

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7227002号
(P7227002)

(45)発行日 令和5年2月21日(2023.2.21)

(24)登録日 令和5年2月13日(2023.2.13)

(51)国際特許分類
G 0 5 B 23/02 (2006.01)F I
G 0 5 B 23/02 3 0 1 Y

請求項の数 25 (全42頁)

(21)出願番号	特願2018-518482(P2018-518482)	(73)特許権者	512132022 フィッシュ - ローズマウント システムズ, インコーポレイテッド アメリカ合衆国 テキサス 78681 - 7430 ラウンド ロック ウエスト ル
(86)(22)出願日	平成28年10月7日(2016.10.7)		イス ヘナ ブルバード 1100 ビルディング 1 エマーソン プロセス マネジメント
(65)公表番号	特表2018-530072(P2018-530072 A)	(74)代理人	110002860 弁理士法人秀和特許事務所
(43)公表日	平成30年10月11日(2018.10.11)	(72)発明者	ロウ, ゲーリー ケー. アメリカ合衆国 テキサス 78633 ジョージタウン ミシェル コート 110 シェリフ, ゴッドフリー アール.
(86)国際出願番号	PCT/US2016/056047	(72)発明者	アメリカ合衆国 テキサス 78717 最終頁に続く
(87)国際公開番号	WO2017/062801		
(87)国際公開日	平成29年4月13日(2017.4.13)		
審査請求日	令和1年10月7日(2019.10.7)		
(31)優先権主張番号	62/239,657		
(32)優先日	平成27年10月9日(2015.10.9)		
(33)優先権主張国・地域又は機関	米国(US)		
前置審査			

(54)【発明の名称】 プロセス制御システムの安全イベントの視覚化を経時的に提供するためのシステム及び方法

(57)【特許請求の範囲】

【請求項1】

プロセスプラント内の安全イベントを視覚化するコンピュータ実装の方法であって、前記方法が、

一組の原因及び一組の結果を有する原因結果マトリックス (C E M) にアクセスすることであって、前記一組の原因の各々が、前記プロセスプラント内の条件を表し、前記一組の結果の各々が、前記プロセスプラント内で行われるべき結果を表し、前記一組の結果の各々が、原因 - 結果の対として 1 以上の原因に関連され、それによって、前記原因 - 結果の対の各原因 - 結果の対に対して、前記原因 - 結果の対の対応する原因の発生に応答して前記原因 - 結果の対の対応する結果を起動させ、前記プロセスプラント内の一組の監視される安全イベントの各々が 1 以上の前記原因 - 結果の対に対応することと、

ユーザインターフェースを介して、前記一組の監視される安全イベントのうちの 1 つの監視される安全イベントの選択を受信することと、

前記監視される安全イベントの現在の状態を判定することであって、前記現在の状態は、前記 C E M の安全ロジックに対応する前記監視される安全イベントの安全レベルを示し、前記安全レベルは、前記監視される安全イベントに対応する前記 1 以上の原因 - 結果の対に基づいて定常状態からの状態の低下の程度を示し、前記状態の低下の程度はトリップされた前記監視される安全イベントに対応する前記 1 以上の原因 - 結果の対の数に基づいて判定されることと、

(i) 前記監視される安全イベントの指示、及び (ii) 前記監視される安全イベント

の前記現在の状態を前記ユーザインターフェースに表示することと、
前記監視される安全イベントの状態の変化を検出することと、
前記状態の変化に従って、前記監視される安全イベントの更新された状態を前記ユーザインターフェースに表示することと、を含む、コンピュータ実装の方法。

【請求項 2】

前記監視される安全イベントの前記状態の変化を検出することが、
期間の終了を検出することを含む、請求項 1 に記載のコンピュータ実装の方法。

【請求項 3】

前記監視される安全イベントの前記現在の状態を表示することが、
前記監視される安全イベントの前記現在の状態を第 1 のグラフィカルオブジェクトとして表示することを含み、
前記監視される安全イベントの前記更新された状態を表示することが、
前記監視される安全イベントの前記更新された状態を、前記第 1 のグラフィカルオブジェクトと異なる、第 2 のグラフィカルオブジェクトとして表示することを含む、請求項 1 または 2 に記載のコンピュータ実装の方法。

【請求項 4】

前記監視される安全イベントの前記現在の状態を表示することが、
前記監視される安全イベントの前記現在の状態を少なくとも 2 つの第 1 のグラフィカルオブジェクトとして表示することを含み、
前記監視される安全イベントの前記更新された状態を表示することが、
前記監視される安全イベントの前記更新された状態を少なくとも 2 つの第 2 のグラフィカルオブジェクトとして表示することを含む、請求項 1 から 3 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 5】

前記監視される安全イベントの前記更新された状態を表示することが、
前記監視される安全イベントの前記現在の状態と前記更新された状態との間の変化の程度を判定することと、
前記変化の程度を前記ユーザインターフェースに表示することと、を含む、請求項 1 から 4 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 6】

期間の終了に応答して、前記監視される安全イベントの前記更新された状態が変化しなかったと判定することと、
前記監視される安全イベントの前記更新された状態を前記ユーザインターフェースに表示することと、をさらに含む、請求項 1 から 5 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 7】

前記ユーザインターフェースを介して、前記監視される安全イベントの前記更新された状態の選択を受信することであって、前記監視される安全イベントが、一組の関連付けられた条件を有することと、
前記一組の関連付けられた条件の各々の条件状態を前記ユーザインターフェースに表示することと、をさらに含む、請求項 1 から 6 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 8】

前記ユーザインターフェースを介して、前記監視される安全イベントの前記更新された状態の選択を受信することであって、前記監視される安全イベントが、存在する関連付けられた条件を有することと、
前記関連付けられた条件を存在させる、前記プロセスプラント内のデバイスの指示を前記ユーザインターフェースに表示することと、をさらに含む、請求項 1 から 7 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 9】

10

20

30

40

50

(i) 前記一組の監視される安全イベントのうちの追加的な 1 つの監視される安全イベントの追加的な指示、及び (i i) 前記追加的な監視される安全イベントの追加的な現在の状態を前記ユーザインターフェースに表示することと、

前記追加的な監視される安全イベントの追加的な状態の変化を検出することと、

前記追加的な状態の変化に従って、前記追加的な監視される安全イベントの追加的な更新された状態を前記ユーザインターフェースに表示することと、をさらに含む、請求項 1 から 8 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 1 0】

プロセスプラント内の安全イベントを視覚化するためのシステムであって、
ユーザインターフェースと、

(i) 一組のコンピュータ実行可能命令、及び (i i) 一組の原因及び一組の結果を有する原因結果マトリックス (C E M) を記憶するように構成されたメモリであって、前記一組の原因の各々が、前記プロセスプラント内の条件を表し、前記一組の結果の各々が、前記プロセスプラント内で行われるべき結果を表し、前記一組の結果の各々が、原因 - 結果の対として 1 以上の原因に関連され、それによって、前記原因 - 結果の対の各原因 - 結果の対に対して、前記原因 - 結果の対の対応する原因の発生に応答して前記原因 - 結果の対の対応する結果を起動させ、前記プロセスプラント内の一組の監視される安全イベントの各々が 1 以上の前記原因 - 結果の対に対応する、メモリと、

前記ユーザインターフェース及び前記メモリとインターフェースされたプロセッサであって、前記一組のコンピュータ実行可能命令を実行して、前記プロセッサに、

前記ユーザインターフェースを介して、前記一組の監視される安全イベントの監視される安全イベントの選択を受信させ、

前記監視される安全イベントの現在の状態を判定させ、

前記現在の状態は、前記 C E M の安全ロジックに対応する前記監視される安全イベントの安全レベルを示し、前記安全レベルは、前記監視される安全イベントに対応する前記 1 以上の原因 - 結果の対に基づいて定常状態からの状態の低下の程度を示し、前記状態の低下の程度はトリップされた前記監視される安全イベントに対応する前記 1 以上の原因 - 結果の対の数に基づいて判定され、

前記ユーザインターフェースに、(i) 前記監視される安全イベントの指示、及び (i i) 前記監視される安全イベントの前記現在の状態を表示させ、

前記監視される安全イベントの状態の変化を検出させ、

前記ユーザインターフェースに、前記状態の変化に従って、前記監視される安全イベントの更新された状態を表示させる、プロセッサと、を備える、システム。

【請求項 1 1】

前記監視される安全イベントの前記状態の変化を検出するために、前記プロセッサが、期間の終了に応答して、前記監視される安全イベントの前記状態の変化を検出するように構成される、請求項 1 0 に記載のシステム。

【請求項 1 2】

前記ユーザインターフェースに、前記監視される安全イベントの前記現在の状態を表示させるために、前記プロセッサが、

前記ユーザインターフェースに、前記監視される安全イベントの前記現在の状態を第 1 のグラフィカルオブジェクトとして表示させるように構成され、

前記ユーザインターフェースに、前記監視される安全イベントの前記更新された状態を表示させるために、前記プロセッサが、

前記ユーザインターフェースに、前記監視される安全イベントの前記更新された状態を、前記第 1 のグラフィカルオブジェクトと異なる、第 2 のグラフィカルオブジェクトとして表示させるように構成される、請求項 1 0 または 1 1 に記載のシステム。

【請求項 1 3】

前記ユーザインターフェースに、前記監視される安全イベントの前記現在の状態を表示させるために、前記プロセッサが、

10

20

30

40

50

前記ユーザインターフェースに、前記監視される安全イベントの前記現在の状態を少なくとも2つの第1のグラフィカルオブジェクトとして表示させるように構成され、前記ユーザインターフェースに、前記監視される安全イベントの前記更新された状態を表示させるために、前記プロセッサが、

前記ユーザインターフェースに、前記監視される安全イベントの前記更新された状態を少なくとも2つの第2のグラフィカルオブジェクトとして表示させるように構成される、請求項10から12のいずれか1項に記載のシステム。

【請求項14】

前記ユーザインターフェースに、前記監視される安全イベントの前記更新された状態を表示させるために、前記プロセッサが、

前記監視される安全イベントの前記現在の状態と前記更新された状態との間の変化の程度を判定し、

前記ユーザインターフェースに、前記変化の程度を表示させるように構成される、請求項10から13のいずれか1項に記載のシステム。

【請求項15】

前記プロセッサが、

前記ユーザインターフェースを介して、前記監視される安全イベントの前記更新された状態の選択を受信し、前記監視される安全イベントが、一組の関連付けられた条件を有し、

前記ユーザインターフェースに、前記一組の関連付けられた条件の各々の条件状態を表示させるようにさらに構成される、請求項10から14のいずれか1項に記載のシステム。

【請求項16】

前記プロセッサが、

前記ユーザインターフェースを介して、前記監視される安全イベントの前記更新された状態の選択を受信するようにさらに構成され、前記監視される安全イベントが、存在する前記関連付けられた条件を有し、

前記ユーザインターフェースに、前記関連付けられた条件を存在させる、前記プロセスプラント内のデバイスの指示を表示させるようにさらに構成される、請求項10から15のいずれか1項に記載のシステム。

【請求項17】

前記プロセッサが、

前記ユーザインターフェースに、(i)前記一組の監視される安全イベントのうちの追加的な1つの監視される安全イベントの追加的な指示、及び(ii)前記追加的な監視される安全イベントの追加的な現在の状態を表示させ、

前記追加的な監視される安全イベントの追加的な状態の変化を検出し、

前記ユーザインターフェースに、前記追加的な状態の変化に従って、前記追加的な監視される安全イベントの追加的な更新された状態を表示させるようにさらに構成される、請求項10から16のいずれか1項に記載のシステム。

【請求項18】

プロセスプラント内の安全イベントを視覚化するコンピュータ実装の方法であって、前記方法が、

一組の原因及び一組の結果を有する前記プロセスプラントに対応する原因結果マトリックス(CEM)の安全ロジックにアクセスすることであって、前記一組の原因の各々が、前記プロセスプラント内の条件を表し、前記一組の結果の各々が、1以上の前記条件に基づいて前記プロセスプラント内で行われるべき行動を表し、前記一組の結果の各々が原因-結果の対として1以上の原因に関連され、前記プロセスプラント内の一組の監視される安全イベントの各々が1以上の前記原因-結果の対に対応することと、

ユーザインターフェースを介して、前記一組の監視される安全イベントのうちの1つの監視される安全イベントの選択を受信することと、

前記監視される安全イベントの現在の状態を判定することであって、前記現在の状態は、前記安全ロジックに対応する前記監視される安全イベントの安全レベルを示し、前記安

10

20

30

40

50

全レベルは、前記監視される安全イベントに対応する前記 1 以上の原因 - 結果の対に基づいて定常状態からの状態の低下の程度を示し、前記状態の低下の程度はトリップされた前記監視される安全イベントに対応する前記 1 以上の原因 - 結果の対の数に基づいて判定されることと、

前記監視される安全イベントの前記現在の状態を前記ユーザインターフェースに表示することと、
を含む、コンピュータ実装の方法。

【請求項 19】

前記ユーザインターフェースを介して、前記監視される安全イベントの 1 以上の表示のメニューを表示することと、

ユーザインターフェースを介して、表示のタイプの選択を受信することと、及び、
前記ユーザインターフェースを介して、前記表示タイプに従って前記監視される安全イベントの前記現在の状態を表示することと、
をさらに含む、請求項 18 に記載のコンピュータ実装の方法。

【請求項 20】

前記表示タイプの前記選択は、1 以上のシンボル、カラー、グラフィック、線、数値表示の選択である、

請求項 19 に記載のコンピュータ実装の方法。

【請求項 21】

前記ユーザインターフェースを介して、第 2 の監視される安全イベントの選択を受信することと、及び、

前記ユーザインターフェースを介して、前記監視される安全イベントの前記現在の状態に一致して、前記第 2 の監視される安全イベントの現在の状態を表示することと、
をさらに含む、請求項 18 から 20 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 22】

前記選択された監視される安全イベントが前記 1 組の結果からの結果に対応することを判定することと、及び、

前記ユーザインターフェースを介して、前記結果に対応する 1 以上の条件に関連する情報を表示することと、

をさらに含む、請求項 18 から 21 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 23】

前記ユーザインターフェースを介して、前記表示された監視される安全イベントの要素の選択を受信することと、及び

前記ユーザインターフェースを介して、前記選択された要素に対応するデバイスを含む前記システム構成のビューを表示することと、

をさらに含む、請求項 18 から 22 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 24】

前記監視される安全イベントが前記パワープラントの 1 以上の条件に対応する結果に対応することを判定することと、及び

前記ユーザインターフェースを介して、前記条件の各々の指示を表示することと、
をさらに含む、請求項 18 から 22 のいずれか 1 項に記載のコンピュータ実装の方法。

【請求項 25】

プロセスプラント内のイベントを検出するコンピュータ実装の方法であって、前記方法が、

ユーザが前記プロセスプラント内の監視されるべき 1 以上の安全イベントを定義できるようにすることであって、前記 1 以上の安全イベントの各安全イベントが一組の原因及び二組の結果を有する原因結果マトリックス (C E M) に対応し、前記一組の原因の各々が前記プロセスプラント内の条件を示し、前記一組の結果の各々が前記プロセスプラント内で行われるべき結果を示し、前記一組の結果の各々が原因 - 結果の対として 1 以上の原因に関連され、それによって、前記原因 - 結果の対の各原因 - 結果の対に対して、前記原因

10

20

30

40

50

- 結果の対の対応する原因の発生に応答して前記原因 - 結果の対の対応する結果を起動させ、前記 1 以上の安全イベントの各安全イベントが 1 以上の前記原因 - 結果の対に対応することと、

ユーザインターフェースを介して、プラント内の監視されるべき前記 1 以上の安全イベントの安全イベントの選択を受信することと、

プロセッサを介して、前記選択された安全イベントの前記現在の状態を検出し、前記現在の状態は前記選択された安全イベントに対応する前記 1 以上の原因 - 結果の対に基づいて定常状態からの状態の低下の程度を示し、前記状態の低下の程度はトリップされた前記選択された安全イベントに対応する前記 1 以上の原因 - 結果の対の数に基づいて判定されることと、及び、

表示デバイスで、トリップされた前記選択された安全イベントに対応する前記 1 以上の原因 - 結果の対の前記数に関連するグラフィカル指示として前記選択された安全イベントの状態を表示することと、

を含む、コンピュータ実装の方法。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願

本特許出願は、2015年10月9日に出願された「A System and Method for Configuring Separated Monitor and Effect Blocks of a Process Control System」という名称の米国仮特許出願第62/239,657号の出願日に対する優先権及び利益を主張する、正規の特許出願であり、参照により明示的に本明細書に組み込まれる。

【0002】

本開示は、一般に、プロセスプラント内のプロセス制御システムを管理することに関し、より具体的には、プロセス制御システムと関連付けられた原因結果マトリックス（CEM）を構成すること、及び該CEMに関連する監視ブロック及び結果ブロックを作成することに関する。

【背景技術】

【0003】

化学、石油、または他のプロセスで使用されているようなプロセス制御システムは、典型的に、アナログ、デジタル、またはアナログ/デジタルバスもしくはラインを介して、少なくとも1つのホストもしくはオペレーターワークステーションに、及び1つ以上のフィールドデバイスに通信的に結合された、1つ以上のプロセスコントローラを含む。例えば、弁、弁ポジショナ、スイッチ、及びトランスマッタ（例えば、温度、圧力、及び流量センサ）であり得るフィールドデバイスは、弁の開閉、及びプロセスパラメータの測定等のプロセスプラント内の機能を行う。プロセスコントローラは、フィールドデバイスによって作成されるプロセス測定値及び/またはフィールドデバイスに関する他の情報を示す信号を受信し、この情報を使用して制御ルーチンを実装し、次いで、制御信号を発生させ、該制御信号は、バスまたはラインを通じてフィールドデバイスに送信されて、プロセスの動作を制御する。フィールドデバイス及びコントローラからの情報は、典型的に、オペレーターワークステーションによって実行される1つ以上のアプリケーションが利用できるようになり、オペレータが、プロセスを構成すること、プロセスの現在の状態を見ること、プロセスの動作を修正すること等の、プロセスに関する任意の所望の機能を行うことを可能にする。

【0004】

加えて、多くのプロセスでは、毒性化学物質の漏洩、爆発等のプラント内の深刻な危険をもたらし得る、またはつながり得る問題が生じたときに、プロセスプラント内の重大な安全関連の問題を検出して、自動的に弁を閉じる、デバイスからの電力供給を停止する、プラント内の流れを切り換える等のために、別個の安全システムが提供される。これらの

安全システムは、典型的に、標準的なプロセス制御コントローラとは別の、論理ソルバーと称される1つ以上の別個のコントローラを有し、該コントローラは、プロセス制御プラント内に設置された別個のバスまたは通信ラインを介して、安全なフィールドデバイスに接続される。論理ソルバーは、安全フィールドデバイスを使用して、特定の安全スイッチまたは遮断弁の位置、プロセスにおけるオーバーフローもしくはアンダーフロー、重要な発電もしくは制御デバイスの動作、障害検出デバイスの動作等の重要なイベントと関連付けられたプロセス条件を検出し、それによって、プロセスプラント内の「イベント」を検出する。単一の条件または2つ以上の条件の同時発生であり得るイベント（典型的に、「原因」と称される）が検出されると、安全コントローラは、弁を閉じる、デバイスをオフにする、プラントのセクションからの電力供給を停止する等の、イベントの有害な性質を制限するために、いくつかのアクション（典型的に、「結果」と称される）をとる。一般に、これらのアクションまたは結果は、安全デバイスを、プロセスプラント内の深刻なまたは危険な条件を防止するように設計された、トリップまたは「安全」動作モードに切り換えることを含む。

【0005】

マネージャ及びエンジニア等のプロセスプラントのオペレータは、典型的に、関連する原因結果を記憶するデータベース構造を維持する。例えば、マトリックスは、複数の行及び列を有することができ、各行は、原因に対応し、各列は、結果に対応し、また、マトリックスの各セルは、特定の原因結果関係に対応する。セルは、原因及び結果それぞれの関係を示す様々なトリガーによってポピュレートすることができる。いわゆる原因結果マトリックス（CEM）は、一般に、制御システムまたはプラントの安全設計を定義する要求文書に従って構成される。制御エンジニアは、CEMを利用して、適宜に安全設計が実装されるように制御システムをエンジニアリングすることができる。しかしながら、そのようなCEMは、マトリックスの定義されたサイズによって制限され、また、しばしば、全ての所望の原因 / 結果データの関係を扱うのに十分な大きさではない。さらに、そのようなCEMは、連鎖化、リンク化、平準化、ループ化等のより複雑 / 精巧な原因 / 結果を扱うことができない。またさらに、大きいCEMは、制御論理への実装が面倒であり、したがって、実装中にエラーを起こしやすい。適切に動作させるための安全システムの故障は、プラント人員側の深刻な負傷、さらには死亡に、また、潜在的に数百万ドルのプラント内の設備及び材料の破壊につながる可能性があるので、CEMにおけるエラーは深刻になり得るため、安全システムでは、正確なCEMを維持することが必須である。

【発明の概要】

【0006】

プロセスプラントのプロセス制御システムは、原因結果マトリックス（CEM）において定義された制御論理を達成するように実装または設計することができる安全システムを有することができ、CEMは、視覚表現で表示される、プロセスプラントの安全アクションの概要である。全般的に言えば、CEMは、プロセスプラント内の様々な安全プロトコルまたはプロシージャの基本的な原因結果関係を定義する。一般に、CEMは、一組の入力と、一組の出力を含むことができ、一組の入力の各々は、プロセスプラント内の条件を表し、一組の出力の各々は、プロセスプラント内で行われるべき結果またはアクションを表す。さらに、一組の入力及び一組の出力のうちの少なくともいくつかは、原因 - 結果の対として関連され、それによって、対応する結果のアクティビティが、対応する条件または原因の発生に応答する。

【0007】

プロセス制御システムの管理者は、一組の様々な機能ブロックとしてCEMを実装することができる。しかしながら、プロセスプラントの規模及び / または複雑さに応じて、所与のCEMは、多数の原因、結果、及び原因 - 結果の対を含む場合があり、したがって、対応する大量の機能ブロックを実装することが必要になり得る。したがって、この実装は、時間がかかり、複雑で、面倒になる可能性があり、潜在的な実装エラーにつながる。説明されるシステム及び方法によれば、監視機能ブロック及び結果機能ブロックとしてCE

M論理を実装するように説明される、分離されているが相互接続されている一組の機能ブロックとして、CEMをプロセス制御システム内に実装するための様々な技術が提供される。

【0008】

1つの実施形態において、システム及び方法は、CEM内のパターン及びグルーピングを識別することができ、また、識別したパターン及びグルーピングに従って一組の監視ブロック及び一組の結果ブロックを実装することができ、したがって、CEMの実装の複雑さを低減する。一実装例において、CEM内のデータのグルーピング（例えば、CEMの列）は、CEMのその一部分によって定義される論理の数値表現として定義して、CEM論理を実装するために使用される機能ブロック（例えば、監視及び結果ブロック）内にCEMの論理が実装されることを理解し、検証する、簡単であまり複雑でない様態を提供することができる。またさらに、CEM（例えば、CEMの行及び／または列）を分析し、かつ再順序付けまたは再配設して、より良好な、より論理的な、より容易に実装される、等の、一組以上の原因結果ブロックとして実装されるCEM論理のグルーピングを提供するために、ツールを使用することができる。

10

【0009】

本開示は、CEMを管理するための追加的な技術を提供する。具体的には、本明細書で説明されるシステム及び方法は、CEMがインタラクティブ機能を含むように構成するために使用することができる。例えば、構成されたCEMは、CEMの原因結果関係を構成する安全プロトコルを詳述する1つ以上の文書にアクセスするリンクもしくは選択肢、プラントに実装されたときにユーザが特定の結果に関する以前の条件もしくは動作をより容易に理解することを可能にするための、CEMの1つ以上の結果の現在及び／もしくは過去の状態を表すグラフ、ならびにCEMの原因結果に関連するデバイスを含むプロセスプラントの図を含むことができる。

20

【0010】

加えて、CEMに含まれるしばしば大量の情報のため、エンジニアは、プロセス制御システムに含まれる任意の不一致またはエラーを識別することが困難であり得る。本明細書で提供されるシステム及び方法は、さらに、リバースエンジニアリング技術及びシステムを実装して、デバイスによって実際に実装されるCEM論理、及びプロセスプラントの制御論理（またはいくつかの実装例において、プロセス制御システム内の監視及び結果ブロック）を定義する試験CEM、ならびに特定のプロセスプラントのための必要な安全プロトコルを自動的に作成することを可能にする。故に、本明細書で説明されるシステム及び方法は、試験CEMと既存のCEMとを比較して、プラント動作の実際の構成と設計文書に詳述され得る構成との間の任意の不一致またはエラーを識別することができる。

30

【0011】

下で説明される図は、開示されるシステム及び方法の様々な態様をその中に表す。各図は、開示されるシステム及び方法の特定の態様の一実施形態を表すこと、及び図の各々は、その可能な実施形態と一致することを意図することを理解されたい。さらに、可能な限り、以下の説明は、以下の図に含まれる参照番号を参照し、複数の図面において表される特徴は、一貫した参照番号で示される。

40

【0012】

図面には、現在論じられている配設が示されるが、本実施形態は、示される正確な配設及び手段に限定されないことを理解されたい。

【図面の簡単な説明】

【0013】

【図1】 例示的なプロセスプラントのブロック図である。

【図2】 図1に概略的に例示される例示的なワークステーションのブロック図である。

【図3】 例示的な原因結果マトリックスの説明図である。

【図4】 例示的な一組の監視及び結果ブロックの説明図である。

【図5】 監視及び結果ブロックを実装するために使用することができる一組の機能ブロッ

50

クの第 1 の実施例の説明図である。

【図 6】監視及び結果ブロックを実装するために使用することができる一組の機能ブロックの第 2 の実施例の説明図である。

【図 7】プロセスプラントと関連付けられた監視ブロック及び結果ブロックを構成する例示的な方法のフロー図である。

【図 8】原因結果マトリックスの第 2 の実施例の説明図である。

【図 9】別個の論理ブロックに再編成され、構成された図 8 の原因結果マトリックスの第 2 の実施例の説明図である。

【図 10】原因結果マトリックスを再編成する例示的な方法のフロー図である。

【図 11】例示的な数値表現を有する原因結果マトリックスの第 2 の実施例の説明図である。

【図 12】原因結果マトリックス論理の数値表現を算出する例示的な方法のフロー図である。

【図 13】原因結果マトリックスの安全論理に対応する様々なインターリンクされたユーザインターフェースの例示的な説明図である。

【図 14】図 13 のインターリンクされたユーザインターフェース間をナビゲートするための例示的な方法のフローチャートである。

【図 15】試験原因結果マトリックスをリバースエンジニアリングするための例示的な方法のフローチャートである。

【図 16 A】監視される安全イベントを表示する例示的なユーザインターフェースの説明図である。

【図 16 B】監視される安全イベントを表示する例示的なユーザインターフェースの説明図である。

【図 16 C】監視される安全イベントを表示する例示的なユーザインターフェースの説明図である。

【図 16 D】監視される安全イベントを表示する例示的なユーザインターフェースの説明図である。

【図 17】監視される安全イベントを表示するための例示的な方法のフローチャートである。

【図 18】許容及び時間遅延トリガーを含む例示的な原因結果マトリックスの説明図である。

【0014】

図面は、単に例示の目的で、好ましい実施形態を表す。当業者は、本明細書で説明される本発明の原理を逸脱することなく、本明細書で例示されるシステム及び方法の代替の実施形態が用いられ得ることを、以下の議論から容易に認識するであろう。

【発明を実施するための形態】

【0015】

図 1 は、1 つ以上のノード 12、16、18、及び 20 を含む、例示的なプロセスプラント 10 のブロック図である。図 1 の例示的なプロセスプラント 10 において、ノード 12 及び 16 の各々は、例えば Foundation Fieldbus インターフェース、HART インターフェース等とすることができる入力 / 出力 (I/O) デバイス 24 を介して、1 つ以上のフィールドデバイス 22 及び 23 に接続されたプロセスコントローラ 12a、16a を含む。コントローラ 12a 及び 16a はまた、例えば、1 つ以上のバス、イーサネットローカルエリアネットワーク (LAN) 等の有線 LAN、無線 LAN、ワイドエリアネットワーク (WAN)、インターネット等を備えることができるネットワーク 30 を介して、ノード 18 及び 20 の 1 つ以上のホストまたはオペレーターワークステーション 18a 及び 20a にも結合される。コントローラノード 12、16、ならびに該コントローラノードと関連付けられた I/O デバイス 24 及びフィールドデバイス 22、23 は、典型的に、あるときには過酷なプラント環境内の下流に位置付けられ、該プラント環境を通して分配され、オペレーターワークステーションノード 18 及び 20 は、通常、コ

10

20

30

40

50

ントローラ人員によって容易に評価することができる、制御室内または他のあまり過酷でない環境内に位置付けられる。

【 0 0 1 6 】

一般的に言えば、ノード 18 及び 20 のワークステーション 18 a 及び 20 a は、プロセスプラント 10 を構成し、監視するために、ならびに / またはプロセスプラント 10 のデバイス 22、23、24 及びコントローラ 12 a、16 a を管理するために使用されるアプリケーションを記憶し、実行するために使用することができる。例えば、ワークステーション 18 a 及び / または 20 a は、システムナビゲータアプリケーション 15、原因結果分析器ツール 17、プロセス制御構成アプリケーション 19、及びプロセスプラント 10 の安全要求を管理するために実装することができる安全構成アプリケーション 21 等のツールを含むことができる。システムナビゲータアプリケーション 15 は、プロセスプラントの安全要求及びデバイスに関する情報を提供する、ユーザインターフェースのインターネットリンクされたグループを提供するように実装することができる。原因結果分析器ツールは、原因結果マトリックス (C E M) を管理するように、及び / または既知の安全要求及び / または機能ブロックからリバースエンジニアリングすることによって原因結果マトリックスを作成するように実装することができる。さらに、プロセス制御構成アプリケーション 19 及び安全構成アプリケーション 21 は、ワークステーション 18 a 及び / または 20 a を通してプロセスプラントのデバイスを管理する能力をユーザに提供する。構成データベース 32 は、ネットワーク 30 に接続することができ、また、ノード 12、16、18、20 にダウンロードされ、及び / または該ノード内に記憶されたときに、プロセスプラント 10 の現在の構成を記憶する、データヒストリヤン及び / または構成データベースとして動作することができる。構成データベースはまた、C E M を再配設するための規則 31、及び / または数値表現 33 を含むこともできる。

10

【 0 0 1 7 】

一例として Emerson Process Management によって販売される DeltaV (商標) コントローラとすることができます、コントローラ 12 a 及び 16 a の各々は、いくつかの異なる、独立して実行される制御モジュールまたはブロックを使用して制御ストラテジを実装するコントローラアプリケーションを記憶し、実行することができる。制御モジュールは、それぞれ、一般に機能ブロックと称されるもので構成することができ、各機能ブロックは、制御ルーチン全体の一部またはサブルーチンであり、また、(リンクと呼ばれる通信を介して)他の機能ブロックと連動して、プロセスプラント 10 内のプロセス制御ループを実装するように動作する。よく知られているように、機能ブロックは、典型的に、(トランスマッタ、センサ、または他のプロセスパラメータ測定デバイス等と関連付けられた)入力機能、(P I D、ファジー論理等の制御を行う制御ルーチン等と関連付けられた)制御機能、または(弁等の)いくつかのデバイスの動作を制御する出力機能、のうちの 1 つを行って、プロセスプラント 10 内のいくつかの物理的機能を行う。当然ながら、ハイブリッド及び他のタイプの機能ブロックが存在し、利用することができる。フィールドバスプロトコル及び DeltaV (商標) システムプロトコルは、オブジェクト指向プログラミングプロトコルで設計され、実装された制御モジュール及び機能ブロックを使用することができるが、制御モジュールは、例えばシーケンシャル機能ブロック、ラダーロジック等の任意の所望の制御プログラミングスキームを使用して設計することができ、また、機能ブロックまたは任意の他の特定のプログラミング技術を使用して設計されることに限定されない。典型的であるように、プロセス制御ノード 12 及び 16 内に記憶したときの制御モジュールの構成は、ワークステーション 18 a 及び 20 a によって実行されるアプリケーションにアクセスすることができる構成データベース 32 に記憶することができる。機能ブロックは、例えばコントローラ 12 a、16 a に記憶され、それによって実行することができ、これは典型的に、これらの機能ブロックが、標準的な 4 ~ 20 mA デバイス、及び H A R T デバイス等のいくつかのタイプのスマートファイールドデバイスに使用されるときに、または該デバイスと関連付けられるときに当たはまり、または該機能ブロックは、フィールドデバイス自体に記憶し、それによって実装す

20

30

40

50

ることができ、これは、フィールドバスデバイスの場合に当てはまり得る。

【0018】

図1に例示されるシステムにおいて、コントローラ12a及び16aに結合されたフィールドデバイス22及び23は、標準的な4~20mAデバイスとすることができる、またはプロセッサ及びメモリを含む、HART、Profibus、もしくはFoundationフィールドバスフィールドデバイス等の、スマートフィールドデバイスとすることができる。Foundationフィールドバスフィールドデバイス等の（図1において参照番号23が付される）これらのデバイスのうちのいくつかは、コントローラ12a及び16aに実装された制御ストラテジと関連付けられた、機能ブロック等のモジュールまたはサブモジュールを記憶し、実行することができる。当然ながら、フィールドデバイス22、23は、センサ、弁、トランスミッタ、ポジショナ等の任意のタイプのデバイスとすることができる、I/Oデバイス24は、HART、Foundationフィールドバス、Profibus等の任意の所望の通信またはコントローラプロトコルに準拠する任意のタイプのI/Oデバイスとすることができる。

10

【0019】

コントローラ12a及び16aは、それぞれ、メモリに記憶された1つ以上のプロセス制御ルーチンを実装または監督するプロセッサを含み、該プロセス制御ルーチンは、その中に記憶された、または別様にはそれと関連付けられた制御ループを含むことができる。コントローラ12a及び16aは、フィールドデバイス22、23、ワークステーション18a、20a、及びデータベース32と通信して、任意の所望の様式でプロセスを制御する。コントローラ12a及び16aは、それぞれ、任意の所望の様態で制御ストラテジまたは制御ルーチンを実装するように構成することができる。

20

【0020】

プロセスプラント10はまた、プロセス制御ノード12及び16と統合された安全システム14（破線で示す）も含むことができる。安全システム14は、一般に、プロセス制御ノード12及び16によって提供される制御を監視し、オーバーライドして、プロセスプラント10の適切な安全動作を最大にするために、安全計装システム（SIS）として動作させることができる。

【0021】

ノード12及び16の各々は、1つ以上の安全システム論理ソルバー50を含むことができる。論理ソルバー50の各々は、プロセッサ及びメモリを有するI/Oデバイスであり、また、メモリに記憶した安全論理モジュールを実行するように構成される。各論理ソルバー50は、安全システムフィールドデバイス60及び62に制御信号を提供し、及び/またはそれらから信号を受信するように通信的に結合される。加えて、ノード12及び16の各々は、少なくとも1つのメッセージ伝達デバイス（MPD）70を含むことができ、該MPDは、リングまたはバス接続74（図1にはその一部のみが例示される）を介して、他のMPD70に通信的に結合される。安全システム論理ソルバー50、安全システムフィールドデバイス60及び62、MPD70、ならびにバス74は、一般に、図1の安全システム14を構成する。

30

【0022】

図1の論理ソルバー50は、プロセッサと、メモリとを含む任意の所望のタイプの安全システム制御デバイスとすることができる、該メモリは、プロセッサに対して実行して、フィールドデバイス60及び62を使用する安全システム14と関連付けられた制御機能を提供するように適合された安全論理モジュールを記憶する。当然ながら、安全フィールドデバイス60及び62は、上で述べたもの等の、任意の既知の、または所望の通信プロトコルに準拠する、またはそれを使用する、任意の所望のタイプのフィールドデバイスとすることができる。具体的には、フィールドデバイス60及び62は、因習的に別個の専用の安全関連の制御システムによって制御されるタイプの安全関連のフィールドデバイスとすることができる。図1に例示されるプロセスプラント10において、安全フィールドデバイス60は、HARTまたは4~20mAプロトコル等の専用の、またはポイントツー

40

50

ポイントの通信プロトコルを使用するように表されおり、一方で、安全フィールドデバイス 62 は、Fieldbus プロトコル等のバス通信プロトコルを使用するように表される。安全フィールドデバイス 60 は、遮断弁、遮断スイッチ等の機能等の、任意の所望の機能を行うことができる。しかしながら、安全システムフィールドデバイス 60 及び 62 は、他のタイプのデバイスとすることができます、また、任意の所望の有線または無線通信プロトコルを含む、論理ソルバー 50 と通信する他のタイプの通信プロトコルを使用することができます。

【0023】

ノード 12 及び 16 の各々において共通のバックプレーン（図示せず）を使用して、コントローラ 12a 及び 16a を、プロセス制御 I/O カード 24、安全論理ソルバー 50、及び MPD 70 に通信的に結合することができる。コントローラ 12a 及び 16a はまた、ネットワーク 30 にも通信的に結合される。コントローラ 12a 及び 16a、I/O デバイス 24、論理ソルバー 50、MPD 70 は、ネットワーク 30 を介して、ノード 18 及び 20 と通信することができます。

10

【0024】

当業者によって理解されるように、ノード 12、16 のバックプレーン（図示せず）は、論理ソルバー 50 が互いにローカルに通信して、これらのデバイスによって実装される安全機能を協調させること、データを互いに通信すること、及び / または他の統合機能を行うことを可能にする。同様に、ノード 16 のバックプレーン（図示せず）は、論理ソルバー 50 が互いにローカルに通信して、これらのデバイスによって実装される安全機能を協調させること、データを互いに通信すること、及び / または他の統合機能を行うことを可能にする。一方で、MPD 70 は、プロセスプラント 10 の非常に異なる場所に配置された安全システム 14 の一部分が、依然として、互いに通信して、プロセスプラント 10 の異なるノードにおいて、協調させた安全動作を提供することを可能にする。具体的には、MPD 70 は、バス 74 と連動して、プロセスプラント 10 の異なるノード 12 及び 16 と関連付けられた論理ソルバー 50 が、共に通信的にカスケードされて、指定された優先度に従ってプロセスプラント 10 内の安全関連の機能をカスケードできるようにすることを可能にする。MPD 70 及びバス 74 は、ネットワーク 30 の代替となる通信リンクを安全システムに提供する。

20

【0025】

代替的に、プロセスプラント 10 内の異なる場所の 2 つ以上の安全関連の機能は、MPD 70 及び通信ライン 74 の使用を通してプラント 10 の別個の領域またはノード内の個々の安全フィールドデバイスまで専用のラインを引くことを必要とすることなく、インターロックまたは相互接続することができる。換言すれば、MPD 70 及びバス 74 の使用は、安全エンジニアが、事実上プロセスプラント 10 全体を通して分配されるが、異なる安全関連のハードウェアが必要に応じて互いに通信することを可能にするように通信的に相互接続された該ハードウェアの異なる構成要素を有する、安全システム 14 を設計し、構成することを可能にする。この特徴はまた、追加的な安全論理ソルバーが必要とされたときに、または新しいプロセス制御ノードがプロセスプラント 10 に加えられたときに、該追加的な安全論理ソルバーを安全システム 14 に追加することを可能にするという点で、安全システム 14 のスケーラビリティも提供する。論理ソルバー 50 は、典型的に、1 つ以上の原因結果マトリックス（CEM）によって定義される安全論理を実装する制御論理を含むことが理解されるであろう。

30

【0026】

図 2 は、例示的なワークステーション 18a（ワークステーション 20a は、同じまたは類似のデバイスを備えることができる）の構造を概略的に例示するブロック図である。ワークステーション 18a は、少なくとも 1 つのプロセッサ 100 と、揮発性メモリ 104 と、不揮発性メモリ 108 とを含むことができる。揮発性メモリ 104 は、例えば、ランダムアクセスメモリ（RAM）を含むことができる。いくつかの実施形態において、RAM は、停電が生じた場合にデータが失われないように、1 つ以上のバッテリによってバ

40

50

ツクアップすることができる。不揮発性メモリ 108 は、例えば、ハードディスク、リードオンリーメモリ (ROM)、コンパクトディスクROM (CD-ROM)、プログラマブルROM (PROM)、消去可能プログラマブルROM (EPROM)、電気的消去可能プログラマブルROM (EEPROM)、デジタル多用途ディスク (DVD)、フラッシュメモリ等のうちの 1 つ以上を含むことができる。ワークステーション 18a はまた、ワークステーション I/O デバイス 112 も含むことができる。プロセッサ 100、揮発性メモリ 104、不揮発性メモリ 108、及びワークステーション I/O デバイス 112 は、アドレス / データバス 116 を介して相互接続することができる。ワークステーション 18a はまた、少なくとも 1 つの表示デバイス 120 及び少なくとも 1 つのユーザ入力デバイス 124 も含むことができ、これらは、例えば、キーボード、キーパッド、マウス、トラックボール、タッチスクリーン、ライトペン等のうちの 1 つ以上とすることができる。いくつかの実施形態において、揮発性メモリ 104、不揮発性メモリ 108、及びワークステーション I/O デバイス 112 のうちの 1 つ以上は、別個のバスを介してアドレス / データバス 116 (図示せず) からプロセッサ 100 に結合することができ、または直接プロセッサ 100 に結合することができる。

【0027】

表示デバイス 120 及びユーザ入力デバイス 124 は、ワークステーション I/O デバイス 112 と結合される。加えて、ワークステーション 18a は、ワークステーション I/O デバイス 112 を介して、ネットワーク 30 に結合される。ワークステーション I/O デバイス 112 は、図 2 において 1 つのデバイスとして例示されているが、複数のデバイスを備えることができる。加えて、いくつかの実施形態において、表示デバイス 120 及びユーザ入力デバイス 124 のうちの 1 つ以上は、アドレス / データバス 116 に、またはプロセッサ 100 に直接結合することができる。

【0028】

以下、図 1 及び 2 を参照すると、制御ノード 12 及び 16 のうちの 1 つ以上と関連付けられたプロセス制御構成アプリケーション 19 は、1 つ以上のワークステーション 18a 及び 20a に記憶され、それらによって実行することができる。例えば、プロセス制御構成アプリケーション 19 は、不揮発性メモリ 108 及び / または揮発性メモリ 104 に記憶され、プロセッサ 100 によって実行することができる。しかしながら、所望であれば、このアプリケーションは、プロセスプラント 10 と関連付けられた他のコンピュータに記憶され、実行することができる。全般的に言えば、プロセス制御構成アプリケーション 19 は、プログラマ、制御エンジニア、または他の人員が、コントローラ 12a、16a、I/O デバイス 24、及び / またはフィールドデバイス 22、23 によって実装される制御ルーチン、制御モジュール、機能ブロック、プログラム、論理等を作成し、構成することを可能にする。これらの制御ルーチン、制御モジュール、機能ブロック、プログラム、論理等は、次いで、ネットワーク 30 を介して、コントローラ 12a、16a、I/O デバイス 24、及び / またはフィールドデバイス 22、23 のうちの適切なものにダウンロードすることができる。

【0029】

同様に、安全システム 14 と関連付けられた安全システム構成アプリケーション 21 は、ワークステーション 18a 及び 20a のうちの 1 つ以上に記憶され、それらによって実行することができる。例えば、安全システム構成アプリケーション 21 は、不揮発性メモリ 108 及び / または揮発性メモリ 104 に記憶され、プロセッサ 100 によって実行することができる。しかしながら、所望であれば、このアプリケーションは、プロセスプラント 10 と関連付けられた他のコンピュータに記憶され、実行することができる。全般的に言えば、安全システム構成アプリケーションは、プログラマ、安全エンジニア、または他の人員が、論理ソルバー 50 及び / またはデバイス 60、62 によって実装される安全関連の制御ルーチン、安全論理モジュール、機能ブロック、プログラム、論理等を作成し、構成することを可能にする。これらの制御ルーチン、安全モジュール、機能ブロック、プログラム、論理等は、次いで、ネットワーク 30 を介して、コントローラ 12a、16a、I/O デバイス 24、及び / またはフィールドデバイス 22、23 のうちの適切なものにダウンロードすることができる。

10

20

30

40

50

a、論理ソルバー 50、及び／またはフィールドデバイス 60、62 のうちの適切なものにダウンロードすることができる。

【0030】

安全システムは、典型的に、国際電気標準会議（IEC）61131-3 規格によって定義されたいいくつかの言語のうちの1つでプログラムされ、いくつかの事例において、安全論理は、一連の相互接続された機能ブロックまたは他のルーチンで構成することができる。プログラミング言語に関係なく、開始点は、通常、制御及び／または安全アクションの要求を指定する注釈文書である。安全システムにおいて、安全要求は、安全要求仕様（SRS）に文書化されている。下でさらに詳細に説明されるSRSは、プレーンテキスト、論理図、または特性要因図（原因結果マトリックスとも呼ばれる）のいずれかによって表すことができる論理記述を提供することができる。原因結果マトリックス（CEM）は、単純な視覚的表現で安全システムによって提供される安全アクションの概要である。したがって、CEMは、安全論理によって実装される基本的な原因結果関係を定義し、また、安全論理を構成するためのベースである。

【0031】

図3は、任意のタイプの表示デバイスを介して表示することができる、CEM300の1つの例示的な表現を例示する。具体的には、表示デバイスは、安全構成アプリケーション21と関連付けられたユーザインターフェースの一部とすることができます、表示は、例えばワークステーション18aの表示デバイス120を介して、プログラマまたは管理者に提示することができる。従来のプロセス制御システムにおいて利用することができるCEMである例示的なCEM300は、複数の原因及び複数の結果を含む。CEMの原因是、一般に、安全要求仕様によって定義され、また、プロセスプラント10の全体を通して、論理ソルバー50、フィールドデバイス22、23、60、及び62等によって、またはそれらにおいて示される、測定される、または検出される条件に関連する。CEM300において定義される異なる原因C1、C2等は、CEM300の各行と関連付けられる。例えば、1つの原因是、プラントの特定の領域の温度が安全または予め定義された範囲外である、センサ示度であり得る。

【0032】

原因に対応する条件が生じたときに、結果をトリガーすることができ、結果は、プラントで行われるべきアクションとすることができます。CEM300の異なる結果E1、E2等は、CEM300の各列について定義され、かつ該列と関連付けられる。例えば、CEM300の1つの結果（例えば、E3）は、弁を閉じる、アラームを鳴らす等の、プラントで行われるべき安全アクションに関連させることができます。特定の原因（例えば、C2またはC6）が特定の結果（E3）をトリガーしたときには、対応する原因結果の対または関係が存在する。

【0033】

CEM300において、原因結果関係は、各セルにおいて「X」で示され、これは、セルの行と関連付けられた原因によって、セルの列と関連付けられた結果がトリガーされることを示す。これらの関係は、本明細書で原因・結果の対と称することができる。代替の実装例において、セルは、関連付けられた原因結果をどのように関連させることができるのかをより正確に示す、様々な「トリガー」によってポピュレートすることができる。例えば、トリガーは、原因を受信した場合に即時に結果が起動されることを示す「X」、原因を受信した場合に、時間遅延を伴って結果が起動される「T」、原因を受信した場合に結果を許容することを示す「P」、等の形態とすることができます。さらに、空白セルは、特定の原因／結果の対がマトリックスにおいて現在関連していないことを示すことができ、したがって、プラントにおいて起動し得ない（すなわち、原因の発生が、いかなる結果とのトリガー関係も有しない）。

【0034】

例示的なCEM300は、7×7のマトリックスであり、これは、プロセスプラントの典型的な原因結果マトリックスよりも小さくなり得るが、例示のために単純な形態で示さ

10

20

30

40

50

れる。例示的な C E M 3 0 0 は、一組の対応するセルの各々において「 X 」で示される 10 個の原因 / 結果関係を含む。例えば、原因 2 (C 2) は、結果 3 、 4 、及び 5 (E 3 、 E 4 、及び E 5) にそれぞれ対応する各セルに「 X 」を含む。したがって、原因 2 (C 2) の関連付けられたイベントが生じた場合、結果 3 、 4 、及び 5 (E 3 、 E 4 、及び E 5) のそれぞれのアクションは、プラントの安全論理モジュールによって、プロセスプラント内でトリガーすることができる。しかしながら、いくつかの実施形態において、結果 3 、 4 、及び 5 の各々はまた、トリガーされる前に、他の関連付けられた原因が生じることも必要とし得る。例えば、システムで使用される論理に応じて、結果 4 は、結果 4 がトリガーされる前に (すなわち、結果 4 が、原因 2 、 3 、 4 、及び 5 の各々について「 X 」を有するので) 、原因 2 、 3 、 4 、及び / または 5 のうちの 1 つ以上を起動することを必要とし得る。したがって、 C E M によって定義される論理は、「 O R 」論理 (すなわち、結果列における任意の 1 つの原因の発生が、結果の開始をもたらす) に基づくことができ、または「 A N D 」論理 (すなわち、安全論理によって結果がトリガーされる前に、結果列にあらゆる原因が存在しなければならない) に基づくことができる。

【 0 0 3 5 】

別の実施形態において、結果 (結果 4 等) は、どの原因が生じるのかに依存する、異なる様態でトリガーすることができる。例えば、 1 つの関連付けられた原因が生じた場合は、結果 (結果 4 等) を遅延してトリガーすることができ、一方で、 2 つ以上の関連付けられた原因が生じた場合は、結果 (結果 4 等) を即時にトリガーすることができる。さらに、いくつかの関連付けられた原因は、自動トリガーを起動させることができ、一方で、他の原因は、結果 4 等の結果の遅延トリガーを起動させることができる。さらにまた、いくつかの関連付けられた原因は、他の関連付けられた原因とは独立に結果をトリガーすることができ、一方で、他の関連付けられた原因は、それらが 1 つ以上の他の原因と組み合わせて存在するときにのみ結果をトリガーすることができる。提供される実施例は、限定することを意図するものではなく、対応する原因結果の対によって、論理及び / または遅延の任意の組み合わせを実装することができる。

【 0 0 3 6 】

より詳細に論じられるように、 C E M によって定義される論理は、 C E M によって定義される原因結果のサブセットに実装される、多数の一組またはグループの論理に分けることができ、これらの論理の異なるサブセットは、安全論理実装の特定の機能ブロックによって実装することができる。例えば、機能ブロックは、 C E M 3 0 0 に例示される選択された論理ブロック 3 0 5 及び 3 1 0 によって定義される論理を実装するために使用することができる。この事例において、論理ブロック 3 0 5 は、 2 つの原因入力 (C 2 及び C 3) を含み、また、 3 つの結果出力 (E 3 、 E 4 、及び E 5) に対応する。この例示的な実施形態において、論理ブロック 3 0 5 及び 3 1 0 によって実装される論理のサブセットは、単に、 C E M 3 0 0 のポピュレートされたセルのクラスタを認識することによって識別される。ここで、論理ブロック 3 0 5 及び 3 1 0 は、 C E M 3 0 0 の 4 9 個のセルのうちの 1 2 個のみをカバーしている間、 C E M 3 0 0 によって示される重要な情報 (原因 / 結果関係) の大部分を含む。他の実施形態において、論理ブロック 3 0 5 及び 3 1 0 は、より大きくして、及び / または別の論理ブロックの論理を加えて、または識別して、論理ブロック 3 0 5 及び 3 1 0 に含まれていない C E M 3 0 0 の残りのポピュレートされたセルを含むことができる。下でさらに詳細に論じるように、 C E M は、より良好な、またはより効率的な態様で、ポピュレートされたセルをクラスタ化するように再配設し、それによって、論理ブロックを識別することを支援し、次に、機能ブロックを作成する。このクラスタ化は、所与の C E M 3 0 0 の些細な課題のように見え得るが、人間が、数百 (または数千) のセルを伴う C E M のパターンを効率的に識別することはほとんど不可能であり得る。

【 0 0 3 7 】

従来のシステムにおいて、 C E M は、状態機械の機能ブロックによって表され、原因が入力であり、結果が出力である。典型的に、状態機械の機能ブロックは、 C E M の結果ご

とに作成される。その結果、状態機械の機能ブロックは、それらの定義されたサイズによって使用が限定されず、よって、広範囲に増殖させることができる。しかしながら、従来のシステムとは異なり、本システムは、CEMを2つのタイプの機能ブロックに、すなわち、論理の複雑さを低減させ、また、複雑な、または大きいCEMを実装するときに安全システム内の論理実装の最適化を高める役割を果たす、監視ブロック及び結果ブロックに組織化する。

【0038】

より具体的には、別個の監視ブロック及び結果ブロックを使用して、対応するCEMに定義される論理の任意のパターンまたはグループを実装することで、結果から原因を2つの異なるカテゴリのブロックに分けることによって、従来のシステムの欠点に対処する。一般に、監視ブロック(MB)は、原因の抽象的表現であり、結果ブロック(EB)は、結果の抽象的表現である。そのため、システムは、1つ以上の結果ブロックにリンクまたは接続された1つ以上の監視ブロックによって、大きいCEMならびにその原因及び結果を表すことができる。例えば、一組の監視ブロックの出力は、1つ以上の結果ブロックへの入力としての役割を果たすことができ、故に、各結果ブロックの入力は、1つ以上の監視ブロックからの出力として起こり得る。一実施形態において、監視ブロックの出力は、代替的に、または加えて、1つ以上の他の監視ブロックへの入力としての役割を果たすことができる。その結果、監視ブロック及び/または結果ブロックは、最適に任意の所望のCEM論理を実装するために、所望に応じて、連鎖化、入れ子化、階層化、及び/またはレベル化することができる。さらに、CEMを複数のMB及びEBとして表す(及び実装する)ことは、安全システムのより簡単な実装及び保守を可能にし、さらに、より複雑なCEM関係を容易に表し、構成することを可能にする。

10

20

30

【0039】

別個の監視ブロック及び結果ブロックを作成することには、多くの利点がある。具体的には、MB及びEBは、所望に応じてサイズ決定することができ、これは、よりエラーを起こし難い、より迅速でより簡単な実装につながる。また、CEM論理を実装するためにこれらより小さくサイズ決定されたMB及びEBを機能ブロックとして使用する制御または安全システムは、原因の関係を透過的に表すため、より容易に試験し、トラブルシューティングする(または一般に、リバースエンジニアリングする)ことができる。さらに、大きいCEMは、より管理し易いサイズの論理ブロックに分けることができる。さらにまた、複雑な原因結果関係が、別個のMB及びEBを使用することによって、より容易に表される。例えば、階層化、ループ化、入れ子化、連鎖化等は、全て、別個の監視及び結果ブロックを使用して表すことができる。

【0040】

図4は、一組の相互接続した監視ブロック及び結果ブロックの概略図400である。図4の一組の監視及び結果ブロックは、図3のCEM300に対して提供または定義される情報または論理の全てを含む(実装する)。ここで、監視ブロック405及び410は、一般に、論理ブロック305及び310の原因(C2~C5)に対応し、一方で、結果ブロック415及び420は、一般に、論理ブロック305及び310の、またはそれらと関連付けられた結果(E3、E4、E5、E6)に対応する。例えば、監視ブロック1(MB1)405は、CEM300の原因2及び原因3を入力として含む。しかしながら、MB1 405の出力は、(論理ブロック305のように)CEM300の結果3、4、及び5に直接対応しない。通常は直接的な原因結果関係を実装する論理ブロック305及び310の論理を実装するために作成される状態機械とは対照的に、監視ブロックは、入力(他の監視ブロックからの原因及び出力等)及び出力(他の監視ブロックまたは結果ブロックに送信することができる)を含むことができるが、結果には直接対応しない。例えば、MB1 405の出力は、セット400の様々な他の監視及び結果ブロックに送信される。具体的には、監視ブロックMB1(405)の出力401は、結果ブロック1(EB1)415に送信され、監視ブロックMB1の出力402は、監視ブロック2(MB2)に410に送信され、監視ブロックMB1の出力403は、結果ブロック2(EB2)

40

50

420に送信される。

【0041】

MBの出力は、一般に、対応する入力に関する情報を提供する。例えば、出力401は、原因2に関する情報を提供する(MB1 405の対応するセルの「X」が、この関係を示す)。同様に、出力402は、原因2及び/または原因3に関する情報を提供する。例えば、出力402は、原因C2またはC3のいずれかが存在する(例えば、論理的に真である)場合に高く(論理1)になり得、または、出力402は、原因C2及びC3がどちらも存在するときにのみ高くなり得る。当然ながら、他の論理動作を原因C2及びC3に対して行って、専用のORing等の出力402を決定することができる。類似する様態で、MB2 410の出力411は、3つの入力(MB1 405の出力402、原因4、及び原因5)に関する情報を提供する。換言すれば、出力411は、(原因ブロックMB1の出力402を生成した論理によって定義される)原因2及び原因3、ならびに原因4及び原因5に関する情報を提供する。

【0042】

以下、結果ブロック415及び420を参照すると、結果ブロック1(EB1)415は、2つの入力、すなわち、監視ブロックMB1からの出力401(原因2の状態に依存する)、及び原因6(CEM300からの原因6に対応する)を受信する。結果ブロックEB1(415)のみが、1つの結果、すなわち、結果3に対応する。したがって、CEM300のように、結果ブロックEB1(415)は、原因2及び6を相関させて、結果ブロックEB1(415)の出力である結果3を作成する。理解されるように、結果ブロックEB1(415)は、任意の所望の論理を実装することができ、また、出力401(ここでも原因2の状態に関連する)の状態及び原因6(C6)の状態に基づいて遅延する。

【0043】

同様に、結果ブロックEB2(420)は、CEM300の結果4、5、及び6の状態を作成または定義する論理に対応し、それを実装する。結果ブロックEB2(420)の入力を対応する監視ブロックまでさかのぼることで、CEM300の結果4、5、及び6の原因結果関係が、結果ブロックEB2(420)によって達成されることが分かる。具体的には、結果ブロックEB2 420は、監視ブロックMB2に入力される原因4及び5に基づく、及び監視ブロックMB1の出力402に基づく、出力411を受信する。したがって、出力411は、結果ブロックEB2の結果4をトリガーするために使用される原因2、3、4、及び5から導出される値または状態を有する。さらに、結果ブロックEB2は、原因2及び4によって論理的に定義される、及び結果5をトリガーするためにいくつかの論理表現で使用される出力403及び412を受信する。またさらに、結果ブロックEB2は、原因4及び5に基づく論理値に対応する、または該論理値として定義された出力413を受信し、出力413を使用して、結果6をトリガーする。次に、図4の一组の監視及び結果ブロック400は、図3のCEM300に対して以前に提供または定義された関係情報(及び論理)の全てを含む。CEM300を一组の監視及び結果ブロック400に分解することによって提供される利点は、この実施例では明確でないかもしれないが、該利点は、より大きいCEMを分解するときに、より明らかになる。図4に表される一组の監視及び結果ブロック400は、単に一例を意味するものであり、監視ブロック及び結果ブロックは、CEMによって定義される論理を実装するために、無数のサイズ及び構成で作成し、構成することができることに留意されたい。

【0044】

図5は、表示デバイスによって表示することができ、また、CEMの論理またはCEMの一部分を実装する一组の監視及び結果機能ブロック表現する、または表す、構成画面500の説明図の1つの実施例である。構成画面500は、監視及び結果ブロックと関連付けられた論理を概略的に表すこと意図する一组の監視及び結果ブロック400とは対照的に、監視及び結果ブロックのより詳細な機能ブロックの実装例を表す。図5の実施例において、構成画面500は、入力(原因508、原因マスク512、及び論理タイプ506)と、図4の監視ブロックMB1(405)に対応する監視ブロック502と、図4の結

10

20

30

40

50

果ブロック E B 1 (4 1 5) に対応する結果ブロック 5 0 4 とを含む。

【 0 0 4 5 】

監視ブロック 5 0 2 は、2つの原因 5 0 8 、原因マスク入力 5 1 2 、及び論理タイプ 5 0 6 にそれぞれ対応する、4つの入力 (I N _ D 1 及び I N _ D 2 、 I N _ M A S K 及び L O G I C _ T Y P E) を受信する。論理タイプ 5 0 6 は、どのようなタイプの論理が現在の一組の監視及び結果ブロックに実装されているのかを定義する。一実施形態において、論理タイプは、正または負とすることができます。正論理は、原因の全てが最初に「偽」の状態で始まり、トリップされた場合に「真」になることを示すことができる。したがって、1つ以上の原因が「真」である場合、対応する出力は、「真」であり得る。次に、対応する結果ブロックは、1つ以上の「真」の入力を受信することができ、これは、結果ブロックの状態を高めることができ、及び / または結果ブロックをトリガーすることができる。負論理は、類似し得るが、原因が最初に「真」から始まり、原因が生じた場合に、「偽」に設定される。例示的な論理は、限定することを意図しておらず、論理タイプ 5 0 6 はまた、「 A N D 」論理、「 O R 」論理、または監視及び結果ブロックを実装する際に有用であり得る任意の他の論理も含むことができる。

【 0 0 4 6 】

原因マスク入力 5 1 2 は、監視ブロック 5 0 2 によって受信される原因 5 0 8 をフィルタリングするための初期パラメータを表すことができる。監視ブロック 5 0 2 はまた、監視ブロック 5 0 2 を構成するために使用される3つの構成マスク C F G _ M A S K 1 、 C F G _ M A S K 2 、及び C F G _ M A S K 3 5 1 0 も含み、各マスクは、どの原因が各出力に対応するのかを表し、及びいくつかの事例では、マスクされていない入力からの出力を発生させるために使用される論理を表す。構成マスク 5 1 0 は、下でさらに詳細に説明されるように、 C E M から導出される数値表現とすることができます。

【 0 0 4 7 】

監視ブロック 5 0 2 はまた、5つの出力 (O U T _ D 1 ~ O U T _ D 3 5 1 4 、 R A W _ V A L 5 1 6 、及び M A S K _ V A L 5 1 8) も含み、出力 5 1 4 のうちの1つ (O U T _ D 1) は、結果ブロック 5 0 4 への入力としての役割を果たす (図 4 の構成において識別される) 。生の値 5 1 6 は、単に、原因 5 0 8 の受信した値を出力することができ、一方で、マスクされた値 5 1 8 は、原因マスク 5 1 2 を適用した後に、原因 5 0 8 の値を出力することができる。さらに、 O U T _ D 1 ~ O U T _ D 3 は、図 4 の M B 1 4 0 5 の出力 4 0 1 ~ 4 0 3 に対応する。構成マスク 5 1 0 は、どの原因が各出力に対応するのかを示す。例えば、 (構成マスク 5 1 0 のうちの) C F G _ M A S K 1 は、「 A 」に設定され、これは、 (原因 5 0 8 のうちの) 原因 2 のみが (出力 5 1 4 のうちの) O U T _ D 1 に対応することを示すことができる。さらに、 (構成マスク 5 1 0 のうちの) C F G _ M A S K 2 は、「 B 」に設定され、これは、 (原因 5 0 8 のうちの) 原因 2 及び原因 3 が、 (出力 5 1 4 のうちの) O U T _ D 2 に対応することを示すことができる。またさらに、いくつかの事例において、構成マスク 5 1 0 は、16進数等の数値表現とすることができます、これは、どの監視ブロックの入力が特定の監視ブロックの出力を駆動または達成するのかを表し、及び / またはブロックの入力からブロックの出力を発生させるために使用される実際の論理を表す。

【 0 0 4 8 】

図 5 に例示されるように、結果ブロック 5 0 4 は、4つの入力 (I N _ D 1 及び I N _ D 2 5 2 0 、リセット 5 2 2 、及び L O G I C _ T Y P E 5 0 6) 及び2つの出力 (状態 5 2 6 及び O U T _ D 5 2 4) を含むことができる。結果ブロック 5 0 4 の入力 5 2 0 は、監視ブロック 5 0 2 の出力 5 1 4 、さらには、図 3 の C E M 3 0 0 の原因 6 を含む。結果ブロック 5 0 4 の状態 5 2 6 は、結果ブロック 5 0 4 に対応するデバイスの動作状態に対応することができる。換言すれば、いかなる対応する「真」の原因も受信しなかった場合、状態 5 2 6 は、正常であり得る。しかしながら、例えば「真」の値に設定された1つ以上の原因を受信する場合、状態は、新しい状態 (例えば、「警告」、「危険」、「トリガー済」) を示すように変化し得る。リセット入力 5 2 2 は、結果ブロック 5 0 4 が非

10

20

30

40

50

正常状態であるときに必要なアクションが行われた時点で、ユーザが、結果ブロック 504 の状態を正常へ自動的にリセットすることを可能にすることができる。当然ながら、（例えば、原因の入力または監視ブロックの入力の変化状態に応じて）許可のリセット等の、他の状態変化入力も同様に、結果ブロックに提供することができる。さらに、この実施例では論理タイプ 506 が正に設定されているので、受信した入力 520 のうちの 1 つ以上が「真」である場合、この事例において CEM300 の結果 3 に対応する出力 OUT_D は、トリガー済応答であり得る。

【0049】

一例として、図 5 は、原因 2 及び原因 3 の入力が「偽」であり、かつブロック 502 及び 504 両方の論理タイプが正に設定されているときの、（ブロック 502 の）OUT_D1、（ブロック 504 の）IN_D2、及び OUT_D の各々の状態を例示する。この時点で、プロセスプラントにおいて（原因 508 のうちの）原因 2 が起こった場合は、原因 2 の状態が「偽」から「真」に変化し得る。したがって、MB1_502 は、真として入力 IN_D1 を受信する。次いで、対応する出力 OUT_1 もまた、Logical_Type の入力での正論理タイプ及び OUT_D1（出力 1）の構成マスク、すなわち、CFG_MASK1 に基づいて、真に変化する。この実施例において、構成マスク 510 が適用された後に、IN_D1 は、出力（OUT_D1 ~ OUT_D3）514 の各々の値を駆動または達成する。具体的には、OUT_D1 は、「真」に設定されている IN_D1 に基づいて、「真」に設定することができる。したがって、次いで、EB1_504 が少なくとも 1 つの「真」の入力（IN_D1）を受信する。その結果、EB1_504 の状態 526 は、「トリガー済」に変化し、結果ブロック 504 の出力 524 の OUT_D は、図 4 の結果 3 がトリガー済または「真」となることを意味する、「真」に設定される。結果的に、結果がトリガーされ、任意の対応するアクション及び / またはアラームをプロセス制御プラントにおいてオフに設定することができる。

【0050】

図 6 は、表示デバイスによって表示することができ、また、監視及び結果ブロックの構成を表すことができる、構成画面 600 の説明図の別の実施例である。図 6 の実施例は、図 4 からの監視ブロック MB2 及び結果ブロック EB2 に重点を置く。例示の目的で、論理タイプ 606 は、負に設定され、これは、通常状態であるときに、全ての原因が「真」であることを意味する。負論理タイプにおいて、1 つ以上の入力が発生し、状態を「偽」に切り換えるときに、監視ブロックの対応する出力は、「偽」に設定することができ、これは、結果ブロックを受信し、次いで、結果を「真」に設定する結果をトリガーすることができる。

【0051】

さらに、結果ブロック EB604 は、時間遅延入力 608 を含む。この実施例において、結果ブロック 604 の入力 1（IN_D1）は、結果ブロック 604 の Delay_Time1 の入力に「20」が入力されたときに、出力 OUT_D を、20 秒の遅延（DELAY_TIME1）によってトリガーさせる。しかしながら、この実施例の入力 2（IN_D2）は、時間遅延（DELAY_TIME2）がゼロに設定されているので、出力（OUT_D）を、即時にトリガーさせることができる。この実施例は、限定することを意図しておらず、特定の結果ブロックに対して任意の数の遅延及び遅延時間を設定することができる。

【0052】

一実施例として、原因 4 が生じた場合は、原因 4 の状態（結果的に、IN_D2）が「偽」に変化する。図 4 を再び参照すると、原因 4 が、監視ブロック 410 の出力 411 ~ 413 の各々に対応する、またはそれらを達成することができる。したがって、この実施例において、原因 4（及び IN_D2）は、監視ブロック 602 の構成マスクによって実装される論理に従って、MB2 の全ての出力（OUT_D1 ~ OUT_D3）を駆動または達成することができる。具体的には、MB2_602 の OUT_D1 を「偽」に設定することができる。次に、EB2_604 の IN_D2 が「偽」として受信される。したが

10

20

30

40

50

つて、EB2 604のIN_D2に対応する時間遅延(DELAY_TIME2)がゼロの時間遅延に設定されたときには、EB2 604の対応する結果を即時にトリガーすることができる。一実施例として、図4のMB2 410の出力411がMB2 602のOUT_D1に対応する場合は、これらの出力のみが、EB 604においてOUT_D1として表すことができる(CEM300の)結果4を駆動する。したがって、この時点でEB604のOUT_D1をトリガーし、「真」に設定することができる。

【0053】

図5及び6において提供される例示的な監視及び結果ブロックは、実証の目的で単純にすることを意図する。例えば、監視ブロック502は、図5において4つの入力及び5つの出力を有するように示されているが、他の実施形態は、監視ブロックの機能に基づいて、任意の所望の数の必要な入力及び出力を含むことができる。1つの実施形態において、入力IN_Dx及び出力OUT_Dxの数は、一般に、再編成されたCEMの各論理ブロックの入力及び出力の数に対応する。さらに、システムは、1つの監視を実装して、複数の結果ブロック及び追加的な監視ブロックを駆動または達成することができるようマスクを構成することができる。次に、CEMは、層化、ループ化、入れ子化、連鎖化等を行うことができる複数の監視及び結果ブロックに分けることができ、これは、従来の状態機械の実装よりも大きい、プロセス制御プラントのシステムを構成するための柔軟性をシステムに提供することができる。

10

【0054】

図7は、プロセスプラントと関連付けられた監視ブロック及び結果ブロックを構成する例示的な方法700のフロー図である。方法700は、定期的に、及び/または構成エンジニアまたは他のユーザまたは他の安全論理デザイナーによる指示または開始信号等の、トリガーイベントに応答して実装することができる。方法700は、図1に関して論じられるプロセスプラント10等のプロセスプラントの1つ以上の構成要素を含むことができる電子デバイス(例えば、原因結果分析器ツール17)によって行うことができる。

20

【0055】

ブロック710で、電子デバイスは、CEMを受信する、または別様にはそこにアクセスすることができる。特定の実施形態では、論理ブロックを識別する前に、CEMを再配設して、グループのクラスタにおけるスパース性及び他の収集情報を除去することが有益であり得る。ブロック715で、電子デバイスは、自動的にCEMを再配設することができ、及び/またはユーザがCEMを再配設することを可能にすることができます。CEMを自動的に再配設するための方法は、下でさらに詳細に論じられる。ブロック720で、電子デバイスは、一組の監視ブロック及び結果ブロックを識別し、作成して、CEMの論理を実装することができる。ブロック730で、電子デバイスは、監視及び結果ブロックを、CEMを実装する安全または制御論理を設計することができる構成または安全論理エンジニア等のユーザに表示することができる。具体的には、電子デバイスは、表示デバイスに、グラフィカルユーザインターフェース(GUI)を表示させることができ、GUIは、第1の監視ブロック、第2の監視ブロック、及び結果ブロックを示すことができる。さらに、第1の監視ブロック、第2の監視ブロック、及び結果ブロックの各々は、第1の次元及び第2の次元を有するマトリックスで配設された複数のセルを示すことができ、第1の次元に沿った位置は、出力を示すことができ、第2の次元に沿った位置は、入力に対応することができ、よって、第1及び第2の次元に対する複数のセルの位置に基づいて、複数のセルが入力/出力対を定義することができる。

30

【0056】

ブロック740で、電子デバイスは、監視ブロック及び結果ブロックを構成して、またはユーザが該ブロックを構成することを可能にして、CEMの論理を実装することができる。一実施形態において、電子デバイスは、ユーザが、入力デバイスを介して構成データを入力することを可能にする。別の実施形態において、電子デバイスは、CEMを構文解析することによって、構成データを自動的に判定すること、または発生させることができる。実装例によれば、電子デバイスは、第1の監視ブロックの出力のうちの1つを、第2

40

50

の監視ブロックの入力のうちの 1 つとしての役割を果たすように構成することができ、第 1 の監視ブロックの出力のうちの追加的な 1 つ及び第 2 の監視ブロックの出力のうちの 1 つを、結果ブロックへの入力としての役割を果たすように構成することができ、ならびに / または第 1 の監視ブロック、第 2 の監視ブロック、及び結果ブロックの各々の複数のセルのうちの少なくとも 1 つを、それぞれのセルのそれぞれの入力 / 出力対と関連付けられた、及びプロセスプラントの条件に対応する、トリガーに指定することができる。

【 0 0 5 7 】

一実施形態において、監視ブロック及び結果ブロックを構成するために、電子デバイスは、追加的な入力 / 出力対を定義する追加的な複数のセルを有する少なくとも 1 つの追加的な監視ブロックを組み込むことができ、追加的な監視ブロックの少なくとも 1 つの出力を、第 1 の監視ブロック、第 2 の監視ブロック、及び結果ブロックのうちの少なくとも 1 つへの入力としての役割を果たすように構成することができ、また、追加的な複数のセルのうちの少なくとも 1 つを、それぞれの追加的なセルのそれぞれの追加的な入力 / 出力対と関連付けられた、及びプロセスプラントの追加的な条件に対応する、追加的なトリガーに指定することができる。別の実施形態において、監視ブロック及び結果ブロックを構成するために、電子デバイスは、追加的な入力 / 出力対を定義する追加的な複数のセルを有する少なくとも 1 つの追加的な結果ブロックを組み込むことができ、追加的な結果ブロックの少なくとも 1 つの入力を、第 1 の監視ブロックまたは第 2 の監視ブロックのうちの 1 つの出力に対応する構成するように構成することができ、また、追加的な複数のセルのうちの少なくとも 1 つを、それぞれの追加的なセルのそれぞれの追加的な入力 / 出力対と関連付けられた、及びプロセスプラントの追加的な条件に対応する、追加的なトリガーに指定することができる。

10

【 0 0 5 8 】

加えて、一実施形態において、監視ブロック及び結果ブロックを構成するために、電子デバイスは、第 1 の監視ブロック及び第 2 の監視ブロックの各々の入力を構成することができ、第 1 の監視ブロック及び第 2 の監視ブロックのうちの少なくとも 1 つの入力マスクを、該入力マスクが、第 1 の監視ブロック及び第 2 の監視ブロックのうちの少なくとも 1 つの入力と論理的に関連付けられるように構成することができ、トリガーのうちの少なくとも 1 つを、関連付けられた結果を時間遅延を伴って起動させるために、時間遅延トリガーに指定することができ、ならびに / またはトリガーのうちの少なくとも 1 つを、許容トリガーに指定することができる。

20

【 0 0 5 9 】

ブロック 750において、電子デバイスは、構成した監視ブロック及び結果ブロックを記憶することができる。具体的には、電子デバイスは、第 1 の監視ブロック、第 2 の監視ブロック、及び結果ブロックと関連付けられたコンピュータ可読媒体に構成データを記憶することができる。一実施形態において、電子デバイスは、第 1 の監視ブロック、第 2 の監視ブロック、及び結果ブロックの各々の複数のセルを表示デバイスにさらに表示することができ、また、それぞれの複数のセル内のそれぞれのトリガーを示すことができる。

30

【 0 0 6 0 】

当然ながら、方法 700 は、任意の数の方法で共に接続された任意の数の監視及び結果ブロックを作成して、これらの相互接続した監視及び結果ブロックを使用して CEM の論理を実装することができる。各監視ブロックは、任意の数の CEM の原因またはその任意のサブセットを、該監視ブロックへの入力として含むことができ、また、他の監視ブロックの出力に結び付けることができ、それによって、カスケード型監視ブロックを達成することができる。さらに、任意の結果ブロックは、一組の入力から 1 つ以上の結果を判定することができ、また、監視ブロックの出力及び / または任意の原因の入力のいずれかを入力として受信することができる。またさらに、方法 700 で、様々な監視ブロックならびに他の監視ブロック及び結果ブロックを相互接続することができ、またはユーザが該ブロックを相互接続する（すなわち、該ブロック間の接続を定義する）ことを可能にする。そのため、各監視ブロックは、原因信号（直接的な、または別の上流の監視ブロックに入力

40

50

された原因信号から生じた別の中間論理信号の形態での、監視ブロックへの入力)の1つ以上に基づいて、1つ以上の中間論理条件または信号を判定する論理を含む。同様に、各結果ブロックは、該結果ブロックへの一組の入力に基づいて、1つ以上の結果信号を生成し、そのような入力は、監視ブロックの1つ以上から出力される原因信号及び/または中間論理信号である。この様態で、方法700は、中間論理信号を、原因信号のいくつかの論理組み合わせを表す1つ以上の監視ブロックにおいて生じさせることを可能にし、また、この中間論理信号を1つ以上の結果ブロックへの入力として提供または使用することを可能にし、それによって、結果信号を作成するために結果ブロックによって実装される構成、サイズ、及び論理を簡単にする。

【0061】

より小さいCEMの場合は、方法700のブロック715で、パターンを識別することによって、または関連する原因及び結果をグループ化しようとすること等によって、安全エンジニアが、CEMを手動で再配設及び/または構成することを可能にすることができる。そのような再配設は、ユーザによって、グラフィカルユーザインターフェースを介して、手動で実装することができ、CEMの周囲の様々な行及び/または列を移動させて、または再配設して、原因結果関係(例えば、Xの印が付されたセル)が互いに近づくよう、またはより密度の高いグルーピングを形成するように定義するセルをグループ化する。しかしながら、より大きいCEMを再編成するには多数の方法があり、再編成するための最良のオプションを識別することが有益である。故に、プロセス制御システムと関連付けられたCEMを動的かつ自動的に分析し、再編成する機会が存在する。

【0062】

一実施形態において、システム(すなわち、図1のコンピュータシステム)は、一組の規則に基づいて大きいCEMを自動的に再編成するために、原因結果分析器ツール17を実装することができる。一実施形態において、規則31は、図1の構成データベース32に記憶することができ、ならびに/またはワークステーション18a及び/もしくは20aのユーザインターフェースを介して受信することができる。分析器ツール17は、CEMを分析して、一組の規則31を考慮してCEMの最適な、または最適化された構成(すなわち、CEMを再配設して、一組の監視及び結果ブロックを生成する最良の様態)を決定する。一組の規則31は、特定のプロセスプラントの現在のニーズまたは構成に基づいて、エンジニアによって指定するか、または別様には、分析器ツール17等のコンピュータによって自動的に発生させることができる。例えば、一組の規則31は、この論理が実装されるべき対応する論理ソルバー50、MPD70、ならびに/またはフィールドデバイス22、23、24、60、及び62に基づいて、CEMを、特定の原因及び/または結果が共にグループ化された状態で、グループに構成すべきであることを示すことができる。さらに、一組の規則31は、特定のパターンに基づいて、システムの効率に基づいて、及び/または他の基準に基づいて、CEMを再編成して、スペース性を除去するべきであることを示すことができる。別の実施形態において、一組の規則31は、特定の原因及び/または結果(または原因及び/または結果のグループ)を移動させるべきではないことを示すことができる。さらに別の実施形態において、一組の規則31は、再編成する必要がある特定の原因及び/または結果の重みを示すことができ、該重みは、異なる結果につながる多数の規則を適用しようとするときのコンフリクトを解決するために使用される。

【0063】

一実施形態において、一組の規則31は、CEMが、特定の数のグループ及び/または特定のサイズのグループに再編成されるべきであることを示すことができる。一組の規則31は、グループが構成されるべき様態をさらに示すことができる。例えば、一組の規則31は、各集団が、特定の数の、特定の最大数の、または特定の最小数の原因及び/または結果を含むべきであることを示すことができる。一実施形態において、一組の規則31は、グループが、重複する原因及び/または結果を含むべきではないことを示すことができる。規則31はまた、例えば、これらの原因が特定の論理ソルバーによって、もしくは

10

20

30

40

50

特定のノードにおいて検出されるので、または結果を特定の論理ソルバーにおいて特定のノードによって実装することが必要であり得るので、特定の原因または結果を共にグループ化すべきであることを指定することもできる。いずれにしても、CEMが再編成されると、一組の規則31は、さらに、エンジニアがCEMの特定の原因及び/または結果を手動で構成することを可能にすることができる。代替の、または追加的な規則が想定されることを認識されたい。

【0064】

一実現形態において、分析器ツールは、一組の規則31を受信すること、または発生させることができ、該一組の規則は、プロセスプラントの特定の領域に対応する特定の原因及び/または結果のみを再編成すべきであること、またはこれらの原因及び結果を共に、またはグループとして再編成すべきであることを示す。同様に、一組の規則31は、CEMの原因及び結果の特定のサブセットのみを再編成すべきであることを示すことができる。分析器ツールはまた、特定の行及び/または列を「ロックする」一組の規則31を受信して、または発生させて、受信した行及び/または列が再編成中に移動することを防止することもできる。またさらに、分析器ツールは、一組の規則31を受信すること、または発生させることができ、該一組の規則は、正論理（すなわち、原因が「オン」である場合に結果を起動する）に対応する原因を共にグループ化すべきであること、及び負論理（すなわち、原因が「オン」である場合に結果を起動しない）に対応する原因を共にグループ化すべきであることを示す。CEMセルにおいて定義される論理のタイプ（すなわち、実装されるべき論理のタイプ）に基づいてCEMの行及び列をグループ化または再編成する他の様態も、同様に使用することができる。

10

20

30

【0065】

次に、CEMの再編成は、コンピュータによって実装することができ、かつ一組の規則31に基づくことができるマルチパート分析を必要とし得る。コンピュータは、行によって、列によって、グループによって、トリガーによって、対応する論理ソルバー50、MPD70、ならびに/もしくはフィールドデバイス22、23、24、60、及び62に基づいて、または一組の規則31の実装に最良に適している任意の他の要素によって、CEMを分析することができる。例えば、図8は、以前の例示的な図3のCEM300よりもかなり大きい例示的なCEM800である。CEM800は、マトリックスの全体を通して散在するいくつかのポピュレートされたセルを含む。CEM800は、CEM300よりもわずかにより大きいだけであるが、一組の監視及び結果ロックによって実装されるCEM800の論理ロックまたは論理グループを識別する問題が、ますます複雑であることは明白である。さらに、CEM800は、より大きいグループングから散在された、ポピュレートされたセルを含み、これは、監視及び結果ロックを発生させるために使用される論理ロックを効率的に選択する難しさを増加させる。増加したサイズのCEMの場合、論理ロックを手動で選択する、または定義する難しさが非常に増大する。

【0066】

図9は、図8からのCEM800を再編成したバージョンであるCEM900を表す表示の1つの実施例を例示する。図9に例示されるように、CEM900は、3つの主グループまたは論理ロック901、902、及び903を含むように構成されている。例示的な一実施形態において、論理ロック901、902、及び903は、それぞれ、プロセスプラント内の特定の論理ソルバー50に対応することができる。別の実施形態において、コンピュータは、一組の規則31内で定義された基準に基づいて、論理ロック901~903を識別することができる。

40

【0067】

例えば、図9の論理ロック901は、全てがプロセスプラント（例えば、特定の加熱セクション）の特定の物理的な場所に属する、またはプラント制御システムの同じコントローラまたは論理ソルバーによって実装される、一組の結果に対応することができる。さらに、論理ロック902は、全ての原因が全ての結果に関連するグループを認識することによって、CEMからスパース性を除去する分析器ツールによって生じさせることができ

50

きる。具体的には、論理ブロック 902 で、原因 4 ~ 10 の各々は、結果 3 ~ 5 の各々と対になる。論理ブロック 903 は、負論理の原因結果関係のグループに対応することができる。例示的 CEM900 は、3 つの論理ブロックを含んでいるが、CEM は、任意の数の論理ブロックに分けることができ、また、CEM を分析し、再編成するときに分析器ツール 17 によって使用することができる規則 31 のうちのいずれか、または任意の組み合わせまたは規則 31 に、上で説明した、もしくは上で述べられていない任意の他の規則に基づくことができる。論理ブロック 901、902、及び 903 は、それぞれ、図 3 及び 4 に関して上で説明したように、一組の相互接続した監視及び結果ブロックを定義して、CEM のこれらの部分の論理を実装するために使用することができる。

【0068】

図 10 は、プロセス制御システムの安全または制御論理を開発する際に使用される CEM の原因結果マトリックスを再配設し、論理ブロックを定義及び / または管理する例示的な方法 1000 のフロー図である。方法 1000 は、例えば、プラントの構成中、論理の CEM が変更または更新されるたび、等のときに、定期的に、及び / またはトリガーイベントに応答して、実装することができる。方法 1000 は、図 1 に関して論じられるプロセスプラント 10 等のプロセスプラントの 1 つ以上の構成要素を含むことができる電子デバイス（例えば、図 1 の分析器ツール 17）によって行うことができる。ブロック 1010 で、電子デバイスは、一組の入力及び一組の出力（すなわち、一組の原因及び一組の結果）を有する初期原因結果マトリックスにアクセスする。実施形態において、一組の入力の各々は、プロセスプラント内の条件を表すことができ、一組の出力の各々は、プロセスプラント内で行われるべき結果を表すことができる。さらに、一組の入力及び一組の出力のうちの少なくともいくつかは、原因 - 結果の対として関連され、それによって、対応する結果を、対応する条件の発生に応答して起動させることができる。初期原因結果マトリックス（CEM）は、プロセス制御プラントのデータリポジトリに記憶することができ、またはプラントの新しいプロセスを構成するための電子デバイスにおいてユーザによって発生させることができる。初期 CEM はまた、プロセス制御システムの外部のデータベースから受信することもできる。いくつかの実施形態において、初期 CEM は、適切な証明書を有するエンジニアによってのみアクセスすることができ、したがって、初期 CEM へのアクセスを許可するために、ログインまたは他のパスワードが必要であり得る。

【0069】

電子デバイスは、初期 CEM 内の一組の関連するグループの各々を定義することができる。具体的には、ブロック 1020 で、電子デバイスは、一組の関連するグループと関連付けられた一組の規則 31 にアクセスすることができる。具体的には、電子デバイスは、プロセス制御システムの内部または外部のいずれかの 1 つ以上のデータベースを通して、一組の規則 31 にアクセスすることができる。電子デバイスはまた、プロセス制御プラントのエンジニアによって提供される入力として、一組の規則 31 を受信することもできる。さらに、一組の規則 31 は、様々なデータベース及び / または入力を通してアクセスされる様々な規則の組み合わせとすることができます。上でさらに詳しく論じたように、一組の規則 31 は、効率的かつ有効な様態で CEM を再編成することを目的とすることができる。

【0070】

1 つの実施形態において、規則は、一組の出力の指定された部分が、同じ関連するグループ内でなければならないことを指定することができる。別の実施形態において、規則は、一組の入力の一部分が、ある量に達しなければならないことを指定することができる。さらなる実施形態において、規則は、一組の入力も、一組の出力も、一組の関連するグループの間で重複してはならないことを指定することができる。当然ながら、任意の他の所望の規則を使用することができる。

【0071】

ブロック 1030 で、電子デバイスは、CEM において定義された対応する原因 - 結果の対によって定義される一組の規則に従って、一組の出力（結果）の一部分に関連する一

10

20

30

40

50

組の入力（原因）の一部分を識別することができる。さらに、ブロック 1040 で、電子デバイスは、対応する原因・結果の対の一部分が再配設されるように、一組の入力の一部分及び一組の出力の一部分を再配設することができる。ブロック 1040 は、この再配設を行って、上で定義されたように、一組の監視及び結果ブロックを使用して実装される 1 つ以上の機能ブロック論理ユニットを定義することができる。ブロック 1050 は、再配設された CEM を分析し、プロセスが完了したかどうかを判定することができ、完了していない場合は、制御をブロック 1030 に提供して、再配設された CEM に基づいて監視及び結果ブロックの作成をさらに最適化することを目指して CEM を再配設するために使用される他の規則を識別することができる。さらに、ブロック 1050 は、再配設が完了したときに、図 9 の 3 つの論理グルーピング 901、902、及び 903 等の、再配設された CEM 内の論理ブロックまたは論理のグループを定義することができる。

【0072】

一実現形態において、電子デバイスは、ブロック 1050 によって定義される一組の関連するグループに従って、プロセス制御システムの 1 つ以上の機能ブロック論理ユニットをさらに構成することができる。加えて、または代替的に、電子デバイスは、図 11～12 に関して下でさらに詳細に論じられる、関連するグループの 16 進表現を算出すること等によって、再配設された原因・結果の対に従って、一組の関連するグループの各関連するグループについて、関連するグループの、または関連するグループの一部分の数値表現を自動的に算出することができる。

【0073】

分析器ツールが CEM 900 を再編成すると、システムは、CEM 900 を別個の論理グループにさらに分けて、それらの論理グループを実装する監視及び結果ブロックを作成する際の効率をさらに改善することができる。図 11 は、図 9 の CEM 900 の追加的な表現を表す。具体的には、システムは、図 9 の CEM 900 を分析して、一組の相互接続した監視ブロック及び結果ブロックとして機能ブロックを構成するためにシステムが使用することができる、様々な数値表現 1101、1102、及び 1103 を生成することができる。一実施形態において、数値表現 1101～1103 は、それぞれ、再配設された CEM 900 によって定義される原因結果の対によって定義された論理関係の構成に基づいて、16 進値等の値として出力または結果を表す、または定義することができる。この数値は、論理表現として各列を表す従来のシステムとは対照的である。しかしながら、そのような従来のシステムは、論理表現を実装または理解することが難しいので、非効率的である。

【0074】

一実施形態において、システムは、マトリックスの各セルに 2 つの値（例えば、オンまたはオフ、1 または 0 等）のうちの 1 つを指定し、次いで、CEM の行または列の各ビットグループ（例えば、4 衍の 2 進数）を 16 進数に変換することによって、数値表現を算出することができる。例えば、図 11 に例示されるように、出力 14 の数値表現 1101 は、出力 14 と関連付けられたセルの 16 進表現（FE08）であり、ここで、セルの X は、バイナリの「1」として扱われ、空きセルは、バイナリの「0」として扱われる。この算出は、出力 14 を 4 つのビットグループに分け（セル間を太線で区切り、上から下へ：1111、1110、0000、1000 の 4 つのビット数を形成する）、次いで、各ビットグループを 16 進数に変換することによって示すことができる。この事例において、16 進数字で、F = 1111、E = 1110、0 = 0000、及び 8 = 1000 である。出力 14 の数値表現 1101 は、FE08 である。同じ様態で、出力 5 は、上から下へ、ビット 0000、0111、1110、0000 に分けることができるので、数値表現 1102 (07E0) は、出力 5 に対応し、これは、16 進数の 07E0 に翻訳する。類似する様態で、出力 17 は、16 進数の 0072（数値表現 1103）として表すことができる。例示的な数値表現は、限定することを意図しておらず、列及び/または行のうちのいくつかまたは全てが数値表現を指定することができる。さらに、数値表現は、必ずしも 16 進変換である必要はなく、また、任意の他の適切な形態で作成することができ

10

20

30

40

50

る。

【 0 0 7 5 】

C E M内の論理セルの特定のグルーピングについて数値表現を案出することには、多くの利点がある。具体的には、従来のシステムと比較して、列を16進変換することは、より単純であり、表現を発生させるために追加的なゲートまたはプログラミングを必要とせず、記憶するために費やすメモリが少なく、そして、機能ブロック入力に通信するために費やす帯域幅が少ない。さらに、16進値入力は、所望に応じてより容易にエラーを補正して、精度を確保することができ、図15の試験マトリックスに関して下で論じられる。

【 0 0 7 6 】

数値表現は、システムが、安全システム構成環境の原因 / 結果関係を構成することをさらに可能にすることができます。具体的には、数値表現は、システムが、多数の原因及び結果の関係を定義することを可能にすることができます。さらに、数値表現は、全ての行及び / または列全体を单一の数値に取り出すことによって、構成エラーを排除することを補助することができる。加えて、数値表現は、原因 / 結果関係の変化を識別し、また、C E Mの変化を管理するために必要な労力をさらに低減させる、単純で効率的な方法を提供することができる。

【 0 0 7 7 】

例えば、数値表現は、図4～6の監視及び結果ブロック等の機能ブロックの構成マスクとして実装することができる。したがって、これらの数値表現は、監視及び / または結果ブロックの特定の結果について実装されるべき論理を実際に識別することができます。数値表現は、どの入力が各特定の出力に対応するのかを定義し、したがって、特定の出力に対応しない入力を切り離す（すなわち、マスクする）ことができる。例えば、図11の結果14の数値表現1101は、結果14から原因6～13の全てを切り離すことができる。換言すれば、監視ブロックは、原因1～16の全てを受信することができるが、数値表現1101をマスクとして実装した場合は、単に、原因1～5及び14～16を結果14に相關させる。

【 0 0 7 8 】

さらに、システムは、可能なセル値の範囲が2つを超えるときに（例えば、セルが、値なし、X、1T（時間遅延を示す）、P（原因の許容を示す）等の、多数の異なるトリガーを定義することができるときに）、数値表現を適合させることができます。例えば、例示的な4つの可能な交差値の範囲について、システムは、2つの16進変換を行って、結果として生じる数値表現を発生させることができます。換言すれば、各セルの4つの可能な異なる値は、2ビット数の4つの可能な値のうちの1つとして表すことができ、これは、各セルが、図11に示される1ビット値の代わりに、2ビット値によって定義されることを意味する。この事例において、各一組の文字列の2つの隣接するセルは、16進数に変換することができる4ビット値を形成する。その結果、このシナリオにおける数値表現は、図11に示される数値表現の2倍の長さになるが、C E Mを実装する論理において使用される、より多い数の潜在的な論理表現を表すことができるよう、より動的である。代替的に、システムは、ベース16以外の適切なベースを使用して数値表現を算出することができ、次いで、随意に、（所望であれば）数値表現を機能ブロック（すなわち、監視及び結果ブロック）の16進の入力値に変換することができる。

【 0 0 7 9 】

図12は、C E M内の値または要素の数値表現を作成 / 算出するための例示的な方法のフローチャートである。ブロック1210で、原因及び結果分析器ツール17は、C E Mにアクセスすることができる。例示的な一実施形態において、C E Mは、継続する前に再配設することができる。ブロック1220で、ツール17は、原因のサブセットを識別することができる。一実施形態において、原因のサブセットは、上で説明したように、特定の論理ブロックに属すること、及び / または一組の規則によって定義することができる。次に、ブロック1230で、ツール17は、原因のサブセットの单一の次元マトリックスを定義することができる。一次元マトリックスは、C E Mの特定の結果に対応するこ

10

20

30

40

50

できる。次に、ブロック 1240 で、ツール 17 は、一次元マトリックスの数値表現を算出することができる。上で説明したように、ツール 17 は、一次元マトリックスを 2 進文字列及び / または多数の 2 進文字列に変換することができる。一実施形態において、ツールは、次いで、1 つ以上の 2 進文字列の 16 進表現または任意の他の適切な数値表現への変換に進むことができる。算出した数値表現は、数値表現 33 として図 1 の構成データベース 32 等のリポジトリに記憶することができる。

【0080】

ブロック 1250 で、次いで、数値表現 33 を使用して、一組の機能ブロック（例えば、監視及び結果ブロック）を構成することができる。例えば、上で説明したように、数値表現 33 は、1 つ以上の監視ブロックの構成マスクとして実装することができる。

10

【0081】

本明細書で説明されるシステムの別の態様において、システムナビゲータアプリケーションは、プロセスプラントの関連する安全情報を提供する異なるユーザインターフェース画面の間を迅速にナビゲートする能力をユーザに提供する。そのような情報は、CEM、監視及び結果ブロック、安全文書、ならびにシステム構成表示において見つけることができる。いくつかの実施形態において、これらの異なるユーザインターフェースは、同じ安全論理の異なる視覚的表現を提供する。したがって、本発明は、インターリンクされた一組のユーザインターフェースの間をナビゲートするために、（図 1 の）ナビゲータツール 15 を提供する。例えば、図 13 は、インターリンクされた一組のユーザインターフェースの例示的な説明図 1300 である。

20

【0082】

いくつかの例示的なプロセスプラントにおいて、安全プロトコルは、いくつかの言語のうちの 1 つでプログラムされる。プログラミング言語に関係なく、安全プロトコルの開始点は、通常、プロセスプラントの制御及び / または安全アクションの要求を指定する注釈文書である。安全計装システム（SIS）等の、他の例示的なプロセスプラントにおいて、安全要求は、安全要求仕様（SRS）として知られている文書において文書化されている。

【0083】

SRS に関する情報のうちの 1 つは、識別された安全計装機能（SIF）のリストである。各 SIF は、特定の危険から保護し、また、定義されたレベルのリスク低減を提供する。SIS は、1 つ以上の SIF で構成される。いくつかの実施形態において、いくつかの安全システムは、個々の SIF の区別なく、SIS 構成において全ての SIF を組み合わせる。さらに、いくつかの安全システムは、SIF アプローチに従って、SIF ベースの SIS 構成を可能にする。

30

【0084】

SRS は、通常、異なるセクションを含む。セクションのうちの 1 つは、ブレーンテキスト、論理図、または特性要因図（すなわち、原因結果マトリックス）のいずれかによって表すことができる論理記述である。上で述べたように、いくつかの安全システムは、SIS 構成内の全ての SIF を組み合わせ、CEM の視覚化は、そのような実施形態を実装する際に非常に好都合であり得る。

40

【0085】

1 つの実施形態において、ナビゲータアプリケーション 15 は、エンジニアが、CEM 内の所与の原因（及び / または結果）を選択して、選択された原因（及び / または結果）を説明する特定の文書にナビゲートすることを可能にする。例えば、原因を選択することで、SRS 内の特定の SIF の説明にリダイレクトすることができる。この特徴は、エンジニアが、原因及び / または結果と関連付けられた特定の安全論理を見ることを可能にする。一実施形態において、エンジニアはまた、所与の SIF と関連付けられた安全モジュール（システム構成）を選択することもでき、次いで、CEM から適切な SIF を表示するユーザインターフェースにリダイレクトすることができる。さらに、エンジニアは、CEM の要素を選択して、CEM の特定の要素に関連するシステム構成強調表示デバイス、

50

論理ブロック、機能ブロック、監視及び結果ブロック等の表示にリダイレクトすることができる。安全または制御モジュールから、ユーザはまた、SRSまたは制御いずれかの注釈に関する適切なセクションにリダイレクトすることもできる。換言すれば、現在のシステムは、エンジニアが、CEM、SRS、またはシステム構成のビューの間をシームレスに切り換えることを可能にすることができます。

【0086】

例えば、エンジニアがCEM1310を表示する図13のユーザインターフェースの原因及び/または結果を選択した場合は、エンジニアを、CEMの選択された原因及び/または結果に関連する特定のデバイスを含むシステム構成を示す表示画面1320にリダイレクトすることができる。例えば、システム構成1320は、CEMの選択された原因及び/または結果に関連するタンク、弁、トランスマッタ、ポンプ、パイプ、センサ等のシンボルを含むことができる。この実施例では、温度計アイコン1321が強調表示されており、選択された原因または結果が温度センサの示度に対応することを示す。

【0087】

さらに、CEM1310またはシステム構成1320のいずれかから、エンジニアは、SRS1330等のプロセスプラントの安全プロトコルを説明する文書にアクセスすることができる。図13は、関連する安全プロシージャを説明するアイコン1331及びテキスト1332を含む安全要求仕様の一部分の例示的な表示1330を例示する。ナビゲーターアプリケーション15は、エンジニアが、表示の間をトグルすることを可能にし、以前はアクセスすることが煩雑であって情報及び洞察をエンジニアに提供する。さらに、インターフェース1310、1320、及び/または1330のうちの任意の1つから、ユーザは、(例えば、CEM1310の)関連する選択された要素を実装する論理を含む一組の監視及び結果ブロック(または他の機能ブロックもしくは論理)を表示するユーザインターフェース1340にアクセスすることができる。

【0088】

例示的な一実施形態において、エンジニアは、CEM1310(またはSRS1330)、システム構成1320、または監視及び結果ブロック1340の要素を右クリックして、ドロップダウンメニューにアクセスすることができる。ドロップダウンメニューは、他の表示ビュー(1310、1320、1330、及び1340等)及び/または(下の図16A~Dに関して説明されるような)他のビューのうちの1つにアクセスする能力を含むオプションをエンジニアに提供することができる。

【0089】

ユーザは、CEM1310から(またはCEM1310の個々のセル、原因、または結果から)またはシステム構成1320から、一般バイパス原理の定義、証明試験要求等の要求仕様(SRS1330)内の特定のセクション(複数可)まで、容易にナビゲートすることができる。

【0090】

この機能は、文書の構成と設計との間を行ったり来たりするシームレスな移行を提供して、構成の検証、変化の管理、トラブルシューティング、及び証明試験を容易にする。

【0091】

図14は、プロセス制御システムによって制御されるプロセスプラントの安全要求仕様(SRS)に含まれる情報へのアクセスを可能にする例示的な方法1400のブロック図を表す。方法1400は、サーバまたは別様には任意のタイプの電子デバイスによって容易にでき、サーバは、コンテンツを表示するように構成されたユーザインターフェースを備える、またはそれに接続することができる。SRSは、サーバによってアクセスすることができるメモリに記憶することができる。

【0092】

方法1400は、ブロック1410から始まり、そこで、サーバは、CEMをユーザインターフェースに表示することができる。実施形態において、(CEM)は、一組の原因及び一組の結果を含む一組の要素を含むことができ、一組の原因の各々は、プロセスプラ

10

20

30

40

50

ント内の条件を表すことができ、一組の結果の各々は、プロセスプラント内で行われるべき結果を表すことができる。さらに、一組の原因及び一組の結果のうちの少なくともいくつかは、原因・結果の対として関連され、それによって、対応する結果を、対応する条件の発生に応答して起動させることができる。

【0093】

ブロック1420で、サーバは、ユーザインターフェースを介して、一組の要素のうちの1つの要素の選択を受信することができる。具体的には、サーバは、一組の原因のうちの1つの原因または一組の結果のうちの1つの結果の選択を受信することができる。選択を受信することに応答して、ブロック1430で、サーバは、SRSから、一組の要素のうちの1つの要素と関連付けられた一組の情報にアクセスすることができる。具体的には、サーバは、SRSから、選択された原因または選択された結果と関連付けられた一組の情報にアクセスすることができる。実施形態によれば、サーバは、SRSから、選択された要素と関連付けられた配管及び計装図(P&ID)、選択された要素と関連付けられた安全計装機能(SIF)の説明、または他の情報にアクセスすることができる。

10

【0094】

ブロック1440で、サーバは、一組の情報をユーザインターフェースに表示することができる。一実施形態において、サーバはまた、ユーザインターフェースを介して、選択された要素と関連付けられた安全論理を表示するように構成されたアプリケーションを開始することもできる。加えて、一実施形態において、サーバは、ユーザインターフェースを介して、ユーザインターフェースに表示される一組の情報の一部分の追加的な選択を受信し、SRSから、一組の情報の一部分と関連付けられた追加的な一組の情報にアクセスし、そして、追加的な一組の情報をユーザインターフェースに表示することができる。さらに、一実施形態において、サーバは、ユーザインターフェースを介して、ユーザインターフェースに表示される一組の情報の一部分の追加的な選択を受信することができ、一組の情報の一部分は、CEMの一組の要素のうちの追加的な1つの要素に対応することができる、また、CEM及び追加的な要素の指示をユーザインターフェースに表示することができる。

20

【0095】

いくつかの実施形態において、大きいCEMは、数千の原因結果の対を含み得る。結果的に、これらの大きいCEMは、数百の監視ブロック、結果ブロック、及び数値表現に分けられ得る。多数のデータ構造にわたって散存する大量の情報のため、ユーザは、プロセス制御システムの安全論理が正確に実装されていることを手動で確認することが不可能であり得る。以前のプロセス制御システムは、構成したプロセス制御システムが必要な安全プロトコルを満たすことを厳格に検証するための手段が欠如していた。換言すれば、以前のシステムは、プロセスプラントの安全を管理するために実装されていたCEM及び機能ブロックの精度を試験する、いかなる様態も有していなかった。本開示は、プロセスプラントに現在実装されている安全論理を自動的に検証することができるツール(例えば、原因結果分析器ツール17)を提供する。

30

【0096】

1つの態様において、原因結果分析器ツール17は、プロセスプラント(またはその一部分)の構成を自動的に横断して、構築したままの、または構成したままのシステムの1つ以上の試験CEMを発生させることができる。一実施形態において、ツール17は、機能ブロック(すなわち、監視及び結果ブロック)、及びプロセスプラントの現在実装されている安全論理を表す数値表現に基づいて、リバースエンジニアリングを通して試験CEMを構築することができる。次いで、試験CEMと、要求定義CEM(プロセスプラントによって必要とされる安全論理の正確な表現であることが知られているCEM)とを比較することができる。この比較は、不一致または他のエラーを明らかにすることができ、次いで、それをユーザに提示することができる。

40

【0097】

図15は、原因結果マトリックスの安全論理を検証するための例示的な方法のブロック

50

図である。ブロック 1510 で、分析器ツール 17 は、原因結果マトリックスの安全論理を表す 1 つ以上の機能ブロックの構成を決定することができる。一実施形態において、機能ブロックは、上で説明したように、入力、出力、及び数値表現を含む監視及び結果ブロックである。原因結果分析器ツール 17 は、監視及び結果ブロック (M E B) の構成を決定するために、M E B に実装される論理及び / または M E B の数値表現等のいくつかの要因を考慮しながら、M E B の入力及び出力を横断することができる。例えば、ツール 17 は、上の図 5 及び 6 で説明したように、一組の監視及び結果ブロックを受信することができる。ツール 17 は、結果ブロックの出力から開始し、結果ブロックの入力から入力のソース (すなわち、結果ブロックに、及び / または 監視ブロックの出力に直接供給される原因) まで横断することができる。ツール 17 は、次いで、数値表現に基づいて、監視ブロックの出力を監視ブロックの対応する入力まで追跡することができる。ツール 17 は、各原因結果の対の各関係が識別されるまで、各結果について反復的に M E S を横断する、このプロセスを続けることができる。

【0098】

ブロック 1520 で、ツール 17 は、決定した構成に基づいて、試験 C E M を発生させることができる。ツール 17 は、監視及び結果ブロックの決定した構成に基づいて、識別した原因結果の対で試験 C E M をポピュレートすることができる。試験 C E M が作成されると、ツール 17 は、試験 C E M 37 をデータリポジトリ (図 1 の構成データベース 32 等) に記憶することができる。試験 C E M 37 は、本明細書で説明される C E M のいずれかとして実装することができる。

【0099】

ブロック 1530 で、ツール 17 は、要求定義 C E M にアクセスすることができる。1 つの実施形態において、要求定義 C E M 35 は、データリポジトリ (図 1 の構成データベース 32 等) に記憶することができる。他の実施形態において、ツール 17 は、プロセスプラント及び S R S のデバイスの現在の構成、ならびに他の安全文書に基づいて、要求定義 C E M を作成することができる。一実施形態において、要求定義 C E M 35 は、一組の原因及び一組の結果を含むことができ、原因結果の対の関係は、プロセスプラントの安全要求に基づく。要求定義 C E M 35 は、本明細書で説明される C E M のうちのいずれかとして実装することができる。

【0100】

ブロック 1540 で、ツール 17 は、試験 C E M 37 と、要求定義 C E M 35 とを比較して、任意の不一致があるかどうかを判定することができる。不一致は、試験 C E M 37 から要求定義 C E M 35 までの原因結果の対の間の任意の違いを含むことができる。例えば、原因結果の対は、同じトリガータイプ (例えば、許容、即時、遅延) 及び / または同じ論理タイプ (A N D / O R) によって相關していない場合がある。

【0101】

ツール 17 は、1 つ以上の判定した不一致のうちのいずれかを表示することができる。一実施形態において、ツール 17 は、図 13 及び 16 a ~ d において説明したようなユーザインターフェースのうちのいずれかに不一致を強調表示することができる。換言すれば、ツール 17 は、誤った論理が C E M 、監視及び結果ブロック、 S R S 文書、及び / またはシステム構成ユーザインターフェースに実装されている場所を強調表示することができる。

【0102】

図 13 及び 14 に関して上で論じた機能に加えて、プロセスプラントの 1 つ以上のデバイス、原因、及び / または結果の状態履歴を見ることも有益であり得る。本発明の別の態様は、プロセス制御システムの安全システムの状態を監視するためのユーザインターフェースビューを提供し、該状態は、典型的に、物理的設備及び / または安全試験結果に基づいているか、または「鍵が付され」、また、様々な原因または結果の現在及び過去の状態を示して、C E M 論理がプラントにいつ及びどのように実装されているのかを知るいくつかの能力をユーザに提供することができる。例えば、エンジニアは、設備の特定のデバイ

スもしくは一部、またはそのグループを表示する（図13に関して上で論じたような）ビューを提示することができ、次いで、さらに、表示ビューに示される、監視されるデバイス／設備の各部の現在及び／または過去の安全状態、原因信号等を提示する（例えば、図16A～D）、表示ビューにアクセスするために使用することができる。加えて、または代替的に、安全エンジニアは、プラントの設備の特定の物理的デバイス／一部に対して安全試験を行うことができ、そして、結果を表示ビューに表示することができる（例えば、図16A～D）。以前のシステムは、物理的デバイス／設備の多数の異なる表示ビューにアクセスすることを必要とすることなく、または特定の診断を行って試験結果を取得しなければならないことによって、プラントの（またはプラントの所望の領域の）全体的な安全状態をエンジニアが迅速に監視し、アクセスすることを可能にしなかった。以前のシステムは、エンジニアにとって不都合であつただけでなく、緊急的な状況の間に、貴重な時間が失われ、一方で、エンジニアは、対象とする状態データを取得する、または見つけるために、多数のビューまたは試験の実行を切り抜けなければならない。

【0103】

本明細書で説明されるシステム及び方法は、（特定のデバイス、設備、または試験結果ではなく）監視される安全イベントの現在の状態及び／または状態の変化の容易にアクセス可能な表示ビューを提供する。システムは、システム全体または領域全体の安全イベント／入力状態を、単一の表示ビューまたは視覚化にまとめて、経時的な安全イベントの変化を取り込み、安全表示ビュー全体の視覚化した安全イベントをデバイス／設備／試験結果にリンクする。

【0104】

「安全イベント」は、監視される条件の論理表現である。例示的な一実施形態において、CEMの各監視される入力（原因）は、監視される安全イベントとすることができます。加えて、各結果は、監視される安全イベントとすることができます。監視されることが所望される各安全イベントのそれぞれの状態及び／または状態のそれぞれの変化は、安全イベント視覚化ビュー上の異なるオブジェクト／アイテム／グラフィカルアイテムによって表される。例えば、各監視されるイベントは、色付きのドットによって提示することができ、異なる色によって、異なる現在の状態（例えば、赤色 - 不具合、青色 - 注意、黒色 - OK）を示す。加えて、または代替的に、例えば異なる色または表現によって、現在の状態の変化を（2進数で、及び／または変化の程度によって）表すことができる。そのような状態及び／または状態の変化は、経時的に取り込み、保存することができる。実際には、表示ビューは、監視されるイベントについて、回転するスナップショットの時間を提供することができ、また、異なるレートで（例えば、2分ごとに、20分ごとに、2時間ごとに）監視される安全イベントの異なるセクションを含むことができる。

【0105】

図16A～Dは、経時的な安全イベントの状態の変化の例示的な説明図である。図16A～Dにおいて、イベントE1は、監視される安全イベントである。経時的なE1の現在の安全状態は、時間軸に沿った形状のランニングラインとして表示ビュー上に表すことができ、（図16Aに表されるような）各形状は、異なる状態を表す。グラフ1600において、円形は、通常の状態を表し、正方形は、警告的な状態を表し、三角形は、危険な状態を表す。代替的に、経時的なE1の安全状態の変化は、変化が生じた時点に基づいて表すことができる。例えば、図16Bにおいて、グラフ1610は、「0」によって定常状態（またはいかなる状態変化もないこと）を示し、「-」によって安全状態の低下を示し、「+」によって安全状態の上昇を示す。安全状態及びその変化は、任意のタイプの数値またはグラフィカル形態で示すことができることを認識されたい。図16Cは、数字として状態を表示するグラフ1620を例示し、0からの負の増加は、追加的な状態の低下を表す。所望であれば、安全状態の変化の程度を表すことができる。例えば、グラフのy軸は、正常から外れた低下の範囲を示すことができ、経時的な安全状態は、線グラフ、または図16Dのドットの棒グラフ1630のように見え得る。加えて、監視されるイベントのドットの色付きランニングラインの単一の線を、ある色から別の色へ緩やかにモーフィ

10

20

30

40

50

ング / シェーディングして、安全状態の悪化及び / 改善を示すことができる。

【 0 1 0 6 】

上の実施例は、限定することを意図しておらず、数字、シンボル、色、グラフィックス、及び / または線の任意の組み合わせを表示して、エンジニアが、監視されるイベントの安全レベルを迅速に評価することを可能にすることができます。さらに、状態及び / または様々なイベントの状態の変化は、所望であれば、後処理のために記憶することができます。

【 0 1 0 7 】

一実施形態では、グラフ 1 つ以上のグラフを、一緒に及び / または同時に表示することができます。所望の監視されるイベントのグルーピングは、例えば、プラントの領域、機能、（例えば、バッチプロセスの特定の段階中の）特定の条件または要因等によって、接近させて表示することができる。エンジニアは、表示ビューをマスクして、一見するだけで対象の特定の安全イベントを見る能够になるようになり得る。

10

【 0 1 0 8 】

さらに、視覚化は、より抽象的な安全イベントを提供することができます。上で論じたように、監視される安全イベントは、監視及び結果ロック等の、監視されるイベントのグループの抽象化とすることができます。

【 0 1 0 9 】

例えば、図 5 及び 6 を参照すると、イネーブラまたは結果ロック E B 1 及び E B 2 への直接入力が、監視される安全イベントであり得、及び / または結果ロック E B 1 及び E B 2 の各結果 E 1、E 2 等が、監視される安全イベントであり得る。各所望の監視されるイベントが含まれる条件または入力の各々は、その状態の変化の程度に寄与し得る。例えば、イベントをトリガーするために、監視されるイベントが 4 つの条件を必要とする場合、1 つの条件が存在するときに、監視されるイベントの状態は、「 - 1 」であり得、2 つの条件が存在するときに、監視されるイベントの状態は、「 - 2 」であり得、3 つの条件が存在するときに、監視されるイベントの状態は、「 - 3 」であり得、4 つ全ての条件が存在するときに、状態は「 X 」または「トリップ済」であり得る。したがって、一例として、図 1 6 A ~ 1 6 D のシンボル、数字、ドット等は、結果の（または原因の）異なる可能な状態を表すことができ、または設定される原因の数、または設定する必要がある原因の総数を外れた真状態である、もしくは結果信号をトリップまたは開始するために真状態である原因の数を表すことができる。

20

【 0 1 1 0 】

さらに、特定の安全状態または状態変化インジケータのクリックまたは他のユーザ指示は、ユーザを、対応する条件（複数可）の詳細へ自動的にリンクすることができます。上で説明したように、エンジニアは、安全イベント視覚化グラフから、S R S、システム構成、及び / または C E M の表示にアクセスすることができます。例えば、監視されるイベントをトリップすることが必要である 4 つの条件の上の例を参照すると、上の例のグラフ 1 6 2 0 の監視されるイベントについて安全視覚化が「 - 1 」を示し、ユーザが「 - 1 」をクリックした場合、「 - 1 」の安全状態に対応する条件を生じさせる設備のデバイスまたは一部を含むシステム構成の表示ビューを表示することができます。

30

【 0 1 1 1 】

図 1 7 は、プロセスプラント内の安全イベントを視覚化する例示的な方法 1 7 0 0 のブロック図を表す。方法 1 7 0 0 は、サーバまたは別様には任意のタイプの電子デバイスによって容易にすることができます。サーバは、コンテンツを表示するように構成されたユーザインターフェースを備える、またはそれに接続することができます。

40

【 0 1 1 2 】

方法 1 7 0 0 は、サーバが、一組の原因及び一組の結果を有する C E M にアクセスする（ブロック 1 7 1 0 ）ときに開始することができます。実施形態において、一組の原因の各々は、プロセスプラント内の条件を表すことができ、一組の結果の各々は、プロセスプラント内で行われるべき結果を表すことができ。さらに、一組の原因及び一組の結果のうちの少なくともいくつかは、原因 - 結果の対として関連され、それによって、対応する結

50

果を、対応する条件の発生に応答して起動させることができ、一組の原因及び一組の結果は、プロセスプラント内の一組の監視される安全イベントを表すことができる。

【 0 1 1 3 】

サーバは、ユーザインターフェースを介して、一組の監視される安全イベントのうちの1つの監視される安全イベントの選択を受信することができる（ブロック1720）。さらに、サーバは、監視される安全イベントの指示及び監視される安全イベントの現在の状態をユーザインターフェースに表示することができる（ブロック1730）。実施形態において、サーバは、1つ以上の第1のグラフィカルオブジェクトとして現在の状態を表示することができる。

【 0 1 1 4 】

サーバは、監視される安全イベントの状態の変化を検出することができる（ブロック1740）。一実施形態において、サーバは、期間の終了に応答して、状態の変化を検出することができる。サーバはまた、状態の変化に従って、監視される安全イベントの更新された状態をユーザインターフェースに表示することもできる（ブロック1750）。実施形態において、サーバは、1つ以上の第1のグラフィカルオブジェクトと異なり得る1つ以上の第2のグラフィカルオブジェクトとして、更新された状態を表示することができる。さらに、実施形態において、サーバは、現在の状態と、監視される安全イベントの更新された状態との間の変化の程度を判定することができ、また、変化の程度をユーザインターフェースに表示することができる。

【 0 1 1 5 】

一実施形態において、サーバは、さらに、期間の終了に応答して、監視される安全イベントの更新された状態が変化しなかったと判定することができ、また、監視される安全イベントの更新された状態をユーザインターフェースに表示することができる。加えて、または代替的に、サーバは、ユーザインターフェースを介して、監視される安全イベントの更新された状態の選択を受信することができ、監視される安全イベントは、一組の関連付けられた条件を有することができ、また、一組の関連付けられた条件の各々の条件状態をユーザインターフェースに表示することができる。加えて、または代替的に、サーバは、ユーザインターフェースを介して、監視される安全イベントの更新された状態の選択を受信することができ、監視される安全イベントは、存在する関連付けられた条件を有することができ、また、存在するべき関連付けられた条件を生じさせているプロセスプラント内のデバイスの指示をユーザインターフェースに表示することができる。

【 0 1 1 6 】

加えて、または代替的に、サーバは、監視される安全イベント、監視される安全イベントの現在の状態、及び監視される安全イベントの更新された状態を表すデータをメモリに記憶することができる。さらに、加えて、または代替的に、サーバは、（i）一組の監視される安全イベントのうちの追加的な1つの監視される安全イベントの追加的な指示、及び（ii）追加的な監視される安全イベントの追加的な現在の状態をユーザインターフェースに表示し、追加的な監視される安全イベントの状態の追加的な変化を検出し、また、追加的な状態の変化に従って、追加的な監視される安全イベントの追加的な更新された状態をユーザインターフェースに表示することができる。

【 0 1 1 7 】

上で提供した例示的なCEMは、例示の目的であることを意図する、単純な表現である。図18は、時間遅延トリガー、許容トリガー、即時トリガー、及びリセットトリガーを含む、より複雑なCEMである、例示的なCEM1800を例示する。CEM1800は、現実のCEMのより正確な表現である、CEMのより実質的な一実施例である。図18に例示されるようなCEM1800において、「X」のみを含むセルは、即時トリガーの結果を表すことができる。さらに、「R」のみでポピュレートされた任意のセルは、原因を受信した場合に、リセットするように結果がトリガーされることを表すことができる。文字「T」で始まるCEM1800のセルは、原因が結果を直接トリガーするが、ある時間遅延を伴うことを示すことができる。時間遅延は、所定の増分に設定することができる

10

20

30

40

50

。例えば、「T1」が10秒の時間遅延に対応することができ、「T2」が20秒の時間遅延に対応することができる、等である。

【0118】

C E M 1 8 0 0 はまた、数字のみを含むセルも含み、それによって、これらのセルは「イネーブラ」に対応することができる。具体的には、セルの数字は、イネーブラが属するグループを識別し、各グループについて1つ以上のイネーブラが存在し得る。数字で始まるだけでなく、かつ他の文字も有するC E M 1 8 0 0 のセルは、イネーブラもトリガーされる場合に、結果のみをトリガーする関係を表すことができる。いくつかの実施形態において、対応するイネーブラ（または複数のイネーブラ）に関連するあらゆるセルは、トリガーされるべき結果について「オン」でなければならない。他の実施形態において、特定のイネーブラグループの原因の任意の組み合わせを組み合わせて、結果をトリガーすることができる。同様に、イネーブラの任意の組み合わせは、結果をトリガーすることを必要とし得る。

【0119】

例えば、図18に例示されるようなC E M 1 8 0 0において、原因1801は、グループ1のイネーブラである。したがって、原因1802がトリップされる場合、対応する結果は、原因1801もトリップされない限り、トリガーされ得ない。この原因結果関係は、イネーブラが「オン」である場合にのみ結果がトリガーされるので、許容関係とみなされる。この実施例を続けると、セル1803は、原因結果関係がイネーブラグループ1に属すること、及び原因信号1802が高くなり、かつイネーブラ原因信号1801も高いまたはオンであるときに、長さT1に対応する時間遅延を伴って結果1805がトリガーされることを示す。セル1804は、原因結果関係が、許容であり、かつイネーブラグループ1に属すること、ならびに原因信号1802が有効にされ（すなわち、原因信号1801がオンである）かつトリップされた場合に、結果1806が即時にトリガーされることを示す。

【0120】

C E M 1 8 0 0 は、上で説明した以前の方法700、1000、1200、1400、1500、及び1700の全てに実装することができる。さらに、C E MまたはC E Mの論理を実装するための本明細書で説明される監視及び結果ブロックは、例えばC E M 1 8 0 0 の複雑な論理機能及び相互に関係する論理機能を、または他のC E Mの任意の他の論理機能を実装するために使用することができる。C E M 1 8 0 0 の付加的な複雑さにもかかわらず、依然として、C E Mを管理する利点を適用する。さらに、例示的なC E Mは、限定することを意図しておらず、上で説明した方法700、1000、1200、1400、1500、及び1700のうちのいずれかを、安全論理をプロセスプラントに実装するために使用される原因結果マトリックスの任意の将来の実施形態と共に実装することができる。

【0121】

図7、10、12、14、15、及び17の方法700、1000、1200、1400、1500、方法1700の各々は、ソフトウェア、ファームウェア、もしくはハードウェア、またはソフトウェア、ファームウェア、及び／もしくはハードウェアのいくつかの組み合わせによって実装することができる。加えて、図7、10、12、14、15、17のフロー図は、ルーチンとして説明したが、これらのフロー図は、ソフトウェア、ハードウェア、ファームウェア、またはソフトウェア、ファームウェア、及び／もしくはハードウェアの組み合わせによって実装することができる。

【0122】

上で説明したユーザインターフェース等のユーザインターフェースの実施形態は、全体的または部分的に、例えばソフトウェアプログラムに従って構成された、プロセッサによって実装することができる。例えば、ワークステーション18aもしくは20a、またはいくつかの他のコンピュータは、全体的または部分的に、上で説明したユーザインターフェースを実装することができる。ユーザインターフェースの実施形態を実装するためのソ

ソフトウェアプログラムは、ハードディスク、RAM、バッテリバックアップされたRAM、ROM、CD-ROM、PROM、EPROM、EEPROM、DVD、フラッシュメモリ等の有形媒体に、またはプロセッサと関連付けられたRAM等のメモリに記憶されたソフトウェアに具現化することができるが、当業者は、プログラム全体またはその一部を、代替的に、プロセッサ以外のデバイスによって実行すること、ならびに／または既知の様態でファームウェア及び／もしくは専用のハードウェアに具現化することができることを容易に認識するであろう。

【0123】

本発明は、様々な修正物及び代替の構成物が可能であるが、本発明の特定の例示的な実施形態は、図面に示され、本明細書に詳細に記載される。しかしながら、本発明を、開示される特定の形態に限定することは意図しておらず、それとは逆に、添付の特許請求の範囲によって定義されるような本開示の趣旨及び範囲の範囲内に入る全ての修正物、代替の構成物、及び均等物を網羅することを意図することを理解されたい。

10

20

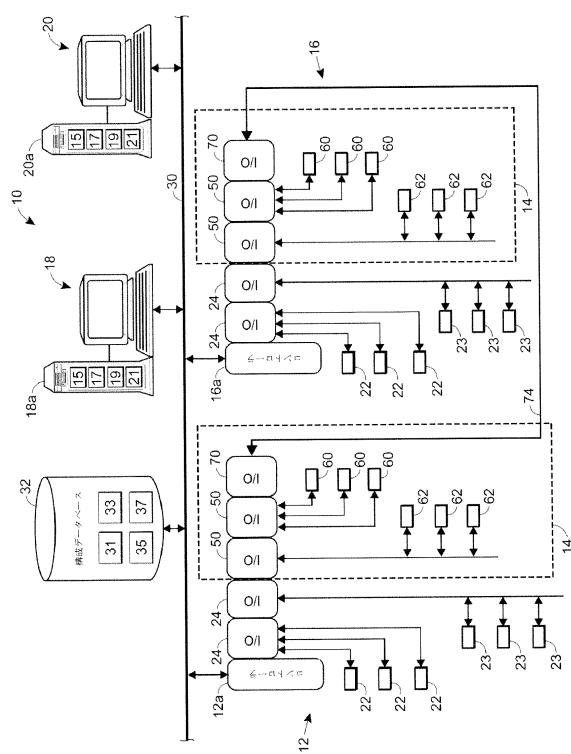
30

40

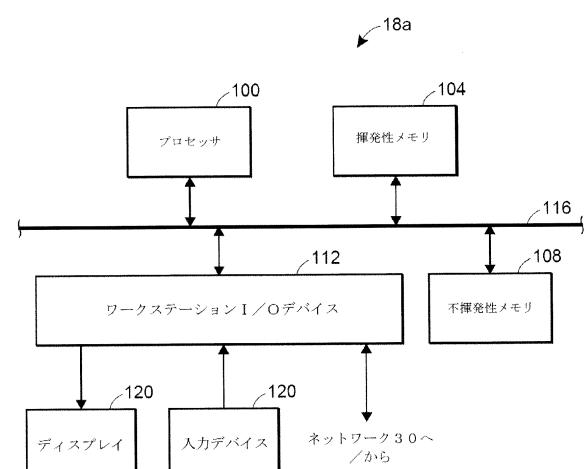
50

【図面】

【図 1】



【図 2】



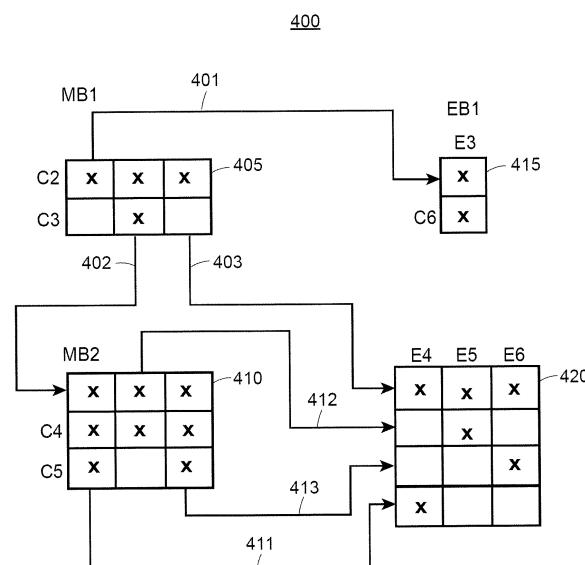
10

20

【図 3】

	結果1	結果2	結果3	結果4	結果5	結果6	結果7
原因1							
原因2			X	X	X		305
原因3				X			
原因4				X	X	X	310
原因5				X		X	
原因6			X				
原因7							

【図 4】

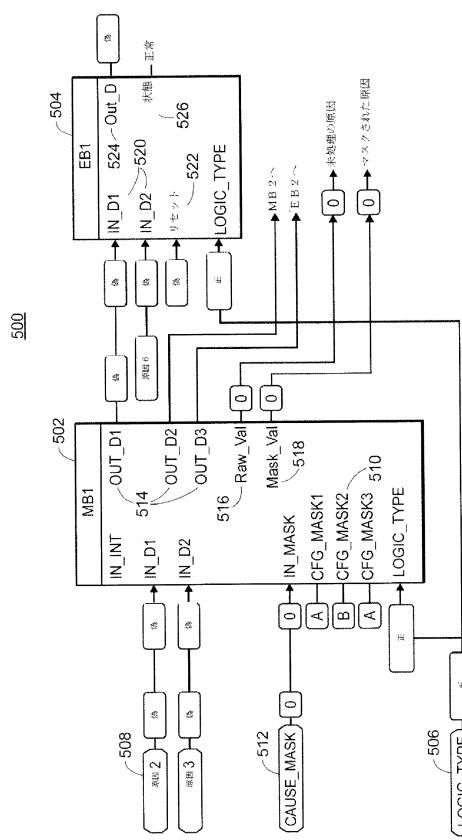


30

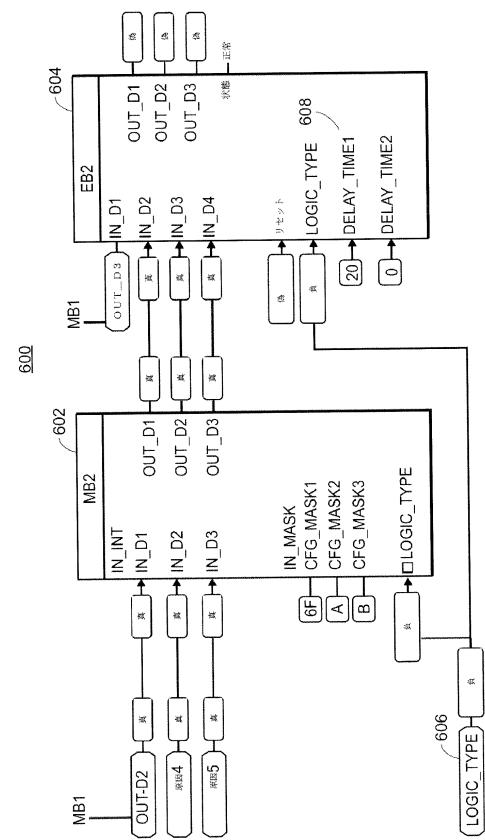
40

50

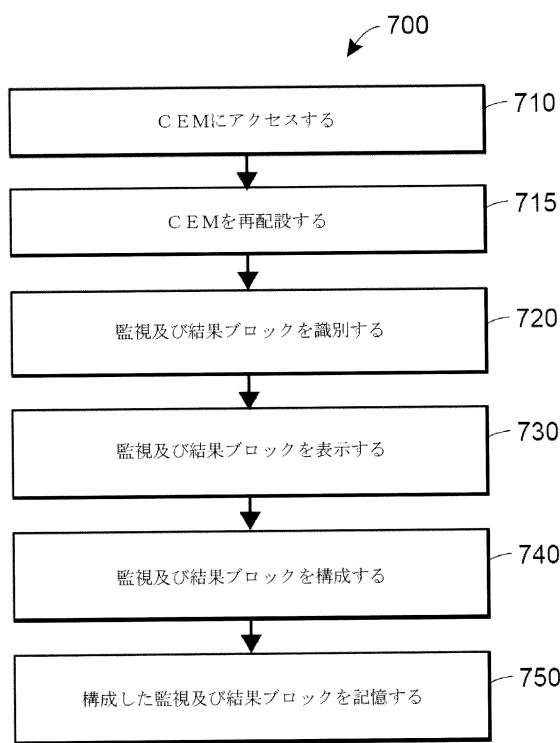
【図 5】



【図 6】



【図 7】



【図 8】

Table showing the relationship between causes and results (図 8). The table has 16 rows (原因 1 to 原因 16) and 17 columns (結果 1 to 結果 17).

		結果 1	結果 2	結果 3	結果 4	結果 5	結果 6	結果 7	結果 8	結果 9	結果 10	結果 11	結果 12	結果 13	結果 14	結果 15	結果 16	結果 17
原因 1	X																	
原因 2	X	X													X	X		
原因 3	X	X													X	X		
原因 4			X	X	X										X	X		
原因 5			X	X	X										X	X		
原因 6			X	X	X										X	X		
原因 7			X	X	X										X	X		
原因 8			X	X	X	X	X	X	X	X							X	
原因 9			X	X	X	X	X	X	X	X							X	
原因 10			X	X	X	X	X	X	X	X							X	
原因 11	X	X																
原因 12																		
原因 13																		
原因 14															X	X		
原因 15															X	X		
原因 16															X	X		

10

20

30

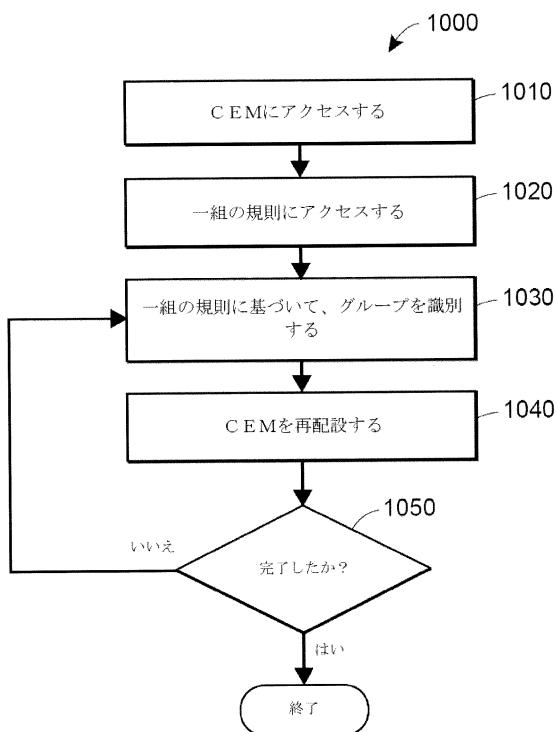
40

50

【図 9】

	結果 16	結果 1	結果 2	結果 3	結果 4	結果 5	結果 6	結果 7	結果 8	結果 9	結果 10	結果 11	結果 12	結果 13	結果 14	原因 14	原因 15	原因 1	原因 2	原因 3	原因 4	原因 5	原因 6	原因 7	原因 8	原因 9	原因 10	原因 11	原因 12	原因 13
原因 14																														
原因 15																														
原因 1																														
原因 2																														
原因 3																														
原因 4																														
原因 5																														
原因 6																														
原因 7																														
原因 8																														
原因 9																														
原因 10																														
原因 16																														
原因 11																														
原因 12																														
原因 13																														

【図 10】



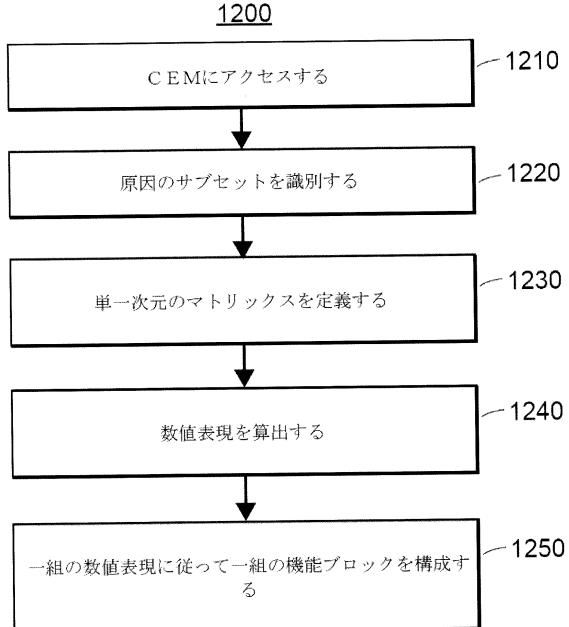
10

20

【図 11】

	結果 16	結果 1	結果 2	結果 3	結果 4	結果 5	結果 6	結果 7	結果 8	結果 9	結果 10	結果 11	結果 12	結果 13	結果 14	原因 14	原因 15	原因 1	原因 2	原因 3	原因 4	原因 5	原因 6	原因 7	原因 8	原因 9	原因 10	原因 11	原因 12	原因 13
原因 14																														
原因 15																														
原因 1																														
原因 2																														
原因 3																														
原因 4																														
原因 5																														
原因 6																														
原因 7																														
原因 8																														
原因 9																														
原因 10																														
原因 16																														
原因 11																														
原因 12																														
原因 13																														

【図 12】

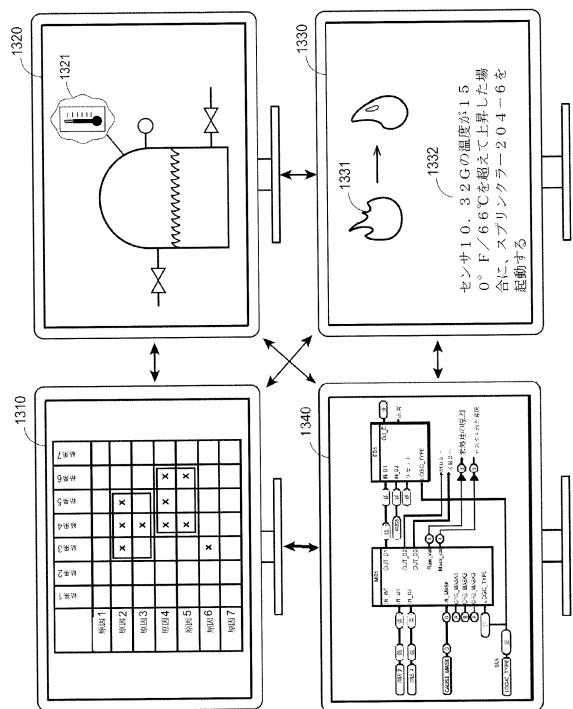


30

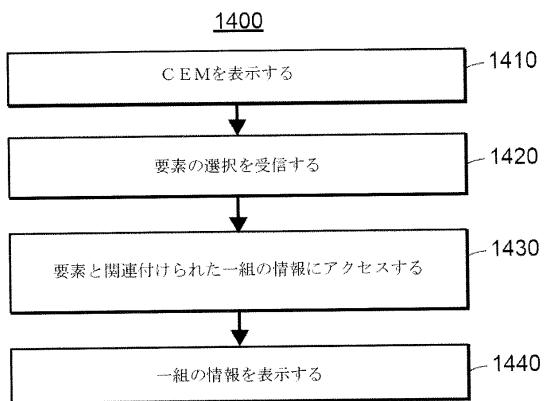
40

50

【図13】



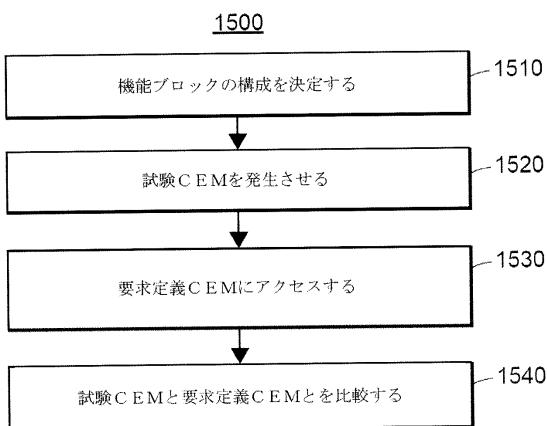
【図14】



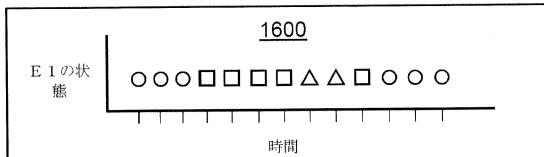
10

20

【図15】



【図16A】

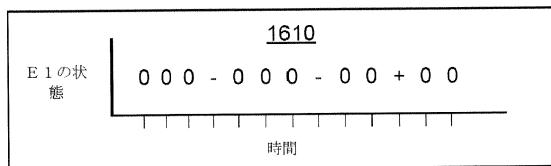


30

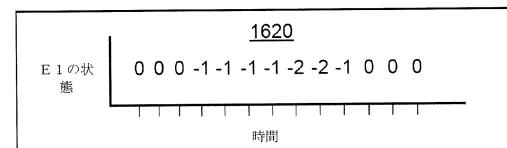
40

50

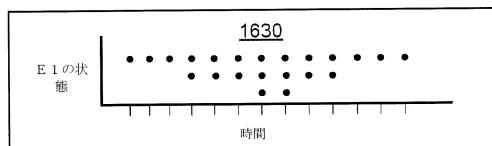
【図16B】



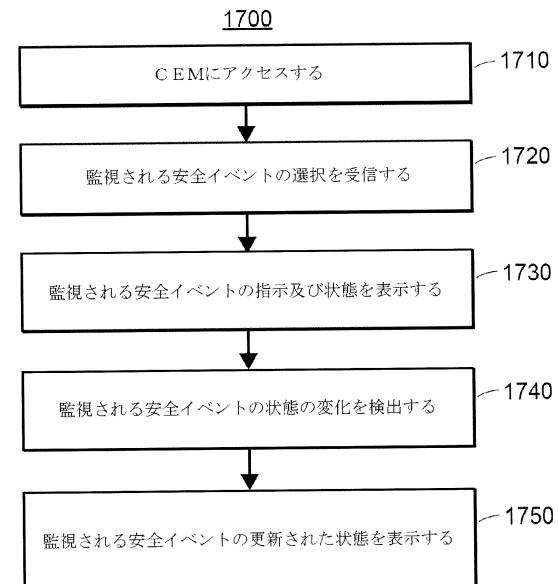
【図16C】



【図16D】



【図17】



【図 1 8】

Diagram illustrating a grid layout with various labels and data entries. The grid is organized into columns and rows, with specific cells highlighted or marked with symbols.

Key labels and their approximate locations:

- 1801 (row 1801)
- 1802 (row 1802)
- 1803 (row 1803)
- 1804 (row 1804)
- 1805 (row 1805)
- 1806 (row 1806)
- 10 (value 10)
- 20 (value 20)
- 30 (value 30)
- 40 (value 40)
- 50 (value 50)

Grid data (approximate values):

	1	2	3	4	5	6	7	8	9	10
XS-110549	24									
XS-110647	25			1	1			1		
XS-110648	26			2	2			2		
XS-110649	27			2	2			2		
	28									
	29									
	30									
	31									
XZS-110450	32			1T1	1X			1X		
XZS-110451	33			2T1	2X			2X		
XZS-110452	34			3T1	3X			3X		
XS-110460	35			1T1	1X			1X		
XS-110461	36			2T1	2X			2X		
XS-110462	37			3T1	3X			3X		
XS-110463	38			T1	X			X		
XZS-100451	39									
XZS-100452	40			T2	T1	X		X		
	41									
	42									
	43									
	44									
HZS-530507	45			R	R			R		
	46									
	47									

10

20

30

40

50

フロントページの続き

オースチン ウエスト ドーマン ドライブ 16410

審査官 堀内 亮吾

(56)参考文献 特開2004-303247 (JP, A)

米国特許出願公開第2014/0277619 (US, A1)

(58)調査した分野 (Int.Cl., DB名)

G 05 B 23/02