

(19)日本国特許庁(JP)

(12)公表特許公報(A)

(11)公表番号

特表2022-507707
(P2022-507707A)

(43)公表日 令和4年1月18日(2022.1.18)

(51)国際特許分類	F I	テーマコード(参考)
G 0 6 F 9/38 (2006.01)	G 0 6 F 9/38 3 7 0 C	5 B 0 1 3
G 0 6 F 13/28 (2006.01)	G 0 6 F 13/28 3 1 0 C	5 B 0 6 1
G 0 6 F 15/78 (2006.01)	G 0 6 F 15/78 5 6 0	5 B 0 6 2

審査請求 未請求 予備審査請求 未請求 (全35頁)

(21)出願番号	特願2021-527135(P2021-527135)	(71)出願人	591025439 ザイリンクス インコーポレイテッド X I L I N X I N C O R P O R A T E D アメリカ合衆国 カリフォルニア州 9 5 1 2 4 - 3 4 0 0 サン ホセ ロジック ドライブ 2 1 0 0
(86)(22)出願日	令和1年11月11日(2019.11.11)	(74)代理人	110002077 園田・小林特許業務法人
(85)翻訳文提出日	令和3年7月14日(2021.7.14)	(72)発明者	ソー, ソーレン アメリカ合衆国 カリフォルニア 9 5 1 2 4 , サン ノゼ, ロジック ドライブ 2 1 0 0
(86)国際出願番号	PCT/US2019/060744	(72)発明者	タルワラ, イドリス アイ. アメリカ合衆国 カリフォルニア 9 5 1 2 4 , サン ノゼ, ロジック ドライブ
(87)国際公開番号	WO2020/106482		
(87)国際公開日	令和2年5月28日(2020.5.28)		
(31)優先権主張番号	16/194,911		
(32)優先日	平成30年11月19日(2018.11.19)		
(33)優先権主張国・地域又は機関	米国(US)		
(81)指定国・地域	AP(BW,GH,GM,KE,LR,LS,MW,MZ,NA ,RW,SD,SL,ST,SZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,RU,TJ,TM),EP(AL,A T,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR ,GB,GR,HR,HU,IE,IS,IT,LT,LU,LV,MC, 最終頁に続く		最終頁に続く

(54)【発明の名称】 集積回路中の算出ユニットをプログラムおよび制御すること

(57)【要約】

集積回路(150)(IC)が、IC(150)の複数の算出ユニット(180)による実行のためのホストプロセッサ(105)からのコマンドに対応する複数のスロットを有するコマンド待ち行列(162)と、コマンド待ち行列(162)中の複数のスロットに対応する複数のロケーションを有するコマンド要求レジスタ(164)とを含むことができる。コマンド要求レジスタ(164)は、コマンド待ち行列(162)内に記憶された新しいコマンド(208)を指示する割り込みを生成するように設定される。IC(150)は、コマンド要求レジスタ(164)からの割り込みに応答して、新しいコマンド(208)を実行するための、複数の算出ユニット(180)からの、アイドルである選択された算出ユニット(180-1)を決定するように設定されたコントローラ(168)を含むことができる。IC(150)は、新しいコマンド(208)を利用可能な算出ユニットに提供するように設定された算出ユニットダイレクタメモリアクセス回路をも含むことができる。

【選択図】図1

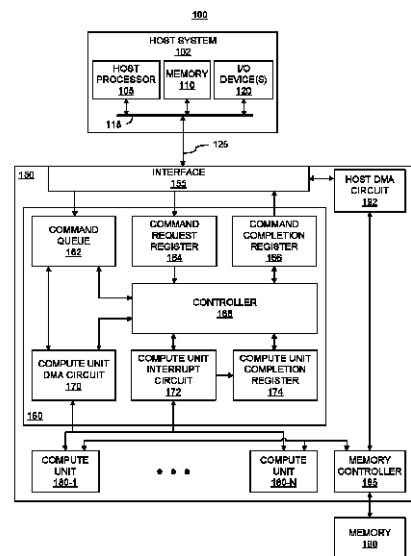


FIG. 1

【特許請求の範囲】**【請求項 1】**

集積回路であって、

前記集積回路の複数の算出ユニットによる実行のためのホストプロセッサからのコマンドを記憶するように設定された複数のスロットを有するコマンド待ち行列と、

前記コマンド待ち行列中の前記複数のスロットに対応する複数のロケーションを有するコマンド要求レジスタであって、前記コマンド要求レジスタが、前記コマンド待ち行列内に記憶された新しいコマンドを指示する割込みを生成するように設定された、コマンド要求レジスタと、

前記コマンド要求レジスタからの前記割込みに応答して、前記新しいコマンドを実行するための、前記複数の算出ユニットからの、アイドルである選択された算出ユニットを決定するように設定されたコントローラと、

前記新しいコマンドを前記選択された算出ユニットに提供するように設定された算出ユニットダイレクトメモリアクセス回路と

を備える、集積回路。

10

【請求項 2】

前記新しいコマンドが、前記新しいコマンドを実行することが可能な前記複数の算出ユニットのうちの 1 つまたは複数を指定する第 1 のリストを含む、請求項 1 に記載の集積回路。

【請求項 3】

前記コントローラが、前記複数の算出ユニットのうちのどれがアイドルであるかを指定する第 2 のリストを維持することと、前記第 1 のリストを前記第 2 のリストと比較することによって、前記新しいコマンドを実行するための前記選択された算出ユニットを決定することを行うように設定された、請求項 2 に記載の集積回路。

20

【請求項 4】

前記コントローラが、前記選択された算出ユニットのためのベースアドレスを指定するために前記コマンド待ち行列中の前記新しいコマンドを更新するように設定された、請求項 2 に記載の集積回路。

【請求項 5】

前記コントローラが、前記選択された算出ユニットの前記ベースアドレスを前記第 1 のリストに上書きするように設定された、請求項 4 に記載の集積回路。

30

【請求項 6】

前記コントローラが、前記新しいコマンドを記憶する、前記コマンド待ち行列中の前記複数のスロットのうちの選択されたスロットの参照を、前記算出ユニットダイレクトメモリアクセス回路に提供するように設定された、請求項 4 に記載の集積回路。

【請求項 7】

前記算出ユニットダイレクトメモリアクセス回路が、前記新しいコマンドからの前記ベースアドレスを使用して、前記選択されたスロット中の前記新しいコマンドを前記選択された算出ユニットに提供するように設定された、請求項 6 に記載の集積回路。

【請求項 8】

前記コマンド要求レジスタが読取り時クリアレジスタである、請求項 1 に記載の集積回路。

40

【請求項 9】

前記算出ユニットダイレクトメモリアクセス回路が、前記新しいコマンドを提供したことに応答して、前記選択された算出ユニットの動作を開始するようにさらに設定された、請求項 1 に記載の集積回路。

【請求項 10】

集積回路であって、

複数の算出ユニットのうちの選択された算出ユニットから、前記選択された算出ユニットが処理し終わったことを指示する割込みを受信したことに応答して、前記選択された算出

50

ユニットが処理し終わったという信号を出すように設定された算出ユニット割込み回路と、
 前記算出ユニット割込み回路からの前記信号に応答して割込みを生成するように設定された算出ユニット完了レジスタと、
 前記算出ユニット完了レジスタからの前記割込みに応答して、ホストプロセッサからオフロードされた複数のコマンドのうちの前記選択されたコマンドが、前記選択された算出ユニットによって実行されたと決定し、前記選択されたコマンドを出力するように設定されたコントローラと、
 前記複数のコマンドのうちの前記選択されたコマンドが処理し終わったことを指示する、前記ホストプロセッサへの割込みを生成するように設定されたコマンド完了レジスタとを備える、集積回路。

10

【請求項 1 1】

前記算出ユニット完了レジスタが、前記複数の算出ユニットに対応する複数のロケーションを含み、前記算出ユニット割込み回路は、前記選択された算出ユニットに対応する、前記複数のロケーションのうちの前記選択されたロケーションに書き込むことによって、前記選択された算出ユニットが処理し終わったことを前記算出ユニット完了レジスタに通知する、請求項 1 0 に記載の集積回路。

【請求項 1 2】

前記コントローラが、前記複数のコマンドと、前記複数のコマンドを実行する、前記複数の算出ユニットのうちの前記対応する算出ユニットとのリストを維持するように設定された、請求項 1 0 に記載の集積回路。

20

【請求項 1 3】

前記算出ユニット完了レジスタまたは前記コマンド完了レジスタが読取り時クリアレジスタである、請求項 1 0 に記載の集積回路。

【請求項 1 4】

前記算出ユニット割込み回路が、前記ホストプロセッサからのさらなるコマンドを実行するために、前記選択された算出ユニットをリアームするように設定された、請求項 1 0 に記載の集積回路。

【請求項 1 5】

コマンド待ち行列内の新しいコマンドの利用可能性を指示する、ホストプロセッサからの通知に応答して、コントローラへの割込みを生成することと、
 前記コントローラが、前記割込みに応答して、前記新しいコマンドを実行するための、複数の算出ユニットからの、アイドルである選択された算出ユニットを決定することと、
 前記コントローラが、前記選択された算出ユニットと前記コマンド待ち行列中の前記新しいコマンドとを算出ユニットダイレクトメモリアクセス回路に通知することと、
 前記算出ユニットダイレクトメモリアクセス回路を使用して、前記新しいコマンドを前記選択された算出ユニットに提供することと
 含む、方法。

30

【発明の詳細な説明】

【技術分野】

40

【0 0 0 1】

本開示は、集積回路（IC）に関し、より詳細には、ハードウェアアクセラレーションのために IC 内の算出ユニットをプログラムおよび制御することに関する。

【背景技術】

【0 0 0 2】

異種コンピューティングプラットフォーム（HCP: heterogeneous computing platform）は、ホストプロセッサと 1 つまたは複数の他の異なる処理デバイスとを含むデータ処理システムのタイプを指す。ホストプロセッサは、一般的に、中央処理ユニット（CPU）として実装される。ホストプロセッサは、インターフェース回路を通して他の処理デバイスに結合される。他の処理デバイスは、アーキテクチ

50

ヤ上、ホストプロセッサとは異なる。さらに、処理デバイスは、ホストプロセッサからオフロードされた、動作を実施すること、たとえば、アプリケーションを実行することと、動作の結果をホストプロセッサにとって利用可能にすることとを行うことが可能である。

【0003】

いくつかの場合には、処理デバイスは、プログラムコードを実行するように適応される。そのような処理デバイスは、一般的に、ホストプロセッサとは異なる命令セットアーキテクチャを有する。これらの他の処理デバイスの例は、限定はしないが、（1つまたは複数の）グラフィックス処理ユニット（GPU）、（1つまたは複数の）デジタル信号プロセッサ（DSP）などを含む。

【0004】

他の場合には、ホストプロセッサからオフロードされた動作を実施する処理デバイスは、プログラムコードをハードウェアアクセラレートする（hardware accelerate）ように適応されたデバイスを含む。これらの処理デバイスは、オフロードされた動作をハードウェアとして実装する回路を含む。回路は、プログラムコードの実行と機能的に等価である様式で動作する（たとえば、オフロードされた動作）。ハードウェアアクセラレーションが可能な処理デバイスの例は、フィールドプログラマブルゲートアレイ（FPGA）、部分的にプログラム可能な集積回路（IC）、特定用途向けIC（ASIC）など、プログラマブルICを含む。明らかに、HCPは、1つまたは複数がプログラムコードを実行するように適応され、1つまたは複数の他のものがプログラムコードをハードウェアアクセラレートするように適応された、処理デバイスの組合せを含み得る。

【0005】

プログラムコードをハードウェアで実装することは、より速い動作、低減された電力消費、冗長性など、様々な利益を提供し得る。アプリケーションをハードウェアとして実装することが、より速い動作を含むことも含まないこともある利益を提供するという事実にもかかわらず、プログラムコードおよび/またはアプリケーションの、ハードウェアでの実装は、一般的に、「ハードウェアアクセラレーション」と呼ばれる。

【0006】

いずれの場合にも、ホストプロセッサは、処理デバイスに動作をオフロードし、処理デバイスから結果を取り出すことの責任を負う。処理デバイスとの間で必要なデータを効率的に移動するホストプロセッサの能力は、HCPの全体的効率および性能に著しく影響を及ぼすことがある。

【発明の概要】

【0007】

1つまたは複数の実施形態は、集積回路（IC）を対象とする。本ICは、本ICの複数の算出ユニットによる実行のためのホストプロセッサからのコマンドを記憶するように設定された複数のスロットを有するコマンド待ち行列と、コマンド待ち行列中の複数のスロットに対応する複数のロケーションを有するコマンド要求レジスタとを含む。コマンド要求レジスタは、コマンド待ち行列内に記憶された新しいコマンドを指示する割込みを生成するように設定される。本ICは、コマンド要求レジスタからの割込みに応答して、新しいコマンドを実行するための、複数の算出ユニットからの、アイドルである選択された算出ユニットを決定するように設定されたコントローラと、新しいコマンドを選択された算出ユニットに提供するように設定された算出ユニットダイレクトメモリアクセス回路とを含む。

【0008】

いくつかの実施形態では、新しいコマンドは、新しいコマンドを実行することが可能な複数の算出ユニットのうちの1つまたは複数を選択する第1のリストを含み得る。

【0009】

いくつかの実施形態では、コントローラは、複数の算出ユニットのうちのどれがアイドルであるかを指定する第2のリストを維持することと、第1のリストを第2のリストと比較することによって、新しいコマンドを実行するための選択された算出ユニットを決定する

10

20

30

40

50

こととを行うように設定され得る。

【0010】

いくつかの実施形態では、コントローラは、選択された算出ユニットのためのベースアドレスを指定するためにコマンド待ち行列中の新しいコマンドを更新するように設定され得る。

【0011】

いくつかの実施形態では、コントローラは、選択された算出ユニットのベースアドレスを第1のリストに上書きするように設定され得る。

【0012】

いくつかの実施形態では、コントローラは、新しいコマンドを記憶する、コマンド待ち行列中の複数のスロットのうちの選択されたスロットの参照を、算出ユニットダイレクトメモリアクセス回路に提供するように設定され得る。

10

【0013】

いくつかの実施形態では、算出ユニットダイレクトメモリアクセス回路は、新しいコマンドからのベースアドレスを使用して、選択されたスロット中の新しいコマンドを選択された算出ユニットに提供するように設定され得る。

【0014】

いくつかの実施形態では、コマンド要求レジスタは読取り時クリアレジスタ(clear-on-read register)であり得る。

【0015】

いくつかの実施形態では、算出ユニットダイレクトメモリアクセス回路は、新しいコマンドを提供したことに応答して、選択された算出ユニットの動作を開始するようにさらに設定され得る。

20

【0016】

1つまたは複数の実施形態は、ICを対象とする。本ICは、複数の算出ユニットのうちの選択された算出ユニットから、選択された算出ユニットが処理し終わったことを指示する割込みを受信したことに応答して、選択された算出ユニットが処理し終わったという信号を出すように設定された算出ユニット割込み回路と、算出ユニット割込み回路からの信号に応答して割込みを生成するように設定された算出ユニット完了レジスタとを含む。本ICは、算出ユニット完了レジスタからの割込みに応答して、ホストプロセッサからオフロードされた複数のコマンドのうちの選択されたコマンドが、選択された算出ユニットによって実行されたと決定し、選択されたコマンドを出力するように設定されたコントローラと、複数のコマンドのうちの選択されたコマンドが処理し終わったことを指示する、ホストプロセッサへの割込みを生成するように設定されたコマンド完了レジスタとを含む。

30

【0017】

いくつかの実施形態では、算出ユニット完了レジスタは、複数の算出ユニットに対応する複数のロケーションを含み得る。算出ユニット割込み回路は、選択された算出ユニットに対応する、複数のロケーションのうちの選択されたロケーションに書き込むことによって、選択された算出ユニットが処理し終わったことを算出ユニット完了レジスタに通知し得る。

40

【0018】

いくつかの実施形態では、コントローラは、複数のコマンドと、複数のコマンドを実行する、複数の算出ユニットのうちの対応する算出ユニットとのリストを維持するように設定され得る。

【0019】

いくつかの実施形態では、算出ユニット完了レジスタは読取り時クリアレジスタであり得る。

【0020】

いくつかの実施形態では、コマンド完了レジスタは読取り時クリアレジスタであり得る。

【0021】

50

いくつかの実施形態では、算出ユニット割込み回路は、ホストプロセッサからのさらなるコマンドを実行するために、選択された算出ユニットをリアームする (r e a r m) ように設定され得る。

【 0 0 2 2 】

1つまたは複数の実施形態は、方法を対象とする。本方法は、コマンド待ち行列内の新しいコマンドの利用可能性を指示する、ホストプロセッサからの通知に回答して、コントローラへの割込みを生成することと、コントローラが、割込みに回答して、新しいコマンドを実行するための、複数の算出ユニットからの、アイドルである選択された算出ユニットを決定することと、コントローラが、選択された算出ユニットとコマンド待ち行列中の新しいコマンドとを算出ユニットダイレクトメモリアクセス回路に通知することとを含む。本方法は、算出ユニットダイレクトメモリアクセス回路を使用して、新しいコマンドを選択された算出ユニットに提供することをも含む。

10

【 0 0 2 3 】

いくつかの実施形態では、本方法は、実行の完了を指示する、選択された算出ユニットからの割込みを受信したことに応答して、選択された算出ユニットが実行を完了したことを指示する、コントローラへの割込みを生成することをさらに含み得る。本方法は、コントローラを使用して、新しいコマンドが、選択された算出ユニットに割り当てられたと決定することと、新しいコマンドが実行を完了したことを指示する、ホストプロセッサへの割込みを生成することをも含み得る。

【 0 0 2 4 】

いくつかの実施形態では、選択された算出ユニットとコマンド待ち行列中の新しいコマンドとを算出ユニットダイレクトメモリアクセス回路に通知することは、新しいコマンドを記憶する、コマンド待ち行列中の複数のスロットのうちの選択されたスロットに、選択された算出ユニットのベースアドレスを書き込むことと、複数のスロットのうちの選択されたスロットの参照を、算出ユニットダイレクトメモリアクセス回路に提供することとを含み得る。

20

【 0 0 2 5 】

いくつかの実施形態では、複数の算出ユニットからの選択された算出ユニットを決定することは、新しいコマンド内から、新しいコマンドを実行することが可能な複数の算出ユニットのうちの1つまたは複数を選択する第1のリストを読み取ることと、第1のリストを、アイドルである、複数の算出ユニットのうちの算出ユニットを選択する第2のリストと比較して、共通算出ユニットを選択された算出ユニットとして決定することとを含み得る。

30

【 0 0 2 6 】

いくつかの実施形態では、選択された算出ユニットとコマンド待ち行列中の新しいコマンドとを算出ユニットダイレクトメモリアクセス回路に通知することは、選択された算出ユニットのベースアドレスを新しいコマンドの第1のリストに上書きすることと、新しいコマンドが、コマンド待ち行列中の複数のスロットのうちの選択されたスロットに記憶される、第1のリストに上書きすることと、選択されたスロットの参照を算出ユニットダイレクトメモリアクセス回路に提供することとを含み得る。

40

【 0 0 2 7 】

本発明の概要セクションは、いくつかの概念を導入するために提供されるにすぎず、請求される主題の重要な、または本質的な特徴を識別するために提供されるものではない。本発明の構成の他の特徴は、添付の図面および以下の発明を実施するための形態から明らかになる。

【 0 0 2 8 】

本発明の構成は、添付の図面において例として示される。しかしながら、図面は、本発明の構成を、図示される特定の実装形態のみに限定するものと解釈されるべきではない。様々な態様および利点が、以下の発明を実施するための形態を検討し、図面を参照すると明らかになる。

50

【図面の簡単な説明】**【0029】**

【図1】本開示内で説明される1つまたは複数の実施形態による、異種コンピューティングプラットフォームの一例を示す図である。

【図2】本開示内で説明される1つまたは複数の実施形態による、スケジューリングシステムによって実施される例示的な動作を示す図である。

【図3A】図3A～図3Bは、まとめて、本開示内で説明される1つまたは複数の実施形態による、HCPのための動作の例示的な方法を示す図である。

【図3B】図3A～図3Bは、まとめて、本開示内で説明される1つまたは複数の実施形態による、HCPのための動作の例示的な方法を示す図である。

【図4】集積回路(IC)のための例示的なアーキテクチャを示す図である。

【発明を実施するための形態】**【0030】**

本開示は、新規の特徴を定義する特許請求の範囲で締めくくるが、本開示内で説明される様々な特徴は、図面とともにその説明を考慮することにより、より良く理解されると考えられる。本明細書で説明される(1つまたは複数の)プロセス、(1つまたは複数の)機械、(1つまたは複数の)製造物およびその任意の変形形態は、例示のために提供される。本開示内で説明される特定の構造的および機能的詳細は、限定するものとして解釈されるべきではなく、単に、特許請求の範囲のための基礎として、およびほぼすべての適切に詳細な構造において説明される特徴を様々に採用するように当業者に教示するための代表的基礎として解釈されるべきである。さらに、本開示内で使用される用語および句は、限定するものではなく、むしろ、説明される特徴の理解可能な説明を提供するものである。

【0031】

本開示は、集積回路(IC)に関し、より詳細には、ハードウェアアクセラレーションのためにIC内の算出ユニット(CU)をプログラムおよび制御することに関する。本開示内で説明される本発明の構成によれば、IC内のCUの動作を監視、追跡、および制御することが可能であるスケジューリングシステムが説明される。CUは、異種コンピューティングプラットフォーム(HCP)のホストプロセッサからオフロードされた動作を実施することを担当する、IC内のハードウェアブロックまたは回路である。スケジューリングシステムは、CUを含む同じIC内に実装され、コマンドの形態で提供された、動作のホストプロセッサからCUへのオフローディングを支援する。

【0032】

従来のHCPでは、スケジューリングは、ホストプロセッサによって実行されるソフトウェアプロセスとして実装される。ホストプロセッサは、動作を実施するためにCUがいつ利用可能になるかを決定するために、ICの個々のCUを継続的にポーリングするというタスクを与えられる。CUのポーリングは、かなりの量のホストプロセッサ時間を消費する。さらに、スケジューリングプロセスはホストプロセッサによって実行されるので、ホストプロセッサは、ICにコマンドを送る前に、CUが利用可能になるまで待たなければならない。ホストプロセッサは、コマンドを実行するためにCUが利用可能になったと決定したことに応答してのみ、コマンドを送る。したがって、CUが利用可能になる時間と、CUがコマンドに対する実行を始めることが可能になる時間との間に、遅延が生じる。この遅延は、CUによって必要とされるデータを含むコマンドを、ホストプロセッサからCUを含むICに転送するのに必要な時間を含む。

【0033】

本明細書で説明される本発明の構成によれば、スケジューリングシステムは、CUを含む同じIC中で実装される。スケジューリングシステムは、IC内のCUの利用可能性を追跡することが可能である。スケジューリングシステムは、ホストプロセッサから送られたコマンド(たとえば、動作)を受信することも可能である。スケジューリングシステムは、CUが(1つまたは複数の)コマンドを実行するために利用可能になるような時間まで、コマンド待ち行列にコマンドを記憶することが可能である。したがって、ホストプロセ

10

20

30

40

50

ッサは、利用可能なCUについて継続的にポーリングする必要なしに、および/またはICにコマンドを送る前にCUが利用可能になるのを待つ必要なしに、コマンドを送り、他のタスクを続けることが可能である。

【0034】

コマンドがスケジューリングシステム内で待ち行列に入れられ、CUが利用可能になると、CUへのコマンドの転送は、ホストプロセッサ関与を必要としない。スケジューリングシステムは、CUにコマンドを転送する。スケジューリングシステムとCUとは同じIC中にあるので、CUにコマンドを転送するための時間は比較的小さい。したがって、CUが利用可能になる時間と、CUがコマンドの実行を始める時間との間の遅延が低減され、それにより、HCPの速度を増加させ、レイテンシを低減する。

10

【0035】

スケジューリングシステムは、さらに、コマンドがいつ実行し終わったかを決定することが可能である。コマンドが実行し終わったと決定したことに応答して、スケジューリングシステムは、ホストプロセッサに通知することが可能である。たとえば、スケジューリングシステムは、コマンドが実行を終えたと決定したことに応答して、スケジューリングシステム内に含まれるコマンド完了レジスタに書き込むことが可能である。コマンド完了レジスタは、コマンドのステータスと、実行を終えた特定のコマンドとを確認するために、ホストプロセッサによって読み取られ得る。コマンド完了レジスタの使用は、ホストプロセッサが、ICのCUの各々を個々に管理することとは対照的に「コマンド」レベルで動作することを可能にする。

20

【0036】

図を参照しながら、本発明の構成のさらなる態様が以下でより詳細に説明される。例示を単純および明快にするために、図に示されている要素は、必ずしも一定の縮尺で描かれているとは限らない。たとえば、要素のうちのいくつかの寸法は、明快のために、他の要素に対して誇張され得る。さらに、適切と見なされる場合、対応する、類似する、または同様の特徴を指示するために、参照番号が図の間で繰り返される。

【0037】

図1は、本開示内で説明される1つまたは複数の実施形態による、HCP100の一例を示す。図1の例では、HCP100は、ホストシステム102と、ハードウェアアクセラレーションデバイスとして使用されるIC150とを含む。

30

【0038】

ホストシステム102は、コンピュータまたはサーバなど、データ処理システムとして実装され得る。ホストシステム102は、インターフェース回路115を通してメモリ110および1つまたは複数の入出力(I/O)デバイスに結合されたホストプロセッサ105を含む。ホストシステム102は、メモリ110内にコンピュータ可読命令(プログラムコード)を記憶することが可能である。メモリ110は、コンピュータ可読記憶媒体の一例である。ホストプロセッサ105は、インターフェース回路115を介してメモリ110からアクセスされるプログラムコードを実行することが可能である。

【0039】

メモリ110は、たとえば、ローカルメモリおよびバルク記憶デバイス(bulk storage device)など、1つまたは複数の物理メモリデバイスを含み得る。ローカルメモリは、概してプログラムコードの実際の実行中に使用される(1つまたは複数の)非永続的メモリデバイスを指す。ローカルメモリの例は、ランダムアクセスメモリ(RAM)、および/または、プログラムコードの実行中のプロセッサによる使用のために好適である様々なタイプのRAM(たとえば、ダイナミックRAMまたは「DRAM」あるいはスタティックRAMまたは「SRAM」)のいずれかを含む。バルク記憶デバイスは、永続的データ記憶デバイスを指す。バルク記憶デバイスの例は、限定はしないが、ハードディスクドライブ(HDD)、ソリッドステートドライブ(SSD)、フラッシュメモリ、読取り専用メモリ(ROM)、消去可能プログラマブル読取り専用メモリ(EPROM)、電氣的消去可能プログラマブル読取り専用メモリ(EEPROM)、または他の

40

50

好適なメモリを含む。ホストシステム 102 は、プログラムコードが実行中にバルク記憶デバイスから取り出されなければならない回数を低減するために少なくともあるプログラムコードの一時的記憶を行う 1 つまたは複数のキャッシュメモリ（図示せず）をも含む得る。

【0040】

インターフェース回路 115 の例は、限定はしないが、システムバスと入出力（I/O）バスとを含む。インターフェース回路 115 は、様々なバスアーキテクチャのいずれかを使用して実装され得る。バスアーキテクチャの例は、限定はしないが、拡張業界標準アーキテクチャ（EISA）バス、アクセラレーテッドグラフィックスポート（AGP）、ビデオエレクトロニクス規格協会（VESA）ローカルバス、ユニバーサルシリアルバス（USB）、および周辺構成要素相互接続エクスプレス（PCIe）バスを含む得る。

10

【0041】

I/O デバイス 120 は、直接、または介在する I/O コントローラを通してのいずれかで、ホストシステム 102、たとえば、インターフェース回路 115 に結合され得る。I/O デバイス 120 の例は、限定はしないが、キーボード、ディスプレイデバイス、ポインティングデバイス、1 つまたは複数の通信ポート、およびネットワークアダプタを含む。ネットワークアダプタは、ホストシステム 102 が、介在するプライベートまたは公衆ネットワークを通して他のシステム、コンピュータシステム、リモートプリンタ、および/またはリモート記憶デバイスに結合されるようになることを可能にする回路を指す。モデム、ケーブルモデム、イーサネットカード、およびワイヤレストランシーバが、ホストシステム 102 とともに使用され得る異なるタイプのネットワークアダプタの例である。

20

【0042】

1 つまたは複数の実施形態では、メモリ 110 は、ホストプロセッサ 105 によって実行され得るオペレーティングシステム（図示せず）および 1 つまたは複数のアプリケーション（図示せず）を記憶する。ランタイムライブラリ、および IC 150 のためのドライバも、ホストプロセッサ 105 によって実行され得る。1 つまたは複数の実施形態では、ランタイムライブラリは、ホストプロセッサ 105 によって実行される（1 つまたは複数の）アプリケーションとリンクされるか、またはさもなければ、それと一体化され得る。ランタイムライブラリは、ドライバを使用して IC 150 と通信するために使用される機能を含む。

30

【0043】

ホストシステム 102 は、実装されるデバイスおよび/またはシステムの特定のタイプに応じて、図示された構成要素よりも少数の構成要素、または図 1 に示されていない追加の構成要素を含む得る。さらに、含まれる特定のオペレーティングシステム、（1 つまたは複数の）アプリケーション、および/または I/O デバイスは、システムタイプに基づいて変動し得る。さらに、例示的な構成要素のうちの 1 つまたは複数は、別の構成要素に組み込まれるか、またはさもなければ、別の構成要素の一部を形成し得る。たとえば、プロセッサが、少なくともあるメモリを含む得る。ホストシステム 102 は、図 1 のアーキテクチャまたはそれと同様のアーキテクチャを使用して各々実装される単一のコンピュータあるいは複数のネットワーク化されたまたは相互接続されたコンピュータを実装するために使用され得る。

40

【0044】

IC 150 は、通信リンク 125 を介してホストシステム 102 に通信可能にリンクされる。たとえば、IC 150 は、ホストシステム 102 内に含まれ得る回路板に結合され得る。1 つまたは複数の実施形態では、通信リンク 125 は PCIe リンクである。ただし、通信リンク 125 は様々な異なる接続および/または通信プロトコルのうちのいずれかを使用して実装され得ることと、提供された例は限定として意図されていないことを諒解されたい。

【0045】

IC 150 は、インターフェース 155 と、スケジューリングシステム 160 と、複数の

50

C U 1 8 0 と、随意に、メモリコントローラ 1 8 5 とを含む。C U 1 8 0 は、ハードウェアアクセラレーションのために使用される回路ブロックである。C U 1 8 0 は、ホストプロセッサ 1 0 5 からオフロードされた動作、たとえば、コマンドを実施するために使用され得る。図 1 の例では、C U 1 8 0 は、スケジューリングシステム 1 6 0 によって制御される。

【 0 0 4 6 】

インターフェース 1 5 5 は、通信リンク 1 2 5 を介してホストシステム 1 0 2 と通信することが可能である。特定の実施形態では、インターフェース 1 5 5 は、ダイレクトメモリアクセス (D M A) 回路を含む P C I e インターフェースである。インターフェース 1 5 5 の D M A 部分は、メモリコントローラ 1 8 5 を介して、コマンド待ち行列 1 6 2、コマンド要求レジスタ 1 6 4、コマンド完了レジスタ 1 6 6、および / またはメモリ 1 9 0 など、1 つまたは複数のレジスタおよび / またはメモリを読み取り、ならびに / あるいは 1 つまたは複数のレジスタおよび / またはメモリに書き込むことが可能である。

10

【 0 0 4 7 】

スケジューリングシステム 1 6 0 は、コマンド待ち行列 1 6 2 と、コマンド要求レジスタ 1 6 4 と、コマンド完了レジスタ 1 6 6 と、コントローラ 1 6 8 と、C U D M A 回路 1 7 0 と、C U 割込み回路 1 7 2 と、C U 完了レジスタ 1 7 4 とを含む。別々に示されているが、1 つまたは複数の他の実施形態では、インターフェース 1 5 5 はスケジューリングシステム 1 6 0 の一部と見なされ得る。

【 0 0 4 8 】

コマンド待ち行列 1 6 2 は、ホストシステム 1 0 2 からオフロードされたコマンドを受信することが可能である。コマンド待ち行列 1 6 2 は、サポート回路をもつメモリとして実装され得る。たとえば、コマンド待ち行列 1 6 2 は、サポート論理をもつランダムアクセスメモリ (R A M) として実装され得る。1 つまたは複数の実施形態では、コマンド待ち行列 1 6 2 は、I C 1 5 0 内に 1 つまたは複数のブロック R A M (B R A M) として実装される。コマンド待ち行列 1 6 2 は、(たとえば、ランタイムライブラリおよび / またはドライバを実行する際に) ホストプロセッサ 1 0 5 によって書き込まれ、コントローラ 1 6 8 によって読み取られ、書き込まれ得る。コマンド待ち行列 1 6 2 は、「スロット」と呼ばれる固定サイズの複数の領域に区分され得る。各スロットは、ホストプロセッサ 1 0 5 からのコマンド (たとえば、1 つのコマンド) を記憶することが可能である。

20

30

【 0 0 4 9 】

特定の実施形態では、各スロットは同じサイズである。一態様では、スロットのサイズは、スケジューリングシステム 1 6 0 によってハンドリングされることになる最も大きいコマンドのサイズがそのとき知られているので、スケジューリングシステム 1 6 0 を含む回路設計のコンパイル時間において設定され得る。スロットは、スケジューリングシステム 1 6 0 によってハンドリングされる最も大きいコマンドを記憶するようにサイズ決定され得る。1 つまたは複数の他の実施形態では、スロットのサイズは、ホストプロセッサ 1 0 5 によって実行されるランタイムライブラリによって設定され得る。いずれの場合も、ホストプロセッサ 1 0 5 は、コマンド待ち行列 1 6 2 の利用可能なスロットに新しいコマンドを書き込むことが可能である。

40

【 0 0 5 0 】

コマンド要求レジスタ 1 6 4 は、新しいコマンドがコマンド待ち行列 1 6 2 のスロット内でいつ利用可能になるかをコントローラ 1 6 8 に通知することが可能である。コマンド要求レジスタ 1 6 4 は、後で、たとえば、コマンド待ち行列 1 6 2 にコマンドを書き込んだことに応答して、または書き込んだ直後に、ホストシステム 1 0 2 によって書き込まれ得る、サポート回路をもつメモリとして実装され得る。

【 0 0 5 1 】

コマンド要求レジスタ 1 6 4 は、ホストシステム 1 0 2 によって書き込まれ得る複数のロケーションを含む。コマンド要求レジスタ 1 6 4 内の各ロケーション、たとえば、ビットロケーションは、たとえば 1 対 1 で、コマンド待ち行列 1 6 2 内のスロットに対応する。

50

コマンド要求レジスタ164に書き込むホストシステム102のアクションは、コマンド要求レジスタ164に、新しいコマンドがコマンド待ち行列162の-slot内で利用可能であることを指示する、コントローラ168への割込みを生成させる。割込みに応答して、コントローラ168は、コマンド待ち行列162のどの1つまたは複数の-slotが新しいコマンドを含んでいるかを決定するために、コマンド要求レジスタ164を読み取る。コマンド要求レジスタ164を含むことは、新しいコマンドがその中で利用可能であるかどうかを決定するためにコマンド待ち行列162を継続的にポーリングしなければならないことからコントローラ168を解放する。コマンド要求レジスタ164は、読取り時クリアレジスタとして実装され得る。したがって、コントローラ168によって読み取られると、コマンド要求レジスタ164のコンテンツはクリアされる。

10

【0052】

コマンド要求レジスタ164によってコントローラ168への割込みが生成され、さらなるコマンドがコマンド待ち行列162に記憶されると、ホストプロセッサ105は、コマンド待ち行列162中のさらなる新しいコマンドを書き込まれた異なる-slotに対応する、コマンド要求レジスタ164中のロケーションに書き込むことを続ける。コマンド要求レジスタ164によって生成された割込みは、新しい割込みを生成するのではなく、少なくともコントローラ168が割込みをサービスする(service)まで、セットされたままであり得る。コントローラ168は、コマンド要求レジスタ164から割込みを受信したことに応答して、新しいコマンドを記憶する、コマンド待ち行列162の-slotを決定するために、コマンド要求レジスタ164を読み取ることが可能である。コントローラ168がコマンド要求レジスタ164を読み取ると、コマンド要求レジスタ164のコンテンツはクリアされる。

20

【0053】

コマンド完了レジスタ166は、(1つまたは複数の)特定のコマンドがいつCU180上の実行を終えたかをホストプロセッサ105に通知することが可能である。コマンド完了レジスタ166は、サポート回路をもつメモリとして実装され得る。コマンド完了レジスタ166は、コマンド待ち行列162中に含まれている-slotの数よりも大きいかまたはそれに等しい数のロケーション、たとえばビットロケーションを含むようにサイズ決定され得る。特定の実施形態では、コマンド完了レジスタ166中の各ロケーションは、たとえば1対1で、コマンド待ち行列162の-slotに対応する。コマンド完了レジスタ166中のロケーションに記憶された値は、コマンド待ち行列162の対応する-slotに記憶されたコマンドが実行を終えたかどうかを指示する。たとえば、コマンド完了レジスタ166中のビット位置「X」に書き込まれた「1」は、コマンド待ち行列162の-slot「X」に記憶されたコマンドが実行を完了したことを指示する。

30

【0054】

1つまたは複数の実施形態では、コントローラ168は、対応するコマンドがCU180中で実行を完了したことを指示するために、コマンド完了レジスタ166中の特定のロケーションを書き込むことが可能である。コマンド完了レジスタ166は、コマンドが実行を完了したことを指示する、ホストプロセッサ105への割込みを生成することが可能である。ホストプロセッサ105は、ランタイムライブラリおよび/またはドライバの実行によって、コマンド待ち行列162からの、いずれかのコマンドが実行し終わったかどうか、および/またはどのコマンドが実行し終わったかを決定するために、コマンド完了レジスタ166中のビットを読み取ることが可能である。

40

【0055】

特定の実施形態では、コマンド完了レジスタは複数のサブレジスタを含み得、各サブレジスタが、ホストプロセッサ105によって他のサブレジスタから独立して読み取られ得る。さらに、サブレジスタの各々は、(1つまたは複数の)コマンドの完了を指示するためにそのサブレジスタ内の1つまたは複数のロケーションが書き込まれたことを指示する、ホストプロセッサ105への独立した割込みを生成することが可能である。

【0056】

50

例示的なおよび非限定的な例として、コマンド待ち行列 162 中に 128 個の slots がある場合について考える。コマンド完了レジスタ 166 は、(たとえば、1対1で)コマンド待ち行列 162 の 128 個の slots に対応する 128 ビットロケーションをも含み得る。この例では、コマンド完了レジスタ 166 は 4 つのサブレジスタを含み、各サブレジスタは 32 ビットレジスタである。

【0057】

たとえば、コマンド待ち行列 162 の (たとえば、slot 0 ~ 127 のうちの) slot 10 からのコマンドが実行を完了した場合、コマンド完了レジスタ 166 中の、slot 10 に対応する (たとえば、ビット 0 ~ 31 のうちの) ビットロケーション 10 が、更新される。ビットロケーション 10 は、複数のサブレジスタのうちの第 1 のサブレジスタ中に位置する。複数のサブレジスタのうちの第 1 のサブレジスタは、ホストプロセッサ 105 への割込みを生成する (たとえば、セットする)。割込みは、ホストプロセッサ 105 が第 1 のサブレジスタを読み取る時間まで、セットされたままであり得る。したがって、slot 0 ~ 31 中の他のコマンドが実行を完了した場合、第 1 のサブレジスタ中のコマンド完了レジスタ 166 中の対応するビットロケーションが更新され、割込みは、ホストプロセッサ 105 が第 1 のサブレジスタを読み取るまで、セットされたままである。各サブレジスタはそれ自体の割込みを生成することが可能であるので、ホストプロセッサ 105 は、4 つのサブレジスタの各々に対応するすべての 128 ビットを読み取るのではなく、第 1 のサブレジスタ (たとえば、32 ビット) のみを読み取る必要がある。

10

【0058】

この例を続けると、コマンド待ち行列 162 の slot 35 からのコマンドが実行を完了した場合、コマンド完了レジスタ 166 中の、slot 35 に対応するビットロケーション 35 が、更新される。ビットロケーション 35 は、複数のサブレジスタのうちの第 2 のサブレジスタ中に位置する。複数のサブレジスタのうちの第 2 のサブレジスタは、ホストプロセッサ 105 への割込みを生成する。割込みは、ホストプロセッサ 105 が第 2 のサブレジスタを読み取る時間まで、セットされたままであり得る。したがって、slot 32 ~ 63 中の他のコマンドが実行を完了した場合、第 2 のサブレジスタ中のコマンド完了レジスタ 166 中の対応するビットロケーションが更新され、割込みは、ホストプロセッサ 105 が第 2 のサブレジスタを読み取るまで、セットされたままである。各サブレジスタはそれ自体の割込みを生成することが可能であるので、ホストプロセッサ 105 は、4 つのサブレジスタの各々に対応するすべての 128 ビットを読み取るのではなく、第 2 のサブレジスタ (たとえば、32 ビット) のみを読み取る必要がある。

20

30

【0059】

上記の例から、割込みがコマンド完了レジスタ 166 の特定のサブレジスタのために生成され、同じサブレジスタ内で対応するロケーションを有するコマンドを実行するさらなる CU が実行を完了すると、割込みは、単にセットされたままである。ホストプロセッサ 105 は、割込みをサービスするとき、特定のサブレジスタを読み取り、サブレジスタ中の実行を終えたコマンドの各々を決定する。

【0060】

1 つまたは複数の実施形態では、コントローラ 168 は、IC 150 中に埋め込まれ、プログラムコードを実行することが可能であるプロセッサとして実装される。特定の実施形態では、プロセッサは、ハードワイヤードである。他の実施形態では、プロセッサは、IC 150 のプログラマブル回路を使用して実装される「ソフトプロセッサ」である。コントローラ 168 は、コントローラ 168 に本明細書で説明される様々な動作を実施させるファームウェアを実行し得る。

40

【0061】

コントローラ 168 は、コマンド待ち行列 162 内の slot (および、そこに記憶されたコマンド) のステータスを追跡することが可能である。コントローラ 168 は、CU 180 のステータスを追跡することも可能である。たとえば、コントローラ 168 は、各 CU がアイドル (たとえば、利用可能) であるのか開始済み (たとえば、ビジー) であるの

50

かを指定するCU180のリストを維持することが可能である。CU180のステータスを追跡することによって、コントローラ168は、コマンド待ち行列162の(1つまたは複数の)スロットに記憶された(1つまたは複数の)新しいコマンドを実行するために(1つまたは複数の)アイドルCU180を選択することが可能である。

【0062】

図1の例では、コントローラ168は、CU180の状態を決定するためにCU180への直接アクセスを有しない。コントローラ168は、コマンドの完了に回答してそれぞれのCU180によって生成された割込みに基づいてCU180の状態を決定することが可能である。割込みは、以下でより詳細に本明細書で説明されるように、CU割込み回路172に向けられる。

【0063】

CU DMA回路170は、CU180を設定し、開始することが可能である。コントローラ168は、たとえば、所与のコマンドを実行するようにCU180を選択した後に、実行されるべき特定のスロット(たとえば、およびコマンド)をCU DMA回路170に通知する。述べられたように、スロットは、コマンドを実行するために選択される特定のCU180を指定するために更新される。CU DMA回路170は、スロットからのコマンドを、実行のために選択されたCU180に提供することが可能である。

【0064】

CU割込み回路172は、CU180によって生成された割込みを受信およびサービスすることが可能である。CU割込み回路172は、CU180によって生成された割込みをハンドリングすることが可能であり、それにより、割込みをサービスしなければならないことからコントローラ168を解放する。CU割込み回路172は、CU180がさらなるコマンドを実行するためにCU DMA回路170によって新たにプログラムされ得るように、コマンドの実行を完了したCU180をリアーム(たとえば、リセット)することも可能である。

【0065】

1つまたは複数の実施形態では、CU割込み回路172をCU180に結合するインターフェースは、コントローラ168をクロック制御するために使用されるものよりも遅いクロック周波数において動作する制御インターフェースであり得る。CU180をリアームすることをCU割込み回路172にオフロードすることによって、コントローラ168は、他の機能を自由に実施することができ、より遅いインターフェースを介してリアーム動作が完了するのを待つ必要がない。

【0066】

1つまたは複数の実施形態では、CU割込みは、連続的にワイヤ接続され、たとえば、インデックス付けされ、したがって、CU割込みが発動したとき、CU割込み回路172は、割込みワイヤのインデックスに基づいて、割込みを始動した特定のCU180を識別することが可能である。CU割込み回路172は、割込みインデックスを特定のCUアドレスにマッピングするアドレステーブルを含み得る。特定の実施形態では、アドレステーブルは、ルックアップテーブルRAM(たとえば、LUTRAM)から構築され、CUの対応する割込みインデックスへの1:1マッピングを有するCUアドレスの連続リストを含んでいる。CU割込み回路172は、割込みを生成した特定のCUをリアームするためにアドレステーブルからのアドレスを使用することが可能である。CU割込み回路172は、受信された割込みのインデックスに基づいてCU完了レジスタ174中の正しいロケーションに書き込むことが可能である。

【0067】

CU完了レジスタ174は、CU割込み回路172によって書き込まれ、コントローラ168によって読み取られる。CU完了レジスタ174は、複数のロケーション、たとえばビットロケーションを含み、各ロケーションが、(たとえば、1対1で)特定のCU180に対応する。CU割込み回路172が、コマンドの実行を完了した選択されたCU180から割込みを受信したことに回答して、CU割込み回路172は、選択されたCU18

10

20

30

40

50

0に対応するCU完了レジスタ174中のロケーションに書き込む。

【0068】

CU完了レジスタ174中のロケーションに書き込むことと協調して、CU割込み回路172は、コントローラ168への割込みをも生成する。特定の実施形態では、コントローラ168がCU完了レジスタ174を読み取る前に、CU180からの複数の割込みがCU割込み回路172によって処理され得る。したがって、CU割込み回路172は、複数のCU割込みラインを効果的に合体させて、コントローラ168への単一の割込みラインにする。

【0069】

CU割込み回路172によってコントローラ168への割込みが生成され、異なるCU180からさらなる割込みが受信されると、CU割込み回路172は、割込みを生成した異なるCUに対応する、CU完了レジスタ174中のロケーションに書き込むことを続ける。CU割込み回路172によって生成された割込みは、セットされたままである。コントローラ168は、CU割込み回路172から割込みを受信したことに応答して、CU完了レジスタ174中の書き込まれた特定のロケーションに基づいて、実行を終えた(1つまたは複数の)特定のCU180を決定するために、CU完了レジスタ174を読み取ることが可能である。CU完了レジスタ174は、読取り時クリアレジスタとして実装され得る。したがって、CU完了レジスタ174のコンテンツは、コントローラ168がCU完了レジスタ174を読み取ったことに応答してクリアされ得る。

【0070】

コントローラ168は、CU180とコマンド(たとえば、コマンド待ち行列162の-slot)との間でトランスレートすることが可能である。コントローラ168は、CU完了レジスタ174の各セットされたビットを、割込みを生成したCU上で前に開始されたコマンド(たとえば、slot)と関連させることができる。たとえば、コントローラ168は、CUと、コマンド待ち行列162のslotに基づいてCUによって実行されるコマンドとを関連させるデータ構造を維持することが可能である。コマンドが特定のCU180によって実行されたとき、コントローラ168は、そのCUを、コマンドを記憶するコマンド待ち行列162の特定のslotに関連付けることが可能である。

【0071】

コントローラ168は、実行を完了した特定のコマンドを指示するためにコマンド完了レジスタ166中のロケーションを更新することがさらに可能である。たとえば、CU180-1が、slot5に記憶されたコマンドの実行を完了する場合について考える。CU割込み回路172は、CU180-1から割込みを受信し、CU180-1に対応する、CU完了レジスタ174中のロケーションを更新する。コントローラ168は、CU割込み回路172から割込みを受信し、CU完了レジスタ174を読み取ることによってCU180-1がコマンドの実行を完了したと決定する。コントローラ168は、CU180-1がslot5からのコマンドを実行したと決定することが可能である。したがって、コントローラ168は、コマンド完了レジスタ166中の、slot5に対応するロケーションに書き込むことが可能である。説明されたように、コマンド完了レジスタ166に書き込むことにより、ホストプロセッサ105への割込みが生成される。

【0072】

図1の例では、コントローラ168は、コマンド待ち行列162、コマンド要求レジスタ164、コマンド完了レジスタ166、CUDMA回路170、CU割込み回路172、およびCU完了レジスタ174に結合される。1つまたは複数の実施形態では、コントローラ168は、直接接続を使用してそのような回路ブロックに結合され得る。他の実施形態では、コントローラ168は、相互接続回路を通してそのような回路ブロックに結合され得る。インターフェース155は、直接接続または相互接続回路を介して、コマンド待ち行列162、コマンド要求レジスタ164、およびコマンド完了レジスタ166に結合され得る。同様に、CU180は、直接接続または相互接続回路を介してCUDMA回路170および/またはCU割込み回路172に結合され得る。

10

20

30

40

50

【0073】

相互接続回路の一例は、アドバンスドマイクロコントローラバスアーキテクチャ（AMBA：Advanced Microcontroller Bus Architecture）拡張可能インターフェース（AXI：AMBA eXtensible Interface）バスである。AMBA AXIバスは、回路ブロックおよび/またはシステムの間オンチップ接続を確立するために使用するための埋込みマイクロコントローラバスインターフェースである。AXIは、相互接続回路の例示的な例として提供され、本開示内で説明される実施形態の限定として意図されていない。使用され得る相互接続の他の例は、限定はしないが、他のタイプのバス、ネットワークオンチップ（NoC）、クロスバー、または他のタイプのスイッチを含む。

10

【0074】

1つまたは複数の実施形態では、メモリコントローラ185はIC150内に実装される。一例では、メモリコントローラ185は、ハードワイヤード回路ブロックとして実装される。別の例では、メモリコントローラ185は、IC150のプログラマブル回路を使用して実装される。1つまたは複数の他の実施形態では、メモリコントローラ185は、IC150の外部に、たとえば、IC150に結合された別のIC中で、実装され得る。

【0075】

メモリ190は、メモリコントローラ185を介してCU180によってアクセスされ（たとえば、読み取られ、および/または書き込まれ）得るオフチップメモリとして含まれ得る。1つまたは複数の実施形態では、ホストプロセッサ105も、インターフェース155、ホストDMA回路192、およびメモリコントローラ185を介して、メモリ190にアクセス（たとえば、メモリ190を読み取り、および/またはメモリ190を書き込み）得る。メモリ190は、IC150が結合される回路板に結合され得る。したがって、特定の実施形態では、CU180は、メモリ190からコマンド実行のための入力引数を取り出し、実行されたコマンドの結果をメモリ190に記憶し得る。ホストプロセッサ105は、次いで、メモリ190から結果を取り出し得る。

20

【0076】

説明と例示の容易さのために、「オープンコンピューティング言語（Open Computing Language）」または「OpenCL（商標）」という専門用語が、本出願全体にわたって使用される。HCPをサポートする様々な異なるフレームワークおよび/または言語のうちいずれかが使用され得ることを諒解されたい。したがって、本発明の構成は、OpenCLに限定されるものではない。むしろ、本開示内で説明される実施形態は、様々な好適なおよび異なるHCPフレームワークのうちいずれかとともに使用され得る。本開示内で説明される実施形態とともに使用され得る他のHCPおよび/またはハードウェアアクセラレーションフレームワークの例は、限定はしないが、オープンマルチプロセッシング（Open Multi-Processing）（OpenMP（登録商標））およびCUDA（登録商標）を含む。

30

【0077】

図2は、本開示内で説明される1つまたは複数の実施形態による、図1のスケジューリングシステム160によって実施される例示的な動作を示す。図2の例では、スケジューリングシステム160のいくつかの回路ブロックは、説明の明快さおよび容易さのために省略される。

40

【0078】

描かれているように、コントローラ168は、複数のデータ構造、たとえば、テーブルまたはリストを含む。たとえば、コントローラ168は、コマンド待ち行列162中に含まれるスロット、およびスロットがコマンドを含むかどうかをリストするスロットリスト202と、CU、および実行のためにCUに割り当てられたスロット（たとえば、コマンド）をリストするスロットCUリスト204と、CU、およびアイドルまたは開始済みとしての各CUのステータスをリストするCUリスト206とを含む。CUリスト206は、各CUのためのベースアドレスをも指定し得る。

50

【 0 0 7 9 】

図 2 の例では、ホストプロセッサ 1 0 5 は、コマンド待ち行列 1 6 2 のスロット 2 1 0 内に新しいコマンド 2 0 8 を記憶し、スロット 2 1 0 に対応する、コマンド要求レジスタ 1 6 4 中のロケーションに書き込んだ。コマンド要求レジスタ 1 6 4 中のロケーションが書き込まれたことに応答して、コマンド要求レジスタ 1 6 4 は、コントローラ 1 6 8 への割込みを生成する。コントローラ 1 6 8 は、割込みに応答して、コマンド要求レジスタ 1 6 4 中のどのロケーションが書き込まれたかを決定するために、コマンド要求レジスタ 1 6 4 を読み取り、コマンド待ち行列 1 6 2 の対応するスロット中の新しいコマンドの利用可能性を指示する。述べられたように、コントローラ 1 6 8 によって読み取られると、コマンド要求レジスタ 1 6 4 はクリアされる。

10

【 0 0 8 0 】

コントローラ 1 6 8 は、スロット 2 1 0 内の新しいコマンド 2 0 8 を読み取ることが可能である。たとえば、コントローラ 1 6 8 は、コマンド要求レジスタ 1 6 4 を読み取ることによって、スロット 2 1 0 が、そこに記憶された新しいコマンドを有すると決定する。特定の実施形態では、コントローラ 1 6 8 は、新しいコマンド 2 0 8 を処理するために必要な時間の量を低減するために、新しいコマンド 2 0 8 の選択された部分（たとえば、全体よりも小さい部分）のみを読み取ることが可能である。図 2 の例では、新しいコマンド 2 0 8 は領域 2 1 2 を含む。領域 2 1 2 は、新しいコマンド 2 0 8 を実行することが可能である C U のリストを指定する。コマンド待ち行列 1 6 2 のスロットに書き込まれた各コマンドは、コマンドを実行することが可能である、したがって、コマンドを実行することを許容される C U をリストする、領域 2 1 2 などの領域を含み得る。いくつかの C U は、いくつかのコマンドを実行することが可能であり得るにすぎない。

20

【 0 0 8 1 】

図 2 の例では、コントローラ 1 6 8 は、新しいコマンド 2 0 8 の残部を読み取ることなしに領域 2 1 2 を直接読み取ることが可能である。特定の実施形態では、領域 2 1 2 は、新しいコマンド 2 0 8 を実行することが可能である特定の C U を指定する、C U インデックスのリストを含む。たとえば、領域 2 1 2 は、1、2、4、および 5 のインデックスを指定し得る。コントローラ 1 6 8 は、新しいコマンド 2 0 8 を実行することを許可された C U（たとえば、インデックス 1、2、4、および 5）のうちのいずれかが利用可能であるかどうかを決定するために、領域 2 1 2 において指定された C U（たとえば、C U 1 8 0 - 1、1 8 0 - 2、1 8 0 - 4、および 1 8 0 - 5）のリストを C U リスト 2 0 6 と比較する。

30

【 0 0 8 2 】

コントローラ 1 6 8 は、アイドル（たとえば、利用可能）であり、また、新しいコマンド 2 0 8 を実行することが可能である C U を、C U リスト 2 0 6 から選択することが可能である。コントローラ 1 6 8 は、たとえば、一致を決定するために、領域 2 1 2 からの C U インデックスを C U リスト 2 0 6 からの利用可能な C U と比較することが可能である。例示の目的で、C U 1 8 0 - 1 はアイドルであり得、C U 1 8 0 - 2、1 8 0 - 4、および 1 8 0 - 5 は開始済みである。コントローラ 1 6 8 は、C U 1 8 0 - 1 を選択し、C U 1 8 0 - 1 のベースアドレスを領域 2 1 2 に書き込むこと、たとえば、C U 1 8 0 - 1 のベースアドレスを、領域 2 1 2 と領域 2 1 2 中に含まれている許容 C U のリストとに上書きすることが可能である。領域 2 1 2 に上書きすることは、領域 2 1 2 中に含まれている情報が、コントローラ 1 6 8 を対象とし、新しいコマンド 2 0 8 を実行するための C U が選択されると、新しいコマンド 2 0 8 を実行するために必要とされないので、メモリを節約する。1 つまたは複数の他の実施形態では、コントローラ 1 6 8 は、新しいコマンド 2 0 8 のいかなる部分にも上書きすることなしに、選択された C U のベースアドレスをスロット 2 1 0 内に記憶し得る。

40

【 0 0 8 3 】

選択された C U、たとえば C U 1 8 0 - 1 のベースアドレスを領域 2 1 2 に書き込むことは、選択された C U を、スロット 2 1 0 に、および新しいコマンド 2 0 8 に効果的に割り

50

当てる。コントローラ 168 は、選択された CU (たとえば、CU 180 - 1) とスロット 210 との間の関連付けを指定するためにスロット CU リスト 204 を更新することも可能である。さらに、コントローラ 168 は、選択された CU 180 - 1 が開始済みであることを指示するために CU リスト 206 を更新する。

【0084】

コマンド 208 を実行するための CU を選択すると、コントローラ 168 は、スロット 210 のインデックスを CU DMA 回路 170 に受け渡すことが可能である。例示の目的で、スロット 210 のスロットインデックスは 32 であり得る。コントローラ 168 は、スロットインデックス 32 を CU DMA 回路 170 中のレジスタに書き込むことが可能である。CU DMA 回路 170 は、コントローラ 168 からスロットインデックスを受信したことに応答して、スロット 210 に対応するスロットインデックス 32 にアクセスし、選択された CU 180 - 1 のベースアドレスを読み取る。

10

【0085】

CU DMA 回路 170 は、CU 180 - 1 のベースアドレスを使用して、コマンド 208、たとえば、コマンド 208 中に含まれているデータを、実行のために CU 180 - 1 に転送することが可能である。CU DMA 回路 170 は、コントローラ 168 が CU への実際のデータ転送に関与しないように、選択された CU に直接データを転送することが可能である。データが CU 180 - 1 に転送されると、CU DMA 回路 170 は、CU 180 - 1 に、コマンドを実行することを開始するかまたは始めるようにという信号を出す。

20

【0086】

選択された CU に転送された新しいコマンド 208 は、限定はしないが、選択された CU がそれに基づいて動作すべき入力引数および/または選択された CU が結果を書き込むべきであるメモリアドレスを含んでいる、メモリ (たとえば、メモリ 190) 中のロケーションへのアドレスまたはポインタを含む、様々な情報を含み得る。

【0087】

図 3 A および図 3 B は、まとめて、本開示内で説明される 1 つまたは複数の実施形態による、HCP のための動作の例示的な方法 300 を示す。方法 300 は、CU を含む IC にコマンドをオフロードするために、図 1 に関して説明された HCP などの HCP によって実施される例示的な動作を示す。方法 300 は、HCP 100 が動作している状態において始まり得る。ホストプロセッサ 105 は、少なくとも部分的に、IC 150 およびその中に含まれているスケジューリングシステム 160 と通信するための機能を提供するランタイムライブラリの実行を通して、図 3 A および図 3 B を参照しながら説明される動作を実施することが可能である。述べられたように、ホストプロセッサ 105 はさらに、IC 150 と通信するためのドライバを実行し得る。

30

【0088】

例示の目的で、方法 300 は、単一のコマンドを受信すること、およびそのコマンドを実行する単一の CU の動作に関して説明される。本開示内で説明される原理は、複数のコマンドがホストシステムから受信され、複数の異なる CU によって実行される場合に適用され得ることを諒解されたい。

40

【0089】

ブロック 302 において、ホストプロセッサは、コマンド待ち行列中の利用可能なスロットにコマンドを書き込む。ホストプロセッサは、たとえば、アプリケーションを実行する際に、コマンドがハードウェアアクセラレーションのために IC にオフロードされるべきであると決定する。そうする際に、ホストプロセッサ、たとえば、ランタイムライブラリは、コマンド待ち行列内のスロットの利用可能性を追跡し、コマンド待ち行列内のそのような利用可能なスロットにコマンドを書き込むことが可能である。

【0090】

ブロック 304 において、ホストプロセッサは、ブロック 302 において書き込まれたコマンド待ち行列のスロットに対応する、コマンド要求レジスタ中のロケーション、たとえ

50

ば、スロットインデックスに書き込む。たとえば、ホストプロセッサは、ブロック 3 0 2 においてスロットを書き込んだことに応答して、または書き込んだ直後に、コマンド要求レジスタ中のロケーションに書き込むことが可能である。

【 0 0 9 1 】

ブロック 3 0 6 において、コマンド要求レジスタは、ブロック 3 0 4 においてホストプロセッサがロケーションに書き込んだことに応答して、コントローラへの割込みを生成する。ブロック 3 0 8 において、コントローラは、コマンド要求レジスタからの割込みに応答して、コマンド待ち行列中のどのスロットがホストプロセッサからの新しいコマンドを含むまたは記憶するかを決定する。たとえば、コマンド要求レジスタからの割込みに応答して、コントローラは、コマンド要求レジスタ中のどのロケーションが書き込まれたかを決定するために、コマンド要求レジスタを読み取る。各ロケーションは、コマンド待ち行列の特定のスロットに対応し得る。コントローラがコマンド要求レジスタを読み取ると、コマンド要求レジスタのコンテンツはクリアされる。説明されたように、方法 3 0 0 は単一のコマンドのハンドリングを示すが、割込みがコマンド要求レジスタによって生成されるから、複数のコマンドがコマンド待ち行列に書き込まれていることがある。したがって、コントローラは、コマンド要求レジスタを読み取ることから、新しいコマンドを含む、コマンド待ち行列の複数のスロットの各々を決定することが可能である。

10

【 0 0 9 2 】

ブロック 3 1 0 において、コントローラは、どの（1つまたは複数の）CUが新しいコマンドを実行することが可能であるかを決定する。たとえば、コントローラは、コマンド待ち行列中のスロットから新しいコマンドの少なくとも一部分を読み取ることが可能である。特定の実施形態では、コントローラは、新しいコマンドを実行することが可能であるCUのインデックスのリストを指定する、新しいコマンドの特定の領域のみを読み取ることが可能である。コントローラは、新しいコマンドを実行することが可能であるCUを決定するために新しいコマンドの全体を読み取る必要がなく、それにより、CU中の新しいコマンドの実行を始動することについてのレイテンシを低減する。

20

【 0 0 9 3 】

ブロック 3 1 2 において、コントローラは、新しいコマンドを実行することが可能であるアイドルCUを選択する。たとえば、コントローラは、新しいコマンド自体から読み取られた、新しいコマンドを実行することが可能である（1つまたは複数の）CUを、アイドル（たとえば、利用可能）である（1つまたは複数の）CUのリストと比較することが可能である。コントローラは、新しいコマンドを実行するための両方のリスト上に含まれるCU（たとえば、新しいコマンドを実行することが可能であり、アイドルであるCU）を選択する。たとえば、コントローラは、新しいコマンドを実行することが可能であるものとして指定されたCUを、アイドルCUと一致させることが可能である。

30

【 0 0 9 4 】

ブロック 3 1 4 において、コントローラは、選択されたCUを新しいコマンドに割り当てる。たとえば、コントローラは、選択されたCUのためのベースアドレスを新しいコマンドを含むスロットに書き込むことが可能である。1つまたは複数の実施形態では、コントローラは、ベースアドレスをコマンドに付加する。1つまたは複数の他の実施形態では、コントローラは、選択されたCUのベースアドレスを、新しいコマンド自体内の新しいコマンドを実行することが可能であるCUのリストに上書きする。

40

【 0 0 9 5 】

特定の実施形態では、コントローラは、さらに、選択されたCUが現在開始済み（たとえば、さらなるコマンドを実行するために使用するために利用不可能）であることを指示するためにCUのステータスを追跡するために、および/または、選択されたCUと新しいコマンドとの間の関連付けを記憶するために使用される内部データ構造を更新することが可能である。コントローラは、選択されたCUを開始済みとマークしたが、選択されたCUは、動作を開始するようとの信号を実際は出されなかったことを諒解されたい。

【 0 0 9 6 】

50

ブロック 3 1 6 において、コントローラは、新しいコマンドを含んでいるスロットのインデックスを C U D M A 回路に書き込む。ブロック 3 1 8 において、C U D M A 回路は、スロットのインデックスを受信したことに応答して、新しいコマンドを含んでいるスロットから、選択された C U のベースアドレスを読み取る。たとえば、C U D M A 回路は、インデックスを使用してスロットにアクセスし、さらに、コントローラが選択された C U のベースアドレスを記憶した所定のロケーションに対してコマンドにインデックス付けすることが可能である。

【 0 0 9 7 】

ブロック 3 2 0 において、C U D M A 回路は、新しいコマンドを含んでいるスロットのコンテンツを選択された C U にコピーする。C U D M A 回路は、選択された C U に新しいコマンドを提供するために、選択された C U のベースアドレスを使用する。スロットのコンテンツ、たとえば、新しいコマンドは、選択された C U に入力されるべき引数がどこに記憶されたかを指定するメモリアドレスと、選択された C U が、新しいコマンドを実行することによって生成された結果を記憶するべきであるメモリアドレスとを含む。C U のベースアドレスは、コピーされる必要がない。ブロック 3 2 2 において、C U D M A 回路は、選択された C U の動作を開始する。

10

【 0 0 9 8 】

ブロック 3 2 4 において、選択された C U は、新しいコマンドの実行を完了すると、C U 割込み回路への割込みを生成する。述べられたように、C U からの割込ラインはインデックス付けされ、それにより、C U 割込み回路が、実行を完了した特定の C U を決定することを可能にし得る。

20

【 0 0 9 9 】

ブロック 3 2 6 において、選択された C U から割込みを受信したことに応答して、C U 割込み回路は、選択された C U に対応する、C U 完了レジスタ中のロケーションに書き込み、コントローラへの割込みを生成する。

【 0 1 0 0 】

ブロック 3 2 8 において、C U 割込み回路は、選択された C U をリアームすることが可能である。たとえば、C U がコマンドの実行を完了したとき、C U 割込み回路は、別のコマンドを実行するために使用されるより前に、選択された C U をリアームしなければならない。C U 割込み回路は、インターフェースを介して、選択された C U に対するいくつかのトランザクションを開始することによって、選択された C U をリアームする。C U 割込み回路は、C U から割込みを受信したことに応答して、これらの動作を実施することが可能である。説明されたように、C U 割込み回路が C U と通信するためのインターフェースは、コントローラよりも遅いクロック周波数において動作する。C U をリアームすることを実施するために C U 割込み回路を使用することによって、コントローラは、C U をリアームすることから解放され、いくつかのトランザクションを開始し、いくつかのトランザクションがより遅いインターフェースを介して完了するのを待つこととは対照的に、他の動作を実施し、それにより、スケジューリングシステムの全体的性能を改善することが可能である。

30

【 0 1 0 1 】

ブロック 3 3 0 において、コントローラは、C U 割込み回路から割込みを受信したことに応答して、コマンドの実行を終えた C U 、たとえば、この例では選択された C U を決定する。たとえば、コントローラは、C U 割込み回路からの割込みに応答して、C U 完了レジスタを読み取ることが可能である。コントローラは、C U 完了レジスタ中のどのロケーションが書き込まれたかに基づいて、実行を完了した特定の C U を決定する。C U 完了レジスタは読取り時クリアレジスタとして実装され得るので、C U 割込み回路は、さらなる C U からさらなる割込みを受信したことに応答して、C U 完了レジスタ中のロケーションに書き込むことを続け得る。しかしながら、C U 割込み回路は、さらなる割込みを生成することを続ける必要はない。C U 割込み回路は、コントローラが C U 完了レジスタを読み取ることによって応答するまで、前に生成された割込みをセットされたままに保ち得、C U

40

50

完了レジスタは、読み取られると、クリアされる。

【0102】

ブロック332において、コントローラは、ブロック330において決定されたCUを、CUによって実行されたコマンドにマッピングする。たとえば、コントローラは、CUへのコマンド待ち行列中のスロットのマッピングを指定する、コントローラによって維持される内部データ構造から、ブロック330において決定されたCUによって実行された特定のコマンド、たとえば、この例では新しいコマンドを決定することが可能である。

【0103】

ブロック334において、コントローラは、選択されたCUによる実行を終えた新しいコマンドに対応する、コマンド完了レジスタ中のロケーションに書き込む。ブロック336において、コントローラによって書き込まれたことに応答して、コマンド完了レジスタは、ホストプロセッサへの割り込みを生成する。特定の形態では、割り込みは、ホストシステムにおいて実行されるICのためのドライバに提供される。

10

【0104】

コマンド完了レジスタは読み取り時クリアレジスタとして実装され得るので、コントローラは、さらなるCUが実行を完了したことを指示するさらなる割り込みを受信したことに応答して、コマンド完了レジスタ中のロケーションに書き込むことを続け得る。しかしながら、コマンド完了レジスタは、ホストプロセッサへのさらなる割り込みを生成することを続ける必要はない。コマンド完了レジスタは、ホストプロセッサがコマンド完了レジスタを読み取ることによって応答するまで、前に生成された割り込みをセットされたままに保ち得、コマンド完了レジスタは、読み取られると、クリアされる。

20

【0105】

コマンド完了レジスタが複数のサブレジスタを含む形態では、各サブレジスタは独立して動作し得る。すなわち、各サブレジスタは、他のサブレジスタから独立してホストプロセッサへの割り込みを生成し、その特定のサブレジスタがホストシステムによって読み取られるまで、その割り込みを、セットされたものとして維持することが可能である。各サブレジスタは、他のサブレジスタから独立してクリアされる読み取り時クリアレジスタとして実装され得る。

【0106】

ブロック338において、コマンド完了レジスタからの割り込みに応答して、ホストプロセッサは、実行を終えた特定のコマンドを決定するために、コマンド完了レジスタを読み取る。説明されたように、コマンド完了レジスタは、次いで、クリアされる。特定の形態では、ホストプロセッサは、たとえば、カーネルドライバを実行する際に、コマンド待ち行列中のスロットの数に対応するスロットインデックスのリストを維持する。ホストプロセッサは、ちょうどコントローラがCU利用可能性を追跡することが可能であるように、コマンド待ち行列中のスロットの利用可能性を追跡することが可能である。

30

【0107】

ブロック340において、ホストプロセッサは、新しいコマンドの実行から結果を取り出す。説明されたように、選択されたCUによって生成された結果データは、新しいコマンド自体内でホストプロセッサによって最初に指定されたロケーションにおいて、メモリに記憶される。ホストプロセッサは、コマンドを生成し、ICに提供することの一部として、CUのための引数が記憶されるメモリアドレスと、CUが得られたデータを書き込むメモリアドレスとを管理する。ホストプロセッサは、CUによる消費のためにメモリアドレスにおいて引数を記憶し、メモリアドレスから結果を取り出すことが可能である。

40

【0108】

本明細書で説明される例示的な形態は、スケジューリングシステムがCUの動作をどのように制御するかを示す。ホストシステムは、コマンドが実行されたこと、およびそのようなコマンドがいつ実行を完了したかに気づいている。ホストシステムは、所与のコマンドを実行するために使用される特定のCUに気づいていない。

【0109】

50

本開示内で説明される本発明の構成によれば、コントローラは、利用可能なコマンドおよび利用可能なＣＵに基づいて、順不同のハードウェアスケジューリングを実施することが可能である。コントローラは、マルチユーザおよびマルチタスク動作が可能である。本明細書で説明されるスケジューリングシステムは、ホストシステムとＩＣとの間のインターフェースを介して生じたレイテンシを低減するかまたは隠すことが可能である。さらに、本明細書で説明されるスケジューリングシステムは、ホストプロセッサと、スケジューリングシステム中に含まれるコントローラとにかけられる作業負荷を低減し、それにより、ＨＣＰ全体の性能を増加させる。

【 0 1 1 0 】

図４は、ＩＣのための例示的なアーキテクチャ４００を示す。一態様では、アーキテクチャ４００は、プログラマブルＩＣ内に実装され得る。たとえば、アーキテクチャ４００は、ＦＰＧＡを実装するために使用され得る。アーキテクチャ４００はまた、ＩＣのシステムオンチップ（ＳｏＣ）タイプを表し得る。ＳｏＣは、プログラムコードを実行するプロセッサと、１つまたは複数の他の回路とを含むＩＣである。他の回路は、ハードワイヤード回路、プログラマブル回路、および／またはそれらの組合せとして実装され得る。回路は、互いと、および／またはプロセッサと協働して動作し得る。

10

【 0 1 1 1 】

図示のように、アーキテクチャ４００は、いくつかの異なるタイプのプログラマブル回路、たとえば、論理、ブロックを含む。たとえば、アーキテクチャ４００は、マルチギガビットトランシーバ（ＭＧＴ：multi-gigabit transceiver）４０１、設定可能論理ブロック（ＣＬＢ）４０２、ランダムアクセスメモリブロック（ＢＲＡＭ）４０３、入出力ブロック（ＩＯＢ）４０４、設定およびクロッキング論理（ＣＯＮＦＩＧ／ＣＬＯＣＫＳ）４０５、デジタル信号処理ブロック（ＤＳＰ）４０６、特殊なＩ／Ｏブロック４０７（たとえば、設定ポートおよびクロックポート）、ならびにデジタルクロックマネージャ、アナログデジタル変換器、システム監視論理などの他のプログラマブル論理４０８を含む、多数の異なるプログラマブルタイルを含み得る。

20

【 0 1 1 2 】

いくつかのＩＣでは、各プログラマブルタイルは、プログラマブル相互接続要素（ＩＮＴ）４１１を含み、ＩＮＴ４１１は、各隣接するタイル中の対応するＩＮＴ４１１との間の規格化された接続を有する。したがって、ＩＮＴ４１１は、まとめると、示されているＩＣのためのプログラマブル相互接続構造を実装する。各ＩＮＴ４１１は、図４の上部に含まれる例によって示されているように、同じタイル内のプログラマブル論理要素との間の接続をも含む。

30

【 0 1 1 3 】

たとえば、ＣＬＢ４０２は、ユーザ論理を実装するようにプログラムされ得る設定可能論理要素（ＣＬＥ）４１２と、単一のＩＮＴ４１１とを含み得る。ＢＲＡＭ４０３は、１つまたは複数のＩＮＴ４１１に加えてＢＲＡＭ論理要素（ＢＲＬ）４１３を含み得る。一般的に、タイル中に含まれるＩＮＴ４１１の数は、タイルの高さに依存する。描かれているように、ＢＲＡＭタイルは、５つのＣＬＢと同じ高さを有するが、他の数（たとえば、４つ）も使用され得る。ＤＳＰタイル４０６は、適切な数のＩＮＴ４１１に加えてＤＳＰ論理要素（ＤＳＰＬ）４１４を含み得る。ＩＯＢ４０４は、たとえば、ＩＮＴ４１１の１つのインスタンスに加えてＩ／Ｏ論理要素（ＩＯＬ）４１５の２つのインスタンスを含み得る。ＩＯＬ４１５に接続された実際のＩ／Ｏパッドは、ＩＯＬ４１５のエリアに制限されないことがある。

40

【 0 1 1 4 】

図４に描かれている例では、ダイの中心の近くの、たとえば、領域４０５、４０７、および４０８から形成された、列状エリアが、設定、クロック、および他の制御論理のために使用され得る。この列から延びる水平エリア４０９が、プログラマブルＩＣの幅にわたってクロックおよび設定信号を分散させるために使用され得る。

【 0 1 1 5 】

50

図 4 に示されているアーキテクチャを利用するいくつかの IC は、IC の大部分を作り上げる規則的な列状構造を損なう追加の論理ブロックを含む。追加の論理ブロックは、プログラマブルブロックおよび/または専用回路であり得る。たとえば、PROC 410 として示されているプロセッサブロックが、CLB および BRAM のいくつかの列にまたがる。

【0116】

一態様では、PROC 410 は、IC のプログラマブル回路を実装するダイの一部として作製される専用回路として、たとえば、ハードワイヤードプロセッサとして実装され得る。PROC 410 は、個々のプロセッサ、たとえば、プログラムコードを実行することが可能な単一のコアから、1 つまたは複数のコア、モジュール、コプロセッサ、インターフェースなどを有するプロセッサシステム全体まで、複雑さに幅がある様々な異なるプロセッサタイプおよび/またはシステムのいずれかを表し得る。

10

【0117】

別の態様では、PROC 410 は、アーキテクチャ 400 から省略され、説明されるプログラマブルブロックの他の種類のうちの 1 つまたは複数と置き換えられ得る。さらに、そのようなブロックは、PROC 410 の場合のようにプログラムコードを実行することができるプロセッサを形成するためにプログラマブル回路の様々なブロックが使用され得るという点で、「ソフトプロセッサ」を形成するために利用され得る。

【0118】

「プログラマブル回路」という句は、IC 内のプログラマブル回路要素、たとえば、本明細書で説明される様々なプログラマブルまたは設定可能回路ブロックまたはタイル、ならびに、IC にロードされた設定データに従って様々な回路ブロック、タイル、および/または要素を選択的に結合する相互接続回路を指す。たとえば、CLB 402 および BRAM 403 など、PROC 410 の外部にある、図 4 に示されている回路ブロックは、IC のプログラマブル回路と見なされる。

20

【0119】

概して、プログラマブル回路の機能性は、設定データが IC にロードされるまで確立されない。FPGA など、IC のプログラマブル回路をプログラムするために、設定ビットのセットが使用され得る。(1 つまたは複数の) 設定ビットは、一般的に、「設定ビットストリーム」と呼ばれる。概して、プログラマブル回路は、設定ビットストリームを IC に最初にロードしなければ、動作可能でないか、または機能可能でない。設定ビットストリームは、プログラマブル回路内に特定の回路設計を効果的に実装する。回路設計は、たとえば、プログラマブル回路ブロックの機能的態様と、様々なプログラマブル回路ブロックの間の物理的接続性とを指定する。

30

【0120】

「ハードワイヤード」または「ハード化 (harden)」される、すなわち、プログラマブルでない回路が、IC の一部として製造される。プログラマブル回路とは異なり、ハードワイヤード回路または回路ブロックは、設定ビットストリームのローディングを通して IC の製造後に実装されない。ハードワイヤード回路は、概して、たとえば、設定ビットストリームを、IC、たとえば、PROC 410 に最初にロードすることなしに機能可能である、専用回路ブロックおよび相互接続を有すると見なされる。

40

【0121】

いくつかの事例では、ハードワイヤード回路は、IC 内の 1 つまたは複数のメモリ要素に記憶されたレジスタセッティングまたは値に従ってセットまたは選択され得る 1 つまたは複数の動作モードを有し得る。動作モードは、たとえば、IC への設定ビットストリームのローディングを通してセットされ得る。この能力にもかかわらず、ハードワイヤード回路が、IC の一部として製造されたとき、動作可能であり、特定の機能を有するので、ハードワイヤード回路はプログラマブル回路と見なされない。

【0122】

SoC の場合、設定ビットストリームは、プログラマブル回路内に実装されるべきである

50

回路と、P R O C 4 1 0またはソフトプロセッサによって実行されるべきであるプログラムコードとを指定し得る。いくつかの場合には、アーキテクチャ400は、適切な設定メモリおよび/またはプロセッサメモリに設定ビットストリームをロードする専用設定プロセッサを含む。専用設定プロセッサは、ユーザ指定のプログラムコードを実行しない。他の場合には、アーキテクチャ400は、設定ビットストリームを受信し、設定ビットストリームを適切な設定メモリにロードし、および/または実行のためのプログラムコードを抽出するために、P R O C 4 1 0を利用し得る。

【0123】

スケジューリングシステム160は、図4に関して説明されるようなアーキテクチャを有するICを使用して実装され得る。たとえば、プログラマブル回路内のB R A Mおよび/またはレジスタは、追加のプログラマブル回路から作成されたさらなるサポート回路とともに、本明細書で説明されるレジスタのうちの1つまたは複数を実装するために使用され得る。P R O C 4 1 0は、コントローラ168を実装するために使用され得る。プログラマブル回路は、インターフェース155とC U 1 8 0とを実装するために使用され得る。特定の実施形態では、メモリコントローラ185は、ハードワイヤードである。他の実施形態では、メモリコントローラ185はまた、プログラマブル回路を使用して実装される。

10

【0124】

図4は、プログラマブル回路、たとえば、プログラマブルファブリックを含むICを実装するために使用され得る例示的なアーキテクチャを示すことを意図される。たとえば、1つの列中の論理ブロックの数、列の相対幅、列の数および順序、列中に含まれる論理ブロックのタイプ、論理ブロックの相対サイズ、および図4の上部に含まれる相互接続/論理実装形態は、例示にすぎない。実際のICでは、たとえば、C L Bの2つ以上の隣接する列は、一般的に、ユーザ回路設計の効率的な実装を容易にするために、C L Bが現れるところならどこでも含まれる。しかしながら、隣接するC L B列の数は、ICの全体的サイズとともに変動し得る。さらに、IC内のP R O C 4 1 0などのブロックのサイズおよび/または配置は、例示のためのものにすぎず、限定として意図されていない。

20

【0125】

説明のために、特定の名称が、本明細書で開示される様々な発明概念の完全な理解を提供するために記載される。しかしながら、本明細書で使用される専門用語は、本発明の構成の特定の態様を説明するためのものにすぎず、限定するものではない。

30

【0126】

本明細書で定義される単数形「a」、「an」および「the」は、文脈が別段に明確に指示するのでなければ、複数形をも含むものとする。

【0127】

本明細書で定義される「少なくとも1つ」、「1つまたは複数」、および「および/または」という用語は、別段に明記されていない限り、運用において連言的と選言的の両方である、オープンエンド表現である。たとえば、「A、B、およびCのうちの少なくとも1つ」、「A、B、またはCのうちの少なくとも1つ」、「A、B、およびCのうちの1つまたは複数」、「A、B、またはCのうちの1つまたは複数」、および「A、B、および/またはC」という表現の各々は、Aのみ、Bのみ、Cのみ、AとBを一緒に、AとCを一緒に、BとCを一緒に、またはAとBとCを一緒に、を意味する。

40

【0128】

本明細書で定義される「自動的に」という用語は、ユーザ介入なしに、を意味する。本明細書で定義される「ユーザ」という用語は、人間を意味する。

【0129】

本明細書で定義される「コンピュータ可読記憶媒体」という用語は、命令実行システム、装置、またはデバイスが使用するための、あるいはそれとともに使用するためのプログラムコードを含んでいるかまたは記憶する記憶媒体を意味する。本明細書で定義される「コンピュータ可読記憶媒体」は、それ自体は、一時的な伝搬信号でない。コンピュータ可読

50

記憶媒体は、限定はしないが、電子記憶デバイス、磁気記憶デバイス、光記憶デバイス、電磁記憶デバイス、半導体記憶デバイス、または上記の任意の好適な組合せであり得る。本明細書で説明される、様々な形態のメモリが、コンピュータ可読記憶媒体の例である。コンピュータ可読記憶媒体のより具体的な例の非網羅的なリストは、ポータブルコンピュータディスク、ハードディスク、RAM、読取り専用メモリ (ROM)、消去可能プログラマブル読取り専用メモリ (EPROM またはフラッシュメモリ)、電子的消去可能プログラマブル読取り専用メモリ (EEPROM)、スタティックランダムアクセスメモリ (SRAM)、ポータブルコンパクトディスク読取り専用メモリ (CD-ROM)、デジタル多用途ディスク (DVD)、メモリスティック、フロッピーディスクなどを含み得る。

10

【0130】

本明細書で定義される「する場合 (if)」という用語は、文脈に応じて、「するとき (when)」または「すると (upon)」または「に 応答して (in response to)」または「に 反応して (responsive to)」を意味する。したがって、「それが決定された場合」または「[述べられた条件またはイベント] が検出された場合」という句は、文脈に応じて、「決定すると」または「決定したことに 応答して」あるいは「[述べられた条件またはイベント] を検出すると」または「[述べられた条件またはイベント] を検出したことに 応答して」または「[述べられた条件またはイベント] を検出したことに 反応して」を意味すると解釈され得る。

【0131】

本明細書で定義される「に 反応して」という用語および上記で説明されたような同様の言い回し、たとえば、「する場合」、「するとき」、または「すると」は、アクションまたはイベントに容易に 応答または 反応することを意味する。 応答または 反応は、自動的に実施される。したがって、第2のアクションが第1のアクション「に 反応して」実施される場合、第1のアクションの発生と第2のアクションの発生との間に因果関係がある。「に 反応して」という用語は、因果関係を指示する。

20

【0132】

本明細書で定義される「一実施形態 (one embodiment)」、「一実施形態 (an embodiment)」、「1つまたは複数の実施形態」、「特定の実施形態」という用語、または同様の言い回しは、実施形態に関して説明される特定の 特徴、構造、または特性が、本開示内で説明される少なくとも1つの実施形態に含まれることを意味する。したがって、本開示全体にわたる、「一実施形態では (in one embodiment)」、「一実施形態では (in an embodiment)」、「1つまたは複数の実施形態では」、「特定の実施形態では」という句、および同様の言い回しの出現は、必ずしもそうとは限らないが、すべて、同じ実施形態を指し得る。「実施形態」および「構成」という用語は、本開示内では互換的に使用される。

30

【0133】

本明細書で定義される「プロセッサ」という用語は、少なくとも1つのハードウェア回路を意味する。ハードウェア回路は、プログラムコード中に含まれている命令を行うように設定され得る。ハードウェア回路は集積回路であり得る。プロセッサの例は、限定はしないが、中央処理ユニット (CPU)、アレイプロセッサ、ベクトルプロセッサ、デジタル信号プロセッサ (DSP)、FPGA、プログラマブル論理アレイ (PLA)、ASIC、プログラマブル論理回路、およびコントローラを含む。

40

【0134】

本明細書で定義される「出力」という用語は、物理メモリ要素、たとえば、デバイスに記憶すること、ディスプレイまたは他の周辺出力デバイスに書き込むこと、別のシステムに送ることまたは送信すること、エクスポートすることなどを意味する。

【0135】

本明細書で定義される「リアルタイム」という用語は、ユーザまたはシステムが、特定のプロセスまたは決定が行われるのに十分に即時であると感じる、あるいは、プロセッサが

50

、何らかの外部プロセスについていくことを可能にする、処理応答性のレベルを意味する。

【0136】

本明細書で定義される「実質的に」という用語は、具陳された特性、パラメータ、または値が正確に達成される必要がないこと、ただし、たとえば、当業者に知られている許容差、測定誤差、測定精度限界、および他のファクタを含む、偏差または変動が、特性が提供することを意図された効果を妨げない量で生じ得ることを意味する。

【0137】

第1の、第2のなどの用語は、様々な要素を説明するために本明細書で使用され得る。これらの用語は、別段に述べられていない限り、または文脈が別段に明確に指示しない限り、ある要素を別の要素と区別するために使用されるにすぎないので、これらの要素はこれらの用語によって限定されるべきでない。

10

【0138】

コンピュータプログラム製品は、プロセッサに本明細書で説明される本発明の構成の態様を行わせるためのコンピュータ可読プログラム命令をその上に有する（1つまたは複数の）コンピュータ可読記憶媒体を含み得る。本開示内では、「プログラムコード」という用語は、「コンピュータ可読プログラム命令」という用語と互換的に使用される。本明細書で説明されるコンピュータ可読プログラム命令は、コンピュータ可読記憶媒体からそれぞれのコンピューティング/処理デバイスに、あるいはネットワーク、たとえば、インターネット、LAN、WANおよび/またはワイヤレスネットワークを介して外部コンピュータまたは外部記憶デバイスにダウンロードされ得る。ネットワークは、銅伝送ケーブル、光伝送ファイバー、ワイヤレス送信、ルータ、ファイアウォール、スイッチ、ゲートウェイコンピュータ、および/またはエッジサーバを含むエッジデバイスを含み得る。各コンピューティング/処理デバイス中のネットワークアダプタカードまたはネットワークインターフェースは、ネットワークからコンピュータ可読プログラム命令を受信し、そのコンピュータ可読プログラム命令を、それぞれのコンピューティング/処理デバイス内のコンピュータ可読記憶媒体に記憶するためにフォワーディングする。

20

【0139】

本明細書で説明される本発明の構成のための動作を行うためのコンピュータ可読プログラム命令は、アセンブラ命令、命令セットアーキテクチャ（ISA）命令、機械命令、機械依存命令、マイクロコード、ファームウェア命令、あるいは、オブジェクト指向プログラミング言語および/または手続き型プログラミング言語を含む1つまたは複数のプログラミング言語の任意の組合せで書き込まれたソースコードまたはオブジェクトコードのいずれかであり得る。コンピュータ可読プログラム命令は、状態セッティングデータを含み得る。コンピュータ可読プログラム命令は、完全にユーザのコンピュータ上で、部分的にユーザのコンピュータ上で、スタンドアロンソフトウェアパッケージとして、部分的にユーザのコンピュータ上でおよび部分的にリモートコンピュータ上で、あるいは完全にリモートコンピュータまたはサーバ上で実行し得る。後者のシナリオでは、リモートコンピュータは、LANまたはWANを含む任意のタイプのネットワークを通してユーザのコンピュータに接続され得るか、あるいは接続は、（たとえば、インターネットサービスプロバイダを使用してインターネットを通して）外部コンピュータに対して行われ得る。いくつかの場合には、たとえば、プログラマブル論理回路、FPGA、またはPLAを含む電子回路が、本明細書で説明される本発明の構成の態様を実施するために、電子回路を個人化するためにコンピュータ可読プログラム命令の状態情報を利用することによって、コンピュータ可読プログラム命令を実行し得る。

30

40

【0140】

本発明の構成のいくつかの態様が、方法、装置（システム）、およびコンピュータプログラム製品のフローチャート例示図および/またはブロック図を参照しながら本明細書で説明された。フローチャート例示図および/またはブロック図の各ブロック、ならびにフローチャート例示図および/またはブロック図中のブロックの組合せが、コンピュータ可読

50

プログラム命令、たとえば、プログラムコードによって実装され得ることを理解されよう。

【0141】

これらのコンピュータ可読プログラム命令は、汎用コンピュータ、専用コンピュータ、または機械を作り出すための他のプログラマブルデータ処理装置のプロセッサに提供され得、その結果、コンピュータまたは他のプログラマブルデータ処理装置のプロセッサを介して実行する命令は、フローチャートおよび/またはブロック図の1つまたは複数のブロックにおいて指定された機能/行為を実装するための手段を作成する。これらのコンピュータ可読プログラム命令はまた、コンピュータ、プログラマブルデータ処理装置、および/または他のデバイスに特定の様式で機能するように指示することができるコンピュータ可読記憶媒体に記憶され得、その結果、命令が記憶されたコンピュータ可読記憶媒体は、フローチャートおよび/またはブロック図の1つまたは複数のブロックにおいて指定された動作の態様を実装する命令を含む製造品を備える。

10

【0142】

コンピュータ可読プログラム命令はまた、コンピュータ実装プロセスを作り出すために、一連の動作をコンピュータ、他のプログラマブルデータ処理装置または他のデバイス上で実施させるように、コンピュータ、他のプログラマブル装置、または他のデバイスにロードされ得、その結果、コンピュータ、他のプログラマブル装置、または他のデバイス上で実行する命令は、フローチャートおよび/またはブロック図の1つまたは複数のブロックにおいて指定された機能/行為を実装する。

20

【0143】

図中のフローチャートおよびブロック図は、本発明の構成の様々な態様によるシステム、方法、およびコンピュータプログラム製品の可能な実装形態のアーキテクチャ、機能性、および動作を示す。この点について、フローチャートまたはブロック図中の各ブロックは、指定された動作を実装するための1つまたは複数の実行可能な命令を備える、命令のモジュール、セグメント、または部分を表し得る。

【0144】

いくつかの代替実装形態では、ブロック中で言及される動作は、図中で言及される順序から外れて行われ得る。たとえば、関与する機能性に応じて、連続して示されている2つのブロックが、実質的に同時に実行され得るか、またはブロックが、時々、逆の順序で実行され得る。他の例では、ブロックは、概して小さい数字から順に実施され得、さらに他の例では、1つまたは複数のブロックは、変動順で実施され得、結果は、記憶され、後続の、または直後にこない他のブロックにおいて利用される。また、ブロック図および/またはフローチャート例示図の各ブロック、ならびにブロック図および/またはフローチャート例示図中のブロックの組合せが、指定された機能または行為を実施するかあるいは専用ハードウェアとコンピュータ命令との組合せを行う専用ハードウェアベースシステムによって実装され得ることに留意されたい。

30

【0145】

以下の特許請求の範囲において見られ得るすべての手段またはステップおよび機能要素の対応する構造、材料、行為、および等価物は、特に主張されるように、他の請求される要素と組み合わせて機能を実施するための任意の構造、材料、または行為を含むことを意図される。

40

【0146】

1つまたは複数の実施形態は、集積回路(IC)を対象とする。本ICは、本ICの複数の算出ユニットによる実行のためのホストプロセッサからのコマンドを記憶するように設定された複数のスロットを有するコマンド待ち行列と、コマンド待ち行列中の複数のスロットに対応する複数のロケーションを有するコマンド要求レジスタとを含む。コマンド要求レジスタは、コマンド待ち行列内に記憶された新しいコマンドを指示する割込みを生成するように設定される。本ICは、コマンド要求レジスタからの割込みに応答して、新しいコマンドを実行するための、複数の算出ユニットからの、アイドルである選択された算

50

出ユニットを決定するように設定されたコントローラと、新しいコマンドを選択された算出ユニットに提供するように設定された算出ユニットダイレクトメモリアクセス回路とを含む。

【0147】

一態様では、新しいコマンドは、新しいコマンドを実行することが可能な複数の算出ユニットのうちの一つまたは複数を指定する第1のリストを含む。

【0148】

別の態様では、コントローラは、複数の算出ユニットのうちどれがアイドルであるかを指定する第2のリストを維持することと、第1のリストを第2のリストと比較することによって、新しいコマンドを実行するための選択された算出ユニットを決定することとを行うように設定される。

10

【0149】

別の態様では、コントローラは、選択された算出ユニットのためのベースアドレスを指定するためにコマンド待ち行列中の新しいコマンドを更新するように設定される。

【0150】

別の態様では、コントローラは、選択された算出ユニットのベースアドレスを第1のリストに上書きするように設定される。

【0151】

別の態様では、コントローラは、新しいコマンドを記憶する、コマンド待ち行列中の複数のスロットのうち選択されたスロットの参照を、算出ユニットダイレクトメモリアクセス回路に提供するように設定される。

20

【0152】

別の態様では、算出ユニットダイレクトメモリアクセス回路は、新しいコマンドからのベースアドレスを使用して、選択されたスロット中の新しいコマンドを選択された算出ユニットに提供するように設定される。

【0153】

別の態様では、コマンド要求レジスタは読取り時クリアレジスタである。

【0154】

別の態様では、算出ユニットダイレクトメモリアクセス回路は、新しいコマンドを提供したことに応答して、選択された算出ユニットの動作を開始するようにさらに設定される。

30

【0155】

一つまたは複数の実施形態は、ICを対象とする。本ICは、複数の算出ユニットのうち選択された算出ユニットから、選択された算出ユニットが処理し終わったことを指示する割込みを受信したことに応答して、選択された算出ユニットが処理し終わったという信号を出すように設定された算出ユニット割込み回路と、算出ユニット割込み回路からの信号に応答して割込みを生成するように設定された算出ユニット完了レジスタとを含む。本ICは、算出ユニット完了レジスタからの割込みに応答して、ホストプロセッサからオフロードされた複数のコマンドのうち選択されたコマンドが、選択された算出ユニットによって実行されたと決定し、選択されたコマンドを出力するように設定されたコントローラと、複数のコマンドのうち選択されたコマンドが処理し終わったことを指示する、ホストプロセッサへの割込みを生成するように設定されたコマンド完了レジスタとを含む。

40

【0156】

一態様では、算出ユニット完了レジスタは、複数の算出ユニットに対応する複数のロケーションを含み、算出ユニット割込み回路は、選択された算出ユニットに対応する、複数のロケーションのうち選択されたロケーションに書き込むことによって、選択された算出ユニットが処理し終わったことを算出ユニット完了レジスタに通知する。

【0157】

別の態様では、コントローラは、複数のコマンドと、複数のコマンドを実行する、複数の算出ユニットのうち対応する算出ユニットとのリストを維持するように設定される。

【0158】

50

別の態様では、算出ユニット完了レジスタは読取り時クリアレジスタであり得る。

【0159】

別の態様では、コマンド完了レジスタは読取り時クリアレジスタであり得る。

【0160】

別の態様では、算出ユニット割込み回路は、ホストプロセッサからのさらなるコマンドを実行するために、選択された算出ユニットをリームするように設定される。

【0161】

1つまたは複数の実施形態は、方法を対象とする。本方法は、コマンド待ち行列内の新しいコマンドの利用可能性を指示する、ホストプロセッサからの通知に応答して、コントローラへの割込みを生成することと、コントローラが、割込みに応答して、新しいコマンドを実行するための、複数の算出ユニットからの、アイドルである選択された算出ユニットを決定することと、コントローラが、選択された算出ユニットとコマンド待ち行列中の新しいコマンドとを算出ユニットダイレクトメモリアクセス回路に通知することとを含む。本方法は、算出ユニットダイレクトメモリアクセス回路を使用して、新しいコマンドを選択された算出ユニットに提供することをも含む。

10

【0162】

一態様では、本方法は、実行の完了を指示する、選択された算出ユニットからの割込みを受信したことに応答して、選択された算出ユニットが実行を完了したことを指示する、コントローラへの割込みを生成することと、コントローラを使用して、新しいコマンドが、選択された算出ユニットに割り当てられたと決定することと、新しいコマンドが実行を完了したことを指示する、ホストプロセッサへの割込みを生成することとを含む。

20

【0163】

別の態様では、選択された算出ユニットとコマンド待ち行列中の新しいコマンドとを算出ユニットダイレクトメモリアクセス回路に通知することは、新しいコマンドを記憶する、コマンド待ち行列中の複数のスロットのうちの選択されたスロットに、選択された算出ユニットのベースアドレスを書き込むことと、複数のスロットのうちの選択されたスロットの参照を、算出ユニットダイレクトメモリアクセス回路に提供することとを含む。

【0164】

別の態様では、複数の算出ユニットからの選択された算出ユニットを決定することは、新しいコマンド内から、新しいコマンドを実行することが可能な複数の算出ユニットのうちの1つまたは複数を選択する第1のリストを読み取ることと、第1のリストを、アイドルである、複数の算出ユニットのうちの算出ユニットを選択する第2のリストと比較して、共通算出ユニットを選択された算出ユニットとして決定することとを含む。

30

【0165】

別の態様では、選択された算出ユニットとコマンド待ち行列中の新しいコマンドとを算出ユニットダイレクトメモリアクセス回路に通知することは、選択された算出ユニットのベースアドレスを新しいコマンドの第1のリストに上書きすることとあって、新しいコマンドが、コマンド待ち行列中の複数のスロットのうちの選択されたスロットに記憶される、第1のリストに上書きすることと、選択されたスロットの参照を算出ユニットダイレクトメモリアクセス回路に提供することとを含む。

40

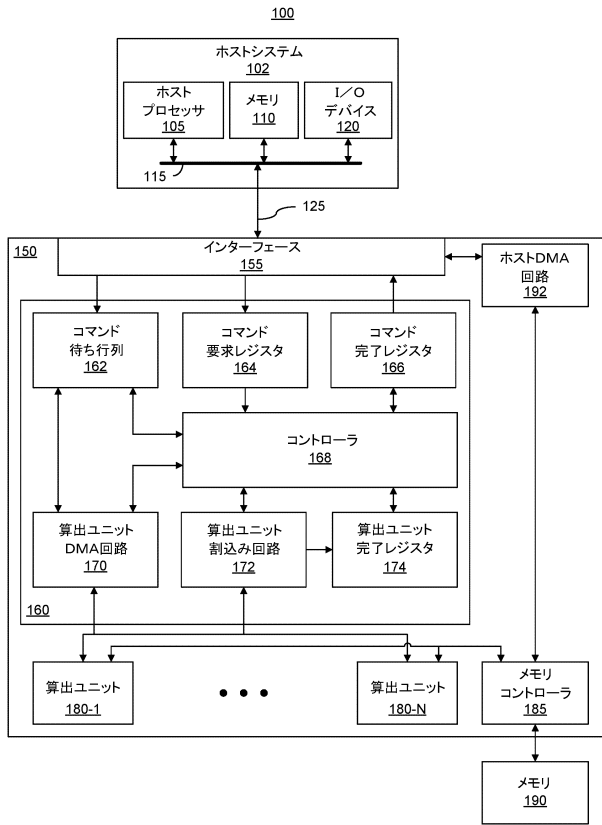
【0166】

本明細書で提供される本発明の構成の説明は、例示のためであり、網羅的なものでも、開示される形式および例に限定されるものでもない。本明細書で使用される専門用語は、本発明の構成の原理、実際の適用例、または市場で見られる技術に対する技術的改善を説明するために、および/あるいは、他の当業者が本明細書で開示される本発明の構成を理解することを可能にするために選定された。説明される本発明の構成の範囲および趣旨から逸脱することなく、修正および変形が当業者に明らかになり得る。したがって、そのような特徴および実装形態の範囲を指示するものとして、上記の開示に対してではなく、以下の特許請求の範囲に対して参照が行われるべきである。

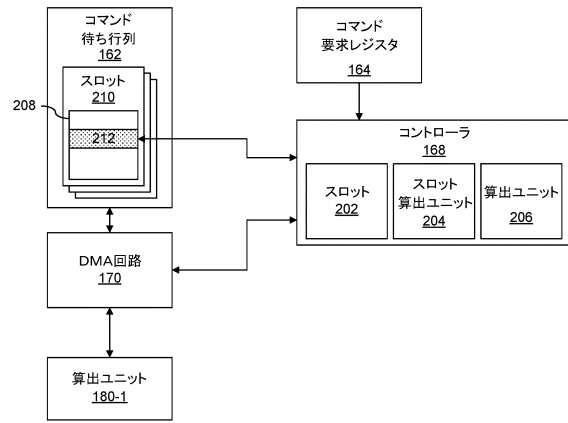
50

【 図 面 】

【 図 1 】



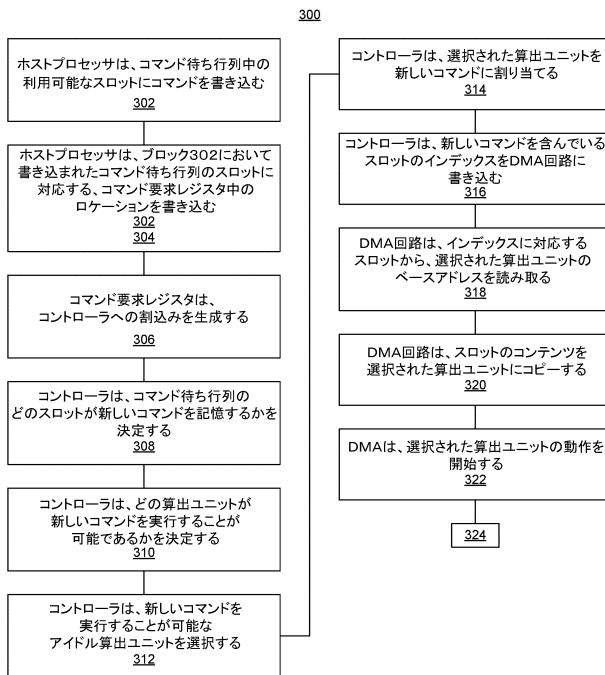
【 図 2 】



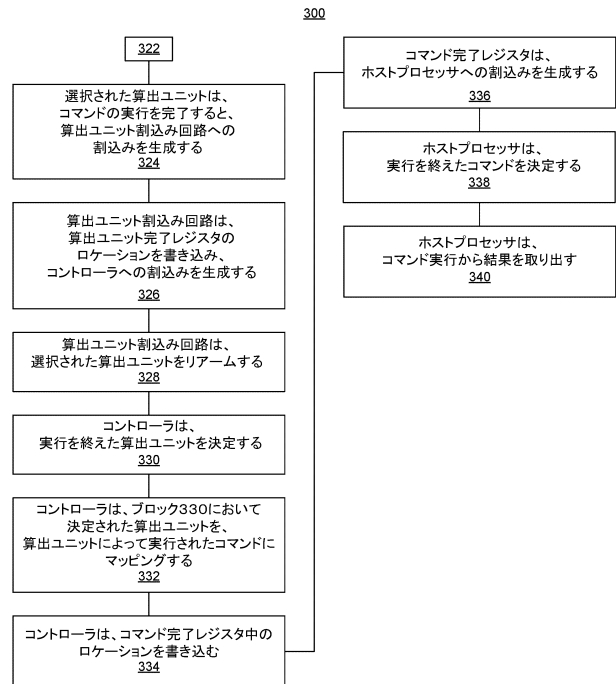
10

20

【 図 3 A 】



【 図 3 B 】

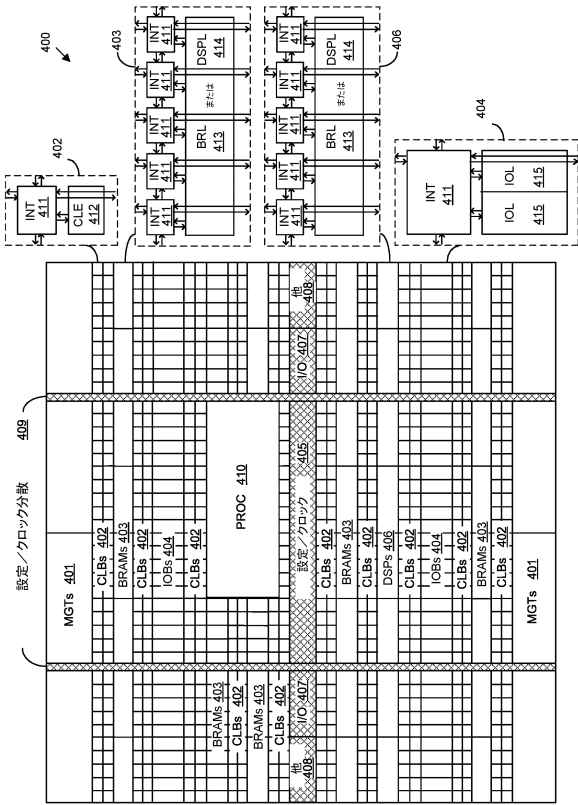


30

40

50

【 図 4 】



10

20

30

40

50

【 国際調査報告 】

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2019/060744

A. CLASSIFICATION OF SUBJECT MATTER INV. G06F9/50 G06F9/54 ADD.		
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) G06F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal, WPI Data		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2010/325638 A1 (NISHIMAKI HISASHI [JP]) 23 December 2010 (2010-12-23) paragraph [0006] - paragraphs [0011], [0041]	1-9, 15
Y	----- US 2018/308197 A1 (VEMBU BALAJI [US] ET AL) 25 October 2018 (2018-10-25) paragraph [0079]	1-9, 15
A	----- US 9 547 535 B1 (WILT NICHOLAS PATRICK [US]) 17 January 2017 (2017-01-17) the whole document -----	1-9, 15
<input type="checkbox"/> Further documents are listed in the continuation of Box C.		<input checked="" type="checkbox"/> 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		*T* 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
E earlier application or patent but published on or after the international filing date		*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
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)		*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
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		*Z* document member of the same patent family
Date of the actual completion of the international search 28 January 2020		Date of mailing of the international search report 06/04/2020
Name and mailing address of the ISA/ European Patent Office, P.O. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Mühlenbrock, Martin

3

Form PCT/ISA/210 (second sheet) (April 2005)

10

20

30

40

50

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2019/060744

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

- 1. Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

- 2. Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

- 3. Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

10

20

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

- 1. As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

- 2. As all searchable claims could be searched without effort justifying an additional fees, this Authority did not invite payment of additional fees.

- 3. As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

- 4. No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:
1-9, 15

30

40

Remark on Protest

- The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- No protest accompanied the payment of additional search fees.

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

10

1. claims: 1-9, 15

An integrated circuit comprising a command queue, a command request register, a controller configured to determine a selected compute unit that is idle from the plurality of compute units to execute a new command, and a compute unit direct memory access circuit (reducing the delay between the time that the compute unit becomes available and the time that the compute unit begins execution of the command, thereby increasing the speed of the heterogeneous computing platform and reducing latency).

2. claims: 10-14

20

An integrated circuit comprising a compute unit interrupt circuit, a compute unit completion register, a controller configured to determine a selected command has finished processing, and a command completion register (ascertaining the status of commands and the particular commands that have finished execution, thereby allowing the host processor to operate at the command level as opposed to managing each of the compute unit individually).

30

40

50

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2019/060744

Patent document cited in search report	Publication date	Patent family member(s)	Publication date	
US 2010325638	A1	23-12-2010	CN 101930380 A	29-12-2010
			JP 5521403 B2	11-06-2014
			JP 2011008340 A	13-01-2011
			US 2010325638 A1	23-12-2010

US 2018308197	A1	25-10-2018	US 2018308197 A1	25-10-2018
			US 2019251652 A1	15-08-2019

US 9547535	B1	17-01-2017	NONE	

10

20

30

40

50

フロントページの続き

MK,MT,NL,NO,PL,PT,RO,RS,SE,SI,SK,SM,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,KM,ML,MR,N
E,SN,TD,TG),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,HN,HR,HU,ID,IL,IN,IR,IS,JO,JP,KE,K
G,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,N
I,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

2 1 0 0

(72)発明者 コッチェル, エラリー

アメリカ合衆国 カリフォルニア 9 5 1 2 4 , サン ノゼ, ロジック ドライブ 2 1 0 0

F ターム (参考) 5B013 DD03

5B061 DD01

5B062 DD04 DD08 DD09 JJ03