(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3573808号
(P3573808)
(45)発行日　平成16年10月6日(2004.10.6)　(24)登録日　平成16年7月9日(2004.7.9)

(51)Int.Cl.[7]　F I
G06F 7/50　G06F 7/50　M
G06F 7/52　G06F 7/52　310A

請求項の数 6　(全 15 頁)

(21)出願番号　特願平6-317609
(22)出願日　平成6年11月28日(1994.11.28)
(65)公開番号　特開平7-200261
(43)公開日　平成7年8月4日(1995.8.4)
審査請求日　平成13年11月21日(2001.11.21)
(31)優先権主張番号　158,646
(32)優先日　平成5年11月29日(1993.11.29)
(33)優先権主張国　米国(US)

(73)特許権者　398038580
ヒューレット・パッカード・カンパニー
HEWLETT-PACKARD COMPANY
アメリカ合衆国カリフォルニア州パロアルト　ハノーバー・ストリート　3000
(74)代理人　100081721
弁理士　岡田　次生
(72)発明者　ルビー・ベイ－ロー・リー
アメリカ合衆国カリフォルニア州ロスアルトスヒルズアサートン・コート　12933
(72)発明者　ジョエル・デイヴィッド・ラム
アメリカ合衆国コロラド州フォートコリンズ　エットン・ドライヴ　1819
最終頁に続く

(54)【発明の名称】論理演算装置

(57)【特許請求の範囲】
【請求項１】
ｉ＝０～Ｎ－１であり、Ｚ0が最下位ビットであり、ＺN-1が最上位ビットであるとした場合、ビットＸｉを有するＸワード及びビットＹｉを有するＹワードの内容に演算を施して、ビットＺｉを有する結果ワードを発生するための装置であって、
前記Ｘ、Ｙ、及び結果ワードを複数のサブ・ワードに区分化し、前記Ｙ及び結果ワードのサブ・ワードの１つが前記Ｘワードの各サブ・ワードに対応するように、前記Ｘ、Ｙ、及び結果ワードを複数のサブ・ワードに選択的に区分化するための手段と、
ｍを０でない整数とした場合、第１の命令に応答して、各Ｘサブ・ワードを2^mで割った値と対応するＹサブ・ワードの合計を発生し、その結果によって結果ワードの前記対応するサブ・ワードが決まるようにするための手段と、から構成され、
合計を発生する手段が、０番目から［Ｎ－１］番目の［μ＋１］対１マルチプレクサに対応する、０番目から［Ｎ－１］番目の単一ビット加算器を備え、各単一ビット加算器は、対応するマルチプレクサによって供給されるビットおよびＹワードの対応するビットを加算し、μはｍ以上であり、
ｐ番目のマルチプレクサの入力は、ＸワードのビットＸｐからＸｐ＋μに接続され、出力は、Ｘｐ＋ｍに接続され、
選択的に区分化するための手段は、サブ・ワードの１つの加算器で発生したキャリアビットが、サブ・ワードの他の１つに伝搬することを選択的に防止するブロック回路と、サブ・ワードの１つのビットＸｉが、サブ・ワードの他の１つの加算器に対応するマルチプレ

クサへ伝搬するのを選択的にブロックするゲートとを備える装置。
【請求項２】
第２の命令に応答して、2^mで割った前記Ｘワードの各サブ・ワードと前記Ｙワードの対応するサブ・ワードの差を発生し、その結果によって前記結果ワードの前記対応するサブ・ワードが決まるようにするための手段がさらに設けられている、請求項１に記載の装置。
【請求項３】
前記2^mで割った結果に丸め誤差が生じた場合、前記Ｘワードの各前記サブ・ワードを割る結果が、次に高位の奇数の整数に丸められる、請求項１または２に記載の装置。
【請求項４】
単一ビット加算器は、
それぞれ前記Ｙワードから導き出されるビットを受信し、Ｙpから導き出される前記ビットを受信するものがｐ番目のＹ入力手段と称される、複数のＹ入力手段と、
それぞれ前記Ｘワードから導き出されるビットを受信し、各前記Ｙ入力手段に対応するものが１つ含まれており、ｐ番目のＹ入力手段に対応するものがｐ番目のＸ入力手段と称される、複数のＸ入力手段と、を備え、
単一ビット加算器の複数の加算ステージは、直列に接続されており、各加算ステージは、前記Ｘ入力手段で受信した１つ以上のビットと前記Ｙ入力手段で受信した対応するビットとを加算するための手段と、１つの加算ステージから直列に接続された次の加算ステージに桁上げビットを伝播するための手段とから構成され、Ｙpに演算を施すものがｐ番目の加算ステージと称され、前記区分化手段は、前記加算ステージが前記Ｘワードの異なるサブ・ワードのビットに対する演算が行われる場合、強制的に前記桁上げビットを実施される演算によって求められる値にする手段を備える請求項１から３のいずれか１項に記載の装置。
【請求項５】
値（Ｘp ＯＲ Ｘp+1 ＯＲ…Ｘp+m）を有する信号を発生し、Ｙpが前記サブ・ワードの１つの最下位ビットである場合、前記信号を前記ｐ番目の加算ステージに接続するための手段がさらに設けられている、請求項４に記載の装置。
【請求項６】
前記区分化手段に、前記桁上げビットが前記求められた値にされる前は、前記桁上げビットの１つが「１」であったことを示す信号を発生する手段がさらに設けられている、請求項４または５に記載の装置。
【発明の詳細な説明】
【０００１】
【産業上の利用分野】
本発明は、コンピュータに関するものであり、とりわけ、それに用いられる論理演算装置に関するものである。
【０００２】
【従来の技術】
コンピュータには、通常、ある最大ビット数の数を加算する加算器が含まれている。マイクロプロセッサ等においては、３２ビット及び６４ビットの長さのワードに関する加算器が一般的である。これらの加算器は、はるかに少ないワードに関する演算も行うが、そうする場合、加算器に含まれる大部分の論理回路は遊休状態になる。例えば、６４ビット加算器は、対応する６４ビット・ワードの最下位部分に２つの８ビット・ワードのそれぞれを配置し、それから、６４ビット・ワードを加算することによって、２つの８ビット・ワードの加算に利用することが可能である。この加算中、ワードのそれぞれにおける７つの上位バイトの加算に関連した論理回路要素が、事実上遊休状態になる。従って、この演算中、加算器の能力の７／８が無駄になっている。
【０００３】
多数の小ワードに対する演算を必要とする計算は、マルチ・メディア・データ処理におい

てしばしば見受けられる。イメージは、通常、各ピクセルが、論理演算装置の最大ワード・サイズよりかなり小さいワードによって表される、ピクセル・アレイとして表現される。グレイ・スケールのイメージは、一般に、そのイメージの対応する位置における光の強度を表した１バイトの整数からなるアレイによって表現される。同様に、サウンド・トラックは、時間の関数としてサウンド・トラックの強度を表す１または２バイトの整数からなるアレイによって表現される。従って、マルチ・メディア・データ処理は、典型的な汎用コンピュータに組み込まれた論理演算装置の計算能力を十分に活用していない。

【０００４】

【発明が解決としようとする課題】

このタイプのデータ処理は、論理演算装置（以下、適宜ＡＬＵ　という）の能力を十分に活用していないだけでなく、ＡＬＵ　においてその演算を行う前に、データのパック及びパック解除を行うことが必要になるので、結果として、非能率的でもある。例えば、記憶空間は、常に不足しているので、イメージのピクセルは、一般に、ワードにパックされる。コンピュータの基本ワード・サイズが３２ビットの場合、グレイ・スケール・イメージのピクセルは、ワード毎に４つずつパックすることが可能である。イメージ内の各ピクセル毎に実施しなければならない演算について考えてみることにする。演算の実施に必要な時間に加えて、プログラムは、各計算の前に、ピクセル情報のパック解除を行い、さらに結果を再パックしなければならない。これらのパック及びパック解除操作によって、演算効率がさらに低下する。

【０００５】

マルチ・メディア演算において直面する計算時間は、法外なものになる可能性がある。従って、和イメージ指令の実行と、和イメージの完成時との間の時間を短縮するため、特殊並列コンピュータ・アーキテクチャがよく用いられる。１つのピクセルまたはサウンド・サンプルに対して実施されるマルチ・メディア処理演算は、他のピクセルまたはサウンド・サンプルに対して実施される演算とは独立している場合が多く、従って、演算は、順番に関係なく、並列に実施することができる。Ｍ　個の加算器を備えたコンピュータは、原理的に、メモリと加算器の間におけるピクセルの移動が障害にならない場合、１／Ｍ$^{t}_{h}$ の時間で結果を得ることが可能である。従って、複数の加算を並列に実施することが可能なコンピュータ・アーキテクチャを備えることが有利である。あいにく、こうした追加ＡＬＵ　及びそれらの制御に必要なハードウェアを設けるコストは、けた外れになる場合が多い。

【０００６】

大データ集合に対して実施されることが多い計算クラスの１つが、２進小数による乗算である。こうした計算は、フィルタリング操作、及び、データ圧縮及び圧縮解除操作において一般的である。これらの場合、乗算は、定数によって行われる。

【０００７】

一般に、本発明の目的は、改良された論理演算装置を提供することにある。

【０００８】

本発明のもう１つの目的は、論理演算装置の幅よりも狭いワードを伴う複数の演算が処理される場合に、効率の高い演算を行う論理演算装置を提供することにある。

【０００９】

本発明のさらにもう１つの目的は、２進小数を伴う複数の乗算の計算に適した論理演算装置を提供することにある。

【００１０】

当業者には、後述の本発明の詳細な説明及び添付の図面から本発明の以上の目的及びその他の目的が明らかになるであろう。

【００１１】

【課題を解決するための手段】

本発明は、ｉ＝０～Ｎ－１であるとした場合、ビットＸｉを有するＸワード及びビットＹｉを有するＹワードの内容に演算を施して、ビットＺｉを有する結果ワードを発生するた

めの論理演算装置である。Ｘ及びＹワードは、サブ・ワードに区分化することが可能である。Ｚ０　は、サブ・ワードの１つの最下位ビットであり、ＺＮ　－１は、サブ・ワードの１つの最上位ビットである。論理演算装置は、マスク・ワードに応答して、Ｘ　、Ｙ　、及び、結果ワードを複数のサブ・ワードに区分化するが、Ｘワードの各サブ・ワードに、Ｙ及び結果ワードのサブ・ワードの１つが対応する。本発明では、各Ｘサブ・ワードを2^m　で割った値と対応するＹサブ・ワードの合計を発生することによって、前記結果ワードの前記対応するサブ・ワードに納められる結果が生じる。ここで、ｍは非負整数である。

【００１２】

本発明の実施例の１つは、オーダ・シーケンスをなすように接続されたＮ個の単一ビット加算器から構成される。各単一ビット加算器ステージは、Ｘワードから第１のビット信号を受信し、Ｙワードから第２のビット信号を受信するようになっており、シーケンスにおけるｉ番目の単一ビット加算器は、ＹレジスタのビットＹｉ　に接続される。各単一ビット加算器は、第１及び第２のビット信号と桁上げ入力信号を加算して、和信号及び桁上げ出力信号を発生し、単一ビット加算器によって発生した和信号は、ＹレジスタのビットＹｉ　に接続され、ビット信号Ｚｉ　が生じることになる。ｉ＝０～Ｎ－１の場合、Ｙｉ　に接続された単一ビット加算器の桁上げ出力信号は、Ｙｉ＋１　に接続された単一ビット加算器の桁上げ入力に接続される。論理演算装置には、Ｎ個のマルチプレクサも含まれており、各加算器ステージに、１つのマルチプレクサが接続され、ｍを指定する信号に応答する。ビットＹｐ　及びＹｐ＋ｍ　がＹワードの同じサブ・ワードに含まれている場合、ｐ番目の加算器ステージに接続されたマルチプレクサは、このステージをＸサブ・ワードのビットＸｐ＋ｍ　に接続する。残りのビットに関して行われる接続は、Ｘサブ・ワードのそれぞれが、符号付き整数を表しているか、符号無し整数を表しているかによって決まる。Ｘサブ・ワードが符号付き整数を表している場合、残りのビットは、適合するＸサブ・ワードの最上位ビットに接続される。そうでなければ、残りのビットは、「０」に接続される。

【００１３】

【実施例】

本発明は、２つの観測に基づくものである。まず、２進小数から成る定数による乗算は、加数の１つに対して実施される右シフトを伴う複数の加算に相当する。２進小数ｆと数ｒの積について考えてみることにする。ｉ＝０～Ｂ－１として、ｆのビットがｂｉ　であれば、積ｐは、（１）式のように書くことができる。

【００１４】

【数１】

$$p = \sum_{i=0}^{b-1} bi * r \qquad \cdots (1)$$

【００１５】

これによって、積を、次の（２）式の形に書くことの可能な、複数の総和に分解することが可能になる。

【００１６】

【数２】

$$X + X / 2^k \qquad \cdots (2)$$

【００１７】
ここで、ｋは、正の整数であり、ｘは、ｒ、または、このタイプの分解における先行する演算の結果に当たる整数である。例えば、２進数１．１０１のｒ倍は、次の（３）式のように書くことが可能である。
【００１８】
【数３】

$$p = r + r/2 + r/8 \quad \cdots (3)$$

【００１９】
この演算は、例えば、下記の（４）式および（５）式のように、２つの演算で実施することが可能である。
【００２０】
【数４】

$$X = r + r/2 \quad \cdots (4)$$

【００２１】
【数５】

$$p = r + r/8 \quad \cdots (5)$$

【００２２】
２進小数が定数の場合、コンパイラは、適合する命令を発生して、乗算を必要な演算に分解することが可能である。これらの演算は、それぞれ、オペランドの一方がｍビットだけ右にシフトしたレジスタの内容である、２つのオペランドの和と表現して書くことが可能である。従って、「ｍだけ右にシフトし、加算する」形の命令を実施するＡＬＵ　を備えていることが有利である。
【００２３】
第２の観測は、別様の従来の加算器を利用して、単一マシン・サイクルで複数の小オペランドの加算が行えるようにする、単純なハードウェアの修正に関するものである。原理上、加算器は、それぞれ、加算器の全幅より狭いサイズのワードに対する演算を行う、複数の区分に分割することが可能である。各区分の和は、加算器の幅以下でなければならない。各マシン・サイクル毎に、これらのより小さいワードに対して、複数の加算演算を実施することが可能である。さらに、加算器を従来の加算器として利用することも可能である。
【００２４】
論理演算装置（以下、ＡＬＵ　という）の幅が任意のサイズのワードに分割される本発明の実施例は、可能であり、さらに詳細に後述するが、本発明の好適な実施例によれば、可能性のあるほんの少数のワード組み合わせに分割することが可能な加算器が得られる。例えば、３２ビットの加算器は、長さが１バイトまたは２バイトのワードを加算可能にする区分に分割することが望ましい。下記の説明を単純化するため、ＡＬＵ　は、半分に分割されるものと仮定する。この場合、各マシン・サイクル毎に、２つのハーフ・ワード加算または１つのフル・ワードを実施することが可能である。
【００２５】

図１を参照すると、本発明によるＡＬＵ　１０は、２つのＮビット・オペランドのワード１２及び１４を受諾する。ｉ＝０～Ｎ－１の場合、第１のオペランドのビットは、Ｘｉで表され、第２のオペランドのビットは、Ｙｉ　で表される。オペランドは、コンピュータのＣＰＵ　におけるレジスタの２つに記憶されるのが普通である。全幅モードで演算する場合、ＡＬＵ　１０は、ｍビットだけ右にシフトしたｘとｙの２の補数和を有するＮビットの結果ワードとしての出力ワード１６を発生する。ＡＬＵ　１０の出力のビットは、下記の説明においてＺｉで表される。加算の結果は、一般に、ＣＰＵ　レジスタの１つに戻される。

【００２６】

下記の説明において、各種ワードのビットは、最下位から最上位まで番号が付けられている。すなわち、Ｘ０　は、オペランドＸ　の最下位ビットであり、ＸＮ－１　は、オペランドＸ　の最上位ビットである。同じ規則が、Ｙ及びＺワードにも用いられる。

【００２７】

本発明によれば、各オペランドを複数のサブ・ワードに分割することができる。単純化のため、本発明の説明は、まず、各オペランドから部分オペランドへの単一分割に関連して行うことにする。この場合、Ｘオペランドの第１のｑビットＸ０　～Ｘｑ－１　は、Ｘワードの最初の部分ワード１８のビットであり、残りのビットＸｑ　～ＸＮ－１　は、Ｘワードの第２の部分オペランドのワード１７のビットである。Ｙオペランドは、同様に部分ワード１９及び２０に分割される。このモードの場合、ビットＺ０　～Ｚｑ－１　は、ｍビットだけシフトした部分オペランド１８と部分オペランド２０の和のビットであり、ビットＺｑ　～Ｚｑ－１　は、それぞれ、ｍビットだけシフトした部分オペランドのワード１７と部分オペランド１９の和のビットである。

【００２８】

実用上の理由から、ｍは、一般に、４以下の整数に制限される。従来のＡＬＵ　には、加算前に、Ｙオペランドの２の補数を発生する機能が含まれている。さらに詳細に後述するように、本発明には、加算前に、Ｘオペランドに対する演算を行う新規のシフタが含まれている。このシフタの導入によって、ＡＬＵ　の性能が劣化しないことを保証するため、シフト演算に固有の遅延は、２の補数ハードウェアを通過する際にＹオペランドが遭遇する遅延程度でなければならない。実際、これによって、ｍは上述の値に制限される。

【００２９】

本発明の望ましい実施例の場合、部分オペランドは、符号付き整数である。オーバ・フローまたはアンダ・フローの場合、その結果は、それぞれ、問題となる長さの符号付き整数に関する最大値または最小値でクランプされる。例えば、Ｎ＝３２で、ハーフ・ワード・シフト（すなわち、１６ビット）及び加算演算が実施される場合、結果は、それぞれ、１６進数の７ＦＦＦ及び８０００でクランプされる。

【００３０】

下記の説明を単純化するため、まず、Ｘオペランドの位置１つ分だけ右にシフトし、次に、シフトしたオペランドを対応するＹオペランドに加算するＡＬＵ　に関連して、本発明の解説を行うことにする。さらに説明を単純化するため、ＡＬＵ　が、加算時に桁上げ伝播を利用する複数の単一ビット加算器から構成されるものであると仮定する。次に、２つのサブ・オペランドに分割することが可能な、そして、加算前に、Ｘオペランドの位置１つ分だけ右にシフトすることが可能な、本発明によるＡＬＵ　３０の一部に関するブロック図である図２を参照されたい。２つのサブ・オペランド間の境界が、ＡＬＵ　３０のビット位置ｋに生じる。すなわち、ビットｋは、１つのサブ・オペランド集合の最上位ビットであり、ビットｋ＋１は、もう１つのサブ・オペランド集合の最下位ビットである。ＡＬＵ　３０の加算セクションは、従来の桁上げ伝播加算器と同様のやり方で、１ビット加算器ステージのアレイから構成されている。典型的な加算手段としての単一ビット加算器が、３１～３５で示されている。本発明では、ステージの減結合によって、ＡＬＵ　が部分ワードに並列加算を施すことが可能になる。各単一ビット加算器は、２つのビット、すなわち、Ｘオペランドから導き出されたビット及びＹオペランドから導き出されたビット

と、ｉ番目のステージに関した、Ｃｉ－１　で表示される、加算器の先行ステージからの桁上げビットを加算して、和ビット及び新しい桁上げビットを発生する。単一ビット加算器３３及び３２で示す２つのステージは、それぞれ、部分オペランド１８及び２０の最上位ビット、及び、部分オペランド１７及び１９の最下位ビットの加算に用いられる、単一ビット加算器である。例えば、単一ビット加算器３３は、ビットＣｋ－１　、Ｘｋ　、及び、Ｙｋ　を加算して、和ビットＳｋ　及び桁上げビットＣｋ　を発生する。以下の説明において、Ｙｐ　に対して機能する加算器のステージは、加算器のｐ　番目のステージと呼ばれる。従来の加算器の場合、各ステージからの桁上げビットは、各ステージの桁上げビット入力を、１ビット加算器のアレイにおけるその前のステージの桁上げビット出力に接続することによって、次のステージに伝播する。

【００３１】

本発明の場合、２つの部分オペランドを分割する境界の直前のステージからの桁上げビットが、ブロック回路３７に接続される。ＡＬＵ　３０が単一ワードとして扱われるレジスタの全内容であるＮビットオ・ペランドのワード１２及び１４に対して演算を施す従来の加算器として用いられる場合、ブロック回路３７は、単一ビット加算器３３の桁上げ出力を単一ビット加算器３２の桁上げ入力に接続する。ＡＬＵ　３０が、各レジスタのビットｋとｋ＋１の間の部分ワード境界に関して２つの並列加算を実施するために利用される場合、ブロック回路３７は、単一ビット加算器３３からの桁上げビットを、加算時に強制的に０にする。これは、境界信号Ｍｋ　によって制御される。他の全ての単一ビット加算器の桁上げ出力は、ＡＬＵ　３０の残りのステージにおいて、従来のやり方で接続される。従って、桁上げビットは、特定の部分オペランドに対する演算を行うＡＬＵ　３０の各セクション内において、従来のやり方で伝播する。各加算器ステージからの和ビットは、Ｚｋ　で表示された出力ポートの対応するビットに接続される。

【００３２】

ＡＬＵ　３０に関する以上の説明は、オペランドの加算が行われると仮定したものであった。ブロック回路３７は、加算器がブロック回路の境界に関して副加算器に分割される加算時に、桁上げビットを０に置き換える。加算器が、加算器が同様に分割される２の補数減算にも利用される場合、桁上げビットは、０ではなく、強制的に１にしなければならない。図２に示すブロック回路３７は、「０」または「１」の値を有する入力Ｆを供給することによって、加算と減算の両方を実施する。ブロック回路３７において、境界が活性状態の場合、Ｆの値が、次のステージに送られる値である。境界が、不活性状態の場合、ブロック回路３７は、桁上げビットＣｋ　を次のステージに伝送するだけである。

【００３３】

従来の加算器の場合、符号付きオーバ・フロー信号は、オペランドの最上位ビット及び結果の最上位ビットから計算される。符号無しオーバ・フローは、オペランドの最上位ビットに対して演算を行う最上位ビット加算器からの桁上げビットから計算される。この特徴を部分オペランドのそれぞれについて実施すべき場合、各部分オペランドの加算によるオーバ・フロー信号は、適合するオーバ・フロー回路に接続される。本発明の実施例の１つでは、オーバ・フロー信号には、ＯＲ演算が施され、結果信号は、オーバ・フローの検出に利用される。このビットは、トラップのトリガに利用することもできるし、あるいは、単一ビット・レジスタの内容とのＯＲ演算が可能である。後者の場合、プログラムはレジスタの内容をチェックし、レジスタに行った最終チェック以降の演算によって、オーバ・フローが生じたか否かを確認する。

【００３４】

上述のように、ＡＬＵ　３０は、加算前に、Ｘレジスタの内容を右に位置１つ分だけシフトさせるように設計されている。これは、Ｘオペランドのビットを適正な単一ビット加算器ステージに接続する２対１マルチプレクサによって、信号ｍに応答して実施される。典型的な多重化手段としてのマルチプレクサが、図２の４１～４５に示されている。マルチプレクサ制御信号が真の場合、Ｘｐ　は、ステージ（ｐ－１）に関する単一ビット加算器に接続される。制御入力が偽の場合、ビットはステージｐに接続される。

【００３５】
ＡＬＵ　が符号付き整数に対して演算を行っている場合、各オペランドの最上位ビットは、符号ビットであるため、その新しい位置にコピーするだけでなく、そのもとの位置で複製しなければならない。従って、２つのサブ・ワードの境界におけるマルチプレクサは、サブ・オペランドの内部にだけしか存在し得ないＸ　ビットに接続されたマルチプレクサとは異なっていなければならない。マルチプレクサ４３は、こうしたマルチプレクサである。ｍが真であって、Ｘレジスタが、ステージＸｋ　がオペランドの最上位ビットになるように区分化されている場合、マルチプレクサ４３は、Ｘｋ　をステージｋに接続する。Ｘレジスタがこのように区分化されていない場合、すなわち、Ｍｋ　が偽の場合には、マルチプレクサ４３は、他のマルチプレクサと同様に動作する。
【００３６】
本発明の望ましい実施例の場合、オペランドの境界におけるステージの位置は、そのビットがＭｐ　で表示されるマスクによって指定される。ビットＭｐ　は、各部分オペランドの最上位ビットの位置を指定する。以下の説明において、このマスクは、境界マスクと呼ぶことにする。マスクのビットは、ＡＬＵ　のレジスタに記憶しておくこともできるし、あるいは、加算器が配置されたプロセッサの命令復号化回路要素によって実行されている命令から直接発生させることも可能である。
【００３７】
本発明の上述の実施例は、桁上げ伝播を利用して、単一ビット加算器から構成された加算器に関連して説明してきたが、当業者には明らかなように、本発明の教示は、多種多様な加算器構成に適用することが可能である。加算器を副加算器に分割することによって、可能性のある各サブ・オペランドの最上位ビットが、副加算器の境界上に位置するようにすることが可能な任意の加算器構成は、本発明において機能するように構成することが可能である。加算器は、問題となる副加算器に区分化される場合、副加算器からの桁上げを中断する回路を導入することによって、変更することが可能である。
【００３８】
本発明の好適な実施例の場合、遅延が少なくなるので、桁上げ先見アーキテクチャが利用される。桁上げ先見加算器の場合、桁上げ発生回路要素は、加算器の各ビットに対応する伝播信号及び発生信号を送り出す。これらの信号を上述の桁上げビットと同様に利用することによって、加算器を並列サブ・ワード加算器に分割できるようにすることが可能である。ステージｋがサブ・ワードの結果の最上位ビットに演算を施すように、加算器を分割すべき場合について考えてみることにする。図２に示すブロック回路３７のようなブロック回路を桁上げ発生論理回路に挿入することによって、実施されている演算のタイプ、すなわち、加算または減算に基づいて、伝播ビット及び発生ビットを強制的に適合する値にすることが可能になる。加算器が、ステージｋにおいて分割されないワードに対して用いられている場合、ブロック回路は、ステージｋに対応する伝播及び発生ビットの値を変更しない。
【００３９】
図２に示す本発明の実施例は、Ｘオペランドを位置１つ分だけ右にシフトするように設計されている。しかし、以上の説明から明らかなように、同じ原理を利用して、任意の量ｍ＜μだけＸオペランドを右にシフトさせるＡＬＵ　を提供することも可能である。この場合、４１～４５に示す２対１マルチプレクサの代わりに、（μ＋１）対１マルチプレクサが用いられる。信号ｍ　は、（μ＋１）の可能性のある位置のうちどの位置が指定されているかを指定することができなければならない。加算器のステージｐは、オペランドの内部ビットに対する演算だけしか行うことができない場合、それに接続されたマルチプレクサは、ステージｐをＸｐ＋ｍ　に接続する。しかし、ステージｐが、可能性のあるサブ・ワードの境界のｍステージ内にある場合、ステージｐに関連したマルチプレクサは、シフト信号ｍだけでなく、マスク・ビットも検査しなければならない。マスク・ビットが、境界がステージｋにおいて活性状態にある、すなわち、Ｘｋ　がサブ・ワードの最上位ビットであることを表している場合について、考えてみることにする。（ｋ－ｍ）　＜ｐ＜ｋ

の場合、ステージｐに関連したマルチプレクサは、Ｘｋ　をステージｐの加算器にコピーしなければならない。留意すべきは、Ｘｋ　は、問題となるマルチプレクサのそれぞれにおける入力の１つに接続されるということである。

【００４０】

可能性のあるオペランド境界を横切る入力接続には、図２に示すゲート４９のようなゲートが含まれていなければならない。このゲートは、境界が活性状態にある場合、境界を横切るデータの伝播をブロックする。すなわち、加算器ステージｐ　が入力ラインＸｑ　に接続されている場合、この接続には、Ｍｋがｐ～ｑの任意のｋについて真の場合、接続を遮断するゲートが含まれていなければならない。

【００４１】

Ｘが常に符号無し整数である場合、ゲート４９のようなブロック・ゲートが必要になるが、マルチプレクサ４３は、他のマルチプレクサ（例えば、４１、４２、４４、及び、４５）と同じマルチプレクサによって構成することが可能である。Ｘが符号付き整数の場合には、Ｍのμビットをマルチプレクサ４３に入力して、他のマルチプレクサと異なるようにしなければならない。この場合、演算モードによって不要になるので、ゲート４９のようなブロック・ゲートは省略することが可能である。

【００４２】

右シフト演算は、丸め誤差を生じる可能性がある。丸め誤差は、１がＸサブ・ワードからシフト・オフされる場合に生じる。上述の本発明の実施例は、切り捨てによって結果を丸める。切り捨ては、整数の除算結果の丸めに利用することが可能であるが、他の形態の丸めによって阻止される可能性のある、望ましくない問題を生じる可能性がある。例えば、ワードの集合に除算が施されており、集合の平均値が重要である場合には、切り捨てによって、エラーが生じることになる。切り捨てによる丸めは、全ての値が、次に小さい整数に丸められるので、平均値のシフトを生じることになる。

【００４３】

本発明の好適な実施例の場合、奇数丸め論理回路を利用して、このタイプのバイアスが阻止される。奇数丸めシステムでは、右シフトによって丸め誤差が生じる場合、結果は最寄りの奇数整数に丸められる。丸め以前の答が正しければ、変化は生じない。１が結果からシフト・オフされる毎に、丸め誤差は生じる。これは、シフト前のｍの最下位ビットに少なくとも１つの「１」が含まれている場合に生じる。ここで、ｍは、シフトされるビット数である。奇数丸めシステムのでは、「１」がワードからシフト・オフされる場合、結果の最下位ビットは、「１」にセットされる。シフト・アウトされる全てのビットが「０」の場合、結果は、正しかったことになり、結果の最下位ビットは変更されない。

【００４４】

ステージｐがサブ・ワードの内部ビットまたはサブ・ワードの最下位ビットに対して演算を行うことができる場合、ステージｐに接続されたマルチプレクサの代わりに、図３に２００で示されたマルチプレクサ構成を用いることによって、本発明に奇数丸めシステムを含めることが可能である。この構成には、丸め信号、シフト信号ｍ、及び、マスク・ビットＭによって制御される（２μ＋１）対１マルチプレクサ２０２が含まれている。マルチプレクサ２０２に対する入力は、２つのグループに分割することができる。グループ２２１は、ステージｐがサブ・ワードの最下位ビットに対する演算を行い、丸め信号Ｒが丸めを実施すべきであると指示している場合に、用いられる。この場合、ステージｐは、グループ２２１のｍ番目の入力を選択することによって、値（Ｘｐ　ＯＲ　Ｘｐ＋１　ＯＲ．．．Ｘｐ＋ｍ　）を備える信号に接続される。グループ２２１のｍ番目の入力は、Ｘｐ　～Ｘｐ＋ｍ　の入力を備えたＯＲ回路に接続される。典型的なＯＲ回路が、２０５～２０７で示されている。丸めを実施すべきでない場合、ステージｐは、グループ２２０のｍ番目の入力に接続される。この後者のモードの場合、マルチプレクサ構成は、サブ・ワードの内部ビットに対する演算だけしかできないステージに接続された他の（μ＋１）対１マルチプレクサと同様の働きをする。

【００４５】

シフト前に、Ｘの最下位（ｍ＋１）ビットが均等に分布している場合、この形式の丸めによって得られる平均誤差がゼロになるのは明らかである。留意すべきは、偶数丸めシステムも、丸めにおけるバイアスを阻止するということである。偶数丸めシステムでは、丸め誤差が生じ、丸め前の結果が奇数の場合には、その結果が、最寄りの偶数の整数に丸められる。しかし、偶数丸め案の実施に必要なハードウェアは、上述のものよりかなり複雑になり、従って、奇数丸めシステムが望ましい。

【００４６】

本発明の上述の実施例は、フル・ワード演算または２つのハーフ・ワードに対する並列演算を可能にするが、当該技術の熟練者には明らかなように、上記技法を利用して、サブ・オペランドの境界及び数の両方または一方が上述のものとは異なる実施例を実施することが可能である。サブ・オペランドの数に対する唯一の制限は、サブ・オペランドのビットの和が、全幅モードで演算を行うＡＬＵ　の幅を超えてはならないということである。同様に、サブ・オペランドの境界は、原理的に、対応するマスク・ビットをセットすることによって、ＡＬＵ　の任意の２つのビット間にセットすることが可能である。

【００４７】

本発明の上述の実施例は、Ｘワードにおける各サブ・ワードを2^m　で割った値とＹワードにおける対応するサブ・ワードの和の計算に関連して行ってきたが、当業者には明らかなように、本発明は、Ｘワードにおける各サブ・ワードを2^m　で割った値とＹワードにおける対応するサブ・ワードの差を求めることも可能である。上述のように、大部分のＡＬＵ　は、Ｙ入力をその２の補数に置き換えるための回路要素を提供する。従って、問題となる差は、Ｙオペランドに対してこの従来の回路要素を利用して計算することができる。

【００４８】

本発明は、区分化可能なＡＬＵ　に関連して説明してきたが、当業者には明らかなように、非区分化データ・ワードに対する演算を行う場合にも、利点がある。この場合、本発明によれば、単一命令に応答して、単一マシン・サイクルで除算及び加算を実施することが可能になる。

【００４９】

本発明の上記実施例は、本発明によって実施される各種演算のトリガ手段としての命令に関連して行ってきたが、当業者には明らかなように、本発明による演算は、記憶されたコンピュータ・プログラムの命令によって発生するのではない、電気信号によってトリガすることも可能である。従って、請求項において用いられる場合の「命令」という用語には、他の形態の信号によってトリガされる演算も含まれるものとする。さらに、当業者には明らかなように、本発明は、コンピュータの一部ではない回路要素にも利用することが可能である。

【００５０】

当業者には、以上の説明、及び、添付の図面から本発明に対する各種修正が明らかになるであろう。従って、本発明は、特許請求の範囲によってのみ制限されるものとする。

【００５１】

以上、本発明の各実施例ついて詳述したが、ここで各実施例の理解を容易にするために、各実施例を要約して、以下に列挙する。

【００５２】

１．　ｉ＝０～Ｎ－１であり、Ｚ０　が最下位ビットであり、ＺＮ－１　が最上位ビットであるとした場合、ビットＸｉを有するＸワード［１２］及びビットＹｉ　を有するＹワード［１４］の内容に演算を施して、ビットＺｉ　を有する結果ワード［１６］を発生するための装置［１０、３０］であって、前記Ｘ、Ｙ、及び、結果ワード［１２、１４、１６］を複数のサブ・ワード［１７、１９、２１、２２］に区分化し、前記Ｙ及び結果ワードのサブ・ワードの１つが前記Ｘワード［１２］の各サブ・ワードに対応するようにするための手段［３７］と、ｍを整数とした場合、第１の命令に応答して、Ｘワードの各Ｘサブ・ワードを2^m　で割った値と対応するＹワードのＹサブ・ワードの合計を発生し、そ

の結果によって、前記結果ワードの前記対応するサブ・ワードが決まるようにするための手段［３１～３５、４１～４５］から構成される論理演算装置である。

【００５３】

２．　さらに、第２の命令に応答して、2^m　で割った前記Ｘ　ワード［１２］の各サブ・ワードと前記Ｙワード［１４］の対応するサブ・ワードの差を発生し、その結果によって、前記結果ワード［１６］の前記対応するサブ・ワードが決まるようにするための手段［３１～３５、４１～４５］が設けられていることを特徴とする前記１に記載の論理演算装置「１０、３０」である。

【００５４】

３．　前記2^m　で割った結果に丸め誤差が生じた場合、前記Ｘワード［１２］の各前記サブ・ワードを割る結果が、次に高位の、奇数の整数に丸められることを特徴とする前記１に記載の論理演算装置「１０、３０」である。

【００５５】

４．　前記Ｘワード［１２］のビットを送るための多重化手段「４１～４５」と、前記Ｘ及びＹワード［１２、１４］から導き出されるビットを加算するための加算手段［３１～３５］が設けられていることと、前記加算手段が、それぞれ、前記Ｙワードから導き出されるビットを受信し、Ｙｐ　から導き出される前記ビットを受信するものがｐ番目のＹ入力手段と称される、複数のＹ入力手段と、それぞれ、前記Ｘワード［１２］から導き出されるビットを受信し、各前記Ｙ入力手段に対応するものが１つ含まれており、ｐ番目のＹ入力手段に対応するものは、ｐ番目のＸ入力手段と称される、複数のＸ入力手段と、それぞれ、前記Ｘ入力手段で受信した１つ以上のビットと前記Ｙ　入力手段で受信した対応するビットを加算するための手段、及び、直列に接続された１つの加算ステージから次の加算ステージに桁上げビットを伝播するための手段からから構成され、Ｙｐ　に演算を施すものが、ｐ番目の加算ステージと称される、直列に接続された複数の加算ステージとから構成されることと、前記区分化手段が、前記加算ステージで、前記Ｘ　ワード［１２］の異なるサブ・ワードのビットに対する演算が行われる場合、強制的に、前記桁上げビットを実施される演算によって求められる値にする手段［３７］から構成されることと、前記多重化手段が、複数の多重化回路［４１～４５］から構成され、各前記加算ステージに対して、前記多重化回路の１つが接続されており、Ｙｐ　及びＹｐ＋ｍ　が、前記Ｙワード［１４］の同じ前記サブ・ワードに含まれている場合、前記ｐ番目の加算ステージに接続された多重化回路が、前記加算ステージをＸｐ＋ｍ　に接続することを特徴とする前記１に記載の論理演算装置「１０、３０」てある。

【００５６】

５．　Ｙｐ　及びＹｐ＋ｍ　が、前記Ｙ　ワード［１４］の異なる前記サブワードに含まれている場合、前記ｐ番目の加算ステージに接続された前記多重化回路［４３］によって、前記Ｘサブ・ワードの１つの最上位ビットが前記ｐ番目の加算ステージに接続されることを特徴とする前記４に記載の論理演算装置［１０、３０］である。

【００５７】

６．　Ｙｐ　及びＹｐ＋ｍ　、前記Ｙワード［１４］の異なる前記サブワードに含まれている場合、前記ｐ番目の加算ステージに接続された前記多重化回路［４３］によって、値０を有する信号が前記ｐ番目の加算ステージに接続されることを特徴とする前記４に記載の論理演算装置［１０、３０］である。

【００５８】

７．　さらに、値（Ｘｐ　ＯＲ　Ｘｐ＋１　ＯＲ．．．Ｘｐ＋ｍ　）を有する信号を発生し、Ｙｐ　が前記サブ・ワードの１つの最下位ビットである場合、前記信号を前記ｐ　番目の加算ステージに接続するための手段が、設けられていることを特徴とする前記４に記載の論理演算装置［１０、３０］である。

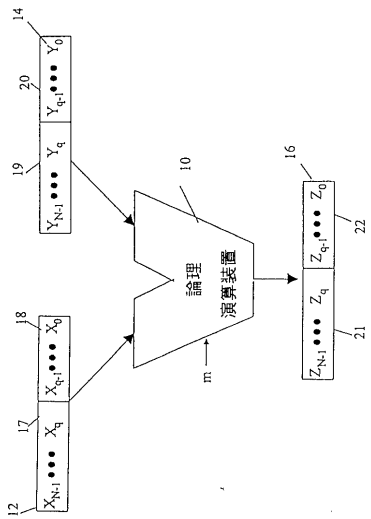
【００５９】

８．　前記区分け手段［３７］に、さらに、前記桁上げビットが、前記求められた値にされる前は、「１」であったことを示す信号を発生する手段が設けられていることを特徴と

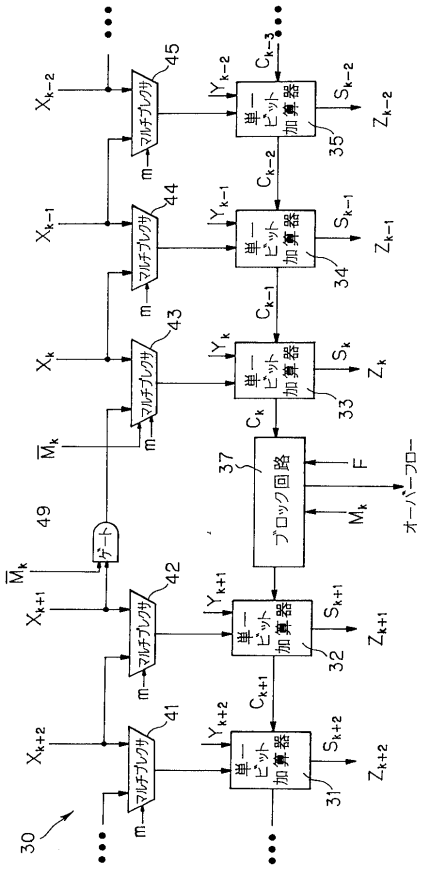
する前記４に記載の論理演算装置［１０、３０］である。
【００６０】
９．　さらに、前記発生信号の値を記憶するための記憶セルが設けられていることを特徴とする前記４に記載の論理演算装置［１０、３０］である。
【００６１】
１０．　ｉ＝０～Ｎ－１であり、Ｚ０　が最下位ビットであり、ＺＮ－１　が最上位ビットであるとした場合、ビットＸｉ　を有するＸワード［１２］及びビットＹｉ　を有するＹワード［１４］の内容に演算を施して、ビットＺｉ　を有する結果ワード［１６］を発生するための装置［１０、３０］であって、ｍを整数とした場合、第１の命令に応答して、前記Ｘワード［１２］を2^m　で割った値とＹワード［１４］の合計を発生するための手段［３１～３５、４１～４５］と、前記加算の結果から前記結果ワード［１６］を発生するための手段から構成される論理演算装置［１０、３０］である。
【００６２】
１１．　さらに、第２の命令に応答し、前記Ｘ　ワード［１２］を2^m　で割った値とＹワード［１４］の差を発生するための手段［３１～３５］が設けられていることと、その結果が、前記結果ワードの前記対応するサブ・ワードに納められることを特徴とする前記１０に記載の論理演算装置［１０、３０］である。
【００６３】
１２．　前記2^m　で割った結果に丸め誤差が生じた場合、前記Ｘワード［１２］を割る結果が、次に高位の、奇数の整数に丸められることを特徴とする前記１０に記載の論理演算装置「１０、３０」である。
【００６４】
１３．　前記Ｘワード［１２］のビットを送るための多重化手段「４１～４５」と、前記Ｘ及びＹワード［１２、１４］から導き出されるビットを加算するための加算手段が設けられていることと、前記加算手段が、それぞれ、前記Ｙワードから導き出されるビットを受信し、Ｙｐ　から導き出される前記ビットを受信するものがｐ番目のＹ入力手段と称される、複数のＹ入力手段と、それぞれ、前記Ｘワード［１２］から導き出されるビットを受信し、各前記Ｙ入力手段に対応するものが１つ含まれており、ｐ番目のＹ入力手段に対応するものは、ｐ番目のＸ入力手段と称される、複数のＸ入力手段と、それぞれ、前記Ｘ入力手段で受信した１つ以上のビットと前記Ｙ入力手段で受信した対応するビットを加算するための手段、及び、直列に接続された１つの加算ステージから次の加算ステージに桁上げビットを伝播するための手段からから構成され、Ｙｐ　に演算を施すものが、ｐ番目の加算ステージと称される、直列に接続された複数の加算ステージと「３１～３５」から構成されることと、前記多重化手段が、複数の多重化回路［４１～４５］から構成され、各前記加算ステージに対して、前記多重化回路の１つが接続されており、Ｙｐ＋ｍ　＜Ｎの場合、前記ｐ番目の加算ステージに接続された前記多重化回路が、前記加算ステージをＸｐ＋ｍ　に接続することを特徴とする前記１０に記載の論理演算装置「１０、３０」である。
【００６５】
１４．　ｐ＋ｍ＞（Ｎ－１）の場合、前記ｐ番目の加算ステージに接続された前記多重化回路［４３］が、前記加算ステージを論理値０の信号に接続することを特徴とする前記１３に記載の論理演算装置「１０、３０」である。
【００６６】
１５．　ｐ＋ｍ＞（Ｎ－１）の場合、前記ｐ番目の加算ステージに接続された前記多重化回路［４３］が、前記加算ステージをＸＮ－１　に接続することを特徴とする前記１３に記載の論理演算装置「１０、３０」である。
【００６７】
１６．　さらに、値（Ｘ０　ＯＲ　Ｘ１　ＯＲ．．．Ｘｍ　）を有する信号を発生し、前記信号を前記０番目加算ステージに接続するための手段［２００］が設けられていることを特徴とする前記１３に記載の論理演算装置［１０、３０］である。

【００６８】
【発明の効果】
以上のように、本発明によれば、ビットＸｉ（ｉ＝０～Ｎ－１）　を有するＸワードとビットＹｉ　を有するＹワードと結果ワードを複数のサブワードに区分化し、Ｙワードと結果ワードのサブワードを区分し、Ｙワードと結果ワードのサブワードの１つがＸワードの各サブワードに対応させ、第１の命令に応答してＸワードのＸサブワードを2^m　（ｍは整数）で割った値を対応するＹワードのＹサブワードの合計を発生し、その結果によって結果のワードの対応するサブワードを決めるようにしたので、論理演算装置の幅よりも狭いワードを伴う複数の演算処理を高効率で行うことができ、２進小数を含む乗算に好適となる。
【図面の簡単な説明】
【図１】２つのレジスタの内容に関連して、本発明によるＡＬＵ　が演算を行う方法を示すブロック図である。
【図２】２つのサブ・オペランドに分割可能であり、加算前に、Ｘオペランドにおいて位置１つ分だけ右にシフトさせることも可能な、本発明によるＡＬＵ　の一部に関するブロック図である。
【図３】結果の最下位ビットに対して奇数丸めを実施するマルチプレクサ構成のブロック図である。
【符号の説明】
１０，３０　ＡＬＵ
３１～３５　単一ビット加算器
３７　ブロック回路
４１～４５　マルチプレクサ
４９　ゲート
２００　マルチプレクサ構成
２０５～２０７　ＯＲ回路

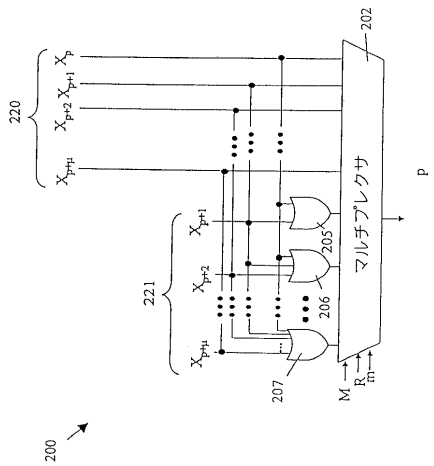
【図１】



【図２】



【図３】



フロントページの続き

審査官　田中　友章

(56)参考文献　特開昭５１－１３４５３９（ＪＰ，Ａ）
特開平０５－２０４６０５（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
G06F　7/50
G06F　7/52　310