたはキャッシュラインをフェッチする必要があるかを特定することを可能にする。フェッチを即時に開始することにより、プリフェッチャ２１００は、ＤＲＡＭページオープン動作の待ち時間をしばしば隠し、その後、プロセッサ上での待ち時間の損失を招くことなく、キャッシュラインの順次ストリームを提供する（ターゲットアドレスに関する基準を形成するトリガアドレスとは非順次的であるが）ことができる。

【０１２９】

図２２は、ＤＲＣ１マネージャ２２３４およびＤＲＣ２マネージャ２２４６を別のエンティティとして示すが、必ずしもそうである必要はない。すなわち、ＤＲＣ１マネージャ２２３４およびＤＲＣ２マネージャ２２４６は、単一の管理エンティティに組み合わせるか、またはマルチレベルキャッシュ２２２０の外部に配置する、あるいはその両方とすることができる。第１レベルのＤＲＣ　２２２２および第２レベルのＤＲＣ　２２２４は、プロセッサ内に常駐する従来のＬ１およびＬ２キャッシュとは構造的および／または機能的に異なるため、マルチレベルキャッシュ２２２０内に格納された予測情報を管理する固有のポリシーが採用される。こうしたポリシーの例には、各戻りデータキャッシュ内のメモリを割り振るためのポリシー、短期から長期のデータストアへ情報をコピーするためのポリシー、および、マルチレベルキャッシュ２２２０と書き戻しキャッシュなどの他のエンティティとの間の一貫性を維持するためのポリシーが含まれる。

【０１３０】

第１に、情報が短期情報から長期情報へと古くなる場合の、第１レベルのＤＲＣ２２２２から第２レベルのＤＲＣ　２２２４への予測情報のコピーを管理するために使用されるコピーポリシーについて考えてみる。データが、ある一定の時間しきい値まで第

１レベルのＤＲＣ　２２２２にある場合、ＤＲＣ１マネージャ２２３４はＤＲＣ２マネージャ２２４６と協働して、ＤＲＣ１データストア２２３２からＤＲＣ２データストア２２４２へとそのデータを転送する。しきい値は一定であるか、またはそうでなければ動作時に変化してもよいことに留意されたい。通常、ＤＲＣ１内にある無効エントリ（すなわち使用可能）がＮ個未満の場合は必ず、古くなったデータは転送されるように構成され、ここでＮはプログラム可能である。動作時に、データが短期ストレージから長期ストレージへとコピーされた場合、第１レベルのＤＲＣ　２２２２内のエントリは消去（すなわち無効化）される。

【０１３１】

第２に、第１レベルのＤＲＣ　２２２２および第２レベルのＤＲＣ　２２２４に予測情報を挿入するための割り振りポリシーについて考えてみる。予測情報を第１レベルのＤＲＣ　２２２２に挿入する場合、ＤＲＣ１マネージャ２２３４は、候補としてロックされたエントリを除いて、ＤＲＣ１データストア２２３２内の任意の無効なエントリを選択する。ＤＲＣ１マネージャ２２３４が予測情報を格納することが可能ないずれの無効エントリも検出しない場合、最も古いエントリを使用してエントリ用のスペースを割り振ることができる。ＤＲＣ２データストア２２４２におけるエントリの割り振りについても同様に、ＤＲＣ２マネージャ２２４６は、第１レベルのＤＲＣ　２２２２から第２レベルのＤＲＣ　２２２４へとコピーされたデータを受け取るためのいくつかのｗａｙのうちのいずれか（たとえば４ｗａｙのうちの１つ）を使用することができる。たとえば、予測アドレスのインデックスは、データを格納する４つのエントリを含むことができる。初期に、ＤＲＣ２データストア２２４２は、使用されていない（すなわち無効化された）ｗａｙ数のうちのいずれか１つを割り振る。しかしながら、すべてのｗａｙが割り当てられている場合、第１のｉｎが第１のｏｕｔである（すなわち、最も古いものが上書きされる）。しかしながら、最も古いエントリが同じ古さであり、かつ有効な場合、ＤＲＣ２マネージャ２２４６はロックされていないエントリを割り振る。最後に、ｗａｙセット内のすべてのエントリがロックされている場合、ＤＲＣ２マネージャ２２４６は、第１レベルのＤＲＣ　２２２２内のエントリを有効として維持しながら、第１レベルのＤＲＣ　２２２２から第２レベルのＤＲＣ　２２２４への書き込みを抑制する。ここでも、典型的には第２レベルのＤＲＣ　２２２４は、第１レベルのＤＲＣ　２２２２からのみ、ストレージ用のデータを受け取ることに留意されたい。

【０１３２】

ＤＲＣ１マネージャ２２３４およびＤＲＣ２マネージャ２２４６が遵守可能な他のポリシーは、一貫性を維持することに関する。ＤＲＣ１マネージャ２２３４は、データが書き込まれることになる書き込みアドレスとマッチするアドレスを有する、任意のエントリのデータを更新することによって、第１レベルのＤＲＣ　２２２２の一貫性を維持する。典型的には、書き戻しキャッシュ２２９０（図２１）は、書き込みアドレスを書き込むためにメモリ（たとえばＤＲＡＭ）に送信するまで、書き込みアドレス（および対応するデータ）を一時的に格納する。書き戻しキャッシュ２２９０内の書き込みアドレスとマッチする読み取り要求のアドレスがあるいくつかのケースでは、マルチレベルキャッシュ２２２０がデータを第１レベルのＤＲＣ　２２２２に転送するのに先立って、書き込みアドレスのデータとメモリのそれとをマージすることに留意されたい。ＤＲＣ２マネージャ２２４６は、書き戻しキャッシュ２２９０内にロードされる場合に、そのアドレスが書き込みアドレスとマッチする任意のエントリを無効化することによって、第２レベルのＤＲＣ　２２２４の一貫性を維持する。第２レベルのＤＲＣ　２２２４がＤＲＣ１からのデータのみを受け取るため、および、第１レベルのＤＲＣ　２２２２がメモリおよび書き戻しキャッシュ２２９０との一貫性を維持するため、第２レベルのＤＲＣ　２２２４は一般に、陳腐化したデータを含むことがない。さらに、ＤＲＣ１からＤＲＣ２へとコピーされる予定の任意のアドレスを、第１に書き戻しキャッシュ（「ＷＢＣ」）２２９０に照らしてチェックすることができる。ＷＢＣ　２２９０内にマッチが見つかった場合、コピー操作は中止される。見つからなかった場合、そのアドレスのＤＲＣ１からＤＲＣ２へのコピーは実行

される。この追加のチェックが、一貫性の維持にさらに役立つ。

【０１３３】

図２３Ａは、本発明の特定の実施形態に従った、第１のアドレスストア２３０５に関する例示的ＤＲＣ１の照会インターフェース２３２３を示す図である。この例では、トリガアドレス（「Ａ０」）２３００（たとえば、プロセッサ要求アドレス）は、入力アドレスとして、共通アドレス部分２３０２ａおよびデルタアドレス部分２３０２ｂからなる。アドレス２３００は、いくつかのケースでの予測アドレス、または他のケースでの書き込みアドレス（一貫性を維持する場合）の、いずれかとすることも可能であることに留意されたい。アドレス２３００が予測アドレスのグループを生成するトリガアドレスの場合、こうしたグループ２３０７は、アドレス（「Ａ１」）２３０１からアドレス（「Ａｍ」）２３０３まで識別されているようなアドレスを含むことが可能であり、ここで「ｍ」は、本発明の少なくとも一実施形態に従って「先読みルックアップ」を実行する際に使用可能な任意数の予測を表す。いくつかのケースでは、「ｍ」はバッチサイズ「ｂ」と等価に設定される。

【０１３４】

ＤＲＣ１アドレスストア２３０５のエントリ２３０６は、それぞれ、第１のエントリ部分２３０６ａ（たとえばタグ）および第２のエントリ部分２３０６ｂ（たとえばインデックス）を含む。特定の実施形態では、第１のエントリ部分２３０６ａおよび第２のエントリ部分２３０６ｂは、それぞれ、共通アドレス部分２３０２ａおよびデルタアドレス部分２３０２ｂに類似している。第２のエントリ部分２３０６ｂは、アドレスに関して、トリガアドレス（「Ａ０」）２３００からその特定のエントリ２３０６への移動を示す。したがって、ＤＲＣ１マッチャ２３１２が、トリガアドレス（「Ａ０」）２３００などの入力アドレスとエントリ２３０６とを比較する場合、共通部分２３０２ａを使用してグループ２３０７のアドレスの共通部分を表すことができる。さらに、アドレス２３００の共通部分２３０２ａが、一般にアドレス（「Ａ１」）２３０１から（「Ａｍ」）２３０３までの共通部分と同様であるため、エントリ２３０６の１つまたは複数の第１のエントリ部分２３０６ａと比較するために使用する必要があるのは共通部分２３０２ａのみである。また、アドレス（「Ａ１」）２３０１から（「Ａｍ」）２３０３までのデルタ部分２３０２ｂを、エントリ２３０６の複数の第２のエントリ部分２３０６ｂとマッチングさせることも可能である。

【０１３５】

一実施形態では、ＤＲＣ１マッチャ２３１２は、共通アドレス部分と第１のエントリ部分とをマッチングするための共通コンパレータ２３０８、およびデルタアドレス部分と第２のエントリ部分とをマッチングするためのデルタコンパレータ２３１０を含む。具体的に言えば、エントリ０からｎ番目のエントリについて、共通部分２３０２ａと第１の部分２３０６ａとが同時に比較され、同じエントリについて、デルタ部分２３０２ｂと第２の部分２３０６ｂとが同時に比較される。いくつかの実施形態では、共通コンパレータ２３０８は、高位ビット（たとえば、３６ビットアドレスのビット３５：１２）を比較するための「ワイド」コンパレータであり、デルタコンパレータ２３１０は、低位ビット（たとえば、３６ビットアドレスのビット１１：６）を比較するための「ナロー」コンパレータである。図２３Ａは、１つのデルタ部分２３０２ｂにつき１つのデルタコンパレータを示すが、いくつかのケースでは、デルタコンパレータ２３１０の数はｍ*ｎに等しく（図示せず）、ここで各デルタコンパレータは、入力として１つのデルタ部分２３０２ｂおよび１つの第２のエントリ部分２３０６ｂを受け取ることに留意されたい。コンパレータサイズは、これらの比較を実行するために必要な物理リソースの量を制限するため、並行してルックアップされるアドレスは、同じメモリページ内に存在するように構成される（たとえば、メモリページサイズは通常４Ｋバイトである）。これによって、交差するページ境界からの先読みルックアップのアドレスは減少するが、これらの構成は、物理リソースに関して先読みルックアップを実行するためのコストを削減する。ここでも、共通部分２３０２ａおよびデルタ部分２３０２ｂはそれぞれ同時に、またはほぼ同時に、エントリ２３

０６と比較されることに留意されたい。

【０１３６】

共通コンパレータ２３０８およびデルタコンパレータ２３１０の出力は、それぞれ、Ｈｂａｓｅ（０）、Ｈｂａｓｅ（１）、．．．Ｈｂａｓｅ（ｍ）、およびＨ０、Ｈ１、Ｈ２、．．．ＨＮであり、ここでそれぞれは０（たとえばマッチなしを示す）または１（たとえばマッチを示す）のいずれかである。この結果は、フィルタリングしているかまたは一貫性を維持しているかに応じて、アクションを実行するためにＤＲＣ１ハンドラ２３１４に送信される、０および１のヒットベクトルを形成する。ヒットリスト生成器２３１３は、範囲「ｒ」（すなわちグループ２３０７）内のどのアドレスがＤＲＣ１アドレスストア２３０５内に常駐するかを示す、ヒットのリスト（「ヒットリスト」）を生成する。アドレスがマッチした（すなわち、その中に予測が格納されている）場合、そのアドレスはヒットリストに含められ、マッチしないアドレス（すなわち予測が格納されていない）はヒットリストから除外される。このヒットリストは、予測を生成するため、またはＤＲＣ１アドレスストア２３０５内の一貫性を管理するために使用される。

【０１３７】

図２３Ｂは、特定の実施形態に従った、図２３ＡのＤＲＣ１照会インターフェース２３２３を使用して並行して検査可能な任意数の例示的入力アドレス２３５２を示す図である。ここでＤＲＣ１照会インターフェース２３５０は、ＤＲＣ１アドレスストア２３０５とマッチングするために、任意の範囲のアドレス２３５２を受け入れることができる。図２３Ａのマッチャ２３１２は、いくつかの入力アドレスにわたって、並行先読みルックアップを実行するために必要な回数だけ複製される。一例として、バッチサイズ「ｂ」が２７に設定された前方順次予測の場合、ＤＲＣ１照会インターフェース２３５０はマッチャに、基本（またはトリガ）アドレスとしてのＡ０と、グループ２３０７としての予測アドレスＡ１からＡ７とを、並行してマッチさせるように要求する。ブラインドバック予測の場合、Ａ（－１）のみが、グループ２３０７として基本アドレスＡ０以外のマッチングを必要とするが、逆順次予測の場合、アドレスＡ（－１）からＡ（－７）がマッチングを必要とする。アドレス２３５２の範囲は、ＤＲＣ１およびＤＲＣ２の両方の照会インターフェースにも同時に、並行して印加できることに留意されたい。

【０１３８】

図２４は、本発明の特定の実施形態に従った、ＤＲＣ２アドレスストア２４０４に関する例示的ＤＲＣ２照会インターフェース２４０３を示す図である。ＤＲＣ２照会インターフェース２４０３は、ＤＲＣ２アドレスストア２４０４のコンテンツとアドレスを比較するために入力アドレス２４０２を受け取るように構成される。この例で、入力アドレス２４０２は、ｔａｇ（Ａ０）などのアドレスの基本部分（たとえばタグ）である。さらにこの例を見ると、ＤＲＣ２アドレスストア２４０４は、バンク０、１、２、および３というメモリの４つのバンク２４０６からなり、それぞれのバンクがエントリ２４１０を含んでいる。このケースでは、エントリ２４１０を４つのｗａｙ（Ｗ０、Ｗ１、Ｗ２、およびＷ３）のうちのいずれか１つに配置できることに留意されたい。

【０１３９】

ＤＲＣ２マッチャ２４３０は、ｔａｇ（Ａ０）をエントリ２４１０と比較するためのいくつかのコンパレータを含む。一般に、ＤＲＣ２アドレスストア２４０４内の任意のマッチングアドレスは、同じｔａｇ（Ａ０）を共有するが、他のビットグループとの関係は異なる（たとえばインデックスごと）場合がある。本発明の特定の実施形態では、タグがＤＲＣ２アドレスストア２４０４内のいずれかのエントリとマッチするかどうかの特定が、一般に以下のように実行される。第１に、各バンク２４０６について、そのバンク内のインデックスのうちの１つが、潜在的マッチングアドレスを探索するために選択される。これは、図２５Ａに示されるように、バンクが特定アドレス（たとえばＡ０）のあるインデックスビットによって識別できる場合、探索用に選択されるバンクは、特定のアドレス（たとえば図２５のＡ０）がどのバンクに常駐するかによって異なるため、バンクごとに異なる場合がある。第２に、各バンク２４０６について選択されたインデックスの４つのｗ

ａｙすべてにアクセスする。次に、４つのｗａｙ（たとえばＷ０からＷ３）に関係して格納されたタグが、この例では基本アドレス２４０２であるｔａｇ（Ａ０）と比較される。一般に、ｔａｇ（Ａ１）などの他のタグと比較することなく、ｔａｇ（Ａ０）と比較するだけで十分である。これは、これらのタグが一般に等しい（たとえば、ｔａｇ（Ａ０）＝ｔａｇ（Ａ１）＝ｔａｇ（Ａ２））と想定されるためである。予測に関する同時探索は、通常、４ｋバイトページなどの同じページ内にある予測に限定され、これによってタグが同じになることに留意されたい。第３に、ＤＲＣ２マッチャ２４３０によってアドレスマッチが実行されると、ヒットベクトルおよび有効ビットの形の結果を使用して、図２７および２８に関連して説明するのと同様に、最終ヒットベクトルが取得される。

【０１４０】

ＤＲＣ２照会インターフェース２４０３のヒット生成器２４４２は、タグの比較結果（「ＴＣＲ」）２４２２をＤＲＣ２マッチャ２４３０から受け取り、さらにそれらの結果を対応する有効ビット２４５０と比較して、順序付けされた予測のセット（「順序付けされた予測」）を生成する。ここで、バンク１、２、３、および４からのタグの比較結果は、それぞれＴＣＲ（ａ）、ＴＣＲ（ｂ）、ＴＣＲ（ｃ）、ＴＣＲ（ｄ）とラベル付けされ、それぞれが、タグが１つまたは複数のエントリ２４１０とマッチするかどうかを表す１つまたは複数のビットを含む。順序付けされた予測は、入力アドレス２４０２とマッチする（またはマッチしない）予測の順序付けされたセットとすることができる。または、順序付けされた予測それぞれを、入力アドレスがＤＲＣ２アドレスストア２４０４内に存在するアドレスを有するかどうかを表す、ビットのベクトルとすることができる。追加のＤＲＣ２マッチャ２４３０が含まれる場合、任意数の入力アドレス２４０２が同様にＤＲＣ２照会インターフェース２４０３とマッチングできることに留意されたい。図２５Ａから２８は、本発明のいくつかの実施形態に従った例示的ヒット生成器を示す図である。

【０１４１】

図２５Ａは、本発明の一実施形態に従った、ＤＲＣ２アドレスストア２４０４に格納されたアドレス（またはその表現）の可能な配置構成を示す図である。以下の考察を簡単にするために、ｗａｙ　Ｗ０、Ｗ１、Ｗ２、およびＷ３は示されていないことに留意されたい。入力アドレスＡ０、Ａ１、Ａ２、およびＡ３は、ＤＲＣ２アドレスストア２４０４に格納される。一例として、順次予測器２１０２（図示せず）は、トリガアドレスＡ０（たとえば４つのｗａｙのうちのいずれかにある）に基づいて、順次予測Ａ１、Ａ２、およびＡ３を生成することができる。第１の配置構成２５０２は、Ａ０がバンク０に格納された結果である。同様に、第２の配置構成２５０４、第３の配置構成２５０６、および第４の配置構成２５０８は、それぞれアドレスＡ０をバンク１、２、および３に格納した結果であり、後続のアドレスは続くトリガアドレス内に順番に格納される。したがって、これらのアドレス（または、タグの形などのその一部）は、一般に、特定の順序のないＤＲＣ２アドレスストア２４０４からの出力である。

【０１４２】

図２５Ｂは、本発明の実施形態に従った、順序付けされていないアドレスおよび対応する有効ビットに基づいて結果を生成する、例示的ヒット生成器２４３０を示す図である。この例では、順次予測器２１０２がトリガアドレスＡ０に基づいて順次予測Ａ１、Ａ２、Ａ３、Ａ４、Ａ５、Ａ６、およびＡ７を生成し、そのすべてが、図に示された特定の配置構成で格納される（すなわち、トリガアドレスＡ０がバンク１に格納され、その他がそれに続く）。ヒット生成器２４３０は、順序付けされていないアドレスＡ２、Ａ６、Ａ１、Ａ５、Ａ０、Ａ４、Ａ３、Ａ７、および順序付けされた有効ビットＶＢ０からＶＢ７を受け取り、それらを順序付けし、それらを比較した後、ビットベクトルまたはアドレスのリスト（マッチするかまたはマッチしないのいずれか）とすることが可能な、結果Ｒ０からＲ７を生成する。予測が無効であることを示す有効ビットは、格納された無効予測がマッチングされないようにすることに留意されたい。これは、有効ビットをアドレスストアのコンテンツとマッチングするための１つの理由である。本発明の特定の実施形態によれば、アドレスＡ２、Ａ１、Ａ０、およびＡ３、またはアドレスＡ６、Ａ５、Ａ４、およびＡ

７のいずれかなどの、８つではなく４つのアドレスが同時に考慮される。したがって、図２５Ｂに示されるように、アドレスＡ０からＡ７を「折り返し」様式で連続してアクセスできるように表す必要はない。しかしながら、図２５ＢのアドレスＡ０からＡ７を同時に考慮するためには、ＤＲＣ２を、同じＲＡＭ（または同じＤＲＣ２）への２つの独立した同時アクセスを実行するための２重ポートのランダムアクセスメモリ（「ＲＡＭ」）として構成することができる。

【０１４３】

図２６は、図２４のヒット生成器２４４２に関するヒット生成器２６００を示す概略図である。ヒット生成器２６００は、ｗａｙ０から３からのアドレスおよび／または各入力アドレスの有効ビットを多重化することによって、結果Ｒ０からＲ７のうちの１つまたは複数を生成し、ここで結果Ｒは、アドレスまたは有効ビットの多重化されたビットを比較することによって決定される。有効ビットが、対応するタグ比較結果（「ＴＣＲ」）によって示されたタグが有効であることを示す場合、そのタグは結果Ｒとして出力される。ＴＣＲをアドレスのタグとするか、あるいは「１」（すなわちＤＲＣ２内でヒット）または「０」（すなわちＤＲＣ２内でヒットせず）のいずれかの値を有するビットとすることができることに留意されたい。図２７および２８に関連して以下で説明するように、アドレスに関するタグ（たとえばｔａｇ（Ａ１））は、一般に、そのタグの単一のＴＣＲビットを表す。

【０１４４】

図２７は、本発明の一実施形態に従った、ヒット生成器２４４２の一例を示す図である。ヒット生成器２４４２は、それぞれｗａｙバンク０、１、２、および３からのアドレスＡ３、Ａ０、Ａ１、およびＡ２に関する順序付けされていないタグを順序付けするように構成された、順序付け器２７０２を含む。しかしながら、アドレスＡ３、Ａ０、Ａ１、およびＡ２に関するタグは、それぞれ、各タグに関するＴＣＲを表す単一のビットを表すことに留意されたい。次に、順序付けされたＴＣＲ（アドレスＡ０、Ａ１、Ａ２、およびＡ３に関する順序付けされたタグとして示される）が、有効ビット２２４４からの有効ビットＶＢ０～ＶＢ３に照らしてテストされる。ＡＮＤ演算子（「ＡＮＤ」）２７０６は、このテストを論理ＡＮＤ関数として実行する。したがって、有効ビットがｔｒｕｅであり、単一ビットのＴＣＲがｔｒｕｅの場合、ヒットが存在し、結果Ｒがこれを反映する。すなわち、結果Ｒ０、Ｒ１、Ｒ２、およびＲ３が順序付けされた予測結果を形成し、これらは、ここでもマッチ／マッチせずを表すビットとするか、あるいは、アドレスに関するマッチしたタグまたはマッチしないタグとすることができる。タグそれ自体がＴＣＲとして使用される（たとえば、ＴＣＲ（ａ）の場合Ｔａｇ（Ａ３））ケースでは、ＡＮＤ演算子２７０６は、対応する有効ビットがゼロの場合それらのビットをマスクする（たとえば、結果Ｒは、その対応する有効ビットがゼロの場合、すべてゼロを含むことになる）ように動作することに留意されたい。

【０１４５】

図２８は、本発明の他の実施形態に従った、ヒット生成器２４４２の他の例を示す図である。ヒット生成器２４４２は、有効ビット２２２４からの順序付けされた有効ビットＶＢ０～ＶＢ３の順序を解除するように構成された、有効ビット（「ＶＢ」）順序付け器２８０２を含む。すなわち有効ビット順序付け器２８０２は、順序ＶＢ０、ＶＢ１、ＶＢ２、およびＶＢ３から、アドレスＡ３、Ａ０、Ａ１、およびＡ２に関するタグによって表されるＴＣＲの順序とマッチする順序ＶＢ３、ＶＢ０、ＶＢ１、およびＶＢ２へと、有効ビットを再順序付けする。次に、アドレスに関する順序付けされていないタグ（すなわち、それらのタグに関する順序付けされていないＴＣＲ）が、ＡＮＤ演算子（「ＡＮＤ」）２８０６によって、同様に順序付けされた有効ビットに照らしてテストされる。順序付けされていない結果Ｒ３、Ｒ０、Ｒ１、およびＲ２は、結果順序付け器２８１０を通過して順序付けされた予測結果としてＲ０、Ｒ１、Ｒ２、およびＲ３を取得するが、これはプリフェッチャ２１００、およびフィルタリング、一貫性などを実行するその要素によって使用可能な形である。有効ビットおよび結果（単なる結果ビットの場合がある）を再順序付け

することによって、それぞれいくつかのビットからなるアドレスを再順序付けするよりも、必要なハードウェアが少なくてすむ。順序付け器２７０２および結果順序付け器２８１０の順序付けは例示的なものであり、ビットを順序付けおよび再順序付けするための他のマッピングも本発明の範囲内であることに留意されたい。

【０１４６】

本発明の特定の実施形態では、非順次予測器２１０４およびマルチレベルキャッシュ２１２０を含む図２１のプリフェッチャ２１００は、ノースブリッジチップの同じ機能のうちの少なくともいくつかを有するメモリプロセッサ内などの、ノースブリッジ－サウスブリッジチップセットアーキテクチャ内に配置される。メモリプロセッサは、ＣＰＵ、グラフィックスプロセッサユニット（「ＧＰＵ」）などの１つまたは複数のプロセッサによって、少なくともメモリアクセスを制御するように設計される。ノースブリッジの実施では、プリフェッチャ２１００はＡＧＰ／ＰＣＩ　Ｅｘｐｒｅｓｓインターフェースを介しＧＰＵに結合することもできる。さらに、プロセッサとメモリとの間のシステムバスとして、フロントサイドバス（「ＦＳＢ」）を使用することもできる。また、メモリはシステムメモリとすることもできる。別法として、メモリプロセッサが実行するのと同様にメモリへのアクセスを制御する働きをする、任意の他の構造、回路、デバイスなどで、マルチレベルキャッシュ２１２０を採用することができる。さらに、マルチレベルキャッシュ２１２０およびその要素、ならびにプリフェッチャ２１００の他の構成要素は、ハードウェアまたはソフトウェアモジュールのいずれか、あるいはその両方からなるものとすることが可能であり、さらに、任意の様式で分散または結合することも可能である。

【０１４７】

説明のために、前述の記述では、本発明を完全に理解するための特定の命名法を使用した。しかしながら、当業者であれば、本発明を実施するために特定の細部が必要でないことを理解されよう。したがって、本発明の特定の実施形態の前述の記述は、例示および説明のために提示されたものである。本発明を網羅するか、または開示された精密な形に限定することは意図されておらず、前述の教示に鑑みて、多くの修正形態および変形形態が可能であることは明らかである。実際、この記述は、本発明のいずれかの特徴または態様をいずれかの実施形態に限定するものとして読むべきではなく、むしろ一実施形態の特徴および態様は他の実施形態と容易に交換可能である。諸実施形態は、本発明の原理およびその実際の応用例を最もよく説明するために選択および説明されたものであり、それによって他の当業者が、企図された特定の使用に好適となるような様々な修正形態によって、本発明および様々な実施形態を最適に利用できるようにするものである。添付の特許請求の範囲およびそれらの等価物が本発明の範囲を画定することが意図されている。

【図面の簡単な説明】

【０１４８】

【図１】本発明の特定の実施形態に従った、メモリプロセッサと共に実施される例示的スペキュレータを示すブロック図である。

【図２】本発明の一実施形態に従った、例示的スペキュレータを示す図である。

【図３Ａ】本発明の特定の実施形態に従った、例示的前方順次予測器を示す図である。

【図３Ｂ】本発明の特定の実施形態に従った、例示的ブラインドバック順次予測器を示す図である。

【図３Ｃ】本発明の特定の実施形態に従った、例示的バックセクタ順次予測器を示す図である。

【図３Ｄ】本発明の特定の実施形態に従った、例示的逆順次予測器の挙動を示す図である。

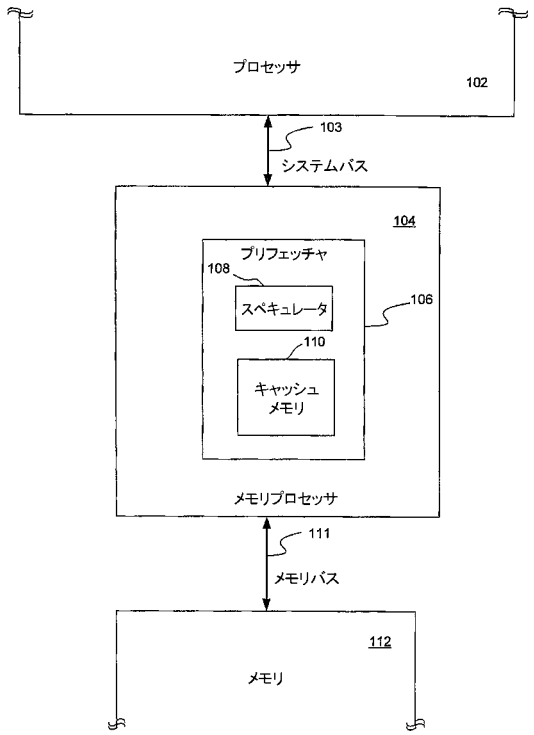
【図４】本発明の一実施形態に従った、例示的非順次予測器を示す図である。

【図５】本発明の一実施形態に従った、インタリーブされた順次アドレスのストリームに対して非順次予測を抑制する例示的技法を示す図である。

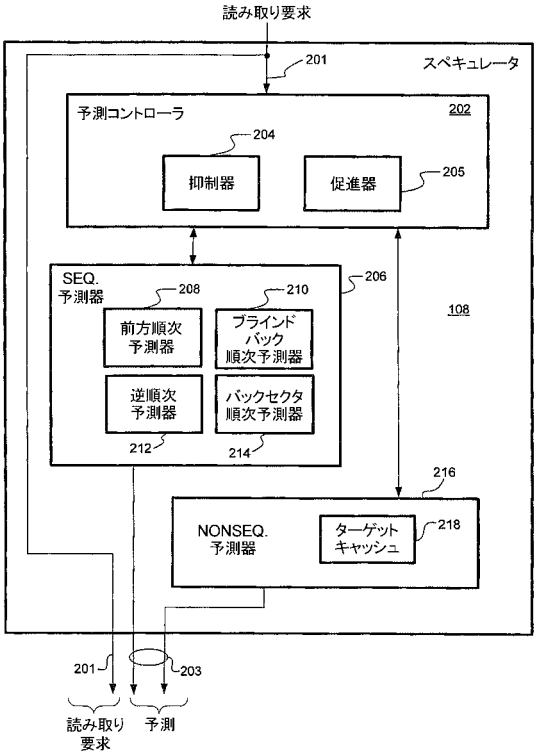
【図６】本発明の一実施形態に従った、複数のスレッドにわたってインタリーブされた順次アドレスに対して非順次予測を抑制する例示的技法を示す図である。

【図７】本発明の特定の実施形態に従った、基本アドレスおよび非順次アドレスの着信時間に基づいて非順次予測を抑制するための他の技法を示す図である。
【図８】本発明の特定の実施形態に従った、予測の生成を促進するための例示的技法を示す図である。
【図９】本発明の一実施形態に従った、予測フィルタを含む他の例示的スペキュレータを示す図である。
【図１０】本発明の特定の実施形態に従った、例示的非順次予測器を実施するプリフェッチャを示すブロック図である。
【図１１】本発明の一実施形態に従った、例示的非順次予測器を示す図である。
【図１２】本発明の実施形態に従った、例示的予測生成器を示す図である。
【図１３】本発明の特定の実施形態に従った、例示的優先順位調整器を示す図である。
【図１４】本発明の特定の実施形態に従った、非順次予測を形成する場合に非順次予測器生成器を動作させるための例示的パイプラインを示す図である。
【図１５】本発明の特定の実施形態に従った、非順次予測を優先順位付けするように優先順位調整器を動作させるための例示的パイプラインを示す図である。
【図１６】本発明の特定の実施形態に従った、メモリプロセッサ内の例示的予測インベントリを示すブロック図である。
【図１７】本発明の一実施形態に従った、例示的予測インベントリを示す図である。
【図１８】本発明の特定の実施形態に従った、インベントリフィルタの例を示す図である。
【図１９Ａ】本発明の特定の実施形態に従った、冗長をフィルタリング除去する例示的技法を示す図である。
【図１９Ｂ】本発明の特定の実施形態に従った、冗長をフィルタリング除去する例示的技法を示す図である。
【図２０】本発明の一実施形態に従った、プリフェッチャ内に配置される他の例示的予測インベントリを示す図である。
【図２１】本発明の特定の実施形態に従った、例示的キャッシュメモリを含むプリフェッチャを示すブロック図である。
【図２２】本発明の一実施形態に従った、例示的マルチレベルキャッシュを示す図である。
【図２３Ａ】本発明の特定の実施形態に従った、第１のアドレスストアに関する例示的第１の照会インターフェースを示す図である。
【図２３Ｂ】図２３Ａの第１の照会インターフェースを使用して並行して検査可能な任意数の入力アドレスを示す図である。
【図２４】本発明の特定の実施形態に従った、第２のアドレスストアに関する例示的第２の照会インターフェースを示す図である。
【図２５Ａ】本発明の一実施形態に従った、第２のアドレスストアに格納される場合の例示的アドレス（またはその表現）の可能な配置構成を示す図である。
【図２５Ｂ】本発明の実施形態に従った、順序付けされていないアドレスおよび順序付けされた有効ビットに基づいて結果を生成する、例示的ヒット生成器を示す図である。
【図２６】本発明の実施形態に従った、図２５のヒット生成器の１つの結果、Ｒを生成するための構成要素を示す概略図である。
【図２７】本発明の特定の実施形態に従った、ヒット生成器の一例を示す図である。
【図２８】本発明の他の実施形態に従った、ヒット生成器の他の例を示す図である。

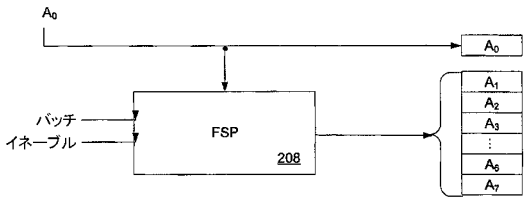
【図１】


プロセッサ
102
103
システムバス
104
プリフェッチャ
108
スペキュレータ
106
110
キャッシュ
メモリ
メモリプロセッサ
111
メモリバス
112
メモリ

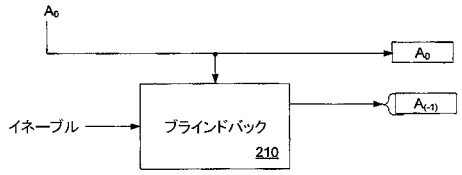
【図２】


読み取り要求
201
スペキュレータ
予測コントローラ
202
204
抑制器
促進器
205
SEQ.
予測器
208
210
206
前方順次
予測器
ブラインド
バック
順次予測器
108
逆順次
予測器
バックセクタ
順次予測器
212
214
216
NONSEQ.
予測器
ターゲット
キャッシュ
218
201
203
読み取り
要求
予測

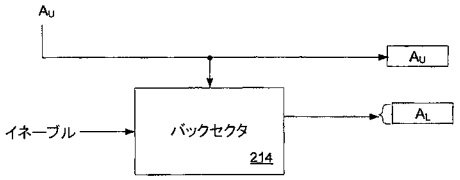
【図３Ａ】


A0
A0
A1
A2
A3
A6
A7
バッチ
イネーブル
FSP
208

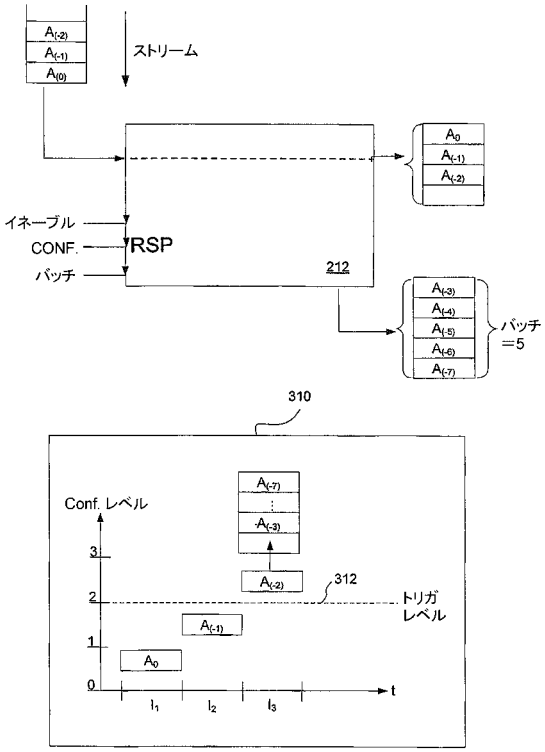
【図３Ｂ】


A0
A0
A(-1)
イネーブル
ブラインドバック
210

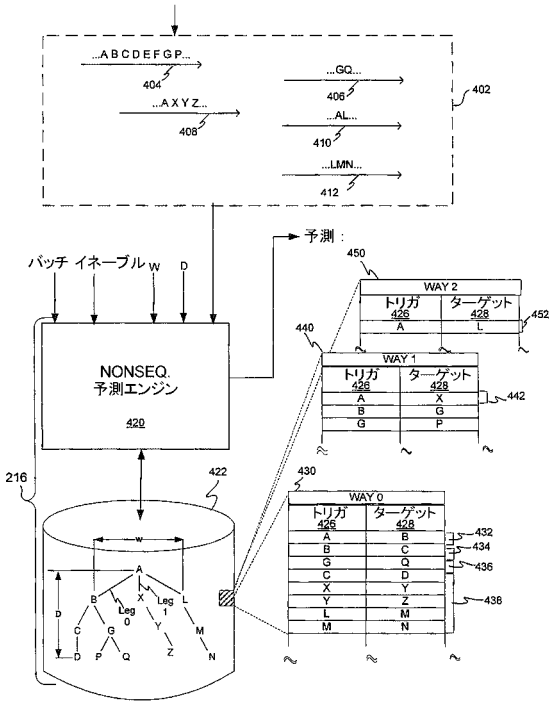
【図３Ｃ】


A_U
A_U
イネーブル
バックセクタ
214
A_L

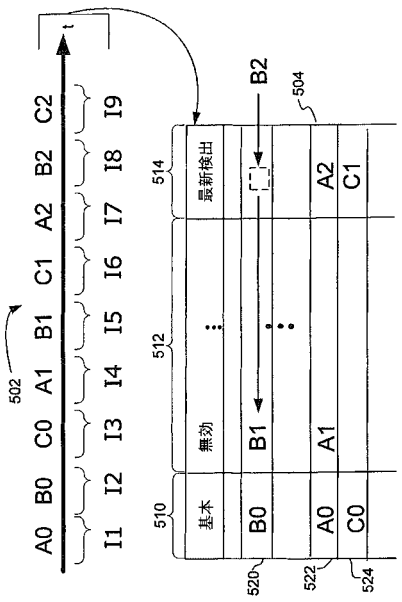
【図３Ｄ】


$A_{(-2)}$
$A_{(-1)}$
$A_{(0)}$
ストリーム
A_0
$A_{(-1)}$
$A_{(-2)}$
イネーブル
CONF.
バッチ
RSP
212
$A_{(-3)}$
$A_{(-4)}$
$A_{(-5)}$
$A_{(-6)}$
$A_{(-7)}$
バッチ
=5
310
Conf. レベル
$A_{(-7)}$
$A_{(-3)}$
$A_{(-2)}$
312
トリガ
レベル
$A_{(-1)}$
A_0
3
2
1
0
t
I_1
I_2
I_3

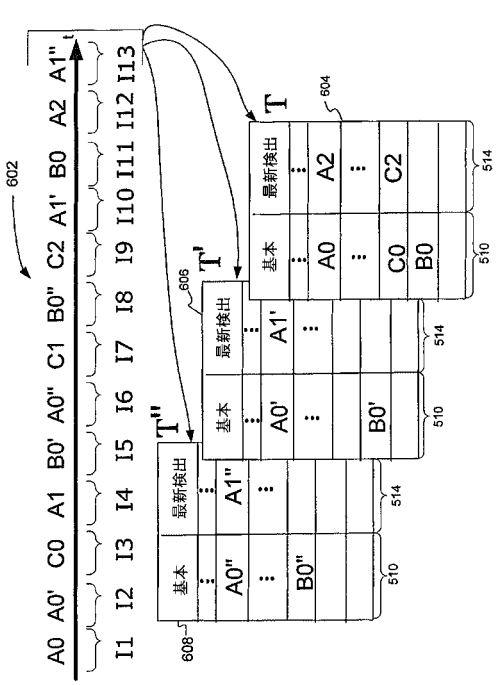
【図４】


...ABCDEFGP...
404
...AXYZ...
408
...GQ...
406
...AL...
410
...LMN...
412
402
予測：
バッチ イネーブル W D
NONSEQ.
予測エンジン
420
216
422
450
WAY 2
トリガ
426
ターゲット
428
A
L
452
440
WAY 1
トリガ
426
ターゲット
428
A
B
G
G
X
P
442
430
WAY 0
トリガ
426
ターゲット
428
A
B
B
C
G
Q
C
D
X
Y
Y
Z
L
M
M
N
432
434
436
436
w
D
A
B
X
Leg
0
Leg
1
L
C
G
Y
M
D
P
Q
Z
N

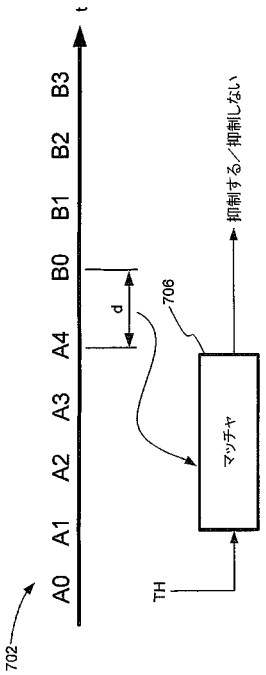
【図５】


t
A0 B0 C0 A1 B1 C1 A2 B2 C2
I1 I2 I3 I4 I5 I6 I7 I8 I9
502
B2
504
514
最新検出
512
無効
510
基本
B1
A1
C1
A2
B0
A0
C0
520
522
524

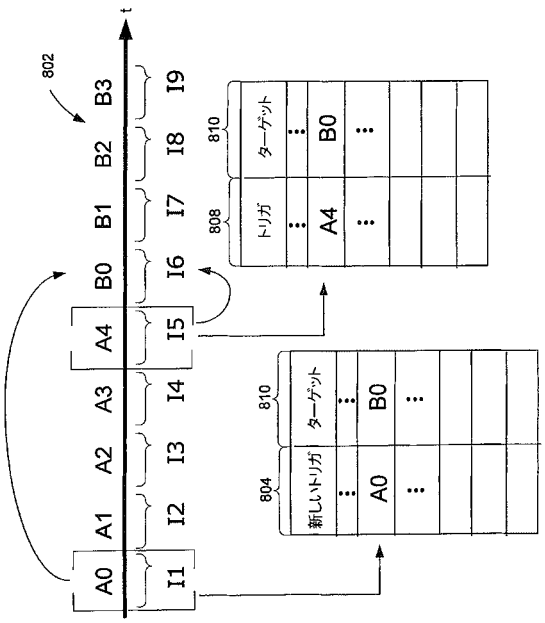
【図６】


602
A0 A0' C0 A1 B0' A0" C1 B0" C2 A1' B0 A2 A1"
I1 I2 I3 I4 I5 I6 I7 I8 I9 I10 I11 I12 I13
t
T
T'
T"
604
606
608
基本
最新検出
510
514

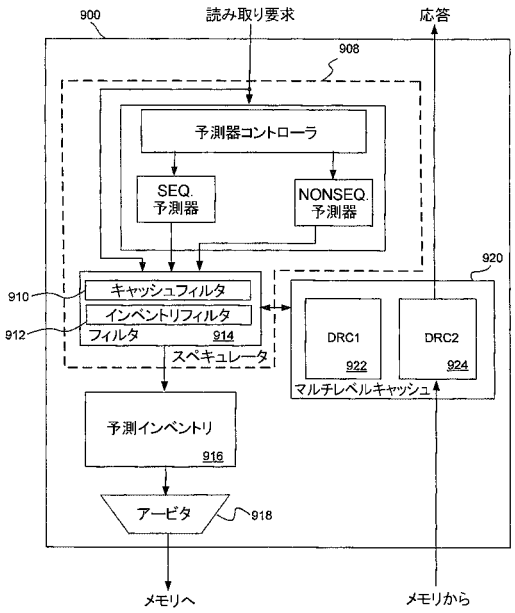
【図７】


A0 A1 A2 A3 A4 B0 B1 B2 B3
t
d
706
マッチャ
抑制する／抑制しない
TH
702

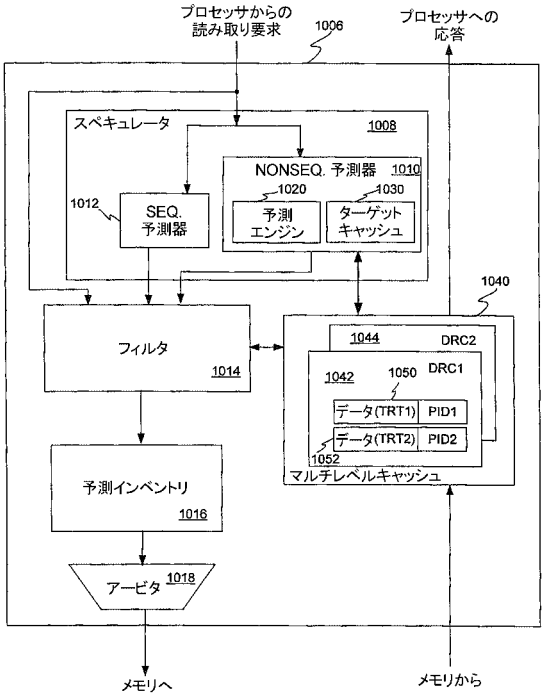
【図８】


802
A0 A1 A2 A3 A4 B0 B1 B2 B3
I1 I2 I3 I4 I5 I6 I7 I8 I9
t
トリガ
ターゲット
新しいトリガ
ターゲット
808
810
804
810

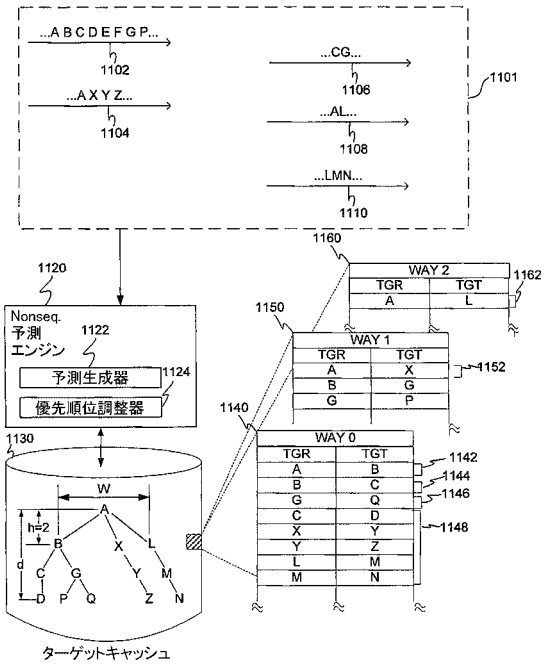
【図９】


900
読み取り要求
応答
908
予測器コントローラ
SEQ.
予測器
NONSEQ.
予測器
910
キャッシュフィルタ
912
インベントリフィルタ
フィルタ
914
スペキュレータ
920
DRC1
922
DRC2
924
マルチレベルキャッシュ
予測インベントリ
916
アービタ
918
メモリへ
メモリから

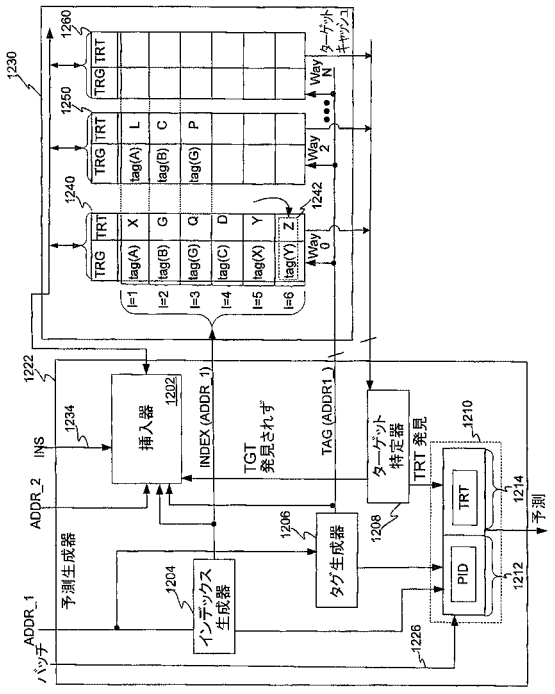
【図１０】



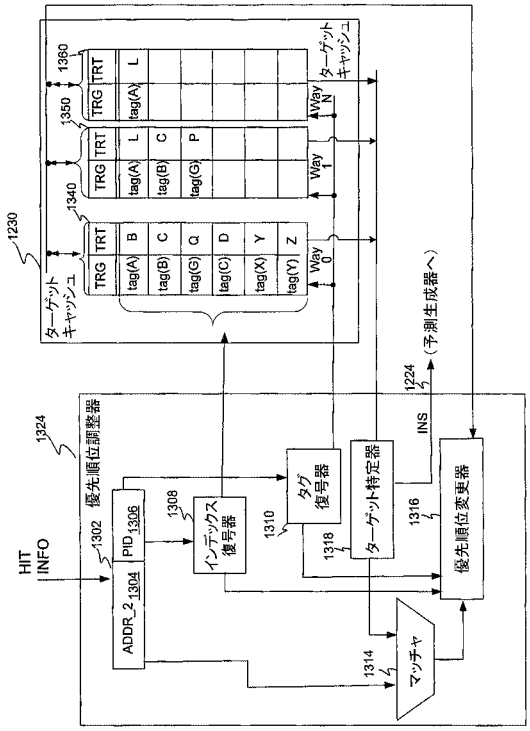
【図１１】



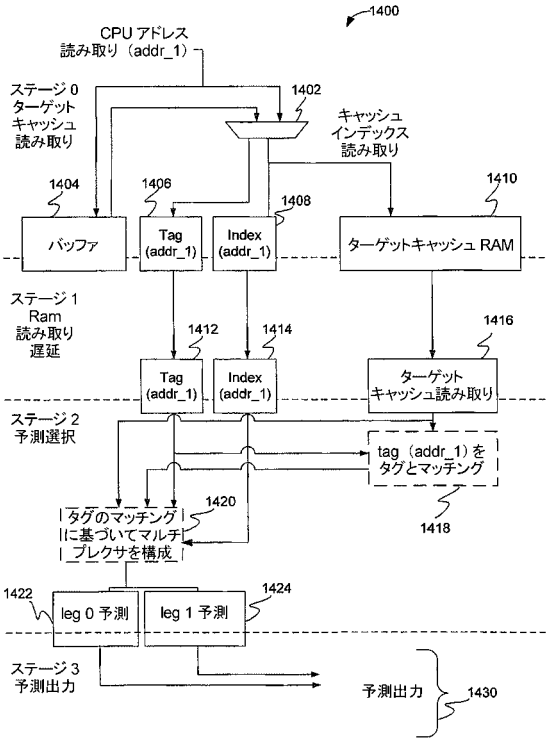
【図１２】



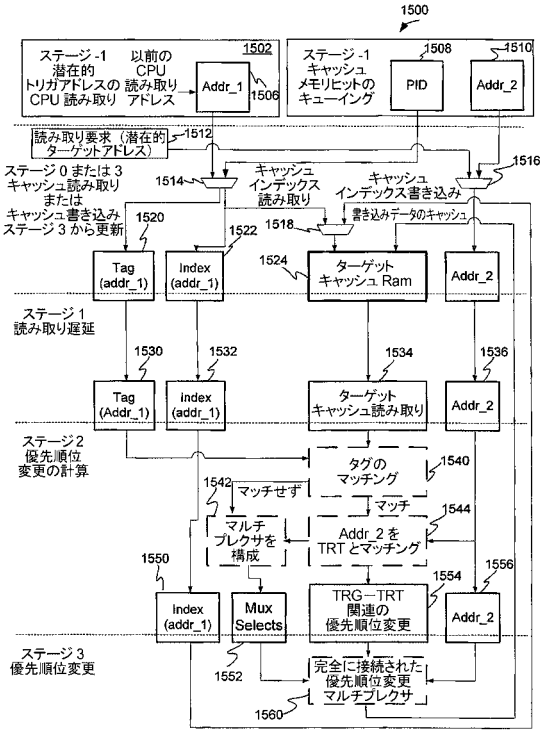
【図１３】



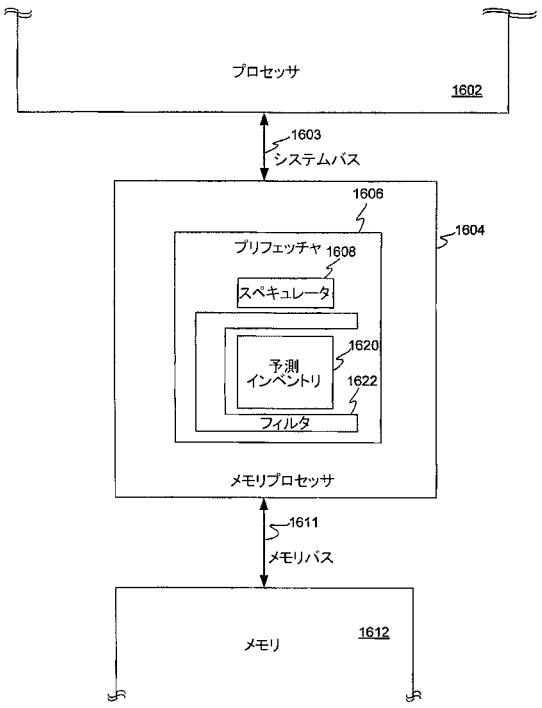
【図１４】



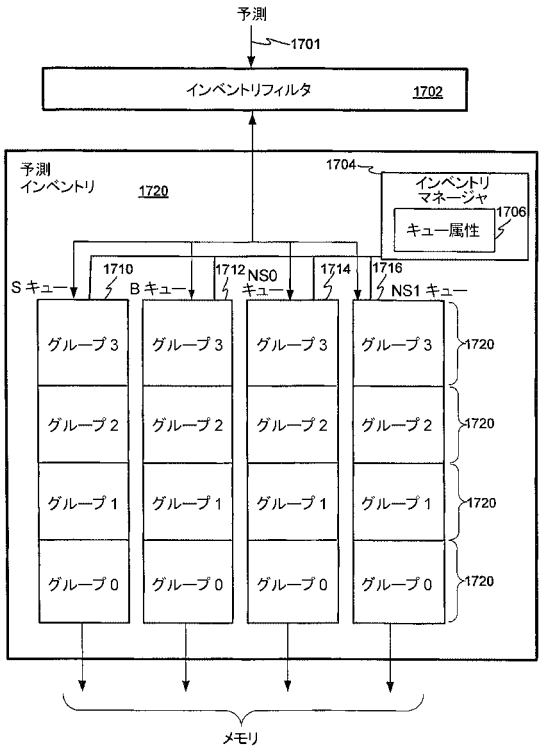
【図１５】



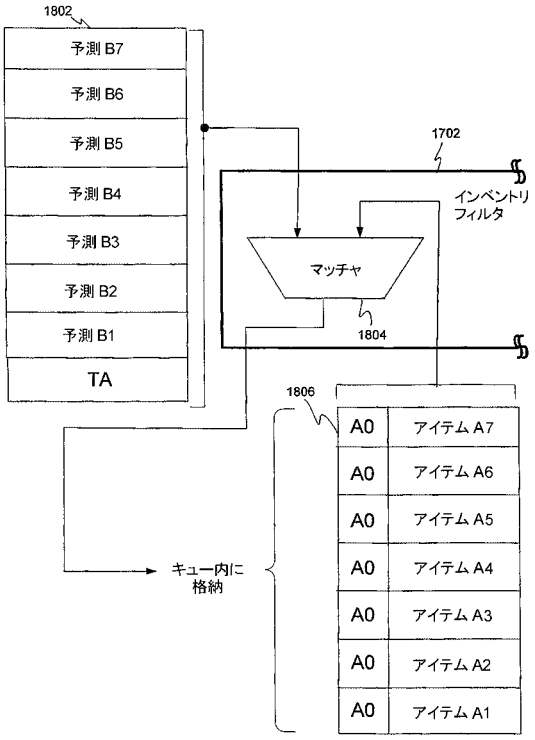
【図１６】



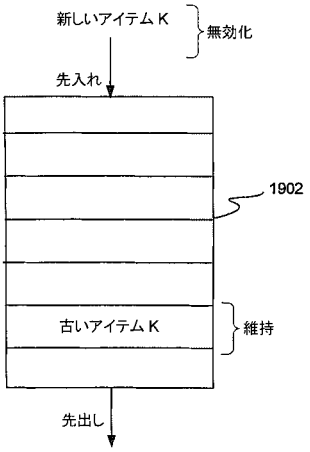
【図１７】



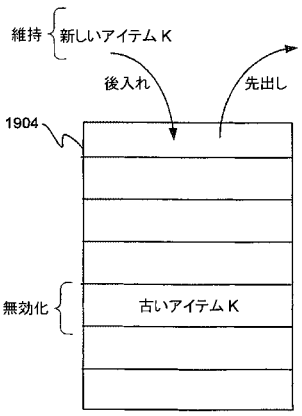
【図１８】



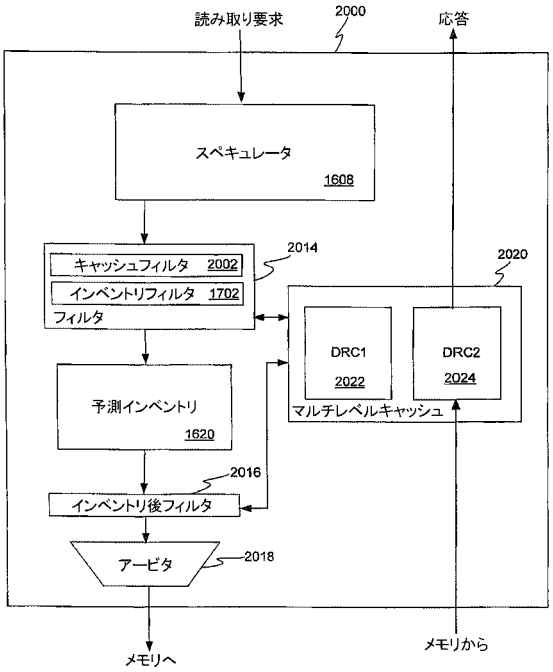
【図１９Ａ】



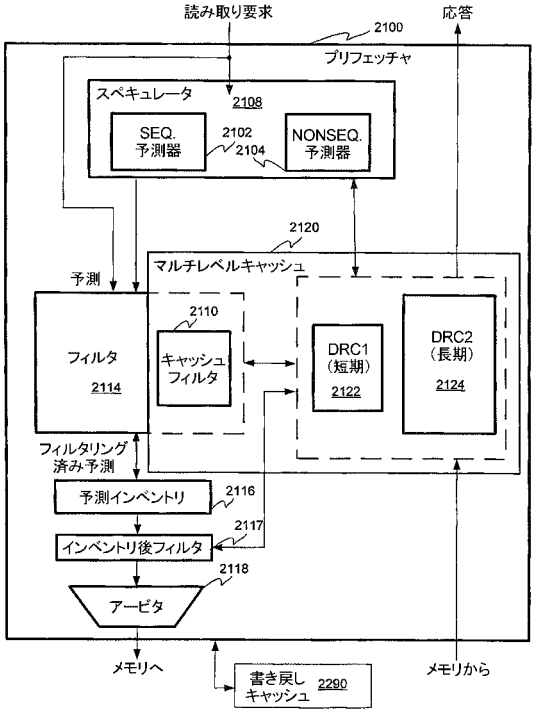
【図１９Ｂ】



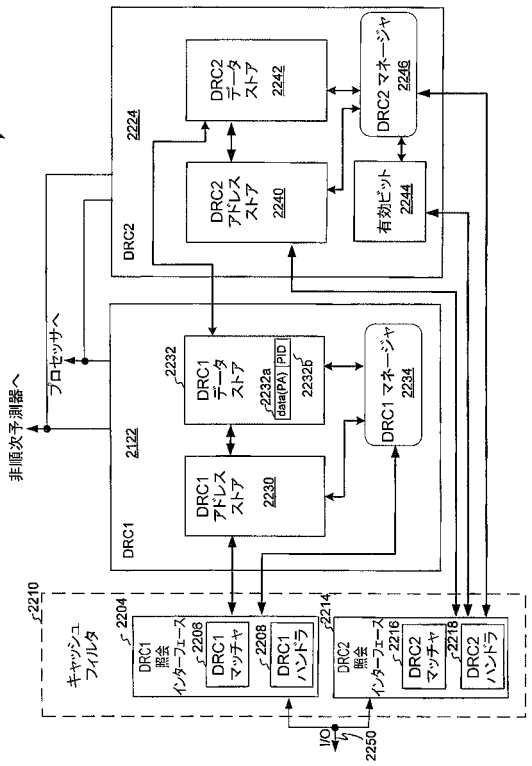
【図２０】



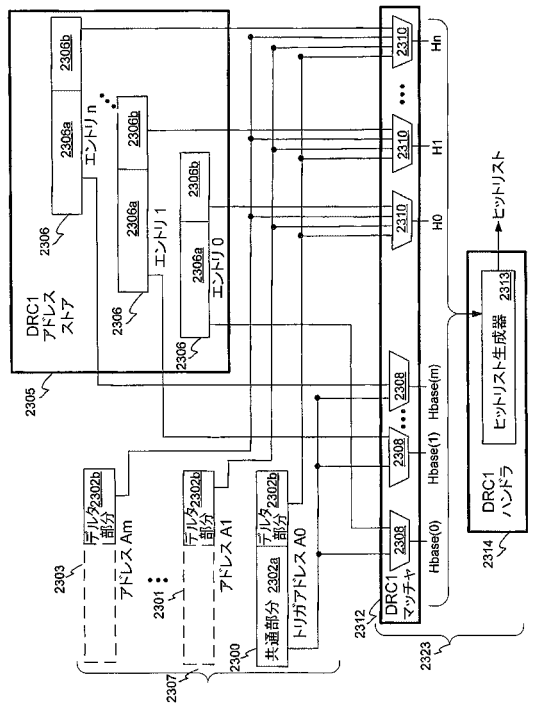
【図２１】



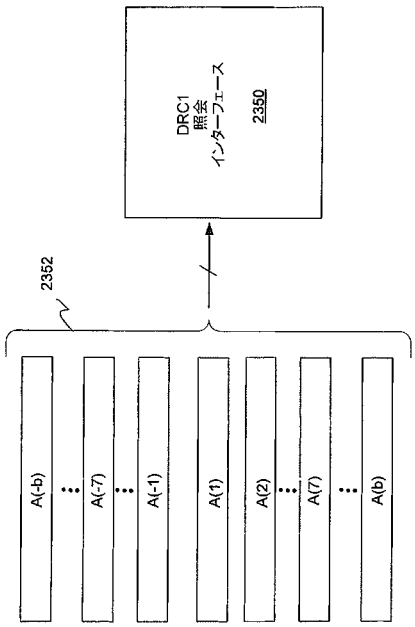
【図２２】



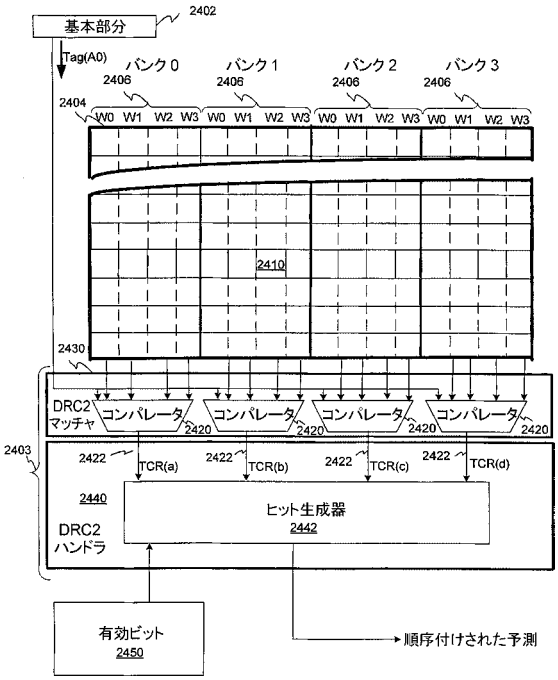
【図２３Ａ】



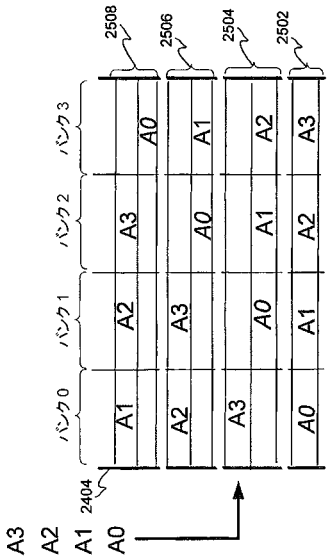
【図２３Ｂ】



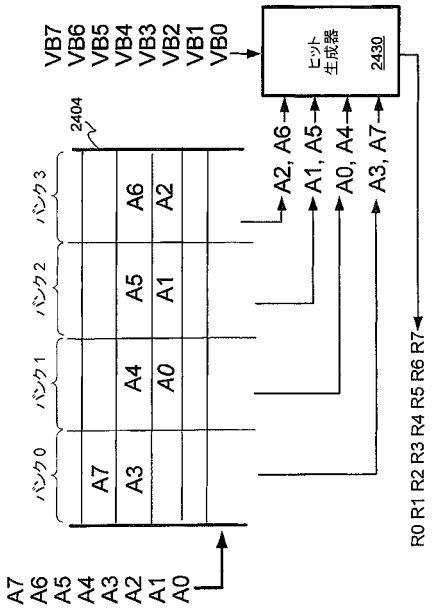
【図２４】



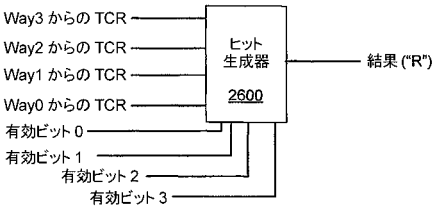
【図２５Ａ】



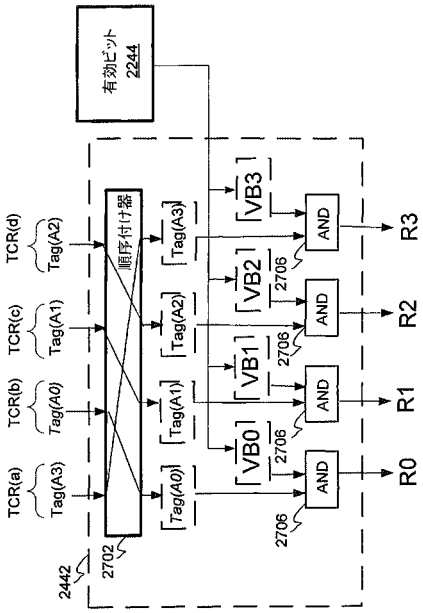
【図２５Ｂ】



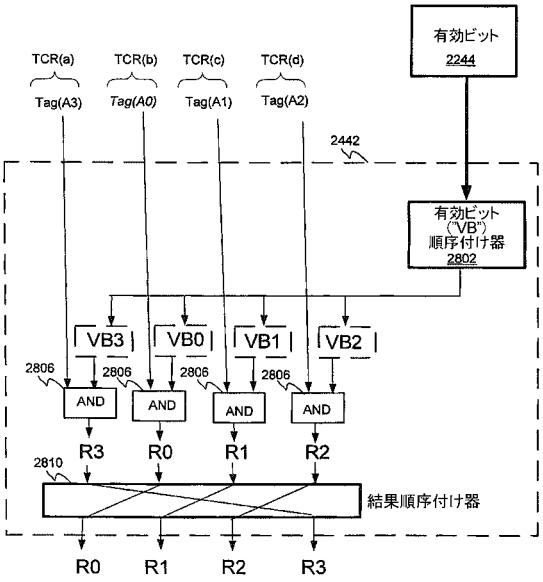
【図２６】



【図２７】


有効ビット
2244
順序付け器
TCR(d)
Tag(A2)
TCR(c)
Tag(A1)
TCR(b)
Tag(A0)
TCR(a)
Tag(A3)
Tag(A3)
Tag(A2)
Tag(A1)
Tag(A0)
VB3
VB2
VB1
VB0
2706
AND
R3
R2
R1
R0
2702
2442

【図２８】


有効ビット
2244
TCR(a)
TCR(b)
TCR(c)
TCR(d)
Tag(A3)
Tag(A0)
Tag(A1)
Tag(A2)
2442
有効ビット
("VB")
順序付け器
2802
VB3
VB0
VB1
VB2
2806
AND
R3
R0
R1
R2
2810
結果順序付け器
R0
R1
R2
R3

フロントページの続き

(31)優先権主張番号 10/920,995
(32)優先日 平成16年8月17日(2004.8.17)
(33)優先権主張国 米国(US)
(31)優先権主張番号 10/921,026
(32)優先日 平成16年8月17日(2004.8.17)
(33)優先権主張国 米国(US)

(72)発明者 ダニラク, ラドスラフ
アメリカ合衆国, カリフォルニア州, サンタ クララ, アパートメント ナンバー２５, プルネリッジ アヴェニュー １８５０
(72)発明者 シメラル, ブラッド, ダブリュー.
アメリカ合衆国, カリフォルニア州, サン フランシスコ, コンド ナンバー３, ドロアーズ ストリート １０４９
(72)発明者 ランゲンドーフ, ブライアン, キース
アメリカ合衆国, カリフォルニア州, ベニシア, センプルズ クロッシング ２７２
(72)発明者 ペスカドア, ステファノ, エー.
アメリカ合衆国, カリフォルニア州, サニーヴェール, レイクミュアー ドライヴ ２６８
(72)発明者 ヴィシェトスキー, ドミトリー
アメリカ合衆国, カリフォルニア州, クパチーノ, シダー スプリング コート １１６２７

審査官 中野 裕二

(56)参考文献 特表２００１－５１６９１９（ＪＰ，Ａ）
特開平０８－２１２０５４（ＪＰ，Ａ）
特開平６－１０３１６９（ＪＰ，Ａ）
特開２００１－１６６９３４（ＪＰ，Ａ）
特開平７－６４８６２（ＪＰ，Ａ）

(58)調査した分野(Int.Cl., ＤＢ名)
G06F 12/08
G06F 9/38