

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4234628号
(P4234628)

(45) 発行日 平成21年3月4日(2009.3.4)

(24) 登録日 平成20年12月19日(2008.12.19)

(51) Int.Cl.

G06F 17/30 (2006.01)

F 1

G06F 17/30 412

請求項の数 2 (全 10 頁)

(21) 出願番号 特願2004-78315 (P2004-78315)
 (22) 出願日 平成16年3月18日 (2004.3.18)
 (65) 公開番号 特開2005-267213 (P2005-267213A)
 (43) 公開日 平成17年9月29日 (2005.9.29)
 審査請求日 平成18年6月2日 (2006.6.2)

(73) 特許権者 501380461
 島津システムソリューションズ株式会社
 京都府京都市中京区西ノ京徳大寺町1番地
 (74) 代理人 100095670
 弁理士 小林 良平
 (72) 発明者 増野 勝広
 京都市中京区西ノ京下合町26 島津システムソリューションズ株式会社内

審査官 柿崎 拓

最終頁に続く

(54) 【発明の名称】リンクテーブル生成方法及びリンクテーブルを用いた信号選択装置

(57) 【特許請求の範囲】

【請求項 1】

複数チャンネルの信号供給先の各チャンネルについて分配すべき信号の信号供給元を記述したリンク情報に基づいて、該信号供給元からそれぞれ得られる信号を選択して信号供給先に分配するに際し、前記リンク情報から、信号供給先の各チャンネルについて同一の信号供給元にリンクされている他のチャンネルの信号供給先の情報を一覧形式で記述したリンクテーブルを生成するリンクテーブル生成方法であって、

a) 予め適宜のハッシュ関数を定めておき、

b) 前記リンク情報に含まれる全ての信号供給元の情報にハッシュ関数を適用してそれぞれハッシュ値に変換し、ハッシュ値と代表的な1チャンネルの信号供給先との対応関係とを記述したハッシュテーブルを作成した後に、

c) 信号供給先の各チャンネルについて所定の順番で、

c1) 前記リンク情報に記述されている信号供給元の情報に前記ハッシュ関数を適用してハッシュ値を再計算し、

c2) 前記ハッシュテーブルを参照して、当該信号供給先が再計算で求まったハッシュ値に対応付けられた信号供給先と同一であるか否かを判定し、

c3) その判定結果に応じて、その信号供給先に対応してリンク先が同一信号供給元である他の信号供給先をリンクテーブルに格納する

ようにしたことを特徴とするリンクテーブル生成方法。

【請求項 2】

10

20

請求項 1 に記載のリンクテーブル生成方法によってリンク情報から生成されたリンクテーブルを記憶しておき、 n チャンネルの信号供給元に対して 1 組のデータセットとして供給する n 個の信号を選択する際に、前記リンクテーブルに基づいて、同一信号供給元にリンクされていて互いに異なる信号供給元に対しては同一の信号を供給することを特徴とする、リンクテーブルを用いた信号選択装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、プロセス制御 / 監視やデータ通信等を目的とした各種システムにおいて複数の信号供給元から収集した信号を選択して演算装置等の信号供給元に出力する処理を行う際に用いられるリンクテーブルの生成方法、及びリンクテーブルを用いた信号選択装置に関する。 10

【背景技術】

【0002】

例えば工場や発電設備、プラント等の制御 / 監視システムでは、コントローラ（制御装置）が多数の制御 / 監視端末からの入力信号を受け、それに基づいて所定の制御演算処理を実行し、その演算結果に基づく制御信号を制御端末へと出力する（例えば特許文献 1 など参照）。図 1 はこうした制御 / 監視システムにおける処理装置の要部のブロック構成図である。

【0003】 20

データ収集部 1 は多数の信号供給元 X_i ($i=1 \sim z$) から所定のサンプリング周期でデータを収集し、収集したデータをデータメモリ 2 に格納する。プロセス制御の場合、信号供給元はプロセスの各所に配置された様々な計測装置等である。一方、データ選択処理部 3 はデータメモリ 2 に格納されたデータの中から所定のデータを選択的に取り出して来て演算処理部 6 の入力に供給する。演算処理部 6 は n チャンネル (CH1 ~ CHn) の入力にそれぞれ 1 個ずつ与えられる n 個のデータを 1 組（これをデータセットと呼ぶ）として、このデータセットに対して所定の演算処理を実行して演算結果を得る。

【0004】

データ選択処理部 3 は、同じデータセットに含まれる n 個のデータを収集するに際して、チャンネル番号順つまり CH1 から昇順にリンク情報 4 に従ってデータメモリ 2 からデータを読み出す。リンク情報 4 は予めオペレータ等により設定される情報であり、その一例を図 2 に示す。この図 2 中の左欄はチャンネル番号 i ($i=1 \sim n$) であり、ここではチャンネル数 n は 10 である。一方、右欄はその各チャンネルに対応付けられている、つまりリンクされている信号供給元である。すなわち、このリンク情報 4 は信号供給先と信号供給元とのリンク関係を示すテーブルである。異なる信号供給先に対し同一の信号供給元が重複してリンクされることも一般的であり、この図 2 の例では、例えば第 1 チャンネル CH1 と第 5 チャンネル CH5 に対して同一の信号供給元 X_1 がリンクされている。

【0005】

上述したようにデータメモリ 2 へのデータの書き込み（図 1 で A 側）の周期とデータメモリ 2 からのデータの読み出し（図 1 で B 側）の周期とは非同期である。すなわち、データメモリ 2 からのデータの読み出しあとは全く無関係にデータメモリ 2 へデータは書き込まれる。そのため、データ選択処理部 3 が或る 1 組のデータセットに属する n 個のデータをデータメモリ 2 から順次読み出していく間に、そのデータセットに含まれる信号供給元から新たなデータがデータメモリ 2 に書き込まれることがあり得る。しかしながら、上述したように異なる信号供給先に対し同一の信号供給元が重複してリンクされている場合、1 組のデータセット内では、その異なる信号供給先に与えるデータは一致している必要がある。

【0006】

具体的に説明すると、例えば図 2 の例の場合、第 1 チャンネル CH1 に対して供給すべき信号供給元 X_1 によるデータ D1 をデータメモリ 2 から読み出した後、第 5 チャンネル CH 50

5に対して供給するべきデータをデータメモリ2から読み出す時点までの間に信号供給元X1による新しいデータD1'がデータメモリ2に書き込まれていたとしても、第5チャンネルCH5に対して供給するべきデータはD1'ではなく、先に第1チャンネルCH1に対して供給したデータD1を読み出して来て供給する必要がある。図2の例では、第2チャンネルCH2と第8チャンネルCH8、第3チャンネルCH3と第10チャンネルCH10、第4チャンネルCH4と第7チャンネルCH7も同様である。これによって同一データセットに含まれるデータの同期が維持されることになる。

【0007】

こうしたデータセット内のデータの同期を維持するようにデータの選択処理をスムーズに行うには、チャンネル番号の順に、それよりも若い（小さな）チャンネル番号であって同一の信号供給元にリンクされているチャンネル番号を記述したリンクテーブルを利用すると便利である。図3は図2に対応するリンクテーブルである。図3において、右欄の一一致CH番号で「0」は一致するものが存在しないことを意味する。このリンクテーブルに従えば、例えば第5チャンネルCH5に対して供給するべきデータを読み出そうとするときに、信号供給元が一致するのが第1チャンネルCH1であることがすぐに判るから、この第1チャンネルCH1に対して供給したものと同じデータを第5チャンネルCH5に供給することができる。

【0008】

そこで、従来の信号選択装置では、図2に示したようなリンク情報4が与えられると、自動的に図3に示したようなリンクテーブルを生成して記憶する。一旦リンクテーブルを生成すればリンク情報4の内容が変更されない限りリンクテーブルを生成し直す必要はないから、通常、リンクテーブルの生成処理は、システム起動時やリンク情報更新時にのみ行われる。図4は、リンク情報に基づいてリンクテーブルを生成するための従来の手法（アルゴリズム）を説明するためのフローチャートである。このアルゴリズムの基本は、対象チャンネルを先頭（番号の小さなものの）から順番に探索してゆき、信号供給元が同一であるチャンネルを個別に抽出するものである。

【0009】

すなわち、まずリンクテーブルTLの内容（図3では一致CH番号の欄）を全て0にクリアし（ステップS50）、チャンネル番号iを1にセット（ステップS51）、変数jを0にセットする（ステップS52）。そして、リンク情報において、チャンネル番号iにリンクされている信号供給元（以下、リンク信号という）X[i]がチャンネル番号jのリンク信号X[j]と一致しているか否かを判定する（ステップS53）。最初はj=0であってリンク信号X[0]は存在しないからステップS53では「No」と判定され、jをインクリメントして（ステップS54）jがiより小であるか否かを判定する（ステップS55）。最初にこのステップS55に来るときにはi=j=1であるから、jはiよりも小でないのでステップS57へと進み、iをインクリメントして、iがチャンネル数nより小であるか否かを判定する（ステップS58）。

【0010】

ステップS52でiがnより小であればステップS52に戻る。例えばn=10であればi=2のときにステップS52へと戻り、上記と同様にステップS52 S53 S54 S55と進むが、j=1、i=2であるのでステップS55からS53へと戻る。そして、第2チャンネルCH2のリンク信号X[2]が第1チャンネルCH1のリンク信号X[1]と一致しているか否かを判定する。図2の例の場合には、X[2]=X2、X[1]=X1であるので、X[2] X[1]であり、ステップS54 S55 S57 S58と進む。図2の例の場合、第4チャンネルCH4までは上記と同様の処理を繰り返すから、CH1～CH4に対応する一致CH番号は0のままである。

【0011】

第5チャンネルCH5（つまりi=5）のときには、j=1であるときにX[5]=X1、X[1]=X1であるので、ステップS53でX[5]=X[1]となりステップS56へと進む。このときには、リンクテーブルTLのチャンネル番号iに対応してjつまり「1」を書き込む

10

20

30

40

50

。 j は順次インクリメントされるから、仮に 3 個以上のチャンネル番号に同一の信号供給元がリンクされている場合、その最も小さな（若い）チャンネル番号が書き込まれることになる。こうした処理を繰り返し行うことにより、図 2 に示すリンク情報に対して図 3 に示すリンクテーブルを生成することができる。

【 0 0 1 2 】

上記アルゴリズムにおける処理は i について 1 から n まで繰り返され、 i が固定されている間に j は 0 から i になるまで繰り返される。或る値 i における変数 j のインクリメントは信号供給元が一致するチャンネル番号が見つかった時点で終了するから、全体の計算量は、全チャンネルが全て相異なる信号供給元にリンクされている場合に最大となる。すなわち、チャンネル数を n とすると、最大の計算量は $\{ (n + 1) \times n \} / 2$ となり、信号点数のほぼ 2 乗に比例する。そのため、チャンネル数 n が多くなるとリンクテーブル TL の生成に要する時間は膨大になる。10

【 0 0 1 3 】

監視 / 制御システム等においては、一般に、同一信号供給元を複数の信号供給先（チャンネル）にリンクさせる割合は総チャンネル数に対してかなり小さいため、上述したように計算量が最大に近くなるケースが殆どであり、リンクテーブル TL の生成に要する時間が長くなる可能性がきわめて高い。このようにリンクテーブル TL の生成に時間を要すると、例えばオペレータがリンク情報の登録内容を変更した場合に新たにリンクテーブル TL を作成し直すための待ち時間が長くなり、オペレータの作業効率を低下させるのみならず、例えば、所定の監視や制御が行えない、いわゆる制御や監視の不感時間が長くなってしまうという問題もある。20

【 0 0 1 4 】

【特許文献 1】特開平8-179816号公報

【発明の開示】

【発明が解決しようとする課題】

【 0 0 1 5 】

本発明はこのような点に鑑みて成されたものであり、その目的とするところは、与えられたリンク情報に基づいてリンクテーブルを生成する際の所要時間を従来よりも短縮化することができるリンクテーブル生成方法と、そうした方法によって生成されたリンクテーブルを用いた信号選択装置を提供することである。30

【課題を解決するための手段】

【 0 0 1 6 】

上記課題を解決するために成された本発明に係るリンクテーブル生成方法は、複数チャンネルの信号供給先の各チャンネルについて分配すべき信号の信号供給元を記述したリンク情報に基づいて、該信号供給元からそれぞれ得られる信号を選択して信号供給先に分配するに際し、前記リンク情報から、信号供給先の各チャンネルについて同一の信号供給元にリンクされている他のチャンネルの信号供給先の情報を一覧形式で記述したリンクテーブルを生成するリンクテーブル生成方法であって、

a) 予め適宜のハッシュ関数を定めておき、

b) 前記リンク情報に含まれる全ての信号供給元の情報にハッシュ関数を適用してそれぞれハッシュ値に変換し、ハッシュ値と代表的な 1 チャンネルの信号供給先との対応関係とを記述したハッシュテーブルを作成した後に、40

c) 信号供給先の各チャンネルについて所定の順番で、

c1) 前記リンク情報に記述されている信号供給元の情報に前記ハッシュ関数を適用してハッシュ値を再計算し、

c2) 前記ハッシュテーブルを参照して、当該信号供給先が再計算で求まったハッシュ値に対応付けられた信号供給先と同一であるか否かを判定し、

c3) その判定結果に応じて、その信号供給先に対応してリンク先が同一信号供給元である他の信号供給先をリンクテーブルに格納する

ようにしたことを特徴としている。50

【0017】

周知のようにハッシュ関数は文字列や数字列を固定長の疑似乱数に変換する関数であり、一般的なデータベース検索などに利用されている。本発明では、このハッシュ関数を利用した検索アルゴリズムつまりハッシュ法を、リンクテーブルを生成する際に同一信号供給元にリンクされている信号供給先が有るか否かを見い出すために利用する。ハッシュ関数としては、それ自体の演算自体に時間を要するものでなく、異なる信号列（ここでは信号供給元の情報）から同一のハッシュ値が導出されない、つまり衝突（collision）ができるだけ発生しないものを選ぶことが望ましい。

【0018】

リンク情報が与えられリンクテーブル生成処理が開始されると、まず、信号供給先を求めるための逆引き表としてのハッシュテーブルを作成する。そのために、リンク情報に含まれる全ての信号供給元の情報にハッシュ関数を適用してそれぞれハッシュ値に変換する。信号供給先と信号供給元とが1対1に対応している場合にはハッシュ値と信号供給先との対応関係は単純であるが、複数の信号供給先に同一の信号供給元がリンクされている場合には、1つのハッシュ値に複数の信号供給先が対応する。そのときには、所定の規則で、例えば最もチャンネル番号の小さなものを代表としてハッシュ値にそのチャンネルを対応付ける。

10

【0019】

その後に、今度は信号供給先の各チャンネルについて所定の順番、例えば番号の最も小さなものから昇順で以て、リンク情報に記述されている信号供給元の情報に前記ハッシュ関数を適用してハッシュ値を再計算する。そして、再計算によって得られたハッシュ値をハッシュテーブルに適用し信号供給先を逆引きし、ハッシュテーブルから得られた信号供給先がその時点で処理対象である信号供給先と同一であるか否かを判定する。ハッシュテーブルから得られた信号供給先がその時点で処理対象である信号供給先と同一でない場合、そのハッシュ値の元となっている信号供給元には少なくとも2つ以上の信号供給先がリンクされていると判断することができる。そこで、上記判定結果に応じて、その信号供給先に対応してリンク先が同一信号供給元である他の信号供給先をリンクテーブルに格納する。こうした処理を全ての信号供給先について繰り返し行うことにより、信号供給先の各チャンネルについて同一の信号供給元にリンクされている他のチャンネルの信号供給先の情報を一覧形式で記述したリンクテーブルを得ることができる。

20

【0020】

すなわち、この方法によれば、従来、或る1つのチャンネルについて信号供給元が同一であるか否かを自チャンネルよりも若番の全チャンネルに亘って順番に探索していた作業が、ハッシュテーブルで逆引きをしてそれを判定するという作業に置き換えられるので、処理手順が非常に簡略化される。もちろん、ハッシュテーブルの作成やハッシュ値の再計算に長い時間を費やすのであれば、処理手順が簡略化されても処理時間の短縮化にならないため、上述したようにハッシュ関数は計算が簡単なものである必要がある。また、本発明に係るリンクテーブル生成方法では、ハッシュ値が同一となるような衝突があるか否かを判定し、衝突がある場合には従来のようにチャンネルの番号順に信号供給元が同一であるか否かを探索していくような処理を組み込んでおくことが好ましい。もちろん、ハッシュ関数を適切に選びさえすれば実際に衝突が起きる可能性はきわめて低いので、実質的にそれによって処理時間が大幅に長くなることは回避できる。

30

【0021】

また、本発明に係る信号選択装置は、上記発明に係るリンクテーブル生成方法によってリンク情報から生成されたリンクテーブルを記憶しておき、nチャンネルの信号供給先に対して1組のデータセットとして供給するn個の信号を選択する際に、前記リンクテーブルに基づいて、同一信号供給元にリンクされていて互いに異なる信号供給先に対しては同一の信号を供給することを特徴としている。

40

【0022】

この信号選択装置によれば、信号供給元からの信号の入力のタイミングとは無関係に、

50

同一信号供給元にリンクされている複数の信号供給先に対しては信号が同一となるデータセットを供給することができる。したがって、例えば信号供給先においてこれら n 個の信号に基づいて演算処理等を行う場合でも、正確な演算結果を得ることを保証することができる。

【発明の効果】

【0023】

本発明に係るリンクテーブル生成方法によれば、リンクテーブルを生成するために処理手順が従来よりも大幅に簡略化されるので、リンクテーブルの生成に要する時間を短縮することができ、しかも特に信号供給先のチャンネル数が多い場合にその時間短縮の効果が顕著である。それによって、リンク情報の内容の変更の際にもオペレータの待ち時間が短くて済み作業を効率的に行うことができるとともに、例えばプロセス制御／監視等においては制御／監視が行えない不感時間を短くして、より良好な制御／監視を継続することができる。10

【発明を実施するための最良の形態】

【0024】

以下、本発明に係るリンクテーブル生成方法の一実施例について図面を参照して説明する。図5及び図6は本実施例におけるリンクテーブル生成のアルゴリズムを示すフローチャートである。

【0025】

本実施例のリンクテーブル生成方法では、リンクテーブルを生成する前にハッシュ関数を利用して図7に示すようなハッシュテーブルTHを作成する。ハッシュ関数はハッシュ値が0～ n より大の整数をとり、且つ異なる信号供給元の情報に対するハッシュ値がなるべく同一とならないように選ぶことが望ましい。また、あまり複雑な計算を伴うものではハッシュ値の算出に時間を要するため、リンクテーブル生成時間を短縮化したいという本来の目的に反する。そこで、ここではこうした条件を満たしつつ計算負荷を軽くするために、 $p = n$ となるような素数 p を選択し、次式をハッシュ関数とした。20

$$h(x) = x \bmod p$$

すなわち、ハッシュ値は x を p で除した剰余である。

【0026】

リンクテーブル生成処理が開始されると、まずハッシュテーブルTHの生成処理を実行する（ステップS10）。ハッシュテーブルTHの生成処理では、まず、ハッシュテーブルTHの右欄（チャンネル番号）を全て0にクリアし（ステップS30）、チャンネル番号 $i = 1$ とし（ステップS31）、リンク情報4において第1チャンネルCH1にリンクされている信号供給元のデータ情報 X_i を上記ハッシュ関数に適用して計算したハッシュ値 $h(X_i)$ をインデックス H_i とする（ステップS32）。

【0027】

次に、ハッシュテーブルTHにおいてインデックス H_i の右欄の内容 $TH[H_i]$ が0であるか否かを判定する（ステップS33）。 $TH[H_i] = 0$ である場合には、その欄には未だ内容の書き込みが為されていないことを意味するから、ステップS37に進み、 $TH[H_i]$ にそのときのチャンネル番号 i を書き込む。一方、 $TH[H_i] \neq 0$ である場合には、その欄には既に内容の書き込みが為されているから、その内容である $TH[H_i]$ が現時点でのチャンネル番号 i よりも大きいか否かを判定する（ステップS34）。

【0028】

$TH[H_i]$ が i よりも大である場合には、上記ステップS37に進み、 $TH[H_i]$ の内容をそのときのチャンネル番号 i に書き換える。すなわち、その欄に既に内容の書き込みが為されている場合であって、 $TH[H_i]$ が i よりも大きい場合には内容をより若いチャンネル番号に変更し、 $TH[H_i]$ が i と同じか i よりも小さい場合には、既に書き込まれているチャンネル番号のほうが若いので $TH[H_i]$ の内容の書き換えの必要はない。

【0029】

そして、次いでチャンネル番号 i をインクリメントし（ステップS35）、 i が n より

10

20

30

40

50

も小さければ(ステップS36で「Yes」)ステップS32へと戻る。 $i = 0 \sim n$ までS32～S37の処理を繰り返すことにより、ハッシュテーブルTHには、ハッシュ値をインデクス Hi として対応するチャンネル番号が格納され、且つ同一のハッシュ値に複数のチャンネル番号が対応する場合には、その中で最も若いチャンネル番号が格納される。対応するチャンネル番号が存在しないハッシュ値をインデクスとする欄には、当初にクリアによって格納された「0」が存在する。こうして、信号供給元の情報がハッシュ値に変換されたハッシュテーブルTHが完成する。

【0030】

こうしてハッシュテーブルTHを作成した後、リンクテーブルTLの作成処理に取り掛かる。すなわち、まずリンクテーブルTLの内容(図3では一致CH番号の欄)を全て0にクリアし(ステップS11)、チャンネル番号 i を1にセットする(ステップS12)。その後に、リンク情報4において第1チャンネルCH1にリンクされている信号供給元の情報 X_i を上記ハッシュ関数に適用して計算したハッシュ値 $h(X_i)$ を計算する。そして、このハッシュ値をインデクスとしてハッシュテーブルTHからチャンネル番号を読み出し、それを k とする(ステップS13)。この k が i と一致しているか否かを判定する(ステップS14)。

10

【0031】

上述したように、或るハッシュ値に複数のチャンネル番号が対応している場合、ハッシュテーブルTHでは、それら複数のチャンネル番号の中で最も若いチャンネル番号が格納されている。したがって、 k が i とが一致している場合には、そのチャンネル番号と同一の信号供給元を有するチャンネル番号でそのチャンネル番号より若番のチャンネルは他には存在しないことになる。そこで、 $i = k$ であればステップS22に進み、 i をインクリメントして、 i がチャンネル数 n より小であるか否かを判定する(ステップS23)。すなわち、この場合には、リンクテーブルTLにおいてチャンネル番号 i に対する右欄にはクリア時に格納された「0」がそのまま残る。 i が n より小さければステップS13に戻りインクリメントされたチャンネル番号について上記と同様の処理を続行する。

20

【0032】

通常、異なる信号供給元に対しては異なるハッシュ値が求まっているから、図2の例の場合、チャンネル番号 i が4まではステップS14で $i = k$ であると判定される処理が繰り返される。 $i = 5$ になると、それに対応して求まるハッシュ値は $i = 1$ のときと同じになるが、ハッシュテーブルTHには最も若いチャンネル番号のみが格納されているから、このハッシュ値に対しては「1」が格納されており $k = 1$ となる。そのため、 $k \neq i$ であるからステップS14からS15へと進み、信号供給元が本当に一致しているかどうかを確認するために X_i が X_k と一致しているか否かを判定する。この判定を行うのは、信号供給元が異なっていても、偶然、ハッシュ値が一致してしまう場合(つまり衝突が生じる)があり得るからである。

30

【0033】

ここで $X_i = X_k$ であれば、信号供給元が同一である、チャンネル番号のより若い他のチャンネルが存在することを意味するから、ステップS16へと進んで、リンクテーブルTLのチャンネル番号 i に対応して k を書き込む。前述のように k は同一信号供給元にリンクされた複数のチャンネルの中で最も若いチャンネル番号であり、 $i = 5$ のときには $k = 1$ であるから、リンクテーブルTLのチャンネル番号CH5の右欄に「1」を書き込む。そして、ステップS22へと進んで i をインクリメントする。

40

【0034】

もし、ステップS15で $X_i \neq X_k$ と判定された場合には、ハッシュ値は一致しているが信号供給元は異なるということを意味する。この場合には、もはやハッシュ値を利用して同一信号供給元にリンクされている他のチャンネルを探索することは不可能であるから、従来のアルゴリズム(図4)のステップS52～S56と同一の手順を踏むステップS17～S21の処理を実行する。すなわち、チャンネル番号が i よりも小さなものにリンクされている信号供給元を1つずつ調べていって、一致するものがあるか否かを判定する。

50

ステップS17～S21の処理を実行する回数が多いと計算時間が長くなるが、ハッシュ関数を適切に決めておけば、異なる信号供給元に対するハッシュ値が同一になることは殆どない。したがって、実際上、ステップS17～S21の処理を実行することは全く無いか、有ったとしてもきわめて希である。こうしたことから、計算回数を算出する上では、このステップS17～S21の実行回数は無視することができる。

【0035】

こうしてiがチャンネル数nに達するまで上記処理を繰り返し実行することによって、図3に示すようなリンクテーブルTLを生成することができる。

【0036】

本実施例のリンクテーブル生成のアルゴリズムによる計算量削減効果について具体的に説明する。いま、例えばチャンネル数nが8000である場合について考える。 10

上述した従来のリンクテーブル生成アルゴリズムの場合、リンクテーブルTLを生成する際に信号供給元が全く重複しない最悪の場合を想定すると、

$$(8000 + 1) \times 8000 / 2 = 32 \times 10^6 \text{ (回)}$$

だけ処理ループを繰り返すことになる。これに対し、本実施例のリンクテーブル生成アルゴリズムの場合、ハッシュテーブルTHを生成する際に8000回の処理ループの繰り返しを行った後に、ハッシュ値の重複さえなければ8000回の処理ループの繰り返しによってリンクテーブルTLが完成する。すなわち、合計で $16000 = 16 \times 10^3$ (回) の処理ループの繰り返しによってリンクテーブルTLが完成することになる。したがって、従来よりも格段に少ない処理ループ繰り返し回数で済む。ハッシュ値の演算には通常の判定、インクリメント等の処理に比べて時間が掛かるものの、それを考慮してもリンクテーブルTLの生成に要する時間は従来よりも大幅に短縮できる。 20

【0037】

なお、上記実施例は本発明の一例にすぎず、本発明の趣旨の範囲で適宜、変形、修正、追加を行っても本願発明に包含されることは明らかである。

【図面の簡単な説明】

【0038】

【図1】本発明によるリンクテーブル生成方法を適用する制御／監視システムにおける処理装置の要部のブロック構成図。

【図2】登録されるリンク情報の一例を示す図。

30

【図3】図2に示すリンク情報に基づいて作成されるリンクテーブルを示す図。

【図4】従来のリンクテーブル生成アルゴリズムを示すフローチャート。

【図5】本発明の一実施例によるリンクテーブル生成アルゴリズムを示すフローチャート。

【図6】図5中のハッシュテーブル生成の処理手順を示すフローチャート。

【図7】図2に示すリンク情報に基づいて作成されるハッシュテーブルの例を示す図。

【符号の説明】

【0039】

1 … データ収集部

2 … データメモリ

40

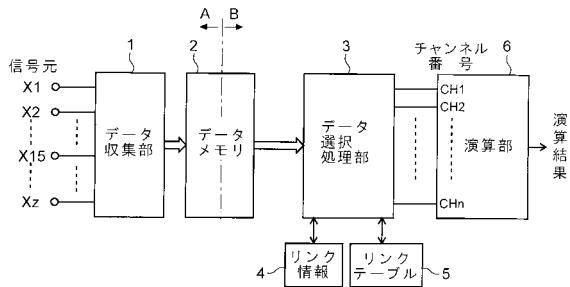
3 … データ選択処理部

4 … リンク情報

5 … リンクテーブル

6 … 演算処理部

【図1】



【図2】

信号供給先 (CH番号)	信号供給元 (X[i])
1	X1
2	X2
3	X7
4	X5
5	X1
6	X3
7	X5
8	X2
9	X15
10	X7

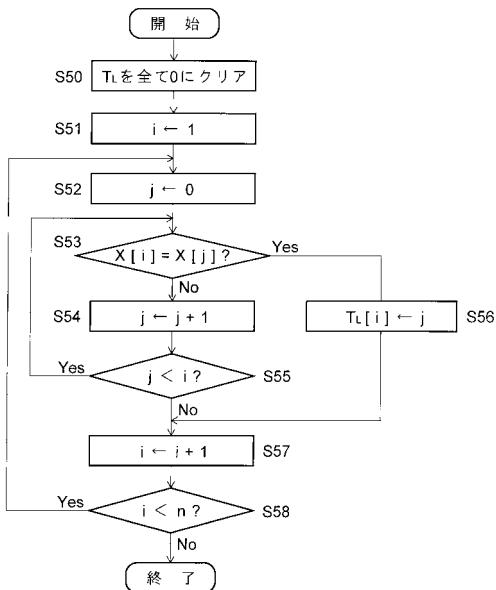
リンク信号元が一致する組み合わせ

【図3】

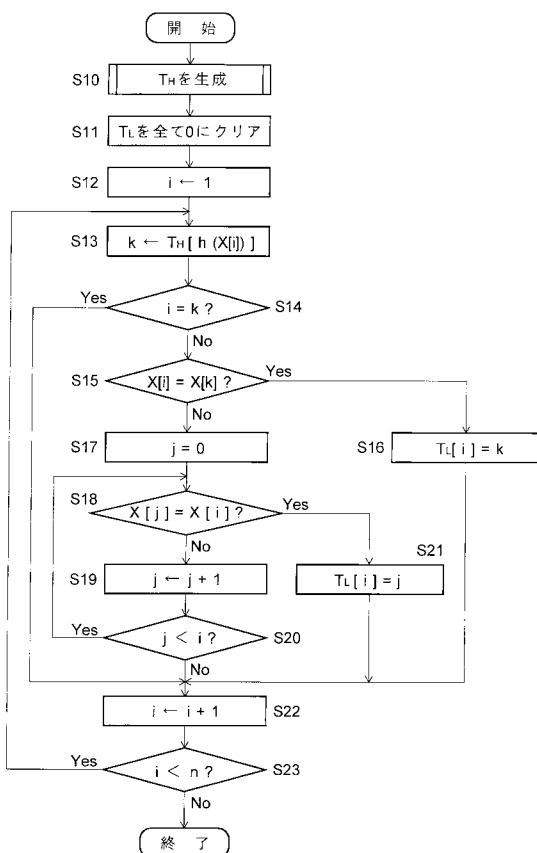
CH番号	一致CH番号
1	0
2	0
3	0
4	0
5	1
6	0
7	4
8	2
9	0
10	3

※0は一致CH番号無しを表す

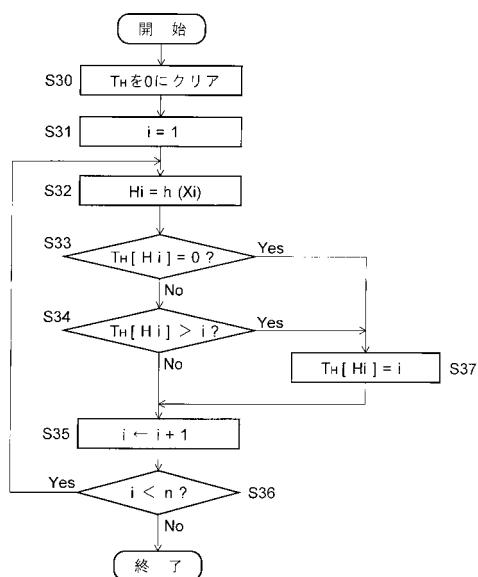
【図4】



【図5】



【図6】



【図7】

ハッシュ値	CH番号
1	0
2	0
3	5
4	0
5	7
6	0
⋮	⋮

フロントページの続き

(56)参考文献 特開平8-179816(JP,A)
特開2000-357115(JP,A)
特開2001-325298(JP,A)
特開平11-85787(JP,A)

(58)調査した分野(Int.Cl., DB名)

G 06 F 17 / 30