

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5801331号
(P5801331)

(45) 発行日 平成27年10月28日 (2015. 10. 28)

(24) 登録日 平成27年9月4日 (2015. 9. 4)

(51) Int. Cl. F I
G 0 6 F 9/48 (2006.01)
 G O 6 F 9/46 4 5 2 H
 G O 6 F 9/46 4 5 2 G

請求項の数 6 (全 18 頁)

(21) 出願番号	特願2013-14585 (P2013-14585)	(73) 特許権者	392026693
(22) 出願日	平成25年1月29日 (2013. 1. 29)		株式会社 N T T ドコモ
(65) 公開番号	特開2014-29662 (P2014-29662A)		東京都千代田区永田町二丁目 1 1 番 1 号
(43) 公開日	平成26年2月13日 (2014. 2. 13)	(74) 代理人	110000752
審査請求日	平成26年7月31日 (2014. 7. 31)		特許業務法人朝日特許事務所
(31) 優先権主張番号	特願2012-145892 (P2012-145892)	(72) 発明者	小西 哲平
(32) 優先日	平成24年6月28日 (2012. 6. 28)		東京都千代田区永田町二丁目 1 1 番 1 号
(33) 優先権主張国	日本国 (JP)		株式会社 エヌ・ティ・ティ・ドコモ内
		(72) 発明者	稲村 浩
			東京都千代田区永田町二丁目 1 1 番 1 号
			株式会社 エヌ・ティ・ティ・ドコモ内
		(72) 発明者	神山 剛
			東京都千代田区永田町二丁目 1 1 番 1 号
			株式会社 エヌ・ティ・ティ・ドコモ内

最終頁に続く

(54) 【発明の名称】 タスク制御装置

(57) 【特許請求の範囲】

【請求項 1】

指定された時刻の変更を許さないように設定された第 1 のタスクについては、当該実行時刻に実行する一方、前記指定された時刻の変更を許すように設定された第 2 のタスクについては、前記第 1 のタスクの実行時刻とは異なる時刻に、当該第 2 のタスクどうしをまとめて実行するタスク制御部を備え、

前記タスク制御部は、

格納部と、

前記第 1 のタスクについては、当該タスクについて設定された実行時刻を前記格納部に格納し、前記第 2 のタスクについては、前記第 1 のタスクの実行時刻とは異なる時刻であって、当該第 2 のタスクどうしをまとめて実行するような実行時刻を設定して前記格納部に格納する格納処理部を備え、

前記格納部に格納された実行時刻が到来したときに前記第 1 のタスク及び前記第 2 のタスクを実行し、

前記格納処理部は、実行時期に周期性のあるタスク群と実行時期に周期性のないタスク群とを含む前記第 2 のタスク群のうち、通信処理を発生させず且つ実行時期に周期性のないタスクについては、前記第 1 のタスクと同時に実行するような実行時期を設定して前記格納部に格納する

ことを特徴とするタスク制御装置。

【請求項 2】

指定された時刻の変更を許さないように設定された第1のタスクについては、当該実行時刻に実行する一方、前記指定された時刻の変更を許すように設定された第2のタスクについては、前記第1のタスクの実行時刻とは異なる時刻に、当該第2のタスクどうしをまとめて実行するタスク制御部を備え、

前記タスク制御部は、
格納部と、

前記第1のタスクについては、当該タスクについて設定された実行時刻を前記格納部に格納し、前記第2のタスクについては、前記第1のタスクの実行時刻とは異なる時刻であって、当該第2のタスクどうしをまとめて実行するような実行時刻を設定して前記格納部に格納する格納処理部を備え、

10

前記格納部に格納された実行時刻が到来したときに前記第1のタスク及び前記第2のタスクを実行し、

前記格納処理部は、通信処理を発生させる前記第2のタスクにおいては、当該第2のタスクにおける通信開始時刻に基づいて、同時に通信が開始されるような実行時刻を設定して前記格納部に格納する

ことを特徴とするタスク制御装置。

【請求項3】

指定された時刻の変更を許さないように設定された第1のタスクについては、当該実行時刻に実行する一方、前記指定された時刻の変更を許すように設定された第2のタスクについては、前記第1のタスクの実行時刻とは異なる時刻に、当該第2のタスクどうしをまとめて実行するタスク制御部を備え、

20

前記タスク制御部は、休止状態から稼動状態に遷移する条件が、前記第2のタスクに対するイベント通知であった場合には、休止状態から稼動状態に遷移した時刻を経過した前記第2のタスクの設定時刻を、休止状態から稼動状態に遷移した時刻以降に実行時刻が設定されている第2のタスクのうち、最も早い実行時刻の第2のタスクのものに設定する

ことを特徴とするタスク制御装置。

【請求項4】

指定された時刻の変更を許さないように設定された第1のタスクについては、当該実行時刻に実行する一方、前記指定された時刻の変更を許すように設定された第2のタスクについては、前記第1のタスクの実行時刻とは異なる時刻に、当該第2のタスクどうしをまとめて実行するタスク制御部を備え、

30

前記タスク制御部は、休止状態から稼動状態に遷移する条件が、前記タスクの実行要求及び前記イベント通知のいずれでもない場合には、休止状態から稼動状態に遷移した時刻を経過した前記第2のタスクの設定時刻を、休止状態から稼動状態に遷移した時刻以降に実行時刻が設定されている第2のタスクのうち、最も早い実行時刻の第2のタスクのものに設定する

ことを特徴とするタスク制御装置。

【請求項5】

指定された時刻の変更を許さないように設定された第1のタスクについては、当該実行時刻に実行する一方、前記指定された時刻の変更を許すように設定された第2のタスクについては、前記第1のタスクの実行時刻とは異なる時刻に、当該第2のタスクどうしをまとめて実行するタスク制御部を備え、

40

前記タスク制御部は、
格納部と、

前記第1のタスク及び前記第2のタスクについて設定された実行時刻を前記格納部に格納する格納処理部と、

前記第1のタスクについては、当該タスクについて前記格納部に格納された実行時刻に実行する一方、前記第2のタスクについては、当該タスクについて前記格納部に格納された実行時刻を、前記第1のタスクの実行時刻とは異なる時刻であって、当該第2のタスクどうしをまとめて実行するような実行時刻に変更して実行する実行部と

50

を備え、

前記タスク制御部は、自タスク制御装置を備えた情報処理装置を含む複数の情報処理装置に一齐に配信される同報情報を受取ったことを検知する同報受信検知部を備え、

前記格納処理部は、前記同報受信検知部が前記同報情報を受取った場合に、当該検知した時点において実行時刻が経過している前記第2のタスクがあれば、当該第2のタスクについての実行時刻を、当該検知した時点以降の実行時刻に上書きして前記格納部に格納する

ことを特徴とするタスク制御装置。

【請求項6】

指定された時刻の変更を許さないように設定された第1のタスクについては、当該実行時刻に実行する一方、前記指定された時刻の変更を許すように設定された第2のタスクについては、前記第1のタスクの実行時刻とは異なる時刻に、当該第2のタスクどうしをまとめて実行するタスク制御部を備え、

10

前記タスク制御部は、

格納部と、

前記第1のタスク及び前記第2のタスクについて設定された実行時刻を前記格納部に格納する格納処理部と、

前記第1のタスクについては、当該タスクについて前記格納部に格納された実行時刻に実行する一方、前記第2のタスクについては、当該タスクについて前記格納部に格納された実行時刻を、前記第1のタスクの実行時刻とは異なる時刻であって、当該第2のタスクどうしをまとめて実行するような実行時刻に変更して実行する実行部と

20

を備え、

前記タスク制御部は、自タスク制御装置を備えた情報処理装置を含む複数の情報処理装置に一齐に配信される同報情報を受取ったことを検知する同報受信検知部を備え、

前記格納処理部は、前記同報受信検知部が前記同報情報を受取った場合に、当該検知した時点において実行時刻が経過している前記第2のタスクがあれば、当該検知した時点において実行時刻が経過している前記第2のタスクについての実行時刻を削除する

ことを特徴とするタスク制御装置。

【発明の詳細な説明】

【技術分野】

30

【0001】

本発明はタスクの制御に関する。

【背景技術】

【0002】

近年、複数のタスクを同時に実行できるマルチタスクOS (Operating System) を搭載する情報処理装置が増加している。マルチタスクOS を搭載するものの一例として、スマートフォンがある。スマートフォンは、従来の携帯装置とは違い、液晶画面のバックライト消灯時においても、アプリケーションプログラム (以下ではアプリという) の各タスクが、サーバ装置とのデータ同期や、GPS (Global Positioning System) や無線LAN (Local Area Network) に関連した処理を行うためにCPUが頻繁に稼動状態となるため電力を消費する。さらに、スマートフォンにおいては無線通信の発生回数も増加するため、基地局が処理すべき信号の信号量も増加し、基地局や移動通信網における負荷が大きくなる。また、複数のスマートフォンが同時に通信を行うこと (以下ではバーストラフィックという) があるが、これにより基地局または移動通信網における負荷が一時的に増大する現象も発生している。

40

【先行技術文献】

【特許文献】

【0003】

【特許文献1】特開平8-255088号公報

【発明の概要】

50

【発明が解決しようとする課題】

【0004】

複数のタスクを制御する技術として、特許文献1には、タスクを時間指定または時刻指定に分離してタスクの実行要求を制御する仕組みが開示されている。しかし、この仕組みは、タスクを2種類に分離してタイマ割り込みの機会を減らすというものに過ぎず、省電力化とネットワーク負荷の軽減という観点ではまだ十分とはいえない。

そこで、本発明は、タスクの制御によって処理負荷の軽減を行うことを目的とする。

【課題を解決するための手段】

【0005】

本発明は、指定された時刻の変更を許さないように設定された第1のタスクについては、当該実行時刻に実行する一方、前記指定された時刻の変更を許すように設定された第2のタスクについては、前記第1のタスクの実行時刻とは異なる時刻に、当該第2のタスクどうしをまとめて実行するタスク制御部を備えることを特徴とするタスク制御装置を提供する。

10

【0006】

前記タスク制御部は、前記タスクを識別するためのタスク識別情報と、当該タスクの実行時刻と、当該タスクが前記第1のタスクか前記第2のタスクであることを示すタイプ情報を含む実行要求情報を取得し、当該タイプ情報に基づいて、当該タスクが前記第1のタスクか前記第2のタスクであるかを判断するようにしてもよい。

【0007】

20

前記タスク制御部は、前記タスクを識別するためのタスク識別情報と、当該タスクの実行時刻とを含む実行要求情報を取得し、当該実行時刻に基づいて、当該タスクが前記第1のタスクか前記第2のタスクであるかを判断するようにしてもよい。

【0008】

前記タスク制御部は、格納部と、前記第1のタスクについては、当該タスクについて設定された実行時刻を前記格納部に格納し、前記第2のタスクについては、前記第1のタスクの実行時刻とは異なる時刻であって、当該第2のタスクどうしをまとめて実行するよう

な実行時刻を設定して前記格納部に格納する格納処理部を備え、前記格納部に格納された実行時刻が到来したときに前記第1のタスク及び前記第2のタスクを実行するようにしてもよい。

30

【0009】

前記格納処理部は、実行時期に周期性のあるタスク群と実行時期に周期性のないタスク群とを含む前記第2のタスク群のうち、実行時期に周期性のあるタスク群については、実行周期に公約数を含むタスクどうしをまとめて実行するような実行時刻を設定して前記格納部に格納するようにしてもよい。

【0010】

前記格納処理部は、実行時期に周期性のあるタスク群と実行時期に周期性のないタスク群とを含む前記第2のタスク群のうち、通信処理を発生させず且つ実行時期に周期性のないタスクについては、前記第1のタスクと同時に実行するような実行時期を設定して前記格納部に格納するようにしてもよい。

40

【0011】

前記格納処理部は、通信処理を発生させる前記第2のタスクにおいては、当該第2のタスクにおける通信開始時刻に基づいて、同時に通信が開始されるような実行時刻を設定して前記格納部に格納するようにしてもよい。

【0012】

前記タスク制御部は、格納部と、前記第1のタスク及び前記第2のタスクについて設定された実行時刻を前記格納部に格納する格納処理部と、前記第1のタスクについては、当該タスクについて前記格納部に格納された実行時刻に実行する一方、前記第2のタスクについては、当該タスクについて前記格納部に格納された実行時刻を、前記第1のタスクの

50

実行時刻とは異なる時刻であって、当該第2のタスクどうしをまとめて実行するような実行時刻に変更して実行する実行部とを備えるようにしてもよい。

【0013】

前記タスク制御部は、休止状態から稼動状態に遷移する条件が、前記第2のタスクに対するイベント通知であった場合には、休止状態から稼動状態に遷移した時刻を経過した前記第2タスクの設定時刻を、休止状態から稼動状態に遷移した時刻以降に実行時刻が設定されている第2タスクのうち、最も早い実行時刻の第2タスクのものに設定するようにしてもよい。

【0014】

前記タスク制御部は、休止状態から稼動状態に遷移する条件が、前記タスクの実行要求及び前記イベント通知のいずれでもない場合には、休止状態から稼動状態に遷移した時刻を経過した前記第2タスクの設定時刻を、休止状態から稼動状態に遷移した時刻以降に実行時刻が設定されている第2タスクのうち、最も早い実行時刻の第2タスクのものに設定するようにしてもよい。

10

【0015】

自装置の状態を管理する状態管理部を備え、前記タスク制御部は、前記状態管理部によって管理されている状態に基づいて前記タスクの実行時刻を変更するか否かを判断してから、当該タスクを実行するようにしてもよい。

【0016】

自装置に入力される情報に基づいて前記タスク制御部の挙動を変更する制御ポリシー変更部を備え、前記タスク制御部は、前記制御ポリシー変更部によって変更された制御ポリシーに従ってタスクの実行を制御するようにしてもよい。

20

【0017】

前記タスク制御部は、自タスク制御装置を備えた情報処理装置を含む複数の情報処理装置に一斉に配信される同報情報を受取ったことを検知する同報受信検知部を備え、前記格納処理部は、前記同報受信検知部が前記同報情報を受取った場合に、当該検知した時点において実行時刻が経過している前記第2のタスクがあれば、当該第2のタスクについての実行時刻を、当該検知した時点以降の実行時刻に上書きして前記格納部に格納するようにしてもよい。

【0018】

30

前記タスク制御部は、自タスク制御装置を備えた情報処理装置を含む複数の情報処理装置に一斉に配信される同報情報を受取ったことを検知する同報受信検知部を備え、前記格納処理部は、前記同報受信検知部が前記同報情報を受取った場合に、当該検知した時点において実行時刻が経過している前記第2のタスクがあれば、当該検知した時点において実行時刻が経過している前記第2のタスクについての実行時刻を削除するようにしてもよい。

【発明の効果】

【0019】

本発明によれば、タスクの制御によって処理負荷の軽減を行うことが可能となる。

【図面の簡単な説明】

40

【0020】

【図1】本発明の実施形態に係る情報処理装置の構成を示すブロック図である。

【図2】情報処理装置の機能構成を示すブロック図である。

【図3】情報処理装置の基本的な動作のタイミングチャートである。

【図4】実行要求情報のフォーマットの一例を示す図である。

【図5】情報処理装置の動作を示すフローチャートである。

【図6】情報処理装置の動作を示すフローチャートである。

【図7】動作の一例を説明する図である。

【図8】動作の一例を説明する図である。

【図9】変形例に係る情報処理装置の機能構成を示すブロック図である。

50

【図 1 0】変形例に係る実行要求情報のフォーマットの一例を示す図である。

【図 1 1】変形例に係る情報処理装置の動作を示すフローチャートである。

【図 1 2】変形例に係る情報処理装置の機能構成を示すブロック図である。

【図 1 3】変形例に係る情報処理装置の機能構成を示すブロック図である。

【図 1 4】変形例に係る情報処理装置の動作を示すフローチャートである。

【図 1 5】変形例に係る動作を一例を説明する図である。

【図 1 6】変形例に係る情報処理装置の動作を示すフローチャートである。

【図 1 7】変形例に係る情報処理装置の機能構成を示すブロック図である。

【図 1 8】変形例に係る情報処理装置の機能構成を示すブロック図である。

【図 1 9】変形例に係る情報処理装置の動作を示すフローチャートである。

10

【発明を実施するための形態】

【0021】

以下、図面を参照して、本発明に係る情報処理装置の好適な実施形態について詳細に説明する。

<構成>

図 1 は、実施形態に係る情報処理装置 100 の全体構成を示す一例である。情報処理装置 100 は、例えば通話及びデータ通信が可能な携帯電話機などのコンピュータであり、移動通信網を介して他の移動通信装置やサーバ装置との間で通話やデータ通信を実現する。情報処理装置 100 は、本発明に係るタスク制御装置に相当する。情報処理装置 100 は、アプリケーション 200、フレームワーク 300、ライブラリ 400、カーネル 500 及びハードウェア 600 を有する。アプリケーション 200 は、例えば、タスク 211 とタスク 212 によって構成されるアプリケーション 210 と、タスク 221 とタスク 222 によって構成されるアプリケーション 220 を含む。フレームワーク 300 は、例えば、タスク 211、212 の実行要求に従いこれらを実行するタスク制御部 310 と、CPU やメインメモリなどのリソースを管理するリソース管理部 320 と、情報処理装置 100 の位置を管理する位置情報管理部 330 をと含む。ライブラリ 400 は、例えば、データベースライブラリ 410 と、電力制御ライブラリ 420 と、Java（登録商標）VM（Virtual Machine）430 とを含む。カーネル 500 は、情報処理装置 100 の電源の管理を行う電源管理部 510 と、表示部 640 の管理を行う画面管理部 520 と、情報処理装置 100 におけるプロセスの管理を行うプロセス管理部 530 とを含む。ハードウェア 600 は、例えば、ベースバンドチップ 610、CPU（Central Processing Unit）620、メモリ（主記憶装置及び補助記憶装置）630、表示部 640、操作部 650、無線 LAN 通信部 660 を有する。

20

30

【0022】

タスク制御部 310 は、タスク制御部 310 が提供する関数を利用して登録されたタスクを、その実行要求情報に応じて実行する。タスクの実行要求情報は、そのタスクを実行する実行時刻と、その実行時刻がプログラマ乃至ユーザ（以下単にプログラマと言う）によって設定された実行時刻か又は情報処理装置 100 のシステムによって設定された実行時刻かを表す設定時刻タイプとからなる。

【0023】

プログラマによって設定された時刻を実行時刻とするタスク（本発明に係る第 1 のタスク、以降、P-Task という）の例として、目覚まし時計に関するタスクがある。目覚まし時計に関するタスクは、プログラマが用意した時刻群の中からユーザが指定した実行時刻（例えば毎朝 7 時）に動作するタスクである。つまり、P-Task は、設定された実行時刻とは異なる時刻に実行されしまうと、サービスとしての価値がないタスクであり、指定された絶対的な実行時刻の変更を許さないように設定されたタスクである。

40

【0024】

一方、プログラマが実行時刻を指定したとしても、システムによって最適化された時刻を実行時刻とするタスク（本発明に係る第 2 のタスク、以降 S-Task という）には、実行時刻に周期性のあるタスクと周期性のないタスクの 2 種類がある。周期性のあるタスクの例

50

として、情報処理装置100が移動通信網を介して通信を行うサーバ装置とのセッションを維持するためのKeep Aliveと呼ばれるタスクがある。このタスクが実行される周期さえ守れば、サーバ装置とのセッションは維持されるため、例えば毎朝7時というような絶対的な時刻にタスクが実行される必要はない。従って、システムは、このタスクの初回の実行時刻を自由に決定することができ、また、その後は、前回のタスク実行との相対的な実行時刻に実行すればよい。また、周期性のないタスクは、実行時刻の制約がなく、いつ実行されてもよいため、このタスクについてもシステムが実行時刻を自由に決定できる。従って、S-Taskは、絶対的な時刻以外の時刻に動作するタスクであり、プログラマによって指定された時刻の変更を許すように設定されたタスクである。

【0025】

<基本動作>

図2は、情報処理装置の機能構成の一例を示す図であり、図3は情報処理装置の基本的な動作のタイミングチャートである。タスク制御部310は、格納処理部311、格納部312及び実行部313を備える。格納部312はメモリ630によって実現され、格納処理部311及び実行部313はCPU620によって実現される。図3に示すように、各アプリケーション（アプリA、アプリB）の実行中において、各タスク（タスクA、B）は、タスク制御部310の関数を呼び出し、引数である実行要求情報をその関数に引き渡すことで、格納処理部311は実行要求情報を取得する（ステップS1）。格納処理部311は、その実行要求情報を順次格納部312に格納する（ステップS2）。ここま

【0026】

ここで、実行要求情報のフォーマットの一例を図4に示す。図4（A）に示すように、実行要求情報は、タスク名、設定時刻タイプ（P-taskかS-taskか）及び実行時刻を含む。なお、タスク名は、タスクを識別する識別情報であればよい。また、図4（B）に示すように、実行要求情報は、タスク名、設定時刻タイプ（P-taskかS-taskか）及び実行時刻のほか、通信の有無情報、通信開始時刻、通信終了時刻及びリソース消費量を含んでいてもよい。

【0027】

一方、タスク実行スレッドは、実行部313の休止状態（いわゆるスリープ）が解除されて稼動状態に移移することを契機として開始される。実行部313が格納部312の実行要求情報へアクセスし（ステップS3）、実行要求情報を確認する（ステップS4）。実行部313が休止状態から稼動状態に移移する条件は、（1）タスクの実行要求に基づくもの、（2）電池残量の変化やネットワーク状況などのイベント通知を行うことに基づくもの、（3）上記（1）（2）以外のもの（例えばカーネルのデバイス管理のためのものや、操作部650の押下によるもの）がある。実行部313は、実行要求情報を確認した結果、上記（1）に該当する場合には、稼動の契機となったタスク（図3ではタスクAとする）を実行する実行命令をアプリケーションAに対して発行する（ステップS5）。これによりタスクAが実行される。

【0028】

<格納処理部の動作>

格納処理部311が実行要求情報を格納部312に格納するときの動作を、図5を用いて説明する。まず、アプリケーションAの新たなタスクAが格納処理部311の関数を呼び出し、引数としての実行要求情報が引き渡された際に、格納処理部311はタスクのタイプがP-TaskかS-Taskかを判断する（ステップS100）。P-Taskであった場合（ステップS100；P-Task）、格納処理部311は、そのタスクAが保持している実行要求情報を、そのまま格納部312へ格納する（ステップS200）。S-Taskであった場合（ステップS100；S-Task）は、格納処理部311は、S-Taskの格納処理を行う（ステップS300）。

【0029】

<S-Taskの格納処理>

次に、S-Taskの格納処理を図6～8を用いて説明する。まず、格納処理部311は、タスクAが周期性を持つタスクかを判断する(ステップS301)。周期的なタスクであった場合には(ステップS301; Yes)、格納処理部311は、既に格納部312に格納されている実行要求情報の周期を取得し(ステップS302)、実行要求情報の周期とタスクAの周期との間に、周期の最小単位(例えば1(分))よりも大きい公約数があるかどうかを判断する(ステップS303)。そのような公約数がある場合(ステップS303; Yes)、格納処理部311は、タスクAの周期をメモリ630に記録する(ステップS304)。

【0030】

公約数がない場合(ステップS303; No)、格納処理部311は、格納部312にまだ比較していないタスクがあるか否かを判断する(ステップS305)。格納部312にまだ比較していないタスクがある場合(ステップS305; Yes)、格納処理部311は、もう一度ステップS302へ戻って上記と同じ処理を行う。全て比較し終えた場合には(ステップS305; No)、格納処理部311は、ステップS304で記録した周期のうちで最大のものを特定する(ステップS306)。そして、格納処理部311は、最大の周期をもつタスクの次回実行時刻をタスクAの初回実行時刻として、タスクAの実行要求情報を格納部312に格納する(ステップS307)。

【0031】

ステップS301～307の一例を図7を用いて説明する。周期が10であるタスクBの実行要求情報と、周期が30であるタスクCの実行要求情報が既に格納部312に格納されている場合に、時刻Tにおいて、周期が15のタスクAの実行要求情報が新たに生成されたとする。この場合、タスクAとタスクBとの最大公約数は5であり、タスクAとタスクCとの最大公約数は15であるから、タスクAの初回実行時刻は、タスクCの次回実行時刻である30となる。

【0032】

図6の説明に戻り、タスクAが周期性を持たないタスクだった場合(ステップS301; No)、格納処理部311は、既に格納部312に格納されている実行要求情報に係るタスクの周期を取得し(ステップS308)、そのタスクが周期性のあるタスクであるか否かを判断する(ステップS309)。周期性のあるタスクである場合(ステップS309; Yes)、格納処理部311は、そのタスクの現在時刻以降で直近の実行時刻をメモリ630に記録する(ステップS310)。格納部312にまだ取得していないタスクがある場合(ステップS311; Yes)、格納処理部311は、もう一度ステップS308へ戻って上記と同じ処理を行う。全て取得し終えた場合には(ステップS311; No)、格納処理部311は、ステップS310で記録した実行時刻のうち、現在時刻に最も近いものをタスクAの初回実行時刻として、タスクAの実行要求情報を格納部312に格納する。

【0033】

ステップS301～307の一例を図8を用いて説明する。タスクB、C、Dの実行要求情報が既に格納部312に格納されている場合に、時刻Tにおいて、タスクAの実行要求情報が新たに生成されたとする。タスクB、C、Dのうち、タスクC、Dは実行時間に周期性があるタスクであり、タスクBは実行時間に周期性がないタスクとする。この場合、タスクBについては周期性がないため、格納処理部311は、ステップS309からステップS311の処理へ遷移する。ここでは全タスクの周期を取得していないため(ステップS311; No)、格納処理部311は、ステップS308の処理に戻り、次のタス

クCの周期性の有無を判断する。タスクCは周期性があるため、格納処理部311は、ステップS310でタスクCの直近の次回実行時刻である20をメモリ630に格納する。同様にタスクDについても直近の次回実行時刻である50がメモリに格納される。格納処理部311は、ステップS312でメモリに格納されている20と50とを比較し、小さい方の20をタスクAの初回実行時刻として格納部312に格納する。

10

20

30

40

50

【0034】

以上のようにして、格納部312に実行要求情報が格納されると、その実行要求情報に含まれる実行時刻が到来したタイミングで実行部313によりタスクが実行される。

【0035】

<変形例>

上述した実施形態は次のような変形が可能である。以下の変形例を互いに組み合わせて実施してもよい。

<変形例1>

実施形態では、情報処理装置100が休止状態から稼動状態に遷移したときの条件がタスクの実行要求によるものである場合には、稼動の契機となったタスクを実行していた。これを、実行部313が、情報処理装置100の状態に関する状態情報（例えば表示部のバックライトの点灯又は消灯、電池残量、リソースの消費量、ネットワーク状態、情報処理装置と電源ケーブルとの切断又は接続などの関する情報）をフレームワークの提供するA P I（Application Program Interface）から取得し、この状態情報を用いてタスク実行を制御するようにしてもよい。この場合、図9に示すように、タスク制御部310は、状態情報を管理する状態管理部314を備える。この状態管理部314はC P U 620及びメモリ630によって実現される。図10は格納部312に格納される実行要求情報のフォーマットの一例を示す図である。情報処理装置100の状態によっては、実行時刻を変更せずに、元々設定されていた実行時刻にタスクを実行することもあるので、タスクの実行時刻の変更前後のものが格納部312に格納される。タスク制御部310は、変更前後の実行時刻のうち、タスクの同時実行をしたほうがよいか又はしないほうがよいかという判断に基づいて、いずれの実行時刻を選択してタスクを実行する。

【0036】

状態情報として表示部640の点灯及び消灯を用いる場合、タスク制御部310は、表示部640が点灯時にはタスクの同時実行を行わず、消灯時はタスクの同時実行を行うよう、処理を切り替える。表示部640の点灯時は、例えばウェブブラウジングなどユーザが表示部640を閲覧している可能性が高く、操作に対する処理実行の応答性への要求が高い。このようなときにタスクを同時実行することにより、タスク実行の応答性が低下してしまうと、ユーザの利便性が低下する可能性が高くなる。一方、表示部640の消灯時は、ユーザが表示部640を閲覧していない可能性が高いため、操作に対する処理実行の応答性への要求が低い。そこで、表示部640の消灯時は、タスクの実行時刻を変更し、省エネルギー化とネットワーク負荷の軽減を行うようタスクの同時実行を行うことが望ましい。

【0037】

図11に示すように、状態管理部314は、表示部640が点灯か消灯かを実行部313に通知し（ステップS800）、実行部313は表示部640が点灯状態であるか消灯状態であるかを判断する（ステップS810）。点灯状態であった場合（ステップS810；Yes）、実行部313は、格納部312内の変更前の実行時刻を参照し（ステップS820）、その実行時刻に実行処理を行う（ステップS830）。一方、表示部640が消灯状態だった場合（ステップS810；No）、実行部313は、格納部312内の変更後の実行時刻を参照し（ステップS840）、その時刻に実行処理を行う（ステップS830）。

【0038】

状態情報として、タスクを実行したときの情報処置装置100におけるリソースの消費量を用いる場合、タスク制御部310は、同時実行の対象となる各タスクによるリソース

消費量を足し合わせた合計値が閾値を超えた場合は、その合計値が閾値以下に収まるよう、リソース消費量の多いタスクから同時実行の処理対象から外していく。例えば、データ量がメインメモリの上限に達した場合、タスクがフリーズまたはクラッシュし、正しく動作しなくなる可能性があるため、このような場合に同時実行の対象となるタスクの数を減

10

20

30

40

50

らすほうがよい。各タスクによるリソース消費量を示す情報は各タスクの実行要求情報に予め含まれていてタスク制御部 310 がそれを参照してもよいし、タスク制御部 310 がタスク実行の際にリソースの状況をモニタリングしてリソース消費量を学習してもよい。

【0039】

状態情報として、移動通信網の基地局との通信又は無線 LAN における電波受信強度や、通信スループットなどのネットワークに関する情報を用いてもよい。この場合、タスク制御部 310 は、通信可能な帯域が余っていると判断した場合は、同時実行するタスクの数を増やしてもよい。

【0040】

状態情報として電池残量を用いてもよい。この場合、タスク制御部 310 は、電池残量が閾値以下の場合はタスクを同時実行して省電力化を図る。この閾値はユーザが任意の値を設定してもよいし、情報処理装置 100 のシステムが自動で設定してもよい。

【0041】

状態情報として、情報処理装置 100 が商用電源に繋がる電源ケーブルと接続されているか否かという情報を用いてもよい。この場合、タスク制御部 310 は、電源ケーブル切断時にはタスクを同時実行して省電力化を図る。

【0042】

<変形例 2>

変形例 1 では、情報処理装置 100 の状態に応じて、タスクの実行時刻を変更して同時実行するかどうかを切り替えていた。これを、情報処理装置 100 内の情報からは得られない外部からの入力情報に基づいて制御ポリシーを変更するようにしてもよい。このようにすれば、ユーザまたは開発者の意図や、ネットワーク状況或いは他の情報処理装置の状況などを反映した制御が可能になる。この場合、図 12 に示すように、タスク制御部 310 は、外部からの入力情報に応じて制御ポリシーを変更する制御ポリシー変換部 330 を備える。この制御ポリシー変換部 330 は CPU 620 及びメモリ 630 によって実現される。この変形例 2 においても、変形例 1 と同様に、制御ポリシーによっては、実行時刻を変更せずに、元々の実行時刻でタスクを実行することがあるので、タスクの実行時刻の変更前後のものが格納部 312 に格納される。

【0043】

例えば図 13 に示すように、制御ポリシー変換部 330 は、サーバ装置 1000 が記憶する各アプリの実行要求情報の変換ルール 1200 に従って、制御ポリシーを変換してもよい。実行部 313 によるタスクの実行履歴（タスク名、実行時刻、通信の有無など）を実行履歴格納部 314 に格納し、この実行履歴を実行履歴監視部 315 が監視する。実行履歴監視部 315 は、各タスクまたはアプリが、特定の時刻で実行されていないかとか、更新頻度が高くないかなどの傾向の有無を判断する。そういった傾向があった場合、実行履歴監視部 315 は、サーバ装置 1000 のアプリリスト 1100 に、タスクまたはアプリ名と実行要求情報を登録する。サーバ装置 1000 は、登録された情報に基づき、変換ルール 1200 を変更する。例えば、アプリが特定の時刻に集中して通信を行う場合、その時刻での通信を避けるような制御ポリシーに変換する。変換ルール 1200 が更新されると、サーバ装置 1000 は全ての情報処理装置 100 の制御ポリシーを変更する。なお、実行履歴格納部 314 はメモリ 630 によって実現され、実行履歴監視部 315 は CPU 620 によって実現される。

【0044】

上記の例では、実行履歴監視部 315 は、サーバ装置 1000 のアプリリスト 1100 に、タスクまたはアプリ名と実行要求情報を登録していたが、ユーザ自身が明示的にサーバ装置 1000 のアプリリスト 1100 へ登録してもよい。また、この変換ルール 1200

とアプリリスト 1100 は各情報処理装置 100 が保持しておき、各情報処理装置 100 がこれらに基づきタスクを制御してもよい。

【0045】

移動通信網の基地局または交換機と制御ポリシー変換部330が通信を行い、特定の時刻に通信が集中して発生しないようタスクの実行時刻を変更してもよい。例えば、基地局または交換機が、毎日午前7:00にバーストラフィックの発生を観測した場合、いくつかの情報処理装置100に対して午前7:00に通信を伴うタスクを実行しないよう、午前7:00に通信を行っているタスクの実行時刻を変更する。

【0046】

また、開発者が開発段階のアプリをテストする際は、情報処理装置100は、タスクの実行時刻を変更せず、元々設定されている実行時刻通りに実行してもよい。この場合、アプリのテストかどうかを判断するために、アプリに開発中であることを表すフラグをつけ、このフラグが付加されているアプリはタスクの実行時刻を変更しない機能をタスク制御部310に付加する。

【0047】

<変形例3>

実施形態では、格納処理部311がタスクの実行時刻を変更したうえで格納部312に格納していた。これを、格納処理部311はタスクの実行時刻を変更せずに格納部312に格納し、実行部313がタスクを実行する際にその実行時刻を変更するようにしてもよい。この場合、図14において、まず実行部313は、休止状態から稼動状態へ遷移すると、格納部312から実行要求情報を取得し、現在時刻と同時刻に設定されているタスクがあるか否かを確認する(ステップS400)。現在時刻と同時刻に設定されているタスクがある場合、つまり休止状態から稼動状態へ遷移した条件がタスクの実行要求によるものであった場合(ステップS400; Yes)、実行部313は、タスクの実行命令を発行して(ステップS410)、タスクを実行する(ステップS420)。一方、現在時刻と同時刻に設定されているタスクがない場合、つまり休止状態から稼動状態へ遷移した条件がタスクの実行要求によるものではなかった場合(ステップS400; No)、実行部313は、格納部312に格納されているS-Taskの実行要求情報を取得する(ステップS430)。そして、実行時刻を経過したS-Taskがあった場合には(ステップS440; Yes)、実行部313は、S-Taskの実行時刻を現在時刻以降で直近のタスク(実行時刻に周期性のあるタスク)の実行時刻に書き換えて再び格納部312に格納する(ステップS450)。このようにすれば、情報処理装置100の稼動状態への遷移の契機が各タスクに対するイベント通知によるものであった場合や、タスクの実行要求でもイベント通知でもない場合において、タスクの分散実行を回避でき、タスクを複数同時実行する機会が増える。

【0048】

<変形例4>

実施形態では、複数のタスクの実行時刻を揃えて同時実行した。しかし、タスクによっては、タスク実行から数秒~数十秒経過して通信処理が発生するものもある。例えば図15に示すように、タスクAとタスクBの実行時刻は同じでも、通信開始時刻がずれて制御信号発生回数は減らないというも考えられる。そこで、タスク制御部310は、複数のタスクを同時実行する際は、各タスクの通信開始時刻が揃うように、各タスクの実行時刻を変更する。この場合、タスクの通信開始時刻は実行要求情報に含まれてしてタスク制御部310がそれを参照してもよいし、タスク制御部310がタスクの通信開始時刻を記した実行履歴情報を元に算出してもよい。例えばタスクAの実行時刻を T_a 、通信開始時間を T_{ar} とすると、タスクAの通信開始時刻は $T_a + T_{ar}$ となる。タスクBについても同様に、実行時刻を T_b 、通信開始時間を T_{br} とすると、タスクBの通信開始時刻は $T_b + T_{br}$ となる。タスクAが先に格納部312へ格納されている場合、格納処理部311はタスクBの実行時刻を、 $T_a + T_{ar} - T_{br}$ と変更して格納部312へ格納する。各タスクの通信開始時刻を示す情報は、各タスクの実行要求情報に予め含まれていてタスク

制御部310がそれを参照してもよいし、タスク制御部310がタスク実行の際に通信の状況をモニタリングして通信開始時刻を学習してもよい。

【0049】

＜変形例5＞

実施形態では、バーストトラフィック回避のため、S-Task同士のみを同時実行の対象としていたが、S-Taskで通信を伴わないものに関してはP-Taskと同時実行してもバーストトラフィックの発生原因とはならない。この場合、図16に示すように、格納処理部311は、タスクAが通信を発生するタスクかを判断する(ステップS700)。通信を発生しないタスクの場合(ステップS700; No)、格納処理部311は、P-TaskであるかS-Taskであるかを問わず、全タスクの中で、タスクAの実行時刻以降で実行されるタスクのうち最も早い実行時刻を設定する(ステップS720)。一方、通信を発生するタスクだった場合(ステップS700; Yes)、実施形態と同様に、S-Task同士が同時実行するようステップS301(図6)の処理へと移る(ステップS710)。

10

【0050】

＜変形例6＞

実施形態では、フレームワーク300の格納処理部311でタスクの実行時刻の変更を実現していたが、アプリケーションプログラムによって格納処理部311を実現してもよい。この場合、図17に示すように、格納処理部311に代えて、その格納処理部311と同等の機能を実現する格納処理アプリケーション220を備える。格納処理アプリケーション220は、全てまたは一部のタスクから実行要求情報を受取り、実施形態のステップS100以降の処理を行う。

20

【0051】

＜変形例7＞

情報処理装置100は、携帯電話機のような通信装置でなくてもよく、スタンドアロン型の情報処理装置であってもよい。また、情報処理装置100において実行されるプログラムは、磁気テープ、磁気ディスク、フロッピー(登録商標)ディスク、光記録媒体、光磁気記録媒体、CD(Compact Disk)、DVD(Digital Versatile Disk)、RAMなどの記録媒体に記録した状態で提供し得る。

【0052】

＜変形例8＞

例えば地震などの緊急事態が発生したことを通知するメッセージ(いわゆる緊急地震速報)などのように、通信網を通じて一斉に配信される情報を、いわゆるスリープなどと呼ばれる休止状態の複数の情報処理装置100が受信した場合(以下では「同報受信」という)に、これらの情報処理装置100はほぼ同時に休止状態から稼動状態に遷移して、そのメッセージを表示したりする。そして、複数の情報処理装置100はそれぞれ稼動状態に遷移したことを契機として、自装置が実行すべきタスクがあればそれを実行する。この場合、複数の情報処理装置100において、通信を伴うS-Taskがほぼ同時に実行されると、バーストトラフィックが発生してしまうことがある。そこで、このようなバーストトラフィックの対策として、図18に示すように、タスク制御部310は、同報受信を検知する同報受信検知部317を備え、これによって同報受信が検知された場合には、格納処理部311が格納部312内のS-Taskの実行要求情報に含まれる実行時刻を書き換えるようにしてもよい。その動作は以下のとおりである。

30

40

【0053】

図19において、まず、同報受信検知部314が同報受信を検知すると(ステップS800; YES)、格納処理部311は、格納部312において、次のステップS810で未だ取得していないS-Taskの実行要求情報があるか否かを判断する(ステップS810)。取得していないS-Taskの実行要求情報があれば(ステップS810; YES)、格納処理部311は、それを取得し(ステップS820)、現在時刻が実行時刻を経過しているか否かを判断する(ステップS830)。前述したように、S-Taskは、システムによって最適化された時刻を実行時刻とするタスクであるから、休止状態では、実行要求情報に含まれる実行時刻を経過しても実行されないこともあり得る。周期性のないS-Taskは、休止状態では実行時刻を経過しても実行されないことはもちろんのこと、周期性のあるS-Task

50

の中にも、次回の実行時刻は設定されているが、必ずしもその実行時刻を正確に守る必要が無く、或る程度の期間をおいて繰り返し実行されるだけでよいタスクもある。この変形例ではそのようなタスクもS-Taskとみなす。ステップS 8 3 0では、このようなS-Taskを見つける。

【0054】

実行時刻を経過している場合（ステップS 8 3 0；YES）、格納処理部311は、そのタスクに周期性があるかを判断する（ステップS 8 4 0）。そのタスクに周期性がある場合（ステップS 8 4 0：YES）、次式1によって得られる時刻を、ステップS 8 2 0で取得した実行要求情報における実行時刻として上書きし、格納部312に格納する（ステップS 8 5 0）。

10

【0055】

$newSetTime = setTime + count \times interval \cdots \text{式1}$

ただし、

$count = 1 + (now - setTime) / interval$ [ただし、 $(now - setTime) / interval$ は整数型として算出される値]

であり、

$newSetTime$ ：上書きする新たな実行時刻

$setTime$ ：ステップS 8 2 0で取得された、上書き前の実行要求情報に含まれている実行時刻

$interval$ ：ステップS 8 2 0で取得された、上書き前の実行要求情報から特定されるタスクの周期

20

now ：同報受信を検知した時刻（つまり現在時刻）である。

【0056】

例えば、 $now = 18:00$ 、 $setTime = 17:55$ 、 $interval = 15$ （分）の場合、 $(now - setTime) / interval = 5 / 15$ だが、前述したようにこれは整数型なので、値は「0」となる。よって、 $count = 1 + 0 = 1$ となり、 $newSetTime = setTime + count \times interval = 17:55 + 1 \times 15$ （分） $= 18:10$ となる。 $newSetTime$ は、必ず、現在時刻 now 以降の値となる。

30

【0057】

一方、タスクに周期性がない場合（ステップS 8 4 0：NO）、格納処理部311は、次式2によって得られる時刻を、ステップS 8 2 0で取得した実行要求情報に実行時刻として上書きし、格納部312に格納する（ステップS 8 6 0）。

【0058】

$newSetTime = setTime + count \times t \cdots \text{式2}$

ただし、

$count = 1 + (now - setTime) / t$ [ただし、 $(now - setTime) / t$ は整数型として算出される値]

であり、

40

$newSetTime$ ：上書きする新たな実行時刻

$setTime$ ：ステップS 8 2 0で取得された、上書き前の実行要求情報に含まれている実行時刻

t ：所定の時間（例えば15分）

now ：同報受信を検知した時刻である。

【0059】

例えば、 $now = 18:00$ 、 $setTime = 17:55$ 、 $t = 10$ （分）の場合、 $(now - setTime) / t = 5 / 10$ だが、前述したようにこれは整数型なので、値は「0」となる。よって、 $count = 1 + 0 = 1$ となり、 $newSetTime = s$

50

$etTime + count \times t = 17:55 + 1 \times 10 \text{ (分)} = 18:05$ となる。 $newSetTime$ は、必ず、現在時刻 now 以降の値となる。

【0060】

なお、ステップS840、850において、格納処理部311は、上記の式1、式2を用いて実行時刻を算出していたが、その目的は、バーストトラフィックを防止するべく、同報受信を検知した時点以降の時刻でS-Taskの実行時刻を情報処理装置100毎に分散して求めることである。したがって、このように実行時刻を適当に分散し得るようなアルゴリズムであれば、式1、式2に代えて使用することが可能である。

【0061】

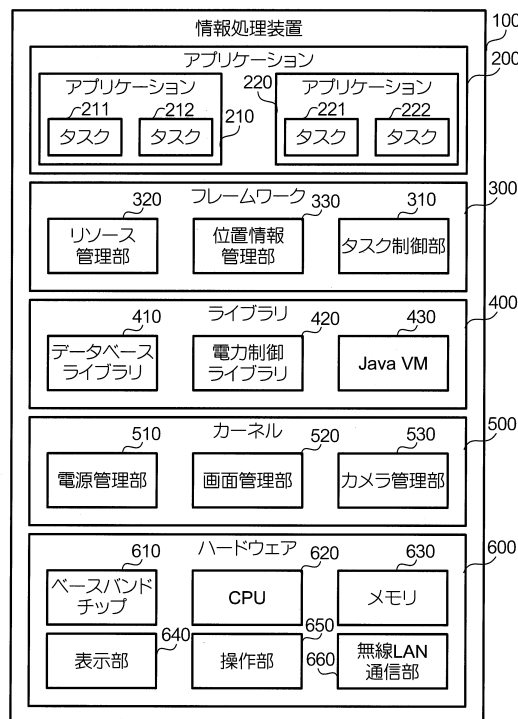
また、格納処理部311は、実行時刻を経過したS-Taskの実行時刻を上記のようにして上書きするのではなく、この実行時刻を削除しても構わない。この場合、実行時刻が削除されたタスクは実行されないことになるが、前述したトラフィックバーストを防止することを重視する場合には、このような実行中止という対策をとってもよい。

【符号の説明】

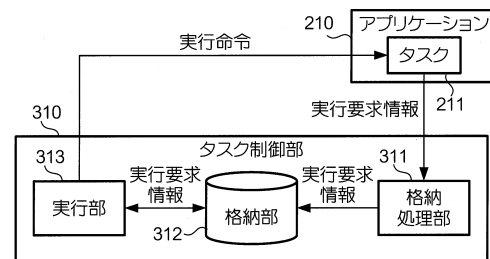
【0062】

100・・・情報処理装置、310・・・タスク制御部、311・・・格納処理部、312・・・格納部、313・・・実行部、314・・・状態管理部、315・・・実行履歴格納部、316・・・実行履歴監視部、317・・・同報受信検知部、330・・・制御ポリシー変換部、1000・・・サーバ装置、1100・・・アプリリスト、1200・・・変換ルール

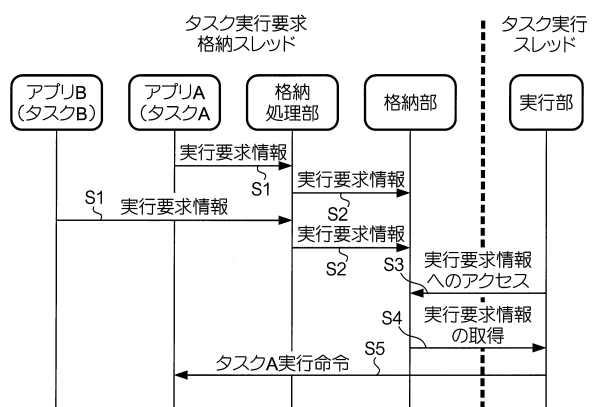
【図1】



【図2】



【図3】



【図 4】

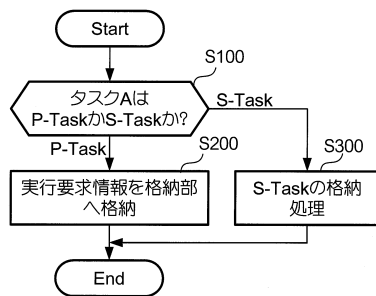
(A)

タスク名	設定時刻タイプ (P or S)	実行時刻
タスク A	P	2012/6/29/12:00

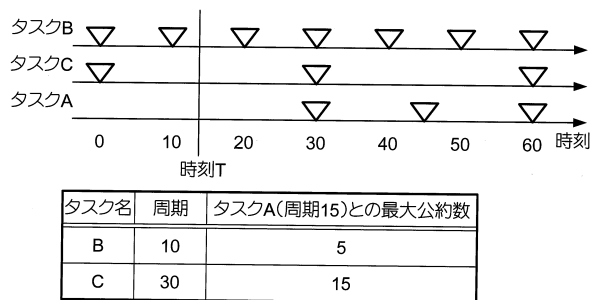
(B)

タスク名	設定時刻タイプ (P or S)	実行時刻	通信の有無情報 (通信有り:1 無し:0)	通信開始時刻 (ms)	通信終了時刻 (ms)	リソース消費量
タスク A	P	2012/6/29/12:00	1	100	3200	CPU:20 メモリ:30 無線送信:100 無線受信:200

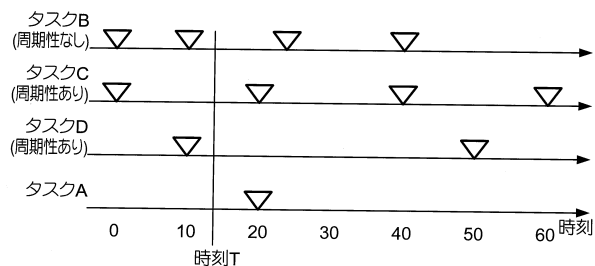
【図 5】



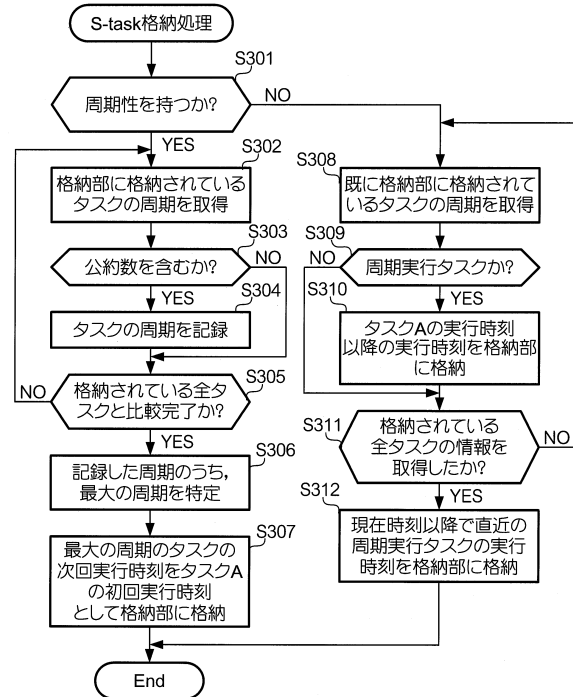
【図 7】



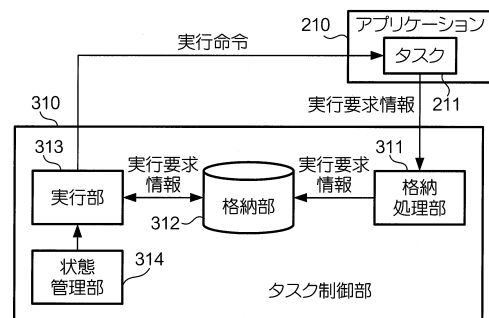
【図 8】



【図 6】



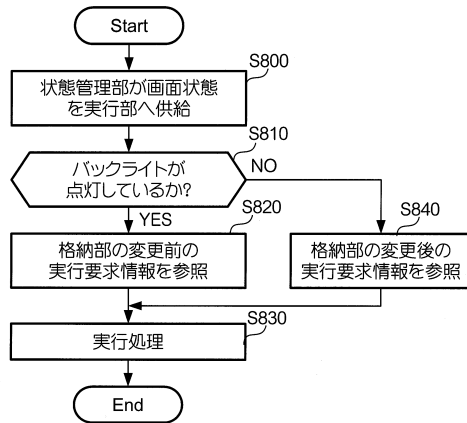
【図 9】



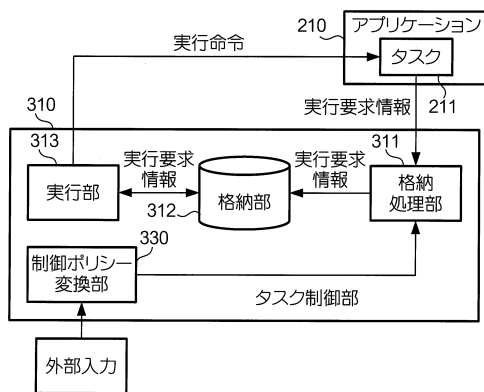
【図 10】

タスク名	設定時刻タイプ (P or S)	変更後の実行時刻	変更前の実行時刻
タスクA	P	2012/6/29/12:00	2012/6/29/11:28

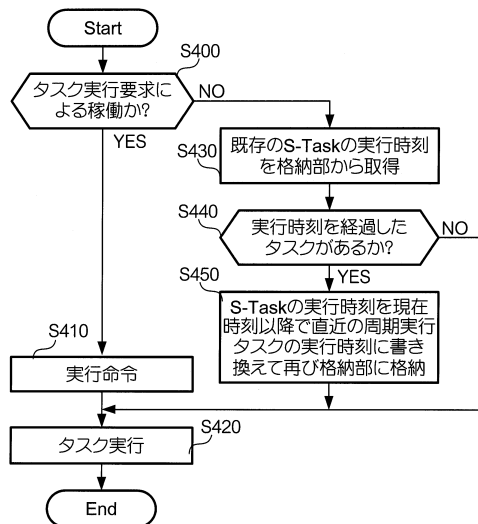
【図 1 1】



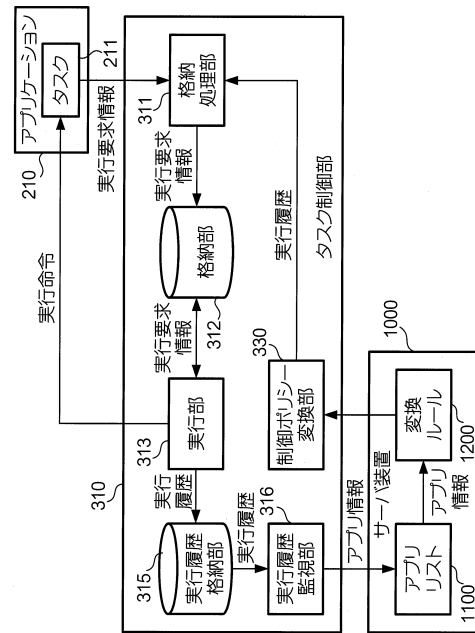
【図 1 2】



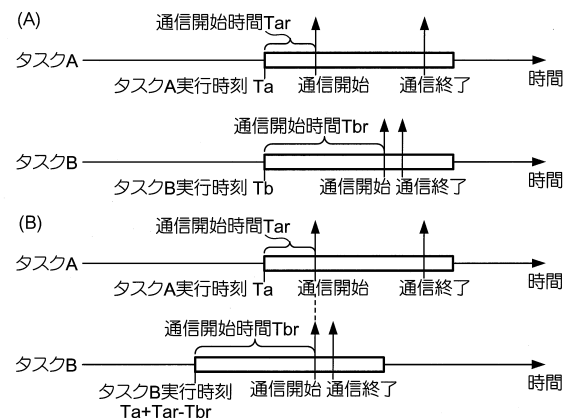
【図 14】



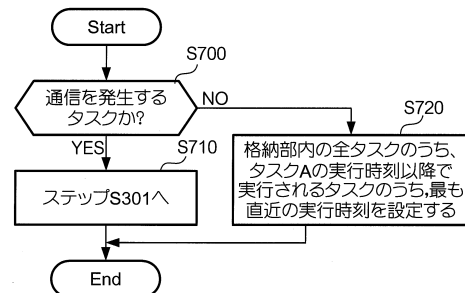
【図 1 3】



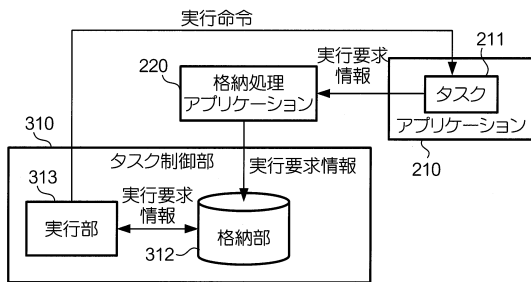
【図 15】



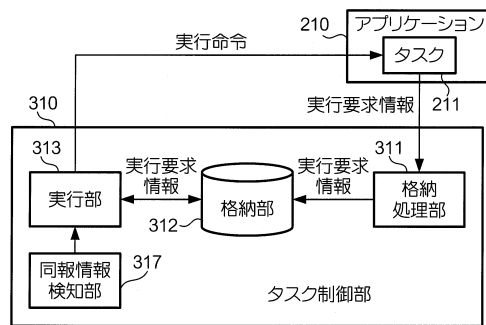
【図 16】



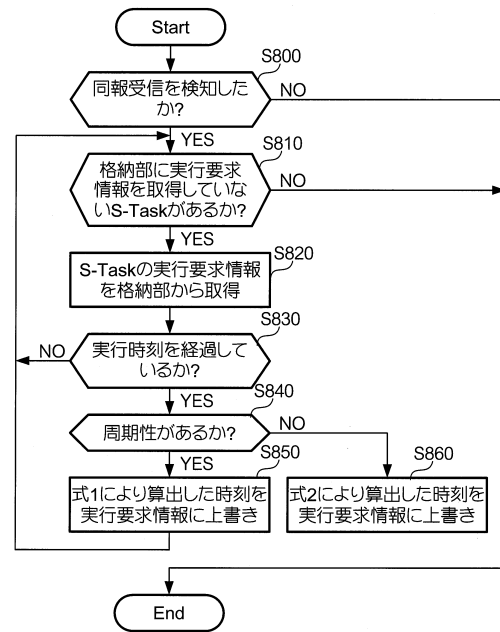
【図 17】



【図 18】



【図 19】



フロントページの続き

(72)発明者 川崎 仁嗣

東京都千代田区永田町二丁目11番1号 株式会社エヌ・ティ・ティ・ドコモ内

審査官 漆原 孝治

(56)参考文献 特開2010-160599 (JP, A)

特開2000-029722 (JP, A)

特開平05-088917 (JP, A)

(58)調査した分野(Int.Cl., DB名)

G06F 9/48