

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5063212号  
(P5063212)

(45) 発行日 平成24年10月31日(2012.10.31)

(24) 登録日 平成24年8月17日(2012.8.17)

(51) Int. Cl. F I  
**G06F 11/14 (2006.01)** G O 6 F 11/14 3 1 0 L  
**G06F 1/24 (2006.01)** G O 6 F 1/00 3 5 0 B

請求項の数 2 (全 11 頁)

(21) 出願番号	特願2007-166383 (P2007-166383)	(73) 特許権者	502129933 株式会社日立産機システム 東京都千代田区神田練塀町3番地
(22) 出願日	平成19年6月25日(2007.6.25)	(74) 代理人	110000442 特許業務法人 武和国際特許事務所
(65) 公開番号	特開2009-3862 (P2009-3862A)	(72) 発明者	柳原 徳久 千葉県習志野市東習志野七丁目1番1号 株式会社日立産機システム内
(43) 公開日	平成21年1月8日(2009.1.8)	(72) 発明者	木原 一 千葉県習志野市東習志野七丁目1番1号 株式会社日立産機システム内
審査請求日	平成22年5月26日(2010.5.26)	(72) 発明者	山田 勉 茨城県日立市大みか町七丁目1番1号 株 式会社日立製作所日立研究所内

最終頁に続く

(54) 【発明の名称】 複数コンポーネントシステム

(57) 【特許請求の範囲】

【請求項1】

少なくともCPUを備えたコンポーネントが共通バスで相互に接続されてなる複数コンポーネントシステムであって、

スイッチ操作によってリセット信号を発生し、前記共通バスとは別設のリセット線を通して、前記コンポーネント夫々に供給する第1のリセット信号発生手段を前記コンポーネントとは別に設け、

前記コンポーネント夫々には、前記CPUからの所定の信号の検知を基に前記CPUの状態を正常か異常かを検出する状態検出手段を設けるとともに、前記状態検出手段の検出結果に応じて、前記CPUのリセットの可否を判定する判定手段を設け、

前記判定手段は、前記状態検出手段からの信号を基に前記CPUが正常状態にあると判定する場合、前記第1のリセット信号発生手段からの前記リセット信号による前記CPUのリセットを禁止し、前記状態検出手段からの信号を基に前記CPUが異常状態にあると判定する場合、前記第1のリセット信号発生手段からの前記リセット信号によって前記CPUをリセットし、

さらに、前記コンポーネント夫々の前記CPUは、正常状態であるとき生存情報を出力して、前記共通バスを介して他の前記コンポーネントに供給するとともに、他の前記コンポーネントからの前記生存情報を前記共通バスを介して取得することにより、他の前記コンポーネントでの前記CPUが正常状態にあるか否かを判定し、

前記コンポーネント毎に、前記CPUが他の前記コンポーネントでの前記CPUを前記

生存情報を基に異常状態にあると判定したとき、リセット信号を発生して前記コンポーネント夫々の前記判定手段に前記リセット線を通して供給する第2のリセット信号発生手段を前記コンポーネントに設け、

前記コンポーネントの前記CPUは夫々、正常状態にあるとき、CPUが異常状態となったコンポーネントでのCPUでの処理結果のデータを前記共通バスを介して取得し、しかる後、前記第2のリセット信号発生手段から前記リセット信号を発生させる

ことを特徴とする複数コンポーネントシステム。

【請求項2】

請求項1において、

前記状態検出手段は、前記CPUからの所定の周期信号毎にカウントをリセットして初期値からのカウントを繰り返し、前記CPUが異常状態になると、前記所定の周期信号が供給されなくなってカウントがオーバーフローして前記CPUが異常状態であることを検出する

ことを特徴とする複数コンポーネントシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、LANなどの少なくともCPU(Central Processing Unit: 中央処理装置)を備えたコンポーネントが複数共通バスを介して接続されて構成された複数コンポーネントシステムに係り、特に、該コンポーネントでの異常状態となったCPUのリセット機構に関する。

【背景技術】

【0002】

従来、リセット線を含む共通バスを介して複数のCPUユニットが接続され、夫々のCPUユニットで全体の処理を分担して実行するようにしたマルチCPUシステムが知られているが、かかるマルチCPUシステムにおいて、実行中のアプリケーションプログラムの指定により、CPUユニットをリセットするようにした技術が提案されている(例えば、特許文献1参照)。

【0003】

この技術は、少なくともCPUユニットを含む複数のサブラック(コンポーネント)からなるシステムであって、このシステムの運用中に、いずれかのサブラックのCPUユニットで、実行中のアプリケーションプログラムの指定により、システムリセット要求が発生するものであり、かかる要求があると、このCPUユニットがリセット権を獲得し、リセット線を介して他のサブラックのCPUユニットをリセットするものである。

【特許文献1】特開平5-290008号公報

【発明の開示】

【発明が解決しようとする課題】

【0004】

ところで、上記特許文献1に記載のシステムでは、このシステムを構成する複数のサブラック、即ち、コンポーネントのCPUユニットが夫々全体の処理を分担して実行するのであるから、システムリセット要求により、全てのコンポーネントのCPUユニットが同時にリセットされるようにしているものであるが、複数のコンポーネントから構成され、個々のコンポーネントが個別に独立の処理を行なうシステムでは、上記特許文献1に記載のシステムのように、リセット信号により、全てのコンポーネントのCPUユニットが同時にリセットされたのでは、正常に処理を行なっているCPUユニットもリセットされることになり、これまでの処理データが失われ、これまでの処理が無駄になってしまうという問題がある。

【0005】

かかる問題を解消するために、例えば、あるコンポーネントでのCPUユニットに異常が発生した場合、このCPUユニットのみをリセットするようにすることが考えられるが

10

20

30

40

50

、このためには、いずれかにCPUユニットが異常状態になる毎に、CPUユニットが異常状態となったコンポーネントをオペレータが見つげ出す作業が必要となり、オペレータにとって非常に手間がかかることになる。

【0006】

さらに、CPUにWDT (Watch Dog Timer : ウォッチドッグタイマ) を設け、このWDTがCPUの状態を常時監視し、CPUが異常状態となったことを検出すると、このCPUをリセットする技術も知られており、CPUユニットを有するコンポーネントの複数からなるシステムにおいて、コンポーネント毎にかかるWDTを設け、そのCPUが異常状態となると、このWDTによってCPUユニットがリセットされるようにすることも考えられ、これにより、異常状態となったCPUユニットのみリセットするようにすることができる。

10

【0007】

ところで、この方法によると、オペレータの意思に関係なく、異常状態となったCPUが自動的にリセットされる。しかしながら、システムに異常がみられた場合、オペレータとしては、例えば、この異常を十分に確認したり、場合によっては、この異常のもとになるコンポーネントを見つけ出し、そのCPUのこれまでの処理データを確保するなどの処置を講じたりしたい場合もあり、このようなオペレータの要求があるにもかかわらず、自動的に異常状態のCPUがリセットされてしまうことになる。

【0008】

本発明の目的は、かかる問題を解消し、異常状態となったコンポーネントのCPUのみをオペレータの希望するタイミングでリセットすることができるようにした複数コンポーネントシステムを提供することにある。

20

【0009】

本発明の他の目的は、さらに、異常状態となったコンポーネントのCPUのみを、その処理データが損なわれないタイミングでリセットすることができるようにした複数コンポーネントシステムを提供することにある。

【課題を解決するための手段】

【0010】

上記目的を達成するために、本発明は、少なくともCPUを備えたコンポーネントが共通バスで相互に接続されてなる複数コンポーネントシステムであって、スイッチ操作によってリセット信号を発生し、前記共通バスとは別設のリセット線を通して、前記コンポーネント夫々に供給する第1のリセット信号発生手段を前記コンポーネントとは別に設け、前記コンポーネント夫々には、前記CPUからの所定の信号の検知を基に前記CPUの状態を正常か異常かを検出する状態検出手段を設けるとともに、前記状態検出手段の検出結果に応じて、前記CPUのリセットの可否を判定する判定手段を設け、前記判定手段は、前記状態検出手段からの信号を基に前記CPUが正常状態にあると判定する場合、前記第1のリセット信号発生手段からの前記リセット信号による前記CPUのリセットを禁止し、前記状態検出手段からの信号を基に前記CPUが異常状態にあると判定する場合、前記第1のリセット信号発生手段からの前記リセット信号によって前記CPUをリセットし、さらに、前記コンポーネント夫々の前記CPUは、正常状態であるとき生存情報を出力して、前記共通バスを介して他の前記コンポーネントに供給するとともに、他の前記コンポーネントからの前記生存情報を前記共通バスを介して取得することにより、他の前記コンポーネントでの前記CPUが正常状態にあるか否かを判定し、前記コンポーネント毎に、前記CPUが他の前記コンポーネントでの前記CPUを前記生存情報を基に異常状態にあると判定したとき、リセット信号を発生して前記コンポーネント夫々の前記判定手段に前記リセット線を通して供給する第2のリセット信号発生手段を前記コンポーネントに設け、前記コンポーネントの前記CPUは夫々、正常状態にあるとき、CPUが異常状態となったコンポーネントでのCPUでの処理結果のデータを前記共通バスを介して取得し、しかる後、前記第2のリセット信号発生手段から前記リセット信号を発生させることを特徴とするものである。

30

40

50

## 【 0 0 1 1 】

また、本発明は、前記複数コンポーネントシステムにおいて、前記状態検出手段は、前記CPUからの所定の周期信号毎にカウントをリセットして初期値からのカウントを繰り返す、前記CPUが異常状態になると、前記所定の周期信号が供給されなくなってカウントがオーバーフローして前記CPUが異常状態であることを検出することを特徴とするものである。

## 【 発明の効果 】

## 【 0 0 1 5 】

本発明によると、オペレータが必要とするタイミングで、異常状態となったコンポーネントのCPUのみをリセットさせることができ、正常状態にあるコンポーネントのCPUをそのまま処理を継続させることができる。

10

## 【 0 0 1 6 】

また、本発明によると、CPUが異常状態となったコンポーネントに対して、正常なコンポーネントのCPUがアクセスでき、CPUが異常状態となったコンポーネントでのこれまでの処理結果を確保するなどの処理をして後、この異常状態となったCPUを自動的にリセットすることができ、かかるリセットによっても、これまでの処理を無駄にすることがない。

## 【 発明を実施するための最良の形態 】

## 【 0 0 1 7 】

以下、本発明の実施形態を図面を用いて説明する。

20

なお、以下の実施形態としては、機械などの制御などに用いられるPLC（プログラマブル・ロジック・コントローラ）を例に説明するが、本発明は、これのみに限るものではなく、夫々が少なくともCPUを備えた複数のコンポーネントからなる任意のシステムに適用可能である。

## 【 0 0 1 8 】

図1は本発明による複数コンポーネントシステムの第1の実施形態の要部を示す概略構成図であって、10は電源、11はリセットスイッチ、20はLAN（Local Area Network）、30はシリアル、40はCF（Compact Flash：コンパクトフラッシュ（登録商標））、21、31、41はCPU、22、32、42は状態検出手段、23、33、43は判定回路、24、34、44はメモリ、25はLAN I/F（Interface）、35はシリアルI/F、45はCF I/F、50はPCI（Peripheral Components Interconnect）バスである。

30

## 【 0 0 1 9 】

同図において、この第1の実施形態は、LAN 20やシリアル30、CF 40などのコンポーネントとこれらに電源電圧を供給する電源10とで構成されており、これらコンポーネント20、30、40は共通バスとしてのPCIバス50によって相互に接続されている。

## 【 0 0 2 0 】

LAN 20は、CPU 21と、CPU 21が実行するプログラムやCPU 21が演算処理するデータを格納するメモリ24と、LAN I/F 25とを備え、さらに、CPU 21をリセットするための状態検出手段22や判定回路23が設けられた構成をなしており、LAN I/F 25によってインターネットとの通信を行なうコンポーネントである。また、シリアル30は、同様にして、CPU 31と、CPU 31が実行するプログラムやCPU 31が演算処理するデータを格納するメモリ34と、LAN I/F 35とを備え、さらに、CPU 31をリセットするための状態検出手段32や判定回路33が設けられた構成をなしており、シリアルI/F 35によってRS 232Cなどの規格に基づいて外部機器との通信を行なうコンポーネントである。さらに、CF 40は、同様にして、CPU 41と、CPU 41が実行するプログラムやCPU 41が演算処理するデータを格納するメモリ44と、CF I/F 45とを備え、さらに、CPU 41をリセットするための状態検出手段42や判定回路43が設けられた構成をなしており、LAN I/F 45に

40

50

よってCFカードなどの記録媒体と通信を行なうコンポーネントである。このように、これらコンポーネント20, 30, 40は、1つの全体処理を分担して行なうものではなく、夫々個別に独立した演算処理を行なうものである。

#### 【0021】

電源10には、図示しない電源スイッチなどの各種操作スイッチとともに、リセットスイッチ11が設けられており、このリセットスイッチ11がオペレータなどによって操作されると、図示しないリセット信号発生手段によって高レベルのリセット信号Rsが発生し、各コンポーネント(LAN20やシリアル30, CF40)に供給される。

#### 【0022】

図2は図1におけるコンポーネントの1つであるLAN20の要部の具体的な構成を示す回路図であって、図1に対応する部分には同一符号を付けている。

10

#### 【0023】

同図において、CPU21とRAM24とLAN I/F25とはPCIバス50を介して接続されている。CPU21は、PCIバス50を介し、RAM24でデータの読み出し、書き込みを行なって演算処理を行なうとともに、LAN I/F25などの各部の制御を行なう。また、CPU21に状態検出手段22と判定回路23とが接続されており、これにより、後述するように、CPU21が異常状態となったとき、電源10でリセットスイッチ11の操作によって供給されるリセット信号Rsにより、このCPU21をリセットする。

#### 【0024】

20

かかる構成は、図1におけるシリアル30, CF40においても同様であり、これらLAN20やシリアル30, CF40の異なるコンポーネント間でPCIバス50が相互に接続されている。

#### 【0025】

LAN20において、状態検出手段22は、例えば、WDT(Watch Dog Timer:ウォッチ・ドッグ・タイマ)である。WDTは、CPU21からの割込信号Irによってリセットされるカウンタからなるものである。CPU21は、正常状態にあるとき、一定の周期でこの割込信号Irを発生し、この割込信号Irが供給される毎に、WDTはリセットされて初期値からのカウントを繰り返す。CPU21が異常状態になると、このCPU21から割込信号Irが供給されなくなるので、WDTはカウントを継続し、オーバフローすると、異常信号Abを発生し、これを判定回路23に供給する。

30

#### 【0026】

判定回路23は、異常状態検出手段22から異常信号Abが供給されないときには、CPU21は正常な状態にあるとしてこのCPU21のリセットを不可とする判定をする。異常状態検出手段22から異常信号Abが供給されたときには、CPU21は異常な状態にあるとしてこのCPU21のリセットを可とする判定をし、リセット信号Rsが供給されると、このCPU21をリセットする。

#### 【0027】

図3は判定回路23の一具体例を示す回路構成図であって、23aはS/RMM(セット/リセットマルチバイブレータ)、23bはANDゲートである。

40

#### 【0028】

同図において、S/RMM23aは、電源10(図1)のリセットスイッチ11(図1)が操作されることによって供給される高レベルのリセット信号Rsの終端エッジ(立下りエッジ)でリセットされ、状態検出手段22(図1)からの異常信号Abの始端エッジでセットされ、セットされたときには、高レベルのQ出力が出力される。このQ出力はANDゲート23bの一方の入力となり、リセット信号RsがANDゲートに他方の入力として供給される。状態検出手段22からの異常信号AbによってS/RMM23aがセット状態にあるときには、リセットスイッチ11の操作によるリセット信号RsがANDゲート23bを通過する。このリセット信号Rsは、CPU21(図1, 図2)に供給され、これをリセットする。

50

## 【 0 0 2 9 】

次に、この第 1 の実施形態の動作について、図 1 を用いて説明する。

## 【 0 0 3 0 】

L A N 2 0 やシリアル 3 0 , C F 4 0 などのコンポーネントの C P U 2 1 , 3 1 , 4 1 が正常に動作しているときには、かかるシステムの図示しない、例えば、制御対象の機器は正常に動作しているが、これらコンポーネントのいずれかの C P U が異常状態となると、制御対象の機器の動作はこの C P U の異常状態の影響を受ける。これによってオペレータはシステムのいずれかのコンポーネントに異常が生じたことを認識することができ、この異常を認識してオペレータがリセットスイッチ 1 1 を操作すると、リセット信号 R s が発生してこれら L A N 2 0 やシリアル 3 0 , C F 4 0 に供給される。

10

## 【 0 0 3 1 】

一方、C P U が異常状態となったコンポーネントでは、これを、例えば、L A N 2 0 とすると、上記のように、状態検出手段 2 2 が C P U 2 1 が異常状態になったことを検出して異常信号 A b を判定回路 2 3 に供給し、判定回路 2 3 が、C P U 2 1 が異常状態にあるとして、この C P U 2 1 のリセットを可とする判定をしている。かかる状態で電源 1 0 からリセット信号 R s が供給されると、この判定回路 2 3 がこの供給されたリセット信号 R s を C P U 2 1 に供給する。これにより、この C P U 2 1 がリセットされる。

## 【 0 0 3 2 】

C P U が正常状態にあるコンポーネント、この場合では、シリアル 3 0 と C F 4 0 では、状態検出手段 3 2 , 4 2 が夫々 C P U 3 1 , 4 1 の正常状態を検出しているので、異常信号 A b を発生せず、判定回路 3 3 , 4 3 は夫々 C P U 3 1 , 4 1 が正常状態にあるとしてこれらのリセットを不可と判定する。このため、C P U 3 1 , 4 1 は夫々、リセット信号 R s が供給されても、リセットされずに継続して演算処理動作を行なう。

20

## 【 0 0 3 3 】

このようにして、いずれかのコンポーネントの C P U が異常状態となってシステムに異常が生じても、そのいずれのコンポーネントに異常が生じたかをオペレータが認識しなくとも、また、認識できなくとも、システムが異常となったことを認識したときにリセットスイッチ 1 1 を操作するだけで、異常状態となったコンポーネントの C P U だけを確実にリセットすることができるし、正常に動作しているコンポーネントの C P U をもリセットするといった好ましくない事態を回避できる。

30

## 【 0 0 3 4 】

しかも、オペレータのリセットスイッチ 1 1 の操作により、即ち、オペレータの要求により、異常状態となったコンポーネントの C P U をリセットするものであるから、かかる C P U のリセットタイミングもオペレータによって決めることができ、例えば、制御対象の機器に所定の処置を施してから、あるいは、異常状態となった C P U のこれまでの処理データを確保できるものであれば、これを確保した後、リセットスイッチ 1 1 を操作して異常状態となったコンポーネントの C P U をリセットするといった処置を施すことができる。このように、処理データを確保できる場合には、リセットした C P U を立ち上げるときに、この確保したデータを利用することができ、異常状態になる直前までの処理が無駄になることがない。

40

## 【 0 0 3 5 】

図 4 は本発明による複数コンポーネントシステムの第 2 の実施形態を示すブロック構成図であって、2 6 , 3 6 , 4 6 はリセット信号発生回路であって、図 1 に対応する部分には同一符号をつけて重複する説明を省略する。

## 【 0 0 3 6 】

同図において、この第 2 の実施形態は、図 1 に示す第 1 の実施形態において、各コンポーネント、即ち、L A N 2 0 , シリアル 3 0 , C F 4 0 に夫々リセット信号発生回路 2 6 , 3 6 , 4 6 を設け、これによってリセットスイッチ 1 1 が操作されたときのリセット信号 R s と同じフォーマットのリセット信号 R s ' を発生するものである。リセット信号 R s ' を発生するコンポーネントは、L A N 2 0 , シリアル 3 0 , C F 4 0 のいずれかの C

50

PUが異常となったとき、CPUが正常状態にあるコンポーネントであり、このリセット信号Rs'がCPUが異常状態にあるコンポーネントに判定回路から取り込まれ、この異常状態にあるCPUをリセットする。

**【0037】**

ここで、図5に示すように、CPU21とRAM24とLAN I/F25とはPCIバスに接続され、CPU21が、PCIバス50を介して、RAM24でデータの読み出し、書き込みを行なって演算処理を行なうとともに、LAN I/F25などの各部の制御を行なう。また、CPU21に状態検出手段22と判定回路23とリセット信号発生回路26とが接続されており、これにより、先の第1の実施形態と同様、CPU21が異常状態となったとき、電源10でリセットスイッチ11の操作によって供給されるリセット信号Rsにより、このCPU21がリセットされるとともに、リセット信号発生回路26から、このリセット信号Rsと同じフォーマットのリセット信号Rs'を発生させることができる。また、この第2の実施形態では、CPU21がPCIバス50を介して他のコンポーネント、この場合、シリアル30やCF40と通信をすることができる。

10

**【0038】**

かかる構成は、図1におけるシリアル30、CF40においても同様であり、これらLAN20やシリアル30、CF40の異なるコンポーネント間でPCIバス50が相互に接続されており、これらLAN20やシリアル30、CF40で後述する必要なデータの通信を行なう。

**【0039】**

各コンポーネントのCPU21、31、41は夫々、正常状態にあるとき、正常な状態にあることを示す生存情報を、PCIバス50を介し、一定の時間間隔で他のコンポーネントに送る。従って、CPU21、31、41は夫々、正常状態にあるときには、他のコンポーネントのCPUが正常状態にあるのか否かを常に認識している。

20

**【0040】**

そこで、いま、シリアル30のCPU31が異常状態となったとすると、第1の実施形態と同様、判定回路33がCPU31をリセット可と判定した状態となるが、これとともに、CPU31は生存情報をPCIバス50に出力しない状態となる。このため、LAN20とCF40とでは、CPU21、41がシリアル30から生存信号を取得できない状態となる。これにより、これらCPU21、41はシリアル30のCPU31が異常状態となったことを認識する。

30

**【0041】**

そこで、CPU21、41のいずれか一方、例えば、CPU21は、シリアル30に対して所定の処理を行ない、この処理の終了後、リセット発生回路26を制御してリセット信号Rs'を発生させる。このリセット信号Rs'はシリアル30とCF40とに供給されるが、シリアル30では、その判定回路33がCPU31のリセットを可とする判定をしており、ANDゲート23b(図3)をオン状態にしているので(即ち、判定回路33がオン状態にあるので)、LAN20のリセット発生回路26で発生されたリセット信号Rs'が供給されると、これをもとに判定回路33をリセットする。

**【0042】**

なお、リセット発生回路26がリセット信号Rs'を発生する前にCPU21によるシリアル30に対する上記の所定の処理とは、例えば、CPU21が、CPU31が異常状態となったことを検出すると、PCIバス50を介してシリアル30のRAM34での処理プログラムやデータを読み込んでRAM24に格納する処理であり、これにより、異常状態になるまでのCPU31で行なわれた処理の結果(処理データ)を確保し、さらに、CPU31が異常状態となる直前からの処理を、他のCPUが、あるいはこのCPU31が立ち上がったときに、継続して行なうことができるようにするものである。これにより、CPU31でこれまで行なわれた処理の結果を無駄にすることがないし、また、PCIバス50を介してシリアルI/F35も制御し、シリアル30の処理を継続して実行できるようにするものである。

40

50

## 【 0 0 4 3 】

また、C F 4 0 が正常に動作している場合には、その判定回路 4 3 は C P U 4 1 が正常状態にあるとしてリセットを不可と判定した状態にある。このため、C F 4 0 では、L A N 2 0 のリセット信号発生回路 2 6 で発生したリセット信号 R s ' はこの判定回路 4 3 で阻止され、これにより、C P U 4 0 はリセットされない。

## 【 0 0 4 4 】

さらに、いずれかのコンポーネントの C P U が異常状態となったとき、残りの全てのコンポーネントの C P U がリセット発生回路からリセット信号 R s ' を発生させるようにしてもよいが、コンポーネントに優先順位が設定されており、いずれかのコンポーネントの C P U が異常状態となったときには、残りのコンポーネントのうちの最上位の順位のコンポーネントの C P U がリセット発生回路からリセット信号 R s ' を発生させるようにしてもよい。

10

## 【 0 0 4 5 】

また、P C I バス 5 0 が不具合となって通信ができなくなった場合には、夫々の C P U 2 1 , 3 1 , 4 1 が生存情報をこの P C I バス 5 0 に出力しても、そのうちのいずれかが他のコンポーネントの C P U に送信されない。そこで、図 4 において、例えば、シリアル 3 0 と C F 4 0 との間で P C I バスが不具合となると、L A N 2 0 の C P U 2 1 は C F 4 0 からの生存情報を受信することができず、この C F 4 0 の C P U 4 1 が異常状態にあるとして、リセット信号発生回路 2 6 からリセット信号 R s ' を発生させる。このリセット信号は C F 4 0 にも供給されるが、この C F 4 0 での C P U 4 1 は生存情報を出力して

20

## 【 0 0 4 6 】

このように、P C I バス 5 0 に不具合が生じて、生存情報が伝達されなくとも、正常状態にある C P U はリセットされず、処理をそのまま続けることになる。なお、このような場合、リセット信号 R s ' が継続して出力されることになり、これにより、P C I バス 5 0 の異常を容易に検出することができる。

## 【 0 0 4 7 】

以上のように、この第 2 の実施形態においても、リセットスイッチ 1 1 を操作することによる第 1 の実施形態と同様の効果が得られるものであるが、さらに、オペレータがスイッチ操作しなくとも、異常状態となっているコンポーネントの C P U を、この C P U のこれまでの処理によって得られたデータを損なうことなく、自動的にリセットさせることができる。

30

## 【 0 0 4 8 】

さらに、以上の各実施形態では、状態検出手段 2 2 , 3 2 , 4 2 は C P U 2 1 , 3 1 , 4 1 のハード的な異常状態を検出するものであったが、例えば、プログラムに不備があるなどしてソフト的な異常を検出する手段を設け、この検出結果も判定回路 2 3 , 3 3 , 4 3 に供給し、C P U 2 1 , 3 1 , 4 1 のハード的に異常状態となったときも、同様にリセットするようにしてもよい。

40

## 【 図面の簡単な説明 】

## 【 0 0 4 9 】

【 図 1 】 本発明による複数コンポーネントシステムの第 1 の実施形態を示す全体斜視図である。

【 図 2 】 図 1 におけるコンポーネントの要部の具体的な構成を示す回路図である。

【 図 3 】 図 1 における判定回路の一具体例を示す回路図である。

【 図 4 】 本発明による複数コンポーネントシステムの第 2 の実施形態を示す全体斜視図である。

【 図 5 】 図 4 におけるコンポーネントの要部の具体的な構成を示す回路図である。

## 【 符号の説明 】

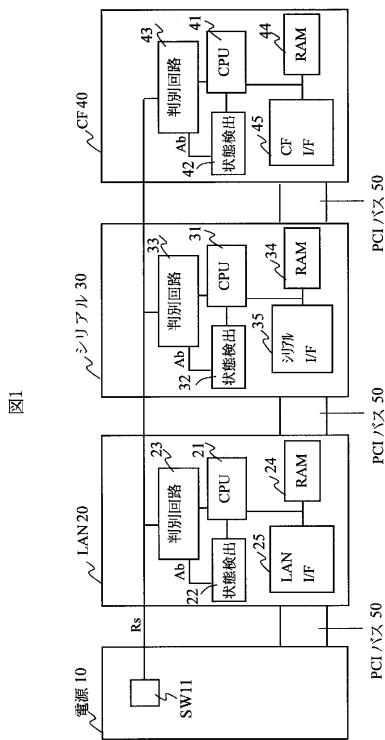
50



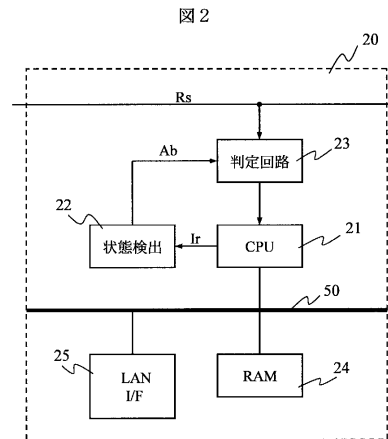
【 0 0 5 0 】

- 1 0 電源
- 1 1 リセットスイッチ
- 2 0 LAN
- 3 0 シリアル
- 4 0 CF
- 2 1 , 3 1 , 4 1 CPU
- 2 2 , 3 2 , 4 2 状態検出手段
- 2 3 , 3 3 , 4 3 判定回路
- 2 4 , 3 4 , 4 4 メモリ
- 2 5 LAN I / F
- 2 5 シリアル I / F
- 2 5 CF I / F
- 2 6 , 3 6 , 4 6 リセット信号発生回路
- 5 0 PCIバス

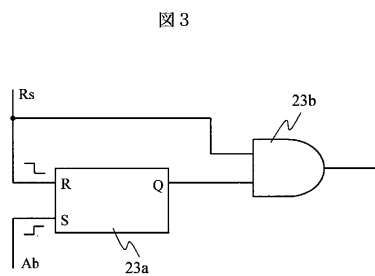
【 図 1 】



【 図 2 】



【 図 3 】



【 図 4 】

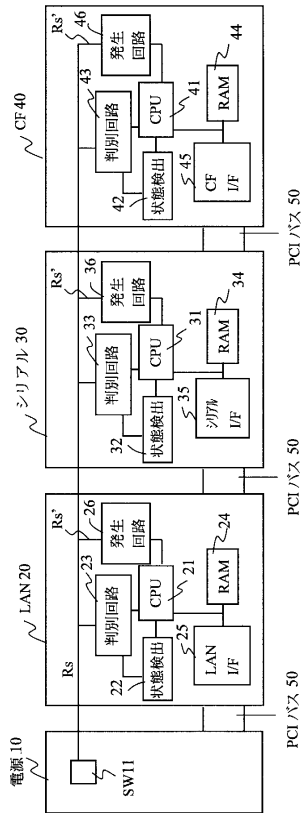


図4

【 図 5 】

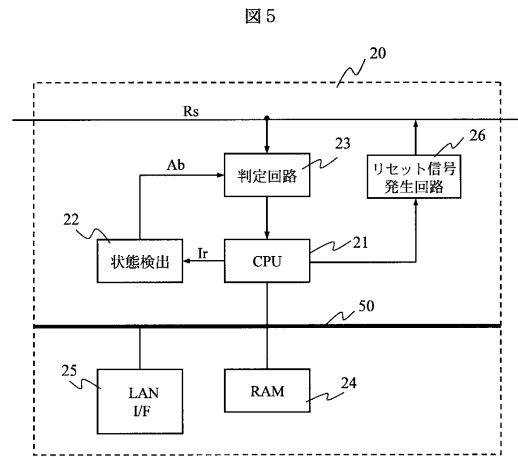


図 5

---

フロントページの続き

- (72)発明者 苗村 万紀子  
千葉県習志野市東習志野七丁目1番1号 株式会社日立産機システム内
- (72)発明者 清野 憲二  
千葉県習志野市東習志野七丁目1番1号 株式会社日立産機システム内

審査官 高橋正徳

- (56)参考文献 特開平05-143197(JP,A)  
特開平07-295687(JP,A)  
特開平05-216855(JP,A)  
特開平04-295956(JP,A)  
特開平05-073344(JP,A)  
特開2000-200203(JP,A)  
特開平05-290008(JP,A)  
米国特許第5086505(US,A)  
特表2009-514084(JP,A)  
英国特許出願公開第2415799(GB,A)  
米国特許出願公開第2005/0086460(US,A1)

(58)調査した分野(Int.Cl., DB名)

G06F 11/14,  
G06F 1/24,  
G06F 11/28-11/34,  
G06F 15/16-15/177