

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3958381号
(P3958381)

(45) 発行日 平成19年8月15日(2007.8.15)

(24) 登録日 平成19年5月18日(2007.5.18)

(51) Int. Cl. F I
G06N 7/02 (2006.01) G O 6 N 7/02 5 5 4 D
 G O 6 N 7/02 5 5 4 B

請求項の数 3 (全 27 頁)

<p>(21) 出願番号 特願平7-349384 (22) 出願日 平成7年12月20日(1995.12.20) (65) 公開番号 特開平8-234986 (43) 公開日 平成8年9月13日(1996.9.13) 審査請求日 平成14年12月11日(2002.12.11) 審査番号 不服2005-20081(P2005-20081/J1) 審査請求日 平成17年10月17日(2005.10.17) (31) 優先権主張番号 08/363,196 (32) 優先日 平成6年12月22日(1994.12.22) (33) 優先権主張国 米国(US)</p>	<p>(73) 特許権者 504199127 フリースケール セミコンダクター イン コーポレイテッド アメリカ合衆国 78735 テキサス州 オースティン ウィリアム キャノン ドライブ ウェスト 6501 (74) 代理人 100116322 弁理士 桑垣 衛 (72) 発明者 ジェイ・グレッグ・ピオット アメリカ合衆国テキサス州78750、オ ースチン、レイクウッド・ドライブ 78 16</p>
--	---

最終頁に続く

(54) 【発明の名称】 効率的なファジー論理操作を行なうためのデータ処理システムおよびその方法

(57) 【特許請求の範囲】

【請求項1】

効率的なファジー論理操作を行なうためのデータ処理システムであって、

(a) メンバシップ関数における帰属度を決定するための命令を受け入れるための入力を有する命令デコーダであって、前記命令は複数の命令からなる命令セットから処理されるべきものであり、かつ前記命令デコーダは出力を有するもの、そして

(b) 前記命令デコーダの出力からの複数の入力信号にตอบสนองして複数の演算および論理機能を実行する実行ユニットであって、前記複数の入力信号はメンバシップ関数における帰属度を決定するための前記命令に応じての少なくとも1つの入力信号を含み、前記実行ユニットはまた前記メンバシップ関数の所定の特性を調べることにより前記メンバシップ関数が複数のメンバシップ関数の内の最後のメンバシップ関数であるか否かを判定するもの、

を具備し、

(c) 前記実行ユニットは、同実行ユニットにおける操作の結果を格納する条件符号レジスタを含み、かつ前記メンバシップ関数が前記複数のメンバシップ関数の内の最後のメンバシップ関数であることを判定したことに応じて、前記条件符号レジスタにおいてキャリービットを所定の論理状態に設定すること、

を特徴とする効率的なファジー論理操作を行なうためのデータ処理システム。

【請求項2】

効率的なファジー論理操作を行なうためのデータ処理システムであって、

10

20

命令を受けるための入力、および複数の制御信号を提供するための出力を有する命令デコーダ、そして

前記命令デコーダの出力に結合された制御入力、および情報バスに結合されたデータターミナルを有する実行ユニット、を具備し、

前記命令デコーダは前記実行ユニットを制御して、前記実行ユニットに前記データターミナルを介して入力値を受け入れさせ、かつファジーメンバシップ命令に応答してメンバシップ関数における前記入力値の帰属度を決定させ、

前記命令デコーダはさらに前記実行ユニットを制御して前記メンバシップ関数が複数のメンバシップ関数の内の最後のメンバシップ関数であるか否かを判定させ、

前記実行ユニットは、
キャリアビットを含む複数の条件符号ビットおよび終了信号を受けるための制御入力端子を有し、前記終了信号を取得して前記キャリアビットをセットする条件符号レジスタ、および

前記命令デコーダに結合された制御入力、および前記終了信号を提供するための出力端子を有するフラグ発生論理回路であって、前記命令デコーダが、前記メンバシップ関数が複数のメンバシップ関数の内の最後のメンバシップ関数であることを判定したことに応じて、前記終了信号を発する前期フラグ発生論理回路、

を具備することを特徴とする効率的なファジー論理操作を行なうためのデータ処理システム。

【請求項 3】

データ処理システムにおける、観察される可変入力ポイントの複数のメンバシップ関数の内の少なくとも 1 つの帰属度を決定する一方で、あるメンバシップ関数が前記複数のメンバシップ関数の内の最後のメンバシップ関数であるか否かを決定する方法であって、

(a) 前記観察される可変入力ポイントの帰属度を決定するために複数の命令からなる命令セットからある命令を受け入れる段階、

(b) 前記複数のメンバシップ関数の各々の境界を規定する第 1 および第 2 の境界値を検出する段階、

(c) 前記複数のメンバシップ関数の各々の前記第 1 および第 2 の境界値の各々を引き続き調べることにより、前記境界が前記複数のメンバシップ関数のエンドポイントを構成するか否かを判定する段階、

(d) 調べられているメンバシップ関数が前記複数のメンバシップ関数の内の最後のメンバシップ関数であることを、前記観察される可変入力ポイントに関する試験段階の終了を通知するための終了信号を提供することによって指示する段階、そして

(e) 条件符号レジスタの選択されたキャリアビットを所定の論理状態に設定する段階、を具備することを特徴とするデータ処理システムにおいて、あるメンバシップ関数が最後のメンバシップ関数であるか否かを決定する方法。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

この発明は、一般的にはデータ処理システムに関し、かつより特定的にはファジー論理操作を行なうためのデータ処理システムに関する。

【 0 0 0 2 】

【従来の技術】

データ処理システムはその入力および出力が 1 または 0 のいずれかとして解釈され、何らの他の可能性も存在できない 2 進機械として機能するよう開発されてきた。これは大部分の状況においてうまく動作するが、時には答えは単に「イエス」または「ノー」ではなく、それらの間の何かである。「ファジー論理」と称される概念が開発されて 2 進論理に基づくデータプロセッサが「イエス」および「ノー」の間の答えを与えることが可能になった。

【 0 0 0 3 】

ファジー論理はファジー境界を有するメンバシップ関数を有する論理システムである。メンバシップ関数は、「温度が暖かい」のような、主観的な表現を典型的なデータプロセッサが認識できる値に変換する。「暖かい」のようなラベルが使用されて、それらの境界が前記ラベルが一方の側で真でありかつ他方の側で偽であるポイントではないある範囲の入力値を識別するために使用される。むしろ、ファジー論理を実施するシステムにおいては、メンバシップ関数の境界は徐々に変化しかつ隣接するメンバシップ集合の境界とオーバーラップすることができる。従って、帰属度 (degree of membership) が一般にある入力値に割当てられる。例えば、ある範囲の温度にわたり2つのメンバシップ関数が与えられれば、ある入力温度は「涼しい」および「暖かい」と名付けられた関数の双方の領域にオーバーラップして存在することになる。メンバシップ関数の各々において帰属度 (すなわち、前記入力温度がメンバシップ集合、涼しいおよび暖かい、の各々に適合するか否か) を決定するためにはさらに処理が必要とされる。

10

【0004】

「ファジー化 (fuzzification)」と称されるステップが使用されて入力をファジー論理を実施するシステムにおけるメンバシップ関数と関係付ける。ファジー化プロセスは、「温度が暖かい」のような主観的な表現に対し具体的な数値を与える。これらの数値は一般に絶対的な真実の環境に制限されない人間の知覚の良好な近似を与えるよう試みるものである。ファジー化ステップの後に、ルール評価ステップ (rule evaluation step) が実行される。該ルール評価ステップの実行の間に、“min-max” 推論と称される技術が使用されてユーザによって規定される言語上のルールに対する数値的な結論が計算される。ルール評価ステップからの結論は「ファジー出力」と称されかつ種々の程度で真となり得る。従って、競合する結果が生じ得る。ファジー論理プロセスにおける最後の処理は「非ファジー化 (defuzzification)」と称される。名前が示すように、非ファジー化はすべてのファジー出力を組合わせてある複合的結果を得るプロセスであり、該複合的結果は標準のデータ処理システムに印加できる。ファジー論理に関するこれ以上の情報については、James M. Sibigtrothによる、“Implementing Fuzzy Expert Rules in Hardware” と題する論文を参照。この論文はAI EXPERTのページ25~31において、1992年4月に発行されている。また、1994年3月15日に発行され、「ファジー論理操作の間に集合におけるメンバシップを決定する回路および方法 (Circuit and Method for Determining Membership in a Set During a Fuzzy Logic Operation)」と題し、本発明と同じ譲渡人に譲渡された、米国特許第5,295,229号を参照されたい。

20

30

【0005】

【発明が解決しようとする課題】

典型的には、ファジー化ステップは周辺装置またはデータプロセッサにより実行されるソフトウェアプログラムによって実施されてきた。そのようなソフトウェアプログラムは多量の処理時間を必要とする数学的方程式を使用して帰属度を計算する。従って、ソフトウェアプログラムは正確な結果を提供するが、前記処理時間のため一般にそれが実施されるシステムの性能が制限される。他の一般的なソフトウェア解決方法は「テーブルルックアップ」ルーチンを使用し、そこではメモリに記憶されたデータのテーブルが特定の入力値に対応する値に対してアクセスされる。数学的な計算よりは早いのが、テーブルルックアップルーチンは実施するのが高価になる多量の専用メモリを必要とする。ファジー化ステップの実施に対するハードウェア的な解決方法はさらに迅速に結果を提供する。しかしながら、ハードウェアの解決方法は一般に多量の柔軟性のない専用回路を必要とする。ファジー化ステップに対するハードウェア的な解決方法は一般にはソフトウェア的な構成よりも高速であるが、ハードウェアによる解決方法は多量の専用回路領域を必要としかつしばしばデータ処理システムにおいて実施するにはあまりにも高価になる。

40

【0006】

50

従って、コストのかかるハードウェアを必要とすることなく、ファジー化ステップを迅速に行うための回路および方法の必要性が存在する。ハードウェアによる解決方法で一般に得られる速度が通常ソフトウェア解決方法に関連するメモリなしに得られることが必要である。

【0007】

この必要性は本発明によって満たされ、本発明の特徴および利点は添付の図面と共に以下の詳細な説明を参照することによりさらに明瞭に理解できるであろう。図面は本発明の唯一の形態を表わすことを意図するものでないことに注意を要する。

【0008】

【課題を解決するための手段】

本発明はデータ処理システムへ入力される1組のファジー操作の終りを迅速にかつ最小量の専用の回路によって指示するための回路および方法を提供する。データ処理システムのユーザが困難なかつ長いソフトウェアプログラムなしに迅速にファジー化ステップを行なうことができるようにする命令が与えられる。単一の入力ラベルのファジー化ステップが前記命令においてエンコードされデータ処理システムにおける過剰な量の付加回路なしに迅速かつ効率的に実行される。特定のシステム入力のためのもののような、1組のファジー化操作の終りがカウンタその他を必要とすることなく容易に検出される。

【0009】

本発明の構成を説明する場合に、用語「肯定する (a s s e r t) 」および「否定する (n e g a t e) 」、およびその種々の文法的な形態が使用されて「アクティブハイ」および「アクティブロー」論理信号の混合物を扱う上での混乱を避けている。「肯定する」は論理信号またはレジスタビットをそのアクティブな、または論理的に真な、状態にすることを言及するために使用されている。「否定する」は論理信号またはレジスタビットをそのインアクティブな、または論理的に偽の、状態にすることを言及するために使用されている。さらに、16進値はその値に先行する“\$”記号によって表示される。

【0010】

図1は、ファジー論理のいくつかの基本的な用語および概念を示すグラフを与える。図1に示されたファジー論理システムにおいては、「システム入力」はカ氏 (d e g r e e s F a h r e n h e i t) における温度である。5つの「メンバシップ関数」はラベルを温度範囲に関係付けるために与えられている。例えば、0度から30度では、温度は「寒い」と名付けられている。同様に、20度から50度では、温度は「涼しい」と名付けられている。メンバシップ集合の各々の境界は隣接集合の境界とオーバーラップしていることに注意を要する。従って、ある温度は1つより多くの集合に存在し得る。例えば、システム入力が68度の温度を示しているものと仮定する。図1を参照すると、68度は「暖かい」および「非常に暖かい」メンバシップ集合の双方の境界内にあり、各々のメンバシップ集合におけるゼロでない帰属度が得られることに注意を要する。前記システム入力は「暖かい」メンバシップ集合に対して\$33の帰属度を有し、かつ前記「非常に暖かい」メンバシップ集合に対しては\$CCの帰属度を有する。もしシステム入力が75度の温度を示していたとすれば、該温度は「非常に暖かい」メンバシップ集合における\$FFの帰属度を有する。同様に、75度の温度は「暑い」メンバシップ集合における\$00の帰属度を有することになる。図1のファジー論理構成においては、帰属度は\$00から\$FFの16進値の範囲におよび、これは0.00~0.996の範囲の小数 (f r a c t i o n) に対応させることができる。

【0011】

図2は、4つの値、すなわち、ポイント1 (P o i n t 1)、ポイント2 (P o i n t 2)、スロープ1 (S l o p e 1)、およびスロープ2 (S l o p e 2)、によって表すことができる単一のメンバシップ集合を示す。「デルタ1 (D e l t a 1) 」と称される第1の値はシステム入力 (ポイント “ X ”) およびポイント1の間の差として計算される。同様に、「デルタ2 (D e l t a 2) 」と称される第2の値はポイント2と前記システム入力との間の差として計算される。デルタ1およびデルタ2の値の計算から

10

20

30

40

50

、本発明の方法に従って帰属度（所属の程度：degree of membership）が与えられる。デルタ1またはデルタ2のいずれかの値がゼロより小さければ、対応するメンバシップ集合に対する帰属度は0.00である。これ以外では、帰属度はデルタ1とスロープ1の積、デルタ2とスロープ2の積、あるいは飽和値のいずれかの内の最小値に等しい。該最小値は試験されているメンバシップ集合における現在の入力値の帰属度となる。ここに述べた例においては、前記飽和値は16進値\$FFを有する。しかしながら、他の構成であってもファジー論理システムのユーザがある特定のシステムの仕様に従って飽和値を選択できるようにすることができる。

【0012】

ニーマニックラベル“MEM”で参照される、本発明の命令はデータ処理システムが複数のメンバシップ集合の内の1つにおけるシステム入力値の帰属度を決定するために一連のステップを行わせる。また、前記MEM命令は「終了(termination)」信号が肯定されかつ記録できるようにする。前記メンバシップ集合の各々はメモリから読出される複数の境界値（ポイント1、ポイント2、スロープ1、およびスロープ2）によって特徴付けられる。もし前記複数の境界値の各々が適切にメモリに格納されかつもし前記システム入力値がMEM命令の実行の前に適切にあるレジスタ位置に格納されていれば、前記複数のメンバシップ集合の内の1つにおけるシステム入力の帰属度が決定できる。続いて、前記複数のメンバシップ集合の各々における前記システム入力の帰属度がメモリ内の所定の位置に記憶される。MEM命令の実施によってデータ処理システムのユーザが伝統的なソフトウェア構成よりも迅速にかつ典型的なハードウェア構成によって必要とされるよりもより少ない専用回路領域によってメンバシップ計算機能を達成できるようになる。例えば、前記データ処理システムの通常動作に必要な演算論理ユニットを、スプリット動作モードで動作するために変更または修正できる。典型的には、演算論理ユニットは所定の時点において単一の16ビットの操作を行うことができる。しかしながら、ここに開示された演算論理ユニットは単一の16ビット操作あるいは2つの同時的な8ビット操作を行うために選択的にイネーブルできる。従って、あるメンバシップ集合における帰属度の計算に必要な演算操作は伝統的に必要とされるよりも短い時間で同時に計算される。非常に少しの付加回路が必要とされるのみである。演算論理ユニットをスプリットモードで動作させるために、単一のイネーブル信号が現存するバッファ回路に与えられる。さらに、MEM命令を実施するためには少しの量の付加回路を必要とするのみである。

【0013】

上に述べた命令および動作方法の1つの構成が図3に示されている。図3は、ファジー化命令が実施できるデータ処理システム10を示す。データ処理システム10は一般に中央処理ユニット(CPU)12、発振器24、電源回路26、タイミング回路28、外部バスインタフェース30、および内部メモリ32を含む。CPU12は一般に実行ユニット14、バス制御論理回路16、命令デコード論理回路18、制御ユニット20、およびシーケンサ22を有する。

【0014】

動作においては、“Osc 1”信号が、水晶のような、外部ソースを介して発振器24に与えられる。該水晶はOsc 1およびOsc 2信号の間に接続されて該水晶が発振できるようにする。Osc 1は「クロック」信号をデータ処理システム10の残りの部分に提供する。水晶発振器の動作はデータ処理技術においてよく知られておりかつ当業者には明らかであるべきである。

【0015】

同様に、電源回路26は外部電源から“Vdd”および“Vss”信号を受ける。Vdd信号は正の5ボルトを提供しかつVss信号は基準、またはグランド電圧を提供する。VddおよびVss信号はデータ処理システム10の残りの構成要素の各々に与えられる。これらの信号のルーティングはデータ処理技術においてよく知られておりかつ当業者には明らかであろう。

【0016】

10

20

30

40

50

タイミング回路 28 はクロック信号を受け取りかつ続いて適切なタイミング信号をタイミング制御バス 38 を介して CPU 12、外部バスインタフェース 30、および内部メモリ 32 の各々に提供する。

【0017】

複数のアドレス値が外部バスインタフェース 30 から外部アドレスバス 35 を介して与えられる。同様に、複数のデータ値が外部データバス 33 を介して外部バスインタフェース 30 に通信される。外部バスインタフェース 30 は外部ユーザとデータ処理システム 10 との間でアドレスおよびデータ値を通信するよう機能する。外部バスインタフェース 30 は複数のアドレスおよびデータ値を、それぞれ、内部アドレスバス 36 および内部データバス 34 を介してデータ処理システム 10 の残りの部分に提供する。内部メモリ 32 はデータ処理システム 10 の適切な動作に必要な情報値を格納するよう機能する。さらに、内部アドレスバス 36 および内部データバス 34 を介して与えられるユーザプログラムにおいて特定されれば、他のデータ値をそこに記憶することができる。

10

【0018】

CPU 12 はデータ処理システム 10 の動作の間に必要な各々の命令を実行する。内部アドレスバス 36 および内部データバス 34 は実行ユニット 14 とデータ処理システム 10 の残りの部分との間で情報を通信する。バス制御論理回路 16 は命令およびオペランドをフェッチする。該命令の各々は次に命令デコード論理回路 18 によってデコードされかつ制御ユニット 20 およびシーケンサ 22 に与えられる。制御ユニット 20 およびシーケンサ 22 はデータ処理システム 10 の計算機的能力を最も効率よく利用するために前記命令の各々の実行のシーケンスを維持する。さらに、制御ユニット 20 はマイクロ-ROM メモリ (図示せず) を含み、該マイクロ-ROM メモリは複数の制御情報を実行ユニット 14、バス制御論理 16、および命令デコード論理 18 の各々にマイクロ-ROM 制御バス 65 を介して提供する。

20

【0019】

実行ユニット 14 は図 4 に非常に詳細に示されている。実行ユニット 14 は概略的にデータバッファ 42、結果選択論理回路 46、フラグ発生論理回路 48、オペランド割当て論理回路 50、A - 入力制御論理回路 52、A - 入力マルチプレクサ 54、演算論理ユニット (ALU) 56、B - 入力マルチプレクサ 58、デルタレジスタ 60、システム入力レジスタ 64、および条件符号レジスタ 69 を含む。ここには示されていないが、当業者は実行ユニット 14 にはさらに他のレジスタも含めることが可能なことを理解すべきである。例えば、実行ユニット 14 はインデックスレジスタまたはアキュムレータを含むことができる。さらに、典型的なデータ処理システムにおいては、システム入力レジスタ 64 はアキュムレータとして実現できる。

30

【0020】

外部情報バス 40 はアドレスおよびデータ情報をデータバッファ 42 に与える。外部情報バス 40 は該アドレスおよびデータ情報を、それぞれ、内部アドレスバス 36 および内部データバス 34 から受け取る。データバッファ 42 は外部情報バス 40 を介して転送される値を情報バス A 66 および情報バス B 68 を介して実行ユニット 14 の残りの部分に提供する。ここには詳細に示されていないが、情報バス A 66 および情報バス B 68 は共に 16 ビットの幅でありかつハイ側 (ビット 15 ~ ビット 8) およびロー側 (ビット 7 ~ ビット 0) に分割される。

40

【0021】

情報バス A 66 のロー側はシステム入力レジスタ 64 の第 1 の入力に双方向的に結合されている。同様に、情報バス B 68 のハイ側はシステム入力レジスタ 64 の第 2 の入力に双方向的に結合されている。「デルタ 1」と称される第 1 のデルタ値が情報バス B 68 のハイ側を介してデルタレジスタの第 1 の入力に与えられる。第 2 のデルタ値、「デルタ 2」、が情報バス B 68 のロー側を介してデルタレジスタ 60 の第 2 の入力に与えられる。デルタレジスタ 60 はまた情報バス A 66 および情報バス B 68 の双方に接続されて実行ユニット 14 の他の構成要素に情報を提供する。アキュムレータおよびインデ

50

クスレジスタのような何らかの付加的なレジスタを同様に情報バス A 66 および情報バス B 68 に結合することかできる。

【0022】

情報バス A 66 は A - 入力マルチプレクサ 54 の第 1 の入力に接続されている。「制御」と名付けられた n ビット幅の信号が A - 入力マルチプレクサ 54 の第 2 の入力に与えられ、ここで n は整数である。A - 入力マルチプレクサ 54 は「A - 入力ロー側」と名付けられた第 1 の 10 ビットの出力を ALU 56 の第 1 の入力に提供しかつ「A - 入力ハイ側」と名付けられた第 2 の 10 ビットの出力を ALU 56 の第 2 の入力に提供する。

【0023】

情報バス B 68 は B - 入力マルチプレクサ 58 の入力に接続されている。B - 入力マルチプレクサ 58 をイネーブルして正しく機能させるのに必要な制御情報は、制御ユニット 20 においてマイクロ - ROM (図示せず) によって発生されかつマイクロ - ROM 制御バス 65 を介して転送される複数の信号を介して制御入力に与えられる。B - 入力マルチプレクサ 58 は「B - 入力ロー側」と名付けられた第 1 の 10 ビットの出力を ALU 56 の第 3 の入力に提供する。さらに、B - 入力マルチプレクサ 58 は「B - 入力ハイ側」と名付けられた第 2 の 10 ビットの出力を ALU 56 の第 4 の入力に提供する。

【0024】

「スプリットモード制御」と名付けられた信号が ALU 56 の第 5 の入力に与えられる。該スプリットモード制御信号は前記 MEM 命令の実行に応じて制御ユニット 20 のマイクロ - ROM メモリによって発生される。ALU 56 は前記入力の各々を処理して複数の結果を提供し、これらは結果バス 120 を介して転送される。結果バス 120 は複数の情報値を結果選択論理回路 46 およびフラグ発生論理回路 48 の各々に提供する。さらに、結果バス 120 は情報バス B 68 に結合されている。

【0025】

結果バス (Results bus) 120 に加えて、フラグ発生論理回路 48 がデルタレジスタ 60 およびデータバッファ 42 の双方に結合されている。フラグ発生論理 48 はデルタレジスタ 60 から「デルタ」信号を受けかつデータバッファ 42 から「バッファド・データ」信号を受け取る。フラグ発生論理回路 48 もまたオペランド割当て論理回路 50 および結果選択論理回路 46 の双方に結合され、フラグバス 70 を介して複数のフラグ値を提供する。さらに、フラグ発生論理回路 48 が ALU 56 に結合されて J_{high} および J_{low} 信号を、それぞれ、ALU 56 の第 6 および第 7 の入力に提供する。

【0026】

オペランド割当て論理回路 50 もまた情報バス A 66、情報バス B 68、およびフラグバス 70 に接続されている。オペランド割当て論理回路 50 は A - 入力制御論理回路 52 に接続されて「マルチプライヤ 1」と名付けられた第 1 の入力信号および「マルチプライヤ 2」と名付けられた第 2 の入力信号の双方を提供する。さらに、オペランド割当て論理回路 50 がデルタレジスタ 60 に結合されて前記デルタ信号を受信する。A - 入力制御論理 52 は A - 入力マルチプレクサ 54 に接続されて前記制御信号を提供する。

【0027】

結果選択論理 46 はフラグバス 70 および結果バス 120 の双方を介して情報を受け取る。該情報は処理されて「選択 (Select)」と名付けられた第 1 の出力および「フォース (Force) \$ FF」と名付けられた第 2 の出力を提供する。

【0028】

フラグ発生論理回路 48 は図 5 に非常に詳細に示されている。フラグ発生論理回路 48 は概略的に NOR ゲート 110、NOR ゲート 112、OR ゲート 114、NOR ゲート 116、NOR ゲート 118、AND ゲート 132、AND ゲート 134、ラッチ 136、AND ゲート 138、NOR ゲート 140、NOR ゲート 142、OR ゲート 144、NOR ゲート 146、NOR ゲート 148、そして OR ゲート 150 を含む。ラッチ 130 および 136 は各々 D フリップフロップとして実施され、該 D フリップフロップはデータ入力 (D)、クロック入力 (C)、リセット入力 (R)、そしてデータ出力 (Q) を有す

10

20

30

40

50

る。Dフリップフロップはデータ処理技術においてはよく知られており、従って、これ以上詳細に説明しない。

【0029】

データバッファ42(図4)は第1の情報値のビット15~8をNORゲート112に提供する。ここに述べた例では、前記第1の情報値は、以後バッファド・データ1信号と称される、第1のバッファリングされたデータ信号を介して与えられる。バッファド・データ1信号はポイント1の値またはスロープ1の値を提供する。NORゲート112の出力は“Z2”と名付けられかつラッチ136のデータ入力(D)およびANDゲート134の第1の入力に与えられる。前記タイミング制御信号はタイミング制御バス38によってラッチ136のクロック入力(C)に提供される。ラッチ136のデータ出力(Q)はANDゲート134の第2の入力に接続されて“ZP2”と名付けられた信号を提供する。ANDゲート134の出力はJ_{phigh}と名付けられかつORゲート144の第1の入力に提供される。

10

【0030】

さらに、バッファド・データ1信号のビット15~12(B15~B12)はNORゲート140に与えられる。NORゲート140の出力は“S2”と名付けられかつNORゲート142の第1の入力に提供される。

【0031】

データバッファ42はまた第2の情報値のビット7~0をNORゲート118およびANDゲート138の双方に提供する。前記第2の情報値は、バッファド・データ2信号と称される、第2のバッファリングされたデータ信号を介して提供される。該バッファド・データ2信号は前記ポイント2の値のビット7~0をANDゲート138に提供しかつスロープ2の値のビット7~0をNORゲート118に提供する。ANDゲート138は出力をラッチ130のデータ入力(D)に提供する。前記NORゲート118の出力が肯定されたとき、“0”のスロープが指示される。ANDゲート138の出力が肯定されたとき、その入力の全てが肯定され、ポイント2のそのB0~B7がすべて“1”であることを示す。そのB0~B7がすべて“1”であるとき、ポイント2は最大値に到達している。NORゲート118は“Z4”と名付けられた出力を提供する。このZ4信号はANDゲート132の第1の入力に転送される。前記タイミング制御信号はタイミング制御バス38によってラッチ130のクロック入力(C)に与えられる。ラッチ130のデータ出力(D)はANDゲート132の第2の入力に接続されて“ZP4”と名付けられた信号を提供する。

20

30

【0032】

ANDゲート132の出力は“J_{plow}”と名付けられかつORゲート150の第1の入力に供給されている。ANDゲート132の出力は、肯定されたとき、ポイントP2に対する最大値およびスロープ2に対する“0”の値を指示し、したがって所定の限界に到達したことを示しかつライン151上の信号J_{plow}が終了信号(termination signal)であることを示す。そのような終了信号が肯定されない場合、特定のシステム入力値のファジー化が継続する。信号J_{plow}は条件符号レジスタ69の入力に供給される。条件符号レジスタ69は実行ユニット14における操作の結果を示すためにいくつかの伝統的なビットを含む。これらのビットはストップディスエーブルビット(S)、X割込みマスクビット(X)、ハーフキャリービット(H)、I割込みマスクビット(I)、負ビット(N)、ゼロビット(Z)、オーバフロービット(V)、およびキャリー/ボロービット(C)を含む。これらのビットは一般に技術的によく知られた伝統的な方法でセットされかつクリアされる。

40

【0033】

しかしながら、フラグ発生論理48がメンバシップ関数が最後のセットであること、すなわち、右の境界が\$FFのエンドポイントおよび\$00のスロープを有することを検出するMEM命令の間に、信号ライン151上の信号J_{plow}はCビットがセットされるようにする。そうでない場合は、MEM命令はこのビットをクリアする。条件符号レジスタ

50

69のCビットをこのようにセットすることはファジー化操作のより効率的な符号化、およびファジー化操作によって必要とされるレジスタの数を低減できるようにする。この有利性につき以下にさらに明瞭に説明する。

【0034】

さらに、バッファド・データ2信号のビット7~4(B7~B4)はNORゲート146に与えられる。NORゲート146の出力は“S4”と名付けられかつNORゲート148の第1の入力に提供される。

【0035】

デルタレジスタ60は第3の情報値のビット15~12をNORゲート110に提供する。この構成においては、前記第3の情報値はデルタ1信号と称される第1のデルタ信号によって与えられる。NORゲート110の出力は“Z1”と名付けられかつまたフラグバス70を介して転送される。このZ1信号はまたNORゲート142の第2の入力に提供される。同様に、デルタレジスタ60は第4の情報値のビット7~4をNORゲート116に提供する。NORゲート116の出力は“Z3”と名付けられかつフラグバス70を介して転送される。前記第4の情報値はデルタ2信号と称される第2のデルタ信号を介して与えられる。前記Z3信号はまたNORゲート148の第2の入力に提供される。

10

【0036】

結果バス120はN_{low}およびN_{high}信号を、それぞれ、ORゲート114の第1および第2の入力に提供する。ORゲート114の出力はN_{mem}と名付けられている。N_{mem}信号はラッチ130およびラッチ136の双方のリセット入力(R)に与えられる。さらに、前記N_{mem}信号はNORゲート142およびNORゲート148の各々の第3の入力に与えられる。

20

【0037】

NORゲート142の出力は“J_{slow}”と名付けられかつORゲート144の第2の入力に与えられる。ORゲート144の出力は“J_{high}”と名付けられている。さらに、NORゲート148の出力は“J_{slow}”と名付けられておりかつORゲート150の第2の入力に与えられる。ORゲート150の出力は“J_{low}”と名付けられている。これらのJ_{low}およびJ_{high}信号はフラグバス70を介して通信される。

【0038】

オペランド割当て論理回路50が図6にさらに詳細に示されている。オペランド割当て論理回路50は概略的に第1のマルチプレクサ72、第2のマルチプレクサ74、第3のマルチプレクサ76、および第4のマルチプレクサ78を含む。

30

【0039】

デルタレジスタ60およびデータバッファ42はそれぞれデルタ1およびスロープ1の値の双方の下位4ビットをマルチプレクサ72の第1および第2の入力に提供する。さらに、デルタレジスタ60およびデータバッファ42はそれぞれデルタ1およびスロープ1の値の全8ビットをマルチプレクサ74の第1および第2の入力に提供する。同様に、デルタレジスタ60およびデータバッファ42はそれぞれ前記デルタ2およびスロープ2の値の下位4ビットをマルチプレクサ76の第1および第2の入力に与える。デルタレジスタ60およびデータバッファ42はそれぞれデルタ2およびスロープ2の値の全8ビットをマルチプレクサ78の第1および第2の入力に与える。

40

【0040】

前記“Z1”信号はフラグバス70を介してマルチプレクサ72の第3の入力に提供される。さらに、前記Z1信号はフラグバス70を介してマルチプレクサ74の第3の入力に提供される。同様に、“Z3”と名付けられた信号が、それぞれ、マルチプレクサ76およびマルチプレクサ78の第3の入力に与えられる。同様に、前記Z3信号はフラグバス70を介して転送される。

【0041】

マルチプレクサ72は4ビットの乗数1(multiplier 1)信号を出力し、かつマルチプレクサ76は4ビットの乗数2信号を提供する。マルチプレクサ74は「被乗

50

数 (Multipl icand) 1」と名付けられた 8 ビットの信号を出力し、この信号は情報バス A 66 のハイ側を介して転送される。マルチプレクサ 78 は「被乗数 2」と名付けられた 8 ビットの信号を出力し、これは情報バス A 66 のロー側を介して転送される。

【0042】

ALU56 が図 7 にさらに詳細に示されている。ALU56 は概略的に複数の加算器回路 (80, 82, 84, 86, 88, 92, 94, 96, 98 および 100)、バッファ 90、ハイキャリールックアヘッド論理回路 102、ローキャリールックアヘッド論理回路 104、ハイステータスフラグ回路 106、およびローステータスフラグ回路 108 を含む。

10

【0043】

前記 A - 入力ロー側信号はそれぞれ “A0”, “A1”, “A2”, “A8” および “A9” と名付けられた第 1 の複数のビットを提供する。ビット A0 は加算器 100 の第 1 の入力への入力である。同様に、ビット A1 ~ A9 はそれぞれ加算器 98、加算器 96、加算器 94 および加算器 92 の第 1 の入力にそれぞれ提供される。前記 B - 入力ロー側信号はそれぞれ “B0”, “B1”, “B2”, “B8” および “B9” と名付けられた第 2 の複数のビットを提供する。ビット B0 は加算器 100 の第 2 の入力に提供される。同様に、ビット B1 ~ B9 はそれぞれ加算器 98、加算器 96、加算器 94 および加算器 92 の第 2 の入力に各々与えられる。図 7 には詳細に示されていないが、ビット A3 ~ A7 および B3 ~ B7 の各々は第 1 の複数の加算器 (図示せず) のそれぞれの 1 つに提供される。前記第 1 の複数の加算器 (図示せず) の各々は加算器 92 ~ 100 と同様に機能しかつ、従って、同様に構成されている。

20

【0044】

“C0” と名付けられた信号がローキャリールックアヘッド論理 104 の第 1 の入力および加算器 100 の第 3 の入力の双方に与えられる。加算器 100 は “S0” と名付けられた和信号を出力する。さらに、加算器 100 は “P0” と名付けられた伝搬信号および “G0” と名付けられた発生信号の双方をローキャリールックアヘッド論理回路 104 に提供する。

【0045】

“C1” と名付けられた信号が加算器 98 の第 3 の入力に提供される。加算器 98 は “S1” と名付けられた和信号を提供する。加算器 98 もまた伝搬信号 “P1” および発生信号 “G1” を発生する。同様に、“C2” と名付けられた信号が加算器 96 の第 3 の入力に与えられる。加算器 96 は “S2” と名付けられた和出力、“P2” と名付けられた伝搬信号、および “G2” と名付けられた発生信号を提供する。同様に、“C8” と名付けられた信号は加算器 94 の第 3 の入力に与えられる。加算器 94 は “S8” と名付けられた和信号、“P8” と名付けられた伝搬信号、および “G8” と名付けられた発生信号を出力する。“C9” と名付けられたキャリー信号は加算器 92 に対し第 3 の入力を提供する。加算器 92 は “S9” と名付けられた和信号、“P9” と名付けられた伝搬信号および “G9” と名付けられた発生信号を出力する。加算器 92 ~ 100 によって出力される伝搬および発生信号の各々はローキャリールックアヘッド論理回路 104 に提供される。

30

40

【0046】

ローキャリールックアヘッド論理回路 104 は “C10” と名付けられたキャリー信号をバッファ 90 に提供する。スプリットモード制御信号もまたバッファ 90 に与えられる。バッファ 90 は “Cin” と名付けられた信号を提供する。

【0047】

前記 A - 入力ハイ側信号はそれぞれ “A10”, “A11”, “A12”, “A18” および “A19” と名付けられた第 1 の複数のビットを提供する。ビット A10 は加算器 88 の第 1 の入力に入力される。同様に、ビット A11 ~ A19 は、それぞれ、加算器 86、加算器 84、加算器 82 および加算器 80 の第 1 の入力に各々与えられる。前記 B - 入力ハイ側信号はそれぞれ “B10”, “B11”, “B12”, “B18” および “B19”

50

”と名付けられた第2の複数のビットを提供する。ビットB10は加算器88の第2の入力に提供される。同様に、ビットB11～B19は、それぞれ、加算器86、加算器84、加算器82および加算器80の第2の入力に各々提供される。図7には詳細に示されていないが、ビットA13～A17およびB13～B17の各々は第2の複数の加算器（図示せず）のそれぞれの1つに与えられる。前記複数の加算器（図示せず）の各々は加算器80～88と同様に機能しかつ、従って、同様に構成されている。

【0048】

前記Cin信号は加算器88の第3の入力に提供される。加算器88は“S10”と名付けられた和信号を出力する。さらに、加算器88は“P10”と名付けられた伝搬信号および“G10”と名付けられた発生信号の双方をハイキャリールックアヘッド論理回路102に提供する。

10

【0049】

“C11”と名付けられた信号は加算器86の第3の入力に与えられる。加算器86は“S11”と名付けられた和信号を提供する。加算器86はまた伝搬信号“P11”および発生信号“G11”を発生する。同様に、“C12”と名付けられた信号が加算器84の第3の入力に与えられる。加算器84は“S12”と名付けられた和出力、“P12”と名付けられた伝搬信号、および“G12”と名付けられた発生信号を提供する。同様に、“C18”と名付けられた信号が加算器82の第3の入力に与えられる。加算器82は“S18”と名付けられた和信号、“P18”と名付けられた伝搬信号、および“G18”と名付けられた発生信号を出力する。“C19”と名付けられたキャリー信号は加算器80への第3の入力を提供する。加算器80は“S19”と名付けられた和信号、“P19”と名付けられた伝搬信号、および“G19”と名付けられた発生信号を出力する。加算器80～88によって出力される伝搬および発生信号の各々はハイキャリールックアヘッド論理回路102に与えられる。ハイキャリールックアヘッド論理回路102は“C20”と名付けられたキャリー信号を出力する。

20

【0050】

図7には示されていないが、加算器92～100およびローキャリールックアヘッド論理回路104によって発生されるキャリー、和、伝搬、および発生信号の各々はローステータスフラグ回路108にも供給される。Jlow信号もまたローステータスフラグ回路108に供給される。同様に加算器80～88およびハイキャリールックアヘッド論理回路102によって発生されるキャリー、和、伝搬、および発生信号の各々もまたハイステータスフラグ回路106に供給される。Jphigh信号もまたハイステータスフラグ回路106に供給される。ローステータスフラグ回路108はそれぞれ“Slow”，“Vlow”，“Nlow”，“Hlow”および“Zlow”と名付けられた複数の信号を供給する。ハイステータスフラグ回路106はそれぞれ“Shigh”，“Vhigh”，“Nhigh”，“Hhigh”および“Zhigh”と名付けられた複数の信号を供給する。ローステータスフラグ回路108およびハイステータスフラグ回路106の双方からの各々の出力は結果バス120を介して転送される。

30

【0051】

ファジー論理操作の実行中に、データ処理システム10のユーザはシステム入力のファジー化を行うためにニーモニク形式MEM(Membership Evaluation)を有する命令を使用することができる。ここに説明する本発明の構成においては、ユーザはファジー化されるべき入力をMEM命令の実行に先立ちシステム入力レジスタ64に格納しなければならない。さらに、ユーザはまた実行ユニット14における第1のインデクスレジスタ（図示せず）にポインタを格納しなければならない。該ポインタはあるメンバシップ集合の台形を識別するのに必要な4つの点のスタートアドレス位置を指し示す。図2において前に述べたように、前記4つの点はそれぞれポイント1、ポイント2、スロープ1、およびスロープ2と名付けられている。他の形式では、ユーザはこれらの値の各々を前記命令のオペランドとして提供することができる。そのような技術はデータ処理技術においてよく知られておりかつ従って、その構成はここでは詳細に説明しない。ユー

40

50

ザはMEM命令の実行の後のファジー化されたデータのために記憶位置を特定しなければならない。1つの形式では、該ファジー化されたデータの記憶位置を示すために第2のインデクスレジスタを使用することができる。さらに、前記システム入力値はアキュムレータ64に格納されるべきである。

【0052】

本発明のこの構成においては、MEM命令の実行は図8に示されるフローチャートに従って行われる。該フローチャートはMEM命令の実行の間に行われる各々の機能の要約した概観(overview)を提供する。該機能の各々のより詳細な説明は後にMEM命令の実行の1つの例について行われる。

【0053】

図8に示されるように、MEM命令の実行の第1のステップは該MEM命令がデコードされた場合に行われる。前記インデクスレジスタに格納されたアドレスが読出されかつ内部メモリ32におけるアドレスXからポイント1およびポイント2の値を同時にアクセスするために使用される。該アドレスは次に増分されて次のアドレス、X+2、を指し示す。

【0054】

次にデルタ1およびデルタ2の値が同時に発生される。デルタ1の値はポイント1の値をシステム入力レジスタ64の内容から減算した結果として発生される。デルタ2の値はシステム入力レジスタ64の内容を前記ポイント2の値から減算した結果として発生される。デルタ1およびデルタ2の値はともにデルタレジスタ60に格納される。

【0055】

次に、もし前記ポイント1の値が\$00に等しければ、ZP2信号が肯定されかつ2進値1と共にラッチされる。同様に、もし前記ポイント2の値が\$FFに等しければ、ZP4信号が肯定されかつ2進値1と共にラッチされる。

【0056】

次に、スロープ1およびスロープ2の値が内部メモリ32のアドレスX+2から読出されかつ同時にデータバッファ42に格納される。該アドレスは次に増分されて次のアドレス、X+4、をさし示す。

【0057】

前記デルタ1およびデルタ2の値が調べられていずれかが負であるか否かが判定される。もしデルタ1またはデルタ2の値のいずれかが負であれば、システム入力値は現在調べられているメンバシップ集合の外側にある。従って、データバッファ42の内容がクリアされ\$00の帰属度にされる。さらに、それぞれラッチ136および130に格納された前記ZP2およびZP4の値がクリアされる。

【0058】

次に、デルタ1およびデルタ2の値の各々の上位4ビットが調べられてもしいずれかが\$0の16進値に等しいか否かが判定される。もしデルタ1の値の上位4ビットが\$0に等しければ、Z1フラグが肯定される。同様にデルタ2の値の上位4ビットが\$0に等しければ、Z3フラグが肯定される。

【0059】

もしZ1フラグが肯定されていれば、デルタ1の値は乗数(Multiplier)1の値でありかつスロープ1の値は被乗数(Multiplicand)1の値である。そうでなければ、スロープ1の値が乗数1の値でありかつデルタ1の値が被乗数1の値である。もしZ3フラグが肯定されていれば、デルタ2の値が乗数2の値でありかつスロープ2の値が被乗数2の値である。そうでなければ、スロープ2の値が乗数2の値でありかつデルタ2の値が被乗数1の値である。

【0060】

デルタ1の値とスロープ1の値の積が次に発生される。さらに、デルタ2の値とスロープ2の値の積が同時に2つの同時的乗算操作によって発生される。

【0061】

次にもしデルタ1の値とスロープ1の値の積が\$FFより大きければハイサチュレーション

10

20

30

40

50

ン・フラグ (S_{high}) が肯定される。さらに、もし J_{high} 信号が肯定されれば S_{high} 信号が肯定される。もしスロープ 1 またはデルタ 1 の値のいずれもその上位 4 ビットにおいて \$ 0 の 16 進値を持たなければ J_{high} が肯定される。 J_{high} はまたポイント 1 およびスロープ 1 の値が \$ 00 の 16 進値を持つ特別の場合に肯定される。もしポイント 1 の値が \$ 00 に等しければ、 $ZP2$ 信号が肯定され、かつもしスロープ 1 の値が \$ 00 に等しければ、 $Z2$ 信号が肯定される。

【 0062 】

同様に、もしデルタ 2 の値とスロープ 2 の値の積が \$ FF より大きければローサチュレーション・フラグ (S_{low}) が肯定される。さらに、もし J_{low} 信号が肯定されれば S_{low} 信号が肯定される。もしスロープ 2 またはデルタ 2 の値のいずれもその上位 4 ビットにおいて \$ 0 の 16 進値を持たなければ J_{low} が肯定される。さらに、 $Z1$ および $Z2$ 信号は否定されなければならない。 J_{low} はまたポイント 2 の値が \$ FF の 16 進値を持ちかつスロープ 2 の値が \$ 00 の 16 進値を持つ特別の場合に肯定される。もしポイント 2 の値が \$ FF に等しければ、 $ZP4$ 信号が肯定され、かつもしスロープ 2 の値が \$ 00 に等しければ、 $Z4$ 信号が肯定される。もし $Z4$ および $ZP4$ 信号の双方が肯定されれば、前記 J_{plow} 信号が肯定されて終了条件が検出されたことを示す。

10

【 0063 】

もし S_{high} および S_{low} 信号が共に肯定されれば、前記システム入力の帰属度は \$ FF である。もし S_{high} が否定されかつ S_{low} が肯定されれば、システム入力の帰属度はデルタ 1 の値とスロープ 1 の値の積に等しい。そうでなければ、帰属度はデルタ 2 およびスロープ 2 の値の積に等しい。

20

【 0064 】

次に前記帰属度はユーザによって特定された内部メモリ 32 のアドレスに提供されかつ MEM 命令の実行が終了する。

【 0065 】

図 8 に示されるフローチャートはデータ処理システム 10 における MEM 命令の実行の概観を与えている。より詳細な例を次に説明する。

【 0066 】

動作において、ユーザは外部に記憶されたあるいは内部メモリ 30 に記憶されたソフトウェアプログラムによって MEM 命令をデータ処理システム 10 に与えることができる。 MEM 命令がデータ処理システム 10 の外部のソースから提供されれば、該 MEM 命令は外部データバス 33 を介して外部バスインタフェース 30 に入力されることになる。外部バスインタフェース 30 は次に該 MEM 命令を内部データバス 34 を介して CPU 12 に与える。もし該 MEM 命令が内部メモリ 32 におけるソフトウェアプログラムによって与えられれば、該 MEM 命令は内部データバス 34 を介して CPU 12 に与えられることになる。 CPU 12 においては、実行ユニット 14 が該 MEM 命令をバス制御論理回路 16、制御ユニット 20、およびシーケンサ 22 の各々から受信した制御信号に応じて命令デコード論理回路 14 に提供する。命令デコード論理回路 18 は次に該 MEM 命令をデコードして該 MEM 命令の適切な実行に必要な複数の制御および情報信号を提供する。

30

【 0067 】

MEM 命令の受信およびデコードに応じて、実行ユニット 14 はシステム入力をファジー化する ($fuzzify$) ために必要なステップを開始する。前に述べたように、ファジー化されるべき値を示すシステム入力は MEM 命令の実行の前にデータ処理システム 10 のユーザによってシステム入力レジスタ 64 に格納されている。

40

【 0068 】

前記システム入力値に加えて、データ処理システム 10 のユーザはまたファジー論理操作の特定の構成における各々のメンバシップ集合を規定する 4 つの値を入力することを要求される。例えば、図 1 において、もし温度がファジー化されていれば、5 つのメンバシップ集合が規定される必要がある。従って、全システムにおける各々のメンバシップ集合を適切に区別するのに合計 20 の値が必要になる。図 2 において前に説明しかつ示したよう

50

に、1つのメンバシップ集合を識別するのに必要な値はポイント1、ポイント2、スロープ1、およびスロープ2である。これらの境界の各々は8ビットの2進値によって表される。

【0069】

ファジー化操作の間に、メンバシップ集合の各々がシステム入力値に関して計算されなければならない。図1に示されるファジー論理システムの場合は、システム入力の温度に対し5つのメンバシップ集合の計算がある。従って、各々のメンバシップ集合における帰属度を決定するのに必要な時間を最小化することが重要である。

【0070】

前記インデクスレジスタにおけるポインタは内部メモリ32におけるポイント1の値のアドレスを指し示す。内部メモリ32からアクセスされた時、前記ポイント1の値は外部情報バス40を介してデータバッファ42に与えられる。データバッファ42はそのポイント1の値を情報バスA 66を介してA-入力マルチプレクサ54に渡す。A-入力マルチプレクサ54は続いて該ポイント1の値をALU56に提供し、該ALU56においてその値はシステム入力値から減算されてデルタ1の値を提供する。デルタ1の値は情報バスB 68を介してデルタレジスタ60に与えられる。さらに、データバッファ42は前記ポイント1の値をバッファド・データ1信号を介してフラグ発生論理回路48に提供する。図5のフラグ発生論理回路48において、該ポイント1の値の各ビットがNORゲート112の入力に提供される。NOR操作の結果、すなわちZ2信号、は前記タイミング制御信号に従って決定される時点でラッチ136に格納される。

10

20

【0071】

本発明のこの構成においては、ポイント1およびポイント2の値は同時に内部メモリ32からアクセスされる。双方の値は外部情報バス40を介してデータバッファ42に転送される。データバッファ42はそのポイント2の値を情報バスB 68を介してB-入力マルチプレクサ58に受け渡す。B-入力マルチプレクサ58は次にそのポイント2の値をALU56に与え、該ALU56においてシステム入力値が減算されてデルタ2の値を提供する。情報バスB 68はデルタ2の値をデルタレジスタ60に提供する。さらに、データバッファ42は該ポイント2の値をバッファド・データ2信号を介してフラグ発生論理回路48に提供する。図5のフラグ発生論理回路48においては、該ポイント2の値の各ビットはANDゲート138の入力に与えられる。該AND操作の結果は前記タイミング制御信号に従って決定される時点でラッチ130に格納される。該AND操作はポイント2の値が\$FFに等しいか否かを検出する。

30

【0072】

ALU56は上に述べた各々の減算操作を実行し、それによってデルタ1およびデルタ2の値が同時に計算されるようにする。ALU56は通常の動作モードにある場合は16ビットの演算操作を行う。さらに、ALU56はスプリットモードの動作状態にある時2つの独立の8ビットの計算を同時に行うことができる。2つの8ビットの計算は同時に行われるから、16ビットの操作のみを行っていたALUの場合のように何らのサイクルも浪費されない。従って、ALU56は8ビットの演算を行う場合に実行ユニット18の速度および効率を大幅に増大する。ALU56の動作は後により詳細に説明する。

40

【0073】

インデクスレジスタのポインタは次に内部メモリ32のスロープ1の値のアドレスを指し示す。内部メモリ32は3ビットのスロープ1の値および8ビットのスロープ2の値を外部情報バス40を介してデータバッファ42に与える。データバッファ42は前記スロープの値の双方を16ビットのバッファド・データ信号を介してフラグ発生論理48に受け渡す。

【0074】

デルタ1およびデルタ2の値の計算に際しかつスロープ1およびスロープ2の値の受信に応じて、フラグ発生論理回路48はイネーブルされて現在試験されているメンバシップ集合における帰属度を決定するのに必要な複数のフラグを発生する。図5に示されるように

50

、デルタ1、デルタ2、スロープ1、およびスロープ2の値の上位4ビットはそれぞれNORゲート110、NORゲート116、NORゲート140、およびNORゲート146に提供される。

【0075】

NORゲート140はS2信号を発生しかつNORゲート146はS4信号を発生する。S2信号はスロープ1の値の上位4ビットが\$0の16進値を有する場合に肯定される。同様に、S4信号はスロープ2の値の上位4ビットが\$0の16進値を有する場合に肯定される。

【0076】

フラグ発生論理48においては、スロープ1およびスロープ2の値はそれぞれバッファド・データ信号によってNORゲート112およびNORゲート118に提供される。Z2フラグはスロープ1の値が16進\$00に等しい場合に肯定される。NORゲート112によって発生されるZ2フラグはラッチ136のデータ入力およびANDゲート134に与えられる。Z2フラグは前記タイミング制御信号が第1の所定の論理値を有する場合にラッチ136においてラッチされる。さらに、該タイミング制御信号が第2の所定の論理値を有する場合に、ラッチ136は前にラッチされたZP2信号をANDゲート134に提供する。ANDゲート134はJ_{p high}信号を発生するために使用される。

10

【0077】

Z4フラグはスロープ2の値が16進\$00に等しい場合に肯定される。NORゲート118によって発生されるZ4フラグはANDゲート132に提供される。ANDゲート138の出力はラッチ130のデータ入力に与えられる。ANDゲート138の出力は前記タイミング制御信号が第1の所定の論理値を有する場合にラッチ130においてラッチされる。さらに、前記タイミング制御信号が第2の所定の論理値を有する場合に、ラッチ130は前にラッチしたZP4信号をANDゲート132に提供する。ANDゲート132はJ_{p low}信号を発生するために使用される。前記J_{p low}信号の状態は図4の条件符号レジスタ69にラッチされ終了条件が検出されたことを記録する。

20

【0078】

Z1フラグ信号はデルタ1の値の上位4ビットの各々が否定された場合にのみ肯定される。同様に、Z3フラグ信号はデルタ2の値の上位4ビットが\$0の16進値を有する場合にのみ肯定される。各フラグ、Z1およびZ3、は次にオペランド割当てステップの間に使用され、該オペランド割当てステップは後にさらに詳細に説明する。

30

【0079】

デルタ1およびデルタ2の値に加えて、結果バス120もまたN_{high}およびN_{low}信号をORゲート114に転送してN_{mem}信号を発生する。N_{low}信号はALU56によって提供されるデルタ2の値がゼロより小さい場合に肯定される。同様に、N_{high}信号はもしALU56によって提供されるデルタ1の値がゼロより小さければ肯定される。前に述べたように、デルタ1またはデルタ2の値のいずれかがゼロより小さければシステム入力は試験されているメンバシップ集合のメンバではない。

【0080】

従って、N_{mem}信号が肯定された時、システム入力は現在試験されているメンバシップ集合においてゼロの帰属度を有することになる。続いて、N_{mem}信号がA-入力制御論理52に与えられる。肯定された時、該N_{mem}信号はA-入力制御論理52がA-制御信号を肯定できるようにし、それによってA-入力マルチプレクサ52が\$00の16進値をALU56に提供できるようにする。続いて、ALU56は\$00の値を情報バスB68を介してデータバッファ42に提供する。

40

【0081】

さらに、N_{mem}信号が肯定された時、ラッチ130はリセットされそれによってZP4信号が否定される。従って、J_{p low}信号は誤って肯定されることはない。同様に、ラッチ136がリセットされそれによってZP2信号が否定されるようにする。従って、J_{p high}信号は誤って肯定されることはない。

50

N_{mem} 信号、 S_2 信号、および Z_1 信号は各々 NOR ゲート 142 に提供されて J_{shigh} 信号を発生する。 J_{shigh} および J_{phigh} 信号は次に OR 演算されて J_{high} 信号を提供する。

【0082】

J_{phigh} 信号はポイント 1 の値およびスロープ 1 の値が共に \$00 の 16 進値を有する特別の場合を表示するために提供される。そのような場合は、メンバシップ集合が、帰属度がポイント 1 の境界値において \$FF の値を有するように配置された部分台形である場合に発生する。図 1 に示される「寒い」温度のメンバシップ関数はこの特別の場合の例である。 J_{high} 信号は ALU56 に提供されて飽和フラグ、 S_{high} 、が前記入力値が評価されているメンバシップ集合内にある場合に肯定される。 J_{high} 信号もまた Z_1 、 S_2 および N_{mem} 信号がすべて否定されれば肯定される。これはデルタ 1 の値もスロープ 1 の値も \$0F またはそれ以下ではなく、かつ現在の入力値が前記メンバシップ集合内にある場合に対応する。

10

【0083】

N_{mem} 信号、 S_4 信号、および Z_3 信号は各々 NOR ゲート 148 に与えられて J_{slow} 信号を発生する。 J_{slow} 信号および J_{plow} 信号は次に OR 演算されて J_{low} 信号を提供する。

【0084】

J_{low} 信号はポイント 2 の値が \$FF の 16 進値を有しかつスロープ 2 の値が \$00 の 16 進値を有する特別の場合を表示するために与えられる。そのようなケースは、メンバシップ集合が、帰属度がポイント 2 の境界値における \$FF で終了するように配置された部分台形である場合に発生する。もちろん、前記 J_{low} 信号はまたデルタ 2 およびスロープ 2 の値の積が \$FF より大きい場合に肯定される。しかしながら、信号 J_{plow} は、肯定されたとき、最後のファジー集合を示す、\$FF および \$00 のスロープ 2 で終了するファジー集合のみを示し、かつ、したがって、現在のシステム入力値に対するファジー化の終了を示す。図 1 に示された「暑い」温度のメンバシップ関数はこの特別の場合の例である。 J_{low} 信号は ALU56 に提供されて飽和フラグ、 S_{low} 、が前記入力値が現在評価されているメンバシップ集合内にある場合に肯定されるようにする。 J_{low} 信号もまた Z_3 、 S_4 および N_{mem} 信号がすべて否定されれば肯定される。これはデルタ 2 の値もスロープ 2 の値も \$0F またはそれ以下ではなく、かつ現在の入力値が前記メンバシップ集合内にある場合に対応する。

20

30

【0085】

しかしながら、もし前記システム入力値が現在試験されているメンバシップ集合内にある場合は、 N_{mem} 信号は否定されかつ Z_1 および Z_3 フラグ信号がオペランド割当て論理 50 および結果選択論理 46 の双方に提供されて前記メンバシップ集合内のデイグリーオブメンバシップを決定する。

【0086】

あるメンバシップ集合における帰属度は該メンバシップ集合を規定する台形の飽和点、デルタ 1 およびスロープ 1 の積、またはデルタ 2 およびスロープ 2 の積の内の最小値を得ることにより決定される。図 2 で示される例においては、前記飽和点は \$FF の 16 進値を有し、これは 0.996 の小数であると解釈される。

40

【0087】

デルタ 1 およびスロープ 1、そしてデルタ 2 およびスロープ 2 の双方の積を計算するために、高速乗算操作を保証するための方法が開発された。技術上一般に知られているように、乗算操作は乗数 (multiplier) が被乗数 (multiplicand) より小さい場合により迅速に完了する。もし、この例のように、前記メンバシップ集合の飽和値が \$FF であれば、数学的な解析によって積を形成するデルタまたはスロープ項の少なくともいずれか 1 つは \$0F より小さいかまた等しくなければならないことが分かる。しかしながら、この場合の 1 つの例外は現在のシステム入力値が前記メンバシップ集合の外側にあるかあるいは前記メンバシップ集合の飽和領域にある場合に発生する。

50

【 0 0 8 8 】

前記デルタおよびスロープ値の各々の上位4ビットを調べることにより、\$0Fまたはそれにより小さな16進値を有する第1の項は乗数として使用できる。したがって、乗算操作を完了させるのに必要な時間が短縮される。もしデルタ1およびスロープ1の値が双方とも\$0Fより大きな16進値を有しておれば、J_{h i g h}信号はS_{h i g h}飽和フラグが肯定されるようにする。同様に、もしデルタ2およびスロープ2の値の双方とも\$0Fより大きな16進値を有していれば、J_{l o w}信号はS_{l o w}飽和フラグを肯定されるようにする。

【 0 0 8 9 】

Z1フラグ信号はデルタ1またはスロープ1の値が前記4ビットの乗数1信号として提供されるべきか否かを示すために使用される。同様に、Z3フラグ信号はデルタ2またはスロープ2の値が4ビットの乗数2信号として提供されるべきか否かを示すために使用される。図6に示されるオペランド割当て論理50は行なわれるべき2つの乗算操作の各々に対する乗数の値を決定するための1つの構成を与える。

10

【 0 0 9 0 】

図6に示されるように、マルチプレクサ72はデルタ1またはスロープ1の値が前記4ビットの乗数1信号に割当てられるべきか否かを判定する。もしZ1フラグ信号が肯定されれば、デルタ1の値が前記4ビットの乗数1信号を介して転送される。もしZ1信号が肯定されていないならば、4ビットの乗数1信号にはスロープ1の値が割当てられることになる。さらに、もしZ1信号が肯定されていれば、マルチプレクサ74はスロープ1の値を8ビットの被乗数1信号としてA_{h i g h}に提供する。もしZ1信号が否定されていれば、デルタ1の値が8ビットの被乗数1信号に割当てられる。

20

【 0 0 9 1 】

同様に、マルチプレクサ76はデルタ2またはスロープ2の値がZ3フラグ信号の値に基づき乗数2信号に割当てられるべきであるか否かを判定する。もしZ3フラグ信号が肯定されれば、マルチプレクサ76はデルタ2信号を乗数2信号として提供する。もしZ3信号が肯定されていないならば、4ビットの乗数2信号はスロープ2の値を割当てられる。さらに、もしZ3フラグ信号が肯定されていれば、マルチプレクサ78はスロープ2信号を被乗数2信号として提供する。もしZ3信号が否定されていれば、デルタ2の値が8ビットの被乗数1信号に割当てられる。

30

【 0 0 9 2 】

乗数1および乗数2信号はオペランド割当て論理50からA-入力制御論理回路52に転送される。A-入力制御論理回路52は乗数1および乗数2信号の各々を処理してA-入力マルチプレクサ54の特定のシフト動作を決定するA-制御信号を提供する。

【 0 0 9 3 】

動作の間に、デルタ1またはスロープ1の値のいずれかである、乗数1信号はA-入力制御論理52に与えられて前記被乗数1信号が第1の乗算操作を行うために0, 1または2のいずれだけシフトされるべきかを決定する。さらに、A-入力制御論理52は被乗数1信号を否定するかあるいはゼロにする。A-制御信号はA-入力マルチプレクサ54に提供されて行われるべき操作の形式を表示する。被乗数1信号は情報バスA₆₆のハイ側を介してA-入力マルチプレクサ54に転送される。A-入力制御論理52は被乗数1信号をA-制御信号に従って修正しかつその修正された被乗数1信号をALU56に提供する。同様に、デルタ2またはスロープ2の値のいずれかである、乗数2信号もまたA-入力制御論理52に提供される。乗数2信号は第2の乗算操作を同時に行うために被乗数2信号を0, 1または2のいずれかだけシフトするために使用される。さらに、A-入力制御論理52は被乗数2信号を否定しあるいはゼロにすることができる。乗数2信号は情報バスA₆₆のロー側を介してA-入力マルチプレクサ54に転送される。A-入力制御論理52は被乗数2信号をA-制御信号に従って修正し、かつ次にその修正された被乗数2信号をALU56に提供する。

40

【 0 0 9 4 】

50

B - 入力マルチプレクサ 5 8 の制御は M E M 命令のデコード中に発生されかつマイクロ - R O M 制御バス 6 5 を介して転送される外部のマイクロ - R O M 制御信号によって与えられる。そのようなマイクロ - R O M 制御信号の発生、ルーティング、および使用はデータ処理技術においてよく知られており、従って、ここで詳細には説明しない。ここに述べた例においては、\$ 0 0 の初期化された部分積が B - 入力マルチプレクサ 5 8 を介して A L U 5 6 に与えられる。

【 0 0 9 5 】

A - 入力制御論理回路 5 2 から適切な A - 制御信号を受信すると、A - 入力マルチプレクサ 5 4 は修正された被乗数 1 および修正された被乗数 2 信号を、それぞれ、A L U 5 6 のハイ側およびロー側に提供する。B - 入力マルチプレクサ 5 8 は同時に初期化された部分積を、それぞれ、A L U 5 6 のハイ側およびロー側に提供する。

10

【 0 0 9 6 】

A L U 5 6 は図 7 にさらに詳細に示されている。動作においては、前記修正された被乗数 2 信号が加算器 9 2 ~ 1 0 0 のそれぞれの 1 つに提供される。“ a 9 ” と名付けられた、前記修正された被乗数 2 信号のビット 9 は加算器 9 2 の第 1 の入力に接続されている。同様に、修正された被乗数 2 信号のビット 8 ~ 0 は、それぞれ、加算器 9 4 , 9 6 , 9 8 および 1 0 0 の第 1 の入力に接続されている。“ b 9 ” と名付けられた、前記初期化された部分積のビット 9 は加算器 9 2 の第 2 の入力に接続されている。前記初期化された部分積のビット 8 ~ 0 はまた、それぞれ、加算器 9 4 , 9 6 , 9 8 および 1 0 0 の第 2 の入力に接続されている。

20

【 0 0 9 7 】

“ C 0 ” と名付けられたキャリー信号が加算器 1 0 0 の第 3 の入力に与えられる。該 C 0 信号は一般に実行されている操作および乗数 2 信号の値に応じて A L U 5 6 の外部のソースによって与えられる。これらの入力信号の各々に基づき、加算器 1 0 0 は 3 つの出力信号を提供する。第 1 の出力信号、すなわち S 0 信号、は次式で表わされるものに等しい。

$$S 0 = a 0 = : b 0 = : c 0 \quad (1)$$

なお、ここで = : なる記号はほぼ等しいことを表わすものとする。

【 0 0 9 8 】

第 2 の出力信号は前記発生、または G 0 信号であり、これは次式に等しい。

$$G 0 = a 0 \cdot b 0 \quad (2)$$

30

【 0 0 9 9 】

第 3 の出力信号は伝搬、または P 0 信号であり、これは次式に等しい。

$$P 0 = a 0 + b 0 \quad (3)$$

【 0 1 0 0 】

前記 G 0 および P 0 信号の双方に応じて、ローキャリールックアヘッド論理回路 1 0 4 は C 1 信号を発生し、これは加算器 9 8 の第 3 の入力に与えられる。該 C 1 信号は次のように表わされる。

$$C 1 = G 0 + P 0 \cdot C 0 \quad (4)$$

【 0 1 0 1 】

a 1 , b 1 および C 1 信号の各々に応じて、加算器 9 8 は “ G 1 ” と名付けられた信号を発生しかつ “ P 1 ” と名付けられた信号をローキャリールックアヘッド論理回路 1 0 4 に伝搬する。

40

【 0 1 0 2 】

G 1 , P 1 および C 1 信号に応じて、ローキャリールックアヘッド論理回路 1 0 4 は C 2 信号を発生し、これは加算器 9 6 の第 3 の入力に与えられる。該 C 2 信号は次の式で計算される。

$$C 2 = G 1 + P 1 (G 0 + P 0) C 0 \quad (5)$$

【 0 1 0 3 】

加算器 9 6 は次に “ G 2 ” と名付けられた信号を発生しかつ “ P 2 ” と名付けられた信号をローキャリールックアヘッド論理回路 1 0 4 に伝搬する。

50

【 0 1 0 4 】

続いて、ローキャリールックアヘッド論理回路 1 0 4 はキャリー信号を発生し、該キャリー信号は A L U 5 6 のロー側の残りの加算器 (9 4 , 9 2) の各々の第 3 の入力に与えられる。各々の残りのキャリー信号は前の発生項および前の伝搬およびキャリー項の積の和として計算される。ここには示されていないが、A - 入力ロー側および B - 入力ロー側によって与えられる値のビット 3 ~ 7 の和を計算するために付加的な加算器が使用される。

【 0 1 0 5 】

最後の加算器、すなわち加算器 9 2、から伝搬 (p r o p a g a t e) および発生 (g e n e r a t e) 項を受信すると、ローキャリールックアヘッド論理回路 1 0 4 は “ C 1 0 ” と名付けられたキャリー信号を発生する。該 C 1 0 信号はバッファ 9 0 に与えられる。スプリットモード制御信号に応じて、バッファ 9 0 は前記 C 1 0 キャリー信号を A L U 5 6 のハイ側の次の加算器に送るか、あるいはそこで発生した他のキャリー値を提供する。スプリットモード制御信号は実行ユニット 1 4 の外部のソースによって与えられる。例えば、M E M 命令の実行の間に、スプリットモード制御信号が命令のデコード中に発生される。さらに、該スプリットモード制御信号はデータ処理システムのユーザによって特定される命令以外のソースによって与えられてもよい。

【 0 1 0 6 】

もし前記 C 1 0 キャリー信号が加算器 8 8 に C i n キャリー信号として送られれば、A L U 5 6 は技術的によく知られた典型的な 1 6 ビットの演算論理ユニットとして動作する。しかしながら、もし該 C i n キャリー信号がローキャリールックアヘッド論理回路 1 0 4 以外のソースによって与えられれば、A L U 5 6 はスプリット動作モードで動作する。スプリット動作モードにおいては、A L U 5 6 は関係のない 2 つの 8 ビットの結果を同時に提供することができる。M E M 命令の実行の間に、A L U 5 6 はスプリット動作モードで動作して同時にデルタ 1 信号およびスロープ 1 信号の積ならびにデルタ 2 信号およびスロープ 2 信号の積の双方を提供する。さらに、A L U 5 6 はまたスプリットモードで動作している間に同時にデルタ 1 およびデルタ 2 の値を計算している。従って、スプリット動作モードで動作することにより、A L U 5 6 は 8 ビットの操作の結果をより迅速かつ効率的に提供することができる。

【 0 1 0 7 】

さらに、A L U 5 6 においては、ローステータスフラグ回路 1 0 8 は A L U 5 6 のロー側の動作に関するステータス情報を提供する。ローステータスフラグ回路 1 0 8 は複数のステータス信号、S _{1 0 w} , V _{1 0 w} , N _{1 0 w} , H _{1 0 w} および Z _{1 0 w} を提供し、これらの各々は結果バス 1 2 0 を介して転送される。ここには示されていないが、加算器 9 2 ~ 1 0 0 の出力の各々はローステータスフラグ回路 1 0 8 に提供される。そのようなルーティングは一般にデータ処理技術において知られておりかつ当業者に容易に実施されるべきものである。ステータス信号の各々は A L U 5 6 のロー側によって発生される複数の和およびキャリー信号を組み合わせることによって決定される。

【 0 1 0 8 】

ここに述べた本発明の構成においては、S _{1 0 w} 信号は次の式で表わされる。

$$S_{10w} = S_9 + S_8 + J_{10w} \quad (6)$$

【 0 1 0 9 】

V _{1 0 w} , N _{1 0 w} , H _{1 0 w} および Z _{1 0 w} 信号はそれぞれ次のように表わされる。

$$V_{10w} = C_6 = : C_7 \quad (7)$$

$$N_{10w} = S_8 \quad (8)$$

$$H_{10w} = C_3、および \quad (9)$$

$$Z_{10w} = \text{反転} (S_7 + S_6 + S_5 \dots + S_0) \quad (10)$$

【 0 1 1 0 】

さらに、加算器 9 2 ~ 1 0 0 の各々によって出力される和信号は A L U 5 6 から結果バス 1 2 0 を介して転送される。加算器 9 2 ~ 1 0 0 によって出力される和信号の各々は乗数 2 および被乗数 2 信号の積のそれぞれのビットを表わす。次に、乗数 2 および被乗数 2 信

10

20

30

40

50

号の積はデルタ2信号およびスロープ2信号の積に等しい。

【0111】

動作の間に、ALU106の上位側は下位側と同様に機能する。“Cin”と名付けられたキャリー信号は加算器88の第3の入力に与えられる。前に述べたように、Cin信号はALU56の外部のソースによって与えられてもよく、あるいはローキャリールックアヘッド論理回路104からの前記C10キャリー信号によって与えられてもよい。前記3つの入力信号の各々に基づき、加算器88は3つの出力信号を提供する。第1の出力信号、すなわちS10信号は次式で表される。

$$S10 = a10 \oplus b10 \oplus Cin \quad (11)$$

【0112】

第2の出力信号は発生、またはG10信号であり、これは次式に等しい。

$$G10 = a10 \cdot b10 \quad (12)$$

【0113】

第3の出力信号は伝搬、またはP10信号であり、これは次式に等しい。

$$P10 = a10 + b10 \quad (13)$$

【0114】

前記G10およびP10信号の双方にตอบสนองして、ハイキャリールックアヘッド論理回路102はC11信号を発生し、これは加算器86の第3の入力に与えられる。該C11信号は次式で表わされる。

$$C11 = G10 + P10 \cdot C10 \quad (14)$$

【0115】

a11, b11およびC11信号の各々に応じて、加算器86は“G11”と名付けられた発生信号および“P11”と名付けられた伝搬信号をハイキャリールックアヘッド論理回路102に提供する。

【0116】

G11, P11およびC11信号に応じて、ハイキャリールックアヘッド論理回路102はC12信号を発生し、これは加算器84の第3の入力に与えられる。該C12信号は次の式によって計算される。

$$C12 = G11 + P11(G10 + P10)C10 \quad (15)$$

【0117】

加算器84は次に“G12”と名付けられた発生信号および“P12”と名付けられた伝搬信号をハイキャリールックアヘッド論理回路102に提供する。

【0118】

次に、ハイキャリールックアヘッド論理回路102はキャリー信号を発生し、該キャリー信号はALU56のハイ側の残りの加算器(80, 82)の各々の第3の入力に提供される。残りのキャリー信号の各々は前の発生項および前の伝搬項とキャリー項との積の和として計算される。ここでは図示されていないが、付加的な加算器が使用されてA-入力ハイ側およびB-入力ハイ側の双方によって与えられる値のビット13~17の和を計算する。

【0119】

最後の加算器、すなわち加算器80、から伝搬および発生項を受信すると、ハイキャリールックアヘッド論理回路102は“C20”と名付けられたキャリー信号を提供する。該C20信号は後の操作に使用するために結果バス120を介してALU56から出力される。

【0120】

さらに、ALU56において、ハイステータスフラグ回路106はALU56のハイ側の動作に関するステータス情報を提供する。ハイステータスフラグ回路106は複数のステータス信号、S_{high}, V_{high}, N_{high}, H_{high}およびZ_{high}を提供し、これらの各々は結果バス120を介して転送される。加算器80~88の出力の各々はハイステータスフラグ回路106に提供される。前と同様に、そのようなルーティング

10

20

30

40

50

はデータ処理技術において一般に知られておりかつ当業者に容易に実施されるべきものである。前記ステータス信号の各々はALU56のハイ側によって発生される複数の和およびキャリー信号を組み合わせることによって決定される。

【0121】

ここに述べた本発明の構成において、前記 S_{high} 信号は次の式で表わされる。

$$S_{high} = S_{19} + S_{18} + J_{high} \quad (16)$$

【0122】

前記 V_{high} 、 N_{high} および H_{high} 信号はそれぞれ次のように表わされる。

$$V_{high} = C_{16} = : C_{17} \quad (17)$$

$$N_{high} = S_{18} \quad (18)$$

$$H_{high} = C_{13}、および \quad (19)$$

$$Z_{high} = \text{反転}(S_{19} + S_{18} + S_{17} + \dots + S_{10}) \quad (20)$$

【0123】

さらに、加算器80～88の各々によって出力される和信号は結果バス120を介してALU56から転送される。加算器80～88によって出力される前記和信号の各々は乗数1および被乗数1信号の積のそれぞれのビットを表わす。従って、乗数1信号および被乗数1信号の積はデルタ1信号およびスロープ1信号の積に等しい。

【0124】

ALU56のハイおよびロー側からのステータスビットの各々は次に結果選択論理46に提供される。結果選択論理46は続いて前記 S_{low} および S_{high} 信号の各々を使用してデルタ1およびスロープ1信号の積、デルタ2およびスロープ2信号の積、または\$FFの飽和値のいずれが調べられているメンバシップ集合における帰属度として供給されるべきかを決定する。

【0125】

システム入力値がメンバシップ集合の台形内にある全ての場合に、 S_{high} が肯定されるか、 S_{low} が肯定されるか、或いは S_{high} および S_{low} の双方が肯定される。もし前記 S_{low} および S_{high} 信号の双方が肯定されれば、システム入力は前記飽和点(この例では、\$FF)の帰属度を有する調べられているメンバシップ集合のメンバである。従って、 S_{low} および S_{high} の双方が肯定される場合は、結果選択論理回路46はフォース(Force)\$FF信号を肯定してデータバッファ42の内容を\$FFになるようにシステム入力が前記メンバシップ集合のメンバでありかつ\$FFの帰属度を有することを示す。

【0126】

もし S_{low} 信号が肯定されているが、 S_{high} 信号は否定されている場合は、結果選択論理回路46はデルタ1およびスロープ1信号の積をデータバッファ42に格納するために選択(Select)信号を肯定する。同様に、もし S_{low} 信号が否定されているが、 S_{high} 信号が肯定されている場合は、結果選択論理回路は前記選択信号を否定してデルタ2およびスロープ2信号の積をデータバッファ42に格納させる。

【0127】

もし S_{high} および S_{low} の双方が否定されれば、システム入力値は前記境界値によって規定されるメンバシップ集合の台形内に無いことになる。システム入力値がメンバシップ集合の台形の外側にある全ての場合に、 N_{mem} 信号が肯定される。該 N_{mem} 信号は、次に、A-入力制御論理52に乗算操作の間にALU56にゼロが入力されることを選択させる。結果選択論理回路46は前記フォース\$FF信号および前記選択信号の双方を否定しかつ\$00の16進値がデータバッファ42に格納される。結果選択論理回路46は、従って、MEMの帰属度の計算の結果の値を決定しかつその結果をデータバッファ42に提供する。実行ユニット14の外部のソースから要求された場合、データバッファ42はその結果を外部情報バス40を介して提供する。

【0128】

【発明の効果】

10

20

30

40

50

単一のソフトウェア命令を使用してメンバシップ集合の帰属度を決定する回路および方法が提供された。ここに説明した例においては、該ソフトウェア命令はMEM命令である。該MEM命令によってプログラムされた時、データ処理システム10は非常に高速で実行される単一のソフトウェア命令によってメンバシップ集合の帰属度を決定することができる。従来の構成は多量のソフトウェアコードを必要とし、これは一般に実行するのにかなり多くの時間を必要とする。帰属度の計算を行う典型的なソフトウェアプログラムにおいて、MEM命令はかなりの数のソフトウェア命令を軽減する。例えば、MC68HC11の命令セットを使用して実施されるソフトウェアプログラムは典型的には帰属度の計算を行うのに31個の命令を必要とする。MEM命令はMC68HC11のプログラムの31個の命令全てに置換えられる。更に、MEM命令は結果を帰属度の計算の従来のソフトウェアによる実施の場合よりも71サイクルまで早く提供する。

10

【0129】

MEM命令は帰属度の計算をより早くかつ効率的に行うことができるが、その理由はそれがデータ処理システム10の命令セットのメンバであるからである。命令セットのメンバである命令はその命令をエミュレートするために該データ処理システム10をプログラムする外部ルーチンよりも迅速に実行されることはよく知られている。更に、オペランド割り当て論理回路50によって行われる機能はまたMEM命令を実現するデータ処理システム10の性能を増強する働きを成す。オペランド割り当て論理回路50は帰属度の計算の間に使用される乗数または被乗数の内の少なくとも1つは上位4ビットにおいて\$0の16進値を持たなければならないことを認識する。従って、上位4ビットに\$0を有する値

20

【0130】

更に、2つの8ビットの結果が同時に生成されるスプリットモードでALU56を動作させることにより、更に時間が節約される。ALU56はスプリット動作モードにある時2つの8ビットの結果を生成し、スプリット動作モードにない場合に16ビットの結果を生成するために使用できる独特の回路を提供する。ここに説明した実施例の性格によって、8ビットの結果のみが前記デルタ値およびスロープ値の減算および乗算の間に生成される。従って、ALU56はMEM命令の実行中にスプリットモードで動作しかつ2つの減算または乗算操作が同時に行われる。従って、これらの操作を行うのに通常必要な時間はスプリットモードの動作の間は実効的に半分になる。

30

【0131】

通常、カウンタが使用されてユーザが評価されたファジー集合の数をカウントできるようにする。本発明は\$FFの最大ポイント値および\$00のスロープによってトリガされる終了信号を使用することにより最後の集合を識別してカウンタの必要性を除去する。この利点は一連のメンバシップ関数における入力値の帰属度を決定するために米国特許第5,295,229号の教示にしたがって書くことができるアセンブリ言語プログラムを比較することにより説明できる。

【0132】

```
LDAA  SYSTEM__INPUT
LDAB  #5    ;メンバシップ関数のためのループカウンタ
L1    MEM   ;単一のメンバシップ計算
DBNZ  B, L1 ;ゼロでなければカウンタを減分して分岐
```

40

【0133】

本発明に係わるデータ処理システムにおいては、このアセンブリ言語プログラムは次のように書くことができる。

【0134】

```
LDAA  SYSTEM__INPUT
L1    MEM   ;単一のメンバシップ計算
BCC   L1   ;キャリークリアであれば分岐
```

50

【 0 1 3 5 】

データ処理システム 10 においては、B C C 命令は D B N Z 命令よりも高速度で実行するが、それはループカウンタをたえず減分する必要がないからである。さらに、このシーケンスは使用するレジスタが 1 つ少ない。

【 0 1 3 6 】

この技術は最後のメンバシップ関数が \$ 0 0 の右スロープおよび \$ F F の右エンドポイントを有することを保証することができるファジーシステムにとって有用である。この条件は一般的なものである。しかしながら、最後のメンバシップ関数に対してこの制約を持たないシステムにおいては、前記最初の 4 命令シーケンスを依然として使用することができる。

10

【 0 1 3 7 】

ここに説明した本発明の構成は実例によってのみ与えられている。しかしながら、ここに述べた機能を実行するために数多くの他の構成も存在し得る。例えば、メンバシップ集合を定義するのに必要なポイントは、メモリにあらかじめ格納されたデータ値としてよりはむしろ、M E M 命令のオペランドとしてデータ処理システム 10 のユーザによって提供することができる。更に、A L U 5 6 は 3 2 ビットの結果が得られるように構成することもできる。乗算または減算操作のいずれかの場合に、前記ロー側およびハイ側の各々はスプリット動作モードの間に 1 6 ビットの結果を生成することになる。

【 0 1 3 8 】

本発明の原理がここに説明されたが、当業者には前記説明は実例によってのみ行われかつ本発明の範囲を制限するものとして行われたものでないことは明らかに理解されるべきである。

20

【 0 1 3 9 】

他の実施形態では、M E M 命令はエンドポイントと異なるファジー論理集合の特性を検出することによりあるファジー論理集合が最後のファジー論理集合であることを判定することができる。例えば、別のシステムではポイント 2 の値をポイント 1 の値より小さくなるようセットすることにより最後のファジー論理集合の発生を示すことができる。この場合、M E M 命令はこの別の特性を検出することになる。さらに、M E M 命令は完全に C P U 内でまたは C P U とコプロセッサの双方によって実行することができる。後者の場合、C P U は命令デコーダを含むものと考えられかつコプロセッサは実行ユニットを含むものと考えられる。したがって、添付の特許請求の範囲により、本発明の真の精神および範囲内にあるこの発明のすべての変形をカバーするものと考えている。

30

【 図面の簡単な説明 】

【 図 1 】 ファジー論理を説明するために使用される幾つかの概念および基本的な用語を示すグラフである。

【 図 2 】 図 1 のメンバシップ集合を詳細に示すグラフである。

【 図 3 】 本発明に係わるデータ処理システムの構成を示すブロック図である。

【 図 4 】 図 3 の実行ユニットの構成を示すブロック図である。

【 図 5 】 図 4 のフラグ発生論理回路の構成を示す論理回路である。

【 図 6 】 図 4 のオペランド割り当て論理回路の構成を示すブロック図である。

40

【 図 7 】 図 4 の演算論理ユニットの構成を示すブロック図である。

【 図 8 】 本発明に係わる M E M 命令の実行の間に行われる機能の流れを示すフローチャートである。

【 符号の説明 】

- 10 データ処理システム
- 12 中央処理ユニット (C P U)
- 14 実行ユニット
- 16 バス制御論理
- 18 命令デコード論理
- 20 制御ユニット

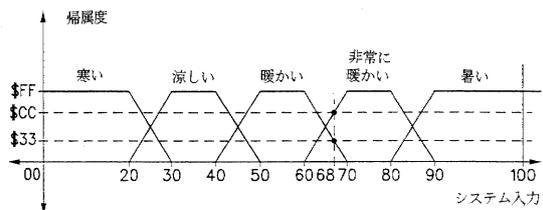
50

- 2 2 シーケンサ
- 2 4 発振器
- 2 6 電源回路
- 2 8 タイミング回路
- 3 0 外部バスインタフェース
- 3 2 内部メモリ
- 3 4 内部データバス
- 3 6 内部アドレスバス
- 3 8 タイミング制御バス
- 4 2 データバッファ
- 4 6 結果選択論理
- 4 8 フラグ発生論理
- 5 0 オペランド割り当て論理
- 5 2 A - 入力制御論理
- 5 4 A - 入力マルチプレクサ
- 5 6 A L U
- 5 8 B - 入力マルチプレクサ
- 6 0 デルタレジスタ
- 6 4 システム入力レジスタ
- 6 6 情報バス A
- 6 8 情報バス B
- 6 9 条件符号レジスタ
- 7 0 フラグバス
- 1 2 0 結果バス

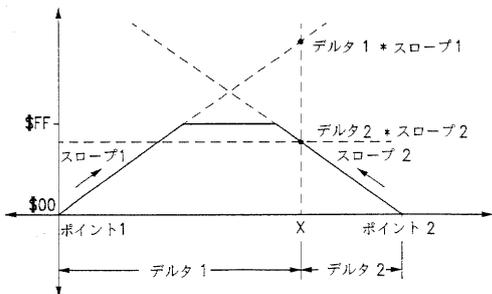
10

20

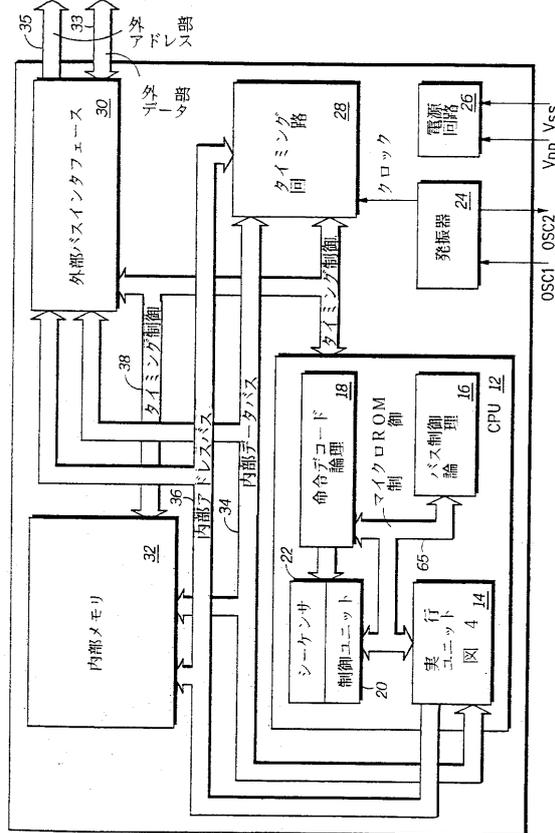
【 図 1 】



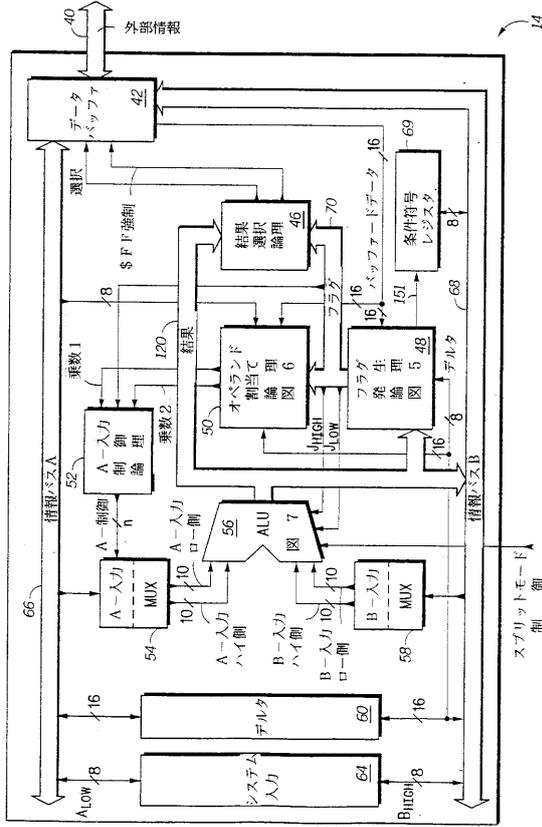
【 図 2 】



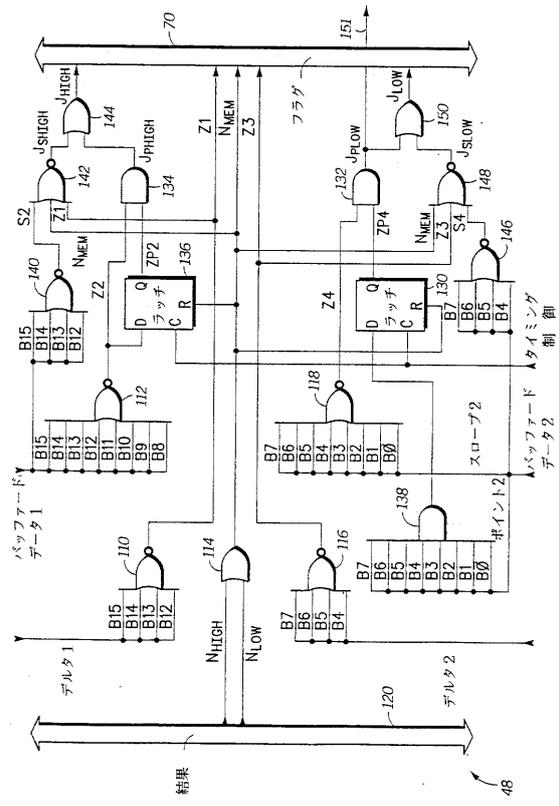
【 図 3 】



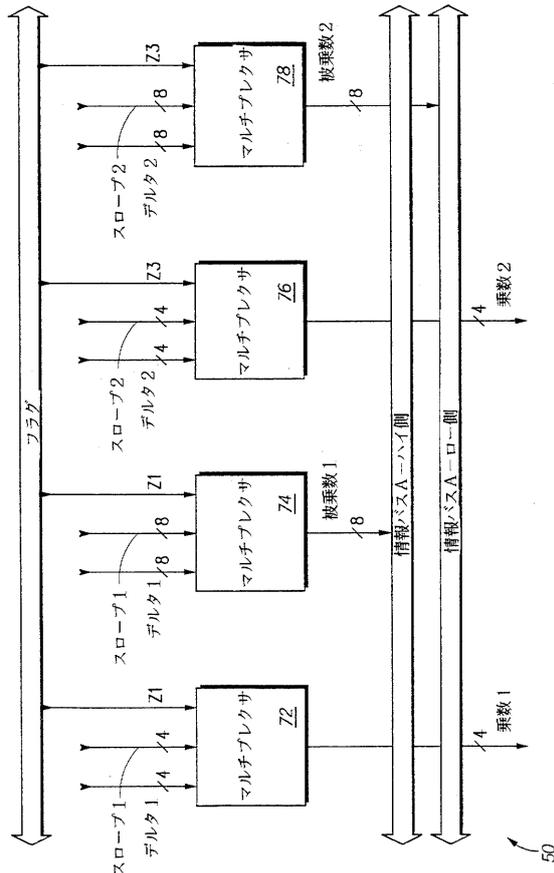
【 図 4 】



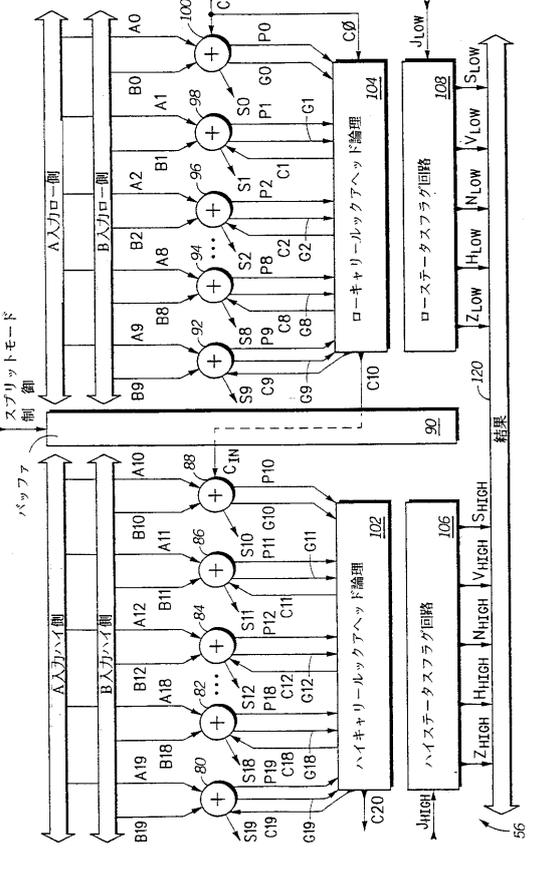
【 図 5 】



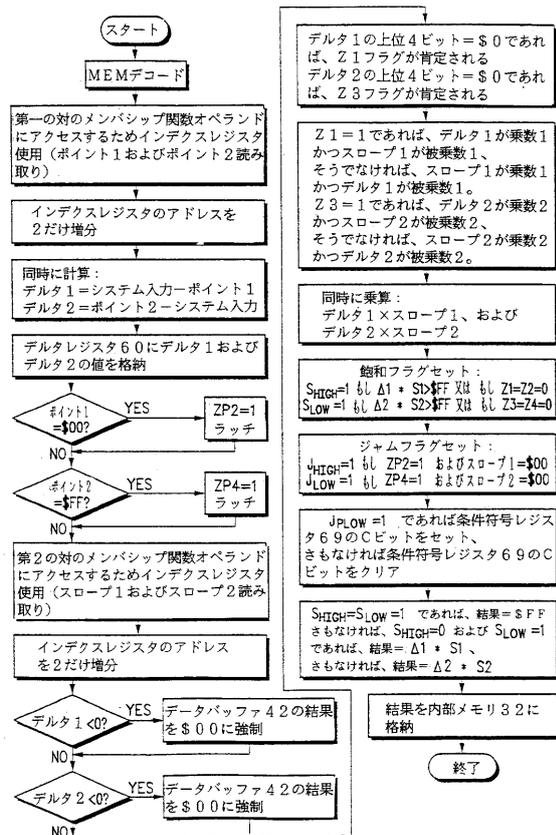
【 図 6 】



【 図 7 】



【 図 8 】



フロントページの続き

合議体

審判長 井関 守三

審判官 相崎 裕恒

審判官 青木 重徳

- (56)参考文献 特開平6 - 35708 (JP, A)
特開平6 - 51987 (JP, A)
特開平5 - 35308 (JP, A)
特開平6 - 301545 (JP, A)

- (58)調査した分野(Int.Cl., DB名)
G06N5/00-7/04