

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4555088号  
(P4555088)

(45) 発行日 平成22年9月29日(2010.9.29)

(24) 登録日 平成22年7月23日(2010.7.23)

(51) Int.Cl.	F I
<b>G06F 17/30 (2006.01)</b>	G06F 17/30 350D
	G06F 17/30 409

請求項の数 18 (全 20 頁)

(21) 出願番号	特願2004-569774 (P2004-569774)	(73) 特許権者	505073842
(86) (22) 出願日	平成15年8月28日(2003.8.28)		シスコ・システムズ・インコーポレーテッド
(65) 公表番号	特表2005-537592 (P2005-537592A)		ド
(43) 公表日	平成17年12月8日(2005.12.8)		アメリカ合衆国・95134・カリフォルニア州・サンノゼ・ウェストタスマンドライブ・170
(86) 国際出願番号	PCT/US2003/027518	(74) 代理人	100064621
(87) 国際公開番号	W02004/021170		弁理士 山川 政樹
(87) 国際公開日	平成16年3月11日(2004.3.11)	(74) 代理人	100098394
審査請求日	平成18年8月28日(2006.8.28)		弁理士 山川 茂樹
(31) 優先権主張番号	60/406,834	(72) 発明者	シャランパニ, ハーシュバードハン
(32) 優先日	平成14年8月28日(2002.8.28)		アメリカ合衆国・95051・カリフォルニア州・サンタクララ・ハバードアベニュー・558
(33) 優先権主張国	米国 (US)		
(31) 優先権主張番号	10/650,363		
(32) 優先日	平成15年8月27日(2003.8.27)		
(33) 優先権主張国	米国 (US)		

最終頁に続く

(54) 【発明の名称】 データ内のパターンの高速文脈サーチ及び特徴付けを実行するためのプログラム可能な規則処理装置

(57) 【特許請求の範囲】

【請求項1】

文脈サーチを実行するための規則プロセッサであって、  
コンテンツを示す複数の入力ペイロードを格納する複数のサーチ・レジスタと、  
 前記複数のサーチ・レジスタに連結され、指定された1つ又は複数のパターンを、前記サーチ・レジスタのコンテンツを検索するための検索パターンとして提供して前記コンテンツのパターンと照合させ、双方のパターンが一致する場合は一致ラインに一致信号を出力するサーチ命令の実行により前記サーチ・レジスタの前記コンテンツに対して1つ又は複数の文脈サーチを実行するサーチ実行エンジンと  
 を含み、

前記サーチ実行エンジンが、

前記複数のサーチ・レジスタに連結されており、前記複数のサーチ・レジスタのコンテンツが複製され、サーチ対象として記憶されるサーチ・アレイと、

前記サーチ・アレイに連結されたソータと

を含み、

前記サーチ・レジスタはM行×1バイトのデータを保持し、前記サーチ・アレイはM行×所定バイト数のデータの保持が可能であって前記データのサーチ結果の出力が可能で、M個の前記一致ラインを有し、

前記ソータは、前記サーチ・レジスタのサーチ範囲の開始点と終了点により指定された開始終了範囲、および前記サーチ・アレイからの前記一致ラインの信号に基づいて前記サ

ーチ・アレイのサーチ結果を処理することを特徴とする規則プロセッサ。

【請求項 2】

各サーチ命令が、前記複数のサーチ・レジスタと前記サーチ実行エンジンとに提供された請求項 1 に記載の規則プロセッサ。

【請求項 3】

前記 1 つ又は複数のサーチ命令の少なくとも 1 つが、前記複数のサーチ・レジスタ内の前記コンテンツに対してサーチされるべきパターンと、前記コンテンツに対してマスクされるべきパターンを示す 0 又はそれ以上のサーチ命令のためのパラメータとを指定する請求項 1 に記載の規則プロセッサ。

【請求項 4】

1 つのパラメータが、前記パターンの部分集合を前記サーチ・レジスタ内の前記コンテンツに対してサーチできるようにするために、マスクされるべき前記パターンの部分を指定する請求項 3 に記載の規則プロセッサ。

【請求項 5】

前記マスクされるべきパターンの前記部分が、前記パターン内の特定バイトをマスク・オフするようにマスク・ベクトルによって指定される請求項 4 に記載の規則プロセッサ。

【請求項 6】

前記サーチ実行エンジンが、前記サーチ・レジスタ内の前記コンテンツのサーチに成功したことを示す少なくとも 1 つの結果出力を生成する請求項 1 に記載の規則プロセッサ。

【請求項 7】

前記少なくとも 1 つの結果出力が、前記命令の少なくとも 1 つで指定されたパターンと、前記複数のサーチ・レジスタ内の前記コンテンツとの間で一致が生じた、前記複数のサーチ・レジスタ内の場所の指示を含む請求項 6 に記載の規則プロセッサ。

【請求項 8】

少なくとも 1 つのサーチ命令が、前記サーチを制御するために使用するパラメータを示す前記サーチの開始終了範囲又は前記サーチを制御するために前記パラメータを記憶するメモリ内へのポインタを指定する、フィールドを含む請求項 1 に記載の規則プロセッサ。

【請求項 9】

前記サーチ・レジスタ内のデータに適用されるべき 1 つ又は複数のサーチ命令を記憶するためのメモリをさらに含む請求項 1 に記載の規則プロセッサ。

【請求項 10】

前記サーチ命令が、前記サーチ・レジスタ内の前記コンテンツで任意のパターンに対するサーチを前記サーチ実行エンジンに実行させる請求項 1 に記載の規則プロセッサ。

【請求項 11】

1 つ又は複数のサーチ命令を前記サーチ実行エンジンに実行させるための命令シーケンスをさらに含む請求項 1 に記載の規則プロセッサ。

【請求項 12】

前記 M 個の一致ラインのそれぞれが、前記サーチ・アレイに記憶された一群のデータに関連付けられており、前記 1 つ又は複数のサーチ命令の 1 つによって指定された前記検索パターンが、前記サーチ・アレイに記憶された、それに関連付けられた一群のデータと一致するか否かを示す請求項 1 に記載の規則プロセッサ。

【請求項 13】

前記ソータは、前記一致ラインの 1 つ又は複数が一致するか否かを示す第 1 の出力と、前記サーチ実行エンジンが、1 つ又は複数のサーチ命令によって指定された前記検索パターンに呼応して、前記 1 つ又は複数の文脈サーチを実行した結果を示す第 2 の出力とを有する請求項 1 に記載の規則プロセッサ。

【請求項 14】

前記ソータが、  
前記パターンと、前記サーチ・アレイの最上位側と末尾側のうち一方の側に関して前記サーチ・アレイに記憶されたデータとの間の一致の第 1 の発生に対応した前記 M 個の一致

10

20

30

40

50

ラインに対応する、前記サーチ・アレイ内の場所を示すための、優先順位エンコーダをさらに含む請求項 1 に記載の規則プロセッサ。

【請求項 15】

前記ソータが、

前記パターンと、前記サーチ・アレイの最上位側に関して前記サーチ・アレイに記憶されたデータとの間の一致の第 1 の発生に対応した前記 M 個の一致ラインに対応する、前記サーチ・アレイ内の場所を示すための昇順優先順位エンコーダと、

前記サーチ・アレイの末尾側に関して前記サーチ・アレイに記憶されたデータと前記パターンとの間の一致の第 1 の発生に対応した前記 M 個の一致ラインに対応する、前記サーチ・アレイ内の場所を示すための降順優先順位エンコーダと、

前記 M 個の一致ラインの範囲における一致数を特定するためのカウンタと、

前記昇順優先順位エンコーダ、前記降順優先順位エンコーダ、前記カウンタに連結されており、第 1 の出力を有し、前記昇順優先順位エンコーダ、前記降順優先順位エンコーダ、前記カウンタのいずれか 1 つの出力を前記ソータの前記第 1 の出力として選択するよう動作可能なセレクタと

をさらに含む請求項 1 に記載の規則プロセッサ。

【請求項 16】

一組のサーチ・レジスタにコンテンツをロードするステップと、

前記サーチ・レジスタでサーチされるべきサーチ命令によって示されたパターンを提示するステップと、

前記パターンと前記サーチ・レジスタに記憶された前記コンテンツとの間でパターン照合を実行するステップと、

前記パターン照合を実行し、双方のパターンが一致する場合は一致ラインに一致信号を出力するステップと

を含み、

コードを一連のサーチ命令に変換するステップと、

パターン照合が前記複数のサーチ命令のそれぞれに対して実行されるように、前記一連のサーチ命令を連続した周期で実行するステップと

をさらに含み、

前記サーチ・レジスタは M 行 × 1 バイトのデータを保持し、複数の前記サーチ・レジスタのコンテンツが複製され、サーチ対象として記憶されるサーチ・アレイは M 行 × 所定バイト数のデータの保持が可能であって前記データのサーチ結果の出力が可能な M 個の前記一致ラインを有し、

前記サーチ・レジスタのサーチ範囲の開始点と終了点により指定された開始終了範囲、および前記サーチ・アレイからの前記一致ラインの信号に基づいて前記サーチ・アレイのサーチ結果を処理するステップを含むことを特徴とする処理。

【請求項 17】

前記サーチ・アレイの数行に関連付けられた複数の一致ラインを生成するステップであって、前記複数の一致ラインの一致ラインが、前記パターンと前記一致ラインに関連付けられた 1 行との間で一致が生じたか否かを示すステップと、

前記サーチ命令が指定する情報に応じて、少なくとも一組の前記一致ラインに対して前記 1 つ又は複数のオペレーションを実行するステップと、

前記一致ラインの 1 つ又は複数が、前記パターンと前記 1 つ又は複数のオペレーションを実行した結果と一致するか否かに関する指示を出力するステップと

をさらに含む請求項 16 に記載の処理。

【請求項 18】

1 行に対するデータが、シフトされた形式で隣接行に記憶されるように、データを記憶し、複製し、インタリーブするために、前記サーチ・レジスタをローディングするステップが実行される請求項 16 に記載の処理。

【発明の詳細な説明】

10

20

30

40

50

## 【関連出願】

## 【0001】

本願は、2002年8月28日出願の、米国仮特許出願第60/406,834号の非仮出願である。

## 【技術分野】

## 【0002】

本発明は、情報処理の分野に関し、詳細には、コンテンツの解析と処理の分野に関する。

## 【背景技術】

## 【0003】

コンピューティングと通信の重要な傾向は、コンテンツの解析と処理に富んだ環境の登場をもたらしている。これらの環境は、メッセージ、ドキュメント又はパケットなどのコンテンツのサーチ、構文解析、解析、解釈、変換などためのファクションのクラスに関して高性能でかつプログラム可能性であることを必要とする。このような豊富なコンテンツの解析と処理を重視する顕著な分野は、コンテンツ認知ネットワーク、コンテンツベースのセキュリティ・システム、調査、分散コンピューティング、無線通信、コンピュータのヒューマン・インターフェース、情報の記憶や検索のシステム、意味ウェブ上のコンテンツ・サーチ、バイオ情報科学などを含む。

## 【0004】

コンテンツ認知ネットワークの分野は、パケットとメッセージをどこに経路指定又は転送するかを決定するために、パケット又はメッセージ内部にあるコンテンツのサーチと検査を必要とする。このような検査は、ネットワーク接続のデータ転送速度である「ワイヤ・スピード」で伝送中のメッセージに対して実行される必要がある。最近のネットワークのワイヤ・レートが100Mbit/秒から40Gbit/秒までの範囲であるとするならば、コンテンツ検査ファクションが実行されなければならない速度に対しては、甚大な重圧が掛かる。

## 【0005】

コンテンツベースのセキュリティ・システムや調査・監視システムには、セキュリティ違反又は侵入の可能性があるか否かを判定するために、メッセージ又はパケットのコンテンツを解析し、一組の規則を適用することが必要である。通常、最近のネットワーク侵入検出システム(NIDS)上では、すべての潜在的なシステムの脆弱性を発見するために、多数のパターン、規則、式がワイヤ・スピードで入力ペイロードに適用される必要がある。ネットワークとコンピューティングインフラストラクチャが継続的に発展するとするならば、新たな脆弱性も引き続き発生することになる。さらに、検出から逃れるために、侵入者は益々高度な攻撃を使用する。侵入検出システムは、システム上のすべての周知の攻撃を検出することが可能であり、かつ新しい攻撃を示す異常で不審な挙動を検出するのに十分なほど知的でなければならない。これらすべての要因は、コンテンツの解析と処理に対して、プログラム可能性と非常な高性能の両方を要求することに繋がる。

## 【0006】

分散型のクラスタ化されたコンピュータの出現により、タスクは、現在、複合ジョブを完了するために相互に共同し通信する、複数のコンピュータ又はサーバに分散されている。この分散は、コンピュータ通信の急速な増加をもたらし、したがってこのようなメッセージ処理に対して高性能が要求される。汎用データ交換に対する新しい標準としてXML(拡張マークアップ言語)が登場したことにより、アプリケーションは、「アプリケーション層データ転送」としてXMLを使用して相互に通信する。メッセージとドキュメントは、現在、XMLマークアップに埋め込まれている。すべてのメッセージ処理は、まずXMLドキュメントを構文解析し、関連コンテンツを抽出し、解釈することを必要とし、次いであらゆる変換とフィルタリングを必要とする。これらのファクションは高速メッセージで実行される必要があるため、これらはコンピュータ処理にとって非常に要求が厳しくなる。

10

20

30

40

50

## 【 0 0 0 7 】

解放された通信及び無線ネットワークの成長により、無線デバイスからの情報のアクセスが増加した。クライアント・デバイスの軽量という要因を考えれば、このデバイスに配信されるデータはフィルタリングされ、そのペイロードは小さく維持されることが重要である。将来の環境は、回線インフラストラクチャからのXMLコンテンツを無線インフラストラクチャの軽量なコンテンツ（無線マークアップ言語又はWMLを使用して）に変換することになるであろう。無線ネットワークの使用が増加するに伴い、このコンテンツ変換ファンクションが、その取り扱いのための効率的なソリューションが必要となるほどに一般的になるであろう。

## 【 0 0 0 8 】

もう一つの重要な新たに発生する必要性は、スピーチのようなヒューマン・インターフェースを使用してコンピュータと通信し、対話する能力である。スピーチ処理や自然言語処理は、コンテンツ・サーチ、字句解析、コンテンツ構文解析、文法処理に極度に集約している。ボイス・ストリームが一度テキストに変換されると、スピーチ・システムは、そのスピーチを理解するために、多量の語彙や構文・意味論規則を、着信テキスト・ストリームに適用する必要がある。

## 【 0 0 0 9 】

世界中に広まったウェブの登場と成長は、甚大なコンピュータの負荷を情報検索（IR）システムに課した。情報は、高速でこのウェブに追加され続ける。この情報は、通常、ワードの網羅的語彙に対して完全に索引が付けられ、サーチ・エンジンとIRシステムのデータベースに追加される。情報が継続的に作成され、追加されるので、インデクサは「常時オン」である必要がある。効率的なリアルタイムの文脈サーチを行うために、索引付けファンクション用の高性能パターン照合・システムがある必要がある。

## 【 0 0 1 0 】

豊富なコンテンツの解析と処理を重視する別の分野は、バイオ情報科学の分野である。遺伝子解析とプロテオミクスは、遺伝子配列と構造に対する複合サーチや解析アルゴリズムのアプリケーションを必然的に伴う。ここでもまた、このような計算は、高性能のサーチ、解析、解釈の能力を必要とする。

## 【 0 0 1 1 】

したがって、登場しつつある将来のコンピュータ環境と通信環境は、コンテンツの豊富な解析と処理を重視する。このような環境は、ドキュメント、メッセージ又はパケットのコンテンツに対するサーチ、字句解析、構文解析、特徴付け、解釈、フィルタリング、変換などのファンクションの効率的かつプログラム可能なソリューションを必要とする。

## 【 0 0 1 2 】

これらの豊富なコンテンツ処理ファンクションの中心には、文脈やコンテンツベースのサーチとナビゲーション、さらに豊富な連想ルックアップを実行するためのオペレーションがある。

## 【 0 0 1 3 】

従来技術では、サーチとルックアップの処理は、通常、2つの方法のうちの1つで実行されてきた。第1に、このような処理は、コンテンツ・アドレスサブル・メモリ（CAM：content addressable memories）、コンパレータ・ハードウェア、専用論理の組み合わせなどを使用する、固定された特定用途向けIC（ASIC）ソリューションを使用して実行されてきた。例えば、サーチ規則はコンテンツ・アドレスサブル・メモリに記憶され、データは、一度に1バイト又は1語シフトさせて構造全体がストリーミングされる。あるいは、特定のコンパレータが、着信データ内の特定値を認識するために固定された位置に配置される。一致の発生は、専用論理によって目標アプリケーションの要件のように記録され、消費される。固定のASIC方式は性能を向上させることができるが、容易なプログラム可能性に欠け、したがってそのアプリケーションは大幅に制約を受ける。さらに、対象のソリューションごとに特定チップを設計し、調整することに関連する費用は法外なものである。

10

20

30

40

50

## 【 0 0 1 4 】

第2に、従来の汎用マイクロプロセッサは、豊富なサーチとルックアップ・ファンクション、さらにそれに関連するコンテンツ処理を取り扱うために使用されてきた。マイクロプロセッサは、完全にプログラム可能なデバイスであり、発展しつつある問題の必要性に対処することが可能である。これは、ソフトウェアを容易に再プログラムすることによって、新しい機能性を再展開することができる。しかし、従来のマイクロプロセッサは、豊富なコンテンツの解析と処理に提供できる性能レベルが限定されている。

## 【 0 0 1 5 】

コンテンツの解析に対する性能の制限は、マイクロプロセッサのアーキテクチャの設計及び発展に特有のものである。コンピューティング・ユニットとして始まったマイクロプロセッサは、1、2、4、8バイト・ワードに対して算術演算を実行する。その後、コンピュータ処理の分野が発展するに従い、登場しつつある分野に対処するために、より多くの機能がマイクロプロセッサに漸次追加された。その結果、汎用マイクロプロセッサは、非常に広範囲に亘るアプリケーション全体に対して機能するが、特にどれか1つに対して非常によく調整されたものではない。基本的に、コンテンツの解析の必要性に適用される場合、マイクロプロセッサのアーキテクチャには、2つの主な制限がある。つまり、(1)大きなデータ・セットに対して超並列の微粒子パターン照合と比較のオペレーションを同時に実行するファンクションに欠けること、(2)入力データに基づいた多数の高速な状態遷移と効率的な複数方向制御フロー変更を行う能力に欠けることである。

## 【 0 0 1 6 】

マイクロプロセッサの命令セットは、命令が単一の順序付けられたシーケンスで実行される必要のあるスカラー命令セットである。典型的なマイクロプロセッサの命令セットは、レジスタに記憶された単一64ビットの量と、異なるレジスタに記憶された別の64ビットの量を比較する。この比較は、位置合わせされた2つのオペランドによって実行される。この比較がパターン・サーチの目的で実行される場合、この比較は、これらオペランドの1つ又は両方を毎回バイトの可変数だけシフトさせた後で、反復的にインボークする必要がある。しばしば、このような反復されたシフトは、反復ごとにループの末尾のコードからループの最上のコードまで制御を転送する制御フロー変更により1ループで実行される。マイクロプロセッサにおける制御フロー変更は、新しいコードのシーケンスに分岐することによって達成される。最近のマイクロプロセッサは高度にパイプライン化されているので(カリフォルニア州サンタクララのIntel社製のPentium(登録商標)III及びPentiumIVプロセッサのような製品の20~30ステージ程度)、分岐のために被る性能の犠牲は著しい。マイクロプロセッサのパイプライン全体は、取られた分岐上でフラッシュされる必要がある。したがって、制御フロー変更に応じて、所望の経路から命令がパイプラインに十分に供給された状態にしておくために、そのようなプロセッサに高度な分岐予想技術が適用される必要がある。しかし、大部分の分岐予想技術は、経験則による、統計的な性能の改善だけを提供する。この結果、制御フロー変更は、その大部分が、配信可能な性能レベルに失速と非決定状態とを導入する。

## 【 0 0 1 7 】

マイクロプロセッサを最大限活用するために、多数のサーチとパターン照合アルゴリズムが発展した。Boyer-Mooreアルゴリズムは、所与のデータ・セットでのパターンの発生を発見するために、マイクロプロセッサで利用される最もよく知られた技術の1つであると広くみなされている。このアルゴリズムは、一度に1つのパターンしか処理せず、1データ・セットで複数のパターンをサーチする場合には反復的に呼び出す必要がある。サーチされるべきパターンごとに、このアルゴリズムは当該データ・セットを通して連続的に進行し、事前に特徴付けられたパターンから得た観察に基づいて選択的な比較を行う。このアルゴリズムは、所与のデータ・セット内の合計比較数を低減することにより、他のパターン照合アルゴリズムよりも優れた性能を提供する。しかし、このアルゴリズムの連続的であるという性質により、その性能は、マイクロプロセッサ・アーキテクチャの基本的な制約、すなわちスカラー命令セットと分岐に対して課せられた犠牲とにより

10

20

30

40

50

制限を受ける。

【0018】

マイクロプロセッサの上記のアーキテクチャ上の制限があるため、従来のマイクロプロセッサの効率や能力には、上記の登場しつつあるコンピューティング及び通信環境によって大きな問題が課される。これらの議論を支持するために、いくつかのデータ・ポイントを提供することができる。例えば、Snortのようなネットワーク侵入検出システム(NIDS)では、着信パケットに対して数百ストリングの署名検出を加えることが既に望ましいとされている。8バイト・パターンの署名によるこの作業負荷を、Boyer-Mooreパターン照合アルゴリズムの改善版を利用する商用マイクロプロセッサ・ベースのシステム内の3GHz Pentium IVプロセッサに対して実行することにより、10 10  
パケット・レートは50Mbps未満に制限される。同様に、そのようなプラットフォーム上のXMLドキュメントの構文解析は、10MB/s範囲に制限され、スピーチ処理は制約された文法と語彙に対してリアルタイム・ストリームに制限される。これらのデータ・ポイントは、2003又は2004の従来型マイクロプロセッサが、豊富なコンテンツの解析と処理を100Mbps範囲程度の速度で配信することができることを示す。しかし、その時間枠により、1Gbpsから10Gbpsの間のデータ転送速度は、企業ネットワークや環境では珍しいものではなくなる。明らかに、従来のマイクロプロセッサが提供できる性能と、環境が要求する性能の間には、1から2桁の大幅な不一致がある。目標速度で所望のファンクションのいくつかを実行するために複数の並列マイクロプロセッサ・システムを利用することは可能だが、これはシステムのコストを大幅に増大させる 20  
。これらの目標ファンクションに対して、より効率的なソリューションが求められていることは明らかである。

【発明の開示】

【発明が解決しようとする課題】

【0019】

コンテンツの解析と処理に対してより適しており、メッセージ、パケット又はドキュメントのコンテンツの文脈サーチ、字句解析、構文解析、解釈、変換を含む一組のファンクションに対して効率的な、プログラム可能な処理装置に対する新しいソリューションが求められている。

【課題を解決するための手段】

【0020】

文脈サーチを実行するための規則プロセッサに関して本明細書で開示される方法及び装置を説明する。一実施態様では、上記プロセッサは、複数の入力ペイロード・サーチ・レジスタとそのサーチ・レジスタに連結されたサーチ実行ハードウェアとを含む。本ハードウェアは、1つ又は複数のパターン・サーチを指定し、かつ1つ又は複数のパターンをサーチ・レジスタのコンテンツに与える1つ又は複数の命令の実行に呼応して、並列パターン照合によりサーチ・レジスタのコンテンツに対して1つ又は複数の文脈サーチを実行する。

【発明を実施するための最良の形態】

【0021】

本発明は、本発明の様々な実施形態に関して以下で提供される詳細な説明と添付の図面から、より完全に理解されよう。しかし、これら説明と図面は、本発明を特定の実施形態に限定するものとして解釈されるべきではなく、説明と理解のためのものである。

【0022】

1ドキュメント、メッセージ、その他のコンテンツの中にある任意の長いパターンの高速文脈サーチを実行するためのプログラム可能な規則処理装置を説明する。規則プロセッサは、コンテンツのサーチと解析に合わせて調整されたアーキテクチャを提供する。一実施形態では、規則プロセッサが、コンテンツ・ペイロードに対する規則の高速並列と再帰的順序付けを可能にする。これによって、規則、すなわち文法に集約した作業負荷の効率的な処理が可能になる。

10

20

30

40

50

## 【 0 0 2 3 】

アーキテクチャとしては、規則プロセッサは、一組の入力ペイロード・サーチ・レジスタを利用する。サーチ・レジスタ・ファイルは、様々なサーチ規則に提示されるべき入力データ（すなわちコンテンツ・ペイロード）を保持する。データはメモリからサーチ・レジスタにロードしても、又は規則プロセッサ内の他の資源に対してサーチ・レジスタから出し入れしてもよい。一実施形態では、サーチ・レジスタのコンテンツに対して様々なサーチを実行することができる。これらのサーチは、ペイロード・データに提示されるサーチ命令又はサーチ規則の形式で指定される。サーチ実行ハードウェアは、サーチ・レジスタに連結されている。このハードウェアは、サーチ・ファンクションの結果を計算するために必要とされるパターン照合と処理ファンクションを実行する。

10

## 【 0 0 2 4 】

図 1 は、サーチ・レジスタ 1 0 1 とサーチ実行ハードウェア 1 0 2 とを含む規則プロセッサの一実施形態のブロック図である。サーチ命令 1 0 3 は、サーチ・レジスタ 1 0 1 とサーチ実行ハードウェア 1 0 2 とに提示される。プロセッサは、規則 / 命令メモリ 1 0 4 と称される命令ストアと、一実施形態では命令シーケンサ 1 0 5 と命令ポインタ 1 0 6 とを含む命令の流れを制御するための装置とをさらに含む。

## 【 0 0 2 5 】

一実施形態では、サーチ・レジスタ 1 0 1 は、それぞれ 1 バイトずつの 2 K エントリを有するレジスタ・ファイルを含む。ここで、サーチ・レジスタ 1 0 1 内の値は、1 1 ビットのレジスタ・アドレスによってアドレス指定される。したがって、サーチされるべき 2 K B のコンテンツ・データをサーチ・レジスタ 1 0 1 にロードすることができる。

20

## 【 0 0 2 6 】

典型的なサーチは、命令又は規則をサーチ・レジスタに提示することを必然的に伴う。この規則は、1 つ又は複数の追加サーチパラメータと共に 1 つのパターンを指定する。一実施形態では、サーチ・ファンクションが多数の結果を返す。これらは、サーチ・レジスタ内のパターンとコンテンツとの間で一致が見つけられたか否かの指示と、ペイロード・サーチ・レジスタ内で一致が生じた場所を示す一致場所とを含む。

## 【 0 0 2 7 】

追加サーチ制御パラメータが、規則プロセッサによってサーチ実行ハードウェア 1 0 2 に提供される。サーチ命令は、目標サーチ・パターンを含む一組のバイトと共にマスク・ベクトルを提供する。マスク・ベクトルは、目標パターンの 1 つ又は複数のバイトに対応するビットで構成することができる。一実施形態では、サーチ・オペレーション中に無視されるべき目標パターンの特定バイトが、マスク・ベクトル内の対応するビットを 0 又は 1 の所定の論理レベルにセットすることにより選択される。したがって、サーチに使用される目標パターンのサイズを縮小することができる。さらに、規則処理命令は、サーチウィンドウを構成する開始場所と終了場所、又はサーチが制約されるサーチ・レジスタ 1 0 1 のバイト範囲を指定することができる。

30

## 【 0 0 2 8 】

サーチ命令に対する追加パラメータは、サーチが不成功だった場合に規則プロセッサが利用する分岐アドレスを含むことができる。サーチ実行ハードウェア 1 0 2 のデータ経路の幅よりも著しく長い数ストリングのバイトのサーチが、全く一致なしに失敗するか、又は最初の数バイトだけがサーチ・レジスタ 1 0 1 のコンテンツと一致した後で失敗した場合に、このファンクションは規則プロセッサの性能を強化する。規則プロセッサは、現行ストリングに対する最後のサーチ命令の後に続く命令に分岐することによって、現行ストリングに対する残りのサーチ命令を省略することができる。

40

## 【 0 0 2 9 】

サーチ命令の一例は、ウィンドウ・ファインド・ファースト・フォーワード (windowed-find-first-forward) 命令である。一実施形態では、ウィンドウ・ファインド・ファースト・フォーワードサーチにおいて、ある規則内にある、又はある規則によって指定された 8 バイト・パターン、8 ビットのマスク、サーチ・レジスタ 1 0 1 内にある 2 K B のコンテ

50



ンツ・データ（例えば、ドキュメントデータ）の開始バイトを指し示す開始場所オフセット・アドレス、さらにサーチ・レジスタ101内にある2KBのコンテンツ・データ（例えば、ドキュメントデータ）の終了バイトを指し示す終了場所オフセット・アドレスが与えられると、サーチは、このアドレスが終了場所オフセット・アドレスより前に開始するならば、マスクされたパターンと一致する指定された開始場所アドレスの後の最初のストリングのサーチ・レジスタ101内にある開始アドレス（例えば、11ビット・ベクトル）を戻す。別の例では、ウィンドウ・ファインド・ファースト・リバーズ(windowed-find-first-reverse)サーチを実行することができる。一実施形態では、このウィンドウ・ファインド・ファースト・リバーズ・サーチで、規則内の8バイト・パターン、8ビットのマスク、サーチ・レジスタ101内の2KBのコンテンツの開始バイトを指し示す開始場所オフセット・アドレスと、サーチ・レジスタ101内の2KBのコンテンツの終了バイトを指し示す終了場所アドレスとが与えられると、サーチは、このアドレスが開始場所オフセット・アドレスより後に開始するならば、マスクされたパターンと一致する指定された終了場所アドレスの前の最後のストリングの開始アドレス（例えば、11ビット・ベクトル）を戻す。

10

#### 【0030】

規則プロセッサは、ペイロード・データに適用されるべき規則又は一組の規則を含む制御ストア又は規則メモリ104も提供する。一実施形態では、メモリ104は、複数の一組の規則、複数の命令シーケンス、又はサーチ・レジスタ101で適用され検出される必要のあるパターン、規則、式又は文法を記述したコードを保持する。規則語彙は、限定はしないが、いくつかのレジスタに配信された個別及び複数の一致情報、規則プロセッサの出力ペイロードのオフセット、アドレスを生成するためのプリミティブ、さらにはサーチ結果に適用されるべき論理演算子と計算演算子と共に、完全一致又は部分一致によるグローバルサーチ又はローカル(windowed:窓付き)サーチを含めて、オペレーションの範囲を指定することができる。このような規則は、上記の様々なパラメータを指定する複数のフィールドから構成することができる。各パラメータは、規則内で直接的に指定しても、あるいは使用されるべき値を含んだレジスタ又はメモリ場所に対するポインタを使用することによって間接的に指定してもよい。直接的な指定と間接的な指定の両方が可能な実施形態では、そのようなフィールドはそれぞれに、直接的指定と間接的指定のどちらが利用されているかを示す追加のサブフィールドを含むことができる。

20

30

#### 【0031】

次の説明では、本発明の完全な理解を提供するために多数の詳細が述べられる。しかし、当業者には、本発明は、これらの具体的詳細なしに実行できることが明らかになるだろう。他の場合、本発明を分かりにくくすることを避けるために、周知の構造及びデバイスは、詳細にではなくブロック図形式で示される。

#### 【0032】

この後に続く詳細な説明の一部は、コンピュータ・メモリ内のデータ・ビットに対するオペレーションのアルゴリズムと記号表現に関して示される。これらのアルゴリズムの記述及び表現は、データ処理の当業者が、他の当業者にその作業のコンテンツを最も効果的に伝えるために使用する手段である。ここでは、また一般的には、アルゴリズムは、所望の結果をもたらす首尾一貫した一連のステップである。これらのステップは、物理量の物理的操作を要求するステップである。必須ではないが、通常は、これらの量は、記憶され、転送され、結合され、比較され、それ以外の操作を受ける電気信号又は磁気信号の形式をとる。原則として一般的な用途であるという理由から、これらの信号をビット、値、要素、記号、文字、用語、数字などと称することが場合によっては便利であると証明されている。

40

#### 【0033】

しかし、これら及び類似の用語はすべて、適切な物理量に関連付けられるべきであり、これらの量に適用された便宜的なラベルに過ぎないことを銘記されたい。次の解説から明らかになるように、特に具体的に述べられない限り、本明細書を通して、「処理」又は「

50

コンピューティング」又は「計算」又は「決定」又は「表示」などの用語を使用した解説は、コンピュータ・システムのレジスタ及びメモリ内で物理（電子）量で表現されたデータを操作し、コンピュータ・システムのメモリ又はレジスタ又は他のそのような情報記憶、伝送又は表示デバイス内で物理量として同様に表現される他のデータに変換する、コンピュータ・システム又は類似の電子コンピューティング・デバイスの動作及び処理に言及するものと理解されたい。

【0034】

本発明は、本明細書のオペレーションを実行するための装置にも関する。この装置は、要求された目的のために特別に構築しても、コンピュータに記憶されたコンピュータ・プログラムによって選択的にアクティブにされ又は再構成された汎用コンピュータを含んでもよい。このようなコンピュータ・プログラムは、限定はしないが、フロッピー（登録商標）・ディスク、光ディスク、CD-ROM、光磁気ディスクを含むいかなるタイプのディスク、読み出し専用メモリ（ROM）、ランダム・アクセス・メモリ（RAM）、EPROM、EEPROM、磁気カード又は光カード又は電子命令を記憶するのに適しており、それぞれがコンピュータ・システム・バスに連結されたいかなるタイプの媒体のような、コンピュータ可読記憶媒体に記憶することができる。

10

【0035】

本明細書で提示したアルゴリズム及び表示は、いかなる特定のコンピュータ又は他の装置にでも本質的に関連するものではない。様々な汎用システムは、本明細書の教示に関連したプログラムと共に使用することができるが、要求される方法ステップを実行するためのさらに特化した装置を構築することが便利であると証明することができる。様々なこれらのシステムに要求される構造は、以下の説明から明らかになる。さらに、本発明は、いかなる特定のプログラミング言語に関して説明されるものでもない。本明細書で説明するような本発明の教示を実施するために、様々なプログラミング言語を使用することができることが理解されよう。

20

【0036】

機械可読媒体は、情報を機械（例えば、コンピュータ）によって可読の形式で記憶又は送信するためのいかなる機構をも含む。例えば、機械可読媒体は、読み出し専用メモリ（「ROM」）、ランダム・アクセス・メモリ（「RAM」）、磁気ディスク記憶媒体、光記憶媒体、フラッシュ・メモリ・デバイス、電子信号、光信号、音響信号又は他の形式の伝搬信号（例えば、搬送波、赤外線信号、デジタル信号など）などを含む。

30

【0037】

規則エンジン・アーキテクチャ例

規則処理アーキテクチャを、コンテンツ・ペイロードに対する規則の並列かつ再帰的順序付けを考慮した、コンテンツの解析用規則プロセッサでの使用に関して説明する。このアーキテクチャは、複数の高速コンテンツ・ベースの状態遷移を行う能力に結びつけられた並列パターン照合能力を提供する。

【0038】

一実施形態では、規則プロセッサは、サーチ・レジスタに含まれるコンテンツに対する、プログラム・ストアからの実行のための規則を適用するための、命令又は規則シーケンスを含む。規則とコンテンツは、例えばパターン照合、字句解析、構文解析、解釈ファンクションのような、1つ又は複数のオペレーションに合わせて特に調整された規則語彙をサポートする実行エンジンによって消費される。

40

【0039】

一実施形態では、規則プロセッサは、ドキュメント、ストリーム、メッセージ又はパケットの任意の位置から開始される、任意の長いパターンに対する、固定されたパターン又は固定されていないパターンの優先づけられたかつ方向付けられたサーチと、窓付きの範囲限定されたサーチの一連を実行する。パターンだけでなく範囲制御とプログラム制御フロー（例えば、分岐アドレス）を、プログラム・ストア内に含まれる規則で静的に指定しても、規則内で述べられたポインタ又は索引を使用してレジスタ・ファイルから間接的に

50

動的に選択してもよい。

【 0 0 4 0 】

規則プロセッサは、一部には、サーチ・レジスタに連結された、特化したパターン照合・ハードウェア構造を使用してサーチを実行する。一実施形態では、規則プロセッサは、豊富なサーチ、ソート、優先順位付けファンクションをサポートする。一実施形態では、規則処理ハードウェアは、サーチ・アレイと、サーチ・レジスタに含まれる 2 K b のコンテンツ・データ（例えば、ドキュメント・データ）上で直接的に動作するソータ・ブロックとを有する 4 ステージのパイプラインとして構成される。4 ステージは、（ 1 ）規則メモリからの規則フェッチと規則復号、（ 2 ）間接フィールドの規則アセンブリ、（ 3 ）サーチ・レジスタ内の値に対するサーチ実行オペレーション、（ 4 ）結果の配信の前の、サーチ・オペレーションの結果に対するソート・オペレーションである。

10

【 0 0 4 1 】

規則プロセッサは、サーチの 1 つ又は複数の規則をサーチ・レジスタ構造に提示する。一実施形態では、サーチ・レジスタは、各エントリが 1 バイト幅である、2 K B のレジスタ・ファイルである。サーチされるべきデータがサーチ・レジスタ・ファイルにロードされる。各規則は、記憶されたデータの中にパターンがあるか否かを判定するために、サーチ・レジスタ・ファイルに提示されたパターンを指定する。パターンをさらに構成し、かつ/又はサーチで使用中のパターンのサイズを縮小するために、マスクを提供することもできる。

【 0 0 4 2 】

図 1 は、規則プロセッサの一実施形態のブロック図である。図 1 を参照すると、サーチ・レジスタ 1 0 1 とサーチ実行ハードウェア 1 0 2 は、サーチ命令 1 0 3 と共に集合的に示されている。サーチ命令 1 0 3 は、図 2 A でさらに説明される。図 2 A を参照すると、サーチ命令 2 0 1 は、サーチ・オペレーションのタイプを記述するオペコード 2 0 1 a と、サーチ・パターン 2 0 1 b と、現行サーチ命令に関連したパターン内のバイトを指定するマスク 2 0 1 c と、現行サーチ命令に関連したサーチ・レジスタ内の場所の開始と終了の限界をそれぞれに指定する 2 つのオフセット 2 0 1 d、2 0 1 e とを含む。サーチ実行ユニット 2 0 2 は、一実施形態では、サーチ・オペレーションの成功の指示から構成されて、限定はしないが、サーチ・オペレーションのサーチ命令を満たすサーチ・レジスタ内の場所を示す索引のような、1 つ又は複数のパラメータをさらに含む結果 2 0 3 を出力する。

20

30

【 0 0 4 3 】

サーチ実行ハードウェア 2 0 2 は、図 2 A に示すように、サーチ・レジスタ 2 0 2 a とソータ 2 0 2 b とを含む。サーチ・レジスタ 2 0 2 a は、サーチを受けることのできるドキュメント、メッセージ、パケット又はいかなる他の周知のデータ・ソースからのコンテンツであってもよいサーチ可能なデータを記憶する。サーチ・レジスタ 2 0 2 a のサイズは、任意の M バイトであってもよく、一実施形態では、それぞれに N バイトの M 行の、サーチ・アレイ 2 0 2 c と呼ばれる、より大きなアレイに構成される。サーチ・レジスタ 2 0 2 a からのデータがサーチ・アレイ内に複製して記憶される。サーチ・レジスタのこの実施形態は、N 個の隣接バイト・パターンに対するサーチに関して、より高い性能を含むので、複数の理由から好ましい。

40

【 0 0 4 4 】

一実施形態では、サーチされるべきデータは、アドレス・デコーダ 2 1 2 によって復号される、アドレス・ジェネレータ 2 1 1 から生成されたアドレスに基づいて、データ・ジェネレータ 2 1 3 を使用することによりサーチ・レジスタ 2 0 2 a に記憶される。記憶の処理は、サーチ・レジスタ 2 0 2 a にあるサーチ可能データのサイズを記録することも必然的に伴う。サーチ・レジスタ 2 0 2 a の容量よりもサイズの小さいデータ・セットの場合、サーチ・レジスタ 2 0 2 a は、サーチ・オペレーションを適切なデータに制約する機構を提供する。一実施形態では、データ・ジェネレータ 2 1 3 は、サーチ・アレイ 2 0 2 c の残りの場所にあるサーチ・オペレーションに対して無視されるデータになることが演

50

繹的に確立されるパターンを記憶することができ、又は一代替形態では、サーチ・レジスタ 202 a は、サーチ・アレイ 202 c の適切な場所をサーチ・オペレーションへの参加から無効にする。

【0045】

指定されたパターンを見つける能力に加えて、サーチ・レジスタ 202 a は、文字データに対する大文字小文字区別なしサーチのような、特別なサーチを実行する能力も提供することができる。このような特別なサーチ及び他の特別なサーチをサポートするために、サーチ・レジスタ 202 a は、サーチ可能データの各バイトに関連付けられた追加情報を記憶することができる。一実施形態では、サーチ・レジスタ 202 a は、大文字小文字区別なしサーチ又は定義済みクラスの文字に属する文字に対するサーチを可能にするサーチ可能データそれぞれに関連付けられた特別なビット・ベクトルを記憶することができる。

10

【0046】

サーチ可能データがサーチ・アレイ 202 c を通して構成される方式とは関係なく、サーチ・アレイ 202 c はパターン 201 b とマスク 201 c を受け取る。パターン 201 b がサーチ・アレイ 202 c のエントリと比較される。一実施形態では、サーチ・アレイ 202 c は N バイトの M 行を有する。ここで、N は、パターン 201 b 内と同じバイト数である。マスク 201 c は、サーチ中のパターンの一部ではないパターン 201 b のそれらのバイトの指示を提供する。すなわち、パターン 201 b が N バイト未満のパターンである場合、マスク 201 c は、サーチ・アレイ 202 c がパターン 201 b のどのバイトを無視すべきかを指定する。一実施形態では、サーチ・アレイ 202 c は、サーチ・アレイ 202 c の M 行のそれぞれに対する、サーチ中パターンがその特定の行に記憶されたコンテンツ・データと一致したか否かを示す出力線を有する。一実施形態では、出力が 1 の場合、パターンは、特定行内のコンテンツ・データと一致している。サーチ・アレイ 202 c の M 個の出力ラインは、ソータ 202 b の入力に連結される。

20

【0047】

ソータ 202 b は、サーチされるべきサーチ・レジスタ 202 a の範囲の開始点と終了点をそれぞれ示すオフセット 201 d、201 e を受け取るように連結されている。一実施形態では、これらのオフセットは  $\log_2 M$  ビット数である。サーチ・アレイ 202 c からの一致指示ラインと、オフセット 201 d、201 e により指定された開始終了範囲に基づいて、ソータ 202 b はサーチ・アレイ 202 c の結果を処理する。このような処理は、1 つ又は複数のオペレーションの処理を含む。これらのオペレーションは、オペレーションのタイプに従い特定の一致索引を出力する索引解決ファンクション (index resolution function) であってよい。一実施形態では、これらのオペレーションは、Find\_First\_Forward、Find\_First\_Reverse、Find\_Population\_Count を含む。これらのオペレーションは、サーチ命令 201 のオペコード 201 a により指定される。ソータ 202 b は、サーチ・アレイ 202 c からの一致指示ラインと連動して後続オペレーションで使用することのできる、前のオペレーションの中間又は最終結果を記憶することができる。このようにして、前のオペレーションの結果を利用する一連のオペレーションを発行することにより、サーチ可能データ・セットを通して漸次ナビゲートするために、ソータ 202 b を使用することができる。さらに、ソータ 202 b は、前のオペレーションの結果を、任意の数の他のオペレーションが実行された後で実行することのできる後続オペレーションで使用するように記憶しておくために、レジスタ・ファイルに連結することもできる。ソータ 202 b の結果は、規則プログラム・オフセット (例えば、分岐アドレス) を生成するため、又はその生成を支援するために、図 1 の命令シーケンサ 105 のような、規則プロセッサ命令シーケンサにも連結することができる。

30

40

【0048】

処理後、ソータ 202 b は、一致があるか否かを示す出力と、その一致に関連付けられた索引とを生成する。索引は、サーチ・レジスタ 202 a の最上位に関して最初の一致が生じる場所又は最後の一致が生じる場所である、サーチ・レジスタ 202 a の場所 (アド

50

レス)を示すことができる。あるいは、索引は、オフセットによって示される範囲内で生じた一致数を示すことができる。

【0049】

オフセットによって指定された範囲は動的に変更できることに留意されたい。例えば、オフセット201dとオフセット201eによって指定される範囲にサーチ・アレイ202cのすべての行を含めておいて、最初のサーチ命令を最初にサーチ・アレイ202cに適用することができる。しかし、前のサーチ命令によって指定された範囲内にある一致ラインを含む場所からサーチが始まるように、最初のサーチ命令と一致が識別された後で、開始と終了の範囲を後続のサーチ命令で変更することができる。この能力は、汎用レジスタ・ファイルの値を参照するために規則又は命令のフィールドを可能にした、間接化技法のファンクションを使用して達成することができる。

10

【0050】

図2Aでは、一実施形態で、サーチ・アレイ202cはそれぞれ8バイトの2K行から構成される。したがって、サーチ・レジスタ202aは2Kバイトのデータを保持する。サーチ・アレイ202cは複製されたデータを保持する。サーチ・レジスタ202aの区別されたバイトで開始する各8バイトのストリングは、サーチ・アレイ202cのバイトの異なる行として記憶される。これらのストリングは、サーチ・レジスタ202aからの区別されたバイトと、さらにサーチ・レジスタ202aのそのバイトに続く7個の連続したバイトとから構成される。したがって、サーチ・アレイ202cの各行は、前行の7個の最上位バイトと、さらにそれら7バイトの右に付加された1つのすぐ上のバイトとを保持する。

20

【0051】

一実施形態では、適切な8バイトのデータをソース・データから各行へ供給するデータ・ジェネレータ213によってサーチ・アレイ202cにデータがロードされる。

【0052】

一実施形態では、8バイトのサーチ・パターンがそれぞれのサーチ命令に提示される。このサーチ・パターンは、サーチ・アレイ202cの8列のそれぞれが独自のバイトで提示されるように、サーチ・アレイ202cと位置合わせされる。これを図2Bに示す。図2Bを参照すると、状態バイト1から8が、行1から2Kのそれぞれに対してサーチ・アレイに記憶されている。図2Bに示す2Kの行と8列のサーチ・アレイの1要素として記憶されている各バイトについて、1個の信号ライン310が存在する。例えば、行1のバイト1は信号ライン310<sub>11</sub>を生成し、行1のバイト2は信号ライン310<sub>12</sub>を生成し、行2のバイト1は信号ライン310<sub>21</sub>を生成し、等々である。記憶されているバイトが、その要素が属する同じ列に提示されるサーチ・パターンのバイトと一致する場合、各バイトに対する信号はサーチ・オペレーション中にアサートされる。この実施形態では、各行が8バイトを含むので、8個の信号ライン、例えば310<sub>11</sub>から310<sub>18</sub>が、当該行内のバイト・レベルの一致のそれぞれを示すために使用される。各行に対するバイト・レベルの一致は、行とマスク・リゾルバー・ブロック311のマスク102cのマスクと論理積演算される。論理積演算関数の結果は、各行で一致が生じたか否かの指示である。この実施形態では、サーチ・アレイは2K行を含むので、2Kの一致ラインがソータに出力される。このアレイ内の2つのバイト要素312の回路構成を図4に示す。この回路は、1クロック期間中にそのすべての行内で同時サーチを実行することにより、完全な並列サーチ・オペレーションを提供する方法で動作する。1回のサーチ・オペレーションが実行される際、1列内のすべてのバイト・レベル一致ライン401は、それぞれの記憶されたバイトとの一致を同時に示す。すべての列に対して同時に1つのサーチ・オペレーションがインボークされ、各行のマスクされた短縮ブロックが1つの行レベル一致を示すことを可能にする。したがって、サーチ・アレイのこの実施形態では、単一クロック中に、8個の隣接バイトからなるサーチ・レジスタ内のすべての2Kストリングの並列サーチが実行され、その結果が2K一致ラインに対して示される。

30

40

【0053】

50

図3Aは、ソータの一実施形態のブロック図である。図3Aを参照すると、サーチ・アレイからの一致ライン310は、範囲マスク及び選択ユニット301に接続され、これに入力される。一実施形態では、一致ライン310は一致1から一致2048までを含む。範囲マスク及び選択ユニット301は、さらなる処理を実行するために、サーチ・アレイからのM個の一致ラインの行の範囲を指定する一对のオフセットを受け取る。一実施形態では、オフセットは、出力を提供するために一致ラインと論理積演算することのできる、2Kマスク・ビットに変換される11ビットの数である。このような例を図3Bに示すが、同図では、開始範囲に対するオフセットは、ビットの剰余を1ビットとして1つ又は複数の0に変換され、範囲の終了に対するオフセットは、すべてのビットがそれ以降1になる特定ポイントまで末尾から通してすべて0に変換される。これらのレジスタを一致ラインと論理積演算することにより、当該範囲外の他の一致ラインはマスクされた状態で（例えば、所定の論理レベルに変更されて）、指定の開始と終了範囲内で生じた一致は変更されずに出力される。

10

## 【0054】

範囲マスク及び選択ユニット310の出力は、索引解決ファンクション・ユニット302の入力と連結される。一実施形態では、索引解決ファンクション・ユニット302は、範囲マスク及び選択ユニット301の出力に対して実行される1つ又は複数の関数を含む。例えば、図示するように、ソータは、非マスク一致ラインによって示されるように、指定されたNバイト・パターンとサーチ・アレイのコンテンツ・データの間の、（サーチ・アレイの最上位に関する）最初の一致の発生を見つけるために、1つの昇順優先順位エンコーダ302Aを含む。非マスク一致ラインによって示されるように、Nバイト・パターンとサーチ・アレイのコンテンツ・データの間の、（サーチ・アレイの最上位に関する）最後の一致の発生を見つけるために、降順優先順位エンコーダ302Bを含むこともできる。ポピュレーション・カウンター302Cは、非マスク一致ラインによって示されるように、Nバイト・パターンとサーチ・アレイのデータの間で生じる一致数を示す。他の索引セクタを使用してもよい。

20

## 【0055】

索引解決ファンクション・ユニット302の出力は、オペコード102aを受け取るために同様に連結された索引結合及び選択ユニット303に入力される。オペコード102aは、サーチ命令内で指定され、索引解決ファンクション出力の1つをソータの出力として選択する。索引結合及び選択ユニット303は、索引322と共に一致があることを示す一致インジケータ321を生成する。これは、出力昇順優先順位エンコーダ302Aが選択された場合は最初の一致の発生であり、降順優先順位インジケータ302Bの出力が選択された場合は最後の一致の発生である、データのサーチ・アレイ内の場所を示し、ポップ・カウンター302Cが選択された場合は非マスク一致ライン302Bの一致数を示すなどする。これらの出力の演算、すなわち一致インジケータ321と索引322に引き続き、一致インジケータ321又は他の類似の技術に応じて、命令メモリ（例えば、命令メモリ104）の指定アドレスに分岐して、汎用レジスタ内の出力をソーティングし、後続命令の間接指定を利用することにより、これらを後続の1つ又は複数のサーチ命令の実行を制御するために利用することができる。

30

40

## 【0056】

図5は、サーチ装置を含む規則プロセッサのマイクロ・アーキテクチャを示す。図5を参照すると、サーチ命令が命令メモリ501に記憶されている。命令は、命令フェッチ・ポインタ・レジスタ602を使用してフロー制御により選択される。命令はデコーダ503によって復号される。各命令の個々の部分集合は、当該命令から取られるか、又は汎用レジスタ・ファイル504からフェッチされる。次いで、各命令の様々な部分集合が、別々のユニット、すなわち上記のように、サーチ・アレイ505、ソート・ユニット506aとそれに続く特徴付けユニット506bとからなるソータ506、従来型演算論理回路（ALU）507に供給される。一実施形態では、各命令の処理は、上記のように、（i）命令フェッチ・ステージ508、（ii）命令アセンブリ・ステージ509、（iii）サ

50

ーチ/実行ステージ510、及び(iv)結果のソート及び配信及び/又は分岐ステージ511からなる4ステージのパイプラインの後に続く。

【0057】

一実施形態では、規則エンジン命令フォーマットは、128ビットの規則フォーマットを含む。128ビットの規則は、規則プロセッサに対して様々なハードウェア・エンジンに指令を発行する様々なフィールドを含んだ部分集合に分割される。一実施形態では、サーチ部分集合は、サーチ/ソート・オペコード・フィールド(5ビット)、パターン・フィールド(一実施形態では、これは、8バイト値又は8バイト値を提供する場所へのポインタを、当該命令内の8バイトがポインタであるか否かを指定する追加ビットと共に含んだ、65ビットである)、バイト・レベル・マスク・フィールド(一実施形態では、8ビット)10、開始場所アドレス・フィールド(一実施形態では、このフィールドは、11ビット値又は11ビット値を提供するレジスタへのポインタを含んでおり、また開始場所アドレスがその11ビット値又はそのようなポインタを含んでいるか否かを示すための追加ビットを含めて、12ビットである)、終了場所アドレス・フィールド(一実施形態では、このフィールドは、11ビット値又は11ビット値を提供するレジスタへのポインタを、終了場所アドレス情報がポインタであるか否かを指定するための追加ビットと共に含んだ12ビットである)、サーチ・オペレーションの結果が戻されるべき場所を指定する結果レジスタ・フィールド(一実施形態では、このフィールドは6ビットである)、さらに分岐アドレス・フィールド(一実施形態では、このフィールドは、19ビット値又は19ビット値を提供するレジスタへのポインタを、分岐アドレス情報がポインタであるか否かを指定するための追加ビットと共に含んだ、20ビットである)を含む。20

【0058】

図6は、上記規則プロセッサの一実施形態が処理することのできる、一組の規則の擬似コード601の一例を示す。この規則は、複数のパターンを有するが、これらは、このようなパターンがメッセージ又はドキュメント又はパケット内に存在することのできる場所についていくつかの特定の制約を伴う。この制約は、BEFORE及びANDのようなキーワードの使用により擬似コードで表現される。簡約化のために、この例では、ストリング間に追加デリミタを入れずに601のパターンが使用されるが、これは実際にもこの通りであってよい。また図6は、例示規則プロセッサに対応するマイクロコードの一覧602である。命令のフォーマットは上記の通りである。説明のために第1の命令603を使用するが、これは、サーチ・レジスタに保持されるサーチ可能ペイロードの開始と終了を示すために(Indirection Flagsを使用することにより)定数値として表現される開始及び終了オフセットと共に、(0xFFをMaskとして使用することにより)8バイトのPattern“cp/bin”が当該サーチに該当するFING\_\_FIRST\_\_FORWARDオペコードからなる。簡略化のために、本明細書では定数の誘導は省略した。このオペコードの結果は汎用レジスタAにロードされるべきものとして示し、最後に分岐アドレスが、図6に示す切断されたマイクロコードに続く命令である、定数値11として指定される。この命令は、サーチ実行ハードウェアに、サーチ・レジスタ202aのバイト場所0x03D及び0x800内の“cp/bin/”の発生をサーチさせる。サーチ・アレイ202c内の一致するすべての行は、マイクロ・アーキテクチャの20パイプラインのサーチ実行ステージ510の終了により、それぞれの一致ライン310をアサートする。ソート及び分岐ステージ511では、ソータ202bは、図3Bに示すように、0x03D及び0x800をビット・ベクトルに変換する。0x03Dから0x800までの場所枠外で開始するいかなる一致をも無効にするように、範囲マスク及び選択関数301を実行するために、ビット・ベクトルが使用される。現在は場所枠内にある残りの一致の中で、この命令のオペコードは、最低数の一致を11ビットのバイナリ符号化された場所に変換するために、索引解決ファンクション302から昇順優先順位エンコーダ302aを選択する。このような一致が見つかり、一致321がアサートされ、索引322が11ビットの場所を保持する。一致が見つからないために一致321がアサートされない場合、命令シーケンサ105は、命令ポインタ106に分岐アドレス0xBを50

ロードする。索引322は、レジスタ・ファイル504の制御回路により、汎用レジスタAにロードされる。汎用レジスタAへのロードと、命令ポインタのロードは、適切ならば、ソート及び分岐実行ステージ511の終了により完了する。第2の命令604、FIND\_FORWARD\_ANCHOREDは、規則エンジン例の豊富な語彙をさらに示す。これは、成功すべきサーチのためのstart\_offsetから照合を始める必要があるという点で、FIND\_FIRST\_FORWARDの異型である。

【0059】

図7は、図5に示す規則プロセッサ例のマイクロ・アーキテクチャ内での、図6に示すマイクロコードの実行を示す。表701は、複数のクロック周期を通じた実行を示す。簡約化のために、すべてのサーチ命令が、サーチ・レジスタ内で指定パターンを見つけることに成功するものと仮定する。実行は、図5に示す4ステージを通してパイプライン形式で進行する。間接指定の使用により、サーチ命令の実行は、直前の命令で計算されたオフセットを使用することができる。したがって、命令1から8は、連続する周期で実行される。命令8は、それぞれにクロック周期8とクロック周期9で計算された汎用レジスタAと汎用レジスタBのコンテンツの比較の結果に応じた、分岐である。この分岐は、クロック周期11と、クロック周期14で完了した命令実行とで取られる。したがって、擬似コード601を使用して記述された複合パターン照合式は、規則プロセッサ例の豊富な命令語彙を使用して14クロック周期でのみ実行される。この例は、ドキュメント、メッセージ又はパケットの動的及び文脈サーチ及び解析を含むファンクションの実行に対する規則プロセッサ例の能力と効率を説明する。

【0060】

上記の説明を読めば、当業者には本発明の多くの変形形態及び修正形態が確実に明らかになるだろうが、説明により図示し説明したいかなる特定の実施形態でも、限定を考慮することを意図するものでないことを理解されたい。したがって、様々な実施形態の詳細の参照は、本発明の本質とみなされる特徴のみをそこに記載した特許請求の範囲を限定することを意図するものではない。

【図面の簡単な説明】

【0061】

【図1】サーチ装置を有する規則プロセッサの一実施形態のブロック図である。

【図2A】サーチ・レジスタ及びサーチ実行ハードウェアの一実施形態のブロック図である。

【図2B】サーチ・アレイの一実施形態のブロック図である。

【図3A】ソータの一実施形態のブロック図である。

【図3B】ソータ内の範囲選択機構の一実施形態のブロック図である。

【図4】サーチ・アレイの一実施形態の回路図である。

【図5】4つの処理ステージを含む規則プロセッサのマイクロ・アーキテクチャ例を示す図である。

【図6】規則プロセッサ例に対する、複合パターン照合一組の規則の擬似コード例及び対応するマイクロコードを示す図である。

【図7】図6に示すマイクロコードの、クロックごとにパイプライン化された実行を示す図である。

10

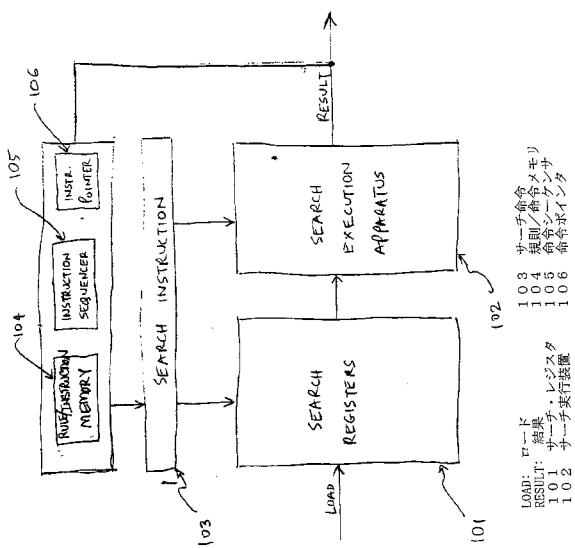
20

30

40



【図1】



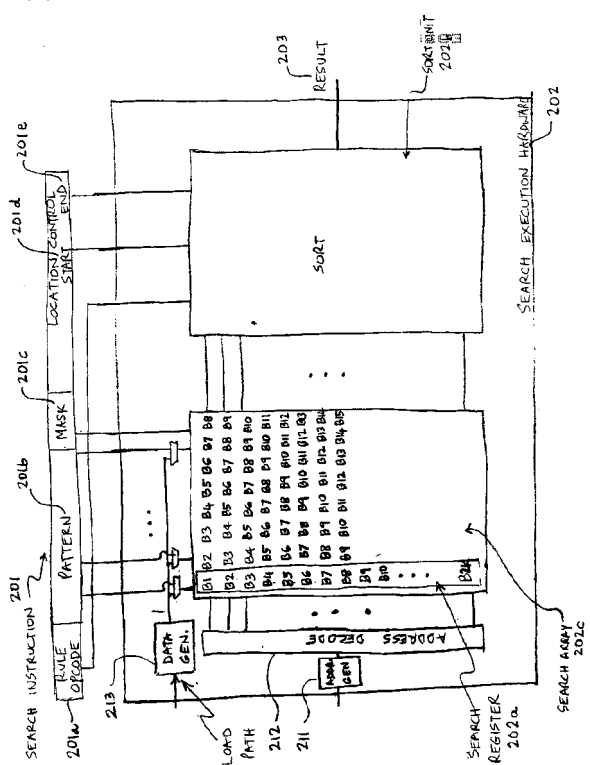
LOAD: ロード  
結果、レジスタ  
101 カード実行装置  
102

SEARCH REGISTERS (102)

SEARCH EXECUTION APPARATUS (106)

RESULT (107)

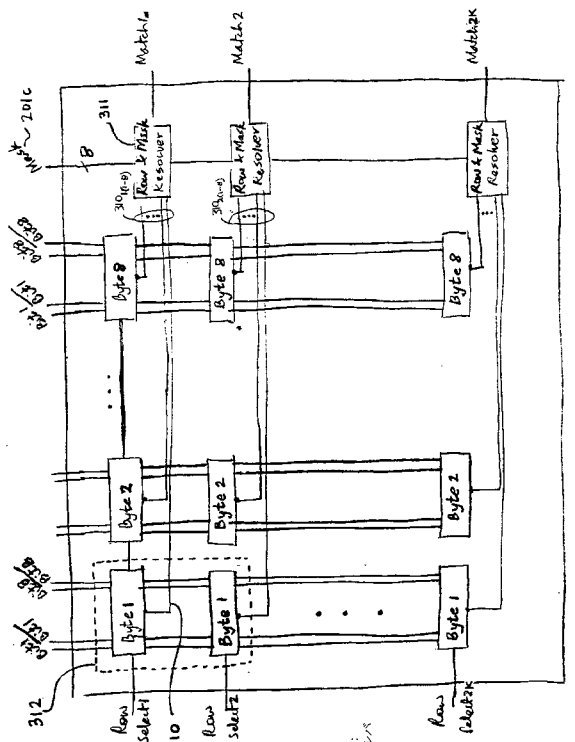
【図2A】



LOCATION CONTROL:  
ロード制御  
201 ソート  
201a ソート制御  
201b バタレン  
201c マスク  
201d 開始  
201e 終了  
202 パターン  
202a サーチ  
202b ソースタ  
202c ソート  
203 結果  
211 ソートレジスタ  
212 ソートレジスタ番号  
213 エネレジスタ

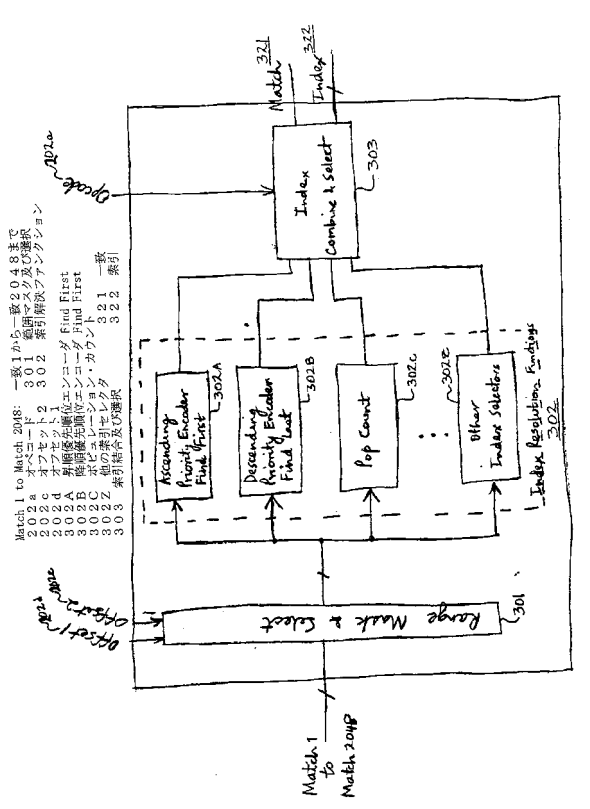
LOAD PATH:  
ロード制御  
201 ソート  
201a ソート制御  
201b バタレン  
201c マスク  
201d 開始  
201e 終了  
202 パターン  
202a サーチ  
202b ソースタ  
202c ソート  
203 結果  
211 ソートレジスタ  
212 ソートレジスタ番号  
213 エネレジスタ

【図2B】



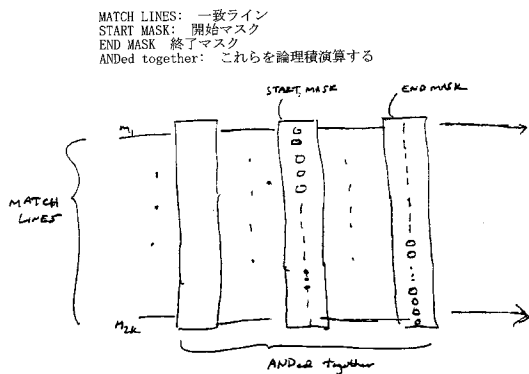
Bit: ビット  
Row Select: 行選択  
Byte: バイト  
Mask: マスク  
Row & Mask Resolver: 行とマスク・リソルバ

【図3A】

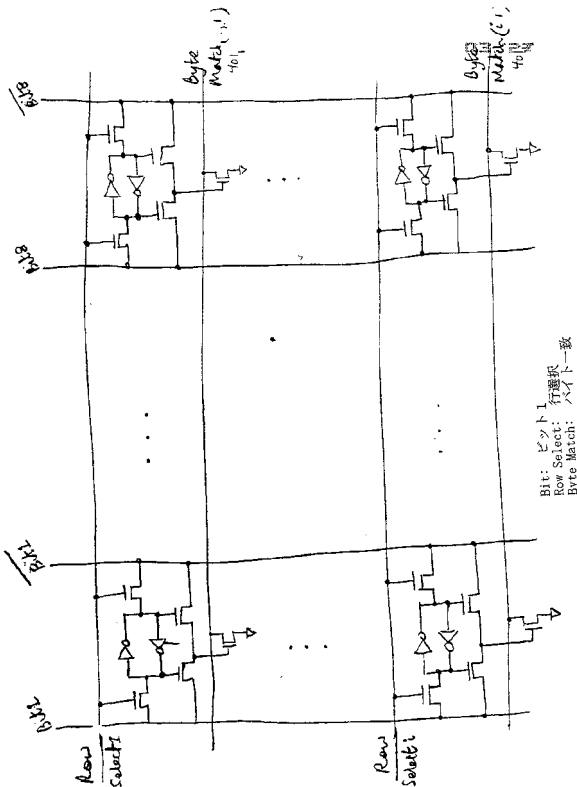


Match 1 to Match 2048: 一致1から一致2048検出  
301 索引解除アクション  
302 索引解除アクション  
303 索引解除アクション  
304 索引解除アクション  
305 索引解除アクション  
306 索引解除アクション  
307 索引解除アクション  
308 索引解除アクション  
309 索引解除アクション  
310 索引解除アクション  
311 索引解除アクション  
312 索引解除アクション

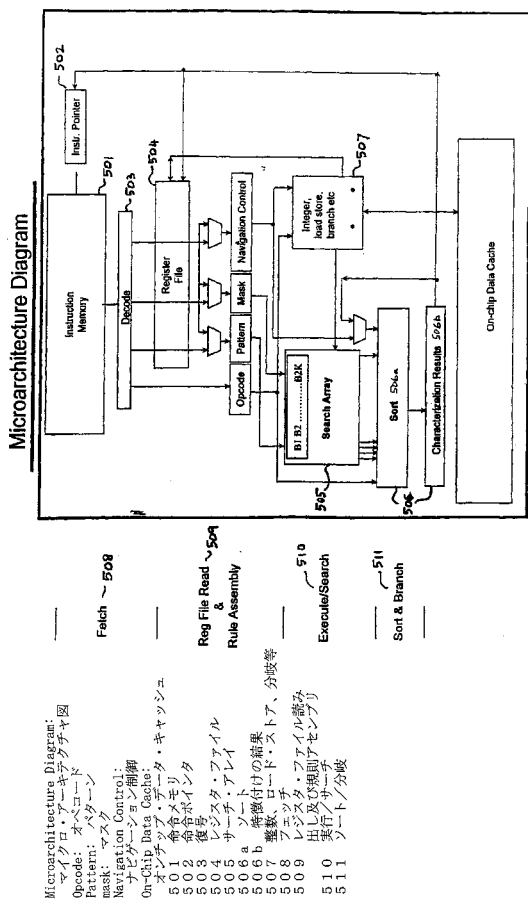
【図3B】



【図4】



【図5】



【図6】

### Example pseudo-code and Rule Engine micro-code

Pseudo-code ~ 601

FIND ("cp finish /url/spool/mail/root" BEFORE "chmod 4755 /s" AND "touch" BEFORE "mail")

Micro-code for an exemplary Rule Engine ~ 602

Opcode	Pattern	Mask	Start Offset	Start Offset Indirect	End Offset	End Offset Indirect	Result Register	Branch Address
1	FIND_FIRST_FORWARD	0xFF	0x03D	0	0x800	0	A	0xB
2	FIND_FORWARD_ANCHORED	0xFF	<A + B>	1	0x800	0	A	0xB
3	FIND_FORWARD_ANCHORED	0xFF	<A + B>	1	0x800	0	A	0xB
4	FIND_FORWARD_ANCHORED	0xFF	<A + B>	1	0x800	0	A	0xB
5	FIND_FORWARD_ANCHORED	0xFF	<A + B>	1	0x800	0	A	0xB
6	FIND_FORWARD_ANCHORED	0xF8	<A + B>	1	0x800	0	A	0xB
7	FIND_FORWARD_FORWARD	0xF8	0x03D	0	0x800	0	B	0xB
8	CMPIB->A->BRANCH(0)	"mail"	<A + B>	1	0x800	0	C	0xB
9	FIND_FIRST_FORWARD	0xF0	<B + B>	1	0x800	0	C	0xB
10	FIND_FIRST_FORWARD	"mail"	<B + B>	1	0x800	0	C	0xB

Example pseudo-code and Rule engine micro-code: 擬似コード例及び規則エンジンのマイクロコード  
 Pseudo-code: 擬似コード例  
 Rule Engine: 規則エンジン例に対するマイクロコード  
 Opcode: オペコード  
 Pattern: パターン  
 Mask: マスク  
 Start Offset: 開始オフセット  
 Start Offset Indirect: 開始オフセット  
 End Offset: 終了オフセット  
 End Offset Indirect: 終了オフセット  
 Result Register: 結果レジスタ  
 Branch address: 分岐アドレス

Execution of example micro-code: pipeline diagram

	Clock 1	Clock 2	Clock 3	Clock 4	Clock 5	Clock 6	Clock 7	Clock 8	Clock 9	Clock 10	Clock 11	Clock 12	Clock 13	Clock 14
To1	<b>Fetch</b>	1	2	3	4	5	6	7	8	-	10	-	-	-
	<b>Assemble</b>	-	1	2	3	4	5	6	7	8	-	10	-	-
	<b>Search/ Execute</b>	-	-	1	2	3	4	5	6	7	8	-	10	-
	<b>Sort &amp; Branch</b>	-	-	-	1	2	3	4	5	6	7	8	-	10

Execution of example micro-code: pipeline diagram: マイクロコード側の実行：パイプライン図  
 Clock: クロック  
 Assemble: アセンブル  
 Search/execute: カーチ/実行  
 Sort & Branch: ソート及び分岐

---

フロントページの続き

(72)発明者 パティル, ラジェッシュ

アメリカ合衆国・95051・カリフォルニア州・サンタクララ・サリバン ドライブ・3833

審査官 上嶋 裕樹

(56)参考文献 特開平04-315259(JP, A)

特開平06-202849(JP, A)

特開昭62-044829(JP, A)

特開2001-209563(JP, A)

高橋 恒介, テキスト検索プロセッサ, 日本, 社団法人電子情報通信学会, 1991年12月25日, 第1版, 第59-78頁

(58)調査した分野(Int.Cl., DB名)

G06F 17/30