

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6617618号
(P6617618)

(45) 発行日 令和1年12月11日(2019.12.11)

(24) 登録日 令和1年11月22日(2019.11.22)

(51) Int.Cl.		F I			
G06F 11/34	(2006.01)	G06F 11/34	1 7 6		
G06F 11/30	(2006.01)	G06F 11/30	1 4 0 A		

請求項の数 8 (全 29 頁)

(21) 出願番号	特願2016-48034 (P2016-48034)	(73) 特許権者	000005223
(22) 出願日	平成28年3月11日 (2016. 3. 11)		富士通株式会社
(65) 公開番号	特開2017-162337 (P2017-162337A)		神奈川県川崎市中原区上小田中4丁目1番1号
(43) 公開日	平成29年9月14日 (2017. 9. 14)	(74) 代理人	100092978
審査請求日	平成30年12月10日 (2018.12.10)		弁理士 真田 有
		(72) 発明者	細田 智博
			神奈川県川崎市中原区上小田中4丁目1番1号 株式会社富士通コンピュータテクノロジーズ内
		(72) 発明者	牧野 司
			神奈川県川崎市中原区上小田中4丁目1番1号 株式会社富士通コンピュータテクノロジーズ内

最終頁に続く

(54) 【発明の名称】 情報処理装置および制御プログラム

(57) 【特許請求の範囲】

【請求項 1】

複数のプロセッサを備える情報処理装置であって、
前記複数のプロセッサのうちの第1のプロセッサが、
当該第1のプロセッサの動作履歴情報を収集してログ記憶部に格納するログ収集部と

、
前記ログ記憶部に格納された前記動作履歴情報を、不揮発性の記憶装置に移動させるログ出力部と、

当該第1のプロセッサが低負荷状態から高負荷状態に遷移した場合に、前記複数のプロセッサのうち最も低負荷の第2のプロセッサに、前記ログ出力部としての機能を移管させる移管処理部と

しての機能を実行することを特徴とする、情報処理装置。

【請求項 2】

前記移管処理部は、前記第1プロセッサが高負荷状態から低負荷状態に遷移した場合に、前記ログ出力部としての機能を移管した前記第2のプロセッサに対して、前記ログ出力部としての機能の移管を解除する通知を送信することを特徴とする、請求項1記載の情報処理装置。

【請求項 3】

前記第1のプロセッサが、
前記第1のプロセッサによって実行される前記ログ出力部が前記記憶装置に移動させる

10

20

前記動作履歴情報に対応するプロセッサを示す第 1 のログ出力対象情報と、

前記第 1 のプロセッサによって実行されるログ出力部としての機能の移管先の前記第 2 のプロセッサを示す移管先プロセッサ情報とを備え、

前記移管処理部が、

前記第 2 のプロセッサに前記ログ出力部としての機能を移管させる際に、前記第 1 のログ出力対象情報から前記第 1 のプロセッサの登録を削除するとともに、前記移管先プロセッサ情報に前記第 2 のプロセッサを追加する

ことを特徴とする、請求項 2 記載の情報処理装置。

【請求項 4】

前記移管処理部が、

前記第 2 のプロセッサに前記ログ出力部としての機能の移管を解除する通知を送信する際に、前記第 1 のログ出力対象情報に前記第 1 のプロセッサを追加するとともに、前記移管先プロセッサ情報から前記第 2 のプロセッサの登録を削除する

ことを特徴とする、請求項 3 記載の情報処理装置。

【請求項 5】

前記第 2 のプロセッサが、

前記第 2 のプロセッサによって実行されるログ出力部が前記記憶装置に移動させる前記動作履歴情報に対応するプロセッサを示す第 2 のログ出力対象情報を備え、

前記第 1 のプロセッサから前記ログ出力部としての機能を移管させる通知を受信すると、前記第 2 のログ出力対象情報に前記第 1 のプロセッサを追加する

ことを特徴とする、請求項 2 ～ 4 のいずれか 1 項に記載の情報処理装置。

【請求項 6】

前記第 2 のプロセッサが、

前記第 1 のプロセッサから前記ログ出力部としての機能の移管を解除する通知を受信すると、前記第 2 のログ出力対象情報から前記第 1 のプロセッサの登録を削除する

ことを特徴とする、請求項 5 記載の情報処理装置。

【請求項 7】

前記複数のプロセッサのうち第 3 のプロセッサに異常が検出された場合に、前記複数のプロセッサのうち前記第 1 プロセッサおよび第 3 のプロセッサ以外のプロセッサの動作を停止させる停止処理部

を備えることを特徴とする、請求項 1 ～ 6 のいずれか 1 項に記載の情報処理装置。

【請求項 8】

複数のプロセッサを備える情報処理装置において、

前記複数のプロセッサのうちの第 1 のプロセッサに、

当該第 1 のプロセッサの動作履歴情報を収集してログ記憶部に格納する処理と、

前記ログ記憶部に格納された前記動作履歴情報を不揮発性の記憶装置に移動させる処理と、

当該第 1 のプロセッサが低負荷状態から高負荷状態に遷移した場合に、前記複数のプロセッサのうち最も低負荷の第 2 のプロセッサに、前記動作履歴情報を不揮発性の記憶装置に移動させる処理を移管させる処理と

を実行させる制御プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、情報処理装置および制御プログラムに関する。

【背景技術】

【0002】

KVM (Kernel-based Virtual Machine) や Xen 等の仮想化ソフトや、カーネルの開発を行なう際に、CPU (Central Processing Unit) のコントロールレジスタ値等を直接設定し、CPU やメモリ管理ユニット (Memory Management Unit: MMU) の動作モー

10

20

30

40

50

ドを変更する。

【 0 0 0 3 】

プログラムミスやハードウェア異常により、CPUのレジスタ設定に誤りが生じた場合、CPUやカーネルに異常が発生し、システムが停止（ハングアップ）した状態になる。システムが停止状態になるとTTY（Tele-Typewriter：コンソール）経由での操作が不能となり、その時の動作状況を知るためのログ採取を行なうことが出来ず、原因調査が難航する。

【 0 0 0 4 】

そこで、従来のコンピュータシステムにおいては、例えばICE（In-Circuit Emulator）やHW（Hard Ware）シミュレータのような設備や開発環境（調査用専用機器）を別途備えることで、CPUのレジスタ情報等を採取している。

10

【 0 0 0 5 】

また、近年、複数のCPUを搭載したハードウェアでカーネル（kernel）を動作させる手法として、対称型マルチプロセッシング（Symmetric Multiprocessing：SMP）カーネルが用いられる場合がある。

【 0 0 0 6 】

図16はSMPカーネルを説明するための図である。

【 0 0 0 7 】

この図16に示すように、SMPを用いた並列処理システムにおいては、複数のCPU #0～#3を用いてSMPカーネル（以下、単にカーネルという場合がある）が実行され、更に、このカーネル上でアプリケーションが実行される。そして、全てのCPU #0～#3に対して対称的、均一的に処理が割り付けられる。

20

【 0 0 0 8 】

SMPカーネルにおいては、カーネルスケジューラが各CPUに処理を分散させて効率よく処理をさせる。

【 先行技術文献 】

【 特許文献 】

【 0 0 0 9 】

【 特許文献 1 】 特開 2 0 0 4 - 1 8 5 6 0 2 号 公 報

【 特許文献 2 】 特開 2 0 0 4 - 3 1 0 5 1 4 号 公 報

30

【 特許文献 3 】 特開 2 0 1 1 - 7 0 6 5 5 号 公 報

【 発明の概要 】

【 発明が解決しようとする課題 】

【 0 0 1 0 】

しかしながら、例えば、客先のコンピュータシステムで異常が発生した場合のように、障害について調査するための環境をすぐ揃えることができない場合がある。

【 0 0 1 1 】

客先のコンピュータシステムがSMPカーネルの異常やSMPカーネルが動作するCPUの異常により動作停止した際には、その場でICEやHWシミュレータ等の調査用専用機器を用意することができず、また、調査専用のCPUを割り当てることができない。このようにSMPカーネルが停止した状態では調査用のログ採取を行なうことができず、原因調査が難航し、問題解決が長期化するという課題がある。

40

【 0 0 1 2 】

また、障害調査のため動作状況を過去に遡り分析する場合に、事前にカーネルやファームウェアに仕込んだログだけでは想定した状態のログしか残らない。そのため、想定外の障害が発生すると分析を行なうための情報を得ることができず、問題解決が長期化してしまうという課題もある。

【 0 0 1 3 】

1つの側面では、本発明は、プロセッサの動作履歴情報を確実に収集できるようにすることを目的とする。

50

【課題を解決するための手段】**【0014】**

このため、この情報処理装置は、複数のプロセッサを備える情報処理装置であって、前記複数のプロセッサのうちの第1のプロセッサが、当該第1のプロセッサの動作履歴情報を収集してログ記憶部に格納するログ収集部と、前記ログ記憶部に格納された前記動作履歴情報を、不揮発性の記憶装置に移動させるログ出力部と、当該第1のプロセッサが低負荷状態から高負荷状態に遷移した場合に、前記複数のプロセッサのうち最も低負荷の第2のプロセッサに、前記ログ出力部としての機能を移管させる移管処理部としての機能を実行する。

【発明の効果】

10

【0015】

一実施形態によれば、プロセッサの動作履歴情報を確実に収集できる。

【図面の簡単な説明】**【0016】**

【図1】実施形態の一例としてのコンピュータシステムのハードウェア構成を示す図である。

【図2】実施形態の一例としてのコンピュータシステムの機能構成を示す図である。

【図3】実施形態の一例としてのコンピュータシステムにおけるログ収集部の処理を説明するための図である。

【図4】実施形態の一例としてのコンピュータシステムにおけるログバッファを説明するための図である。

20

【図5】実施形態の一例としてのコンピュータシステムにおける詳細ログ収集部の処理を説明するための図である。

【図6】実施形態の一例としてのコンピュータシステムにおけるログ出力を説明するための図である。

【図7】実施形態の一例としてのコンピュータシステムにおけるログ出力の概要を説明するための図である。

【図8】実施形態の一例としてのコンピュータシステムにおけるログ収集部による処理を説明するフローチャートである。

【図9】実施形態の一例としてのコンピュータシステムにおけるログ出力処理制御部の移管元としての処理を説明するフローチャートである。

30

【図10】実施形態の一例としてのコンピュータシステムにおけるログ出力処理制御部の移管先としての処理を説明するフローチャートである。

【図11】実施形態の一例としてのコンピュータシステムのログ出力部の処理を説明するフローチャートである。

【図12】実施形態の一例としてのコンピュータシステムの詳細ログ収集部の処理を説明するフローチャートである。

【図13】実施形態の一例としてのコンピュータシステムにおける、移管元のログ出力処理制御部と移管先のログ出力処理制御部との処理を例示するシーケンス図である。

【図14】実施形態の一例としてのコンピュータシステムにおける、移管元のログ出力処理制御部と移管先のログ出力処理制御部との処理を例示するシーケンス図である。

40

【図15】実施形態の一例としてのコンピュータシステムにおける、移管元のログ出力処理制御部と移管先のログ出力処理制御部との処理を例示するシーケンス図である。

【図16】SMPカーネルを説明するための図である。

【発明を実施するための形態】**【0017】**

以下、図面を参照して本情報処理装置および制御プログラムに係る実施の形態を説明する。ただし、以下に示す実施形態はあくまでも例示に過ぎず、実施形態で明示しない種々の変形例や技術の適用を排除する意図はない。すなわち、本実施形態を、その趣旨を逸脱しない範囲で種々変形して実施することができる。又、各図は、図中に示す構成要素のみ

50

を備えるという趣旨ではなく、他の機能等を含むことができる。

【0018】

(A) 構成

図1は実施形態の一例としてのコンピュータシステムのハードウェア構成を示す図、図2はその機能構成を示す図である。

【0019】

本コンピュータシステム1は、図1に示すように、複数(図1に示す例では4つ)のCPU(第1のプロセッサ, 第2のプロセッサ, 第3のプロセッサ)10-1~10-4, メモリ11および記憶装置12を備える。

【0020】

なお、図1に例示するコンピュータシステム1においては、便宜上、マウスやキーボード等の入力装置や、表示装置の図示を省略している。また、コンピュータシステム1には、これら以外の機器をそなえてもよい。また、図1に示す例においては、コンピュータシステム1が4つのCPU10-1~10-4を備えているが、これに限定されるものではなく、3つ以下、または5つ以上のCPU10を備えてもよい。

【0021】

メモリ11はROM(Read Only Memory)及びRAM(Random Access Memory)を含む記憶メモリである。メモリ11のROMには、ソフトウェアプログラム(制御プログラム)やこのプログラム用のデータ類が書き込まれている。メモリ11上のソフトウェアプログラムは、CPU10-1~10-4に適宜読み込まれて実行される。又、メモリ11のRAMは、一次記憶メモリあるいはワーキングメモリとして利用される。

【0022】

メモリ11のRAMの記憶領域は、図2、図3、図5等に示すように、ログバッファ111-1~111-4, デバッグカーネル動作用領域112およびCPU負荷情報格納領域113として用いられる。

【0023】

ログバッファ111-1~111-4には、各CPU10-1~10-4のそれぞれから採取されたログ情報(動作履歴情報; ログ情報, CPU情報, カーネル情報)が格納される。具体的には、ログバッファ111-1には、CPU10-1から収集されたログ情報が格納される。同様に、ログバッファ111-2~111-4には、CPU10-2~10-4から収集されたログ情報がそれぞれ格納される。

【0024】

このように、ログバッファ111-1~111-4は、CPU10に関するログ情報を格納するログ記憶部として機能する。

【0025】

以下、ログバッファを示す符号としては、複数のログバッファのうち1つを特定する必要があるときには符号111-1~111-4を用いるが、任意のログバッファを指すときには符号111を用いる。

【0026】

RAMは揮発性メモリであり、本コンピュータシステム1の電源断時にはログバッファ111に格納されたログ情報は失われる。

【0027】

デバッグカーネル動作用領域112には、デバッグカーネルプログラムが格納されており、CPU10-1~10-4が、このデバッグカーネル動作用領域112からデバッグカーネルプログラムを読み出して実行することで、デバッグカーネルとしての機能が実行される。なお、このデバッグカーネルとしての機能の詳細については後述する。

【0028】

記憶装置12は、ハードディスクドライブ(Hard disk drive: HDD)、SSD(Solid State Drive)、ストレージクラスメモリ(Storage Class Memory: SCM)等の記憶装置であって、種々のデータを格納するものである。記憶装置12は不揮発性を有してお

10

20

30

40

50

り、本コンピュータシステム 1 の電源断時においても、記憶装置 12 に保存されたログ情報は失われることはない。

【0029】

また、記憶装置 12 の記憶領域の一部として設けられたログ用ディスク領域 121 には、後述するログ出力部 102 - 1 ~ 102 - 4 によって、ログバッファ 111 - 1 ~ 111 - 4 から読み出されたログ情報が格納される。

【0030】

CPU 10 - 1 ~ 10 - 4 は、種々の制御や演算を行なう処理装置であり、メモリ 11 に格納された OS (Operating System) やプログラムを実行することにより、種々の機能を実現する。これらの CPU 10 - 1 ~ 10 - 4 は互いに同様の構成を有する。

10

【0031】

本コンピュータシステム 1 は、複数の CPU 10 - 1 ~ 10 - 4 を備えるマルチプロセッシングシステムであり、これらの複数の CPU 10 - 1 ~ 10 - 4 を用いて SMP を実現する。

以下、CPU を示す符号としては、複数の CPU のうち 1 つを特定する必要があるときには符号 10 - 1 ~ 10 - 4 を用いるが、任意の CPU を指すときには符号 10 を用いる。

【0032】

また、以下、CPU 10 - 1 を CPU # 0 という場合がある。同様に、CPU 10 - 2 , 10 - 3 , 10 - 4 を、それぞれ CPU # 1 , # 2 , # 3 という場合がある。

20

【0033】

そして、CPU 10 が、制御プログラムを実行することにより、図 2 に示すように、ログ収集部 101 - 1 ~ 101 - 4 , ログ出力部 102 - 1 ~ 102 - 4 , ログ出力処理制御部 103 - 1 ~ 103 - 4 , 詳細ログ収集部 104 - 1 ~ 104 - 4 および CPU 負荷収集部 105 - 1 ~ 105 - 4 として機能する。

【0034】

ここで、図中、“ - 1 ” , “ - 2 ” , “ - 3 ” , “ - 4 ” の文字を有する各符号は、同じ文字を有する各符号どうしの間に、それぞれ対応関係があることを示しており、具体的には、符号 “ - 1 ” が付された各機能は CPU 10 - 1 によって実現される機能であることを示す。同様に、符号 “ - 2 ” , “ - 3 ” , “ - 4 ” が付された各機能は、それぞれ CPU 10 - 2 , CPU 10 - 3 , CPU 10 - 4 によって実現される機能であることを示す。

30

【0035】

すなわち、ログ収集部 101 - 1 , ログ出力部 102 - 1 , ログ出力処理制御部 103 - 1 , 詳細ログ収集部 104 - 1 および CPU 負荷収集部 105 - 1 としての各機能は、それぞれ CPU 10 - 1 が制御プログラムを実行することにより実現される。

【0036】

同様に、ログ収集部 101 - 2 ~ 101 - 4 , ログ出力部 102 - 2 ~ 102 - 4 , ログ出力処理制御部 103 - 2 ~ 103 - 4 , 詳細ログ収集部 104 - 2 ~ 104 - 4 および CPU 負荷収集部 105 - 2 ~ 105 - 4 としての各機能は、CPU 10 - 2 ~ 10 - 4 が制御プログラムをそれぞれ実行することにより実現される。

40

【0037】

以下、ログ収集部を示す符号としては、複数のログ収集部のうち 1 つを特定する必要があるときには符号 101 - 1 ~ 101 - 4 を用いるが、任意のログ収集部を指すときには符号 101 を用いる。

【0038】

また、以下、ログ出力部を示す符号としては、複数のログ出力部のうち 1 つを特定する必要があるときには符号 102 - 1 ~ 102 - 4 を用いるが、任意のログ出力部を指すときには符号 102 を用いる。

【0039】

50

さらに、以下、ログ出力処理制御部を示す符号としては、複数のログ出力処理制御部のうち1つを特定する必要があるときには符号103-1~103-4を用いるが、任意のログ出力処理制御部を指すときには符号103を用いる。

【0040】

また、以下、詳細ログ収集部を示す符号としては、複数の詳細ログ収集部のうち1つを特定する必要があるときには符号104-1~104-4を用いるが、任意の詳細ログ収集部を指すときには符号104を用いる。

【0041】

さらに、以下、CPU負荷収集部を示す符号としては、複数のCPU負荷収集部のうち1つを特定する必要があるときには符号105-1~105-4を用いるが、任意のCPU負荷収集部を指すときには符号105を用いる。

【0042】

なお、これらのログ収集部101、ログ出力部102、ログ出力処理制御部103、詳細ログ収集部104およびCPU負荷収集部105としての機能を実現するためのプログラム（制御プログラム）は、例えばフレキシブルディスク、CD（CD-ROM、CD-R、CD-RW等）、DVD（DVD-ROM、DVD-RAM、DVD-R、DVD+R、DVD-RW、DVD+RW、HD DVD等）、ブルーレイディスク、磁気ディスク、光ディスク、光磁気ディスク等の、コンピュータ読取可能な記録媒体に記録された形態で提供される。そして、コンピュータはその記録媒体からプログラムを読み取って内部記憶装置または外部記憶装置に転送し格納して用いる。又、そのプログラムを、例えば磁気ディスク、光ディスク、光磁気ディスク等の記憶装置（記録媒体）に記録しておき、その記憶装置から通信経路を介してコンピュータに提供するようにしてもよい。

【0043】

ログ収集部101、ログ出力部102、ログ出力処理制御部103、詳細ログ収集部104およびCPU負荷収集部105としての機能を実現する際には、内部記憶装置（本実施形態ではメモリ11のRAMやROM）に格納されたプログラムがコンピュータのマイクロプロセッサ（本実施形態ではCPU10）によって実行される。このとき、記録媒体に記録されたプログラムをコンピュータが読み取って実行するようにしてもよい。

【0044】

CPU負荷収集部105は、自身が実行されているCPU10（以下、自CPU10という）の負荷を表す情報を収集する。なお、CPU10の負荷は、例えばCPU使用率を用いて表すことができる。CPU10の負荷状態は既知の種々の手法で収集することができ、その説明は省略する。また、以下、自CPU10以外の他のCPU10を他CPU10という場合がある。

【0045】

また、CPU負荷収集部105は、収集した自CPU10の負荷状態を示す情報（CPU負荷情報）を、そのCPU10を特定するための情報（例えば、CPU番号：詳細は後述）と関連付けて、メモリ11のCPU負荷情報格納領域113に格納する。

【0046】

なお、メモリ11においてはCPU10毎にCPU負荷情報格納領域113を設け、CPU負荷収集部105は、収集したCPU負荷情報を、自CPU10に対応して設けられたCPU負荷情報格納領域113に格納してもよい。

【0047】

ログ収集部101は、自CPU10についてのログ情報を収集する。ログ情報は、CPU10の実行履歴（動作履歴）を表す情報（動作履歴情報）であって、例えば、CPU10のレジスタ（図示省略）に格納された情報（CPUレジスタ情報）や、カーネルのスケジューラ情報である。なお、ログ収集部101が収集するログ情報は、これらに限定されるものではなく、適宜変形して実施することができる。

【0048】

ログ収集部101は、CPU10から収集したログ情報を、ログバッファ111に格納

10

20

30

40

50

する。以下、CPU 10から収集したログ情報を、ログバッファ111に格納することをログ収集という場合がある。

【0049】

図3は実施形態の一例としてのコンピュータシステム1におけるログ収集部101の処理を説明するための図、図4はログバッファ111を説明するための図である。

【0050】

前述の如く、ログバッファ111は、CPU 10毎に備えられている。以下、ログバッファ111-1をCPU 10#0用ログバッファ111-1という場合がある。同様に、ログバッファ111-2, 111-3, 111-4を、それぞれCPU#1用ログバッファ111-2, CPU#2用ログバッファ111-3, CPU#3用ログバッファ111-4という場合がある。

10

【0051】

また、ログバッファ111-1~111-4は、それぞれリングバッファ構造を有し、複数個(本実施形態においては例えば1000個)分のログを保存する領域を持つ。

【0052】

図4においては、CPU#0用ログバッファ111-1を例示しており、0~999までの1000個の格納領域に、それぞれログ収集部101-1によって収集されたログ情報(CPU情報およびカーネル情報)が順次格納される。

【0053】

なお、ログバッファ111において、ログ収集部101によってログ情報が格納される格納領域の位置(先頭位置)は、ログ格納ポインタによって特定される。また、ログバッファ111において、後述するログ出力部102によってログ情報が読み出される格納領域の位置(先頭位置)は、ログ出力ポインタによって特定される。

20

【0054】

ログ収集部101は、所定の第1の間隔(例えば、1ms毎)でログ収集を行なう。具体的には、ログ収集部101は、例えば、割込みコントローラ1011を用いて、定期的(例えば、1ms毎)にCPU 10に対して割込みを発生させる(図3中の矢印P1参照)。

【0055】

CPU 10は、割込みコントローラ1011から割込みが入力されると、ログ収集用の割込みハンドラ1012を実行する(図3中の矢印P2参照)。

30

【0056】

このログ収集用の割込みハンドラ1012は、カーネル起動時に登録される。割込みハンドラ1012は、CPU 10にログ収集部101としての機能を実行させる(図3中の矢印P3参照)。すなわち、ログ収集部101は、CPU情報およびカーネル情報の採取を行ない、CPU 10に割り当てられたログバッファ111へ出力させる。

【0057】

また、ログ収集部101は、所定のタイミングで後述するログ出力処理制御部103に対して通知を行ない、ログ出力部102によるログ出力(詳細は後述)を行なわせる(図3中の矢印P4参照)。

40

【0058】

例えば、ログ収集部101は、ログ収集を100回行なったタイミングで、ログ出力処理制御部103にログ出力を行なわせるための通知を発行する。

【0059】

また、ログ収集部101は、ログ収集を行なった後に、自身が実行されているCPU 10に詳細ログ収集部104としての機能を実行させる。

【0060】

詳細ログ収集部104は、本コンピュータシステム1に備えられた複数のCPU 10のいずれかにおいて障害が発生した場合に機能する。

【0061】

50

図 5 は実施形態の一例としてのコンピュータシステム 1 における詳細ログ収集部 104 の処理を説明するための図である。

【0062】

詳細ログ収集部 104 としての機能は、上述したログ収集部 101 によって実行を開始される（図 5 中の符号（1）参照）。

【0063】

各詳細ログ収集部 104 は、自身が実行されている CPU 10 以外の他の CPU 10 のそれぞれの CPU ステータスを監視する（図 5 中の符号（2）参照）。

【0064】

そして、いずれかの CPU（第 3 のプロセッサ）10 において何らかの異常が発生した場合に、各 CPU 10 で実行される詳細ログ収集部 104 のうち、最初にこの異常発生を検知した詳細ログ収集部 104 が、自身が実行されている CPU 10（自 CPU 10）と当該異常が発生している CPU（第 3 のプロセッサ）10 とを除いた、他の CPU 10 を停止させる。

10

【0065】

すなわち、詳細ログ収集部 104 は、複数の CPU 10 のうちの CPU（第 3 のプロセッサ）10 に異常が検出された場合に、これらの複数の CPU 10 のうち自 CPU（第 1 プロセッサ）10 および異常が検出された CPU 10 以外の CPU 10 の動作を停止させる停止処理部として機能する。

【0066】

20

異常が発生している CPU 10 を除いた他の CPU 10 を停止させることで、リングバッファ構成を有するログバッファ 111 のログ情報が、これらの CPU 10 の動作に伴って生成されるログ情報により上書き消去されてしまうことを阻止することができる。

【0067】

また、詳細ログ収集部 104 が、自 CPU 10 を停止対象から除外することで、自 CPU 10 において、詳細ログ収集部 104 としての機能やデバッグカーネルを実行することができる。

【0068】

以下に、図 5 に示す例において、CPU # 1 において異常が発生したことを CPU # 3 の詳細ログ収集部 104 - 4 が最初に検知した例について示す。

30

【0069】

詳細ログ収集部 104 - 4 は、正常に動作中の CPU # 0 , # 2 を停止させるとともに（図 5 の符号（3）参照）、デバッグカーネル動作領域 112 からデバッグカーネルプログラムを読み出して、自 CPU 10 - 4 に実行（起動）させる（図 5 中の符号（4）参照）。

【0070】

ユーザは、実行されたデバッグカーネルの機能を用いて、種々の入力操作等を行ない、異常が検出された CPU # 3 についての、メモリダンプや周辺デバイスのレジスタダンプ等を収集し、また、本コンピュータシステム 1 の再起動を行なう。

【0071】

40

ログ出力部 102 は、ログバッファ 111 に格納されたログ情報を読み出してログ用ディスク領域 121 に格納（出力）する。以下、ログバッファ 111 に格納されたログ情報を読み出してログ用ディスク領域 121 に格納（出力）することをログ出力という場合がある。

【0072】

前述の如く、RAM は揮発性メモリであるので、本コンピュータシステム 1 の電源断時にはログバッファ 111 に格納されたログ情報は失われる。これに対して、記憶装置 12 は不揮発性を有しており、本コンピュータシステム 1 の電源断時においても、記憶装置 12 に保存されたログ情報は失われることはない。

【0073】

50

本コンピュータシステム 1 においては、ログ出力部 1 0 2 が、ログバッファ 1 1 1 に格納されたログ情報を、記憶装置 1 2 のログ用ディスク領域 1 2 1 に移動（出力）されることで、ログ情報が保護される。

【 0 0 7 4 】

ログ出力部 1 0 2 は、後述するログ出力処理制御部 1 0 3 からの指示（制御）に従って、ログ出力を行なう。

【 0 0 7 5 】

例えば、ログ出力部 1 0 2 は、ログ出力処理制御部 1 0 3 からの指示に従って、自 CPU 1 0 に対応するログバッファ 1 1 1 に格納されたログ情報を記憶装置 1 2 のログ用ディスク領域 1 2 1 に格納（出力）する。

10

【 0 0 7 6 】

また、ログ出力部 1 0 2 は、ログ出力処理制御部 1 0 3 から、CPU 1 0 を特定する識別情報（以下、対象 CPU 番号という）を受け取り、この対象 CPU 番号によって特定される CPU 1 0 に対応するログバッファ 1 1 1 からログ情報を読み出してログ用ディスク領域 1 2 1 に出力する。

【 0 0 7 7 】

この際、ログ出力部 1 0 2 は、ログバッファ 1 1 1 において、ログ出力ポインタによって特定される位置から、ログ格納ポインタによって特定される位置までの範囲の情報（ログ情報）をログ用ディスク領域 1 2 1 に出力する。また、ログ出力部 1 0 2 は、ログバッファ 1 1 1 のログ出力ポインタの値をログ格納ポインタの値を用いて上書きすることで更新する。

20

【 0 0 7 8 】

なお、ログ出力部 1 0 2 は、ログ出力処理制御部 1 0 3 の指示に従って、前述した第 1 の間隔（例えば、1 m s 毎）よりも長い第 2 の間隔（例えば、1 0 m s 毎）で、ログ出力を行なう。

【 0 0 7 9 】

例えば、ログ出力部 1 0 2 は、ログ出力処理制御部 1 0 3 の指示に従って、ログ収集部 1 0 1 がログ情報の収集を 1 0 0 回実行する毎に、ログ出力を行なう。

【 0 0 8 0 】

図 6 は実施形態の一例としてのコンピュータシステム 1 におけるログ出力を説明するための図である。

30

【 0 0 8 1 】

ログ収集部 1 0 1 が、例えばログ収集を 1 0 0 回行なったタイミングでログ出力処理制御部 1 0 3 に対して通知を行なうと（図 6 中の矢印 P 5 参照）、ログ出力処理制御部 1 0 3 はログ出力部 1 0 2 にログ出力を実行させる（図 6 中の矢印 P 6 参照）。

【 0 0 8 2 】

ログ出力部 1 0 2 は、ログ出力処理制御部 1 0 3 の指示に従ってログ出力を行なう。すなわち、ログ出力部 1 0 2 は、ログバッファ 1 1 1 に格納されたログ情報を記憶装置 1 2 のログ用ディスク領域 1 2 1 に出力する（図 6 中の矢印 P 7 参照）。

【 0 0 8 3 】

40

そして、本コンピュータシステム 1 においては、ログ出力部 1 0 2 は、ログ出力処理制御部 1 0 3 からの指示に従い、自 CPU 1 0 のログ情報の他に、他の CPU 1 0 のログ情報の出力を行なうこともある。

【 0 0 8 4 】

ログ出力処理制御部 1 0 3 は、ログ出力部 1 0 2 によるログ出力を制御する。

【 0 0 8 5 】

ログ出力処理制御部 1 0 3 は、自 CPU 1 0 が低負荷状態から高負荷状態に変化した場合に、自 CPU 1 0 におけるログ情報の出力を、低負荷状態の他の CPU 1 0 のログ出力部 1 0 2 に実行させる。

【 0 0 8 6 】

50

すなわち、ログ出力処理制御部 103 は、自 CPU 10 が低負荷状態から高負荷状態に遷移した場合に、本コンピュータシステム 11 に備えられた複数の CPU 10 - 1 ~ 10 - 4 のうち最も低負荷の他の CPU 10 に、ログ出力部 102 としての機能を移管させる移管処理部として機能する。

【0087】

以下、自 CPU 10 におけるログ情報の出力を他の CPU 10 のログ出力部 102 に実行させることをログ出力を移管するという。これに対して、他の CPU 10 のログ情報を出力することを、ログ出力を代替実行するという。

【0088】

そして、他 CPU 10 にログ出力を移管する側を移管元という場合があり、他 CPU 10 からログ情報を移管された側を移管先という場合がある。

10

【0089】

また、ログ出力処理制御部 103 は、他 CPU 10 にログ出力を移管している状態において、自 CPU 10 の負荷状態が高負荷から低負荷に変化した場合に、他 CPU 10 に移管していたログ情報の出力を自 CPU 10 のログ出力部 102 に実行させるよう切り替える。以下、他 CPU 10 に移管していたログ出力を、自 CPU 10 において実行するように切り替えることを、移管を解除する、もしくはログ出力を取り戻すという場合がある。

【0090】

ログ出力処理制御部 103 は、複数 CPU 10 間における、ログ出力の移管や移管の解除を制御するために、ログ出力対象情報 114 および代替 CPU 情報 115 を管理する。

20

【0091】

ログ出力対象情報 114 は、自 CPU 10 のログ出力部 102 がログ出力を行なう対象である CPU 10 を特定する情報である。

【0092】

本実施形態においては、このログ出力対象情報 114 として、CPU 10 を特定する CPU 番号を用いる例を示す。以下、CPU 番号 0, 1, 2, 3 が、CPU # 0, # 1, # 2, # 3 をそれぞれ示すものとする。

【0093】

そして、ある CPU 10 において、他 CPU 10 のログ出力を代替実行する場合には、このログ出力対象情報 114 に、移管元の CPU 10 の CPU 番号が登録される。

30

【0094】

ログ出力部 102 は、このログ出力対象情報 114 に登録されている CPU 番号の CPU 10 についてログ出力を行なう。

【0095】

以下、ログ出力対象情報 114 に登録されている CPU 番号をログ出力対象番号という場合がある。

【0096】

代替 CPU 情報 115 は、自 CPU 10 についてのログ出力の移管先の CPU 10 を特定するための情報である。本実施形態においては、この代替 CPU 情報 115 としても CPU 番号を用いる。すなわち、代替 CPU 情報 115 は、自 CPU 10 についてのログ出力の移管先のプロセッサ (10) を示す移管先プロセッサ情報に相当する。

40

【0097】

この代替 CPU 情報 115 に登録されている CPU 番号の CPU 10 が、自 CPU 10 のログ情報の出力 (ログ出力) を行なう。

【0098】

以下、代替 CPU 情報 115 に登録されている CPU 番号を、代替 CPU 番号という場合がある。

【0099】

ログ出力処理制御部 103 は、他 CPU 10 にログ出力を移管する移管元としての処理と、他 CPU 10 からログ情報を移管された移管先としての処理とを備える。

50

【 0 1 0 0 】

(1) 移管元としてのログ出力処理制御部 1 0 3 の処理

移管元の C P U 1 0 において、ログ出力処理制御部 1 0 3 は、C P U 負荷収集部 1 0 5 が前回採取した自 C P U 1 0 の C P U 負荷情報を C P U 負荷情報格納領域 1 1 3 から取得する。また、ログ出力処理制御部 1 0 3 は、C P U 負荷収集部 1 0 5 に現在の C P U 負荷情報を収集させる。

【 0 1 0 1 】

ログ出力処理制御部 1 0 3 は、このようにして取得した C P U 負荷情報の変化に応じて、処理を切り換える。例えば、以下の (i) ~ (iv) の各場合について、ログ出力処理制御部 1 0 3 の処理について示す。

10

【 0 1 0 2 】

(i) 現在の C P U 負荷が低負荷、且つ、前回の C P U 負荷が低負荷の場合

ログ出力処理制御部 1 0 3 は、ログ出力部 1 0 2 に、ログ出力対象情報 1 1 4 として保存された C P U 番号が示す C P U 1 0 に対応するログバッファ 1 1 1 に格納されたログ情報をログ用ディスク領域 1 2 1 に格納させる。

【 0 1 0 3 】

(ii) 現在の C P U 負荷が高負荷、且つ、前回の C P U 負荷が低負荷の場合

ログ出力処理制御部 1 0 3 は、C P U 負荷情報格納領域 1 1 3 を参照して他 C P U 1 0 の C P U 負荷をそれぞれ取得し、負荷が最も低い (最低負荷) の C P U 1 0 を特定する。そして、この最低負荷の C P U 1 0 の C P U 番号を代替 C P U 情報 1 1 5 に保存する。すなわち、負荷が最も低い C P U 1 0 をログ出力の移管先として決定する。

20

【 0 1 0 4 】

また、ログ出力処理制御部 1 0 3 は、代替 C P U 情報 1 1 5 として保存された C P U 番号が示す C P U 1 0 のログ出力処理制御部 1 0 3 に対して、ログ出力を移管することを示す通知 (移管通知) を送信する。

【 0 1 0 5 】

ログ出力処理制御部 1 0 3 は、移管通知として、例えば、ログ出力対象情報 1 1 4 として保存された C P U 番号 (ログ出力対象番号) を通知 (送信) する。なお、移管通知は、これに限定されるものではなく、例えば、移管通知を示す何らかの信号を送信することで行なってもよい。ログ出力処理制御部 1 0 3 は、このような移管通知を示す信号とともに、ログ出力対象番号を送信することが望ましい。

30

【 0 1 0 6 】

これにより、移管元の C P U 1 0 は、最低負荷の他 C P U 1 0 に対して、自身のログ出力対象情報 1 1 4 に登録された C P U 番号 (ログ出力対象番号) の C P U 1 0 のログ出力を移管する。

【 0 1 0 7 】

また、ログ出力処理制御部 1 0 3 は、移管先の他 C P U 1 0 に通知したログ出力対象番号 (C P U 番号) に、自 C P U 1 0 および移管先 C P U 1 0 とは別他 C P U 1 0 の C P U 番号が含まれている場合には、この他 C P U 1 0 のログ出力処理制御部 1 0 3 に対して、代替 C P U 情報 1 1 5 に登録されている C P U 番号を通知する。

40

【 0 1 0 8 】

これにより、自 C P U 1 0 が移管先 C P U 1 0 として機能していた場合に、その移管元他 C P U 1 0 のログ出力処理制御部 1 0 3 に対して、その移管元 C P U 1 0 のログ出力が更に移管されたことを通知する。

【 0 1 0 9 】

(iii) 現在の C P U 負荷が低負荷、且つ、前回の C P U 負荷が高負荷の場合

ログ出力処理制御部 1 0 3 は、代替 C P U 情報 1 1 5 として保存された C P U 番号が示す C P U 1 0 (移管先 C P U 1 0) に対して、移管していたログ出力の取り戻しを行なう旨のメッセージを通知する。このログ出力の取り戻しを行なう旨のメッセージは、移管先の C P U 1 0 に対して、ログ出力の移管を解除することを通知する。以下、ログ出力の取

50

り戻しを行なう旨のメッセージの通知を移管解除通知という場合がある。また、ログ出力を取り戻すことを、ログ出力対象番号を取り戻すと表現する場合がある。

【 0 1 1 0 】

他 C P U 1 0 に対してログ出力の移管解除通知を送信したログ出力処理制御部 1 0 3 は、代替 C P U 情報 1 1 5 として保存されている C P U 番号を削除するとともに、ログ出力対象情報 1 1 4 に自 C P U 1 0 の C P U 番号を設定する。

【 0 1 1 1 】

これにより、ログ出力処理制御部 1 0 3 は他 C P U 1 0 に移管していたログ出力が自 C P U 1 0 に戻される。すなわち、ログ出力の移管が解除される。

【 0 1 1 2 】

そして、ログ出力処理制御部 1 0 3 は、自 C P U 1 0 のログ出力部 1 0 2 に対して、ログ出力対象情報 1 1 4 に登録された自 C P U 1 0 のログ出力を実行させる。

【 0 1 1 3 】

(iv) 現在の C P U 負荷が高負荷、且つ、前回の C P U 負荷が高負荷の場合何も処理せず、処理を終了する。

【 0 1 1 4 】

(2) 移管先としてのログ出力処理制御部 1 0 3 の処理

移管先の C P U 1 0 において、ログ出力の移管解除通知を受信した場合、すなわち、ログ出力を取り戻す旨のメッセージ (取り戻し要求) を受信した場合には、ログ出力処理制御部 1 0 3 は、ログ出力対象情報 (第 2 のログ出力対象情報) 1 1 4 から取り戻し要求の送信元の C P U 1 0 の C P U 番号を削除する。これにより、当該 C P U 1 0 のログ出力部 1 0 2 によるログ出力対象から、取り戻し要求の送信元の C P U 1 0 のログ情報が除外される。すなわち、当該 C P U 1 0 に対するログ出力の移管が解消される。

【 0 1 1 5 】

また、移管先の C P U 1 0 において、他の C P U 1 0 のログ出力処理制御部 1 0 3 からログ出力対象番号を通知するメッセージを受信した場合には、受け取った C P U 番号 (ログ出力対象番号) を、自 C P U 1 0 のログ出力対象情報 1 1 4 へ追加保存する。

【 0 1 1 6 】

また、移管先の C P U 1 0 において、代替 C P U 番号を通知するメッセージを受信した場合には、受け取った C P U 番号を代替 C P U 情報 1 1 5 に保存する。

【 0 1 1 7 】

(B) 動作

先ず、上述の如く構成された実施形態の一例としてのコンピュータシステム 1 におけるログ出力の概要を、図 7 を用いて説明する。

【 0 1 1 8 】

各 C P U 1 0 において、ログ収集部 1 0 1 が、それぞれ自 C P U 1 0 のログ情報を収集して、ログバッファ 1 1 1 における対応する領域に格納する (図 7 中の符号 (1) 参照) 。

【 0 1 1 9 】

各 C P U 1 0 において、ログ出力処理制御部 1 0 3 は、自 C P U 1 0 の負荷情報を C P U 負荷情報格納領域 1 1 3 から採取する。ログ出力処理制御部 1 0 3 は自 C P U 1 0 の負荷状態に基づき、ログ出力を自 C P U 1 0 のログ出力部 1 0 2 により行なうかを判断する (図 7 中の符号 (2) 参照) 。

【 0 1 2 0 】

自 C P U 1 0 の C P U 負荷が高負荷であって、且つ、前回測定した C P U 負荷が低負荷である場合に、ログ出力処理制御部 1 0 3 は、負荷が最も低い他 C P U 1 0 にログ出力を移管する (図 7 中の符号 (3) 参照) 。

【 0 1 2 1 】

なお、例えば、C P U 負荷の値が所定の閾値以上である場合に、自 C P U 1 0 の C P U 負荷が高負荷であると判断することができる。また、C P U 負荷の値が所定の閾値未満で

10

20

30

40

50

ある場合に、自CPU10のCPU負荷が低負荷であると判断することができる。

【0122】

その後、ログ出力部102によるログ出力が行なわれる。すなわち、ログバッファ111のログ情報が、記憶装置12のログ用ディスク領域121に複写（退避）される（図7中の符号（4）参照）。

【0123】

なお、上記ログ出力は、ログ出力が他CPU10に移管された場合には、その他CPU10のログ出力部102によって行なわれる。また、ログ出力が他CPU10に移管されない場合には、自CPU10のログ出力部102によって行なわれる。

【0124】

次に、実施形態の一例としてのコンピュータシステム1におけるログ収集部101による処理を、図8に示すフローチャート（ステップA1～A10）に従って説明する。

【0125】

本処理は、例えば、割込みコントローラ1011からCPU10に対して割込みが入力されると、CPU10がログ収集用の割込みハンドラ1012を実行することで開始される。

【0126】

ステップA1において、ログ収集部101は、ログ情報として、CPU10からCPU情報（例えば、CPUレジスタ情報）を取得する。

【0127】

ステップA2において、ログ収集部101は、ログ情報として、CPU10からカーネル情報（例えばスケジューラ情報）を取得する。

【0128】

なお、これらのステップA1，A2の実行順序はこれに限定されるものではなく、ステップA2の処理をステップA1より先に行なってもよく、また、これらの処理を同時に行なってもよい。

【0129】

ステップA3において、ログ収集部101は、ステップA1，A2において取得した情報（ログ情報）を、ログバッファ111におけるログ格納ポインタによって示される領域に格納する。

【0130】

ステップA4において、ログ収集部101は、ログバッファ111におけるログ格納ポインタを、ログ情報の格納後の状態に合わせて更新する。

【0131】

ステップA5において、ログ収集部101は、ログ出力実行カウンタの値をインクリメントする。ログ出力実行カウンタは、ログ収集部101により行なわれたログ収集の実行回数を計数するためのカウンタである。このログ出力実行カウンタは、例えば、メモリ11のRAM等に格納される。

【0132】

ステップA6において、ログ収集部101は、ログ出力実行カウンタのカウント値が100以上であるかを確認する。

【0133】

ログ出力実行カウンタのカウント値が100以上である場合には（ステップA6のYesルート参照）、ステップA7に移行する。ステップA7において、ログ収集部101はログ出力処理制御部103の移管元としての機能を実行させる。なお、このログ出力処理制御部103の移管元としての機能の詳細は、図9を用いて後述する。

【0134】

ステップA8において、ログ収集部101は、ログ出力実行カウンタを0クリアする。その後、ステップA9において、ログ収集部101は、割込みコントローラ1011による割込みステータスをクリアする。

10

20

30

40

50

また、ステップ A 6 における確認の結果、ログ出力実行カウンタのカウンタ値が 1 0 0 未満である場合にも（ステップ A 6 の N o ルート参照）、ステップ A 9 に移行する。

【 0 1 3 5 】

その後、ステップ A 1 0 において、ログ収集部 1 0 1 は、詳細ログ収集部 1 0 4 による処理を実行させ、処理を終了する。なお、詳細ログ収集部 1 0 4 による処理の詳細は、図 1 2 を用いて後述する。

【 0 1 3 6 】

次に、実施形態の一例としてのコンピュータシステム 1 におけるログ出力処理制御部 1 0 3 の移管元としての処理を、図 9 に示すフローチャート（ステップ B 1 ~ B 1 5 ）に従って説明する。

【 0 1 3 7 】

ステップ B 1 において、ログ出力処理制御部 1 0 3 は、C P U 負荷情報格納領域 1 1 3 から前回の自 C P U 1 0 の C P U 負荷情報を取得する。

【 0 1 3 8 】

ステップ B 2 において、ログ出力処理制御部 1 0 3 は、C P U 負荷収集部 1 0 5 を介して最新の C P U 負荷情報を取得する。

【 0 1 3 9 】

ステップ B 3 において、ログ出力処理制御部 1 0 3 は、ステップ B 2 において収集した C P U 負荷情報を C P U 負荷情報格納領域 1 1 3 に保存する。

【 0 1 4 0 】

ステップ B 4 において、ログ出力処理制御部 1 0 3 は、ステップ B 2 において収集した C P U 負荷情報に基づき自 C P U 1 0 が高負荷状態であるかを確認する。

【 0 1 4 1 】

確認の結果、自 C P U 1 0 が高負荷状態である場合には（ステップ B 4 の Y e s ルート参照）、ステップ B 5 に移行する。

【 0 1 4 2 】

ステップ B 5 において、ログ出力処理制御部 1 0 3 は、ステップ B 1 において取得した、自 C P U 1 0 の前回の C P U 負荷情報に基づき、前回の C P U 負荷が高負荷状態であったかを確認する。

【 0 1 4 3 】

確認の結果、前回の C P U 負荷も高負荷状態であった場合には（ステップ B 5 の Y e s ルート参照）、現在の C P U 負荷が高負荷、且つ、前回の C P U 負荷が高負荷の場合の処理が行なわれる。すなわち、ログ出力処理制御部 1 0 3 はそのまま処理を終了する。

【 0 1 4 4 】

また、前回の C P U 負荷は高負荷状態でなかった場合には（ステップ B 5 の N o ルート参照）、ステップ B 6 に移行する。ステップ B 6 ~ B 1 0 においては、現在の C P U 負荷が高負荷、且つ、前回の C P U 負荷が低負荷の場合の処理が行なわれる。

【 0 1 4 5 】

ステップ B 6 において、ログ出力処理制御部 1 0 3 は、C P U 負荷情報格納領域 1 1 3 から、本コンピュータシステム 1 に備えられた他の C P U 1 0 の C P U 負荷をそれぞれ取得する。

【 0 1 4 6 】

ステップ B 7 において、ログ出力処理制御部 1 0 3 は、これらの複数の他の C P U 1 0 のうち、負荷が最も低い（最低負荷）の C P U 1 0 を特定する。そして、この最低負荷の C P U 1 0 の C P U 番号を代替 C P U 情報 1 1 5 に保存する。すなわち、負荷が最も低い C P U 1 0 をログ出力の移管先として決定する。

【 0 1 4 7 】

ステップ B 8 において、ログ出力処理制御部 1 0 3 は、代替 C P U 情報 1 1 5 として保存された C P U 番号が示す C P U 1 0 のログ出力処理制御部 1 0 3 に対して、ログ出力対象情報 1 1 4 として保存された C P U 番号（ログ出力対象番号）を通知するメッセージを

10

20

30

40

50

送信する。

【 0 1 4 8 】

これにより、最低負荷の他 C P U 1 0 に対して、自身のログ出力対象情報 1 1 4 に登録された C P U 番号 (ログ出力対象番号) の C P U 1 0 のログ出力を移管する。

【 0 1 4 9 】

ステップ B 9 において、ログ出力処理制御部 1 0 3 は、他 C P U 1 0 に通知した C P U 番号に他 C P U 1 0 の C P U 番号が含まれているかを確認する。

【 0 1 5 0 】

確認の結果、他 C P U 1 0 に通知した C P U 番号に他 C P U 1 0 の C P U 番号が含まれている場合には (ステップ B 9 の Y e s ルート参照) 、ステップ B 1 0 に移行する。

10

【 0 1 5 1 】

ステップ B 1 0 においては、ログ出力処理制御部 1 0 3 は、この他 C P U 1 0 のログ出力処理制御部 1 0 3 に対して、代替 C P U 情報 1 1 5 に登録されている C P U 番号を通知するメッセージを送信する。

【 0 1 5 2 】

これにより、自身にログ出力を移管していた移管元の他 C P U 1 0 のログ出力処理制御部 1 0 3 に、その C P U 1 0 のログ出力が移管されたことを通知する。その後、処理を終了する。

【 0 1 5 3 】

また、ステップ B 9 における確認の結果、他 C P U 1 0 に通知した C P U 番号に他 C P U 1 0 の C P U 番号が含まれていない場合も (ステップ B 9 の N o ルート参照) 、処理を終了する。

20

【 0 1 5 4 】

ステップ B 4 における確認の結果、自 C P U 1 0 が高負荷状態でない場合には (ステップ B 4 の N o ルート参照) 、ステップ B 1 1 において、ログ出力処理制御部 1 0 3 は、ステップ B 1 において取得した、自 C P U 1 0 の前回の C P U 負荷情報に基づき、前回の C P U 負荷が高負荷状態であったかを確認する。

【 0 1 5 5 】

確認の結果、前回の C P U 負荷が低負荷状態であった場合には (ステップ B 1 1 の N o ルート参照) 、ステップ B 1 5 に移行する。ステップ B 1 5 においては、現在の C P U 負荷が低負荷、且つ、前回の C P U 負荷が低負荷の場合の処理が行なわれる。

30

【 0 1 5 6 】

ステップ B 1 5 において、ログ出力処理制御部 1 0 3 は、ログ出力部 1 0 2 に、ログ出力対象情報 1 1 4 として保存された C P U 番号の C P U 1 0 について、ログ出力を行なう。すなわち、C P U 番号が示す C P U 1 0 に対応するログバッファ 1 1 1 に格納されたログ情報をログ用ディスク領域 1 2 1 に格納させる。その後、処理を終了する。

【 0 1 5 7 】

一方、ステップ B 1 1 における確認の結果、前回の C P U 負荷が高負荷状態であった場合には (ステップ B 1 1 の Y e s ルート参照) 、ステップ B 1 2 に移行する。ステップ B 1 2 ~ B 1 5 においては、現在の C P U 負荷が低負荷、且つ、前回の C P U 負荷が高負荷の場合の処理が行なわれる。

40

【 0 1 5 8 】

ステップ B 1 2 において、ログ出力処理制御部 1 0 3 は、代替 C P U 情報 1 1 5 として保存された C P U 番号が示す C P U 1 0 (移管先 C P U 1 0) に対して、ログ出力を取り戻す旨のメッセージを通知する。すなわち、ログ出力の移管解除通知が送信される。

【 0 1 5 9 】

ステップ B 1 3 において、ログ出力処理制御部 1 0 3 は、代替 C P U 情報 1 1 5 として保存された C P U 番号を削除する。

【 0 1 6 0 】

また、ステップ B 1 4 において、ログ出力処理制御部 1 0 3 は、ログ出力対象情報 1 1

50

4 に自 C P U 1 0 の C P U 番号を設定する。

【 0 1 6 1 】

その後、ステップ B 1 5 に移行した後、処理を終了する。

【 0 1 6 2 】

次に、実施形態の一例としてのコンピュータシステム 1 におけるログ出力処理制御部 1 0 3 の移管先としての処理を、図 1 0 に示すフローチャート（ステップ C 1 ~ C 7 ）に従って説明する。

【 0 1 6 3 】

ステップ C 1 において、ログ出力処理制御部 1 0 3 は、他 C P U 1 0 のログ出力処理制御部 1 0 3 からメッセージを受信する。

10

【 0 1 6 4 】

ステップ C 2 において、ログ出力処理制御部 1 0 3 は、受信したメッセージがログ出力対象番号の取り戻しを通知するものであるかを確認する。

【 0 1 6 5 】

確認の結果、受信したメッセージがログ出力対象番号の取り戻しを通知するものである場合には（ステップ C 2 の Y e s ルート参照）、ステップ C 3 に移行する。

【 0 1 6 6 】

ステップ C 3 においては、ログ出力処理制御部 1 0 3 は、ログ出力対象情報 1 1 4 からメッセージの送信元 C P U 1 0 の C P U 番号を削除して、処理を終了する。

【 0 1 6 7 】

20

ステップ C 2 における確認の結果、受信したメッセージがログ出力対象番号の取り戻しを通知するものでない場合には（ステップ C 2 の N o ルート参照）、ステップ C 4 に移行する。

【 0 1 6 8 】

ステップ C 4 において、ログ出力処理制御部 1 0 3 は、受信したメッセージがログ出力対象番号を通知するものであるか、すなわち、ログ出力対象番号を受け渡すものであるかを確認する。

【 0 1 6 9 】

確認の結果、受信したメッセージがログ出力対象番号を通知するものである場合には（ステップ C 4 の Y e s ルート参照）、ステップ C 5 に移行する。

30

【 0 1 7 0 】

ステップ C 5 においては、ログ出力処理制御部 1 0 3 は、他 C P U 1 0 から受け取った C P U 番号（ログ出力対象番号）を、ログ出力対象情報 1 1 4 に追加保存して、処理を終了する。

【 0 1 7 1 】

ステップ C 4 における確認の結果、ログ出力処理制御部 1 0 3 は、受信したメッセージがログ出力対象番号を通知するものでない場合（ステップ C 4 の N o ルート参照）、ステップ C 6 に移行する。

【 0 1 7 2 】

ステップ C 6 においては、ログ出力処理制御部 1 0 3 は、他 C P U 1 0 から受信したメッセージが代替 C P U 番号を通知するものであるかを確認する。

40

【 0 1 7 3 】

確認の結果、受信したメッセージが代替 C P U 番号を通知するものである場合には（ステップ C 6 の Y e s ルート参照）、ステップ C 7 に移行する。

【 0 1 7 4 】

ステップ C 7 においては、ログ出力処理制御部 1 0 3 は、他 C P U 1 0 から受け取った C P U 番号（代替 C P U 番号）を、代替 C P U 情報 1 1 5 に保存して、処理を終了する。

【 0 1 7 5 】

また、ステップ C 6 における確認の結果、受信したメッセージが代替 C P U 番号を通知するものでない場合には（ステップ C 6 の N o ルート参照）、処理を終了する。

50

【 0 1 7 6 】

次に、実施形態の一例としてのコンピュータシステム 1 のログ出力部 1 0 2 の処理を、図 1 1 に示すフローチャート（ステップ D 1 ～ D 3 ）に従って説明する。

【 0 1 7 7 】

ステップ D 1 において、ログ出力部 1 0 2 は、ログ出力処理制御部 1 0 3 からログ出力対象番号を受け取る。

【 0 1 7 8 】

ステップ D 2 において、ログ出力部 1 0 2 は、ログ出力対象番号で示される CPU 1 0 に対応するログバッファ 1 1 1 において、ログ出力ポイントによって特定される位置から、ログ格納ポイントによって特定される位置までの間の情報（ログ情報）をログ用ディスク領域 1 2 1 に出力する。

10

【 0 1 7 9 】

ステップ D 3 において、ログ出力部 1 0 2 は、ログバッファ 1 1 1 のログ出力ポイントの値をログ格納ポイントの値を用いて上書きすることで更新し、処理を終了する。

【 0 1 8 0 】

次に、実施形態の一例としてのコンピュータシステム 1 の詳細ログ収集部 1 0 4 の処理を、図 1 2 に示すフローチャート（ステップ E 1 ～ E 9 ）に従って説明する。

【 0 1 8 1 】

ステップ E 1 において、詳細ログ収集部 1 0 4 は、他 CPU 1 0 のそれぞれの CPU ステータスを監視する。

20

【 0 1 8 2 】

ステップ E 2 において、詳細ログ収集部 1 0 4 は、他 CPU 1 0 において異常が発生しているかを確認する。

【 0 1 8 3 】

確認の結果、他 CPU 1 0 に異常が発生していない場合には（ステップ E 2 の N o ルート参照）、処理を終了する。

【 0 1 8 4 】

一方、ステップ E 2 における確認の結果、他 CPU 1 0 のいずれかにおいて異常が発生している場合には（ステップ E 2 の Y e s ルート参照）、ステップ E 3 に移行する。

【 0 1 8 5 】

ステップ E 3 においては、詳細ログ収集部 1 0 4 は、自 CPU 1 0 と異常が発生している CPU 1 0 とを除いた、他の全ての CPU 1 0 を停止させる。

30

【 0 1 8 6 】

ステップ E 4 において、詳細ログ収集部 1 0 4 は、メモリ 1 1 のデバッグカーネル動作領域 1 1 2 からデバッグカーネルを読み出して実行する（起動させる）。

【 0 1 8 7 】

ステップ E 5 において、詳細ログ収集部 1 0 4 は、図示しないキーボードやマウス等の入力装置を介して入力される、ユーザからの指示を待つ。

【 0 1 8 8 】

ユーザから指示入力が行なわれると、ステップ E 6 において、詳細ログ収集部 1 0 4 は、入力された指示内容を確認する。

40

【 0 1 8 9 】

入力された指示が再起動を指示するものである場合には、詳細ログ収集部 1 0 4 は、ステップ E 7 において、本コンピュータシステム 1 の再起動を実行して、処理を終了する。

【 0 1 9 0 】

一方、入力された指示がメモリダンプの採取を指示するものである場合には、詳細ログ収集部 1 0 4 は、ステップ E 8 において、本コンピュータシステム 1 のメモリ 1 1 のメモリダンプを取得して、ステップ E 5 に戻る。

【 0 1 9 1 】

また、入力された指示が周辺レジスタのレジスタダンプの採取を指示するものである場

50

合には、詳細ログ収集部 104 は、ステップ E9 において、本コンピュータシステム 1 の周辺レジスタ 15 のレジスタダンプを取得して、ステップ E5 に戻る。

【0192】

図 13 ~ 図 15 は実施形態の一例としてのコンピュータシステム 1 における、移管元のログ出力処理制御部 103 と移管先のログ出力処理制御部 103 との処理を例示するシーケンス図である。図 14 は図 13 に後続する処理を示し、図 15 は図 14 に後続する処理を示す。

【0193】

なお、図 13 においては、便宜上、CPU # 1, # 2 だけを例示している。また、CPU # 1 においては、ログ出力対象情報 114 には CPU 番号 “ 1 ” が設定されており、代替 CPU 情報 115 には CPU 番号は設定されていない。一方、CPU # 2 においては、ログ出力対象情報 114 には CPU 番号 “ 2 ” が設定されており、代替 CPU 情報 115 には CPU 番号は設定されていない。

また、CPU # 2 は低負荷状態であるものとする。

【0194】

このような状態において、CPU # 1 (第 1 のプロセッサ：移管元) が低負荷状態から高負荷状態に遷移すると、CPU # 1 のログ出力処理制御部 103 は、CPU # 2 (第 2 のプロセッサ：移管先) に対してログ出力対象番号 “ 1 ” を通知する (図 13 中の符号 F1 参照)。

【0195】

これにより、CPU # 1 (第 1 のプロセッサ：移管元) においては、ログ出力対象情報 (第 1 のログ出力対象情報) 114 にはログ出力対象番号 (CPU 番号) は未登録 (なし) の状態となり、また、代替 CPU 情報 (移管先プロセッサ情報) 115 には、ログ出力対象番号の送信先である CPU # 2 の CPU 番号 “ 2 ” が設定される。

【0196】

一方、CPU # 2 (第 2 のプロセッサ：移管先) においては、ログ出力対象情報 (第 2 のログ出力対象情報) 114 には CPU 番号 “ 1 ” が追加されることで、“ 1, 2 ” が設定される。なお、代替 CPU 情報 115 には CPU 番号は未設定 (なし) のままである。

【0197】

その後、図 14 に示すように、CPU # 2 が低負荷状態から高負荷状態に遷移したものを示す。

【0198】

なお、図 14 においては、便宜上、CPU # 1 ~ # 3 だけを例示している。また、CPU # 3 においては、ログ出力対象情報 114 には CPU 番号 “ 3 ” が設定されており、代替 CPU 情報 115 には CPU 番号は設定されていない。また、CPU # 1 においては、ログ出力対象情報 114 には CPU 番号が設定されておらず、代替 CPU 情報 115 には CPU 番号 “ 2 ” が設定されている。

また、CPU # 3 は低負荷状態であるものとする。

【0199】

このような状態において、CPU # 2 が低負荷状態から高負荷状態に遷移すると、CPU # 2 (第 1 のプロセッサ：移管元) のログ出力処理制御部 103 は、CPU # 3 (第 2 のプロセッサ：移管先) に対してログ出力対象番号 “ 1, 2 ” を通知する (図 14 中の符号 F2 参照)。

【0200】

これにより、CPU # 2 (第 1 のプロセッサ：移管元) においては、ログ出力対象情報 (第 1 のログ出力対象情報) 114 にはログ出力対象番号 (CPU 番号) は未登録 (なし) の状態となり、また、代替 CPU 情報 (移管先プロセッサ情報) 115 には、ログ出力対象番号の送信先である CPU # 3 の CPU 番号 “ 3 ” が設定される。

【0201】

一方、CPU # 3 (第 2 のプロセッサ：移管先) においては、ログ出力対象情報 (第 2

10

20

30

40

50

のログ出力対象情報) 114にはCPU番号“1, 2”が追加されることで、“1, 2, 3”が設定される。なお、代替CPU情報115にはCPU番号は未設定(なし)のままである。

【0202】

また、CPU#2のログ出力対象情報114は、CPU#1に対して、代替CPU番号を送信する(図14中の符号F3参照)。

【0203】

これにより、CPU#1においては、ログ出力対象情報114にはログ出力対象番号(CPU番号)は未登録(なし)の状態であり、また、代替CPU情報115には、代替CPU番号の送信元であるCPU#2のCPU番号“2”が設定される。

10

【0204】

その後、図15に示すように、CPU#2が高負荷状態から低負荷状態に遷移したものを

する。

なお、図15においても、便宜上、CPU#1~#3だけを例示している。

【0205】

CPU#2が高負荷状態から低負荷状態に遷移すると、CPU#2(第1のプロセッサ:移管元)のログ出力処理制御部103は、CPU#3に対してログ出力対象番号の取り戻しを通知する(図15中の符号F4参照)。

【0206】

これにより、CPU#3(第2のプロセッサ:移管先)においては、ログ出力対象情報(第2のログ出力対象情報)114のログ出力対象番号(CPU番号)から、ログ出力対象番号の取り戻しの要求元のCPU10のCPU番号“2”が削除され、ログ出力対象情報114にはCPU番号“1, 3”が設定される。また、代替CPU情報115は引き続き、CPU番号は未設定(なし)のままである。

20

【0207】

一方、CPU#2においては、CPU番号“2”が取り戻されることで、ログ出力対象情報(第1のログ出力対象情報)114にはCPU番号“2”が設定される。なお、代替CPU情報115にはCPU番号は未設定(なし)のままである。

【0208】

(C)効果

30

このように、本発明の一実施形態としてのコンピュータシステム1によれば、SMPカーネルの動作環境化において、ICEやHWシミュレータ等を用いることなくログ情報を収集することができ、カーネルや仮想化ソフトの障害調査が可能となる。

【0209】

また、ログ情報を採取するための専用のプロセッサやハードウェアを備える必要がないので経済的であり、また、既存のコンピュータシステム1において容易に実施することができる。

【0210】

本コンピュータシステム1に異常が発生した場合に、その直前に動作していたCPU10やカーネルの情報を記憶装置12に退避させることができ、ICEやシミュレータ等がない環境においてもカーネルや仮想化ソフトの障害調査が可能となる。

40

【0211】

また、過去に動作していた情報を採取することにより、障害の直接の原因が過去に遡らないとわからない場合でも、障害調査が可能となる。

【0212】

自CPU10が低負荷状態から高負荷状態に遷移した場合に、ログ出力処理制御部103が、複数のCPU10のうち、CPU負荷が最も低いCPU10にログ出力部102としての機能を移管させる。

【0213】

これにより、比較的CPU負荷の高いログ出力処理を、低負荷状態の他のCPU10に

50

実行させることで、複数のＣＰＵ１０が実行中の他のプロセスに遅延等の影響を与えることがなく、本コンピュータシステム１を安定して動作させることができる。例えば、ログ収集部１０１としての機能に影響を与えることなくログ出力を実施することができる。

【０２１４】

また、ＳＭＰを構成する複数のＣＰＵ１０間において、負荷を分散させることができ、これによっても、本コンピュータシステム１を安定して動作させることができる。

【０２１５】

また、ログ出力を他ＣＰＵ１０に移管した状態で、自ＣＰＵ１０が低負荷状態に遷移した場合に、ログ出力処理制御部１０３が移管先のＣＰＵ１０からログ出力を取り戻す。これによっても、複数のＣＰＵ１０間において負荷を分散させることができる。

10

複数のＣＰＵ１０のうちいずれかのＣＰＵ１０において何らかの異常が発生した場合に、各ＣＰＵ１０で実行される詳細ログ収集部１０４のうち、最初にこの異常発生を検知した詳細ログ収集部１０４が、自ＣＰＵ１０と当該異常が発生しているＣＰＵ１０とを除いた、他のＣＰＵ１０を停止させる。これにより、リングバッファ構成を有するログバッファ１１１のログ情報が、これらのＣＰＵ１０の動作に伴って生成されるログ情報により上書き消去されてしまうことを阻止することができる。

【０２１６】

(Ｄ)その他

そして、開示の技術は上述した実施形態に限定されるものではなく、本実施形態の趣旨を逸脱しない範囲で種々変形して実施することができる。本実施形態の各構成及び各処理は、必要に応じて取捨選択することができ、あるいは適宜組み合わせてもよい。

20

【０２１７】

また、上述した開示により本実施形態を当業者によって実施・製造することが可能である。

【０２１８】

(Ｅ)付記

(付記１)

複数のプロセッサを備える情報処理装置であって、
前記複数のプロセッサのうちの第１のプロセッサが、

当該第１のプロセッサの動作履歴情報を収集してログ記憶部に格納するログ収集部と

30

、
前記ログ記憶部に格納された前記動作履歴情報を、不揮発性の記憶装置に移動させるログ出力部と、

当該第１のプロセッサが低負荷状態から高負荷状態に遷移した場合に、前記複数のプロセッサのうち最も低負荷の第２のプロセッサに、前記ログ出力部としての機能を移管させる移管処理部と

しての機能を実行することを特徴とする、情報処理装置。

【０２１９】

(付記２)

前記移管処理部は、前記第１プロセッサが高負荷状態から低負荷状態に遷移した場合に、前記ログ出力部としての機能を移管した前記第２のプロセッサに対して、前記ログ出力部としての機能の移管を解除する通知を送信することを特徴とする、付記１記載の情報処理装置。

40

【０２２０】

(付記３)

前記第１のプロセッサが、

前記第１のプロセッサによって実行される前記ログ出力部が前記記憶装置に移動させる前記動作履歴情報に対応するプロセッサを示す第１のログ出力対象情報と、

前記第１のプロセッサによって実行されるログ出力部としての機能の移管先の前記第２のプロセッサを示す移管先プロセッサ情報とを備え、

50

前記移管処理部が、

前記第2のプロセッサに前記ログ出力部としての機能を移管させる際に、前記第1のログ出力対象情報から前記第1のプロセッサの登録を削除するとともに、前記移管先プロセッサ情報に前記第2のプロセッサを追加することを特徴とする、付記2記載の情報処理装置。

【0221】

(付記4)

前記移管処理部が、

前記第2のプロセッサに前記ログ出力部としての機能の移管を解除する通知を送信する際に、前記第1のログ出力対象情報に前記第1のプロセッサを追加するとともに、前記移管先プロセッサ情報から前記第2のプロセッサの登録を削除することを特徴とする、付記3記載の情報処理装置。

10

【0222】

(付記5)

前記第2のプロセッサが、

前記第2のプロセッサによって実行されるログ出力部が前記記憶装置に移動させる前記動作履歴情報に対応するプロセッサを示す第2のログ出力対象情報を備え、

前記第1のプロセッサから前記ログ出力部としての機能を移管させる通知を受信すると、前記第2のログ出力対象情報に前記第1のプロセッサを追加することを特徴とする、付記2～4のいずれか1項に記載の情報処理装置。

20

【0223】

(付記6)

前記第2のプロセッサが、

前記第1のプロセッサから前記ログ出力部としての機能の移管を解除する通知を受信すると、前記第2のログ出力対象情報から前記第1のプロセッサの登録を削除することを特徴とする、付記5記載の情報処理装置。

【0224】

(付記7)

前記複数のプロセッサのうち第3のプロセッサに異常が検出された場合に、前記複数のプロセッサのうち前記第1プロセッサおよび第3のプロセッサ以外のプロセッサの動作を停止させる停止処理部

30

を備えることを特徴とする、付記1～6のいずれか1項に記載の情報処理装置。

【0225】

(付記8)

複数のプロセッサを備える情報処理装置において、

前記複数のプロセッサのうちの第1のプロセッサに、

当該第1のプロセッサの動作履歴情報を収集してログ記憶部に格納する処理と、

前記ログ記憶部に格納された前記動作履歴情報を不揮発性の記憶装置に移動させる処理と、

当該第1のプロセッサが低負荷状態から高負荷状態に遷移した場合に、前記複数のプロセッサのうち最も低負荷の第2のプロセッサに、前記動作履歴情報を不揮発性の記憶装置に移動させる処理を移管させる処理とを実行させる制御プログラム。

40

【0226】

(付記9)

前記第1プロセッサが高負荷状態から低負荷状態に遷移した場合に、前記動作履歴情報を不揮発性の記憶装置に移動させる機能を移管した前記第2のプロセッサに対して、前記移管を解除する通知を送信する処理を前記第1のプロセッサに実行させることを特徴とする、付記8記載の制御プログラム。

50

【 0 2 2 7 】

(付 記 1 0)

前記第 1 のプロセッサが、前記第 1 のプロセッサによって実行される前記動作履歴情報を不揮発性の記憶装置に移動させる処理が移動させる前記動作履歴情報に対応するプロセッサを示す第 1 のログ出力対象情報と、前記第 1 のプロセッサによって実行される前記動作履歴情報を不揮発性の記憶装置に移動させる処理の移管先の前記第 2 のプロセッサを示す移管先プロセッサ情報とを備え、

前記第 2 のプロセッサに前記動作履歴情報を不揮発性の記憶装置に移動させる処理を移管させる際に、前記第 1 のログ出力対象情報から前記第 1 のプロセッサの登録を削除するとともに、前記移管先プロセッサ情報に前記第 2 のプロセッサを追加する
処理を前記第 1 のプロセッサに実行させることを特徴とする、付記 9 記載の制御プログラム。

10

【 0 2 2 8 】

(付 記 1 1)

前記第 2 のプロセッサに前記移管を解除する通知を送信する際に、前記第 1 のログ出力対象情報に前記第 1 のプロセッサを追加するとともに、前記移管先プロセッサ情報から前記第 2 のプロセッサの登録を削除する

処理を前記第 1 のプロセッサに実行させることを特徴とする、付記 1 0 記載の制御プログラム。

【 0 2 2 9 】

(付 記 1 2)

前記第 2 のプロセッサが、前記第 2 のプロセッサによって実行される前記動作履歴情報を不揮発性の記憶装置に移動させる処理が前記記憶装置に移動させる前記動作履歴情報に対応するプロセッサを示す第 2 のログ出力対象情報を備え、

前記第 1 のプロセッサから前記動作履歴情報を不揮発性の記憶装置に移動させる処理を移管させる通知を受信すると、前記第 2 のログ出力対象情報に前記第 1 のプロセッサを追加する

処理を前記第 2 のプロセッサに実行させることを特徴とする、付記 9 ～ 1 1 のいずれか 1 項に記載の制御プログラム。

20

【 0 2 3 0 】

(付 記 1 3)

前記第 1 のプロセッサから前記移管を解除する通知を受信すると、前記第 2 のログ出力対象情報から前記第 1 のプロセッサの登録を削除する

処理を前記第 2 のプロセッサに実行させることを特徴とする、付記 1 2 記載の制御プログラム。

【 0 2 3 1 】

(付 記 1 4)

前記複数のプロセッサのうち第 3 のプロセッサに異常が検出された場合に、前記複数のプロセッサのうち前記第 1 プロセッサおよび第 3 のプロセッサ以外のプロセッサの動作を停止させる処理

を前記第 1 のプロセッサに実行させることを特徴とする、付記 8 ～ 1 3 のいずれか 1 項に記載の制御プログラム。

40

【 符号の説明 】

【 0 2 3 2 】

1 コンピュータシステム

1 0 - 1 ～ 1 0 - 4 , 1 0 C P U

1 0 1 ログ収集部

1 0 1 1 割込みコントローラ

1 0 1 2 割込みハンドラ

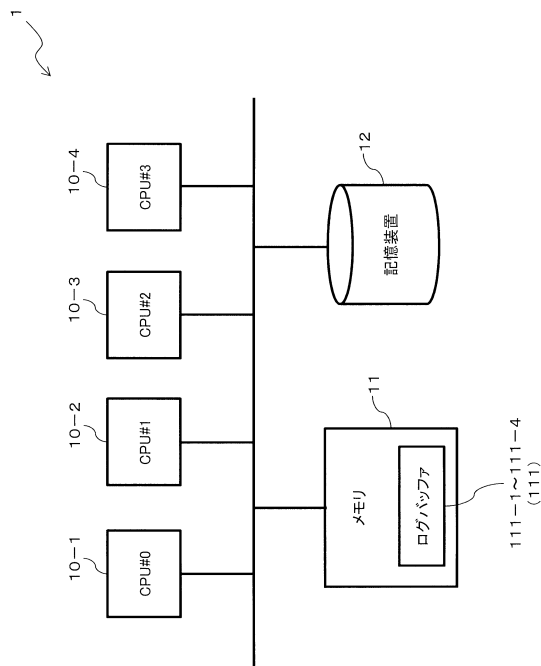
1 0 2 ログ出力部

50

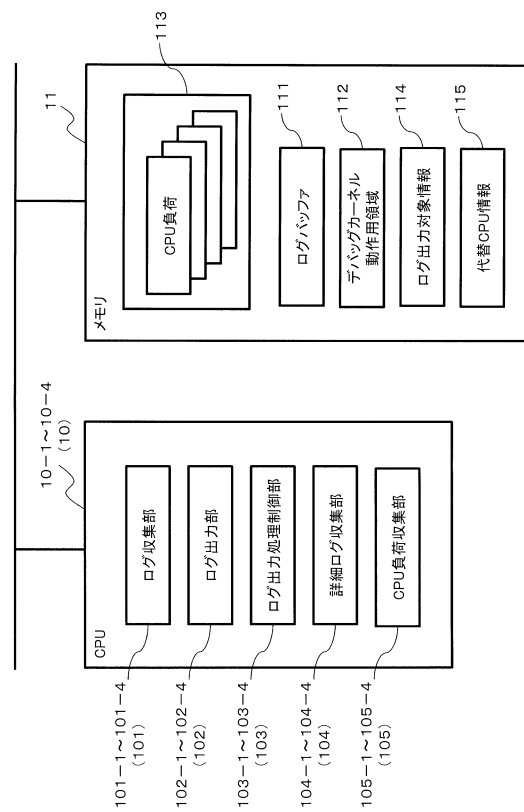
- 103 ログ出力処理制御部
- 104 詳細ログ収集部
- 105 CPU負荷収集部
- 11 メモリ
 - 111 ログバッファ
 - 112 デバッグカーネル動作領域
 - 113 CPU負荷情報格納領域
 - 114 ログ出力対象情報
 - 115 代替CPU情報
- 12 記憶装置
 - 121 ログ用ディスク領域

10

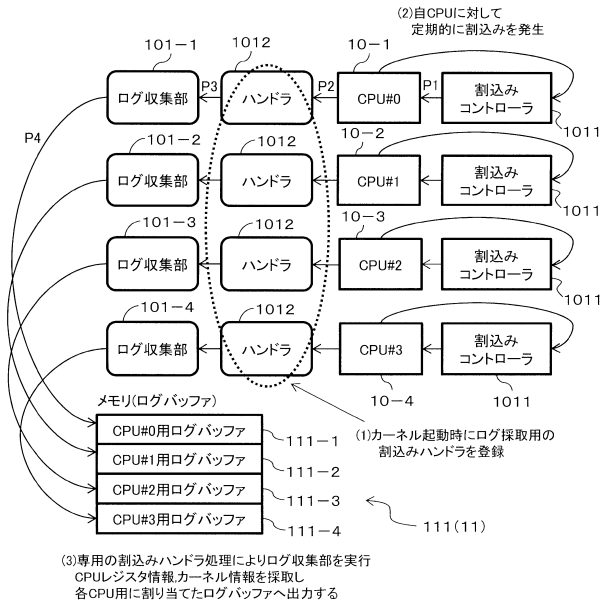
【図1】



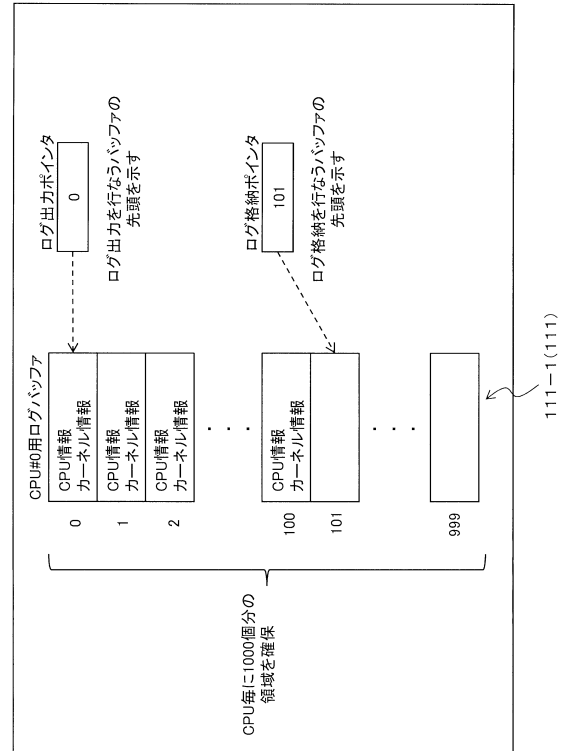
【図2】



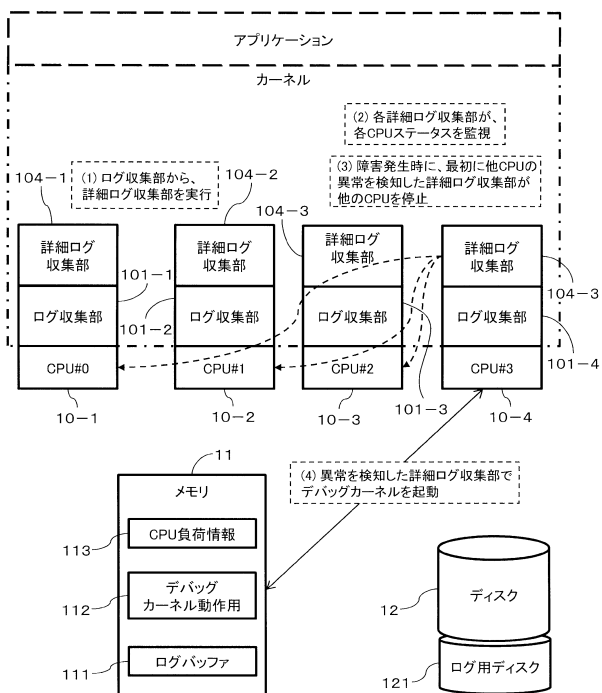
【図3】



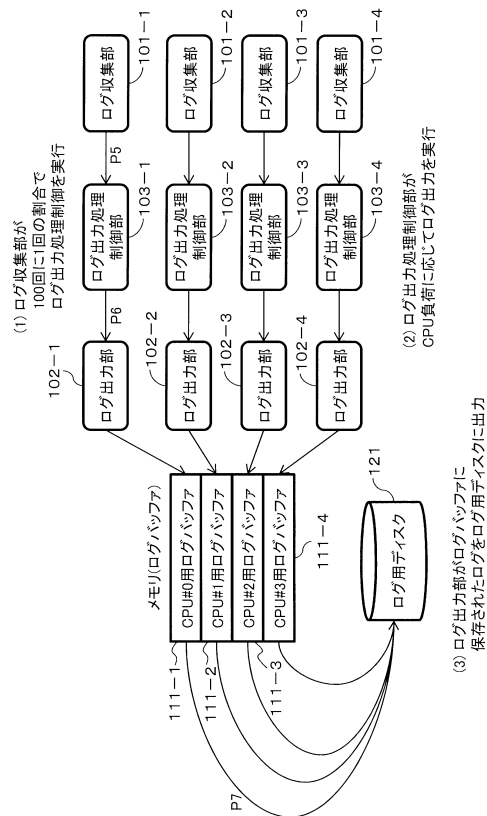
【図4】



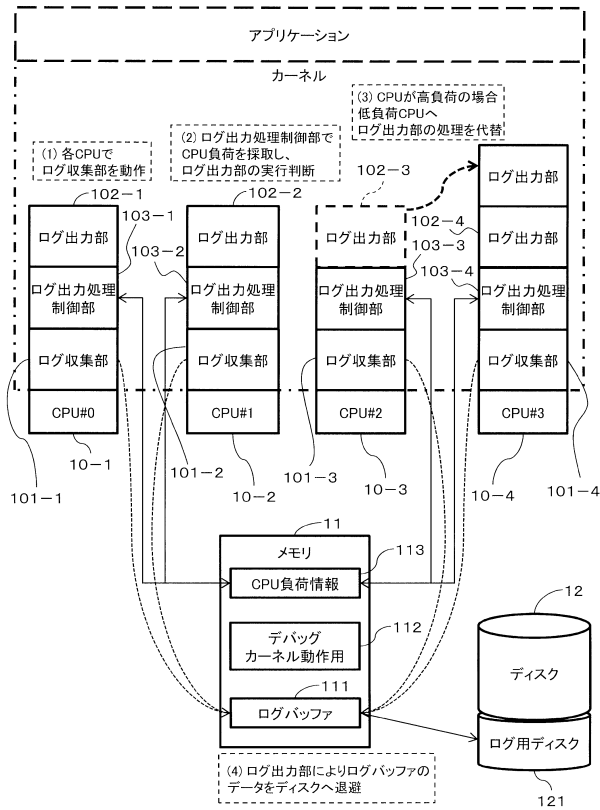
【図5】



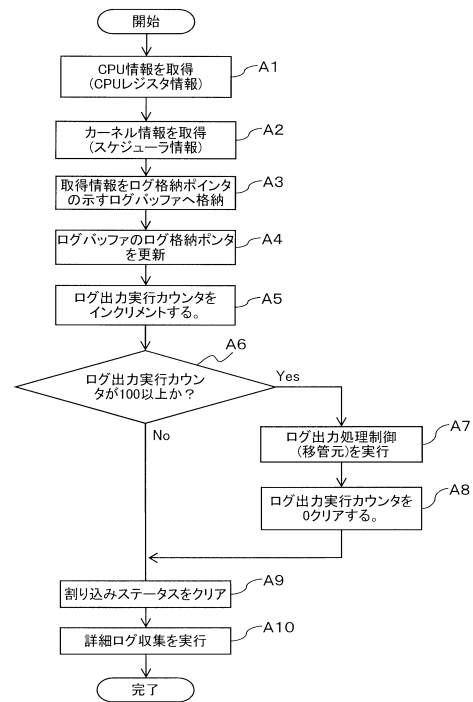
【図6】



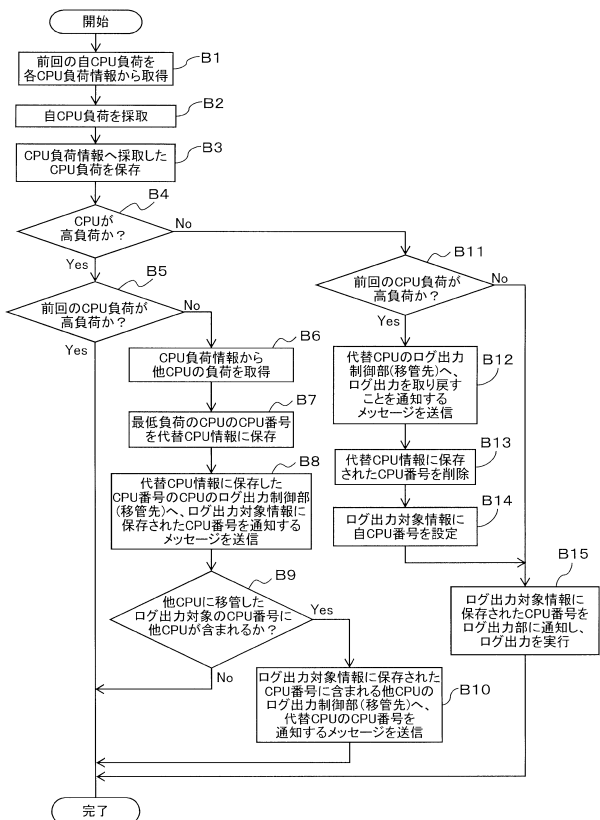
【図 7】



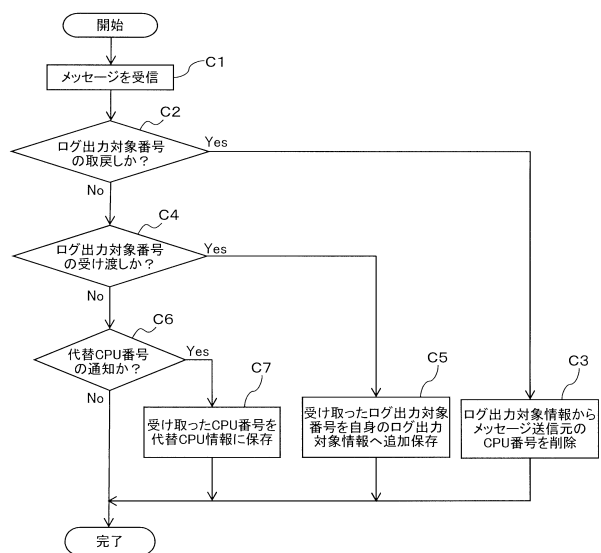
【図 8】



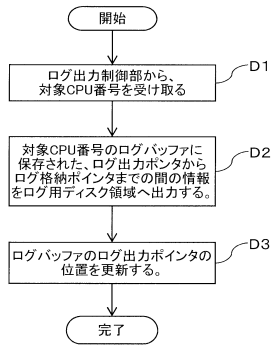
【図 9】



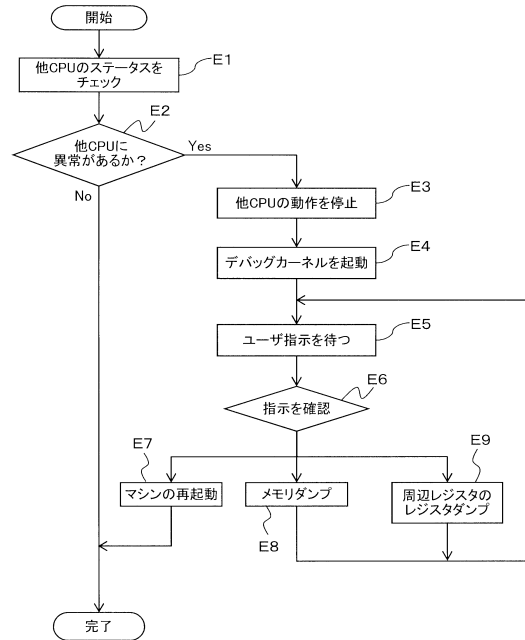
【図 10】



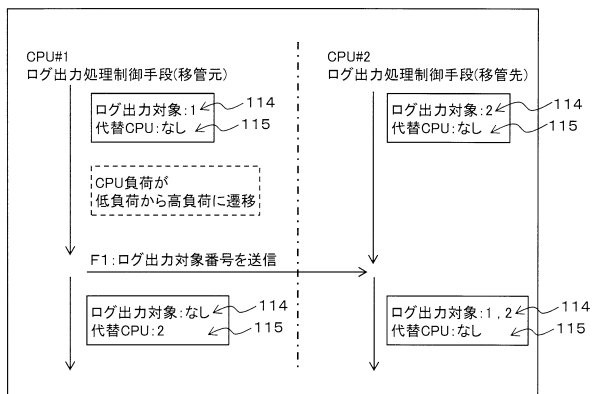
【 ㄨ 1 1 】



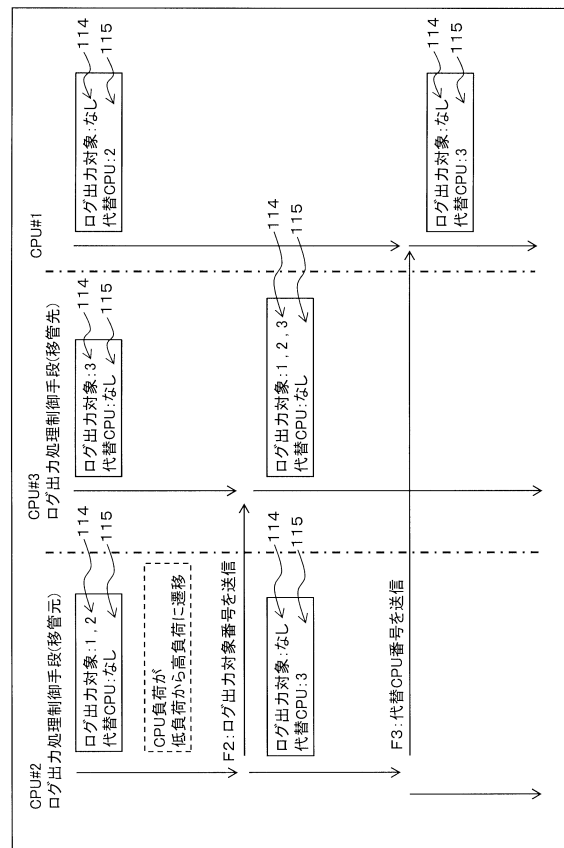
【 図 1 2 】



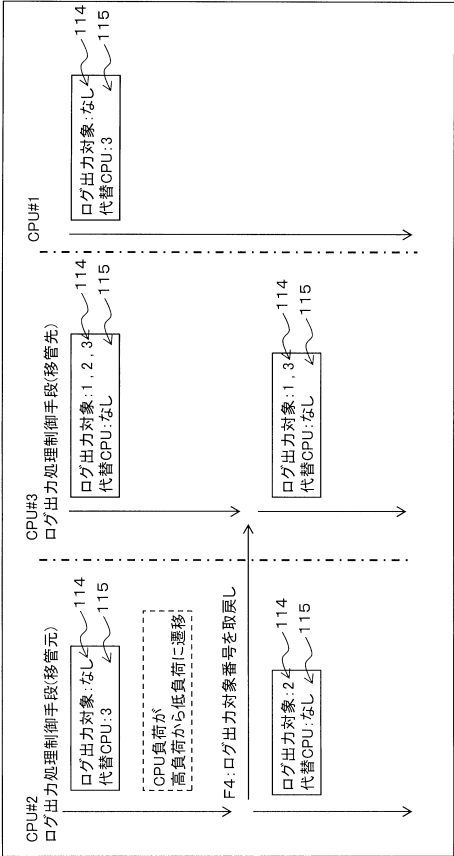
【 図 1 3 】



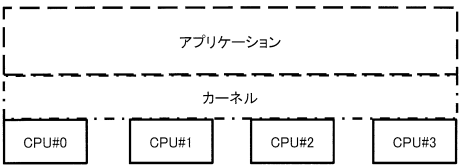
【 図 1 4 】



【図 15】



【図 16】



フロントページの続き

審査官 多賀 実

(56)参考文献 特開2014-010711(JP,A)
国際公開第2012/004854(WO,A1)
特開2014-206786(JP,A)
特開2006-085372(JP,A)
特開平2-027449(JP,A)

(58)調査した分野(Int.Cl., DB名)
G06F 11/07
G06F 11/28 - 11/36
G06F 9/50