

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4209335号
(P4209335)

(45) 発行日 平成21年1月14日(2009. 1. 14)

(24) 登録日 平成20年10月31日(2008. 10. 31)

(51) Int.Cl.

F I

G 0 6 F 12/16 (2006. 01)

G 0 6 F 12/16 3 2 0 F

G 1 1 C 15/04 (2006. 01)

G 1 1 C 15/04 6 3 1 F

請求項の数 19 (全 28 頁)

(21) 出願番号 特願2003-581025 (P2003-581025)
 (86) (22) 出願日 平成14年10月14日(2002. 10. 14)
 (65) 公表番号 特表2005-521940 (P2005-521940A)
 (43) 公表日 平成17年7月21日(2005. 7. 21)
 (86) 国際出願番号 PCT/GB2002/004650
 (87) 国際公開番号 W02003/083669
 (87) 国際公開日 平成15年10月9日(2003. 10. 9)
 審査請求日 平成17年9月30日(2005. 9. 30)
 (31) 優先権主張番号 0207372. 4
 (32) 優先日 平成14年3月28日(2002. 3. 28)
 (33) 優先権主張国 英国 (GB)

(73) 特許権者 504363669
 コグニサイエンス・リミテッド
 イギリス国イングランド、エム13・9エ
 ックスエックス、マンチェスター、グラフ
 トン・ストリート、インキュベーター・ビ
 ルディング
 (74) 代理人 100082049
 弁理士 清水 敬一
 (72) 発明者 フーバー・ステファン・バイラム
 イギリス国イングランド、エスケイ9・5
 ディーユー、チェシャー、ウィルムスロー
 、ゴールセイ・ロード1エイ

審査官 堀江 義隆

最終頁に続く

(54) 【発明の名称】 デジタルメモリ

(57) 【特許請求の範囲】

【請求項 1】

第1の選択可能な状態と第2の選択可能な状態とをそれぞれ有する予め決められた数のビットを含む識別子がそれぞれ割り当てられた複数のアドレスデコーダと、予め決められた長さの複数のワードラインを有するデータメモリとを備え、各アドレスデコーダを付勢して、複数のワードラインの一つを選択することができ、アドレスデコーダは、予め決められた数のビットを有する入力アドレスを受信する手段と、アドレスデコーダの識別子を入力アドレスと比較する手段とを備え、データメモリは、更に、入力アドレス内で第1の選択可能な状態に設定された少なくとも予め決められた最小数のビットが、デコーダの識別子内で第1の選択可能な状態に設定されたビットに相当する場合に、アドレスデコーダを付勢する手段とを備えたことを特徴とするコンピュータシステムに使用するメモリ構造。

10

【請求項 2】

アドレスデコーダの識別子を入力アドレスと比較する手段は、入力アドレス内の第1の選択可能な状態に設定されたビットと、デコーダ識別子内の第1の選択可能な状態に設定されたビットとの間の位置一致(位置対応)を考慮する請求項1に記載のメモリ構造。

【請求項 3】

各アドレスデコーダ識別子は、第1の選択可能な状態に設定された等しい数のビットを有する請求項1又は2に記載のメモリ構造。

【請求項 4】

20

入力アドレスを受信する手段は、第 1 の選択可能な状態に設定された予め決められた数のビットを含むアドレスを受信するように構成される請求項 1 乃至 3 の何れか 1 項に記載のメモリ構造。

【請求項 5】

入力アドレス内の第 1 の選択可能な状態に設定された予め決められた数のビットは、各アドレスデコーダ識別子内の第 1 の選択可能な状態に設定されたビットの数に等しい請求項 3 又は 4 に記載のメモリ構造。

【請求項 6】

単一のビットメモリ内に各ワードラインの各ビットを記憶するように、データメモリは、単一ビットの複数のメモリを備える請求項 1 乃至 5 の何れか 1 項に記載のメモリ構造。

10

【請求項 7】

データメモリは、複数のワードラインの各々に等しい数のビットを有するデータ入力線を備える請求項 1 乃至 6 の何れか 1 項に記載のメモリ構造。

【請求項 8】

データ入力線からアドレスデコーダにより付勢されるワードラインにデータを複写するデータ書込み手段を備えた請求項 7 に記載のメモリ構造。

【請求項 9】

データ入力線は、第 1 の選択可能な状態に設定された予め決められた数のビットを含む入力データを受信するように構成される請求項 7 又は 8 に記載のメモリ構造。

【請求項 10】

20

アドレスデコーダにより付勢されるワードラインの各ビットに記憶される値を合計して、各ビットに対する付勢レベル値を作成する手段を備えた請求項 1 乃至 9 の何れか 1 項に記載のメモリ構造。

【請求項 11】

第 1 の選択可能な状態に設定された予め決められた数のビットを含む出力語を作成する手段を備えた請求項 9 又は 10 に記載のメモリ構造。

【請求項 12】

出力内で第 1 の選択可能な状態に設定されるビットは、最高の付勢レベルを有する予め決められた数のビットである請求項 11 に記載のメモリ構造。

【請求項 13】

30

互いに接続された複数の人工ニューロンを使用して、メモリを駆動し、ニューラルネットワークを形成する請求項 1 乃至 12 の何れか 1 項に記載のメモリ構造。

【請求項 14】

複数のアドレスデコーダニューロンにより複数のアドレスデコーダを表示し、複数のデータニューロンによりデータメモリを表示する請求項 13 に記載のメモリ構造。

【請求項 15】

各々第 1 の選択可能な状態と第 2 の選択可能な状態とを有しかつ予め決められた数のビットを含む識別子がそれぞれ割り当てられた複数のアドレスデコーダと、予め決められた長さの複数のワードラインを有するデータメモリとを備え、各アドレスデコーダを付勢して、複数のワードラインの一つを選択することができ、予め決められた数のビットを有する入力アドレスは、アドレスデコーダに入力され、アドレスデコーダの識別子は、入力アドレスと比較され、入力アドレス内で第 1 の選択可能な状態に設定された少なくとも予め決められた最小数のビットが、デコーダ識別子内の第 1 の選択可能な状態に設定されたビットに相当する場合に、アドレスデコーダが付勢されることを特徴とするコンピュータシステムに使用するメモリの操作法。

40

【請求項 16】

データメモリのデータ入力に入力データが付与され、付勢されたアドレスデコーダにより付勢されたワードラインにデータを書込む請求項 15 に記載の方法。

【請求項 17】

100 未満のアドレスデコーダが何等かの有効な入力アドレスによって付勢されるよう

50

に、予め決められた最小数のビットを設定する請求項 15 又は 16 に記載の方法。

【請求項 18】

何等かの有効な入力アドレスによって 50 に満たないアドレスデコーダを付勢するようにビットの予め決められた最小数を設定する請求項 17 に記載の方法。

【請求項 19】

20 個以下でかつ 11 個より少なくない数のアドレスデコーダを何等かの有効な入力アドレスにより付勢するように予め決められた最小数のビットを設定する請求項 18 に記載の方法。

【発明の詳細な説明】

【技術分野】

10

【0001】

本発明は、デジタルメモリに関連する。

【背景技術】

【0002】

コンピュータシステム及びデジタル電子装置の使用が増大してきたため、多数の種々のメモリ構造が提案されてきた。デジタルメモリを要求する広範囲な適用分野は、種々の異なる特性を有するメモリが要求されることを物語っている。

【0003】

例えば、誤った単一ビットがコンピュータの故障の原因となるので、従来のプログラム記憶型コンピュータに使用するメモリでは、高レベルのデータ完全性が要求される。従来のコンピュータメモリは、特定のアドレスに記憶された情報を検索し、読み出すのに正確に読出しアドレスが知られていることを要する。これは、従来のメモリがメモリ内の特定のロケーションに N2 進数字列（以下、「ワード」という）を記憶するためである。特定の時間に特定のロケーションに唯一のワードが記憶される。

20

【0004】

大量のデータを記憶する適用例では、メモリから読み出されるデータ中に含まれるある程度のエラーは許容され、システムがエラーを見逃すことができる場合もある。また、誤った読出しアドレスを使用してデータを読み出すことが必要な場合もある。本発明は、このような適用に適するメモリに関するものである。

【0005】

30

公知のメモリ構造は、カネルヴァ（Kanerva）により開発され、カネルヴァ・ピー（Kanerva P.）著下記非特許文献 1 及び非特許文献 2 に記載される。ここで、多量のアドレスデコーダが多数のワードラインを有するデータに接続される。各アドレスデコーダは、データメモリ内の異なるワードラインに接続される。各デコーダは、大きな 2 進数の形態でアドレスにより識別される。

【0006】

メモリにデータを書き込むときに、入力アドレスがデコーダのセットに付与され、入力アドレスの予め決められたハミング距離内に識別子を有するデコーダが付勢される。次に、付勢されたデコーダに接続されたワードラインが選択されて書き込みデータを受信する。メモリからデータを取り出すとき、デコーダにアドレスが付与され、多くのワードラインが再び付勢される。付勢された各ワードラインの各ビット（二進数字）を合計して、適切な閾値を適用することにより、読み出すべきメモリにデータを書き込むことができる。このように、データを書き込みかつ検索することによりアドレスを有効に機能させかつデータエラーを最小限に抑制することができる。

40

【特許文献 1】国際特許公報 W090/04830（大学宇宙研究協会）

【非特許文献 1】「疎分配メモリ」MIT プレス、1988 年発行

【非特許文献 2】「自己増殖探索：メモリの統合理論」報告書第 CSLI-84-7 スタンフォード大学 1984 年

【発明の開示】

【発明が解決しようとする課題】

50

【 0 0 0 7 】

カネルヴァが提案したメモリ構造は、多くの欠点を有する。これらの欠点の一つは、効率的にハミング距離の計算を行えば、特に設計したハードウェアを必要とする。

【 0 0 0 8 】

この欠点を克服することを探索する代替メモリ構造が上記特許文献 1 に記載されている。このシステムは、カネルヴァにより提案された基本構造を備えているが、異なる基準を使用して特殊な入力アドレスに応答してどのアドレスデコーダを付勢すべきかを決定しなければならない。このシステムでは、各デコーダアドレスの q ビットのサブセットに特定の「0」値及び「1」値が割り当てられる。システムに入力アドレスを付与するとき、付与されたアドレスの複数の q ビット値は、各アドレスデコーダのビット値と比較される。付与されたアドレスが、これらの q ビットで特定のアドレスデコーダと同一値を備えていれば、そのデコーダが付勢される。

10

【 0 0 0 9 】

本発明は、前記種類の代替メモリ構造を提供することを目的とする。

【課題を解決するための手段】

【 0 0 1 0 】

本発明の第一の特徴は、複数のアドレスデコーダと、データメモリとを備えかつコンピュータシステムに使用するメモリ構造を提供する点にあり、各アドレスデコーダは、予め決められた数のビットを有する識別子が割り当てられ、各ビットは、第 1 の選択可能な状態と第 2 の選択可能な状態とを有し、データメモリは、予め決められた長さを有する複数のワードライン（語線）を有し、各アドレスデコーダを付勢して、複数のワードラインの一つを選択することができ、アドレスデコーダは、予め決められた数のビットを有する入力アドレスを受信する手段と、アドレスデコーダの識別子を入力アドレスと比較する手段とを備え、データメモリは、更に、入力アドレス内で第 1 の選択可能な状態に設定された少なくとも予め決められた最小数のビットが、デコーダの識別子内で第 1 の選択可能な状態に設定されたビットに相当する場合に、アドレスデコーダを付勢する手段とを備えている。

20

【 0 0 1 1 】

本発明により得られる利点と効果は、第 2 の選択可能な状態に設定されたビットを無視して、入力アドレス内の第 1 の選択可能な状態に設定された基準ビット上でアドレスデコーダを付勢することにより部分的に発生する。対照的に、前記従来のシステムは、第 1 の選択可能な状態と第 2 の選択可能な状態とに設定されたビットを比較することによりアドレスデコーダを付勢する。

30

【 0 0 1 2 】

アドレスデコーダの識別子を入力アドレスと比較する手段は、入力アドレス内の第 1 の選択可能な状態に設定されたビットと、デコーダ識別子内の第 1 の選択可能な状態に設定されたビットとの間の位置一致（位置対応）を考慮することが好ましい。

【 0 0 1 3 】

各アドレスデコーダ識別子は、第 1 の選択可能な状態に設定された等しい数のビットを有してもよい。入力アドレスを受信する手段は、第 1 の選択可能な状態に設定された予め決められた数のビットを含むアドレスを受信するように構成される。入力アドレス内の第 1 の選択可能な状態に設定された予め決められた数のビットは、各アドレスデコーダ識別子内の第 1 の選択可能な状態に設定されたビットの数に等しいことが好適である。

40

【 0 0 1 4 】

単一のビットメモリ内に各ワードラインの各ビットを記憶するように、データメモリは、単一ビットの複数のメモリを備えることが好ましい。データメモリは、複数のワードラインの各々に等しい数のビットを有するデータ入力線を備えることが好ましい。また、メモリ構造は、データ入力線からアドレスデコーダにより付勢されるワードラインにデータを複写するデータ書込み手段を有してもよい。データ入力線は、第 1 の選択可能な状態に設定された予め決められた数のビットを含む入力データを受信するように構成される。

50

【 0 0 1 5 】

また、メモリ構造は、アドレスデコーダにより付勢されるワードラインの各ビットに記憶される値を合計して、各ビットに対する付勢レベル値を作成する手段を備えてもよい。更に、メモリ構造は、第 1 の選択可能な状態に設定された予め決められた数のビットを含む出力語を作成する手段を備えてもよい。出力内で第 1 の選択可能な状態に設定されるビットは、最高の付勢レベルを有する予め決められた数のビットでもよい。

【 0 0 1 6 】

互いに接続された複数の人工ニューロンを使用して、メモリを駆動し、ニューラルネットワークを形成してもよい。複数のアドレスデコーダニューロンにより複数のアドレスデコーダを表示して、複数のデータニューロンによりデータメモリを表示してもよい。

10

【 0 0 1 7 】

また、本発明は、コンピュータシステムに使用するニューラルメモリ構造を提供するものであり、このメモリ構造は、各々予め決められた数の入力ニューロンに接続された複数のアドレスデコーダニューロンと、複数のデータニューロンを有するデータメモリとを備え、各アドレスデコーダニューロンを付勢して、複数のデータニューロンの幾つかを選択することができ、アドレスデコーダニューロンは、それが接続された入力ニューロンの発火を表示する信号を受信する手段を備え、アドレスデコーダニューロンは、アドレスデコーダが連結された少なくとも予め決められた数の入力ニューロンから発火信号を受信したとき、データニューロンを付勢する手段を備えている。

20

【 0 0 1 8 】

実際に、全てのアドレスデコーダニューロンを全ての入力ニューロンに接続して、これらの各連結値に「 1 」又は「 0 」の加重値が割り当てられる。このような設置計画では、加重値「 0 」は、特定の入力ニューロンとアドレスデコーダニューロンとの間が非連結と同一の効果を有する。このように、連結値が加重値「 1 」を有するようなアドレスデコーダニューロンに接続された入力ニューロンからの発火信号のみが、前記のように、連結を表示する。

【 0 0 1 9 】

本発明の第 2 の特徴は、コンピュータシステムに使用するメモリの操作法を提供することであり、このメモリは、各々第 1 の選択可能な状態と第 2 の選択可能な状態とを有しかつ予め決められた数のビットを含む識別子がそれぞれ割り当てられた複数のアドレスデコーダと、予め決められた長さの複数のワードラインを有するデータメモリとを備え、各アドレスデコーダを付勢して、複数のワードラインの一つを選択することができ、予め決められた数のビットを有する入力アドレスは、アドレスデコーダに入力され、アドレスデコーダの識別子は、入力アドレスと比較され、入力アドレス内で第 1 の選択可能な状態に設定された少なくとも予め決められた最小数のビットが、デコーダ識別子内の第 1 の選択可能な状態に設定されたビットに相当する場合に、アドレスデコーダが付勢される。

30

【 0 0 2 0 】

データメモリのデータ入力に入力データが付与され、付勢されたアドレスデコーダにより付勢されたワードラインにデータを書込むことができる。100 未満のアドレスデコーダが何等かの有効な入力アドレスによって付勢されるように、予め決められた最小数のビットを設定することが好ましい。ビットの予め決められた最小数は、何等かの有効な入力アドレスによって 50 に満たないアドレスデコーダを付勢するように設定されることがより好ましい。20 個以下でかつ 11 個より少なくない数のアドレスデコーダを何等かの有効な入力アドレスにより付勢するように、予め決められた最小数のビットを設定することが最も好ましい。

40

【 0 0 2 1 】

本発明の第 3 の特徴は、コンピュータメモリの操作を最適化する方法を提供することであり、このコンピュータメモリは、第 1 の選択可能な状態と第 2 の選択可能な状態とをそれぞれ有する予め決められた数のビットを含む識別子がそれぞれ割り当てられた複数のアドレスレコーダと、予め決められた長さの複数のワードラインを有するデータメモリとを

50

備え、各アドレスデコーダを付勢して複数のワードラインの一つを選択することができ、更にデータメモリは、入力アドレスを受信する手段と、デコーダ識別子と入力アドレスとの間の比較が予め決められた比較閾値を超える場合にアドレスデコーダの1つ又はそれ以上を付勢する手段とを備え、最適化法は、有効な入力アドレスに応答して付勢すべきアドレスデコーダの操作上有利な数を決定する過程と、有効な入力アドレスが付勢すべきアドレスデコーダの操作上有利な数に実質的に等しい数のアドレスデコーダを付勢するように、比較閾値を設定する過程とを含む。

【0022】

この比較は、入力アドレス内の第1の選択可能な状態に設定されたビットの数を、各アドレスデコーダ識別子内で第1の選択可能な状態に設定されたビットの数と比較するものである。

10

【0023】

データメモリから最大のエラー無しデータを回復するように、操作上有利な数を決定することができる。下式の関数を使用してこれを行うことができる。

【0024】

$$w=f(h, P_c, D, d)$$

ここで、 w は、付勢すべきアドレスデコーダの操作上有利な数、 h は、第1の選択可能な状態に設定されるデータメモリ内の任意ビットの確率、 D は、各ワードラインが第1の選択可能な状態に設定された数 d のビットを有する場合の各ワードライン内のビット数、 P_c は、データ記憶装置から正しく回復されたビットの所望確率を示す。

20

【0025】

下記の関数を使用できる：

【0026】

【数5】

$$h^{-w} \cdot w = \frac{\ln(1-h)}{\ln(P_c)} \cdot W \cdot \frac{D}{d} \cdot (D-d)$$

【0027】

幾つかのデータをエラーと共に回復しながら、最大のエラー無しデータを回復するように、付勢すべきアドレスデコーダの操作上有利な数を決定することができる。下記の関数を使用して、操作上有利な数を決定することができる：

30

$$w=f(h, D, d)$$

【0028】

ここで、 w は、付勢すべきアドレスデコーダの操作上有利な数、 h は、第1の選択可能な状態に設定されるデータメモリ内の任意ビットの確率、 D は、各ワードラインが第1の選択可能な状態に設定された数 d のビットを有する場合の各ワードライン内のビット数である。

【0029】

下記の関数を使用できる：

【0030】

40

【数6】

$$\frac{(h^{-w}-1)}{w} = -(D-d) \cdot \ln h$$

【0031】

操作上有利な数 w を設定して、下式により与えられるものより大きな値を有することが好ましい。

【0032】

【数 7】

$$\frac{(h^{-w}-1)}{w} = -(D-d) \cdot \ln h$$

及び等式により付与されるものより小さい。

【0033】

【数 8】

$$h^{-w} \cdot w = \frac{\ln(1-h)}{\ln(P_c)} \cdot W \cdot \frac{D}{d} \cdot (D-d)$$

10

【0034】

更に、本発明は、キャリア媒体を保持するコンピュータにより読出し可能なコード手段を提供し、このコード手段は、前記方法による手順及び前記方法を実行するコンピュータプログラムをコンピュータに実行させるものである。

【発明を実施するための最良の形態】

【0035】

以下、本発明による実施の形態の例を図面について以下説明する。

【0036】

図1は、カネルヴァが説明しかつ前記したアドレス構造を示す。アドレスを伝達できる複数のアドレスライン1が設けられる。アドレスは、二進数であり、各アドレスライン1は、nビットアドレスを表示するため、n個のラインを必要とするように、各アドレスライン1は、1ビットのアドレスを伝達する。アドレスラインは、アドレスデコーダ2のアイへの入力を構成し、各アドレスデコーダ2は、nビット二進数の形式の識別子を有する。アドレスデコーダ2は、データを記憶する多数のワードラインを有するデータメモリ3に接続される。各ワードラインは、Mビットの記憶容量を有する。各ワードラインの各ビットは、カウンタ4に記憶される。データメモリに書き込むべきデータは、データ入力ライン5上で伝達される。メモリから読み出されるデータは、データライン8に沿って加算ブロック6に送出され、加算ブロック6からデータ出力ライン7上に出力される。

20

【0037】

アドレスライン1の数は、典型的には1千ビット二進数をアドレスデコーダ2に付与するように約一千本である。1千ビット二進数（約10の301乗）によって非常に多くのアドレスを表示することができるので、各可能なアドレスに対するデータを記憶する独立したワードラインを持つことは実用的ではない。従って、可能なアドレススペース内に大きな任意のアドレスサンプルが実行される。典型的には、100万個のアドレスが実行され、これらの各々は、アドレスデコーダ2に割り当てられる。100万個（10の6乗）のデコーダアドレスは、利用可能なアドレス空間（10の301乗アドレス）内にまばらに分散されるので、カネルヴァにより提案されたメモリは、まばらとして記載される理由である。

30

【0038】

使用の際に、アドレスはアドレスライン1上で伝達され、アドレスデコーダ2により受信される。伝達されるアドレスと各アドレスデコーダのアドレスとの間の比較が行われる。この比較は、伝達されるアドレスのどれだけ多くのビットが各アドレスデコーダ2に関連するアドレスに対して相違するかを演算するハミング距離計算を使用して実施される。ハミング距離計算は、式(1)により表される：

40

【0039】

【数 9】

$$H(x, y) = \sum_{i=1}^N |x_i - y_i| \quad (1)$$

【0040】

50

ここで、 x は、 N 要素を有する二進ベクトル、 y は、 N 要素を有する二進ベクトル、 H は、 x と y との間のハミング距離で、 $x_i \neq y_i$ でのビット位置数である。

【0041】

伝達されるアドレス x の予め決められたハミング距離 r 内のデコードが付勢される。各アドレスに対するこの付勢領域は、半径 r を有しかつ x に中心を有する球とみなすことができる。例えば、カネルヴァは、各アドレスが1000ビットを有し、予め決められたハミング距離 r が451であれば、球形の付勢領域は、アドレスデコードに関連するアドレスの約1/1000を含む。使用の際に、各デコードアドレスと付与される入力アドレスとの間のハミング距離が演算される。この距離が、特定のアドレスデコードの r より小さければ、そのデコードは付勢される。

10

【0042】

カネルヴァにより提案されたシステムでデータをメモリに書き込む過程の詳細を説明する。所与のアドレスデコード2を付勢するとき、メモリのデータ入力端子5に付与されるデータは、付勢されたアドレスデコードに接続されたワードラインに書き込まれる。予め決められたそのアドレスデコードのハミング距離内にアドレスが発生するときはいつでもアドレスデコードの付勢が開始されるので、データは、多くの異なるワードラインに書き込まれる。同様に、同一のアドレスデコードを付勢する多くの種々のアドレスも存在する。これは、多数の種々の入力書込みアドレスに対する特定のワードラインにデータが書き込まれることを意味する。

【0043】

20

各ワードラインは、それぞれ単一ビットのデータを記憶する M 個のカウンを有する。所与のワードラインでのカウンタにより、そのワードラインに記憶される多数の M ビット語の直線の蓄積が可能となる。ワードラインの特定のビットに「1」を付与すれば、対応するカウンタ4は、増加される。特定のビットで「0」を付与すれば、そのビットに関するカウンタ4は、減少される。

【0044】

カウンタは、有限の容量を有し、継続するデータの書き込みによってこの容量に達すれば、データの更なる書込みは無効である。例えば、継続する「1」の書き込みによってカウンタが最大値に達すれば、そのカウンタに更に「1」を書き込むことは無効である。同様に、継続する「0」の書き込みによってカウンタが最小値に達すれば、そのカウンタに更に「0」を書き込むことは無効である。それぞれ±127の範囲を有する8ビットのカウンタを使用すれば通常十分である。実際に、2ビット又は4ビットのカウンタを使用すれば十分な場合もある。しかしながら、カネルヴァにより提案されるメモリでは、いかなるビット長さのアップダウンカウンタも相当の余裕があることに注意すべきである。

30

【0045】

100万個のアドレスデコード2に対して、任意のアドレスは、デコードの約100の予め決められたハミング距離内にある。このように、任意の入力アドレスに対して、入力データは、データメモリ3内に1000ワードラインに書き込まれ、各ワードラインの適当なカウンタ4は、前記のように増加され又は減少される。

【0046】

40

カネルヴァが提案したシステムでメモリからデータを読み出す過程の詳細を説明する。

【0047】

アドレスライン1上に入力アドレスを付与することによりデータがメモリから読み出される。この入力アドレスが、読み出すべきデータを書き込むのに使用したものと同一であれば、同一のアドレスデコードが付勢され、正しいワードラインからデータが読み出される。

【0048】

書込みに使用したアドレスに近いアドレスが付与されれば、データを書き込むために付勢された多くのアドレスデコード2は、データを読み出すために付勢される。しかしながら、書込みのために付勢されたいくつかのデコードは付勢されず、以前に付勢されなかつ

50

た他のデコーダが付勢されよう。これにより、読み出されるデータ内にノイズが存在する。読出し過程と書込み過程との間で共通の十分な数のデコーダがあれば、正しいデータは、なお加算器 6 から出力され、ノイズの影響は減少する。これは、カネルヴァにより記載されるメモリの有利な特徴である。

【 0 0 4 9 】

付勢されるアドレスデコーダに接続される各ワードラインのカウンタ 4 内に保持される値は、読出しサイクル間にデータライン 8 上に出力される。各データライン 8 上に出力される値は、加算器 6 により加算される。予め決められた閾値を超える出力の加算器は、出力データライン 7 上に出力「 1 」を発生し、その閾値を超えないものは、出力「 0 」を発生する。

10

【 0 0 5 0 】

メモリの構造及び機能は、入力アドレス内の 1 千ビットのものを意味し、その同一の入力アドレスから正しく読み出すのに僅かに 6 0 0 ビットが読出しアドレス中に正しいものと要求することをカネルヴァが決定した。例えば、データの書込み間に 1 千のワードライン上にビットを「 1 」に設定すれば、そのビットに対して値を保持するカウンタに接続される加算器 6 は、1 0 0 0 を出力するであろう。しかしながら、データが書き込まれた 1 千ワードラインの大多数を検索（読出し）アドレスにより読み出すのであれば、一般的に加算器は、正しく解釈される値を出力するであろう。このように、カネルヴァにより提案されたメモリは、有用なエラー修正特性を有し、前記のように、ノイズの影響を緩和するものである。

20

【 0 0 5 1 】

N-Mコードは、Mビット二進数のNビットを「 1 」に設定したコードである。従って、如何なるN-Mコードも、「 1 」に設定されるNビット及び「 0 」に設定されたM-Nビットを有する。コードにより表される各値は、「 1 」及び「 0 」が発生する順番で異なる。N-Mコードは、N「 1 」を受信したとき、コードが完全である（即ち、値を受信した）が、N「 1 」より小さい値を受信したとき、コードは不完全である点で自己タイミングである。

【 0 0 5 2 】

本発明のメモリ構造は、図 1 に示す従来のメモリの散在分散特性を有し、メモリに多数のエンハンスメントを付与するようにN-Mコード付けを幾つか適用するものであるが、この点は後述する。本発明で実施するメモリ構造を図 2 に高レベルで図示する。一般的構造は図 1 と類似するので、適切な場合には類似の参照数字を使用する。図 1 のメモリと本発明により提供されるメモリとの僅かな相違は後述の通りである。

30

【 0 0 5 3 】

一般に、メモリは、特定の読出し又は書込みに対して付勢されるwのWアドレスデコーダ 2 を有する。各Wアドレスデコーダは、Aビットアドレスを有し、これらのアドレスの各々は、値「 1 」を有するaビットを備えている。このように、各アドレスデコーダのアドレスは、A-aコードである。アドレスがアドレスデコーダ 2 に入力されるとき、「 1 」値に設定されたiビットと、「 0 」値に設定されたA-iビットとを有する。このように、如何なる入力アドレスもA-iコードである。全ての入力アドレスは、デコーダ(i=a)のアドレスに等しい「 1 」ビット数を有する実施の形態もある。しかしながら、iをaから独立させる点において通常限界利得が存在する。

40

【 0 0 5 4 】

複数のアドレスデコーダ 2 が入力アドレスを受信したとき、各デコーダのA-aコードと、入力アドレスのA-iコードとの間を比較する。各アドレスデコーダは、関連する閾値Tを有する。閾値Tは、入力アドレスとデコーダアドレスとの間で一致しなければならない「 1 」の数を決定する。

【 0 0 5 5 】

デコーダ付勢手順を単純化する例を説明する。これらの例では、3つのアドレスデコーダ（A、B及びC）が存在し、それぞれ8-3コードを有するものと仮定する。これらの

50

デコーダのアドレスは、下記の通りである。

A: 01001001
B: 10001010
C: 01101000

【 0 0 5 6 】

下記の説明では、各アドレスは、指数 0 で始まり、左から右に索引付けされるものとする。

【 0 0 5 7 】

第 1 の実施の形態では、各デコーダの閾値は、入力アドレスの少なくとも 2 つのビットがデコーダアドレスの 2 つのビットに一致するように設定される。即ち、下記が付与される：

アドレス 1 : 01001000 (8 - 2 コード)

【 0 0 5 8 】

入力アドレスのビット 1 及び 4 が「 1 」であり、デコーダ A と C のアドレスのビット 1 及び 4 も「 1 」のとき、デコーダ A と C とが付勢されることは理解できよう。このように、2 つの一致ビットの閾値が達成される。アドレス 1 とデコーダ B との間で「 1 」の 1 つのみが一致 (ビット 4) するので、デコーダ B は、付勢されない。

【 0 0 5 9 】

同様に、下記アドレス：

アドレス 2 : 01000011 (8 - 3 コード)

が付与され、閾値 T が変化しないと仮定すると、アドレス 2 は、ビット 1 と 7 内に「 1 」を有し、デコーダ A が付勢されることを意味することを理解できよう。各デコーダアドレスの「 1 」ビットのみがアドレス 2 内の「 1 」と一致するので、デコーダ B と C とは付勢されない。

【 0 0 6 0 】

前記付勢手順は、W の全体から w デコーダを付勢する。従って、付勢パターンは、W-w コードである。所与の読出しサイクル又は書込みサイクルの間に、最適数又は最適に近い数のアドレスデコーダを確実に発火するように、閾値を設定することが好ましい点で重要であるが、これは、図 1 について説明した過程に類似する。この最適数は、カネルヴァが提案したメモリ内で付勢される数よりも典型的に極めて小さいことが判明した。発明者は、より少ない数のデコーダを付勢して十分に高精度でメモリからデータの読出し及び書込みが可能であること及びカネルヴァのシステムのように、1 千個のデコーダの付勢を要しないことを実現した。メモリシステムのこの特性の詳細な分析は、後述の通りである。

【 0 0 6 1 】

次に、w 付勢デコーダは、データを読み出すべき又は書込むべき w ワードラインを選択する。このように、アドレスデコーダ 2 と、データメモリ 3 のワードラインとの間の連絡は、W-w コードである (即ち、全てにおいて W デコーダの w 付勢デコーダ) 。前記の通り、独立に付勢されるアドレスデコーダを含む付勢手順が与えられれば、w を確実に一定にすることは、通常実用的ではない。アドレスデコーダ 2 からデータメモリ 3 に W コードのこの w を連絡することは、図 2 の矢印 9 により図示する。

【 0 0 6 2 】

w 付勢デコーダは、w ワードラインを選択する。書込みデータを付与して、データをデータメモリ 3 に書込み、読出しデータは、データメモリ 3 から読み出される。読出し又は書込みは、各 w 付勢ワードラインで実行される。入力データ 5 は、D-d コードを表すものと判断され、各ワードラインは、D ビット、「 1 」に設定された「 d 」を含む。各入力 D-d コードであれば、各書込みデータ操作は、d 「 1 」を含む書込みデータである。

【 0 0 6 3 】

各ワードラインの各ビットは、従来の二進単一ビットメモリ内に記憶される。全てのワードラインの全てのビットは、最初に「 0 」に設定される。「 1 」をビットに書込むべきとき、「 0 」値は、「 1 」により置き換えられる。後続する書込みサイクルで「 0 」をそ

10

20

30

40

50

のビットに書込もうとする場合でも、一度ビットが「0」に設定されれば、決して「0」にリセットされない。一度データ書込み操作が1つの値を設定すれば、この値のその後の増加は、単にノイズとなり、従ってその増加を破棄できるためである。この記憶手順は、M-Nコードの単極特性に基づき、その信頼性の理由を後述する。

【0064】

データメモリ3からデータを読み出すべきときに、アドレスデコーダ2にアドレスを付与すると、アドレスデコーダ2は、複数のwアドレスデコーダを付勢する。次に、これらのwアドレスデコーダは、データメモリ3内のwワードラインを付勢する。各語は、Dビットを含み、付勢された各ワードラインの各ビットの内容は、合計されて、D出力を発生する。本明細書では、これらの合計を「付勢レベル」と指称する。全ての書込みデータは、D-dコードであることは知られており、従って、メモリから読み出されるデータもD-dコードである。

10

【0065】

単一の書込みサイクルが特定のアドレスに値を書込み、読出しデータ操作に対して同一のアドレスが付与される場合（更に介在書込みサイクルがなく）、書込みデータ内で「1」に設定される各ビットの付勢レベルは、wであり、書込みデータ内で「0」に設定される各ビットの付勢レベルは、零である。このように、出力データ内でどのビットを「1」に設定すべきかを容易に理解することができる。

【0066】

しかしながら、2つの書込みサイクルが複数の異なる入力書込みアドレスにデータを書込む場合に、第1の書込みサイクルの間に付勢されるアドレスデコーダは、第2の書込みサイクルの間に付勢されるものも存在するかもしれない。即ち、データメモリ3内のワードラインは、第1の書込みサイクルの結果として「1」に設定されるビットと、第2の書込みサイクルの結果として「1」に設定されるビットとを含むものがある。

20

【0067】

第1の書込みアドレスに等しいアドレスをアドレスデコーダ2に付与するとき、読出しデータ手順は、wワードラインを付勢し、前記のように、付勢レベルを計算する。介在書込みサイクルの動作があれば、これらの付勢レベルは、第2の書込みから誘導される幾つかの値を含み、即ち、第1の書込みサイクルの間にこれらの付勢レベルを「1」に設定されるビットよりも小さくすべきであっても、第1の書込みサイクル内で「0」に設定された幾つかのビットは、非零付勢レベルを有する。即ち、dビットより多いと、非零付勢レベルを有する。

30

【0068】

出力データをD-dコードとすべきことが既知であるから、修正された勝利者取り分全て（winner takes all, WTA）のアルゴリズムを適用して、最大の付勢レベルを有するdビットを決定する。これらのdビットは、そのアドレスにデータを書込む間に「1」に設定された可能性の最も高いものであり、従って、出力データ内で「1」に設定される。

【0069】

WTAアルゴリズムは、通常データの集合から単一の勝利者を決定する。この場合に、修正されたアルゴリズム（本明細書では、「k-WTAアルゴリズム」と称する）は、最高の付勢レベルを有する複数のd値を決定し、これらの値を有するビットは、出力内で「1」に設定される。即ち、これらのdビットは、アルゴリズム内の「勝利者」である。適当な分類（ソート）アルゴリズムを使用して、D値を分類し、これらの分類された値から最高のd値を選択することにより、k-WTAアルゴリズムを実行することもできる。

40

【0070】

データメモリにM-Nコード付けを適用することにより、出力データ内に如何に多くの「1」が存在するかを知るように、読出し過程でこの形態を使用することができる。カネルヴァが提案したシステムでは、メモリに書込むデータが二進数の場合に、出力データ中に如何に多くの「1」が存在するかを認識する方法がないから、カネルヴァは、記憶素子として複数のカウンタを使用して、各ビットに書込まれるデータを蓄積し、カウンタの内容

50

を合計し、閾値を適用して出力を発生する。

【 0 0 7 1 】

単一のビットメモリは、「 1 」を超えて増大されないことは前記の通りである。即ち、これは、特定のビットでの値が常に「 1 」又は「 0 」の何れかであるべきだからである。従って、「 1 」を超えて増大することは、そのビットに不都合なあるノイズが記憶されることを意味しなければならない。このノイズは、共通のワードラインを付勢する 2 つの書込みアドレスによって発生される。この場合に、「 1 」を超えて増大できるカウンタを必要とせず、どのビットを「 1 」に設定すべきかを k-WTA アルゴリズムが正確に特定することができる。アップダウンカウンタ及び零閾値を使用するカネルヴァの実施法は、「 0 」又は「 1 」の何れに特定の出力ビットを設定すべきかを決定するのに最早必要としない。このように、本発明は、カウンタではなく、従来の単一ビットメモリ内にデータを記憶させるメモリを提供する。これにより、装置の価格を大幅に低減することができる。また、近時の技術革新により大容量の適当なメモリ (R A M) を安価で効率的に利用することができる。

10

【 0 0 7 2 】

メモリの各ビットに対して有用に記憶できる情報のビット数から見れば、メモリの有効性の一般的な基準は、記憶容量である。この基準を使用して、本発明を実施するシステムは、カネルヴァが提案したシステムより優れた顕著な利点を生ずる。この大記憶容量の主な理由は、より少ない数のワードラインにデータを書込むことから得られる。典型的には、カネルヴァのシステムが数千ワードラインにデータを書込んだのに対し、本発明では、発明者は、カネルヴァのメモリの利益を保持ししかも格段に大きな記憶効率を得ながら、遥かに少ない数のワードラインにデータを書込むことができることを発見したのである。例えば、下記の記憶特性に対して、データを 1 0 0 ワードラインより少なく書込むことが好ましく、5 0 ワードライン以下にデータを書込むことが更に好ましく、1 1 と 2 0 ワードラインの間でデータを書込むことが操作上最大の利点が生ずる。データを書込むべき最適数のワードライン数を決定する方程式と共に、記憶容量について説明する。

20

【 0 0 7 3 】

データメモリ 3 は、相関行列メモリの形態である。相関行列メモリは、例えば、ティー・コーエン著「相関行列メモリ」電気電子学会トランスアクションズ・コンピュータ・C - 2 1 (4) 1 9 7 2 年 4 月第 3 5 3 ~ 3 5 9 頁、エム・ターナー及びジェ・オースチン著「二進数相関行列メモリの整合性」ニューラル・ネットワーク 10 (9) 、 1 9 9 7 年 1 6 3 7 ~ 1 6 4 8 頁において広く研究されてきた。この解析の概要を誘導体 w の基礎として本明細書中に示す。

30

【 0 0 7 4 】

アドレス入力及びデータ入力がそれらの空間全体に均一に分散されると仮定する。データメモリ 3 は、最初に各ビットで「 0 」を含む。書込み過程でのある点で「 1 」に設定されるデータメモリ内の任意のビットの確率を表す占有量 h を使用する。Z 書込み操作後に、期待される占有量は、方程式 (2) で与えられる。

【 0 0 7 5 】

【 数 1 0 】

40

$$h = 1 - \left[1 - \left(\frac{w}{W} \cdot \frac{d}{D} \right) \right]^Z \quad (2)$$

【 0 0 7 6 】

方程式 (2) の下記条件：

【 0 0 7 7 】

【 数 1 1 】

$$\left(\frac{w}{W} \cdot \frac{d}{D} \right)$$

50

【 0 0 7 8 】

は、単一の書込み操作内でデータメモリ内の特定のビットを「1」に設定する確率を与える。1からこの条件を減じると、特定の操作内でそのビットを「1」に設定しない確率値が与えられる。この減算値は、Zの書込み操作の効果を表示するように、更にZ乗される。従って、下記項：

【 0 0 7 9 】

【 数 1 2 】

$$\left[1 - \left(\frac{w}{W} \cdot \frac{d}{D}\right)\right]^Z$$

10

【 0 0 8 0 】

は、Zの書込み操作後に、任意のビットを「1」に設定しない確率を与える。このように、Z操作後に任意のビットを1に設定する確率：hを誘導するため、この条件は、1から差し引かれて方程式(2)を生ずる。

【 0 0 8 1 】

Zは、0であるから、hは、最初に0であり、方程式の右辺は、単純に1-1となることは方程式(2)から理解できよう。より多くのデータがメモリ内に記憶されるとき、hは、1に向かって単調に増加する。

【 0 0 8 2 】

対数を使用して、方程式(2)を下記の通り書き換えることができる。

20

【 0 0 8 3 】

【 数 1 3 】

$$\ln(1-h) = Z \ln \left[1 - \left(\frac{w}{W} \cdot \frac{d}{D}\right)\right] \quad (3)$$

【 0 0 8 4 】

典型的に、dがDより極めて小さく、wがWより極めて小さいと仮定すれば、下記項：

【 0 0 8 5 】

【 数 1 4 】

$$\left(\frac{w}{W} \cdot \frac{d}{D}\right)$$

30

【 0 0 8 6 】

は、小さい。従って、方程式の右辺の自然対数をテイラー級数の初項にほぼ等しいとみなせる。これは、方程式(4)に記載するように、記憶されたデータ項目の数に対する式を与える。

【 0 0 8 7 】

【 数 1 5 】

$$Z = -\ln(1-h) \cdot \frac{W}{w} \cdot \frac{D}{d} \quad (4)$$

40

【 0 0 8 8 】

D-dコードによって表すことができる記号の数を考慮することにより、Z書込み操作後に、メモリ内に記憶される全情報を特定することができる。この情報は、方程式(5)で与えられる。

【 0 0 8 9 】

【 数 1 6 】

$$I(Z) = Z \cdot \log_2 [C_d^d] \text{ bits} \quad (5)$$

【 0 0 9 0 】

50

方程式(5)で与えられる情報容量をデータメモリ内の二進数記憶ロケーションの数と比較することにより、方程式(6)で表される記憶効率を演算することができる。

【 0 0 9 1 】

【 数 1 7 】

$$\eta(Z) = \frac{I(Z)}{D \cdot W} = \frac{Z}{D \cdot W} \cdot \log_2 [C_D^d] \quad (6)$$

【 0 0 9 2 】

ここで、 $\eta(Z)$ は、メモリの記憶効率である。

【 0 0 9 3 】

10

Dの大きな値に対して、方程式(6)の組み合わせ項を下式により近似させることができる：

【 0 0 9 4 】

【 数 1 8 】

$$\log_2 [C_D^d] = -D \cdot \left[\frac{d}{D} \cdot \log_2 \frac{d}{D} + \left(1 - \frac{d}{D} \right) \cdot \log_2 \left(1 - \frac{d}{D} \right) \right] \quad (7)$$

【 0 0 9 5 】

dがDより極めて小さい場合の散在メモリに対して、方程式(7)と(4)とを方程式(6)に代入し、簡素化すると下式が得られる：

20

【 0 0 9 6 】

【 数 1 9 】

$$\eta(Z) = -\ln(1-h) \cdot \frac{1}{w} \cdot \left[\log_2 \frac{D}{d} + 1 \right] \quad (8)$$

【 0 0 9 7 】

ここで、 $\eta(Z)$ は、メモリの記憶効率である。

【 0 0 9 8 】

これは、零エラー状態の下で所与の占有量hに対して最高効率メモリは、D-1 [1ホットコード (1-hot code) : n個の状態を表すための1ヵ所のみが1で他が0のnビット符号] としてデータが書込まれた場合及び最少数のアドレスデコーダが所与の入力アドレスに対して発火する場合である。即ち、d=1で、w=1である。しかしながら、これらの基準に従ってd及びwを設定することは、逆に、下記のように、高容量又は入力エラーの下でメモリの頑健性に影響を与える。

30

【 0 0 9 9 】

データメモリの頑健性を考慮するため、Zデータ語の書込みに続いて、エラー無しで読出しアドレスを付与する場合に、単一のデータ語の読出し(検索)が考慮される。読出しアドレスにより付勢されるwアドレスデコーダは、書込みデータ過程の間に付勢されるwアドレスデコーダと同一である。特定のビットに書込まれたデータが、「1」であれば、各wワードラインは、そのビットに「1」を記憶し、方程式(9)によって付勢レベルが与えられる。

40

【 0 1 0 0 】

【 数 2 0 】

$$E(X_1) = w \quad (9)$$

【 0 1 0 1 】

ここで、 $E(X)$ は、付勢レベルである。

【 0 1 0 2 】

しかしながら、特定のビットに書込まれたデータ値が「0」であれば、期待される付勢レベルは、：

50

【 0 1 0 3 】

【 数 2 1 】

$$E(X_0) = w \cdot h \quad (10)$$

【 0 1 0 4 】

ここで、hは、前記定義の通りである。

【 0 1 0 5 】

hは、1より小さいので、書込んだデータを回復するようにメモリを操作でき、「1」に方程式(9)及び(10)から明らかなように、設定されたビットは、「0」に設定されたものより高い付勢レベルを有する。しかしながら、「0」に設定されたビットの付勢レベルは、統計学上の変動(ばらつき)を生ずることがあり、ある(小さな)確率では、全ての加重値は、他のデータ書込みにより既に「1」に設定されている。この場合に、「0」であるべき出力の付勢レベルは、「1」であるべき出力の付勢レベルと同一である。これは、「誤り1」エラー状態の可能性を与える。書込みエラーがなければ、「誤り0」エラーの可能性は存在しない。

10

【 0 1 0 6 】

「0」出力の一因となる全てのビットが誤って「1」に設定される場合にのみ、「誤り1」エラーが発生する。このように、正確に読み出される「0」ビットの確率は、方程式(11)により付与される。

【 0 1 0 7 】

20

【 数 2 2 】

$$p(x_0 < w) = [1 - h^w] \quad (11)$$

【 0 1 0 8 】

即ち、 h^w は、誤って「1」に設定された付勢レベルの原因となるwビットの確率に対する値を付与する。このように、1からこの値を引くことは、正しく読み出される「0」値の確率、即ち、複数のwビットが誤って「1」に全て設定されなかった確率を与える。

【 0 1 0 9 】

0に設定されたD-dビットが全て正確に読み出されれば、データ語は、正確に読み出される。語を正確に読み出す確率は、方程式(12)により与えられる：

30

【 0 1 1 0 】

【 数 2 3 】

$$p(\text{data word correct}) = [1 - h^w]^{D-d} \quad (12)$$

【 0 1 1 1 】

(「data word correct」は、「正確に読み出されるデータ語」を意味する。)

【 0 1 1 2 】

即ち、D-dビットを全体で取り出すべきならば、方程式(11)の表示は、D-d乗に上昇する。

【 0 1 1 3 】

40

Zデータ語をメモリに書込めば、記憶されたあらゆるデータ語を正確に読み出すことができる確立は、方程式(13)によって付与される。

【 0 1 1 4 】

【 数 2 4 】

$$P_c = [1 - h^w]^{Z \cdot (D-d)} \quad (13)$$

【 0 1 1 5 】

hが増加するので、エラー無し回復(読出し)は、零に向かって減少する。

【 0 1 1 6 】

対数を使用する再書込み方程式(13)は、次の通りである：

50

【 0 1 1 7 】

【 数 2 5 】

$$\ln(P_c) = Z \cdot (D - d) \cdot \ln(1 - h^w) \quad (14)$$

【 0 1 1 8 】

wが1より相当に大きい場合に、 h^w は極めて小さいので、テイラー級数の第1項は、方程式(14)の右辺の対数に対する下記近似式に書き換えることができる：

【 0 1 1 9 】

【 数 2 6 】

$$\ln(P_c) = -Z \cdot (D - d) \cdot h^w \quad (15)$$

10

【 0 1 2 0 】

データメモリを最適な構造に構成するため、アドレスデコーダWの数及び語長Dを固定することが考慮される。

【 0 1 2 1 】

また、データ符号付け(D-d)を固定することが考慮される。正しいデータ回復の確率を最大化するため、 h^w 又は $w \ln h$ を最小化すべきことは方程式(13)から理解できよう。従って、下式の解を求める：

【 0 1 2 2 】

【 数 2 7 】

$$\frac{d}{dw}(w \ln h) = \ln h + \frac{w}{h} \cdot \frac{dh}{dw} = 0 \quad (16)$$

20

【 0 1 2 3 】

再生方程式(4)：

【 0 1 2 4 】

【 数 2 8 】

$$Z = -\ln(1 - h) \cdot \frac{W}{w} \cdot \frac{D}{d} \quad (4)$$

30

【 0 1 2 5 】

この式を下記のように書き換えることができる：

【 0 1 2 6 】

【 数 2 9 】

$$w \cdot \frac{Z \cdot d}{W \cdot D} = -\ln(1 - h) \quad (17)$$

【 0 1 2 7 】

wに関する微分方程式(17)及び方程式(17)から得られるZ値を代入すると、次式が得られる：

40

【 0 1 2 8 】

【 数 3 0 】

$$\frac{dh}{dw} = -\frac{(1 - h) \cdot \ln(1 - h)}{w} \quad (18)$$

【 0 1 2 9 】

方程式(18)から方程式(16)に

【 0 1 3 0 】

【数 3 1】

$$\frac{dh}{dw}$$

【0 1 3 1】

値を代入すると、hの下式が得られる。

【0 1 3 2】

【数 3 2】

$$\ln h + \frac{w}{h} \left(-\frac{(1-h) \cdot \ln(1-h)}{w} \right) = 0 \quad (19) \quad 10$$

【0 1 3 3】

簡素化すると、下式が得られる：

【0 1 3 4】

【数 3 3】

$$\ln h + \frac{1}{h} \cdot (- (1-h) \cdot \ln(1-h)) = 0 \quad (20)$$

【0 1 3 5】

方程式(20)の両辺にhを乗ずると、下式が得られる：

20

【0 1 3 6】

【数 3 4】

$$h \ln h - (1-h) \cdot \ln(1-h) = 0 \quad (21)$$

【0 1 3 7】

方程式(21)は、h=0.5の解をもつ。

【0 1 3 8】

h=0.5のとき（ジェイ・ピー・ナダル及びジー・トゥロウス共著「分散コード型メモリ ネットの情報記憶装置」ネットワーク（1990年）61～74頁参照）、相関行列メモリの最大情報内容が生ずることが予め示されてきた。前記hの最適化は、メモリが如何に多くの情報を保持できるかについてではなく、如何に多くの情報を確実に回復できるかであった。従って、最大回復可能内容は、最大内容に一致する。

30

【0 1 3 9】

方程式(4)から方程式(15)にZ値を代入すると、下式が得られる：

【0 1 4 0】

【数 3 5】

$$h^{-w} \cdot w = \frac{\ln(1-h)}{\ln(P_c)} \cdot W \cdot \frac{D}{d} \cdot (D-d) \quad (22)$$

【0 1 4 1】

40

前記のように得られた最適値h、即ちh=0.5を使用し、W、d及びDが既知であり、エラー無し回復率 P_c の許容可能な確率を使用すれば、wに対する値を演算することができる。

【0 1 4 2】

しかしながら、メモリの特性が付与されれば、完全にエラー無し回復率（これまで仮定したように）は、制限的過ぎる要求である。メモリの情報収容能力に対する別の定義は、エラーと共に他の幾つかの事項を回復することを認めながら、エラー無しで回復できる最大数のデータ事項を確立することである。

【0 1 4 3】

再現能力式(12)：

【0 1 4 4】

50

【数 3 6】

$$p(\text{data word correct}) = [1 - h^w]^{D-d} \quad (12)$$

【0 1 4 5】

エラー E_c と共に他の事項を回復しながら、高精度に回復できるデータ事項の数に対する式を誘導することは可能である。この式は、方程式(23)で与えられる。

【0 1 4 6】

【数 3 7】

$$E_c = Z \cdot [1 - h^w]^{D-d} \quad (23)$$

10

【0 1 4 7】

ここで、Zは、メモリ内の語数である。

【0 1 4 8】

所与の値Zに対して、 E_c を最大化するように、wは、変化され、 $h=0.5$ でこの最大値が生ずる。

【0 1 4 9】

各新しい入力により1つ又はそれ以上の更なる読出しエラーが生ずるまで、データをメモリに書込めば、メモリの最大有用収容能力に達した。下記の場合に、この点に到達する：

【0 1 5 0】

20

【数 3 8】

$$\frac{dE_c}{dZ} = 0 \quad (24)$$

【0 1 5 1】

再書込み方程式(23)は、下記の通りである：

【0 1 5 2】

【数 3 9】

$$\ln E_c = (D-d) \ln Z \cdot [1 - h^w] \quad (25)$$

30

【0 1 5 3】

微分方程式(25)は、下記の通りである：

【0 1 5 4】

【数 4 0】

$$\frac{1}{E_c} \cdot \frac{dE_c}{dZ} = \frac{1}{Z} - \frac{(D-d)}{(1-h^w)} \cdot \ln h \cdot h^w \cdot \frac{dw}{dZ} = 0 \quad (26)$$

【0 1 5 5】

再書込み式(4)：

【0 1 5 6】

40

【数 4 1】

$$wZ = -\ln(1-h) \cdot \frac{WD}{d} \quad (27)$$

【0 1 5 7】

一定のhに対して、wZは、一定(W, D及びdが与えられれば)であり、不変であることが方程式(27)から明らかである。従って、：

【0 1 5 8】

【数 4 2】

$$\frac{dw}{dZ} = -\frac{w}{Z} \quad (28)$$

【0 1 5 9】

方程式(28)を方程式(26)に代入すれば、下式が得られる：

【0 1 6 0】

【数 4 3】

$$\frac{(h^{-w} - 1)}{w} = -(D - d) \cdot \ln h \quad (29)$$

10

【0 1 6 1】

エラーと共に幾つかの語を検索しながら、極力多くの語を正しく読み出すことが目的のとき、方程式(29)は、最適値wに対する別の値を与える。方程式(29)から得られる値wは、方程式(22)から得られるものより小さい。これは、より多くの数（即ち、wを増加する）のアドレスデコーダを付勢することにより、メモリのエラーがより少なくなるが、収容能力もより小さいとの事実を反映するものである。逆に、w値が小さくなるほど、メモリがより多くのエラーを生ずるが、各書込みデータ操作により、データメモリのより少ないワードライン内に値を設定するので、メモリはより大きな収容能力を有する。いずれの場合でも、h=0.5のとき、回収可能な最大情報が生ずる。

20

【0 1 6 2】

前記のように、式を誘導した本発明者は、方程式(22)及び(29)により決定される複数の値の間にwが設定されれば、本発明の具現化を可能にするメモリを十分に改良できることを決定した。

【0 1 6 3】

図3及び図4について前記の解析を説明する。図3は、方程式(23)の三次元線図を示す。Z軸は、メモリに書込まれた数項目を表し、w軸は、読出し操作又は書込み操作により付勢されるアドレスデコーダの数を表し、E_c軸は、方程式(23)により与えられる量を表す。図4は、w軸及びZ軸が図3の各軸と同一の量を表す場合の図3の線図の等高線図を示す。線図に例示するシステムは、4096個のアドレスデコーダ（即ち、W=4096）と、256-11

30

【0 1 6 4】

まず、図3では、メモリに書込まれるデータが増加する（即ち、Zが増加する）ほど、正確に検索できる語数が増加するが、Zが大き過ぎるようになると、急激に低下する（絶壁形状10）。即ち、多くのデータを書込んだ後に、メモリは飽和し、正しく回復できる多くの語が急激に転落する。

【0 1 6 5】

図4について更に解析を行う。等高線は、正しい語(E_c)の期待数は、記憶された情報入力数(Z)及び付勢されたアドレスデコーダ(w)の数と共に変化する。エラー率が約11%で、E_cの最大値は、5332である。この最大値は、ワードライン（W=4096）の数より多い。中央等高線11は、E_c=5000で形成され、次の等高線は、500の段で形成される。線図の南面は、極めて平坦に近く、wが小さい線図の西側を除き、E_cは、Zに極めて近い。線図の西側及び北側での急激な各傾斜により示すように、wが小さくなるか又はZが大きくなると、性能は急落する。

40

【0 1 6 6】

前記の連続的な等高線に他の2つのデータ組が重ねられる。第1のデータ組は、長めの点線で描く3つの双曲線12を含む。各双曲線は、hに対する所与の値を生ずるw値及びZ値を示す。これらの双曲線の内、第1の双曲線Aは、h=0.2での線図を示し、第2の双曲線Bは、h=0.5を示し、第3の双曲線Cは、h=0.8を示す。最適占有率(h=0.5)を示す線は

50

、線図の最大値を通過することを理解できよう。

【 0 1 6 7 】

更に、3つの曲線13は、w値及びZ値が P_c 値にどのような影響を与えるかを示す。左から右に流れる3つの曲線は、それぞれ0.1、0.5及び0.9の P_c 値を表す。曲線13の各最大値は、線 $h=0.5$ 上に存在することは図4から理解できよう。収容能力は、確立 P_c に対して比較的影響されないことを示しながら一緒に接近している。

【 0 1 6 8 】

前記の説明から、エラーと共に回復される他のデータが発生すれば、最大の正しいデータ回復に対する最適値は、線図の最大値にあることは理解できよう。即ち、 $w=11$ である。エラー無し回復の最適値は、中央双曲線Bが曲線13の中央曲線と交差する点から読み出される。（即ち、 $h=0.5$ 線が $P_c=0.5$ 線と交差する点）この場合、 $w=20$ となり、エラーを有する回復値の2倍を超える。エラー無し回復に対してw値を決定するため、 P_c 曲線の最大値を配置する必要がある。しかしながら、この曲線は、極めて平坦であるから、最大値を検出することは非常に困難である。この最大値は、 $h=0.5$ の点で生ずることが知られているので、 $P_c=0.5$ 線と $h=0.5$ との交点を得て、最大エラー無しデータ回復に対するw値を決定することは可能である。従って、本発明により実施される展開により、前記のように設定されたパラメータ（変数）を有する如何なるメモリも、11と20との間のw値で最適に操作される。

【 0 1 6 9 】

前記の解説は、wに対する最適値を決定する新規な方法を提案し、このような決定の利点を提供するものである。しかしながら、各アドレスデコーダは独立して発火するので、各入力アドレスによってwアドレスを正確にかつ確実に付勢することが困難であり、即ち、wにある変動がある。前記の分析は、この情報に照らして修正される。

【 0 1 7 0 】

デコーダを付勢すべき場合に、閾値Tをアドレスデコーダに適用して、入力アドレスとデコーダ識別子との間に対応することが必要なビット数を決定できることは前記の通りである。付勢されたアドレスデコーダは、入力アドレスに対応する入力の少なくともTビットを有するものである。

【 0 1 7 1 】

第1の $M-N_1$ コードが、第2の $M-N_2$ コードと共通して、T「1」を有する確率は、方程式(30)により与えられる：

【 0 1 7 2 】

【数44】

$$p(T) = \frac{C_{N_1}^T \cdot C_{M=N_1}^{N_2-T}}{C_M^{N_2}} \quad (30)$$

【 0 1 7 3 】

入力アドレスと共通してTビットを有するアドレスデコーダが付勢されるので、方程式(31)を使用して、

【 0 1 7 4 】

【数45】

$\hat{w}$

【 0 1 7 5 】

で示すw平均値を誘導することができる。

【 0 1 7 6 】

【数46】

$$\hat{w} = \frac{W}{C_A^a} \cdot \sum_{k=T}^a C_i^k \cdot C_{A-i}^{a-k} \quad (31)$$

【 0 1 7 7 】

エラー無し入力でメモリを操作していれば、同一のアドレスデコーダは、所与の入力アドレスによって常に付勢される。しかしながら、異なる入力アドレスによって付勢される多数のデコーダ内に統計学上のある変動が生ずる。所与の入力アドレスに対して、付勢される何れか1つのアドレスデコーダの確率は、下式により与えられる：

【 0 1 7 8 】

【 数 4 7 】

$$pa = \frac{\hat{w}}{W} \quad (32)$$

【 0 1 7 9 】

10

Wアドレスデコーダは、方程式(31)による平均値及び方程式(33)による分散値と共に、wに対する二項分布を与える：

【 0 1 8 0 】

【 数 4 8 】

$$\sigma_a^2 = W \cdot pa \cdot (1 - pa) \quad (33)$$

【 0 1 8 1 】

特定値w'を得るwの確率は、：

【 0 1 8 2 】

【 数 4 9 】

20

$$P_{w'} = (1 - pa)^{W-w'} \cdot pa^{w'} \cdot C_W^{w'} \quad (34)$$

【 0 1 8 3 】

方程式(13)は、記憶されたデータ値の全てをエラー無しで回復できる確率の式を与える。方程式(34)が付与されれば、方程式(13)を下記の通り修正することができる：

【 0 1 8 4 】

【 数 5 0 】

$$P_c = \prod_{w'} [1 - h^{w'}]^{Z \cdot P_{w'} \cdot (D-d)} \quad (35)$$

30

【 0 1 8 5 】

即ち、方程式(13)を補正して、w値の分散値を考慮することができる。

【 0 1 8 6 】

同様に、正しく回復できるデータ要素（項目）（エラーと共に回復されるものを無視する）の期待数に対する値を付与する方程式(23)は、：

【 0 1 8 7 】

【 数 5 1 】

$$E_c = \sum_{w'} P_{w'} \cdot Z \cdot [1 - h^{w'}]^{(D-d)} \quad (36)$$

【 0 1 8 8 】

40

wの二項分散を考慮して、E_c値及びP_c値に前記補正を行うと、図4の等高線図は、図5のように修正される。等高線、双曲線及び曲線は、図4に示すのと同じの情報を示す。図5の垂直(Z)軸は、図4の各軸と同一の量を表す。w水平を調節して、w平均値を示すことができる。

【 0 1 8 9 】

この場合に、E_cの最大値は、Z=5440及びw=15では、E_c=4445でより低い。この最大値は、最早、h=0.5ではないが、h=0.575で生ずる。エラー無し性能は、wのより低い値で著しく悪化するが、線図の全体的形状は影響を受けない。しかしながら、w二項分散を与えれば、T値を設定して、前記提案したものより高い平均値をwに付与しなければならない。方程式(31)を再編成しかつa、A、i及びwの適当な値を使用して、要求されるT値を決定する

50

ことができる。

【0190】

人工ニューラルネットワーク法を使用して、本発明のシステムを実施することができるが、この実施法について説明する。図6は、適当な人工ニューラルネットワークの基本構造を示す。ニューラルネットワークは、複数のアドレスデコーダニューロン15の層に接続される複数の入力ニューロン14の層を有する。データメモリは、複数のDニューロン16を有する。各データニューロン16は、各アドレスデコーダニューロン15に接続される。各データニューロン16は、書込みデータ入力IP及び読出しデータ出力OPを有する。

【0191】

漏出積分発火スパイクニューロンに基づく実施に特に適するが、他のニューロンモデルを介して適用することもできる。積分付勢ニューロンは、複数のn入力を有し、これらの入力の加重された合計を得る。この合計は、ニューロンの付勢レベルとして知られている。付勢レベルが予め決められた閾値を超えた場合にのみ、ニューロンが発火する。下記の実施は、単一のパルスを出力するスパイクニューロンに基づくが、即ち、ニューロンは、発火してパルスを発生するか又は発火せずパルスを発生しないかの何れかである。ニューロンの出力は、デジタル0/1出力であるとみなすことができる。ニューロンの漏洩特性は、予め決められた時間の間にニューロンにより入力が増加することを意味する。この時間経過後に、付勢レベルは、零に戻る。このように、入力が波状に発火すれば、他の入力が零に漏出（低下）するので、単一の所与の波からの入力のみを付勢レベルの計算に考慮される。先の入力の効果のために、漏洩特性は、誤ってニューロンを確実に発火させないので、重要である。漏洩特性は、システム内のニューロンを活性波間で確実に不活性状態にさせる。

【0192】

A-iコードの形態でアドレスを入力ニューロン14からアドレスデコーダニューロン15に入力する。この入力は、ほぼ同時に到着するA入力ニューロンのi上の一連のiスパイクである。これらのスパイクニューロンは、下記のように、幾つかのアドレスデコーダニューロンの付勢レベルの一因となる。

【0193】

アドレスデコーダは、Wアドレスデコーダニューロン15を有する。入力ニューロン14とアドレスデコーダに関連するニューロンとの連結は固定される。同様に、アドレスデコーダニューロンが発火する閾値も固定される。例えば、図6に簡素化して図示する例では、そのアドレスデコーダニューロンが発火すべき場合に、特定のアドレスデコーダニューロン15に連結された2つの入力ニューロン14の少なくとも1つが発火しなければならない。即ち、各アドレスデコーダニューロンは閾値1を有する。

【0194】

入力ニューロン14とアドレスデコーダニューロン15との間の連結を表すマトリックスを形成することができ、この連結マトリックスは、全ての要素を含む - 即ち、全ての入力ニューロンは全てのアドレスデコーダニューロンに連結する。各アドレスデコーダニューロンは、加重値Aを有する。a加重値は、連結マトリックス内で「1」に設定され、加重値A-aは、「0」に設定される。連結マトリックスの一部を図7に示す。図7は、一つのアドレスデコーダニューロンに対する複数の連結を示す。5つの入力ライン17は、単一のアドレスデコーダニューロン（図示せず）の各入力ライン18にそれぞれ接続される。入力ライン17とアドレスデコーダ入力ライン18との間の複数の連結の各々に加重値「1」又は加重値「0」が割り当てられる。

【0195】

左側から右側に入力ライン17を読出し、上側から下側にアドレスデコーダニューロン入力を読出しながら、第1の入力ラインと第1のアドレスデコーダニューロン入力との間の連結に「0」が割り当てられる一方で、第2の入力ラインと第2のニューロン入力との間の連結に「1」が割り当てられることが理解できよう。

【0196】

5つの入力ライン17と5つのアドレスデコーダ入力18が設けられる。これらのラインの3つの間の連結に加重値「1」が割り当てられる。このように、各アドレスデコーダニューロンに、実際に、識別子としてコード5-3が割り当てられる。加重値「1」を有する複数の連結の各々に対して、関連する入力ライン上の入力ニューロン発火は、アドレスデコーダニューロンの付勢レベルの一因となる。逆に、加重値「0」を有する連結に対して、関連する入力ライン上の入力ニューロン発火は、アドレスデコーダニューロンの付勢レベルの一因とはならない。

【0197】

図7に示す構造は、Wアドレスデコーダニューロン入力に連結されるように各入力ライン17は、反復される。図7に示す例では、各入力ライン17を単一の入力18にのみ連結するように、単一のみのアドレスデコーダニューロンの複数の入力を示す。

【0198】

加重値「1」は、発火入力ニューロンがアドレスデコーダニューロン入力18（即ち、その入力ラインは、アドレスデコーダの識別子内の「1」を表す）に影響を与えることを意味し、加重値「0」は、非連結を有効に表示する（即ち、この入力ライン上のニューロン発火は、アドレスデコーダニューロンに影響を与えない）ことを意味するので、各アドレスデコーダニューロンは、「1」を保有するi入力ニューロンに連結されることのみが必要である。加重値「1」を有する3つの入力ライン17のみがアドレスデコーダニューロン15の入力18に連結される場合のこれを図8に示す。

【0199】

このように、入力とアドレスデコーダニューロンとの間に連結を、「0」値及び「1」値（図7）を有する完全な連結マトリックスとして又は特殊な連結（図8）を含む路線指定テーブルの何れかとして考えることができる。前記2つの考えは、同等である。図7に例示するように、単一のアドレスデコーダニューロンに対する複数の連結のみを図8に示し、類似の連結は、複数のWアドレスデコーダニューロンの各々に要求されよう。

【0200】

一般に、各アドレスデコーダニューロンへの連結（又は「1」）は、任意に選択される。

【0201】

スパイク（入力アドレスを表す）が到達したとき、多くのアドレスデコーダニューロンが発火する。発火ニューロンは、十分な数の加重値「1」入力線上で「1」値を受信して、アドレスデコーダニューロンの付勢レベルを閾値Tを超えて上昇させるものである。アドレスデコーダニューロン15のこの発火は、前記のWコードのwである。

【0202】

データメモリ3は、それぞれWアドレスデコーダニューロン15を有する複数のDニューロン16を有する。複数のDデータメモリニューロンの各々は、唯一の書込みデータ入力Pを有する。WアドレスデコーダニューロンとDデータメモリニューロンとの間の連結は、最初に零加重値を有するように設定される。書込みデータ入力（図示せず）からの連結は、最初に無関係の値を有する。しかしながら、書込みデータニューロンが発火するとき、アドレスデコーダニューロンを発火する関連する付随波を通じて持続するヘップの学習モードに各データメモリニューロンを適用しなければならない。即ち、データを書込むべきときに、各データメモリニューロンを付勢して、書き込みデータニューロンによって付与されるデータを記憶可能にさせなければならない。

【0203】

データメモリにデータを書込む手順を以下説明する。まず、D書込みデータニューロン（図示せず）のdが発火して、対応するdデータメモリニューロン16がヘップの学習モードに移行する。アドレスデコーダニューロン15は、前記のように、発火し、Wコードのwがデータメモリニューロン16に連絡する。アドレスデコーダからスパイクを受け取るwニューロンは、書込みデータ入力ニューロンによってヘップの学習モードに移行さ

10

20

30

40

50

れて、「1」に設定される加重値を有する。

【0204】

付随するアドレスデコーダニューロンスパイクの波が一度通過すれば、全てのニューロンの漏洩特性によって、付勢レベルが「0」に戻る。この時点で、データメモリニューロンは、ハップの学習則モードを停止する。適当な遅延時間経過後に、メモリは、始動すべき更なる読出しサイクル又は書込みサイクルの適当な状態となる。このように、ニューロンの漏洩特性によって、先の書込みサイクルからの付勢が現在の書込みサイクルの操作に確実に影響を及ぼさない。

【0205】

データメモリからデータを読み出すべきとき、書込みデータニューロンは、消勢状態にある。前記のように、Wアドレスデコーダニューロンのwを発火することにより、読出し操作を始動する。先の書込みサイクルによって少なくとも幾つかのニューロンが設定されると仮定すれば、w発火アドレスデコーダニューロンによって、幾つかのデータメモリニューロン上の付勢レベルが上昇する。

【0206】

出力論理の一般化した「勝利者が全てを取る」(k-WTA)形式は、最高の付勢レベルでdデータメモリニューロン及びそれらの各出力OP上の放出スパイクを選択して、データ出力の表示を形成する。前記のように、更なる読出しサイクル又は書込みサイクルが開始される前に、適当な遅延が要求される。多くの既存の方法を使用して、k-WTAアルゴリズムを実行することができる。これらの一方法では、まず最高の付勢レベル発火を伴い、ニューロンと共に付勢レベルを減少する順序で、複数の出力ニューロンは、発火する。このように、dニューロンが発火した後に、最高の付勢レベルを有するdニューロンは、その出力の一因となった。更に、d発火サイクルを保持すると共に、更なる発火を防止する抑制信号を発生する逆ニューロンによって、その後、更なる発火を抑制することもできる。

【0207】

前記のように、アドレスデコーダ論理にM-Nコード付けを適用すれば、多くの利点がある。特に、これらのコードは、データ値内の「1」ビット位置にのみ関連するので、発火ニューロンによって「1」を表す場合に、ニューロンネットワークへの適用に特に適する。二進法では、二進法の神経組織化の実現を困難にするので、非発火ニューロンを表示し、かつ検出することは比較的困難である。このように、イベント(ニューロンの発火)が効力を生ずる場合及び非イベントが効力を生じない場合に、M-Nコード付けの適用により有効なイベント駆動システムを提供する。

【0208】

N-Mコードに対するM値とN値の特定の選択は、ある程度まで適用内容に依存する。例えば、DNAデータを表示すべきバイオインフォーマティクス(生命科学と情報科学、情報工学が融合した学問分野)への適用では、しばしば4-1コードが選択される。DNAは、多数の塩基を有し、各塩基は、G、A、T及びCと通常表示される4つの適当な塩基の組から選択されるので、前記の選択が行われる。このように、各塩基は、コードによって付与される4つの入力の一つ上に「1」により書く塩基が表示される。即ち、例えば：

G=0001

A=0010

T=0100

C=1000

即ち、コード4-1により、4つの異なる値(2ビット二進数)を表示することができる。従って、DNA塩基配列の各塩基を一コードとして表示する一連の4-1コードとしてDNAを表示することができる。この塩基配列を使用して、16-4コードにより4つのDNA塩基配列を表示することができる。これは、コードの第1の4ビット内に「1」ビットの一つが存在し、次の4ビット内に第2の「1」ビットが一つ存在し、以下同様である。

【0209】

例えば、デュアルレールコーディングを使用して、一般的な二進コードをM-Nコードに変換することができる。ここで、二進コードの各ビットは、2つの「レール」に割り当てられる。ビットが「1」に設定されれば、パルスは、そのビットの第1のレール上で送信

10

20

30

40

50

されるが、ビットが「0」に設定されれば、そのビットの第2のレール上でパルスが送信される。図6に示すニューラルネットワークにこのデュアルレール法について、各入力ニューロン14は、2つのデュアルレール入力ニューロンからの入力を受信する。即ち、システム内に10のデュアルレール入力ニューロンが存在する。入力が受信されたとき、1つの「1」が第1のニューロン対の一方上に存在しなければならず、また1つの「1」が第2のニューロン対の一方上に存在しなければならないような制限と共に、これは、10-5コードの形態の中に存在する。「1」レールニューロンから入力を受信する入力ニューロンは、その後前記のように、発火する。

【0210】

このように、デュアルレール法の追加により、前記のように、制限されたM-Nコードを使用して、デュアルレール入力操作されるシステムが提供されるが、如何に多くのニューロンが発火するかについて確実性はなく、入力ニューロンは二進で動作する点に注意すべきである。これは、M-Nシステムと二進法との間に有効な変換を与えるものである。

【0211】

従来のアドレスデコーダ及びデータ記憶装置のワードラインによって又は前記のようにニューラルシステムとして本発明のメモリ構造を実施できることは当業者に理解できよう。また、専用ハードウェア構成部品を作成し又はフィールド・プログラマブル・ゲート・アレイ(FPGAs)等の再構成可能な部品を構成することにより本発明のシステムを実行できることも明らかである。別法として、適当なコンピュータプログラムを書込んで、本発明のシステムを実現して、従来のコンピュータシステム上で駆動することもできる。このようなシステムは、本発明を実施するメモリをシミュレートする方法で従来のコンピュータのハードウェアを使用できる。目標設定した実施が特に好ましいが、広範囲に使用される多くのコンピュータプログラム言語の何れかでコンピュータプログラムを書込むこともできる。

【図面の簡単な説明】

【0212】

【図1】従来公知のメモリ構造を示すブロック図

【図2】本発明によるメモリ構造のブロック図

【図3】図2に示す特定のメモリ構造の記憶容量を示す三次元線図

【図4】図3の三次元線図の等高線図

【図5】独立して付勢されたアドレスデコーダを考慮して補正した図4の等高線図

【図6】本発明のメモリの実行に使用できる人工ニューラルネットワークのブロック図

【図7】図6に類似する一般形態を有する人工ニューラルネットワーク内の入力ニューロンとアドレスデコーダニューロンとの間の接続の代替構造を示すブロック図

【図8】図7に示す代替構造に対する入力ニューロンとアドレスデコーダニューロンとの連結の代替構造を示すブロック図

【符号の説明】

【0213】

14・・・入力ニューロン、 15・・・アドレスデコーダニューロン、 16・・・データニューロン、 IP・・・書込みデータ入力、 OP・・・読出しデータ出力、 17・・・入力ライン、 18・・・入力ライン、

【図 1】

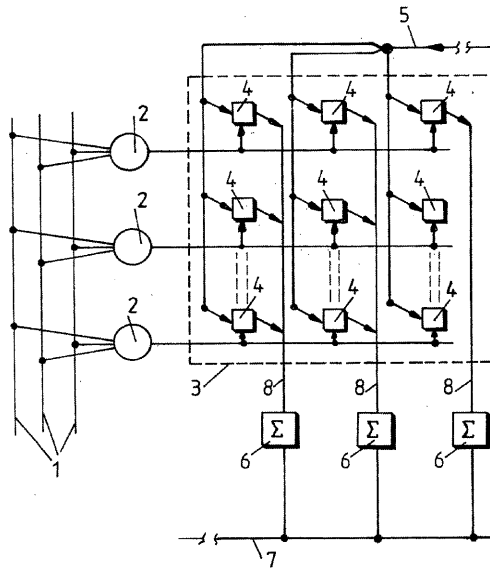
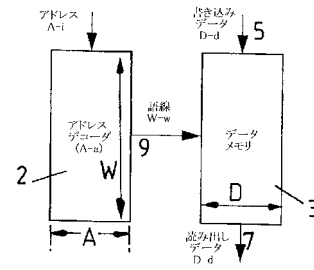


FIG. 1

【図 2】



【図 3】

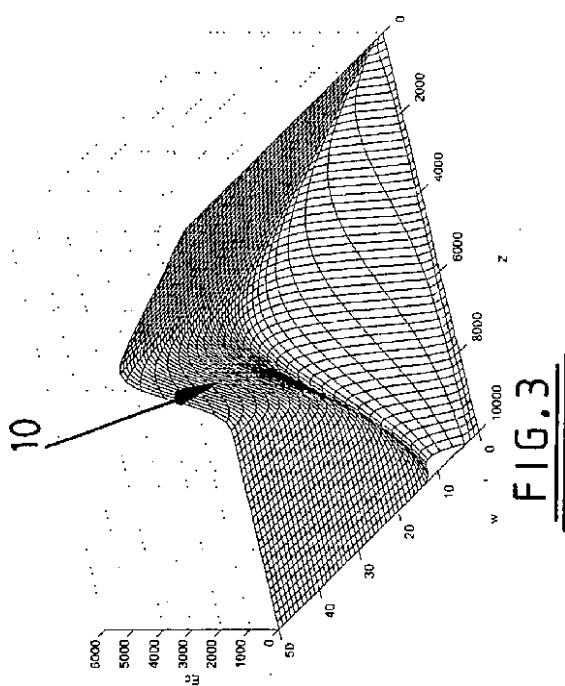
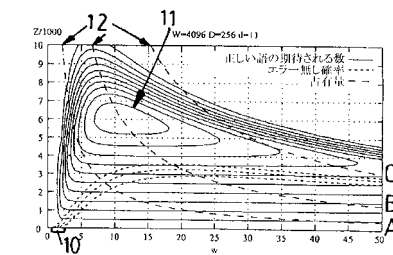
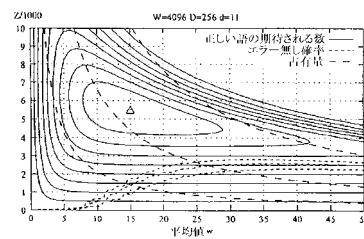


FIG. 3

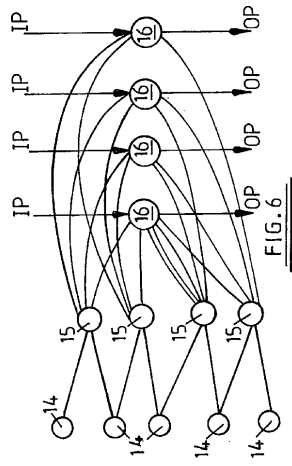
【図 4】



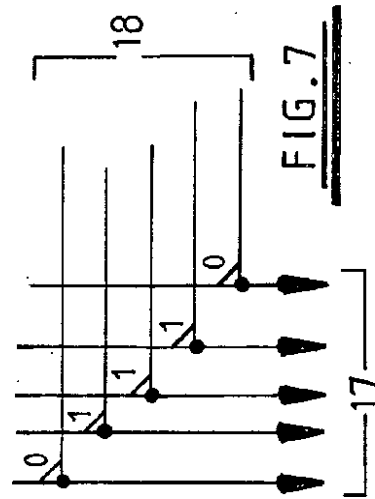
【図 5】



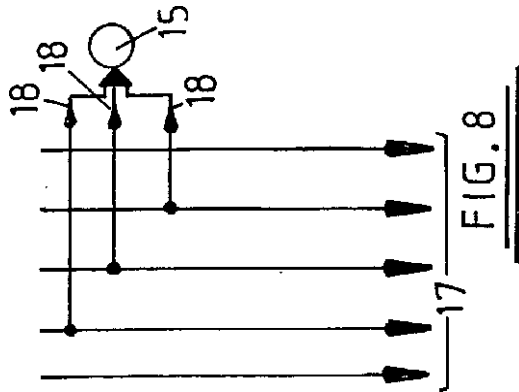
【 図 6 】

FIG. 6

【 図 7 】

FIG. 7

【 図 8 】

FIG. 8

フロントページの続き

(56)参考文献 特開平 0 5 - 2 8 2 8 7 8 (J P , A)
特開平 0 7 - 2 8 2 5 8 8 (J P , A)
特開平 0 7 - 0 0 6 5 8 9 (J P , A)

(58)調査した分野(Int.Cl. , D B 名)

G06F 12/16
G11C 15/04
G06N 3/06