

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5579935号  
(P5579935)

(45) 発行日 平成26年8月27日(2014.8.27)

(24) 登録日 平成26年7月18日(2014.7.18)

(51) Int.Cl. F I  
G O 6 F 11/30 (2006.01) G O 6 F 11/30 3 1 0 D

請求項の数 9 (全 9 頁)

(21) 出願番号	特願2013-529938 (P2013-529938)	(73) 特許権者	390009531
(86) (22) 出願日	平成24年7月25日(2012.7.25)		インターナショナル・ビジネス・マシーンズ・コーポレーション
(86) 国際出願番号	PCT/JP2012/068896		INTERNATIONAL BUSINESS MACHINES CORPORATION
(87) 国際公開番号	W02013/027529		アメリカ合衆国10504 ニューヨーク州 アーモンク ニュー オーチャードロード
(87) 国際公開日	平成25年2月28日(2013.2.28)		
審査請求日	平成26年1月6日(2014.1.6)	(74) 代理人	100108501
(31) 優先権主張番号	特願2011-183312 (P2011-183312)		弁理士 上野 剛史
(32) 優先日	平成23年8月25日(2011.8.25)	(74) 代理人	100112690
(33) 優先権主張国	日本国(JP)		弁理士 太佐 種一
早期審査対象出願			

最終頁に続く

(54) 【発明の名称】 割り込み処理に起因する異常動作の検知

(57) 【特許請求の範囲】

【請求項 1】

多重割り込みシステムにおいて、割り込み処理に起因する異常動作を検知する方法であって、

割り込みの優先度毎に、所定のタイムアウト値を有するウォッチドッグタイマ(WDT)を準備するステップと、

各WDTを対応する優先度の割り込み要求があった時点で起動させるステップと、

少なくとも1つの前記WDTがタイムアウトした場合、当該WDTに対応する優先度よりも少なくとも1レベル以上高い優先度の割り込み要求を受け入れるステップと、を含み、

前記割り込み要求を受け入れるステップにおいて、前記割り込みの優先度の1つに複数の割り込み要因が割り当てられている場合、当該優先度よりも下位レベルのWDTのタイムアウトに起因する割り込み要求を優先させることにより、当該下位レベルの優先度の割り込み処理において異常動作があったことを検知する、方法。

【請求項 2】

前記各WDTを対応する優先度の割り込み要求があった時点で起動させるステップは、前記割り込みの優先度の1つに複数の割り込み要因が割り当てられている場合、当該割り込み要因中の最初の割り込み要求があった時点で対応するWDTを起動させるステップを含む、請求項1の方法。

【請求項 3】

10

20

各々の前記WDTを、対応する割り込み要求が受け入れられプロセッサによって当該割り込み処理が開始された時点でリセットするステップを含む、請求項1または2の方法。

【請求項4】

前記割り込み要求を受け入れるステップは、前記WDTに対応する優先度よりも少なくとも1レベル以上高い複数の優先度の割り込み要求を受け入れる、請求項1～3のいずれか1項の方法。

【請求項5】

前記優先度が最上位の割り込みに対応するWDTがタイムアウトした場合、前記システムをリセットするステップをさらに含む、請求項1～4のいずれか1項の方法。

【請求項6】

多重割り込みシステムにおいて、割り込み処理を制御するコントローラであって、割り込みの優先度毎に設けられ、それぞれ所定のタイムアウト値を有する複数のウォッチドッグタイマ(WDT)と、

デバイスから割り込み要求信号を受け取り、各割り込み要求信号が有する優先度に応じて対応する前記WDTに起動信号を出力する割り込み優先度セクタと、

少なくとも1つの前記WDTがタイムアウトした場合、当該WDTに対応する優先度よりも少なくとも1レベル以上高い優先度の割り込み要求信号をプロセッサへ出力する割り込み処理回路と、を備え、

前記割り込み処理回路は、前記割り込みの優先度の1つに複数の割り込み要因が割り当てられている場合、当該優先度よりも下位レベルのWDTのタイムアウトに起因する割り込み要求信号を優先させることにより、当該下位レベルの優先度の割り込み処理において異常動作があったことを検知するように構成されている、コントローラ。

【請求項7】

前記割り込み処理回路は、前記割り込み要求信号の出力に際して、対応する優先度以下のレベルの優先度を有する他の割り込み要求信号の出力を抑制するように構成されている、請求項6のコントローラ。

【請求項8】

各々の前記WDTは、対応する割り込み要求が受け入れられ前記プロセッサによって当該割り込み処理が開始された時点でリセットされる、請求項6または7のコントローラ。

【請求項9】

前記割り込みの優先度が最上位のWDTがタイムアウトした場合、当該優先度が最上位のWDTは前記システムをリセットする信号を出力する、請求項6～8のいずれか1項のコントローラ。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、多重割り込みシステムに関し、より具体的には、多重割り込みシステムにおける割り込み処理に起因する異常動作の検知に関する。

【背景技術】

【0002】

組み込みシステムなどにおいて、リアルタイム性の保障などのため、割り込みの処理中であっても、処理中の割り込みの優先度(レベル)よりも高い優先度の割り込み要求があった場合、処理中の割り込み処理を中断し、高い優先度の割り込み処理を行い、その完了後に、中断された割り込み処理を再開するように構築されたシステムがある。いわゆる多重割り込みシステム、あるいは割り込みを優先度によってネストさせるシステムである。

【0003】

また、多重割り込みシステムか否かに関わらず、システムが動作していることをハードウェアによって監視するためのウォッチドッグタイマ(Watchdog Timer、以下「WDT」と表記する)を備えているシステムがある。

【0004】

10

20

30

40

50

組み込みシステムなどにおいて、WDTがタイムアウト（エクスパイア）しないにも関わらず、システムが正常に動作しなくなることがある。異常動作の原因は多様であるが、割り込み処理に関連する異常動作も多く、多重割り込みシステムの場合には、自動復旧や原因解析が困難な場合が多い。

【0005】

異常動作の検知が不十分な場合の一例として、（通常のシステムではWDTが1つしかないこともあり）WDTのリセットが、タイマ割り込みによって（あるいはすべての割り込みによって）行われる場合が挙げられる。この場合、WDTはCPU（とその周辺のH/W）が動作していることの確認しか行えない。つまり、周辺回路が割り込み要求を出して、CPUがそれを受け付けていることの確認である。その際、システムが期待している動作を実行できているかの確認は行われていない。

10

【0006】

公開特許公報の62-175840号は、最高レベル（実行優先順位）の処理プログラムの実行の合間に、複数の処理プログラムをレベルに応じて実行するデータ処理システムを開示する。このシステムでは、処理プログラムのレベルに応じた異なるオーバーフロー値を有しかつ対応する処理プログラムの実行終了後にリセットされるWDTを複数設けて、そのいずれか1つのWDTのオーバーフローに基づきデータ処理システムの障害発生を検出する。

【0007】

公開特許公報の10-275097は、複数の処理プログラムをレベルに応じた優先順位のもとに実行するデータ処理システムを開示する。このシステムでは、複数の処理プログラムのそれぞれに個別に対応する複数のWDTを配置し、これら複数のWDTのオーバーフローに基づきデータ処理システムの暴走発生を検出する。

20

【先行技術文献】

【特許文献】

【0008】

【特許文献1】特開昭62-175840号公報

【特許文献2】特開平10-275097号公報

【発明の概要】

【発明が解決しようとする課題】

30

【0009】

特許文献1のデータ処理システムは、タイマによって起動される最高レベルの処理プログラムの実行を前提としており、最高レベルより下位レベル間での割り込み処理、あるいは1つのレベルに複数の処理プログラムが割り当てられている場合の割り込み処理などについて対応していない。また、WDTの起動タイミング等についての開示もない。

【0010】

特許文献2のデータ処理システムでは、複数のWDTが並行して起動することがないので、優先度の異なる複数の割り込みを並行して処理（管理）することができない。

【0011】

したがって、本発明の目的は、優先度の異なる複数の割り込みが並行して発生する場合や、1つのレベル（優先度）に複数の割り込みが並行して発生する場合などにおいても、割り込み処理に起因する異常動作の検知をすることができる、多重割り込み処理を可能にすることである。

40

【課題を解決するための手段】

【0012】

本発明は、多重割り込みシステムにおいて、割り込み処理に起因する異常動作を検知する方法を提供する。その方法は、割り込みの優先度毎に、所定のタイムアウト値を有するWDTを準備するステップと、各WDTに対応する優先度の割り込み要求があった時点で起動させるステップと、少なくとも1つのWDTがタイムアウトした場合、当該WDTに対応する優先度よりも少なくとも1レベル以上高い優先度の割り込み要求を受け入れるス

50

トップとを含み、割り込み要求を受け入れるステップにおいて、割り込みの優先度の1つに複数の割り込み要因が割り当てられている場合、当該優先度よりも下位レベルのWDTのタイムアウトに起因する割り込み要求を優先させることにより、当該下位レベルの優先度の割り込み処理において異常動作があったことを検知する。

**【0013】**

本発明の一態様では、多重割り込みシステムにおいて、割り込み処理を制御するコントローラを提供する。そのコントローラは、割り込みの優先度毎に設けられ、それぞれ所定のタイムアウト値を有する複数のWDTと、デバイスから割り込み要求信号を受け取り、各割り込み要求信号が有する優先度に応じて対応するWDTに起動信号を出力する割り込み優先度セレクトと、少なくとも1つのWDTがタイムアウトした場合、当該WDTに対応する優先度よりも少なくとも1レベル以上高い優先度の割り込み要求信号をプロセッサへ出力する割り込み処理回路と、を備えている。さらに、その割り込み処理回路は、割り込みの優先度の1つに複数の割り込み要因が割り当てられている場合、当該優先度よりも下位レベルのWDTのタイムアウトに起因する割り込み要求信号を優先させることにより、当該下位レベルの優先度の割り込み処理において異常動作があったことを検知するように構成されている。

10

**【0014】**

本発明およびその一態様によれば、割り込みの優先度が同一あるいは異なる複数の割り込みがあった場合において、いずれの優先度での割り込み処理に異常があったかを検知することができる。また、本発明では、優先度の下位レベルのWDTのタイムアウトに起因する割り込み要求を優先させて、言い換えれば、割り込みの処理を開始した時点（下位の優先度の割り込み処理が開始されないことが判明した時点）割り込み処理の異常を検知するので、従来のWDTのタイムアウト時での検知よりもその異常検知の確度（精度）を向上させることが可能となる。

20

**【図面の簡単な説明】****【0015】**

【図1】本発明の割り込み処理に起因する異常動作を検知する方法の流れを説明する図である。

【図2】本発明の割り込み処理を制御するコントローラの構成を示す図である。

**【発明を実施するための形態】**

30

**【0016】**

図面を参照しながら本発明の実施の形態を説明する。図1は、多重割り込みシステムにおいて、割り込み処理に起因する異常動作を検知する方法の流れを説明する図である。なお、本方法は後述する本発明の割り込み処理を制御するコントローラを一例として基本的にハードウェアで実装可能であるが、プログラム（ソフトウェア）による実施を排除するものではなく、両者の協業の形態で実装される場合があることは言うまでもない。

**【0017】**

ステップS11において、複数のWDTを準備する。各WDTは、割り込みの優先度（レベル）毎に、所定のタイムアウト値を有するに構成される。所定のタイムアウト値は、優先度、割り込み要因に応じて設定される。ここで、割り込み要因とは、処理されるプログラム、I/Oを介した各種の入出力などのCPUが処理する内容を意味する。一般に、割り込みの優先度が上がるほどタイムアウト値は短く設定される。

40

**【0018】**

ステップS12において、割り込み要求があった優先度のWDTをその割り込み要求を受けた時点で起動させる。その際、割り込みの優先度の1つに複数の割り込み要因が割り当てられている場合、その割り込み要因中での最初の割り込み要求があった時点に対応するWDTを起動させる。起動したWDTは、その優先度の割り込みの受付時、すなわち対応する割り込み要求が受け入れられプロセッサによって当該割り込み処理が開始された時点で（割り込みハンドラの実行開始時に）リセットされる。

**【0019】**

50

ステップS 1 3において、少なくとも1つのW D Tがタイムアウト（エキスパイア）したか否かを判定する。具体的にはW D Tのカウント値が所定のタイムアウト値以上になったか（オーバーフローしたか）否かを判定する。この判定がN oの場合、割り込み要求が順調に処理されているので、ステップS 1 2の前に戻り、次の割り込み要求を待つ状態となる。

**【 0 0 2 0 】**

ステップS 1 3の判定がY e sの場合、次のステップS 1 4において、そのタイムアウトしたW D Tの優先度よりも少なくとも1レベル以上高い（上位の）優先度の割り込み要求を受け入れる。同時にその上位のW D Tを起動させる。その際、選択される優先度はタイムアウトしたW D Tの1レベル上のみならず、2レベル以上上の優先度であってもよい。また、選択される優先度は1つとは限らず、2以上の複数の優先度を選択してもよい。すなわち、割り込み要求を受け入れるステップS 1 4においては、W D Tに対応する優先度よりも少なくとも1レベル以上高い複数の優先度の割り込み要求を受け入れることができる。

10

**【 0 0 2 1 】**

ステップS 1 4において、割り込みの優先度の1つに複数の割り込み要因が割り当てられている場合、当該優先度よりも下位レベルのW D Tのタイムアウトに起因する割り込み要求を優先させる。例えば、レベルNの優先度の割り込み要因が複数あったとしても、1つ下の（N - 1）レベルの優先度のW D Tのタイムアウトがあった場合、その（N - 1）レベルの優先度の割り込み要求を優先させて（あたかも新たなNレベルとみなして）割り込ませる。この仕組みを採用するのは、後述するステップS 1 6において、当該下位レベル（上の例では（N - 1）レベル）の優先度の割り込み処理において異常動作があったことを検知させるためである。

20

**【 0 0 2 2 】**

ステップS 1 5において、ステップS 1 4で選択された少なくとも1つ上位の優先度のW D Tが、タイムアウト（エキスパイア）したか否かを判定する。具体的な判定は、ステップS 1 3の場合と同様に、W D Tのカウント値が所定のタイムアウト値以上になったか（オーバーフローしたか）否かを判定する。

**【 0 0 2 3 】**

ステップS 1 5の判定がN oの場合、すなわちステップS 1 4で選択された少なくとも1つの上位の優先度の割り込み要求が受け入れられてその割り込み処理が順調に進んだ場合、ステップS 1 6において、下位の優先度での割り込みの異常が検知される。具体的には、上述したステップS 1 4において説明したように、下位レベルのW D Tのタイムアウトに起因した割り込み要求が採用された場合、そのW D Tがタイムアウトした優先度の割り込み処理に異常があったことが検知される。このように、本発明では、優先度の下位レベルのW D Tのタイムアウトに起因する割り込み要求を優先させて、言い換えれば、割り込みの処理を開始した時点（下位の優先度の割り込み処理が開始されないことが判明した時点）割り込み処理の異常を検知するので、後述する各実施例の記載からも明らかなように、従来のW D Tのタイムアウト時での検知よりもその異常検知の確度（精度）を向上させることが可能となる。

30

40

**【 0 0 2 4 】**

ステップS 1 5の判定がY e sの場合、ステップS 1 7において、優先度が最上位の割り込みに対応するW D Tがタイムアウト（エキスパイア）したか否かを判定する。具体的な判定は、ステップS 1 3の場合と同様に、W D Tのカウント値が所定のタイムアウト値以上になったか（オーバーフローしたか）否かを判定する。この判定がN oの場合、ステップS 1 4に戻り、ステップS 1 5でタイムアウトしたW D Tの優先度よりも少なくとも1レベル以上さらに高い（上位の）優先度の割り込み要求を受け入れる。同時にその上位のW D Tを起動させる。

**【 0 0 2 5 】**

ステップS 1 7の判定がY e sの場合、ステップS 1 8においてシステムをリセットす

50

る。このシステムのリセットにより割り込み処理の異常が検知される。

【 0 0 2 6 】

次に、図 2 を参照しながら、本発明の割り込み処理を制御するコントローラについて説明する。図 2 はそのコントローラ 1 0 0 の一実施形態である構成を示す図である。コントローラ 1 0 0 は、多重割り込みシステムの一部として、あるいは外部のコントローラとして構成される。なお、図 2 では、例として 3 つの割り込みの優先度（レベル L 1、L 2、L 3）についての構成を示しているが、これはあくまで一例であって、4 以上の数の任意の優先度の場合は、以下に述べる各構成要素を増やして拡張していけばよい。

【 0 0 2 7 】

図 2 において、コントローラ 1 0 0 は、優先度セクタ 1 1、1 2、1 3、優先度ライン L 1、L 2、L 3、W D T 2 1、2 2、2 3、および割り込み処理回路 3 1 を含む。優先度セクタ 1 1、1 2、1 3 の出力は、それぞれ優先度ライン L 1、L 2、L 3 に接続される。優先度セクタ 1 1、1 2、1 3 は、各デバイス（図示なし）から割り込み要求信号 R 1、R 2、R 3 を受け取り、各割り込み要求信号が有する優先度に応じて対応する優先度ライン L 1、L 2、L 3 に各要求信号を出力する。より具体的には、優先度セクタ 1 1、1 2、1 3 は、割り込み要求信号 R 1、R 2、R 3 が有する割り込み要因に応じて予め設定した優先度に応じて各要求信号を振り分ける。この要求信号は、W D T 2 1、2 2、2 3 の起動信号ともなる。なお、優先度セクタの数は 3 つに限られず少なくとも 2 以上の任意の数でよい。

【 0 0 2 8 】

W D T 2 1 は、優先度 L 1 に対応した W D T であり、その入力が優先度ライン L 1 に接続し、その出力は優先度ライン L 2 に接続する。W D T 2 1 は、所定のタイムアウト値 T 1 を有する。所定のタイムアウト値は、既に図 1 のステップ S 1 1 で説明したように、優先度や割り込み要因に応じて設定される。同様に、W D T 2 2、2 3 は、それぞれ優先度 L 2、L 3 に対応した W D T であり、各入力が優先度ライン L 2、L 3 に接続する。W D T 2 2 の出力は優先度ライン L 3 に接続する。W D T 2 3 の出力は、後述するように、システムをリセットする信号として出力される。W D T 2 2、2 3 は、それぞれ所定のタイムアウト値 T 2、T 3 を有する。図 2 の例では、優先度は L 1 < L 2 < L 3 のように高くなるので、タイムアウト値は、逆に T 1 > T 2 > T 3 のように小さくなる。各 W D T は、対応する割り込み要求が C P U へ送られてその割り込み処理が開始された時点で、C P U あるいは別のリセット回路（図示なし）からのリセット信号 R S 1 によってリセットされる。

【 0 0 2 9 】

W D T のうち割り込みの優先度が最上位の W D T は、自らがタイムアウトした場合、システムをリセットする信号を C P U 内蔵のあるいは外付けのリセット回路（図示なし）へ向けて出力するように構成される。図 2 の例では、最上位の W D T 2 3 がタイムアウトした場合、システムのリセット信号 R S 2 を出力する。このシステムのリセットにより割り込み処理の異常が検知される。

【 0 0 3 0 】

割り込み処理回路 3 1 は、優先度ライン L 1、L 2、L 3 から優先度で割り振られた割り込み要求信号 R L 1、R L 2、R L 3 を受け取り、C P U へ向けて選択的に割り込み要求信号を送信する。ここで選択的な送信とは、割り込み要求信号の出力に際して、対応する優先度以下のレベルの優先度を有する他の割り込み要求信号の出力を抑制することを意味する。例えば、優先度レベル（N - 1）の割り込み要求信号を出力する場合、上位の優先度レベル N 以上の割り込み要求信号を除いて、優先度レベル（N - 1）以下の他の割り込み要求信号の出力を抑制する。

【 0 0 3 1 】

割り込み処理回路 3 1 は、少なくとも 1 つの W D T がタイムアウトした場合、当該 W D T に対応する優先度よりも少なくとも 1 レベル以上高い優先度の割り込み要求信号をプロセッサへ出力するように構成される。例えば、図 2 において、W D T 2 1 がタイムアウト

10

20

30

40

50

した場合、1つ上位の優先度ラインL2からの要求信号RL2をCPUへ向けて選択的に送信する。その際、優先度L2の要求信号RL2のみならず、2レベル上の優先度L3の要求信号RL3を選択して送信してもよい。また、2つの要求信号RL2、RL3を選択して順次送信するようにしてもよい。

【0032】

割り込み処理回路31は、割り込みの優先度の1つに複数の割り込み要因が割り当てられている場合、当該優先度よりも下位レベルのWDTのタイムアウトに起因する割り込み要求信号を優先させるように構成される。例えば、図2において、レベルL2の優先度の割り込み要因が複数(例えばR1とR3)あったとしても、1つ下のレベルL1の優先度のWDT21のタイムアウトがあった場合、そのレベルL1の優先度の割り込み要求RL1を優先させて(あたかもL2レベルの要求RL2とみなして)割り込ませる。この仕組みを採用するのは、当該下位レベル(上の例ではレベルL1)の優先度の割り込み処理において異常動作があったことを検知させるためである。

10

【0033】

次に、本発明の方法、コントローラによる割り込み処理の異常検知例を実施例として以下に説明する。

【実施例1】

【0034】

<優先度レベルNの割り込み処理が間違っていて割り込み要因をクリアしないため、優先度レベルNの割り込み要求が出続ける(異常動作)場合>

20

(1)優先度レベル(N-1)の割り込み要求は、無限に受け付けられない。このため、優先度レベル(N-1)のWDTはタイムアウトし、上位の優先度レベルNの割り込み要求が行われる。この優先度レベルNの割り込み要求は受け付けられる。その際、(異常動作の割り込み要求を含めて)複数の割り込み要因に対して、既に上述したように、タイムアウトした下位の優先度レベルのWDTに対応する割り込み処理が最優先されるので、割り込み処理の異常が検知される。仮に優先順位の判定を間違った場合でも、下記の実施例2の(1)の場合と同様に処理されて異常が検知される。

【0035】

(2)(N-2)以下の優先度レベルの割り込み要求が発端の場合、それぞれの優先度レベルのWDTがタイムアウトし、順次上位の優先度レベルのWDTが起動されてタイムアウトするので、上記(1)の場合と同様に異常が検知される(以下の実施例でも同様)。

30

【実施例2】

【0036】

<優先度レベルNの割り込み処理が無限ループ(異常動作)をしている場合>

(1)優先度レベル(N-1)の割り込み要求は、無限に受け付けられない。このため、優先度レベル(N-1)のWDTはタイムアウトし、上位の優先度レベルNの割り込み要求を行う。優先度レベルNの割り込み要求も無限に受け付けられない。このため、優先度レベルNのWDTはタイムアウトし、上位の優先度レベル(N+1)の割り込み要求を行う。この優先度レベル(N+1)の割り込み要求は受け付けられるので、割り込み処理の異常が検知される。

40

【0037】

(2)優先度レベルNの(新たな)割り込み要求が発端であっても、優先度レベルNのWDTがタイムアウトするので、上記(1)の場合と同様に異常が検知される。

【実施例3】

【0038】

<1つ1つの割り込み処理は正常に処理されているが、CPUの処理能力と比べ割り込み要求が多いため、優先度低レベルNの割り込み処理が行われない(異常動作)場合>

優先度レベルNの割り込み要求が受け付けられず、WDTがタイムアウトし、優先度レベル(N+1)の割り込み要求が行われる。その後、上記各実施例の場合と同様に動作し、異常が検知される。

50

【実施例4】

【0039】

< 割り込み処理中に、全ての割り込みを長時間禁止してしまう（異常動作）場合 >

いずれかの優先度レベルのWDTがタイムアウトし、順次上位の優先度レベルの割り込み要求が行われる。最終的には、最上位のWDTがタイムアウトし、システムのリセットが行われて異常が検知される。システムのリセット以前に割り込みが許可された場合でも、その時点で最上位となっているWDTに対応する割り込みによって、異常が検知される。

【0040】

本発明の実施形態について、図を参照しながら説明をした。しかし、本発明はこれらの実施形態に限られるものではない。本発明はその趣旨を逸脱しない範囲で当業者の知識に基づき種々なる改良、修正、変形を加えた態様で実施できるものである。

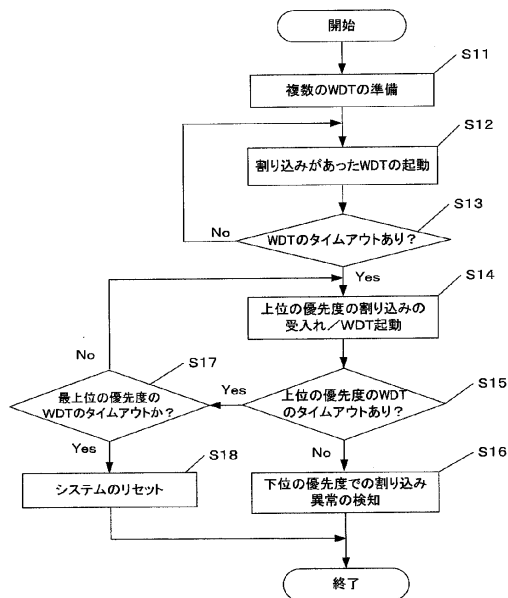
10

【符号の説明】

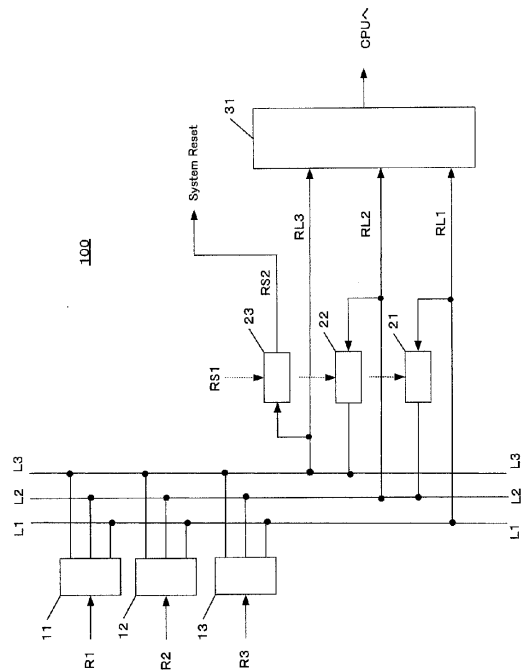
【0041】

- 11、12、13 優先度セレクタ
- 21、22、23 WDT
- 31 割り込み処理回路
- 100 コントローラ

【図1】



【図2】





---

フロントページの続き

(72)発明者 白鳥 敏幸

東京都江東区豊洲五丁目6番52号 NBF豊洲キャナルフロント 日本アイ・ビー・エム株式会社  
社 IBM東京ラボラトリー内

審査官 大塚 俊範

(56)参考文献 特開昭63-308646(JP,A)  
特開平11-237995(JP,A)  
特開平4-155543(JP,A)  
特開平11-161523(JP,A)  
特開2010-9258(JP,A)  
特開2011-258032(JP,A)  
米国特許第5408643(US,A)

(58)調査した分野(Int.Cl., DB名)

G06F 11/28 - 11/34