

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2020年6月18日(18.06.2020)



(10) 国際公開番号

WO 2020/121840 A1

- (51) 国際特許分類:
G06F 9/48 (2006.01) *G06F 9/50* (2006.01)
- (21) 国際出願番号: PCT/JP2019/046724
- (22) 国際出願日: 2019年11月29日(29.11.2019)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (30) 優先権データ:
特願 2018-232407 2018年12月12日(12.12.2018) JP
- (71) 出願人: 日立オートモティブシステムズ株式会社(HITACHI AUTOMOTIVE SYSTEMS, LTD.) [JP/JP]; 〒3128503 茨城県ひたちなか市高場 2 5 2 0 番地 Ibaraki (JP).
- (72) 発明者: 堀 口 辰 也 (HORIGUCHI Tatsuya); 〒1008280 東京都千代田区丸の内一丁目 6 番 6 号 株式会社日立製作所内 Tokyo (JP). 石

郷岡 祐(ISHIGOOKA Tasuku); 〒1008280 東京都千代田区丸の内一丁目 6 番 6 号 株式会社日立製作所内 Tokyo (JP). 小田 裕弘(ODA Yasuhiro); 〒3128503 茨城県ひたちなか市高場 2 5 2 0 番地 日立オートモティブシステムズ株式会社内 Ibaraki (JP). 成沢 文雄(NARISAWA Fumio); 〒3128503 茨城県ひたちなか市高場 2 5 2 0 番地 日立オートモティブシステムズ株式会社内 Ibaraki (JP).

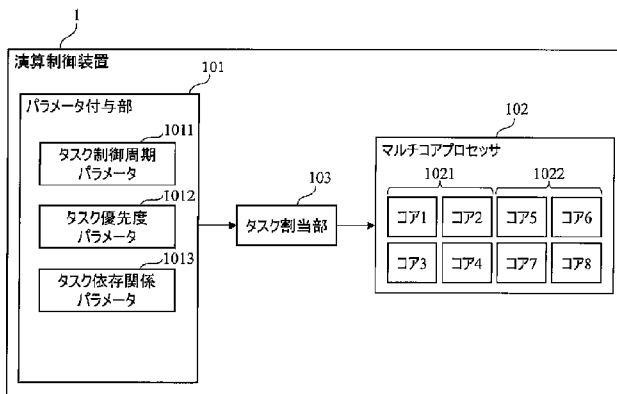
(74) 代理人: 戸田 裕二(TODA Yuji); 〒1008220 東京都千代田区丸の内一丁目 6 番 1 号 株式会社日立製作所内 Tokyo (JP).

(81) 指定国(表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,

(54) Title: ARITHMETIC CONTROL DEVICE

(54) 発明の名称: 演算制御装置

図 1



- 1 Arithmetic control device
101 Parameter assigning unit
102 Multicore processor
103 Task allocating unit
1011 Task control cycle parameter
1012 Task priority parameter
1013 Task dependence relationship parameter
1021 Core 1, core 2, core 3, core 4
1022 Core 5, core 6, core 7, core 8

(57) Abstract: A system that uses a processor having a plurality of cores, wherein fluctuations in processing time can be reduced when different processes are performed simultaneously in the plurality of cores, thus improving behavior predictability of the system, and increasing the probability of guaranteeing execution time constraints. An arithmetic control device according to the present invention comprises a processor having a plurality of cores, and a task allocation unit that allocates a task to the plurality of cores. The task allocation unit determines whether or not it is possible to perform parallel processing on a plurality of sub-tasks constituting the task, and if parallel processing is possible, the task allocation unit allocates the plurality of sub-tasks to the plurality of cores under prescribed conditions.

HN, HR, HU, ID, IL, IN, IR, IS, JO, KE, KG, KH,
KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY,
MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ,
NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT,
QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL,
SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA,
UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) 指定国(表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

添付公開書類：

- 一 国際調査報告（条約第21条(3)）

(57) 要約：複数のコアを有するプロセッサを用いたシステムにおいて、複数のコア上で同時に異なる処理を行った際の処理時間の変動を低減することを可能にし、これによりシステムの挙動予測性を高め、実行時間制約の保証可能性を高める。本発明に係る演算制御装置は、複数のコアを有するプロセッサと、前記複数のコアに対し、タスクを割り当てるタスク割当部とを備える演算制御装置である。前記タスク割当部は、前記タスクを構成する複数のサブタスクを並列化処理の対象とすることが可能であるか否かを判断し、並列化処理が可能である場合には、所定の条件下で前記複数のサブタスクを前記複数のコアに割り当てるよう構成される。

明 細 書

発明の名称： 演算制御装置

技術分野

[0001] 本発明は、演算制御装置に関し、より詳しくはマルチコア又はメニーコアと言われる複数コアを有するプロセッサを制御する演算制御装置に関する。

背景技術

[0002] 近年、組込み用プロセッサにおける性能向上が進み、高負荷な演算処理の導入が自動車（特に自動運転）、ビッグデータ処理、創薬、遺伝子解析、産業機器等の分野において進んでいる。その結果、それらの分野における制御装置において求められる処理性能が増大している。このため、マルチコアもしくはメニーコアプロセッサなど複数コアを有するプロセッサを活用した制御装置（ECU: Electronic Control Unit）が採用され、その処理性能向上が求められている。

[0003] 例えば自動車分野においては、高度な自動運転システムの実現に向け、自動運転を制御する上位の制御装置である自動運転ECUを筆頭に、制御装置の処理負荷の増大が課題となっている。このため、自動車分野においても、2コア、4コア、8コアといったマルチコアプロセッサの導入が開始されている。今後も更にコア数は増大し、コア数が例えば10以上のメニーコアプロセッサの利用開始も想定される。

[0004] 一方で、同分野におけるソフトウェア設計においては、例えばモデルベースデザインからの自動ソースコード生成など、シングルコアプロセッサ向けの設計手法が依然用いられる場合がある。シングルコアプロセッサ向けの既存の設計手法のまま単純にマルチコアプロセッサやメニーコアプロセッサに移行するだけでは、今後の負荷増大トレンドにおける、これらプロセッサの性能メリットを活かすことが難しくなる。特許文献1ではこのような背景に対し、マルチコアプロセッサに対して複数のタスクを効率的に割り当てる（準最適なスケジューリングを提供する）例が開示されている。

[0005] 一方で、特許文献1では、通常時は事前に規定されるスケジューリング方式に基づき各プロセッサコアへの処理を割り当てた静的なスケジューリングを用いるが、負荷が高い状況において、タスクを効率的に実行するための動的スケジューリングを行う例が示されている。自動車分野や産業機器分野においては、制御の安定性のため実行時間制約を保証することが強く求められるが、同文献については実行時間制約への言及はなく、仮に通常動作時（静的スケジューリングで動作する場合）には設計者により適切なスケジューリングが行われていたとしても、少なくとも動的にスケジューリングを行った際の実行時間制約の保証は困難である。特にマルチコアプロセッサにおいては、ある処理が別の処理の実行時間へ影響を与える場合がある。この場合、実行時間の制約を保証することが困難になり得る。このため、実行時間制約を満たすためには、複数のコア上で同時に演算を行った際の処理時間の変動に対応するための方法が必要となる。

先行技術文献

特許文献

[0006] 特許文献1：特開2008-171153号公報

発明の概要

発明が解決しようとする課題

[0007] 本発明の目的は、上述のような複数のコアを有するプロセッサを用いたシステムにおいて、複数のコア上で同時に異なる処理を行った際の処理時間の変動を低減することを可能にし、これによりシステムの挙動予測性を高め、実行時間制約の保証可能性を高めることを可能にした演算制御装置を提供することである。

課題を解決するための手段

[0008] 本発明に係る演算制御装置は、複数のコアを有するプロセッサと、前記複数のコアに対し、タスクを割り当てるタスク割当部とを備える演算制御装置である。前記タスク割当部は、前記タスクを構成する複数のサブタスクを並

列化処理の対象とすることが可能であるか否かを判断し、並列化処理が可能である場合には、所定の条件下で前記複数のサブタスクを前記複数のコアに割り当てるよう構成される。

発明の効果

[0009] 本発明によれば、マルチコアプロセッサを用いる演算制御装置において、複数のコア上で同時に異なる処理を行った際の処理時間の変動を低減することを可能にし、これによりシステムの挙動予測性を高め、実行時間制約の保証可能性を高めることができる。

図面の簡単な説明

[0010] [図1]第1の実施の形態に係るマルチコアプロセッサの演算制御装置のうち、演算処理に関連する構成部分を示した概略ブロック図である。

[図2]第1の実施の形態における、演算制御装置1と、ソフトウェアコンポーネント（搭載機能）との関係を示す概略図である。

[図3]ソフトウェアコンポーネント（搭載機能）の依存関係の一例を説明する概略図である。

[図4]ソフトウェアコンポーネント（搭載機能）と、対応するタスクとの関係を示す表である。

[図5]演算制御装置1に割り当てられる搭載機能をモデルベース設計手法により表現した図である。

[図6]モデルベース設計時に演算制御装置1に割り当てられる搭載機能に対応するタスクを表す表である。

[図7]第1の実施の形態において、タスクの各々について付与される各種のパラメータの一例を説明する表である。

[図8]第1の実施の形態における、データ依存関係を示すタスクグラフの一例である。

[図9]第1の実施の形態において、タスクの各々に付与される優先度のパラメータの一例を示す表である。

[図10]第1の実施の形態において、タスクD内でのサブタスクの並列化を行

わない場合のタイムチャートの一例である。

[図11]タスクDにおいて、並列化処理可能なサブマスクを並列化処理した場合におけるタスクグラフと、設定される平均実行時間との関係を示す表である。

[図12]第1の実施の形態において、タスクD内のサブタスクを並列化した場合におけるタイムチャートの一例である。

[図13]第1の実施の形態における、イベント駆動処理タスクHが発生したこと起因して時間制約違反が発生した場合におけるタイムチャートの一例である。

[図14]第1の実施の形態の変形例1における、タスクDの並列化と優先度設定方法を示す図である。

[図15]第1の実施の形態の変形例1における、タスク優先度パラメータの設定例を示すリストである。

[図16]第1の実施の形態の変形例2における動作を示すタイムチャートの一例である。

[図17]第2の実施の形態に係るマルチコアプロセッサの演算制御装置のうち、演算処理に関連する構成部分を示した概略ブロック図である。

[図18]第2の実施の形態において、タスクの各々について付与される各種のパラメータの一例を説明する表である。

[図19]第2の実施の形態における、デッドラインパラメータの設定例を示すリストである。

発明を実施するための形態

[0011] 以下、添付図面を参照して本実施形態について説明する。添付図面では、機能的に同じ要素は同じ番号又は対応する番号で表示される場合もある。なお、添付図面は本開示の原理に則った実施形態と実装例を示しているが、これらは本開示の理解のためのものであり、決して本開示を限定的に解釈するために用いられるものではない。本明細書の記述は典型的な例示に過ぎず、本開示の特許請求の範囲又は適用例を如何なる意味においても限定するもの

ではない。

[0012] 本実施形態では、当業者が本開示を実施するのに十分詳細にその説明がなされているが、他の実装・形態も可能で、本開示の技術的思想の範囲と精神を逸脱することなく構成・構造の変更や多様な要素の置き換えが可能であることを理解する必要がある。従って、以降の記述をこれに限定して解釈してはならない。

[0013] [第1の実施の形態]

図1を参照して、第1の実施の形態に係る演算制御装置を説明する。以下では、コアの数が8個であるマルチコアプロセッサを制御する演算制御装置を例として説明するが、コアの個数は特定のものに限定されるものではない。例えば、コアが10個以上のメニーコアプロセッサを制御する演算制御装置にも、本開示の技術的思想の範囲と精神を逸脱することなく、同様の方式の採用が可能である。

[0014] 図1は、第1の実施の形態に係るマルチコアプロセッサの演算制御装置のうち、演算処理に関連する構成部分を示した概略ブロック図である。演算制御装置1は、パラメータ付与部101と、マルチコアプロセッサ102と、タスク割当部103とを備える。

[0015] パラメータ付与部101は、車両を制御するために必要となるハードウェア及びソフトウェアに関する情報をパラメータとしてタスクに付与し、記憶部に記憶する機能を有する。ハードウェアに関するパラメータには、マルチコアプロセッサ102のマルチコア構成、バス配線情報、センサ（図示せず）のセンシング情報（センサ値）等が含まれる。

[0016] パラメータ付与部101に記憶されるパラメータは、タスク割当部103による割当を行う場合の判断指標ともなり得る。一例として、パラメータ付与部101は、タスク制御周期パラメータ記憶部1011と、タスク優先度パラメータ記憶部1012と、タスク依存関係パラメータ1013とを備える。タスク制御周期パラメータ記憶部1011は、マルチコアプロセッサ102により実行される各タスクの制御周期を規定するパラメータを記憶する

。また、タスク優先度パラメータ記憶部 1012 は、各タスクが他のタスクとの関係で優先して処理（実行）されるべき度合を示す優先度を規定するパラメータを記憶する。

[0017] タスク依存関係パラメータ記憶部 1013 は、タスク間の依存関係を示す依存関係パラメータを記憶する。依存関係パラメータは、例えばあるタスク A で処理した結果を次のタスク B の入力として処理するため、タスク A の終了がタスク B の開始条件となるような関係を示す。

[0018] マルチコアプロセッサ 102 は、複数のコア 1～8 を有しており、各コア 1～8 はそれぞれ並列にタスク処理を実行可能に構成されている。一例として、マルチコアプロセッサ 102 は、性能の異なる 2 種類のコアを 4 コアずつ有することができる。ここでは、相対的に性能の高いコア 1～4 を高速動作コア 1021、性能の低いコア 5～8 を低速動作コア 1022 と呼称する。性能の段階は 2 段階には限られず、3 段階以上であってもよいし、全てのコアが同一の性能を有していてもよい。

[0019] タスク割当部 103 は、パラメータ付与部 101 から取得されたパラメータ（タスク制御周期、タスク優先度、タスク依存関係など）に基づき、各タスクをマルチコアプロセッサ 102 の各コアへ割り当てる。

[0020] このような演算制御装置 1 に対して要求される搭載機能は、いくつかの形式により与えられる。例えば、図 2 に示すように、各種機能 1、2、3、4・・・が、ソフトウェアコンポーネントとして C 言語などのソースコードにて記載されて演算制御装置 1 に与えられる場合を考える。

[0021] この場合、同ソースコードに基づき、図 3 に示すように、各機能 1、2、3、4・・・（ソフトウェアコンポーネント間）の依存関係が解読される。例えば、機能 2 は、機能 1 の実行が完了した後において実行が可能であり、機能 3 は機能 2 の実行が完了した後において実行が可能であることがソースコードに基づき解読される。

[0022] さらに、各機能 1、2、3、4・・・をマルチコアプロセッサ 102 に割り当てるためのタスクが定義される。各機能 1、2、3、4（ソフトウェア

コンポーネント)とタスクとの関係の一例を図4に示す。

[0023] また異なる入力情報の例として、図5のようなモデルベース設計手法にて記述されたモデルが与えられた場合には、図6のように各機能と対応させる形でタスクが定義される。例えば、機能2、4中のサブ機能2-1、2-2、2-3、4-1、4-2に対しても、それぞれ対応するタスクが定義される。

[0024] タスクが定義されると、図7のテーブルに示すように、タスクの各々について、各種のパラメータが、例えばソフトウェアコンポーネントのソースコードに基づいて生成される。一例としては、図7に示すように、平均実行時間(高速コアを用いた場合)、各タスクに含まれるサブタスクの並列化の可否、及びタスク制御周期等のパラメータが生成される。このうちタスク制御周期のパラメータは、タスク制御周期パラメータ記憶部1011に記憶される。

なお、図7は、機能1~4とは別の、イベント駆動処理(割り込み処理)に対応するタスクHについても平均実行時間を計算し、パラメータを決定した場合について示している。

[0025] 図8に、図7に示したタスクを実行する場合のタスクグラフの一例を示す。タスクグラフは、演算制御装置1により、タスク制御周期、タスク依存関係、その他のパラメータに従って生成される。各タスクは、同一のタスク制御周期を有するタスク毎に分類されるとともに、タスク依存関係に従って順序付けされる。この例では、30msの制御周期で動作するタスクA~Eについての1のタスクグラフが形成され、50msの制御周期で動作するタスクF、Gについて別のタスクグラフが形成されている。

[0026] 図8に例示されるタスクグラフでは、各タスク間の矢印によりタスク間の依存関係が表現されている。この依存関係は、タスク依存関係パラメータ1013に記述される。このようなタスクグラフが形成されると、直列に並ぶ複数のタスクの組は、マルチプロセッサ102中の1つのコアにおいて実行され、異なる組のタスクは、異なるコアにおいて実行され得る。例えば、タ

スク A、B、E は、コア 1 において実行され、タスク C、D はコア 2 において実行され、タスク F、G はコア 3 にて実行され得る。

[0027] また、タスクが定義されると、タスク毎に優先度がパラメータとして定義され、タスク優先度パラメータ記憶部 1012 に記憶される。図 9 は、各タスクについて定義された優先度の一例を示している。この図 9 の数字は、小さい数字であるほど優先度が高いことを意味している。例えば、タスク F、G は優先度「15」であり、タスク A～E の方がより高い優先度「10」を有する。タスク H は、これよりも更に高い優先度「5」を有する。この例では、タスク制御周期が短いタスクほど、高い優先度を与える、という制御ポリシーにより、優先度及びタスク制御周期が各タスクに与えられている。

[0028] 上述のようにして決定されたタスク制御周期、依存関係、優先度に従って各タスクを実行した（演算を実行した）場合におけるタイムチャートの一例を図 10 に示す。この図 10 では、コア上でのタスク切り替えや、タスク分割に伴う時間オーバーヘッドは発生しないものと仮定している。

[0029] タスク依存関係パラメータに従った依存関係に基づき、30ms のタスク制御周期を有するタスク A～E が実行される。このとき、タスク A、B とタスク C、D とは、異なるコア 1、2 により並列に（同時に）実行され得る。しかし、A、C、D、E はその順に実行されなければならない、このため、その処理がネック（クリティカルパス）となり、個々のタスクの実行時間次第では、30 [ms] の周期制約を満たすことは出来ず、時間制約違反が生じてしまう。

[0030] そこで、本実施の形態では、クリティカルパス上にある複数のタスク A、C、D、E のうち、更に複数のサブタスクに分割され、しかもそのサブタスクが並列化可能なタスクに関し、所定の条件下で並列化を実行することで、実行時間を短縮する。サブタスクへの分割は、タスク割当部 103 において、タスク制御周期パラメータ、タスク優先度パラメータ、タスク依存関係等に基づいて判断され、実行され得る。

[0031] 図 11 に、タスク D を複数のサブタスク D_{pre} 、 $D_{p0} \sim D_{p3}$ 、及び D_{post}

に分割し、その一部のサブタスク（例えばサブタスク $D_{p0} \sim D_{p3}$ ）を並列化した場合におけるタスクグラフと、決定される平均実行時間（高速コアの使用時）の例を示す。

[0032] 1つのタスク D から生成された複数のサブタスクは、その中の少なくとも2以上が並列化され、異なるコアにおいて同時に実行可能とされることにより、タスクの総実行時間（平均）を短縮することができる。

[0033] 好適には、タスク D 中の複数のサブタスクを並列化するにあたり、複数のサブタスクを並列化前サブタスク D_{pre} （21）、並列化対象サブタスク D_p （22）（ $D_{p0} \sim D_{p3}$ ）、及び並列化後サブタスク D_{post} （23）に分割することができる。並列化前サブタスク D_{pre1} は、タスク D 中のサブタスク中で最初に実行されるサブタスクである。この並列化前サブタスク D_{pre1} の実行後、複数の並列化対象サブタスク D_p が並列に、複数のコアにおいて実行され得る。また、並列化後サブタスク D_{post} は、並列化対象サブタスク $D_{p1} \sim D_{p3}$ の実行が全て終了した場合に実行されるタスクである。

このように、並列化対象サブタスク D_p の前後に、並列化の対象外のサブタスクを置くことで、実行時間の予測が容易になるとともに、演算制御装置1での処理負荷も軽減することができる。

[0034] 図11のように、1つのタスク D が複数のサブタスクに分割される場合、サブタスクの各々の平均実行時間（高速コアの使用時）の合計 T_{total} は、元のタスク D の平均実行時間 T_d 以下となるように設定する。前述のように、コア上でのタスク切替や、タスク分割に伴う時間オーバーヘッドは仮定しないならば、平均実行時間の合計 T_{total} は、元のタスク D の平均実行時間 T_d と等しい値とすることができる（ $5 \times 6 = 30 \text{ ms}$ ）。なお、この新たに生成されたサブタスク D_{pre} 、 $D_{p0} \sim D_{p3}$ 、 D_{post} の間の依存関係、及び平均実行時間は、パラメータ付与部101に記憶される。

[0035] 図11のようにしてタスク D が複数のサブタスクに分割され且つ並列化処理された場合におけるタイムチャートの一例を、図12に示す。この例では、 30 ms の制御周期を満足するため、並列化処理が可能なサブタスク D_{p0}

～ D_{p3} が優先的に処理され、その前に並列化前サブタスク D_{pre} がタスクCの実行終了後に処理を開始される。並列化処理の開始の際、コア1上でタスクBが実行中であれば、それを一度中断することもできる。これにより、4つのコア1～4の全てを実行可能状態とすることができ、それぞれにサブタスク $D_{p0} \sim D_{p3}$ を割り当て、4つのコア1～4において同時に処理を開始することができる。サブタスク $D_{p0} \sim D_{p3}$ に関しては、並列化前サブタスク D_{pre} の終了後、同期信号を発生させることにより、処理の開始タイミングが揃うようにすることが好適である。サブタスク $D_{p0} \sim D_{p3}$ の実行が全て終了すると、その後に並列化後サブタスク D_{post} の実行を開始することができる。また、コア1では、中断していたタスクBの実行を再開することができる。

[0036] また、50msのタスク制御周期を有しコア3で実行中であるタスクFも、タスクBと同様に、並列化処理が開始可能となった段階で、一度処理を中断する。その後、並列化処理の対象とされたサブタスク D_{p2} の実行をコア3において開始する。サブ D_{p3} の終了後、コア3では再び中断されていたタスクFの処理が再開される。

[0037] なお、タスクFもタスクDと同様に、サブタスクに分解して並列化処理が可能であれば（図7参照）並列化処理の対象としてもよいことは言うまでもない。ただし、タスクFは、タスクDとは異なり、タスク制御周期が長く、このため、優先度がタスクA～Eに比べ低い。このため、並列化処理を実行しなくても、所定のタスク制御周期の中で処理を終了することができる。このため、図12の例では、タスクFに関してはサブタスクの並列化処理は実行していない。このように、並列化処理が可能なタスクにおいても、定められたタスク制御周期の中で処理を完了することが可能であれば並列化処理を実行しないことを選択することが可能である。なお、タスクFをサブタスク F_p に分解して並列化処理を実行する場合においては、サブタスク F_p の優先度は、分割前のタスクFの優先度に比べ低く設定することができる。

[0038] <変形例1>

図13～図15は、第1の実施の形態の変形例1を示している。この変形例1は、並列化対象サブタスク $D_{p0} \sim D_{p3}$ のコア1～4での処理中に割込みタスクHが発生した場合を想定しており、図13はそのタイムチャートの一例を示している。

[0039] タスクHが発生した場合、タスクHは、例えばコア4において、サブタスク D_{p3} の実行を中断して実行される。イベント駆動処理タスクであるタスクHは、タスク制御周期を有するタスクA～Gとは異なり、タスク制御周期とは無関係にイベントの発生に対応してランダムに生成される。このため、タスクHは、設計者が事前に実施タイミングを設計することは難しい。なお、発生ベースでのタスクHの実行を可能にするため、タスクHの優先度は、一般に他のタスクに比べ高く設定され得る。

[0040] しかし、並列化後サブタスク D_{post} は、並列化対象サブタスク D_p の完了後に実行可能となる。このため、タスクHの動作によりタスク D_{post} の実行開始時刻が遅れると、結果、タスクA～Eについての処理がタスク制御周期中に完了せず、時間制約違反が発生する虞がある（図13参照）。

[0041] このように、イベント駆動処理（割込み処理）など、優先度が並列化対象サブタスクに対して高く設定されるタスクが存在する場合、並列化対象サブタスクの一部の処理が中断され、実行時間制約を満たすことができない場合があり得る。

[0042] そこで、この第1の実施の形態の変形例1では、図14及び図15に示すように、全てのサブタスクに対し同一の優先度 P_d を与える代わりに、並列化対象サブタスク D_p の優先度を、並列化処理前サブタスク、及び並列化処理後サブタスクと比較して高い優先度 P_{high} に変更する。この優先度 P_{high} は、イベント駆動処理タスクHの優先度よりも高くされる。例えば、並列化前サブタスク D_{pre} 及び並列化後サブタスク D_{post} に対しては、その前後のタスクC、Eの優先度と同一の優先度（例： $P_d = 15$ ）を付与する一方、並列化対象サブタスク $D_{p0} \sim D_{p3}$ に対しては、タスクC、E、更にはタスクHよりも高い優先度（例： $P_{high} = 1$ ）を付与する。

[0043] このように、並列化前サブタスク D_{pre} 及び並列化後サブタスク D_{post} に関しては、並列化前の優先度を引き継ぎ、他タスクA～C、E～H等との関係で優先度の調整を行することができる。例えば、元のタスクDよりも高い優先度を有する他のタスクがあれば、そちらを優先して実行するよう、優先度を調整することはできる。ただし、並列化対象サブタスク $D_{p0} \sim D_{p3}$ については、他のタスクによる動作阻害が発生した際の実行ゆらぎが、直接的に実行時間制約の違反に繋がってしまう。そのため、並列化対象サブタスク $D_{p0} \sim D_{p3}$ のみについては可能な限り最大の優先度を与える。これにより、並列化対象サブタスクを含む一連のタスクについて、動作保証を行うことができる。

[0044] <変形例2>

図16は、第1の実施の形態の変形例2を示している。この変形例2は、変形例1と同様に、タスクA～Gのコア1～4での処理中に割り込みタスクHが発生した例を示す。ただし、この変形例2は、割り込みタスクHが発生した場合において、コア5を割り込みタスクHの処理専用として利用するよう、タスク割当部103が構成されている。

[0045] 第1の実施の形態を適用するにあたり、イベント駆動処理であるタスクHの動作を阻害することは不可能である場合は、タスクHについて別途専用コアを定義することが可能である。この場合、タスク制御周期を有するタスクA～Gの処理のためのコア群（コア1～4）と、イベント駆動処理専用のコア群（コア5）の2種類が存在するマルチコアプロセッサ102を利用することができる。マルチコアプロセッサ102がこのような構成を取る場合、イベント駆動処理専用コア（コア5）を別途設けることにより、全体的なプロセッサ使用効率は低下することがあり得る。しかし、その一方で、マルチコアプロセッサ102上でのタスク挙動の予測性を高め、周期制約の充足を保証することが可能となる。

[0046] また、この変形例2では、タスク制御周期を有するタスクA～Gの動作が割り込みタスクHにより阻害されないことから、外部メモリやキャッシュ利

用のタイミングが固定的となるため、キャッシュ利用効率向上による処理性能向上効果が期待できる。

[0047] なお、本実施の形態では簡単のため、並列化対象サブタスクが1つのケースで説明しており、その際には並列化対象サブタスクに最高の優先度を与えることが可能となる。しかし、実際には並列化すべきタスクが複数存在する場合が想定される。そのような場合においては、より時間制約の厳しい順に、最高優先度から優先度を順次下げて付与することが可能である。

[0048] また、上記の実施の形態では、コアの種類が1種類であるマルチコアプロセッサ102を前提として説明した。一つのチップ内に複数種類のコアを持つ、ヘテロジニアスなマルチコアもしくはメニーコアプロセッサにも、本実施の形態の方式は採用可能である。

タスク制御周期を有するタスクA～Gがイベント駆動処理であるタスクHに対して処理負荷が高いケースや、イベント駆動処理であるタスクHの出現頻度が低いケースなどでは、このようなヘテロジニアスなコアにおいて、より性能の高いコアへタスク制御周期を有するタスク割り当て、性能の低いコアへイベント駆動処理であるタスクHを割り当てることにより、更なる演算性能向上を図ることが可能となる。

[0049] [第2の実施の形態]

次に、第2の実施の形態に係る演算制御装置を、図17～図19を参照して説明する。

[0050] 図17は、第2の実施の形態に係るマルチコアプロセッサの演算制御装置のうち、演算処理に関連する構成部分を示した概略ブロック図である。第1の実施の形態（図1）と同一の部分に関しては、図17において図1と同一の参照符号を付しているため、以下では重複する説明は省略する。

[0051] この第2の実施の形態では、パラメータ付与部101は、タスク優先度パラメータ記憶部1012は有しておらず、代わりに、各タスクが絶対に処理を完了すべき時刻である絶対デッドライン示すデッドラインパラメータを記憶するデッドラインパラメータ記憶部1014を備えている。すなわち、こ

の第2の実施の形態では、あるタスクをサブタスクに分解して並列化処理を実行する場合において、そのサブタスクの優先度を高くするのではなく、その絶対デッドラインを変更し、パラメータとして記憶されるようにされている。並列化対象のサブタスクの絶対デッドラインが変更されることにより、絶対デッドラインが近いタスクが優先的に実行され、結果としてタスク実行の時間制約を遵守することが可能になる。なお、この例では、絶対期限である絶対デッドラインをパラメータとして生成・記憶しているが、絶対デッドラインに代えて、例えばタスク制御周期の開始時刻からの相対時刻をデッドラインとして生成・記憶することも可能である。

[0052] 図18は、第2の実施の形態において、ソフトウェアコンポーネントのソースコードに基づいて生成された、タスクの各々についての各種パラメータを示すテーブルである。30msのタスク制御周期を有するタスクA～Eについて時間制約違反が生じてしまうと判断される場合には、並列化が可能であるタスクDについてサブタスク (D_{pre} 、 $D_{p0} \sim D_{p3}$ 、 D_{post}) への分割及び並列化が実行される。

[0053] 図19に示すように、並列化対象サブタスク $D_{p0} \sim D_{p3}$ については、絶対デッドラインを元のタスクDの絶対デッドラインに比べ短くすることができる。ここでは、演算限界時間から逆算して絶対デッドラインを20msと定め、元の絶対デッドライン (30ms) よりも短くしている。デッドラインが長いままとされる場合、割込みタスクHが実行により、並列化対象サブタスク $D_{p0} \sim D_{p3}$ のデッドライン保証が困難となる。並列化対象サブタスク $D_{p0} \sim D_{p3}$ の絶対デッドラインをより短く設定することにより、タスク $D_{p0} \sim D_{p3}$ の実行優先度を高めることができる。

[0054] タスクHについては、第1の実施の形態の変形例2と同様に、タスク制御周期を有するタスクの処理用のコア (コア1～4) と、イベント駆動処理のタスクの処理専用コア (コア5) の2種類が存在するマルチコアプロセッサを採用することも可能である。このような構成を取る場合、イベント駆動処理専用コアを別途設けることにより全体的なプロセッサ使用効率が低下す

るデメリットがあるが、マルチコアプロセッサ上でのタスク挙動の予測性を高め、周期制約の充足を保証することが可能となる。また周期処理タスクの動作が阻害されないことから、外部メモリやキャッシュ利用のタイミングが固定的となるため、キャッシュ利用効率向上による処理性能向上効果が期待できると同時に、動的なスケジューリングを行うことにより、第1の実施の形態における静的なスケジューリングよりもタスク設計工数の低減が可能となる。

[0055] 以上、本発明の実施形態を説明したが、これらの実施形態は、例として提示したものであり、発明の範囲を限定することは意図していない。これら新規な実施形態は、その他の様々な形態で実施されることが可能であり、発明の要旨を逸脱しない範囲で、種々の省略、置き換え、変更を行うことができる。これら実施形態やその変形は、発明の範囲や要旨に含まれるとともに、特許請求の範囲に記載された発明とその均等の範囲に含まれる。

上記各構成、機能、処理部、処理手段等は、それらの一部を、例えば集積回路を設計する等によりハードウェアで実現してもよい。また、上記の各構成、機能等は、プロセッサがそれぞれの機能を実現するプログラムを解釈し、実行することによりソフトウェアで実現してもよい。各機能を実現するプログラム、テーブル、ファイル等の情報は、メモリ、ハードディスク、SSD (Solid State Drive) 等の記録装置、ICカード、SDカード、DVD等の記録媒体に格納することができる。

[0056] また例えば、上記実施形態において、制御周期30msのタスク群と50msのタスク群、及びイベント駆動タスクの組合せを示しているが、本発明はこれに限定されるものではない。複数種類の制御周期の存在、異なるデッドライン時刻、イベント駆動処理の有無等に依らず、本発明は並列化により生成されるタスクの処理優先権を変更することで、処理時間の変動を低減することができる、実行時間制約の保証可能性を高めることができる。

符号の説明

[0057] 1…演算制御装置、101…パラメータ付与部、1011…タスク制御周期

パラメータ記憶部、１０１２…タスク優先度パラメータ記憶部、１０１３…タスク依存関係パラメータ記憶部、１０１４…デッドラインパラメータ記憶部、１０２…マルチコアプロセッサ、１０２１、１０２２…コア、１０３…タスク割当部。

請求の範囲

- [請求項1] 複数のコアを有するプロセッサと、
前記複数のコアに対し、タスクを割り当てるタスク割当部と
を備える演算制御装置であって、
前記タスク割当部は、
前記タスクを構成する複数のサブタスクを並列化処理の対象とすることが可能であるか否かを判断し、並列化処理が可能である場合には、所定の条件下で前記複数のサブタスクを前記複数のコアに割り当てるよう構成される ことを特徴とする演算制御装置。
- [請求項2] 前記タスクにパラメータを付与するパラメータ付与部を更に備え、
前記パラメータ付与部は、タスクが他のタスクに比べ優先的に処理されるべき度合を示す優先度をパラメータとして付与し、
前記並列化処理の対象とされた前記サブタスクの優先度を、並列化前の元のタスクの優先度とは異なる優先度に変更する、請求項1に記載の演算制御装置。
- [請求項3] 前記パラメータ付与部は、前記タスクが処理される周期を示すタスク制御周期、及び前記複数のタスクの依存関係を示すタスク依存関係をパラメータとして付与し、
前記タスク割当部は、前記タスク制御周期及び前記タスク依存関係に基づいて、前記複数のコアに対し前記タスクを割り当てる、請求項2に記載の演算制御装置。
- [請求項4] 前記タスクは、前記タスク制御周期とは無関係に発生するイベント駆動処理タスクを含み、
前記パラメータ付与部は、前記並列化処理の対象とされた前記サブタスクに対し、前記イベント駆動処理タスクよりも高い優先度をパラメータとして付与する、請求項3に記載の演算制御装置。
- [請求項5] 前記タスクは、タスク制御周期とは無関係に発生するイベント駆動処理タスクを含み、

前記タスク割当部は、前記タスク制御周期を有するタスクを前記複数のコアのうちの第1のコアに割り当て、前記イベント駆動処理タスクを前記第1のコアとは異なる第2のコアに割り当てる、請求項1に記載の演算制御装置。

[請求項6] 前記タスク割当部は、前記複数のサブタスクを、並列化前サブタスク、前記並列化前サブタスクの実行後に実行される並列化対象サブタスク、及び前記並列化対象サブタスクの実行後に実行される並列化後サブタスクに分割し、前記並列化対象サブタスクが前記並列化処理の対象とされるよう前記複数のコアにタスクを割り当てる、請求項1に記載の演算制御装置。

[請求項7] 前記並列化対象サブタスクは、前記並列化前サブタスク及び前記並列化後サブタスクよりも高い優先度を設定される、請求項6に記載の演算制御装置。

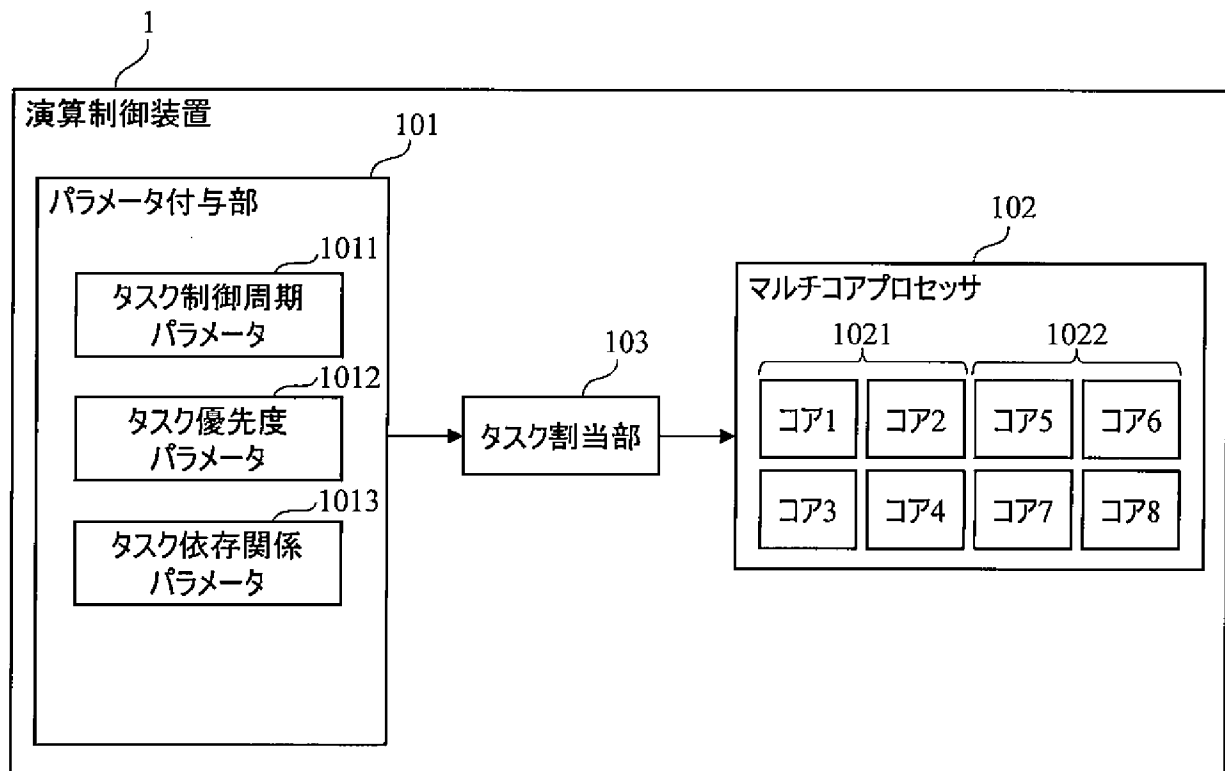
[請求項8] 前記並列化対象サブタスクは、分割前の元のタスクよりも高い優先度を設定される、請求項6又は7に記載の演算制御装置。

[請求項9] 前記タスクにパラメータを付与するパラメータ付与部を更に備え、
前記パラメータ付与部は、タスクの処理が完了すべき時刻を示すデッドラインをパラメータとして付与し、
前記並列化処理の対象とされた前記サブタスクのデッドラインを変更する、請求項1に記載の演算制御装置。

[請求項10] 前記プロセッサは、複数のコアの性能が互いに異なるヘテロジニアスプロセッサである、請求項1に記載の演算制御装置。

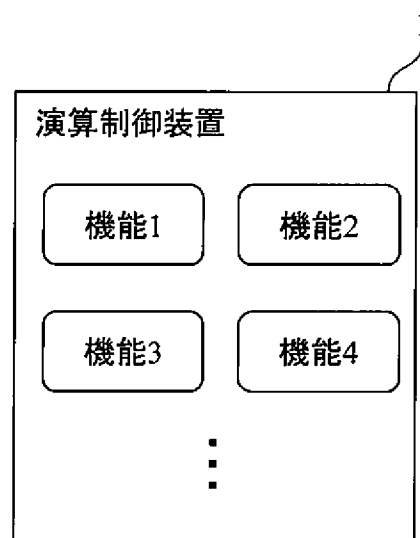
[図1]

図 1



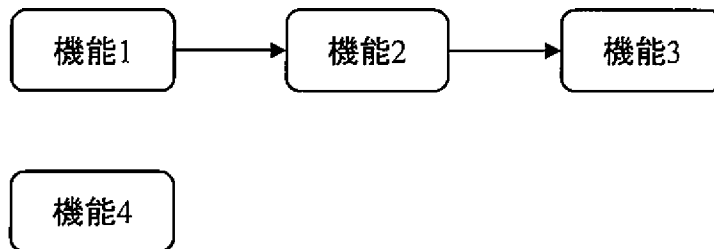
[図2]

図 2



[図3]

図 3



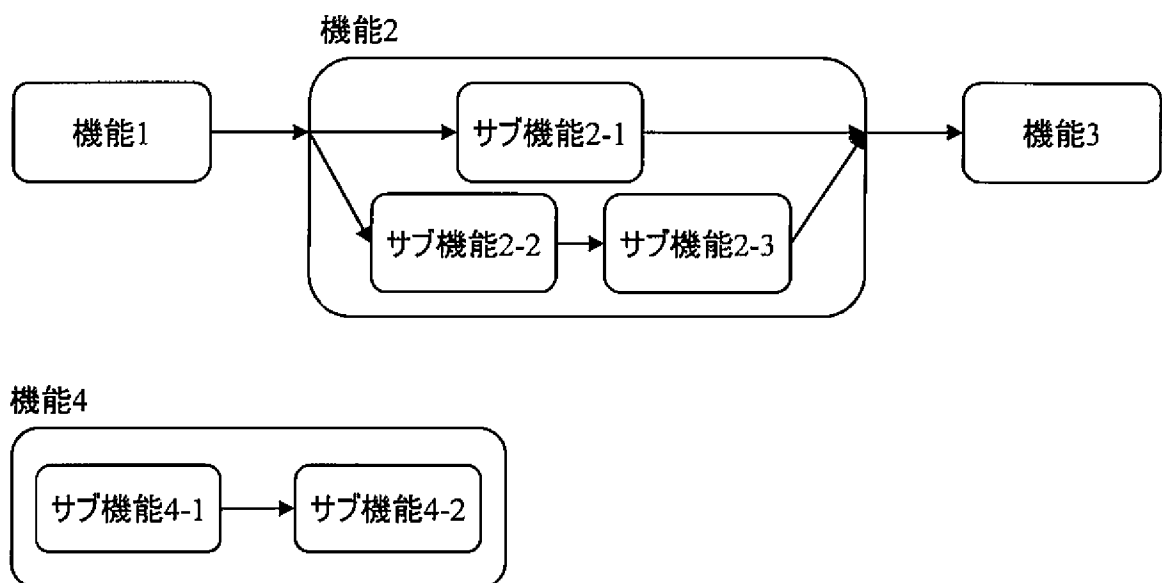
[図4]

図 4

機能	対応するタスク
1	A
2	B, C, D
3	E
4	F, G
...	

[図5]

図 5



[図6]

図 6

機能	対応するタスク
1	A
2-1	B
2-2	C
2-3	D
3	E
4-1	F
4-2	G

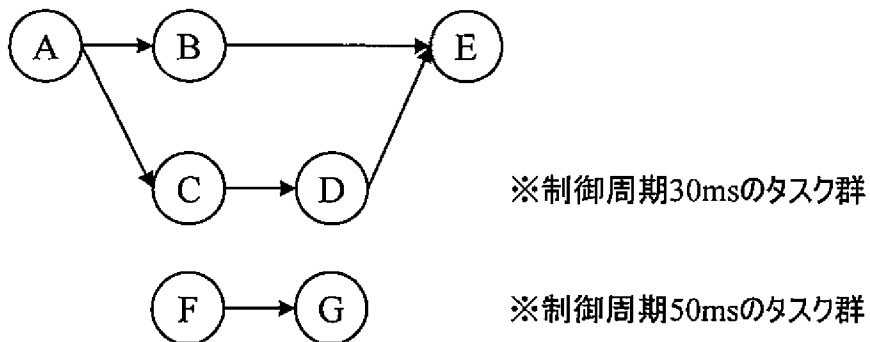
[図7]

図 7

タスク名	平均実行時間[ms] (高速コア)	並列化の可否	タスク制御周期 [ms]
A	5	不可	30
B	15	不可	30
C	5	不可	30
D	20	可	30
E	5	不可	30
F	30	可	50
G	10	不可	50
H	2		イベント駆動処理
...			

[図8]

図 8



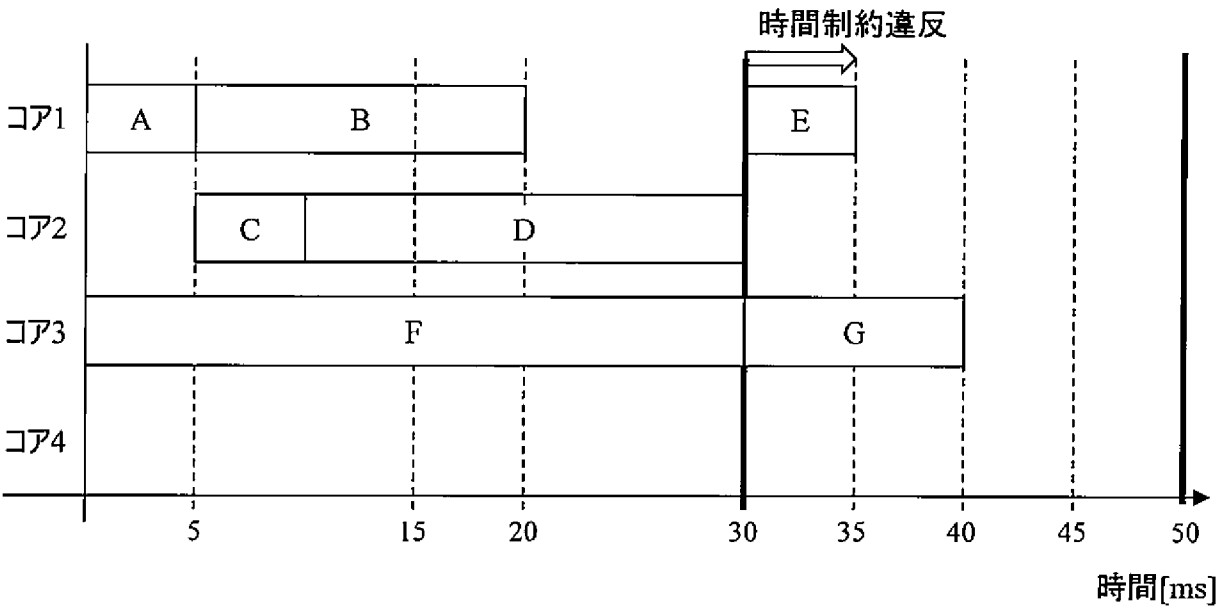
[図9]

図 9

タスク名	優先度
A	10
B	10
C	10
D	10
E	10
F	15
G	15
H	5
...	

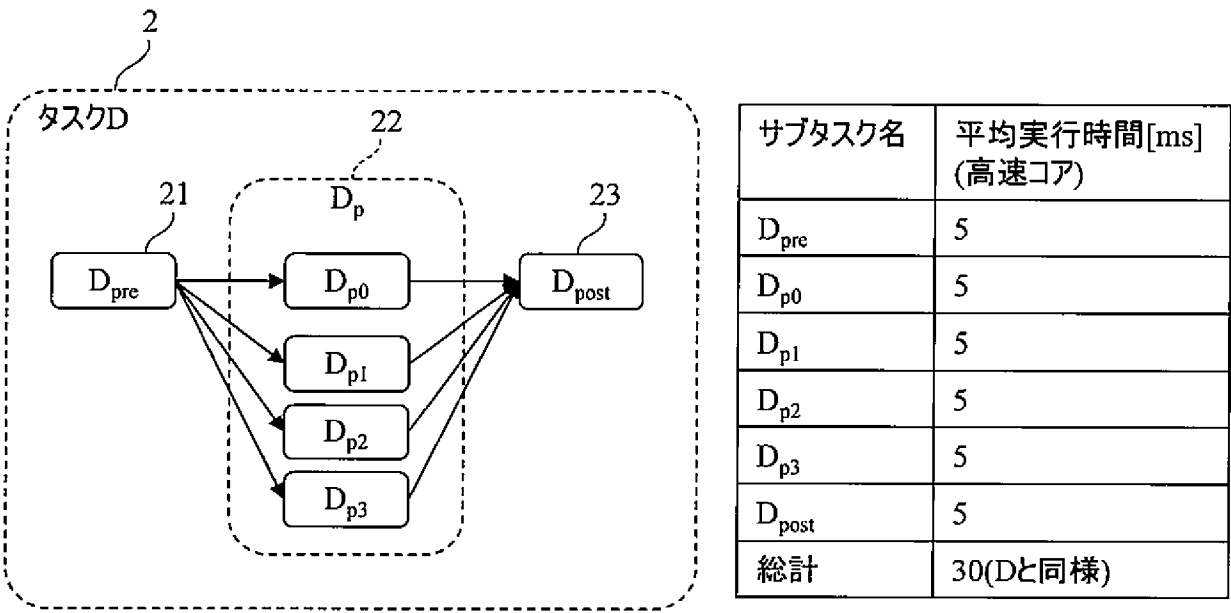
[図10]

図 1 0



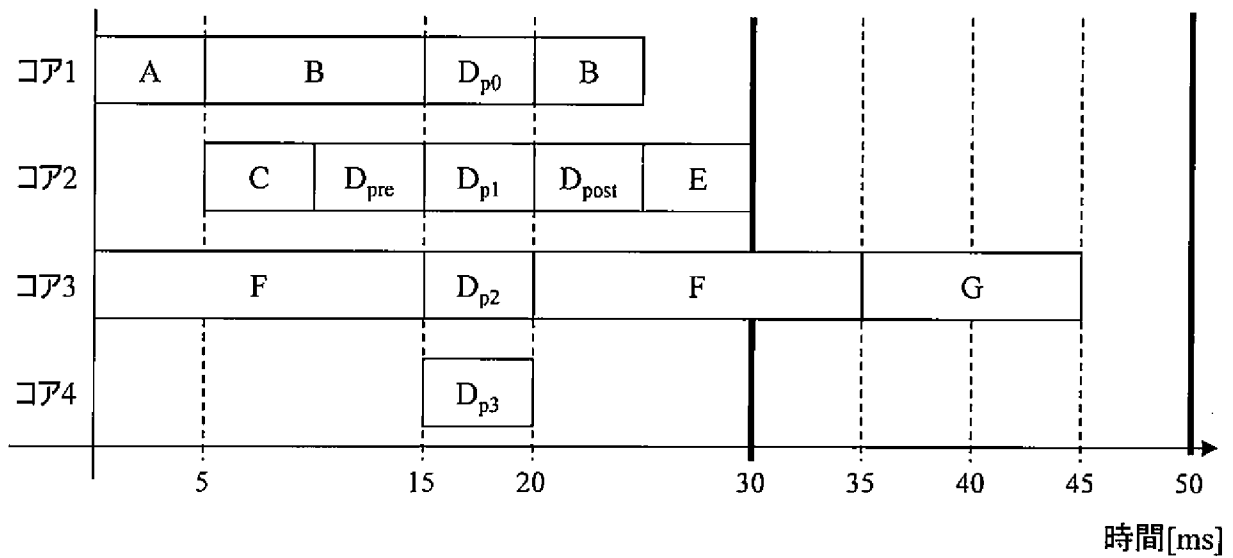
[図11]

図 1 1



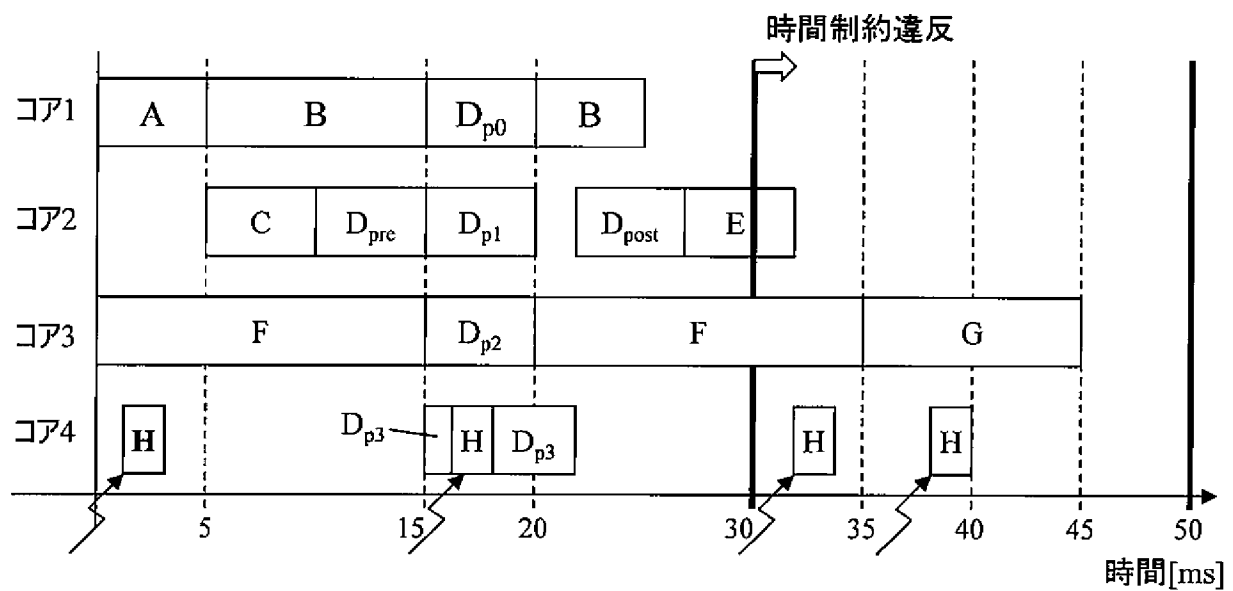
[図12]

図 1 2



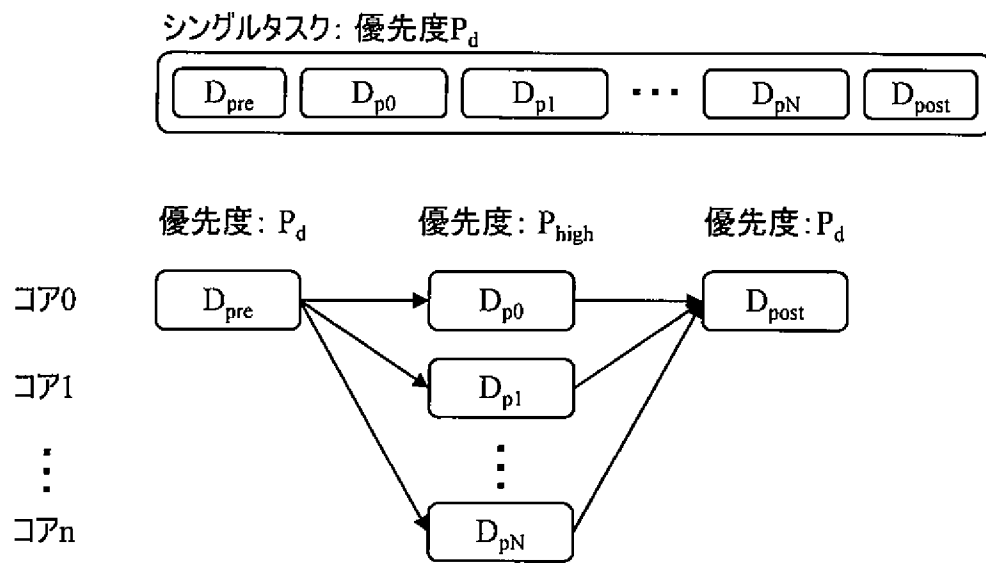
[図13]

図 1 3



[図14]

図 1 4



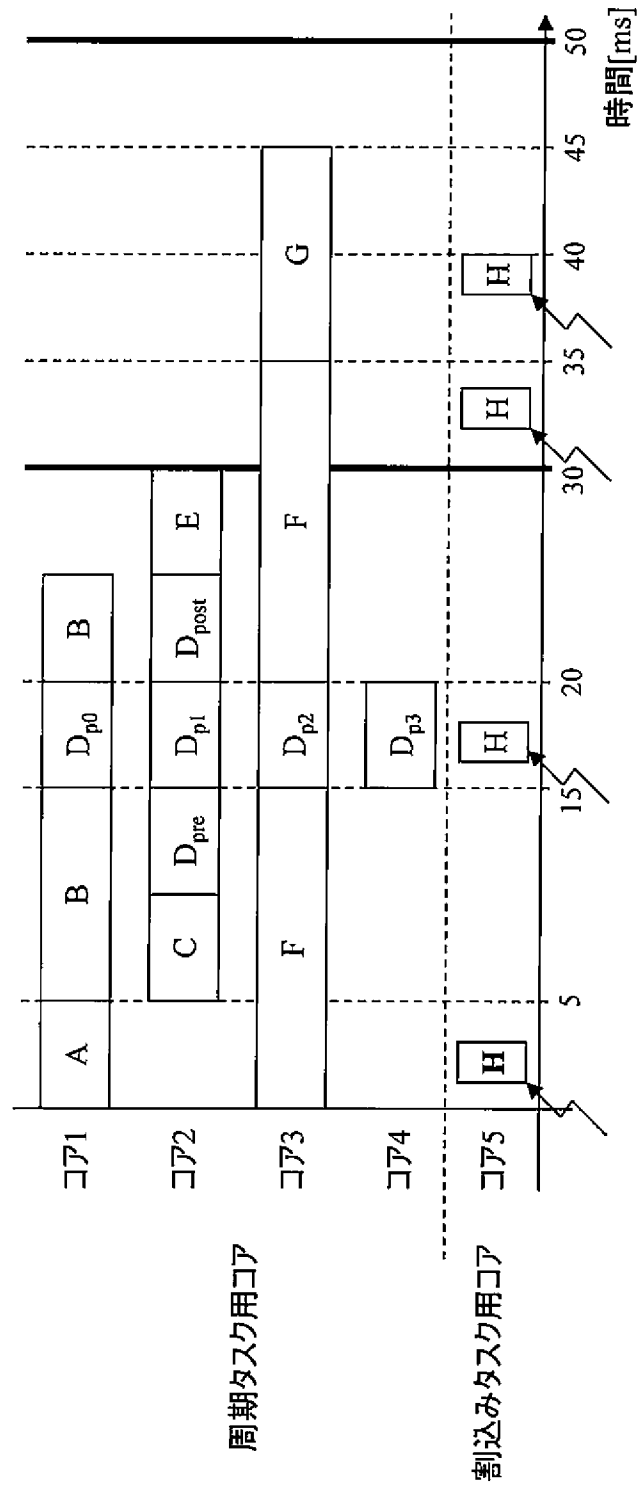
[図15]

図 1 5

タスク名	優先度
A	10
B	10
C	10
D_{pre}	10
D_{p0}	1
D_{p1}	1
D_{p2}	1
D_{p3}	1
D_{post}	10
E	10
F	15
G	15
H	5
...	

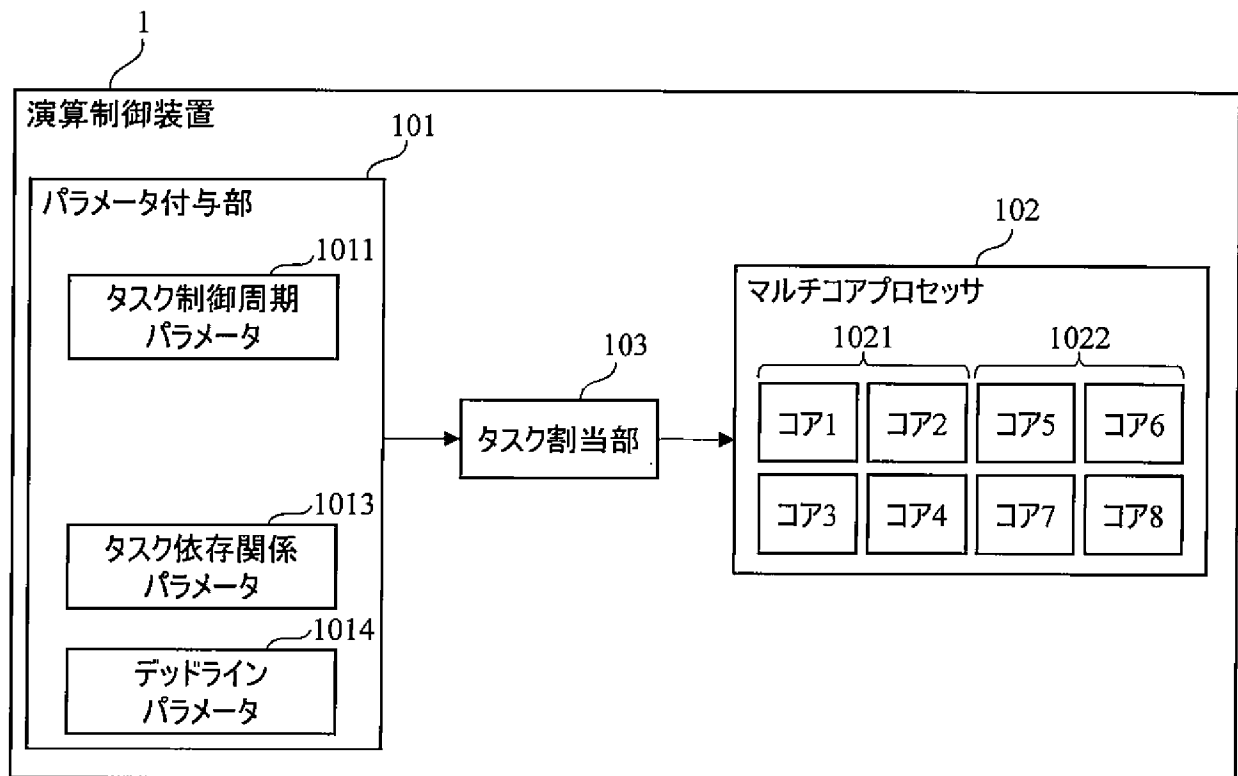
[図16]

図 16



[図17]

図 1 7



[図18]

図 1 8

タスク名	単体実行時間(平均)[ms] (高速コア)	並列化 可否	タスク制御周期 [ms]	絶対デッドライン [ms]
A	5	不可	30	30
B	15	不可	30	30
C	5	不可	30	30
D	20	可	30	30
E	5	不可	30	30
F	30	可	50	50
G	10	不可	50	50
H	2	不可	イベント駆動処理	-
...				

[図19]

図 19

タスク名	絶対デッドライン[ms]
A	30
B	30
C	30
D _{pre}	30
D _{p0}	20
D _{p1}	20
D _{p2}	20
D _{p3}	20
D _{post}	30
E	30
F	50
G	50
H	-
...	

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2019/046724

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/48 (2006.01) i; G06F 9/50 (2006.01) i
FI: G06F9/50 150E; G06F9/48 300B; G06F9/48 300F

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F9/48; G06F9/50

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Published examined utility model applications of Japan	1922-1996
Published unexamined utility model applications of Japan	1971-2020
Registered utility model specifications of Japan	1996-2020
Published registered utility model applications of Japan	1994-2020

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y A	JP 2009-151645 A (MITSUBISHI ELECTRIC CORP.) 09.07.2009 (2009-07-09) paragraphs [0015], [0023], [0025], [0033]-[0036], fig. 1-2	1, 5-6, 10 2-4, 7-9
Y A	JP 2015-225396 A (HITACHI AUTOMOTIVE SYSTEMS, LTD.) 14.12.2015 (2015-12-14) paragraph [0018], fig. 2	1, 5-6, 10 2-4, 7-9
Y	JP 2014-160367 A (MITSUBISHI ELECTRIC CORP.) 04.09.2014 (2014-09-04) paragraph [0033]	5
Y	JP 2016-91137 A (KONICA MINOLTA, INC.) 23.05.2016 (2016-05-23) paragraph [0006]	10
A	JP 2000-99352 A (FUJI XEROX CO., LTD.) 07.04.2000 (2000-04-07) paragraph [0066]	2
A	WO 2014/068950 A1 (NEC CORP.) 08.05.2014 (2014-05- 08) paragraph [0082]	9

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"I" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search
13 March 2020 (13.03.2020)

Date of mailing of the international search report
24 March 2020 (24.03.2020)

Name and mailing address of the ISA/
Japan Patent Office
3-4-3, Kasumigaseki, Chiyoda-ku,
Tokyo 100-8915, Japan

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/JP2019/046724

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
JP 2009-151645 A	09 Jul. 2009	(Family: none)	
JP 2015-225396 A	14 Dec. 2015	US 2017/0039091 A1 paragraph [0031], fig. 2 WO 2015/182295 A1 EP 3151116 A1 CN 106462452 A	
JP 2014-160367 A	04 Sep. 2014	(Family: none)	
JP 2016-91137 A	23 May 2016	US 2016/0127604 A1 paragraph [0008] EP 3021567 A1 CN 105574805 A	
JP 2000-99352 A	07 Apr. 2000	(Family: none)	
WO 2014/068950 A1	08 May 2014	US 2015/0254105 A1 paragraph [0118]	

A. 発明の属する分野の分類（国際特許分類（IPC）） G06F 9/48(2006.01)i; G06F 9/50(2006.01)i FI: G06F9/50 150E; G06F9/48 300B; G06F9/48 300F		
B. 調査を行った分野		
調査を行った最小限資料（国際特許分類（IPC）） G06F9/48; G06F9/50		
最小限資料以外の資料で調査を行った分野に含まれるもの 日本国実用新案公報 1922-1996年 日本国公開実用新案公報 1971-2020年 日本国実用新案登録公報 1996-2020年 日本国登録実用新案公報 1994-2020年		
国際調査で使用した電子データベース（データベースの名称、調査に使用した用語）		
C. 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
Y	JP 2009-151645 A（三菱電機株式会社）09.07.2009（2009-07-09） 段落[0015], [0023], [0025], [0033]-[0036], 図1-2	1, 5-6, 10
A		2-4, 7-9
Y	JP 2015-225396 A（日立オートモティブシステムズ株式会社）14.12.2015（2015-12-14） 段落[0018], 図2	1, 5-6, 10
A		2-4, 7-9
Y	JP 2014-160367 A（三菱電機株式会社）04.09.2014（2014-09-04） 段落[0033]	5
Y	JP 2016-91137 A（コニカミノルタ株式会社）23.05.2016（2016-05-23） 段落[0006]	10
A	JP 2000-99352 A（富士ゼロックス株式会社）07.04.2000（2000-04-07） 段落[0066]	2
A	WO 2014/068950 A1（日本電気株式会社）08.05.2014（2014-05-08） 段落[0082]	9
☐ C欄の続きにも文献が列挙されている。 ☒ パテントファミリーに関する別紙を参照。		
* 引用文献のカテゴリー “A” 特に関連のある文献ではなく、一般的な技術水準を示すもの “E” 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの “L” 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す） “O” 口頭による開示、使用、展示等に言及する文献 “P” 国際出願日前で、かつ優先権の主張の基礎となる出願の日の後に公表された文献 “T” 国際出願日又は優先日後に公表された文献であって出願と抵触するものではなく、発明の原理又は理論の理解のために引用するもの “X” 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの “Y” 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの “&” 同一パテントファミリー文献		
国際調査を完了した日 13.03.2020		国際調査報告の発送日 24.03.2020
名称及びあて先 日本国特許庁(ISA/JP) 〒100-8915 日本国 東京都千代田区霞が関三丁目4番3号		権限のある職員（特許庁審査官） 加藤 優一 5B 6296 電話番号 03-3581-1101 内線 3545

国際調査報告
 パテントファミリーに関する情報

国際出願番号

PCT/JP2019/046724

引用文献			公表日	パテントファミリー文献	公表日
JP	2009-151645	A	09.07.2009	(ファミリーなし)	
JP	2015-225396	A	14.12.2015	US 2017/0039091	A1
				段落[0031], 図2	
				WO 2015/182295	A1
				EP 3151116	A1
				CN 106462452	A
JP	2014-160367	A	04.09.2014	(ファミリーなし)	
JP	2016-91137	A	23.05.2016	US 2016/0127604	A1
				段落[0008]	
				EP 3021567	A1
				CN 105574805	A
JP	2000-99352	A	07.04.2000	(ファミリーなし)	
WO	2014/068950	A1	08.05.2014	US 2015/0254105	A1
				段落[0118]	