

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2008-524691
(P2008-524691A)

(43) 公表日 平成20年7月10日(2008.7.10)

(51) Int.Cl.
G06F 15/80 (2006.01)

F I
G O 6 F 15/80

テーマコード (参考)

審査請求 未請求 予備審査請求 未請求 (全 15 頁)

(21) 出願番号 特願2007-546274 (P2007-546274)
 (86) (22) 出願日 平成17年12月13日 (2005.12.13)
 (85) 翻訳文提出日 平成19年8月15日 (2007.8.15)
 (86) 国際出願番号 PCT/IB2005/054202
 (87) 国際公開番号 W02006/064461
 (87) 国際公開日 平成18年6月22日 (2006.6.22)
 (31) 優先権主張番号 04300918.2
 (32) 優先日 平成16年12月17日 (2004.12.17)
 (33) 優先権主張国 欧州特許庁 (EP)

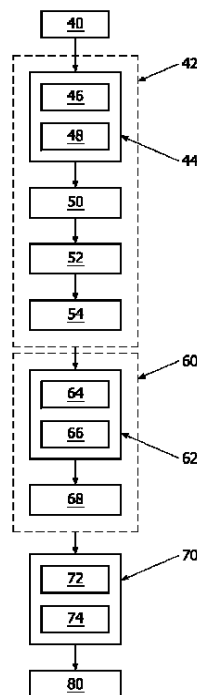
(71) 出願人 507219491
 エヌエックスピー ビー ヴィ
 オランダ国 5656エイジー アインド
 ーフェン ハイ テク キャンパス 60
 (74) 代理人 100147485
 弁理士 杉村 憲司
 (74) 代理人 100134005
 弁理士 澤田 達也
 (72) 発明者 ブェルノ バラリン
 フランス国 75008 パリ ブールバ
 ール ハイスマン 156 ソシエテ シ
 ヴィル エスパイデ内

最終頁に続く

(54) 【発明の名称】 算術／論理演算ツリーの計算

(57) 【要約】

少なくともn個の並列処理要素を備えているマイクロプロセッサにて、算術又は論理演算の、少なくとも第1及び第2ツリーを計算する方法。この方法は、: a) n個の処理要素を用いて、第1ツリーの第1イテレーションの算術又は論理演算をn回並列に実行し(48において)、次いで、b) 第1イテレーションの計算に用いたn個の処理要素から選定したm個の処理要素を用い、他のn-m個の処理要素は第2イテレーションの計算には使用しないで、第1イテレーションの結果間でm回の算術又は論理演算を並列に実行する(66において)ステップを含む。本発明による方法は、第1ツリーの第2イテレーションの計算と並列に、第2ツリーのk算術又は論理演算を、第1ツリーの第2イテレーションの計算には使用されなかったn-m個の処理要素から選定したk個の処理要素を用いて、並列に実行する(66において)ステップを含む。



【特許請求の範囲】**【請求項 1】**

n 回の算術又は論理演算を並列に実行する、少なくとも n 個の並列処理要素を備えているマイクロプロセッサにて、算術又は論理演算の、少なくとも第 1 及び第 2 ツリーを計算する方法であって、各ツリーの計算が複数の連続イテレーションを必要とし、各イテレーションの結果を、次のイテレーションの計算中に使用し、ここに、n は 4 よりも大きな正の整数とし、当該方法が：

a) 前記 n 個の処理要素を用いて、前記第 1 ツリーの第 1 イテレーションの算術又は論理演算を (4 8 にて) n 回並列に実行し、次いで、

b) 前記第 1 イテレーションの計算に使用した n 個の処理要素から選定した m 個の処理要素を用い、他の n - m 個の処理要素は、第 2 イテレーションの計算には使用しないで、ここに、m は n よりも確実に小さい整数として、第 1 イテレーションの結果間で算術又は論理演算を (6 6 にて) m 回並列に実行する、

ステップを含む、算術 / 論理演算ツリーの計算方法において、

k は n - m よりも小さいか、又はそれに等しい整数として、第 1 ツリーの第 2 イテレーションの計算と並列に、第 1 ツリーの第 2 イテレーションの計算には使用されなかった n - m 個の処理要素から選定した k 個の処理要素を用いて、第 2 ツリーの算術又は論理演算を k 回並列に実行するステップを含むようにする、ことを特徴とする算術 / 論理演算ツリーの計算方法。

【請求項 2】

前記第 1 ツリーの第 2 イテレーションの計算に先立つ処理要素間のデータ転送と、前記第 2 ツリーの k 回の算術又は論理演算の実行に先立つ処理要素間のデータ転送とを、並列に実行する、請求項 1 に記載の方法。

【請求項 3】

前記第 2 ツリーの計算に用いる各処理要素を、前記第 1 ツリーの第 2 イテレーションの計算に用いる 2 つの処理要素間にインターリーブさせる、請求項 1 又は 2 に記載の方法。

【請求項 4】

前記マイクロプロセッサは、p 回の同じ算術又は論理演算を並列に実行する p 個の同一処理要素を備え、これら処理要素は共通のダイ上に製造され、p は n より大きいか若しくは等しい整数とする、SIMD マイクロプロセッサとする、請求項 1 ~ 3 のいずれか一項に記載の方法。

【請求項 5】

少なくとも第 1 及び第 2 のバイナリーツリーの和を計算するステップを有している、通信デバイスにおける逆拡散法であって、前記請求項 1 ~ 4 のいずれか一項に記載の計算方法を用いて、n 回の算術又は論理演算を並列に計算する、少なくとも n 個の並列処理要素を備えているマイクロプロセッサにて、前記少なくとも第 1 及び第 2 のバイナリーツリーの和を計算する、電気通信デバイスにおける逆拡散法。

【請求項 6】

請求項 1 ~ 4 のいずれか一項に記載の算術又は論理演算の、第 1 及び第 2 ツリーを計算する方法を実行するための命令を含むメモリであって、これらの命令を、少なくとも n 個の並列処理要素を備えているマイクロプロセッサにて実行させる際に、n は 4 よりも大きな正の整数とする、メモリ。

【請求項 7】

請求項 1 ~ 4 のいずれか一項に記載の算術又は論理演算の、少なくとも第 1 及び第 2 ツリーを計算する方法を実行するための命令を含む、マイクロプロセッサのプログラムであって、これらの命令を、少なくとも n 個の並列処理要素を備えているマイクロプロセッサにて実行させる際に、n は 4 よりも大きな正の整数とする、マイクロプロセッサのプログラム。

【発明の詳細な説明】**【技術分野】**

10

20

30

40

50

【 0 0 0 1 】

本発明は、算術又は論理演算ツリーの計算方法に関する。

【 背景技術 】

【 0 0 0 2 】

算術又は論理演算のツリーは、ツリー構造に従って編成した算術又は論理演算から成る。このツリー構造は複数のノードを含み、各ノードは少なくとも2つの入力端と1つの出力端とを有する。各ノードは、その入力端で受け取ったデータ間の算術又は論理演算に対応し、算術又は論理演算の結果をその出力端に出力する。

【 0 0 0 3 】

ツリー構造は、リーフノードから成る第1レベルと、各々が先行する同じレベルのノードの出力端に直接接続された入力端を有している複数のノードから成る幾つかの中間レベルと、ルートノードを有するルートレベルとの、連続レベルに分けることができる。

【 0 0 0 4 】

このような演算ツリーの計算は、複数回のイテレーションを必要とする。イテレーションの間に、同じレベルのノードに対応する全ての演算が実行される。

【 0 0 0 5 】

典型的に、少なくともn個の並列処理要素を備えているマイクロプロセッサにて、算術又は論理演算の少なくとも第1及び第2ツリーを計算する方法は、

a) n個の処理要素を用いて、第1ツリーの第1イテレーションの算術又は論理演算をn回並列に実行し、次いで、

b) 第1イテレーションの計算に使用したn個の処理要素から選定したm個の処理要素を用い、他のn - m個の処理要素は第2イテレーションの計算には使用しないで、ここに、mはnよりも確実に小さい整数として、第1イテレーションの結果間で算術又は論理演算をm回並列に実行する、ステップを備えている。

【 0 0 0 6 】

第1の演算ツリーを一旦計算したら、第2演算ツリーを同じように処理すべく、処理要素は構成されている。

【 0 0 0 7 】

並列処理要素を有するマイクロプロセッサを用いるこのようなやり方は、第1演算ツリーの第2イテレーションの計算中に、一部の処理要素が使用されないままとなるから、非効率的である。

【 0 0 0 8 】

n個の並列処理要素を備えているマイクロプロセッサの例は、Andrew Paul WallaceによるUS 2003/0088603に開示されている。

【 発明の開示 】

【 発明が解決しようとする課題 】

【 0 0 0 9 】

従って、本発明の目的は、複数の並列処理要素を有しているマイクロプロセッサをもっと有効的に用いて、少なくとも第1及び第2の演算ツリーを計算する方法を提供することにある。

【 課題を解決するための手段 】

【 0 0 1 0 】

本発明は、斯様なマイクロプロセッサで少なくとも第1及び第2の演算ツリーを計算する方法において、当該方法が、kはn - mよりも小さいか、又はそれに等しい整数として、第1ツリーの第2イテレーションの計算と並列に、第1ツリーの第2イテレーションの計算には使用されなかったn - m個の処理要素から選定したk個の処理要素を用いて、第2ツリーの算術又は論理演算をk回並列に実行するステップを含むようにする、演算ツリーの計算方法を提供する。

【 0 0 1 1 】

10

20

30

40

50

上記方法においては、第1演算ツリーの或るイテレーションに対して使用されない $n - m$ 個の処理要素を使用して、第2演算ツリーの算術又は論理演算を並列に計算する。従って、並列に使用される処理要素の数と、並列に実行される演算の回数が増えるために、複数の並列処理要素を備えているマイクロプロセッサの使用効率が向上する。

【0012】

請求項2の特徴事項は、 m 個の処理要素間にデータを転送するのに必要な時間中に、 k 個の処理要素間にもデータを並列に転送するために、処理時間を低減させることにある。

【0013】

請求項3の特徴事項は、処理要素間のデータ転送時間を低減させることにある。

10

【0014】

本発明はまた、上記方法を実行するための命令を有する、メモリ及びマイクロプロセッサのプログラムにも関する。

【0015】

本発明はまた、同一の算術又は論理演算の少なくとも第1及び第2ツリーを計算するために上記方法を使用する、通信デバイスにおける逆拡散法 (de-spreading method) にも関する。

【発明を実施するための最良の形態】

【0016】

図1は、演算ツリー2の一例を示す。例えば、図1において、それぞれのノードは、各列が演算ツリーのレベルに対応する平行な垂直方向の列にて編成されている。図1の最も左の列は、全てのリーフノード LF_i を含み、図1の最も右の列は、ルートノード RN だけを含んでいる。

20

【0017】

各リーフノード LF_i は、処理すべきデータを受け取る2つの入力端を有している。2つのリーフノード LF_i の出力端は、次列のノードの入力端に接続されている。この次列のノードの出力端は、同様に、その次列の入力端に接続されている。一つの列のノードを次の列のノードに接続するというこの方法は、ルートノード RN を有するルート列に達するまで、繰り返される。

【0018】

ここで、演算ツリー2はバイナリツリー (二分木) の和であり、各ノードは、入力された2つのデータの加算に対応する。例えば、演算ツリー2は、CDMA (符号分割多重アクセス) 通信システムにおける拡散信号の、各チップのチップ値のような、極めて多数のデータ加算を計算するのに用いられる。

30

【0019】

演算ツリー2の計算には、複数のイテレーション IT_i が必要である。第1イテレーション IT_1 中には、リーフノード LF_i の算術又は論理演算を実行する。そして、次のイテレーション IT_2 中には、左から第2列のノードに対応する演算を実行する。このような演算処理は、ルートノード RN に対応する演算が実行されるまで、繰り返す。

【0020】

例えば、演算ツリー2に入力されるデータの数 n が256に等しい場合に、演算ツリー2は、8回のイテレーションで計算される。 IT_1 から IT_8 までのイテレーション中に実行される加算回数は、それぞれ、128、64、32、16、8、4、2及び1回である。

40

【0021】

図2は、演算ツリー2を計算するのに用いる、SIMD (Single Instructions Multiple Data) マイクロプロセッサ20を示している。SIMDマイクロプロセッサは特に、多数のデータに同じ処理を同時に実行するのに適している。

【0022】

簡単化のために、図2には、本発明を理解するのに必要な細部のみを示してある。

50

【0023】

マイクロプロセッサ20は、 p 個の多数の並列処理要素 PE_i を備えている。例えば、 p は4086に等しく、インデックス i は1から4086までの範囲の整数とする。全ての並列処理要素 PE_i は、異なるデータ組を、同じプログラムで並列に実行するように設計されている。従って、各処理要素は、算術又は論理演算を実行するためのALU（算術兼論理ユニット）と、このALUによって処理すべきデータ及びこの処理結果を格納するためのデータレジスタ DR_i とを有している。例えばデータレジスタ DR_i は、192ビットまで格納することができる。

【0024】

図2には、 PE_1 、 PE_2 、 PE_i 、 PE_{p-i} 、 PE_p の5つの処理要素のみを示してある。

10

【0025】

マイクロプロセッサ20の処理要素は、相互接続ネットワーク24によって互いに接続されて、1つのデータレジスタ DR_i に格納されているデータを、他の処理要素 PE_j における他のデータレジスタ DR_j へ転送し得るようになっている。各データレジスタ DR_i は、データの転送中に使用されるネットワーク24上のアドレスを有している。

【0026】

それぞれの処理要素 PE_i は、同じシリコンのダイまたは基板上に製造される。

【0027】

マイクロプロセッサ20は、全ての処理要素に同じプログラムが適用され、また、全ての処理要素が同時に同じ命令を実行するので、使いやすい。そのプログラムは、マイクロプロセッサ20に接続したメモリ26に格納されている。

20

【0028】

メモリ26は、マイクロプロセッサ20が図2の方法を実行するための命令を実行する際に、これらの命令を含んでいる。

【0029】

マイクロプロセッサ20で演算ツリー2のような少なくとも2つの演算ツリーを計算する方法を、図3～図6を参照して、8つの同一のバイナリーツリーの和 A 、 B 、 C 、 D 、 E 、 F 、 G 、 H を並列に処理するという特別な場合につき説明する。例えば、以下に述べる方法は、通信デバイスにおいて実施される逆拡散法の一部である。

30

【0030】

各バイナリーツリーの和 A ～ H は256のデータ入力を有し、各データは、論理0か論理1を表す2進数である。

【0031】

さらに、図3の方法を、マイクロプロセッサ20における最初の256個の処理要素 PE_i だけを用いて、 A ～ H の、8つのバイナリーツリーの和を計算する、特定の場について説明する。

【0032】

まず、ステップ40では、各バイナリーツリーの和の各入力データを、マイクロプロセッサ20のデータレジスタ DR_i の1つに格納する。例えば、バイナリーツリーの和 A 、 B 、 C 、 D 、 E 、 F 、 G 、 H の、入力データ A_i 、 B_i 、 C_i 、 D_i 、 E_i 、 F_i 、 G_i 及び H_i は、それぞれ、データレジスタ DR_i に格納する。その結果、データ A_1 、 B_1 、 C_1 、 D_1 、 E_1 、 F_1 、 G_1 及び H_1 が、第1の処理要素 PE_1 のデータレジスタ DR_1 に格納され、また、データ A_{256} 、 B_{256} 、...及び H_{256} が、バイナリーツリーの和 A ～ H を計算するのに使用される、最後の処理要素 PE_{256} のデータレジスタ DR_{256} に格納される。

40

【0033】

その後、フェーズ42において、各バイナリーツリーの和の第1イテレーションを計算する。

【0034】

50

より正確に云えば、ステップ44では、プロセッサ20が、処理要素 $PE_1 \sim PE_{256}$ を用いて、バイナリツリーの和A及びBの第1イテレーションを並列に計算する。

【0035】

ステップ44の間に、マイクロプロセッサ20は、データ A_{2j} をデータレジスタ $DR_{2(j-1)+1}$ へ、また、データ $B_{2(j-1)+1}$ をデータレジスタ DR_{2j} へ転送し、ここに、インデックスjは、1~128までの範囲の整数である。これらの転送動作は、僅か2つの命令、即ち、全てのデータ A_{2j} を移すための命令と、全てのデータ $B_{2(j-1)+1}$ を移すための命令、との2つの命令のみを用いて実行される。

【0036】

データ A_i と B_i は、両方ともデータレジスタ DR_i に記録されるために、僅か2つの転送命令を必要とするだけである。

10

【0037】

図4Aは、データレジスタ $DR_1 \sim DR_8$ 間で行われるデータ転送を図解している。図4A並びに図4B-4D及び図5A-5Dと図6では、マイクロプロセッサ20の要素に、図2におけるのと同じ参照符号を付してある。

【0038】

実行すべき算術又は論理演算、即ち、加算を各記号 ALU_i 内に示してある。 ALU_i に対応するデータレジスタ DR_i に格納される関連データを、 ALU_i の左及び右側に示してある。データレジスタ DR_i のデータから ALU_k への垂直部分を有する矢印は、 ALU_k を用いて算術又は論理演算を実行する前に、このデータをどのように移動させるべきかを示している。例えば図4Aにおいて、データ B_1 は、垂直部分を有する矢印によって ALU_2 に結ばれている。このことは、 ALU_2 を用いる加算を実行する前に、データ B_1 を、データレジスタ DR_2 へ転送しなければならないことを意味している。図4Aにおいて、データ A_1 から ALU_1 への水平方向の矢印は、データ A_1 がデータレジスタ DR_1 に既に格納されていることを示している。

20

【0039】

演算48の間に、ひとたびデータが転送されると、処理要素 $PE_{2(j-1)+1}$ は、データ $A_{2(j-1)+1}$ と A_{2j} との加算を実行して、その結果 $AA_{2(j-1)+1}$ をデータレジスタ $DR_{2(j-1)+1}$ に格納する。

【0040】

演算48の間には、これと並列に、処理要素 PE_{2j} が、データ B_{2j} と $B_{2(j-1)+1}$ の加算を実行し、その結果 BB_{2j} を、データレジスタ DR_{2j} に格納する。

30

【0041】

従って、演算48の間に、256個の処理要素 PE_i の全てが、同時に使用される。

【0042】

次に、ステップ50、52、54において、バイナリツリーの和CとD、バイナリツリーの和EとF、バイナリツリーの和GとH、の第1イテレーションを、それぞれ計算する。各ステップ50、52及び54の各ステップは、AとBの文字を、それぞれ、

- ステップ50ではCとDに、
- ステップ52ではEとFに、
- ステップ54ではGとHに、

40

置き換える以外は、ステップ44と同じである。

【0043】

その後、フェーズ60において、各バイナリツリーの和A~Hの、第2イテレーションを計算する。

【0044】

ステップ62では、バイナリツリーの和A~Dの第2イテレーションを並列に計算する。より正確に云えば、演算64の間に、次のデータを転送する。即ち、

- 結果 AA_{2j+1} をデータレジスタ $DR_{2(j-1)+1}$ に転送し、
- 結果 $BB_{2(j+1)}$ をデータレジスタ DR_{2j} に転送し、

50

- 結果 $CC_{2(j-1)+1}$ をデータレジスタ DR_{2j+1} に転送し、且つ
- 結果 DD_{2j} をデータレジスタ $DR_{2(j+1)}$ に転送する。

【0045】

これらのデータ転送を、図5Aに下向きの矢印によって示してある。

【0046】

そして、演算66の間に、各処理要素 PE_i にて次の加算を並列に実行する。即ち、

- 処理要素 $PE_{2(j-1)+1}$ は、結果 $AA_{2(j-1)+1}$ を結果 AA_{2j+1} に加えて、その結果 $AAA_{2(j-1)+1}$ をデータレジスタ $DR_{2(j-1)+1}$ に格納し

- 処理要素 PE_{2j} は、結果 $BB_{2(j+1)}$ を結果 BB_{2j} に加えて、その結果 BBB_{2j} をデータレジスタ DR_{2j} に格納し、

- 処理要素 PE_{2j+1} は、結果 $CC_{2(j-1)+1}$ を結果 CC_{2j+1} に加えて、その結果 CCC_{2j+1} をデータレジスタ $DR_{2(j+1)}$ に格納し、

- 処理要素 $PE_{2(j+1)}$ は、結果 DD_{2j} を結果 $DD_{2(j+1)}$ に加えて、その結果 $DDD_{2(j+1)}$ をデータレジスタ $DR_{2(j+1)}$ に格納する。

【0047】

その後ステップ68において、バイナリツリーの和E、F、G及びHの第2イテレーションを並列に計算する。ステップ68は、A、B、C、Dの文字を、それぞれ、E、F、G、Hに置き換えること以外は、ステップ62と同じである。ステップ68で行うデータ転送を、図5Bに下向きの矢印によって示してある。

【0048】

ステップ62及びステップ68では、256個の処理要素 PE_i が並列に使用される。

【0049】

次に、ステップ70で、バイナリツリーの和A~Hの第3イテレーションを並列に計算する。先ず、演算72の間に、第2イテレーションの結果のデータ転送を行い、次いで、演算74の間に、バイナリツリーの和の第3イテレーションの加算を実行する。

【0050】

演算72の間に、第2イテレーションの結果は、次のように転送される。即ち、

- 結果 $AAA_{2(j+1)+1}$ はデータレジスタ $DR_{2(j-1)+1}$ に転送され、
- 結果 $BBB_{2(j+2)}$ はデータレジスタ DR_{2j} に転送され、
- 結果 $CCC_{2(j+2)+1}$ はデータレジスタ DR_{2j+1} に転送され、
- 結果 $DDD_{2(j+3)}$ はデータレジスタ $DR_{2(j+1)}$ に転送され、
- 結果 $EEE_{2(j-1)+1}$ はデータレジスタ $DR_{2(j+1)+1}$ に転送され、
- 結果 FFF_{2j} はデータレジスタ $DR_{2(j+2)}$ に転送され、
- 結果 GGG_{2j+1} はデータレジスタ $DR_{2(j+2)+1}$ に転送され、
- 結果 $HHH_{2(j+1)}$ はデータレジスタ $DR_{2(j+3)}$ に転送される。

【0051】

演算74の間には、以下の加算を並列に実行する。即ち、

- 処理要素 $PE_{2(j-1)+1}$ が、結果 $AAA_{2(j+1)+1}$ を結果 $AAAA_{2(j-1)+1}$ に加え、その結果 $AAAA_{2(j-1)+1}$ をデータレジスタ $DR_{2(j-1)+1}$ に格納し、

- 処理要素 PE_{2j} が、結果 $BBB_{2(j+2)}$ を結果 $BBBB_{2j}$ に加え、その結果 $BBBB_{2j}$ をデータレジスタ DR_{2j} に格納し、

- 処理要素 PE_{2j+1} が、結果 $CCC_{2(j+2)+1}$ を結果 $CCCC_{2j+1}$ に加え、その結果 $CCCC_{2j+1}$ をデータレジスタ DR_{2j+1} に格納し、

- 処理要素 $PE_{2(j+1)}$ が、結果 $DDD_{2(j+3)}$ を結果 $DDDD_{2(j+1)}$ に加え、その結果 $DDDD_{2(j+1)}$ をデータレジスタ $DR_{2(j+1)}$ に格納し、

- 処理要素 $PE_{2(j+1)+1}$ が、結果 $EEE_{2(j-1)+1}$ を結果 $EEEE_{2(j+1)+1}$ に加え、その結果 $EEEE_{2(j+1)+1}$ をデータレジスタ $DR_{2(j+1)+1}$ に格納し、

10

20

30

40

50

- 処理要素 $PE_{2(j+2)}$ が、結果 FFF_{2j} を結果 $FFF_{2(j+2)}$ に加え、その結果 $FFF_{2(j+2)}$ をデータレジスタ $DR_{2(j+2)}$ に格納し、
- 処理要素 $PE_{2(j+2)+1}$ が、結果 GGG_{2j+1} を結果 $GGG_{2(j+2)+1}$ に加え、その結果 $GGG_{2(j+2)+1}$ をデータレジスタ $DR_{2(j+2)+1}$ に格納し、
- 処理要素 $PE_{2(j+3)}$ が、結果 $HHH_{2(j+1)}$ を結果 $HHH_{2(j+3)}$ に加え、その結果 $HHH_{2(j+3)}$ をデータレジスタ $DR_{2(j+3)}$ に格納する。

【0052】

従って、演算74の間には、256の全ての処理要素 PE_i が同時に使用される。

【0053】

演算72の間に行うこれらのデータ転送を、図6に下向きの矢印によって示してある。

【0054】

ステップ80では、各バイナリツリーの和A~Hの次のイテレーションを、並列に計算する。この段階における処理要素の数256は、実行すべき並列加算の数よりも大きい。従って、各バイナリツリーの和の第4~第8イテレーションの計算については、詳細に述べる必要はないものと思料する。實際上、8つのバイナリツリーの和A~Hの第4イテレーションを並列に計算するためには、8つのバイナリツリーの和を計算するために最初に割り当てられた256個の処理要素よりも少ない、128個の処理要素 PE_i を必要とするだけである。

【0055】

図3の方法では、バイナリツリーの和Aの第1イテレーションを計算するために最初に割り当てられた幾つかの処理要素 PE_i は、バイナリツリーの和の第2及び第3イテレーションを計算している間に、別のバイナリツリーの和のイテレーションを並列に計算するために使用される。例えば、処理要素 PE_3 は、バイナリツリーの和Aの第1イテレーションを計算するために使用され、そして、バイナリツリーの和Aの第2イテレーションを計算している間に、バイナリツリーの和Cの第2イテレーションを計算するために使用される。処理要素 PE_3 はまた、バイナリツリーの和Aの第3イテレーションを計算している間に、バイナリツリーの和Cの第3イテレーションを計算するためにも使用される。

【0056】

従って、ここで述べた特定の実施態様では、処理要素 $PE_1 \sim PE_{256}$ の全てが、フェーズ42、フェーズ60、そしてステップ70の間に完全に使用される。従って、8つのバイナリツリーの和の計算は、異なる演算ツリーからの演算を混合又はインターリーブするために、高速になる。

【0057】

第1及び第2演算ツリー、例えばバイナリツリーの和A及びBを計算するために使用する処理要素は、インターリーブされる。これは、最初の処理要素 PE_1 と最後の処理要素 PE_{256} を除いて、第1ツリーの計算のために使用する各々の処理要素が、第2ツリーの計算のために使用する2つの処理要素の間に位置付けられることを意味する。このような機構により、データ転送の振幅が低減し、処理速度が速くなる。

【0058】

多くの追加の実施態様が可能である。例えば、本発明による方法は複数のバイナリツリーの和を並列に計算することに限定されず、各ノードを、1つの命令に答えてALUによって実行させることのできる算術又は論理演算とする、もっと一般的な演算ツリーにも適用することができる。例えば、各ノードは、X-OR(排他的論理和演算)または減算を計算するように設計することもできる。

【0059】

SIMDマイクロプロセッサは、1つの命令で複数のデータを並列に実行するためにだけ設計されている。従って上記の方法は、SIMDマイクロプロセッサで実行させる際に、複数の同一演算ツリーを並列に計算するためだけに用いることができる。しかしながら

10

20

30

40

50

、このような制約は、本発明による方法をMIMD (Multiple Instruction Multiple Data) マイクロプロセッサで実施する場合には存在しない。従って、MIMD マイクロプロセッサで実施する場合には、本発明による方法は、異なる演算ツリーを並列に計算するために用いることができる。例えば、バイナリツリーの和及びバイナリツリーの減算は、同時に計算することができる。

【0060】

上記の方法は、2つの入力端を有するノードを備えたバイナリツリーに限定されるものでなく、例えば、2つ以上の入力端を有するノードを備えた演算ツリーにも適用することができる。

【図面の簡単な説明】

【0061】

【図1】 模範的な演算ツリーの概略図である。

【図2】 図1の演算ツリーを計算するのに用いられるSIMD (単一命令/複数データ) マイクロプロセッサの概略図である。

【図3】 図2のマイクロプロセッサを用いて、図1の演算ツリーのような8つの演算ツリーを並列に計算する方法のフローチャートである。

【図4】 図3の方法の計算ステップの説明図である。

【図5】 図3の方法の他の計算ステップの説明図である。

【図6】 図3の方法のさらに他の計算ステップの説明図である。

【図1】

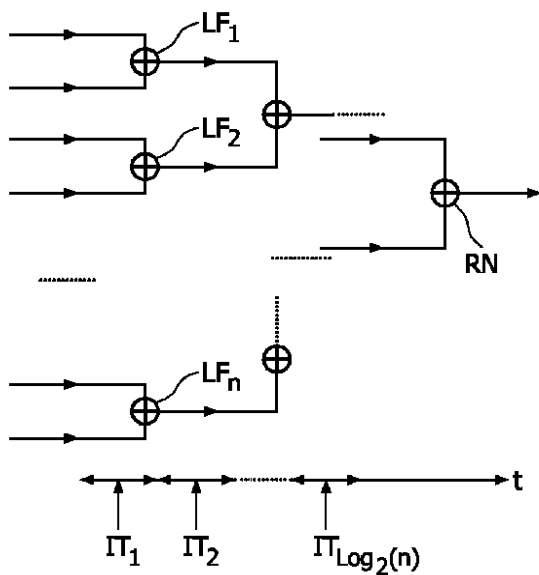


FIG. 1

【図2】

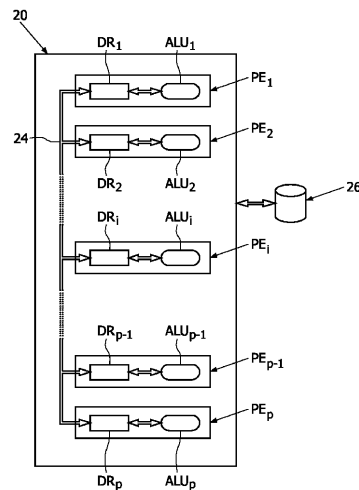


FIG. 2

【 図 3 】

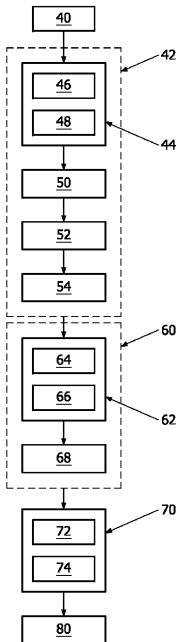


FIG. 3

【 図 4 C 】

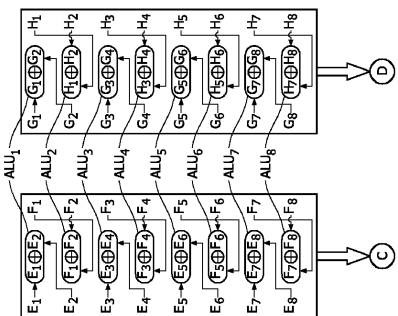


FIG. 4C

【 図 4 D 】

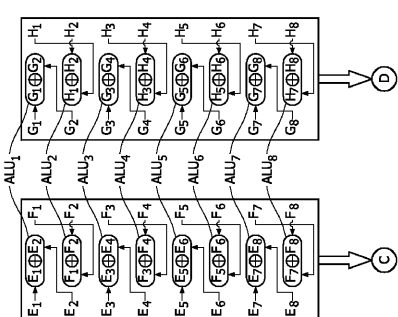


FIG. 4D

【 図 4 A 】

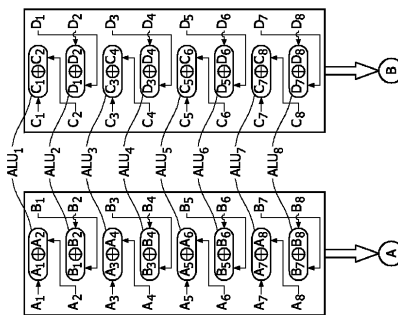


FIG. 4A

【 図 4 B 】

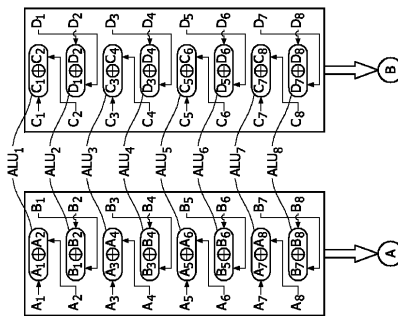


FIG. 4B

【 図 5 A 】

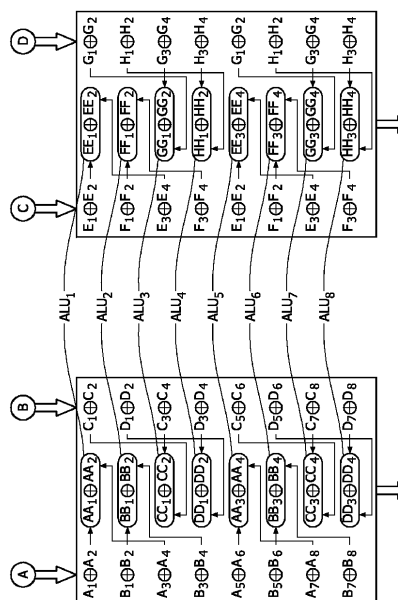


FIG. 5A

FIG. 5B

【 図 5 B 】

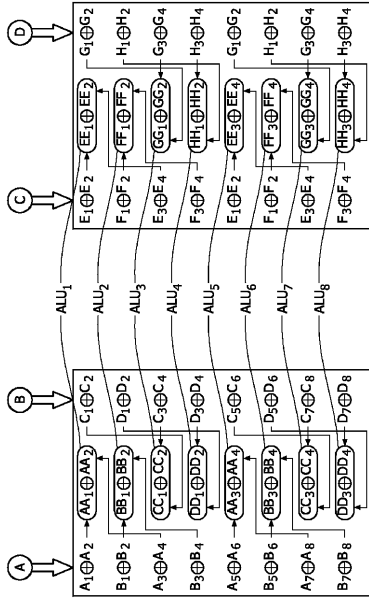


FIG. 5A

FIG. 5B

【 図 6 】

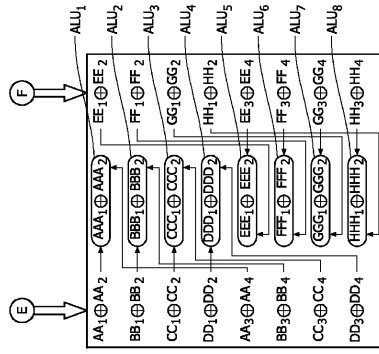


FIG. 6

【 国際調査報告 】

INTERNATIONAL SEARCH REPORT

International application No

PCT/IB2005/054202

A. CLASSIFICATION OF SUBJECT MATTER G06F7/50		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols) G06F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used) EPO-Internal, WPI Data, INSPEC, COMPENDEX, IBM-TDB		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 01/75587 A (INTEL CORPORATION; KOLAGOTLA, RAVI; ALDRICH, BRADLEY, C; ANDERSON, WIL) 11 October 2001 (2001-10-11) figures 2-6 page 6 - page 9	1-7
X	US 2002/062331 A1 (ABDALLAH MOHAMMAD A ET AL) 23 May 2002 (2002-05-23) figures 1-13 PADDH operation/apparatus	1-7
	-/--	
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C.		<input checked="" type="checkbox"/> See patent family annex.
* Special categories of cited documents :		
A document defining the general state of the art which is not considered to be of particular relevance		*T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
E earlier document but published on or after the international filing date		*X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
L document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)		*Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
O document referring to an oral disclosure, use, exhibition or other means		*Z* document member of the same patent family
P document published prior to the international filing date but later than the priority date claimed		
Date of the actual completion of the international search	Date of mailing of the international search report	
24 March 2006	10/04/2006	
Name and mailing address of the ISA/ European Patent Office, P. B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Authorized officer Pfab, S	

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2005/054202

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	INOUE T ET AL: "AN AREA-EFFECTIVE DATAPATH ARCHITECTURE FOR EMBEDDED MICROPROCESSORS AND SCALABLE SYSTEMS" IEICE TRANSACTIONS ON ELECTRONICS, ELECTRONICS SOCIETY, TOKYO, JP, vol. E84-C, no. 8, August 2001 (2001-08), pages 1014-1020, XP001107636 ISSN: 0916-8524 chapter 3: SIMD adder	1-7
A	EP 1 308 835 A (BROADCOM CORPORATION) 7 May 2003 (2003-05-07) cited in the application the whole document	1-7

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/IB2005/054202

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
WO 0175587	A	11-10-2001	AU 4976701 A CN 1422402 A EP 1269308 A2 TW 539989 B	15-10-2001 04-06-2003 02-01-2003 01-07-2003
US 2002062331	A1	23-05-2002	US 6377970 B1	23-04-2002
EP 1308835	A	07-05-2003	NONE	

フロントページの続き

(81)指定国 AP(BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), EP(AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW