

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5340919号
(P5340919)

(45) 発行日 平成25年11月13日 (2013.11.13)

(24) 登録日 平成25年8月16日 (2013.8.16)

(51) Int. Cl.	F I
GO6F 9/30 (2006.01)	GO6F 9/30 310E
GO6F 12/08 (2006.01)	GO6F 9/30 310A
	GO6F 12/08 551J

請求項の数 17 (全 10 頁)

(21) 出願番号	特願2009-509934 (P2009-509934)	(73) 特許権者	595020643
(86) (22) 出願日	平成19年4月20日 (2007.4.20)		クアルコム・インコーポレイテッド
(65) 公表番号	特表2009-535744 (P2009-535744A)		QUALCOMM INCORPORATED
(43) 公表日	平成21年10月1日 (2009.10.1)		ED
(86) 国際出願番号	PCT/US2007/067057		アメリカ合衆国、カリフォルニア州 92
(87) 国際公開番号	W02007/130798		121-1714、サン・ディエゴ、モア
(87) 国際公開日	平成19年11月15日 (2007.11.15)		ハウス・ドライブ 5775
審査請求日	平成20年12月5日 (2008.12.5)	(74) 代理人	100108855
(31) 優先権主張番号	11/381,545		弁理士 蔵田 昌俊
(32) 優先日	平成18年5月4日 (2006.5.4)	(74) 代理人	100091351
(33) 優先権主張国	米国 (US)		弁理士 河野 哲
		(74) 代理人	100088683
			弁理士 中村 誠
		(74) 代理人	100109830
			弁理士 福原 淑弘

最終頁に続く

(54) 【発明の名称】 可変長命令のプリデコーディング

(57) 【特許請求の範囲】

【請求項 1】

プロセッサにより第1の命令の特定の属性を識別することであって、前記第1の命令は、プリデコードビットの第1のセットと関連付けられ、前記識別された特定の属性は、すべての符号化が定義された前記プリデコードビットの第1のセットにおける予め決められた符号化を有する他の属性とは異なる、識別することと、

前記プロセッサにより、プリデコードビットの第2のセットと関連付けられた第2の命令をエミュレートするために前記第1の命令を変更することであって、前記第1の命令の前記識別された特定の属性が、前記プリデコードビットの第2のセットを利用する命令のために定義されていない前記プリデコードビットの第2のセットの利用可能な符号化を使用して、前記プリデコードビットの第2のセットにおいて符号化される、変更することと、

前記プロセッサにより、前記第1の命令の代わりに、前記プロセッサに結合された命令キャッシュに、前記第2の命令と、前記第1の命令の前記識別された特定の属性を用いて符号化された前記プリデコードビットの第2のセットとを格納することと

を備えた、プロセッサにおいて命令をプリデコーディングする方法。

【請求項 2】

前記第1の命令は、例外を引き起こすこととなる命令である、請求項1に記載の方法。

【請求項 3】

前記第1の命令は未定義の命令である、請求項1に記載の方法。

10

20

【請求項 4】

前記プロセッサの命令セットは可変長命令セットであり、前記第 1 および第 2 の命令は異なる長さを有する、請求項 1 に記載の方法。

【請求項 5】

各定義された命令長は、その長さの命令と関連付けられた異なる数のプリデコードビットを有する、請求項 4 に記載の方法。

【請求項 6】

前記第 1 の命令の長さの命令と関連付けられた前記プリデコードビットの第 1 のセットの異なる符号化が、定義され、前記第 1 の命令の前記識別された特定の属性を符号化するために利用可能でない、請求項 1 に記載の方法。

10

【請求項 7】

前記第 2 の命令の長さの命令と関連付けられた前記プリデコードビットの少なくとも 1 つの符号化は、前記第 1 の命令の前記識別された特定の属性を符号化するために利用可能である、請求項 6 に記載の方法。

【請求項 8】

前記第 1 の命令は、それと関連付けられた 2 つのプリデコードビットを有する 16 ビット命令である、請求項 1 に記載の方法。

【請求項 9】

前記第 2 の命令は、それと関連付けられた 4 つのプリデコードビットを有する 32 ビット命令である、請求項 1 に記載の方法。

20

【請求項 10】

前記 32 ビット命令と関連付けられた前記プリデコードビットの 16 の符号化のうちの 1 つは、前記関連付けられた 32 ビット命令を 16 ビット未定義命令として識別する、請求項 9 に記載の方法。

【請求項 11】

メモリから読み出された命令と各命令と関連付けられたプリデコードビットとを格納する命令キャッシュメモリと、

前記メモリと前記命令キャッシュとの間に介在したプリデコーダと、

を備え、前記プリデコーダは、

前記メモリから読み出された第 1 の命令の特定の属性を識別する動作であって、前記第 1 の命令は、プリデコードビットの第 1 のセットと関連付けられ、前記識別された特定の属性は、すべての符号化が定義された前記プリデコードビットの第 1 のセットにおける予め決められた符号化を有する他の属性とは異なる、識別する動作と、

30

第 2 の命令をエミュレートするために前記第 1 の命令を変更する動作であって、前記第 2 の命令は、プリデコードビットの第 2 のセットと関連付けられ、前記第 1 の命令の前記識別された特定の属性が、前記プリデコードビットの第 2 のセットを利用する命令のために定義されていない前記プリデコードビットの第 2 のセットの利用可能な符号化を使用して、前記プリデコードビットの第 2 のセットにおいて符号化される、変更する動作と、

前記第 1 の命令の代わりに、前記命令キャッシュに、前記第 2 命令と、前記第 1 の命令の前記識別された特定の属性を用いて符号化された前記プリデコードビットの第 2 のセットとを格納する動作とをする、

40

メモリから命令を読み出すプロセッサ。

【請求項 12】

メモリから読み出された前記命令は可変長命令セットからの命令である、請求項 11 に記載のプロセッサ。

【請求項 13】

前記命令キャッシュは最短命令の長さに対応する複数の命令格納位置を備え、予め決められた数のプリデコードビットが前記格納位置のそれぞれと関連付けられる、請求項 12 に記載のプロセッサ。

【請求項 14】

50

前記命令キャッシュは最短命令より長い命令を整数倍の格納位置に格納し、各格納位置と関連付けられた前記プリデコードビットを各命令と関連付ける、請求項 1 3 に記載のプロセッサ。

【請求項 1 5】

前記第 1 の命令は、それに関連付けられた 2 つのプリデコードビットを有する 1 6 ビット命令である、請求項 1 1 に記載のプロセッサ。

【請求項 1 6】

前記第 2 の命令は、それに関連付けられた 4 つのプリデコードビットを有する 3 2 ビット命令である、請求項 1 1 に記載のプロセッサ。

【請求項 1 7】

前記 3 2 ビット命令と関連付けられた前記プリデコードビットの 1 6 の符号化のうちの 1 つは、前記関連付けられた 3 2 ビット命令を 1 6 ビット未定義命令であると識別する、請求項 1 6 に記載のプロセッサ。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本開示は一般的にはプロセッサの分野に関し、特に、例えば未定義の命令を識別するためのような、可変長命令をプリデコーディング(pre-decoding)する方法に関する。

【背景技術】

【0 0 0 2】

可変長命令セットアーキテクチャは当該技術分野において周知である。例えば Thumb (1 6 ビット) 拡張が ARM (3 2 ビット) 命令セットに追加されるとき、もたらされる混合物は可変長命令セットと考えられる。他の例として、実行可能な Java (登録商標) コードは各命令が 2 ~ 2 5 6 バイト長の間にある一連のバイトコードから成る。一般的に可変長命令セットは基本サイズ(base size)の命令と、基本サイズの整数倍であるより長い命令とから成る。

【0 0 0 3】

プロセッサアーキテクチャおよび命令セットは時間と共に進化する。特に、機能性は、かつてはそれを達成するために大規模なソフトウェアルーチンが要求されていたものが、半導体技術の進歩につれてハードウェアで達成可能となる。進歩したハードウェア構成を効率的に活用するために、新たな命令が命令セットに追加される。しかし安定したプロセッサアーキテクチャの 1 つの特性は後方互換性である。すなわち、プロセッサアーキテクチャ用に使われたソフトウェアはたとえ最新の機能を実行しなくても、あるいは最新の命令を直接実行しなくても、そのアーキテクチャに合致する、より旧式のプロセッサで実行可能であるべきである。従って共通のプロセッサアーキテクチャは例外を生成する「未定義の」命令(“undefined” instruction)を含み、未定義の命令と関連付けられた機能がソフトウェアにおいて実行される。

【0 0 0 4】

命令デコーディングを最適化するために従来の高性能プロセッサによってとられる 1 つの共通のアプローチはプリデコーディング(pre-decoding)である。プリデコードは、命令キャッシュ(I キャッシュ)に命令を格納する前にメモリから取り出される命令を検査して部分的にデコーディングする論理回路である。プリデコードは I キャッシュに各命令と共に格納される少数のプリデコードビットを生成する。キャッシュから命令を取り出すと、パイプラインデコードは命令デコーディングタスクを単純化するためにプリデコードビットを使用する。プリデコードビットは、例えば、分岐命令を識別したり、可変長命令アーキテクチャの命令長を識別したりする。プリデコードビットは未定義命令を識別するためにも用いられる。

【0 0 0 5】

プリデコーディングを実施する可変長命令セットプロセッサにおいて、少数のプリデコードビットは最短、すなわち基本命令長(base instruction length)と関連付けられる。

10

20

30

40

50

既知の命令キャッシュの実施は、この数のプリデコードビットを（それぞれが基本命令長に対応する）各Ｉキャッシュ格納位置と関連付けることである。Ｉキャッシュ格納位置の整数倍を占めるより長い命令は、より大きい数のプリデコードビットと対応して関連付けられる。命令と関連付けられるプリデコードビットに符号化が可能な個別の属性(property)の数は、従って、基本命令長に対して最小である。基本命令長の利用可能なプリデコードビット符号化の全てが違うように定義されると、未定義の基本長命令を識別するためにプリデコードビットを使用することは、基本命令長と関連付けられたプリデコードビットの数を増加する必要がある。この技術は、全ての格納位置に加えてＩキャッシュに格納されるプリデコードビットの数を次々に増加する。未定義命令の発生は稀で、また基本長より長い命令は豊富なプリデコードビット符号化スペースを有するため、未定義基本長命令を識別するためにプリデコードビットを追加することは高価なＩキャッシュ格納スペースを無駄遣いする可能性がある。

10

【発明の開示】

【発明の概要】

【０００６】

ここに記載した１つ以上の実施例によると、その命令長と関連付けられたプリデコードビット符号化スペースが完全に定義されているとき、プリデコードは命令の属性を表示(indicate)する。異なる長さの命令をエミュレートするために命令を変更することにより、プリデコードは異なる長さの命令と関連付けられたプリデコードビットの利用可能な符号化を利用して属性を表示する。

20

【０００７】

１つの実施例はプロセッサの命令をプリデコーディングする方法に関する。第１の命令の属性が識別される。第１の命令は第２の命令をエミュレートするために変更される。第１の命令の属性を識別する第２の命令のプリデコードビットが生成される。

【０００８】

他の実施例はメモリから命令を読み出す動作をするプロセッサに関する。プロセッサは、メモリから読み出される命令、および、各命令と関連付けられたプリデコードビットを格納する命令キャッシュメモリを含む。プロセッサは更にメモリと命令キャッシュとの間に置かれるプリデコードを含み、プリデコードは第１の命令の属性を識別し、第２の命令をエミュレートするために第１の命令を変更し、第１の命令の属性を識別する第２の命令のプリデコードビットを生成し、命令キャッシュに第２の命令のプリデコードビットを書き込む動作をする。

30

【詳細な説明】

【０００９】

図１はプロセッサ１０の機能ブロックダイアグラムである。プロセッサ１０は制御ロジック１４に従って命令実行パイプライン１２の命令を実行する。パイプライン１２は、１２ａおよび１２ｂのような複数の平行パイプラインを有するスーパースcalarデザインである。各パイプライン１２ａ、１２ｂはパイプステージに配置された様々なレジスタもしくはラッチ１６と、１つ以上の算術論理演算ユニット（ＡＬＵ）１８とを含む。パイプステージレジスタもしくはラッチ１６とＡＬＵ１８とは汎用レジスタファイル２８のレジスタからオペランドを読み出し、および／または汎用レジスタファイル２８のレジスタに結果を書き込む。

40

【００１０】

パイプライン１２ａ、１２ｂは、命令サイド変換索引バッファ(Instruction-side Translation Lookaside Buffer)（ＩＴＬＢ）２２により管理されるメモリアドレス指定および許可と共に、命令キャッシュ（ＩキャッシュもしくはＩ\$）２０から命令を取り出す。データは、主たる変換索引バッファ（ＴＬＢ）２６により管理されるメモリアドレス指定および許可と共に、データキャッシュ（ＤキャッシュもしくはＤ\$）からアクセスされる。様々な実施例において、ＩＴＬＢ２２はＴＬＢ２６の一部のコピーを備える。あるいは、ＩＴＬＢ２２およびＴＬＢ２６は一体化していてもよい。同様にプロセッサ１０の様々な

50

な実施例において、Iキャッシュ20およびDキャッシュ24は一体化すなわち統合されてもよい。Iキャッシュ20および/またはDキャッシュ24における誤りは、メモリアンターフェース34の制御の下で主たる(オフチップ)メモリ36へのアクセスの原因となる。

【0011】

メモリから取り出された命令はIキャッシュ20に格納される前にプリデコード回路37において処理される。プリデコード37は命令を部分的に復号し、Iキャッシュ20の命令と一緒に格納されるプリデコードビットを追加する。命令がIキャッシュ20から取り出されるとき、プリデコードビットは、命令をよりすばやくおよび/またはより効率的に復号するためにパイプライン12a、12b内のデコードパイプステージで使用される各命令に関する情報を提供する。例えばプリデコードビットは可変長命令セットアーキテクチャの命令の長さを識別したり、分岐命令を識別したりする。

10

【0012】

プロセッサ10は様々な周辺機器40、42へのアクセスを制御する入力/出力(I/O)インターフェース38を含む。当業者はプロセッサ10の多くのバリエーションが可能であることを認識するであろう。例えばプロセッサ10はIおよびDキャッシュの一方または双方のために第2レベル(L2)キャッシュを含んでもよい。加えて、プロセッサ10中に描かれる機能ブロックの1つ以上は、特定の実施例から省略されてもよい。

【0013】

図2は、例示的で非限定的な一実施例に関する、プリデコード37を介してメモリ36からIキャッシュ20の1ラインに命令のマッピングを描く機能ブロックダイアグラムである。命令Aは16ビットの命令(この例の基本命令長)である。命令Aがメモリから取り出されると、プリデコード37はIキャッシュ20に命令Aと共に格納される2つのプリデコードビット P_A を付加する。命令Bは B_0 、 B_1 としてメモリ36に格納された32ビット命令である。プリデコード37は P_{B0} 、 P_{B1} としてIキャッシュ20に格納される4つのプリデコードビットを付加する。命令Cは16ビット命令であり、2つのプリデコードビット P_C と共にIキャッシュ20に格納される。

20

【0014】

図2の表に示されるように、2つのプリデコードビットをそれぞれ有する命令AおよびCは4つのプリデコードビット符号化スペースを有する。すなわち、命令長、命令が分岐であるかなどのような4つの個別の属性が命令AおよびCのプリデコードビットに符号化される。4つのプリデコードビットを有する命令Bは16のプリデコードビット符号化スペースを有する。もしアーキテクチャによってサポートされるなら、48ビット命令は6つのプリデコードビット、64の符号化スペースを有するなどとなる。

30

【0015】

ある実施においてAおよびCのような16ビット命令の4つのプリデコードビット符号化の全てが定義されると仮定すると、追加のプリデコードビットがIキャッシュ20の各16ビット格納位置に追加され、プリデコードビットを使用して未定義の16ビット命令を表示する。このことはIキャッシュ20の各格納位置が3つの関連するプリデコードビットを有するという結果になる。16ビット命令は、従って、8つのプリデコード符号化スペースを有し、そのうち5つのみを使用される。加えて、32ビット命令は64のプリデコード符号化スペースを有し、48ビット命令は512のプリデコード符号化スペースを有する。これはプリデコードが表示することができる命令についてのそれぞれの属性の数との大きな隔たりとなるため、16ビット命令長のプリデコードビット数の増加は高価なIキャッシュ20格納の大きな無駄遣いとなる。

40

【0016】

一実施例によると、この例における未定義の16ビット命令は、32ビット命令をエミュレートするために変更される。命令が実際には16ビット未定義命令であるという属性は、32ビット命令と関連付けられた16のプリコード符号化のうちの1つに符号化される。図3において、命令Dは16ビット未定義命令である。命令DをIキャッシュ20に

50

書きこむと、命令Dは、 D_0 および D_1 を備える32ビット命令をエミュレートするために変更される。命令Dが実際は16ビット未定義命令であるということを表示するプリデコードビット P_{D_0} および P_{D_1} が書きこまれる。

【0017】

一実施例においては、16ビット未定義命令だけがパイプライン12によって要求される場合、プリデコード37は、命令Dを捨てて、プリデコードビット P_{D_0} および P_{D_1} のみをIキャッシュ20に書きこむ。他の実施例においては、命令のビットフィールド内のデータが共通あるいはソフトウェアトラップを誘導(vector)するために役立つような場合、命令Dは、Iキャッシュ20に書きこまれるプリデコードビット P_{D_0} および P_{D_1} だけでなく、 D_0 および D_1 のいずれか一方もしくは双方に書き込まれる。さらに他の実施例においては、プリデコード37は命令Dを D_0 あるいは D_1 のうちのいずれかに書きこみ、また、プリデコードビット P_{D_0} および P_{D_1} を書きこむだけでなく、追加の情報を他の16ビットIキャッシュ20格納位置に書きこむ。いずれにしても、32ビット命令をエミュレートし(そのため全プリデコードビットが読まれる)、命令が実際には16ビット未定義命令であることを表示するのはプリデコードビット P_{D_0} および P_{D_1} である。このようにプリデコード37は基本サイズ命令と関連付けられたプリデコードビットを拡張することなく基本サイズ未定義命令を表示することができ、従って非能率的なメモリの使用を避けることができる。

【0018】

図3に描かれる実施例において、命令EもFもプリデコードされず、Iキャッシュ20にロードもされない。ラインの命令の実際の長さがメモリ36とIキャッシュ20との間で変更されるので、キャッシュラインの残りは対応するメモリスペース(例えばEおよびFの双方)に残りの命令の全てを含むことができない。プリデコード37はキャッシュラインの残りを不使用あるいは無効と記録し、もしくは適切なビット位置に無効あるいは未定義プリデコードビットを単純に挿入する。正常な実行においては、プロセッサは、未定義命令が生じるとソフトウェアにトラップし、キャッシュラインは取りかえられる可能性がある。未定義命令により生じるトラップに対処するためにソフトウェア例外処理装置を使用することは、プロセッサ10のようなプロセッサがシリコンあるいは他の半導体材料に固定されるときに意図されなかった命令を、ソフトウェアがエミュレートすることを可能にする。もし他のコードがキャッシュラインに分岐すると、不正確なプリデコードに遭遇し、および/または予期された命令を認識せず、キャッシュラインにリロードさせ、あるいは例外を生じる恐れがある。未定義命令は滅多に生じないと想定されるので、異なる長さの命令をエミュレートする変更された未定義命令が原因で破損したIキャッシュラインの性能インパクトは大きな問題ではない。

【0019】

上述した通り、ここに開示される方法は基本命令長の未定義命令に限定されない。一般的に、通常はプリデコードビットを介して表示されるが、関連する命令長のプリデコードビット符号化スペースが完全に定義されている命令属性については、属性は、異なる長さの命令をエミュレートするために命令を変更して異なる長さ命令の使用可能なプリデコードビット符号化を用いることによって表示される。例えば、先の例においては32ビット命令と関連付けられている4つのプリデコードビットの全16符号化が定義されているが、16ビット命令と関連付けられている2つのプリデコードの4の符号化のうち3つ以下だけが定義されていると仮定する。この場合、32ビット命令プリデコードビットにおいて定義されていない32ビット命令の特定の属性は、16ビット命令をエミュレートするために32ビット命令を変更して16ビット命令と関連付けられている2のプリデコードビットの使用可能なプリデコードビット符号化における関連属性を符号化することにより表示される。この場合、32ビット命令ビットフィールドの半分は失われ、符号化されている属性は(そのように表示される唯一の性能ではないが、命令は未定義であるという事実のような)命令ビットフィールドに依存しないものであることが好ましい。

【0020】

図 4 は、一実施例に従った、命令をプリデコーディングする方法を描写する。I キャッシュ 20 におけるミスがあると、メモリインターフェース 34 はメモリ 36 から少なくとも第 1 の命令を取り出す (ブロック 50)。プリデコーダ 37 は第 1 の命令を検査して、第 1 の命令は未定義であるというような第 1 の命令の属性を識別する (ブロック 52)。プリデコーダ 37 は、異なる長さの第 2 の命令のような、第 2 の命令をエミュレートするために、第 1 の命令を変更する (ブロック 54)。プリデコーダ 37 は更に、第 1 の命令の属性を識別する、第 2 の命令のプリデコードビットを生成する (ブロック 56)。すなわち、第 2 の命令長と関連付けられたプリデコードビットの利用可能な符号化のうち 1 つが第 1 の命令の関係属性 (relevant property) に割り当てられる。プリデコーダ 37 は第 2 の命令と関連プリデコードビットとを I キャッシュ 20 に書きこむ (ブロック 58)。パイプライン 12 が I キャッシュ 20 から第 2 の命令とそのプリデコードビットとを取り出すとき、デコードパイプステージのロジックは、第 1 の命令が未定義であるというような第 1 の命令の属性を識別するために、プリデコードビットを検査する (ブロック 60)。パイプライン 12 はその後、例えばソフトウェアの未定義命令を対処するために例外を引き起こすというような、適切な行動をとる。

10

【0021】

ここに記載された 1 つ以上の実施例によると、その命令長と関連付けられているプリデコードビット符号化スペースが完全に定義されているとき、プリデコーダ 37 は命令の属性を表示する。異なる長さの命令をエミュレートするために命令を変更することによって、プリデコーダ 37 は属性を表示するために異なる長さの命令と関連付けられたプリデコードビットの利用可能な符号化を使用する。より短い命令の属性がより長い命令のプリデコードビット符号化において表示されるときは、1 以上の命令あるいは命令の部分は I キャッシュラインから退去される。より長い命令の属性がより短い命令のプリデコードビット符号化において表示されるとき、命令ビットフィールドのいくつかは失われる。従って、ここに記載の命令属性表示技術は、命令が未定義であるというようなく (このようなケースに限定されないが)、例外を生じる命令属性を表示するために共通して使用される。

20

【0022】

本発明は特定の特徴、態様および実施例に関してここで説明されてきたが、多くの変形、変更および他の実施例が本発明の広い範囲内で可能であることは明らかであり、従って、あらゆる変形、変更および実施例は本発明の範囲内にあるとみなされるべきである。本実施例は、従って、全ての態様において説明的なものであって限定的ではないと解釈されるべきであり、添付の請求項の意味および均等物の範囲内にあるあらゆる変更は、ここに包含されるよう意図される。

30

【図面の簡単な説明】

【0023】

【図 1】図 1 は、本発明のプロセッサの機能ブロックダイアグラム。

【図 2】図 2 は、本発明のメモリから命令キャッシュへの可変長命令のマッピングを図示した機能ブロックダイアグラム。

【図 3】図 3 は、本発明のメモリから命令キャッシュへの未定義命令のマッピングを図示した機能ブロックダイアグラム。

40

【図 4】図 4 は、本発明の 1 の実施例による、命令をプリデコーディングする方法を示す図。

【図 1】

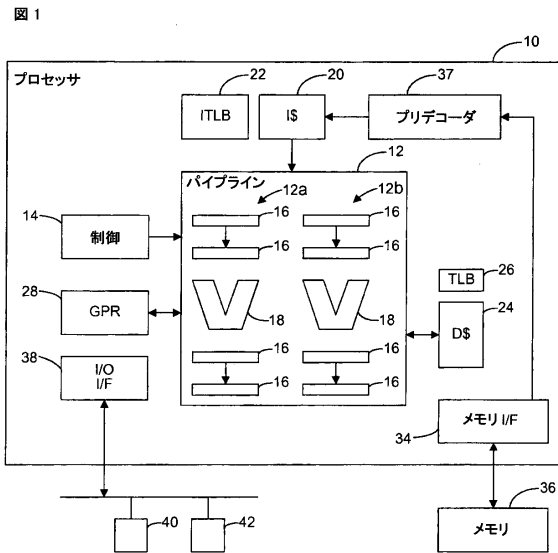


FIG. 1

【図 2】

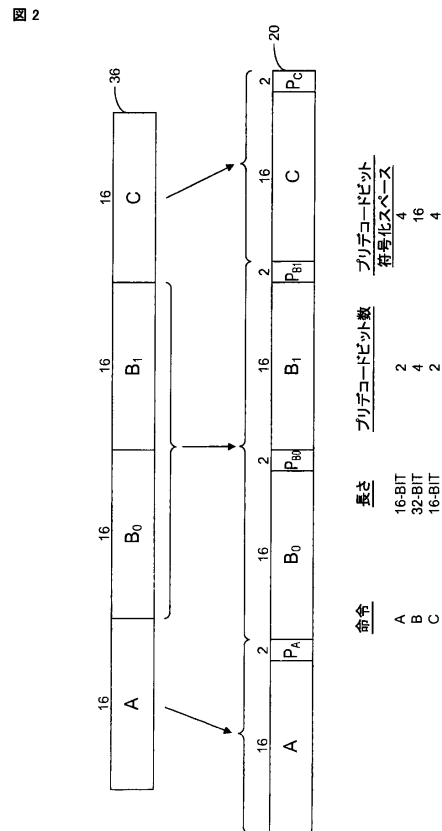


FIG. 2

【図 3】

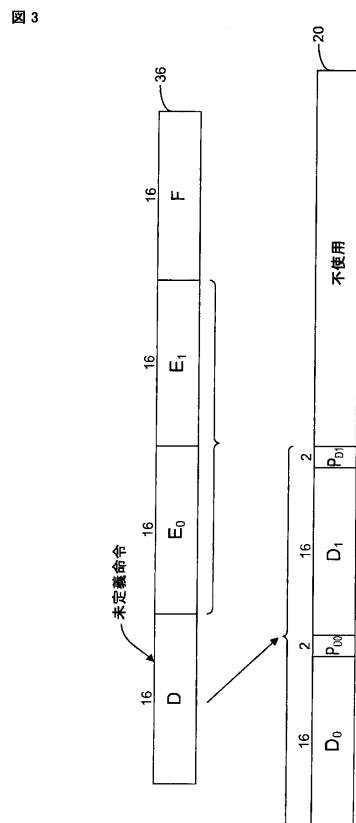


FIG. 3

【図 4】

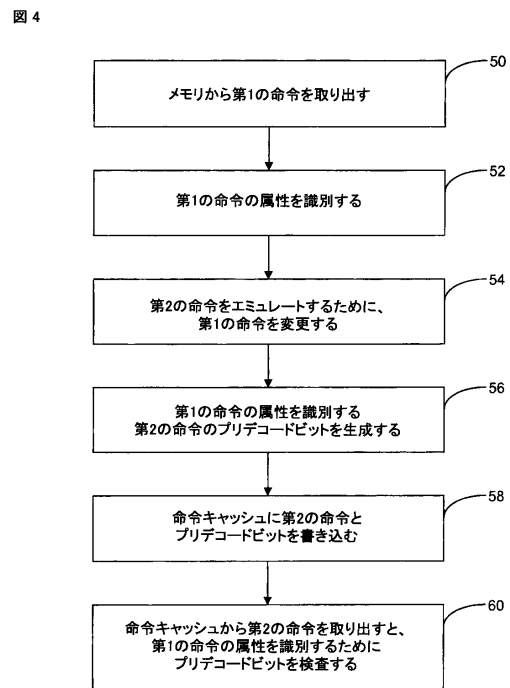


FIG. 4

フロントページの続き

- (74)代理人 100075672
弁理士 峰 隆司
- (74)代理人 100095441
弁理士 白根 俊郎
- (74)代理人 100084618
弁理士 村松 貞男
- (74)代理人 100103034
弁理士 野河 信久
- (74)代理人 100119976
弁理士 幸長 保次郎
- (74)代理人 100153051
弁理士 河野 直樹
- (74)代理人 100140176
弁理士 砂川 克
- (74)代理人 100100952
弁理士 風間 鉄也
- (74)代理人 100101812
弁理士 勝村 紘
- (74)代理人 100070437
弁理士 河井 将次
- (74)代理人 100124394
弁理士 佐藤 立志
- (74)代理人 100112807
弁理士 岡田 貴志
- (74)代理人 100111073
弁理士 堀内 美保子
- (74)代理人 100134290
弁理士 竹内 将訓
- (74)代理人 100127144
弁理士 市原 卓三
- (74)代理人 100141933
弁理士 山下 元
- (72)発明者 スミス、ロドニー・ウェイン
アメリカ合衆国、ノースカロライナ州 27614、ローリー、バスコム・ドライブ 1400
- (72)発明者 ステムベル、ブライアン・マイケル
アメリカ合衆国、ノースカロライナ州 27614、ローリー、ウェルウォーター・コート 1309

審査官 三坂 敏夫

- (56)参考文献 米国特許出願公開第2004/0133764(US, A1)
米国特許出願公開第2005/0262329(US, A1)
特開平03-168836(JP, A)
国際公開第2003/046715(WO, A1)
米国特許第06269384(US, B1)

- (58)調査した分野(Int.Cl., DB名)
G06F 9/30

G 0 6 F 1 2 / 0 8