(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7523964号
(P7523964)

(45)発行日　令和6年7月29日(2024.7.29)

(24)登録日　令和6年7月19日(2024.7.19)

(51)国際特許分類
Ｇ０６Ｆ　8/41 (2018.01)

ＦＩ
Ｇ０６Ｆ　8/41　１７０

請求項の数　26　外国語出願　（全43頁）

(21)出願番号　特願2020-109665(P2020-109665)
(22)出願日　令和2年6月25日(2020.6.25)
(65)公開番号　特開2021-34023(P2021-34023A)
(43)公開日　令和3年3月1日(2021.3.1)
審査請求日　令和5年6月21日(2023.6.21)
(31)優先権主張番号　16/541,979
(32)優先日　令和1年8月15日(2019.8.15)
(33)優先権主張国・地域又は機関
米国(US)

(73)特許権者　593096712
インテル コーポレイション
アメリカ合衆国 ９５０５４ カリフォルニア州 サンタ クララ ミッション カレッジ ブールバード ２２００
(74)代理人　100107766
弁理士　伊東 忠重
(74)代理人　100070150
弁理士　伊東 忠彦
(74)代理人　100135079
弁理士　宮崎 修
(72)発明者　ミカエル ベハール
アメリカ合衆国 ９５０５４ カリフォルニア州 サンタ クララ ミッション カレッジ ブールバード ２２００ インテル
最終頁に続く

(54)【発明の名称】　アクセラレータにおいてヘテロジニアスコンポーネントを設定する方法及び装置

(57)【特許請求の範囲】
【請求項１】
アクセラレータにおいてヘテロジニアスコンポーネントを設定する装置であって、
作業負荷において作業負荷ノードを識別し、該作業負荷ノードについてセレクタを生成するグラフコンパイラと、
計算構築ブロックの入力条件及び出力条件を識別する前記セレクタと
を有し、
前記グラフコンパイラは、前記識別された入力条件及び出力条件を前記セレクタから取得することに応答して、前記作業負荷ノードを前記計算構築ブロックにマッピングする、
装置。
【請求項２】
前記グラフコンパイラは、前記作業負荷において第２作業負荷ノードを識別し、該第２作業負荷ノードについて第２セレクタを生成する、
請求項１に記載の装置。
【請求項３】
前記第２セレクタは、カーネルの第２入力条件及び第２出力条件を識別する、
請求項２に記載の装置。
【請求項４】
前記作業負荷は、前記グラフコンパイラによって取得された前記作業負荷ノードを含むグラフである、

請求項１乃至３のうちいずれか一項に記載の装置。
【請求項５】
前記入力条件は、前記計算構築ブロックの入力要件に対応し、前記出力条件は、前記計算構築ブロックの実行の結果に対応する、
請求項１乃至３のうちいずれか一項に記載の装置。
【請求項６】
前記グラフコンパイラは、前記作業負荷ノードを前記計算構築ブロックにマッピングすることに応答して、実行ファイルを生成する、
請求項１乃至３のうちいずれか一項に記載の装置。
【請求項７】
前記グラフコンパイラは、前記計算構築ブロックに対する前記作業負荷ノードのマッピングを可能にするよう、前記識別された入力条件及び出力条件に基づいて、前記作業負荷ノードと前記計算構築ブロックとの間の変換レイヤを形成するプラグインを更に含む、
請求項１乃至３のうちいずれか一項に記載の装置。
【請求項８】
実行される場合に、少なくとも１つのプロセッサに、少なくとも
作業負荷において作業負荷ノードを識別することと
前記作業負荷ノードについて、該作業負荷ノードを実行するための計算構築ブロックに関連したセレクタを生成することと、
前記計算構築ブロックの入力条件及び出力条件を識別することと、
前記識別された入力条件及び出力条件を取得することに応答して、前記作業負荷ノードを前記計算構築ブロックにマッピングすることと
を実行させる命令を有するプログラム。
【請求項９】
前記命令は、実行される場合に、前記少なくとも１つのプロセッサに、更に
前記作業負荷において第２作業負荷ノードを識別することと、
前記第２作業負荷ノードについて第２セレクタを生成することと
を実行させる、
請求項８に記載のプログラム。
【請求項１０】
前記命令は、実行される場合に、前記少なくとも１つのプロセッサに、更に
カーネルの第２入力条件及び第２出力条件を識別することを実行させる、
請求項８又は９のいずれかに記載のプログラム。
【請求項１１】
前記作業負荷は、前記作業負荷ノードを含むグラフである、
請求項８乃至１０のうちいずれか一項に記載のプログラム。
【請求項１２】
前記入力条件は、前記計算構築ブロックの入力要件に対応し、前記出力条件は、前記計算構築ブロックの実行の結果に対応する、
請求項８乃至１０のうちいずれか一項に記載のプログラム。
【請求項１３】
前記命令は、実行される場合に、前記少なくとも１つのプロセッサに、更に
前記作業負荷ノードを前記計算構築ブロックにマッピングすることに応答して実行ファイルを生成することを実行させる、
請求項８乃至１０のうちいずれか一項に記載のプログラム。
【請求項１４】
前記命令は、実行される場合に、前記少なくとも１つのプロセッサに、更に
前記計算構築ブロックに対する前記作業負荷ノードのマッピングを可能にするよう、前記識別された入力条件及び出力条件に基づいて、前記作業負荷ノードと前記計算構築ブロックとの間の変換レイヤを形成することを実行させる、

請求項８乃至１０のうちいずれか一項に記載のプログラム。
【請求項１５】
作業負荷において作業負荷ノードを識別し、該作業負荷ノードについて、該作業負荷ノードを実行するための計算構築ブロックに関連した選択手段を生成するコンパイル手段と、
前記計算構築ブロックの入力条件及び出力条件を識別する前記選択手段と
を有し、
前記コンパイル手段は更に、前記識別された入力条件及び出力条件を取得することに応答して、前記作業負荷ノードを前記計算構築ブロックにマッピングする、
装置。
【請求項１６】
前記コンパイル手段は更に、前記作業負荷において第２作業負荷ノードを識別し、該第２作業負荷ノードについて第２選択手段を生成する、
請求項１５に記載の装置。
【請求項１７】
前記第２選択手段は更に、カーネルの第２入力条件及び第２出力条件を識別する、
請求項１６に記載の装置。
【請求項１８】
前記作業負荷は、前記作業負荷ノードを含むグラフである、
請求項１５乃至１７のうちいずれか一項に記載の装置。
【請求項１９】
前記入力条件は、前記計算構築ブロックの入力要件に対応し、前記出力条件は、前記計算構築ブロックの実行の結果に対応する、
請求項１５乃至１７のうちいずれか一項に記載の装置。
【請求項２０】
アクセラレータにおいてヘテロジニアスコンポーネントを設定する方法であって、
作業負荷において作業負荷ノードを識別することと、
前記作業負荷ノードについて、該作業負荷ノードを実行するための計算構築ブロックに関連したセレクタを生成することと、
前記計算構築ブロックの入力条件及び出力条件を識別することと、
前記識別された入力条件及び出力条件を取得することに応答して前記作業負荷ノードを前記計算構築ブロックにマッピングすることと
を有する方法。
【請求項２１】
前記作業負荷において第２作業負荷ノードを識別することと、
前記第２作業負荷ノードについて第２セレクタを生成することと
を更に含む、請求項２０に記載の方法。
【請求項２２】
カーネルの第２入力条件及び第２出力条件を識別することを更に含む、
請求項２０又は２１のいずれかに記載の方法。
【請求項２３】
前記作業負荷は、前記作業負荷ノードを含むグラフである、
請求項２０乃至２２のうちいずれか一項に記載の方法。
【請求項２４】
前記作業負荷ノードを前記計算構築ブロックにマッピングすることに応答して実行ファイルを生成することを更に含む
請求項２０乃至２２のうちいずれか一項に記載の方法。
【請求項２５】
前記計算構築ブロックに対する前記作業負荷ノードのマッピングを可能にするよう、前記識別された入力条件及び出力条件に基づいて、前記作業負荷ノードと前記計算構築ブロックとの間の変換レイヤを形成することを更に含む、

請求項２０乃至２２のうちいずれか一項に記載の方法。
【請求項２６】
請求項８乃至１４のうちいずれか一項に記載のプログラムを記憶している少なくとも1つの非一時的なコンピュータ可読記憶媒体。
【発明の詳細な説明】
【技術分野】
【０００１】
本願は、概してプロセッシングに、より具体的には、アクセラレータにおいてヘテロジニアスコンポーネントを設定する方法及び装置に関係がある。
【背景技術】
【０００２】
コンピュータハードウェア製造業者は、コンピュータプラットフォームの様々なコンポーネントで使用されるハードウェアコンポーネントを開発している。例えば、コンピュータハードウェア製造業者は、マザーボード、マザーボード用のチップセット、中央演算処理装置（Central Processing Units，ＣＰＵ）、ハードディスクドライブ（Hard Disk Drives，ＨＤＤ）、ソリッドステートドライブ（Solid State Drives，ＳＳＤ）、及び他のコンピュータコンポーネントを開発している。その上、コンピュータハードウェア製造業者は、作業負荷の処理を加速させる、アクセラレータとして知られている処理要素を開発している。例えば、アクセラレータは、ＣＰＵ、グラフィクスプロセッシングユニット（Graphics Processing Units，ＧＰＵ）、ビジョンプロセッシングユニット（Vision Processing Units，ＶＰＵ）、及び／又はフィールドプログラマブルゲートアレイ（Field Programmable Gate Arrays，ＦＰＧＡ）であることができる。
【図面の簡単な説明】
【０００３】
【図１】アクセラレータにおいてヘテロジニアスコンポーネントを設定するコンピュータシステムの例を表すブロック図である。
【図２】例となるグラフコンパイラ及び１つ以上の例となるセレクタを含むコンピュータシステムの例を表すブロック図である。
【図３】図２の１つ以上のセレクタの中のセレクタの例を表すブロック図の例である。
【図４】図２のグラフコンパイラを表すブロック図の例である。
【図５】例となる第１ＣＢＢ及び例となる第２ＣＢＢを用いて実行される作業負荷を表すパイプラインの例の図式的な説明である。
【図６】図２の実行ファイルを生成するように図２、３及び／又は４のグラフコンパイラ、セレクタ及び／又は１つ以上のセレクタを実装するよう実行され得るプロセスを表すフローチャートである。
【図７】図２の実行ファイルの実行を助けるように図２のクレジットマネージャ及び／又はコンフィグレーションコントローラを実装するよう実行され得るプロセスを表すフローチャートである。
【図８】図２、３及び／又は４の例となるグラフコンパイラ、例となる１つ以上のセレクタ、例となるセレクタ、及び／又はアクセラレータを実装するように図６及び／又は７の命令を実行するよう構造化されたプロセッサプラットフォームの例のブロック図である。
【発明を実施するための形態】
【０００４】
図は、実寸通りではない。一般に、同じ参照番号は、同じか又は類似した部分を参照するために図面及び添付の明細書にわたって使用されることになる。接続参照（例えば、取り付けられる（attached）、結合される（coupled）、接続される（connected）、及びつなぎ合わされる（joined））は、広く解釈されるべきであり、別段示されない限りは、一群の要素間にある中間部材、及び要素間の相対運動を含み得る。そのようなものとして、接続参照は、必ずしも、２つの要素が直接接続されて互いに固定関係にあることを暗示しない。

【０００５】
「第１」、「第２」、「第３」などの記述子は、別々に言及され得る複数の要素又はコンポーネントを識別する場合に本願では使用される。それらの使用文脈に基づいて別段特定又は理解されない限り、そのような記述子は、優先度、物理的な順序若しくはリスト内の配置、又は時間的な順序の意味を有することは意図されず、単に、開示されている例を理解することを容易にするために別々に複数の要素又はコンポーネントに言及するラベルとして使用される。いくつかの例で、「第１」との記述子は、詳細な説明において、ある要素に言及するために使用されることがあり、一方、同じ要素は、「第２」又は「第３」といった別の記述子により特許請求の範囲では言及されることがある。そのような場合に、かような記述子は、単に、複数の要素又はコンポーネントを参照することを簡単にするために使用されることが理解されるべきである。

【０００６】
多くのコンピュータハードウェア製造業者は、作業負荷の処理を加速させるための、アクセラレータとして知られている処理要素を開発している。例えば、アクセラレータは、ＣＰＵ、ＧＰＵ、ＶＰＵ、及び／又はＦＰＧＡであることができる。更に、アクセラレータは、あらゆるタイプの作業負荷を処理可能である一方で、特定のタイプの作業負荷を最適化するよう設計される。例えば、ＣＰＵ及びＦＰＧＡは、より一般的な処理を扱うよう設計され得るが、ＧＰＵは、映像、ゲーム、及び／又は他の物理及び数学に基づく計算を改善するよう設計され得、ＶＰＵは、マシンビジョンタスクの処理を改善するよう設計され得る。

【０００７】
その上、いくつかのアクセラレータは、人工知能（Artificial Intelligence，ＡＩ）アプリケーションの処理を改善するよう特に設計される。ＶＰＵは、特定のタイプのＡＩアクセラレータであり、一方、多くの異なるＡＩアクセラレータが使用可能である。実際に、多くのＡＩアクセラレータは、特定用途向け集積回路（Application Specific Integrated Circuits，ＡＳＩＣ）によって実装され得る。かようなＡＳＩＣに基づくＡＩアクセラレータは、機械学習（Machine Learning，ＭＬ）、ディープラーニング（Deep Learning，ＤＬ）、及び／又はサポートベクトルマシン（Support Vector Machines，ＳＶＭ）、ニューラルネットワーク（Neural Networks，ＮＮ）、再帰型ニューラルネットワーク（Recurrent Neural Networks，ＲＮＮ）、畳み込み型ニューラルネットワーク（Convolutional Neural Networks，ＣＮＮ）、長短期メモリ（Long Short Term Memory，ＬＳＴＭ）、ゲート回帰型ユニット（Gate Recurrent Units，ＧＲＵ）などを含む他の人工マシン駆動型ロジックといった特定のタイプのＡＩに関するタスクの処理を改善するよう設計され得る。

【０００８】
コンピュータハードウェア製造業者はまた、１よりも多いタイプの処理要素を含むヘテロジニアスシステムを開発している。例えば、コンピュータハードウェア製造業者は、ＣＰＵのような汎用の処理要素を、ＦＰＧＡなどの汎用アクセラレータ及び／又はＧＰＵ、ＶＰＵ及び／又は他のＡＩアクセラレータなどのよりカスタマイズされたアクセラレータのいずれかと組み合わせることがある。かようなヘテロジニアスシステムは、システム・オン・チップ（Systems on a Chip，ＳｏＣ）として実装され得る。

【０００９】
開発者がヘテロジニアスシステムで関数、アルゴリズム、プログラム、アプリケーション、及び／又は他のコードを実行したいと望むとき、開発者及び／又はソフトウェアは、コンパイル時点で関数、アルゴリズム、プログラム、アプリケーション、及び／又は他のコードのためのスケジュール（例えば、グラフ）を生成する。スケジュールが生成されると、スケジュールは、（事前（Ahead of Time）又は実行時（Just in Time）パラダイムのいずれかのための）実行ファイルを生成するよう関数、アルゴリズム、プログラム、アプリケーション、及び／又は他のコード仕様と組み合わされる。更に、関数、アルゴリズム、プログラム、アプリケーション、及び／又は他のコードと組み合わされたスケジ

ュールは、ノードを含むグラフとして表現されることがあり、グラフは、作業負荷（workload）を表し、各ノード（例えば、作業負荷ノード）は、その作業負荷の特定のタスクを表す。更には、グラフ内の異なるノード間の接続は、特定の作業負荷ノードが実行されるために必要とされるデータ入力及び／又は出力を表し、グラフの頂点は、グラフの作業負荷ノード間のデータ依存性を表す。

【００１０】

スケジュール（例えば、グラフ）をコンパイルする一般的な実施は、スケジュール（例えば、グラフ）を受け取って、作業負荷の様々な作業負荷ノードを、アクセラレータ内にある様々な計算構築ブロック（Compute Building Blocks，ＣＢＢ）に割り当てるグラフコンパイラを含む。ヘテロジニアスシステムでは、グラフコンパイラは、夫々の独立したＣＢＢと通じるよう個別に構成される。例えば、グラフコンパイラが作業負荷ノードをＤＳＰ及び／又はＤＳＰに位置するカーネルに割り当て及び／又は別なふうに送るために、かようなグラフコンパイラは、ＤＳＰが含む入力及び出力条件（例えば、入力のタイプ及び出力のタイプ）を知っている必要がある。様々な計算構築ブロック（ＣＢＢ）を含むヘテロジニアスシステム、あるいは、様々なＣＢＢで実行されるべき様々な作業負荷ノードを受け取り及び／又は別なふうに取得するヘテロジニアスシステムでは、単一のグラフコンパイラを使用する実行は計算的に集中することになる。更には、ランタイム中のＣＢＢの間の通信及び制御は、システムのヘテロジニアス性質によりしばしば実際的でない。同様に、ＣＢＢの間のデータ交換同期化は、しばしば計算的に負荷が重い。

【００１１】

その上、ヘテロジニアスシステム内に位置する様々なカーネルに対する作業負荷の様々な作業負荷ノードの割り当ては、同様に、グラフコンパイラが夫々の独立したカーネルと通じるよう個別に構成されることを必要とする。加えて、カーネルはしばしば、ユーザによって生成後にアクセラレータにロードされ、そのようなものとして、グラフコンパイラの再設定を必要とする。例えば、グラフコンパイラは、グラフコンパイラの初期設定後にアクセラレータに生成及び／又は別なふうにロードされたカーネルと通じること（例えば、作業負荷ノードを送ること）ができないことがある。

【００１２】

本願で開示される例は、アクセラレータにおいてヘテロジニアスコンポーネントを設定する方法及び装置を含む。本願で開示される例は、如何なる任意のスケジュール及び／又はグラフも用いて動作可能なアクセラレータを含む。例えば、本願で開示される例は、任意のスケジュール及び／又はグラフを効率的に理解しアクセラレータにマッピングすることができるグラフコンパイラを含む。本願で開示されるかような例の動作は、以下で、更に詳細に説明される。

【００１３】

本願で開示される例は、コンパイル時間中の様々なＣＢＢの抽象化及び／又は一般化を含む。本願で開示される例は、ＣＢＢのための共通識別を採用することを含む。例えば、各ＣＢＢは、ヘテロジニアスであろうとなかろうと、そのＣＢＢと相互作用するよう各々のセレクタを生成することによって識別され得る。かような例では、セレクタは、作業負荷内の作業負荷ノードを解析することに応答して生成される。各作業負荷ノードはしばしば、実行するために使用されるＣＢＢのタイプに関する詳細を含むので、セレクタは、かようなＣＢＢと相互作用するように作られ得る。本願で開示される例では、セレクタは、かようなＣＢＢの入力及び／又は出力条件を決定する。セレクタは、作業負荷及び作業負荷内のＣＢＢと通じる（例えば、作業負荷ドメイン及びＣＢＢドメインと通じる）ことが可能な個別エンティティであるよう作られ得る。結果として、グラフコンパイラは、作業負荷ドメインにおいて動作を可能にするプラグインを含む。本願で使用されるように、作業負荷ドメインは、作業負荷に基づく抽象化及び／又は一般のレベルを指す。同様に、本願で使用されるように、ＣＢＢドメインは、ＣＢＢに基づく、作業負荷ドメインよりもより詳細な抽象化及び／又は一般化のレベルを指す。本願で開示されるかような例は、システムに固有であるか、又は後の時点でシステムに含まれるかのいずれかであるＣＢＢの抽

象化を可能にする。

【００１４】

本願で開示される例は、入力及び出力バッファとして識別されるバッファを利用する。本願で開示されるかような例では、プロデューサ（例えば、他のＣＢＢによって使用されるデータを生成及び／又は別なふうに書き込むＣＢＢ）又はコンシューマ（例えば、他のＣＢＢによって生成されたデータを取得及び／又は別なふうに読み出すＣＢＢ）のいずれかとして振る舞うＣＢＢのパイプラインは、バッファを用いて実装される。プロデューサ又はコンシューマのいずれかとして振る舞うＣＢＢのパイプラインを実装することによって、グラフコンパイラは、作業負荷（例えば、グラフ）の作業負荷ノード（例えば、タスク）を各ＣＢＢにサイジング及び／又は割り当てるときに、ジェネリックヒューリスティクス（generic heuristics）（例えば、課題を解決するために設計された技術、作業負荷ドメインで動作する経験則）を使用することができる。本願で開示されるいくつかの例で、グラフコンパイラは、作業負荷ノード（例えば，タスク）を実行するようバッファのスロットのサイズ及び数（例えば、ストレージサイズ）を含むことができる情報を提供し得る。このようにして、例となるクレジットマネージャは、バッファ内のｎ個のスロットに基づいてｎ個のクレジットを生成し得る。ｎ個のクレジットは、従って、ＣＢＢが書き込み又は読み出し可能であるメモリ内の利用可能なｎ個のスペースを示す。クレジットジェネレータは、ｎ個のクレジットを、パッケージ化して、コンフィグレーションコントローラによって決定された対応するプロデューサ及び／又はコンシューマへ送信し、例となるファブリック（例えば、制御及び設定ファブリック）にわたって通信されるように、コンフィグレーションコントローラへ供給する。

【００１５】

更には、本願で開示される例は、グラフコンパイラに対してＣＢＢの標準表現を実装することを含む。本願で開示される例は、作業負荷内の各作業負荷ノードのために設定されたセレクタを含む。セレクタは、対応する作業負荷ノードによって識別されるＣＢＢの標準の入力及び／又は出力条件を識別するよう構成される。更に、かようなセレクタは、それらの入力及び／又は出力条件によって特定される抽象化されたデバイスのリストをグラフコンパイラへ供給するよう構成される。本願で開示される可能な例では、グラフコンパイラは、様々なＣＢＢに対する作業負荷ノード（例えば、タスク）のマッピングを可能にするよう、作業負荷（例えば、グラフ）内の作業負荷ノード（例えば、タスク）と様々なＣＢＢとの間の変換レイヤ（例えば、ＣＢＢドメインと作業負荷ドメインとの間の変換レイヤ）を形成することができるプラグインを含む。加えて、本願で開示されるいくつかの例では、セレクタは、関連するＣＢＢの具体的な要件をグラフコンパイラへ返してもよい。例えば、セレクタは、かようなＣＢＢが動作するためにメモリ割り当ての一定の割合を必要とすることを、グラフコンパイラに伝えてよい。

【００１６】

ランタイム中に、本願で開示される例は、ＣＢＢ間の通信を可能にするようＣＢＢを構成するために使用される共通アーキテクチャを含む。本願で開示される例は、グラフコンパイラによって生成されたパイプラインとともにクレジットのシステムを利用する。かようなシステムは、グラフコンパイラが作業負荷（例えば、グラフ）からの作業負荷ノード（例えば、タスク）をプロデューサ及びコンシューマパイプラインにマッピングし、ＣＢＢ間の通信を可能にすることを可能にする。初期プロデューサ（データを書き込むことを指示する作業負荷ノードを実行するＣＢＢ）として振る舞うＣＢＢが作業負荷ノードの実行を完了すると、クレジットは、次のＣＢＢではなく、そのＣＢＢから見た原点へ返送される。かような原点は、本願で開示される例ではクレジットマネージャであってよい。

【００１７】

図１は、アクセラレータにおいてヘテロジニアスコンポーネントを設定するコンピュータシステム１００を例示するブロック図である。図１の例では、コンピュータシステム１００は、例となるシステムメモリ１０２と、例となるヘテロジニアスシステム１０４とを含む。例となるヘテロジニアスシステム１０４は、例となるホストプロセッサ１０６と、

例となる第１通信バス１０８と、例となる第１アクセラレータ１１０ａと、例となる第２アクセラレータ１１０ｂと、例となる第３アクセラレータ１１０ｃとを含む。例となる第１アクセラレータ１１０ａ、例となる第２アクセラレータ１１０ｂ、及び例となる第３アクセラレータ１１０ｃの夫々は、各々のアクセラレータの動作に取って一般的及び／又は特有である様々なＣＢＢを含む。

【００１８】

図１の例では、システムメモリ１０２は、例えば、フラッシュメモリ、磁気媒体、光学媒体などのような、データを記憶するための如何なるデバイスによっても実装されてよい。更に、例となるシステムメモリ１０２に記憶されるデータは、例えば、バイナリデータ、コンマ区切りデータ、タブ区切りデータ、構造化されたクエリ言語（Structured Query Language，ＳＱＬ）構造、などのような、如何なるデータフォーマットにあってもよい。システムメモリ１０２は、ヘテロジニアスシステム１０４へ結合されている。図１で、システムメモリ１０２は、ホストプロセッサ１０６、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、及び第３アクセラレータ１１０ｃのうちの少なくとも１つの間の共有ストレージである。図１の例では、システムメモリ１０２は、コンピュータシステム１００のローカルの物理ストレージであるが、他の例では、システムメモリ１０２は、コンピュータシステム１００の外にあっても、かつ／あるいは、コンピュータシステム１００に対して別なふうに遠隔にあってもよい。更なる例では、システムメモリ１０２は、仮想ストレージであってもよい。図１の例では、システムメモリ１０２は、不揮発性メモリ（例えば、リードオンリーメモリ（Read Only Memory，ＲＯＭ）、プログラム可能（Programmable）ＲＯＭ（ＰＲＯＭ）、消去可能（Erasable）ＰＲＯＭ（ＥＰＲＯＭ）、電気的消去可能な（Electrically Erasable）ＰＲＯＭ（ＥＥＰＲＯＭ）、など）である。他の例では、システムメモリ１０２は、不揮発性基本入出力システム（Basic Input/Output System，ＢＩＯＳ）又はフラッシュストレージであってよい。更なる例では、システムメモリ１０２は、揮発性メモリであってもよい。

【００１９】

図１で、ヘテロジニアスシステム１０４は、システムメモリ１０２経結合されている。図１の例では、ヘテロジニアスシステム１０４は、ホストプロセッサ１０６及び／又は第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、若しくは第３アクセラレータ１１０ｃのうちの１つ以上で作業負荷を実行することによって、作業負荷を処理する。図１で、ヘテロジニアスシステム１０４は、システム・オン・チップ（ＳｏＣ）である。代替的に、ヘテロジニアスシステム１０４は、如何なる他のタイプのコンピュータ又はハードウェアシステムであってもよい。

【００２０】

図１の例では、ホストプロセッサ１０６は、コンピュータ及び／又はコンピュータデバイス（例えば、コンピュータシステム１００）に関連した動作を実行し及び／又は別なふうにその動作の完了を助けるように命令（マシン読み出し可能な命令）を実行するよう構成された処理要素である。図１の例では、ホストプロセッサ１０６は、ヘテロジニアスシステム１０４のためのプライマリ処理要素であり、少なくとも１つのコアを含む。代替的に、ホストプロセッサ１０６は、（例えば、１よりも多いＣＰＵが利用される例において）コプライマリ 処理要素であってよく、一方、他の例では、ホストプロセッサ１０６は、セカンダリ処理要素であってよい。

【００２１】

図１の表されている例では、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、及び／又は第３アクセラレータ１１０ｃのうちの１つ以上は、ハードウェアアクセラレーションなどの計算タスクのためにヘテロジニアスシステム１０４で実行するプログラムによって利用され得る処理要素である。例えば、第１アクセラレータ１１０ａは、ＡＩのためのマシンビジョンタスクを処理する処理性能及び全体性能を改善するよう設計及び／又は別なふうに構成若しくは構造化されている処理リソースを含む処理要素である（例えば、ＶＰＵ）。

【００２２】
本願で開示される例では、ホストプロセッサ１０６、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、及び第３アクセラレータ１１０ｃの夫々は、コンピュータシステム１００の他の要素及び／又はシステムメモリ１０２と通信している。例えば、ホストプロセッサ１０６、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、及び第３アクセラレータ１１０ｃ、及び／又はシステムメモリ１０２は、第１通信バス１０８を介して通信している。本願で開示されるいくつかの例では、ホストプロセッサ１０６、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、第３アクセラレータ１１０ｃ、及び／又はシステムメモリ１０２は、任意の適切な有線及び／又は無線通信方法により通信し得る。その上、本願で開示されるいくつかの例では、ホストプロセッサ１０６、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、第３アクセラレータ１１０ｃ、及び／又はシステムメモリ１０２は、任意の適切な有線及び／又は無線通信方法によりコンピュータシステム１００の外部の如何なるコンポーネントとも通信し得る。
【００２３】
図１の例では、第１アクセラレータ１１０ａは、例となる畳み込みエンジン１１２、例となるＲＮＮエンジン１１４、例となるメモリ１１６、例となるメモリ管理ユニット（Memory Management Unit，ＭＭＵ）１１８、例となるＤＳＰ１２０、及び例となるコントローラ１２２を含む。本願で開示される例では、畳み込みエンジン１１２、ＲＮＮエンジン１１４、メモリ１１６、メモリ管理ユニット（ＭＭＵ）１１８、ＤＳＰ１２０、及び／又はコントローラ１２２のいずれも、ＣＢＢと称されることがある。本願で開示されるいくつかの例では、メモリ１１６及び／又はＭＭＵ１１８は、基盤要素と称されることがある。例えば、メモリ１１６及び／又はＭＭＵ１１８は、第１アクセラレータ１１０ａの外で実装されてもよい。例となる畳み込みエンジン１１２、例となるＲＮＮエンジン１１４、例となるメモリ１１６、例となるＭＭＵ１１８、例となるＤＳＰ１２０、及び例となるコントローラ１２２の夫々は、例となる第１スケジューラ１２４、例となる第２スケジューラ１２６、例となる第３スケジューラ１２８、例となる第４スケジューラ１３０、例となる第５スケジューラ１３２、及び例となる第６スケジューラ１３４を夫々含む。例となるＤＳＰ１２０及び例となるコントローラ１２２の夫々は、例となるカーネルライブラリ１３６及び例となる第２カーネルライブラリ１３８を更に含む。
【００２４】
図１の表されている例では、畳み込みエンジン１１２は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（Programmable Logic Device(s)，ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（Digital Signal Processor(s)，ＤＳＰ）などがある。畳み込みエンジン１１２は、畳み込みに関連したタスクの処理を改善するよう構成されているデバイスである。更に、畳み込みエンジン１１２は、視覚心像の解析に関連したタスク及び／又はＣＮＮに関連した他のタスクの処理を改善する。
【００２５】
図１の例では、ＲＮＮエンジン１１４は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。ＲＮＮエンジン１１４は、ＲＮＮに関連したタスクの処理を改善するよう構成されているデバイスである。更に、ＲＮＮエンジン１１４は、セグメント化されていない繋がった手書きの認識や発話認識の解析に関連したタスク及び／又はＲＮＮに関連した他のタスクの処理を改善する。
【００２６】

図１の例では、メモリ１１６は、例えば、フラッシュメモリ、磁気媒体、光学媒体などのような、データを記憶するための如何なるデバイスによっても実装されてよい。更に、例となるメモリ１１６に記憶されるデータは、例えば、バイナリデータ、コンマ区切りデータ、タブ区切りデータ、構造化されたクエリ言語（ＳＱＬ）構造、などのような、如何なるデータフォーマットにあってもよい。メモリ１１６は、畳み込みエンジン１１２、ＲＮＮエンジン１１４、ＭＭＵ１１８、ＤＳＰ１２０、及び直接メモリアクセス（Direct Memory Access，ＤＭＡ）機能性を含むコントローラ１２２のうちの少なくとも１つの間の共有ストレージである。更に、メモリ１１６は、畳み込みエンジン１１２、ＲＮＮエンジン１１４、ＭＭＵ１１８、ＤＳＰ１２０、及びコントローラ１２２のうちの少なくとも１つが、ホストプロセッサ１０６から独立したシステムメモリ１０２にアクセスすることを可能にする。図１の例では、メモリ１１６は、第１アクセラレータ１１０ａのローカルの物理ストレージであるが、他の例では、メモリ１１６は、第１アクセラレータ１１０ａの外にあっても、かつ／あるいは、第１アクセラレータ１１０ａに対して別なふうに遠隔にあってもよい。更なる例では、メモリ１１６は、仮想ストレージであってもよい。図１の例では、メモリ１１６は、不揮発性ストレージ（例えば、リードオンリーメモリ（ＲＯＭ）、プログラム可能ＲＯＭ（ＰＲＯＭ）、消去可能ＰＲＯＭ（ＥＰＲＯＭ）、電気的消去可能なＰＲＯＭ（ＥＥＰＲＯＭ）、など）である。他の例では、メモリ１１６は、不揮発性基本入出力システム（ＢＩＯＳ）又はフラッシュストレージであってよい。更なる例では、メモリ１１６は、揮発性メモリであってもよい。

【００２７】

図１の表されている例では、例となるＭＭＵ１１８は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。ＭＭＵ１１８は、メモリ１１６及び／又はシステムメモリ１０２の全てのアドレスへの参照を含むである。ＭＭＵ１１８は更に、畳み込みエンジン１１２、ＲＮＮエンジン１１４、ＤＳＰ１２０、及び／又はコントローラ１２２のうちの１つ以上によって利用される仮想メモリアドレスを、メモリ１１６及び／又はシステムメモリ１０２内の物理アドレスへ変換する。

【００２８】

図１の例では、ＤＳＰ１２０は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。ＤＳＰ１２０は、デジタル信号の処理を改善するデバイスである。例えば、ＤＳＰ１２０は、カメラ及び／又はコンピュータビジョンに関する他のセンサからのデータなどの連続的な現実世界の信号を測定し、フィルタ処理し、及び／又は圧縮する処理を助ける。より一般的には、ＤＳＰ１２０は、第１カーネルライブラリ１３６内の例となるカーネルを介して、他の固定機能ＣＢＢ（例えば、ＲＮＮエンジン１１４、ＣＮＮエンジン、など）によってサーブされない作業負荷からの如何なる作業負荷ノードも実装するために使用される。更には、作業負荷が、第１言語（例えば、ＴｅｎｓｏｒＦｌｏｗ、ＣＡＦＦＥ、ＯＮＮＸ、など）に基づき書かれた１００個の作業負荷ノードを含む場合に、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、及び／又は第３アクセラレータ１１０ｃは、１００個の作業負荷ノードのうちの２０個の作業負荷ノードを固定機能として実行し（例えば、ＲＮＮエンジン１１４、ＣＮＮエンジン、などを用いて実行し）、次いで、１００個の作業負荷ノードのうちの残り８０個の作業負荷ノードを第１カーネルライブラリ１３６内の各々のカーネルを用いて実行し得る。このようにして、同じ言語（例えば、ＴｅｎｓｏｒＦｌｏｗ、ＣＡＦＦＥ、ＯＮＮＸ、など）に基づく如何なる任

意の要素も、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、及び／又は第３アクセラレータ１１０ｃにマッピングされ得る。

【００２９】

図１で、コントローラ１２２は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。コントローラ１２２は、第１アクセラレータ１１０ａの制御ユニットとして実装される。例えば、コントローラ１２２は、第１アクセラレータ１１０ａの動作を指示する。いくつかの例では、コントローラ１２２はクレジットマネージャを実装する。更には、コントローラ１２２は、ホストプロセッサ１０６から受け取られたマシン読み出し可能な命令に如何にして応答するかを畳み込みエンジン１１２、ＲＮＮエンジン１１４、メモリ１１６、ＭＭＵ１１８、及び／又はＤＳＰ１２０のうちの１つ以上に指示することができる。

【００３０】

図１の例では、第１スケジューラ１２４、第２スケジューラ１２６、第３スケジューラ１２８、第４スケジューラ１３０、第５スケジューラ１３２、及び第６スケジューラ１３４の夫々は、畳み込みエンジン１１２、ＲＮＮエンジン１１４、メモリ１１６、ＭＭＵ１１８、ＤＳＰ１２０、及びコントローラ１２２が夫々、第１アクセラレータ１１０ａにオフロード及び／又は別なふうに送信されている作業負荷の部分をいつ実行するかを決定するデバイスである。その上、第１カーネルライブラリ１３６及び第２カーネルライブラリ１３８の夫々は、１つ以上のカーネルを含むデータ構造である。第１カーネルライブラリ１３６及び第２カーネルライブラリ１３８のカーネルは、例えば、ＤＳＰ１２０及びコントローラ１２２での高スループットのために夫々コンパイルされたルーチンである。カーネルは、例えば、コンピュータシステム１００で実行されるべき実行ファイルの実行可能なサブセションに対応する。

【００３１】

本願で開示される例では、畳み込みエンジン１１２、ＲＮＮエンジン１１４、メモリ１１６、ＭＭＵ１１８、ＤＳＰ１２０、及びコントローラ１２２の夫々は、第１アクセラレータ１１０ａの他の要素と通信している。例えば、畳み込みエンジン１１２、ＲＮＮエンジン１１４、メモリ１１６、ＭＭＵ１１８、ＤＳＰ１２０、及びコントローラ１２２は、例となる第２通信バス１４０を介して通信している。いくつかの例では、第２通信バス１４０は、１つ以上のコンピュータファブリック（例えば、設定及び制御ファブリック、データファブリック、など）によって実装されてよい。本願で開示されるいくつかの例では、畳み込みエンジン１１２、ＲＮＮエンジン１１４、メモリ１１６、ＭＭＵ１１８、ＤＳＰ１２０、及びコントローラ１２２は、任意の適切な有線及び／又は無線通信方法を介して通信し得る。その上、本願で開示されるいくつかの例では、畳み込みエンジン１１２、ＲＮＮエンジン１１４、メモリ１１６、ＭＭＵ１１８、ＤＳＰ１２０、及びコントローラ１２２の夫々は、任意の適切な有線及び／又は無線通信方法を介して第１アクセラレータ１１０ａの外の如何なるコンポーネントとも通信し得る。

【００３２】

上記の通り、例となる第１アクセラレータ１１０ａ、例となる第２アクセラレータ１１０ｂ、及び／又は例となる第３アクセラレータ１１０ｃのいずれも、各々のアクセラレータの動作にとって一般的及び／又は特有である様々なＣＢＢを含んでよい。例えば、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、及び第３アクセラレータ１１０ｃの夫々は、メモリ、ＭＭＵ、コントローラ、及びＣＢＢの夫々のための各々のスケジューラなどの一般的なＣＢＢを含む。その上、又は代替的に、第１アクセラレータ１１０ａ、例となる第２アクセラレータ１１０ｂ、及び／又は例となる第３アクセラレータ１１０ｃのいずれにも位置していない外付けＣＢＢが包含及び／又は付加されてもよい。例えば

、コンピュータシステム１００のユーザは、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、及び／又は第３アクセラレータ１１０ｃのうちのいずれか１つを利用して外部ＲＮＮエンジンを作動させてもよい。

【００３３】

図１の例では、第１アクセラレータ１１０ａはＶＰＵを実装し、畳み込みエンジン１１２、ＲＮＮエンジン１１４、及びＤＳＰ１２０（例えば、第１アクセラレータ１１０ａの動作に特有の動作に特有のＣＢＢ）を含み、一方、第２アクセラレータ１１０ｂ及び第３アクセラレータ１１０ｃは、第２アクセラレータ１１０ｂ及び／又は第３アクセラレータ１１０ｃの動作に特有の追加の又は代替的なＣＢＢを含んでもよい。例えば、第２アクセラレータ１１０ｂがＧＰＵを実装する場合に、第２アクセラレータ１１０ｂの動作に特有のＣＢＢは、スレッドディスパッチャ、グラフィクス技術インターフェイス、及び／又はコンピュータグラフィクスの処理及び／又は画像処理の処理速度及び全体性能を改善するために望ましい任意の他のＣＢＢを含むことができる。更に、第３アクセラレータ１１０ｃがＦＰＧＡを実装する場合に、第３アクセラレータ１１０ｃの動作に特有のＣＢＢは、１つ以上の算術論理演算装置（Arithmetic Logic Unit，ＡＬＵ）、及び／又は一般的な計算の処理の処理速度及び全体性能を改善するために望ましい任意の他のＣＢＢを含むことができる。

【００３４】

図１のヘテロジニアスシステム１０４は、ホストプロセッサ１０６、第１アクセラレータ１１０ａ、第２アクセラレータ１１０ｂ、及び第３アクセラレータ１１０ｃを含み、一方、いくつかの例で、ヘテロジニアスシステム１０４は、特定用途向け命令セットプロセッサ（Application Specific Instruction set Processors，ＡＳＩＰ）、フィジクス処理ユニット（Physics Processing Unit，ＰＰＵ）、指定済みＤＳＰ（designated DSPs）、画像プロセッサ、コプロセッサ、浮動小数点ユニット、ネットワークプロセッサ、マルチコアプロセッサ、及びフロントエンドプロセッサを含む任意数の処理要素（例えば、ホストプロセッサ及び／又はアクセラレータ）を含んでよい。

【００３５】

図２は、例となるグラフコンパイラ２０２及び１つ以上の例となるセレクタ２０４を含むコンピュータシステム２００を例示するブロック図である。図２の例では、コンピュータシステム２００は、例となる作業負荷２０６及び例となるアクセラレータ２０８を更に含む。更に、図２で、アクセラレータ２０８は、例となるクレジットマネージャ２１０と、例となる制御及び設定（Control and Configuration，ＣｎＣ）ファブリック２１２と、例となる畳み込みエンジン２１４と、例となるＭＭＵ２１６と、例となるＲＮＮエンジン２１８と、例となるＤＳＰ２２０と、例となるメモリ２２２と、例となるコンフィグレーションコントローラ２２４とを含む。図２の例では、メモリ２２２は、例となるＤＭＡユニット２２６及び１つ以上の例となるバッファ２２８を含む。本願で開示される他の例では、如何なる適切なＣＢＢも、アクセラレータ２０８に包含及び／又は追加されてよい。

【００３６】

図２の表されている例では、例となるグラフコンパイラ２０２は、コンパイルする手段、又はコンパイル手段である。図２の表されている例では、１つ以上のセレクタの中の例となるセレクタは、選択する手段、又は選択手段である。図２の表されている例では、例となるクレジットマネージャ２１０は、クレジットを管理する手段、又はクレジット管理手段である。図２の表されている例では、例となるコンフィグレーションコントローラ２２４は、制御する手段、又は制御手段である。図２の例では、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、メモリ２２２、及び／又はカーネルバンク２３２内のカーネルのいずれも、計算する手段、又は計算手段であってよい。

【００３７】

図２の表されている例では、グラフコンパイラ２０２は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に

、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。図２で、グラフコンパイラ２０２は、１つ以上のセレクタ２０４へ及びアクセラレータ２０８へ結合されている。動作中、グラフコンパイラ２０２は、作業負荷２０６を受け取り、作業負荷２０６を、アクセラレータ２０８によって実行される例となる実行ファイル２３０にコンパイルする。例えば、グラフコンパイラ２０２は、作業負荷２０６を受け取り、作業負荷２０６（例えば、グラフ）の様々な作業負荷ノードをアクセラレータ２０８の様々なＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はＤＭＡユニット２２６のいずれか）に割り当てる。グラフコンパイラ２０２は更に、作業負荷２０６内の各作業負荷ノードに対応する１つ以上のセレクタ２０４の中の例となるセレクタを生成する。その上、グラフコンパイラ２０２は、アクセラレータ２０８のメモリ２２２内の１つ以上のバッファ２２８のためにメモリを割り当てる。本願で開示される例では、実行ファイル２３０は、別個のシステム（例えば、コンパイルシステム及び／又はコンパイルプロセッサ）で生成され、後の使用のために異なるシステム（例えば、デプロイメントシステム、ランタイムシステム、デプロイメントプロセッサ、など）で記憶されてよい。例えば、グラフコンパイラ２０２及び１つ以上のセレクタ２０４は、アクセラレータ２０８とは別個のシステムに位置してよい。

【００３８】

図２で表されている例では、１つ以上のセレクタ２０４は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。１つ以上のセレクタ２０４は、グラフコンパイラ２０２、アクセラレータ２０８、及びＤＳＰ２２０内に位置する例となるカーネルバンク２３２へ結合されている。１つ以上のセレクタ２０４は、作業負荷２０６を取得するようグラフコンパイラ２０２へ結合されている。作業負荷２０６内の各作業負荷ノード（例えば、タスク）は、関連する作業負荷を実行するために使用されるＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はＤＭＡユニット２２６のいずれか）を示す。本願で開示される例では、１つ以上のセレクタ２０４の中のセレクタは、作業負荷ノードごとに生成され、対応するＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はＤＭＡユニット２２６のいずれか）及び／又はカーネルバンク２３２内のカーネルと関連付けられる。１つ以上のセレクタ２０４は、作業負荷２０６に応答してグラフコンパイラ２０２によって生成され、そのようなものとして、様々なＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はＤＭＡユニット２２６のいずれか）及び／又はカーネルバンク２３２内のカーネルの各々の入力及び／又は出力条件を識別することができる。１つ以上のセレクタによるかような識別は、グラフコンパイラ２０２による使用のために、抽象化された知識として表現され得る。かような抽象化された知識は、グラフコンパイラ２０２がアクセラレータ２０８のヘテロジニアス性質とは無関係に動作することを可能にする。

【００３９】

その上、グラフコンパイラ２０２は、作業負荷２０６からの各々の作業負荷ノードを対応するＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はＤＭＡユニット２２６のいずれか）及び／又はカーネルバンク２３２内のカーネルにマッピングするために、１つ以上のセレクタ２０４を利用する。更に、グラフコンパイラ２０２は、対応する作業負荷ノード及び隣接する作業負荷ノード（例えば、作業負荷ノードの結果として得られるコンシューマ及び／又はプロデューサ）

ごに適切な量のクレジットで特定の動作及びパラメータについて対応するＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、お及び／又はＤＭＡユニット２２６のいずれか）を設定する、などのために、１つ以上のセレクタ２０４を利用する。本願で開示されるいくつかの例で、１つ以上のセレクタ２０４は、作業負荷２０６からの各々の作業負荷ノードを対応するＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はＤＭＡユニット２２６のいずれか）及び／又はカーネルバンク２３２内のカーネルにマッピングしてよい。

【００４０】

本願で開示される例では、１つ以上のセレクタ２０４は、グラフコンパイラ２０２に含まれてもよい。本願で開示されるかような例では、追加のセレクタが１つ以上のセレクタ２０４に含まれてもよく、あるいは、代替的に、１つ以上のセレクタ２０４の中の現在のセレクタが、作業負荷２０６及び／又はアクセラレータ２０８の変化（例えば、新しい作業負荷２０６が供給される、追加のＣＢＢがアクセラレータ２０８に加えられる、など）に応答して変更されてもよい。

【００４１】

いくつかの例で、グラフコンパイラ２０２は、データがスケーリングされるべきであることを示す作業負荷２０６からの作業負荷ノードを識別する。データがスケーリングされるべきであることを示すかような作業負荷ノードは、そのようなタスクに関連した１つ以上のセレクタ２０４へ送られる。識別された作業負荷ノードに関連した１つ以上のセレクタ２０４は、グラフコンパイラ２０２が作業負荷を実行するために、ＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はＤＭＡユニット２２６のいずれか）及び／又はカーネルバンク２３２内のカーネルを、そのような識別されたＣＢＢ及び／又はカーネルバンク２３２内のカーネルの識別された入力及び／又は出力条件とともに、識別することができる。

【００４２】

図２の例では、作業負荷２０６は、例えば、アクセラレータによって実行されるグラフ、関数、アルゴリズム、プログラム、アプリケーション、及び／又は他のコードである。いくつかの例で、作業負荷２０６は、グラフ、関数、アルゴリズム、プログラム、アプリケーション、及び／又は他のコードの記述である。作業負荷２０６は、ユーザから取得されたあらゆる任意のグラフ及び／又はあらゆる適切な入力であってよい。例えば、作業負荷２０６は、ディープラーニングトポロジ及び／又はコンピュータビジョンなどのＡＩ処理に関連した作業負荷であってよい。動作中、作業負荷２０６（例えば、グラフ）内の各作業負荷ノードは、特定のＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はＤＭＡユニット２２６のいずれか）、カーネルバンク２３２内のカーネル、及び／又は作業負荷ノード内のタスクを実行するための入力及び／又は出力条件を指定する制約を含む。従って、グラフコンパイラ２０２に含まれる例となるプラグイン２３６は、作業負荷２０６（例えば、グラフ）の作業負荷ノードと関連するＣＢＢ及び／又はカーネルバンク２３２内のカーネルとの間のマッピングを可能にする。プラグイン２３６は、作業負荷２０６（例えば、グラフ）内の作業負荷を割り当てるよう、１つ以上のセレクタ２０４によって取得された抽象化された知識（例えば、夫々のＣＢＢ及び／又はカーネルバンク２３２内のカーネルの各々の標準入力及び／又は出力定義）と相互作用する。本願で開示されるかような例では、プラグイン２３６は、１つ以上のセレクタ２０４によって取得された抽象化された知識（例えば、夫々のＣＢＢ及び／又はカーネルバンク２３２内のカーネルの各々の標準入力及び／又は出力定義）に基づいて、様々なＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はＤＭＡユニット２２６のいずれか）及び／又はカーネルバンク２３２内のカーネルに対する作業負荷２０６内の作業負荷のマッピングを可能にするように、作業負荷２０６（例えば、グラフ）内の作業負荷ノードと様々なＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２

０、及び／又はＤＭＡユニット２２６のいずれか）及び／又はカーネルバンク２３２内のカーネルとの間の変換レイヤを形成し得る。

【００４３】

図２の例では、アクセラレータ２０８は、グラフコンパイラ２０２へ及び１つ以上のセレクタ２０４へ結合されている。本願で開示されるいくつかの例では、コンパイル時間中に、グラフコンパイラ２０２は、コンパイルシステム（例えば、第１プロセッサ）に対して作用し、１つ以上のセレクタ２０４を利用してコンパイル処理を実行（例えば、実行ファイル２３０を生成する）してよい。結果として、グラフコンパイラ２０２は、コンパイルシステム（例えば、第１プロセッサ）で、例となる実行ファイル２３０を生成する。更に、又は代替的に、実行ファイル２３０は、後の使用のためにデータベースに記憶されてよい。例えば、実行ファイル２３０は、コンパイルシステム（例えば、第１プロセッサ）及び／又はあらゆる外部及び／又は内部システム（例えば、デプロイメントシステム、第２プロセッサ、など）で記憶及び実行されてよい。ランタイム中、実行ファイル２３０は、デプロイメントシステム（例えば、図１のシステム１００、第２プロセッサ、など）において動作可能である。コンパイルシステム（例えば、第１プロセッサ）は、デプロイメントシステム（例えば、図１のシステム１００、第２プロセッサ、など）から別の場所で動作可能であってもよい。代替的に、コンパイルシステム及び／又はデプロイメントシステムは、組み合わされてもよく、そのようなものとして、アクセラレータによって直接に実行されている実行ファイル（例えば、実行ファイル２３０）へと任意の作業負荷（例えば、作業負荷２０６）の実行時（Just in Time，ＪＩＴ）コンパイルを可能にし得る。

【００４４】

図２の表されている例では、クレジットマネージャ２１０はＣｎＣファブリック２１２へ結合されている。クレジットマネージャ２１０は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。クレジットマネージャ２１０は、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０のうちの１つ以上に関連したクレジットを管理するデバイスである。いくつかの例で、クレジットマネージャ２１０は、クレジットマネージャコントローラとしてコントローラによって実装可能である。クレジットは、メモリ２２２で利用可能である作業負荷ノードに関連したデータ及び／又は作業負荷ノードの出力のためにメモリ２２２で利用可能な空間の量を表す。他の例では、クレジット及び／又はクレジット値は、データを記憶し及び／又は別なふうに書き込むために利用可能なバッファ（例えば、バッファ２２８の１つ）内のスロットの数を示し得る。

【００４５】

クレジットマネージャ２１０及び／又はコンフィグレーションコントローラ２２４は、グラフコンパイラ２０２から受け取られてコンフィグレーションコントローラ２２４によって分配された実行ファイル２３０に基づいて、メモリ２２２を、所与の作業負荷の各作業負荷ノードに関連した１つ以上のバッファ（例えば、バッファ２２８）に区画することができる。そのようなものとして、クレジットは、データを記憶し及び／又は別なふうに書き込むために利用可能な関連バッファ（例えば、バッファ２２８）内のスロットを表し得る。例えば、クレジットマネージャ２１０は、作業負荷２０６に対応する情報（例えば、設定及び制御メッセージ２３４及び／又はさもなければ、設定メッセージ及び制御メッセージ）を受け取る。例えば、クレジットマネージャ２１０は、コンフィグレーションコントローラ２２４から、ＣｎＣファブリック２１２を介して、プロデューサとして初期化されたＣＢＢ及びコンシューマとして初期化されたＣＢＢを示すコンフィグレーションコントローラ２２４によって決定された情報を受け取る。

【００４６】

本願で開示される例では、特定の作業負荷ノードを実行するよう指示するコンフィグレーションコントローラ２２４から受け取られ命令に応答して（例えば、コンフィグレーションコントローラ２２４が設定及び制御メッセージ２３４を送信することに応答して）、クレジットマネージャ２１０は、対応するクレジットを、初期プロデューサとして振る舞うＣＢＢへ供給し及び／又は別なふうに送信する（例えば、バッファの３つのスロットにデータを書き込むために畳み込みエンジン２１４へ３つのクレジットを供給する）。初期プロデューサとして振る舞うＣＢＢが作業負荷ノードを完了すると、クレジットは、ＣＢＢから見た原点（例えば、クレジットマネージャ２１０）へ返送される。クレジットマネージャ２１０は、プロデューサからクレジットを取得することに応答して、コンシューマとして振る舞うＣＢＢへクレジットを供給し及び／又は別なふうに送信する（例えば、ＤＳＰ２２０は、バッファの３つのスロットからデータを読み出すよう３つのスロットを取得する）。プロデューサ及びコンシューマのかような順序は、実行ファイル２３０を用いて決定される。このようにして、ＣＢＢは、ヘテロジニアス性質にかかわらず、クレジットマネージャ２１０を介して動作するよう、能力の指示を送る。プロデューサＣＢＢは、他のＣＢＢによって利用されるデータを生成し、一方、コンシューマＣＢＢは、他のＣＢＢによって生成されたデータを消費し及び／又は別なふうに処理する。

【００４７】

本願で開示されるいくつかの例では、クレジットマネージャ２１０は、作業負荷ノードの実行が完了しているかどうかを判定するよう構成されてよい。かような例では、クレジットマネージャ２１０は、作業負荷ノードに関連したＣＢＢにおいて全てのクレジットをクリアしてよい。

【００４８】

図２の例では、ＣｎＣファブリック２１２は、クレジットマネージャ２１０、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、メモリ２２２、及びコンフィグレーションコントローラ２２４へ結合されている。本願で開示されるいくつかの例では、メモリ２２２及び／又はＭＭＵ２１６は、基盤要素と称され、ＣｎＣファブリック２１２へ結合されなくてもよい。ＣｎＣファブリック２１２は、クレジットマネージャ２１０、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０のうちの１つ以上がクレジットマネージャ２１０、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、メモリ２２２、及び／又はコンフィグレーションコントローラ２２４のうちの１つ以上との間でクレジットを送受信することを可能にする配線及び少なくとも１つの論理回路の回路網を含む制御ファブリックである。その上、ＣｎＣファブリック２１２は、例となる設定及び制御メッセージ２３４を１つ以上のセレクタ２０４へ及び／又はそれから送信するよう構成される。本願で開示される他の例では、如何なる適切なコンピューティングファブリックも、ＣｎＣファブリック２１２を実装するために使用されてよい（例えば、ＡＸＩ（Advanced eXtensible Interface）など）。

【００４９】

図２の表される例では、畳み込みエンジン２１４は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。畳み込みエンジン２１４は、ＣｎＣファブリック２１２へ結合されている。畳み込みエンジン２１４は、畳み込みに関連したタスクの処理を改善するよう構成されているデバイスである。更に、畳み込みエンジン１１２は、視覚心像の解析に関連したタスク及び／又はＣＮＮに関連した他のタスクの処理を改善する。

【００５０】

図２の表される例では、例となるＭＭＵ２１６は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は

代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。ＭＭＵ２１６は、ＣｎＣファブリック２１２へ結合されている。ＭＭＵ２１６は、メモリ２２２及び／又はアクセラレータ２０８に対して遠隔にあるメモリのアドレスの変換を可能にするデバイスである。ＭＭＵ２１６は更に、クレジットマネージャ２１０、畳み込みエンジン２１４、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０のうちの１つ以上によって利用される仮想メモリアドレスを、メモリ２２２及び／又はアクセラレータ２０８に対して遠隔にあるメモリ内の物理アドレスへ変換する。

【００５１】

図２で、ＲＮＮエンジン２１８は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。ＲＮＮエンジン２１８は、ＣｎＣファブリック２１２へ結合されている。ＲＮＮエンジン２１８は、ＲＮＮに関連したタスクの処理を改善するよう構成されているデバイスである。更に、ＲＮＮエンジン２１８は、セグメント化されていない繋がった手書きの認識や発話認識の解析に関連したタスク及び／又はＲＮＮに関連した他のタスクの処理を改善する。

【００５２】

図２の例では、ＤＳＰ２２０は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。ＤＳＰ２２０は、ＣｎＣファブリック２１２へ結合されている。ＤＳＰ２２０は、デジタル信号の処理を改善するデバイスである。例えば、ＤＳＰ２２０は、カメラ及び／又はコンピュータビジョンに関する他のセンサからのデータなどの連続的な現実世界の信号を測定し、フィルタ処理し、及び／又は圧縮する処理を助ける。

【００５３】

図２の例では、メモリ２２２は、例えば、フラッシュメモリ、磁気媒体、光学媒体などのような、データを記憶するための如何なるデバイスによっても実装されてよい。更に、例となるメモリ２２２に記憶されるデータは、例えば、バイナリデータ、コンマ区切りデータ、タブ区切りデータ、構造化されたクエリ言語（ＳＱＬ）構造、などのような、如何なるデータフォーマットにあってもよい。メモリ２２２は、ＣｎＣファブリック２１２へ結合されている。メモリ２２２は、クレジットマネージャ２１０、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はコンフィグレーションコントローラ２２４のうちの少なくとも１つの間の共有ストレージである。メモリ２２２は、ＤＭＡユニット２２６を含む。更に、メモリ２２２は、コンフィグレーションコントローラ２２４及び／又はクレジットマネージャ２１０によって受け取られた実行ファイルに関連した作業負荷の１つ以上の作業負荷ノードに関連した１つ以上のバッファ２２８に区画され得る。更には、メモリ２２２のＤＭＡユニット２２６は、ＣｎＣファブリック２１２を介してコンフィグレーションコントローラ２２４によって供給されるコマンドに応答して動作する。本願で開示されるいくつかの例では、メモリ２２２のＤＭＡユニット２２６は、クレジットマネージャ２１０、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はコンフィグレーションコントローラ２２４のうちの少なくとも１つが、各々のプロセッサ（例えば、ホストプロセッサ１０６）から独立したアクセラレータ２０８の遠隔にあるメモリにアクセスすることを可能にする。図２の例では、メモリ２２２は、アクセラレータ２０８のローカルの物理ストレージであ

る。更に、又は代替的に、他の例では、メモリ２２２は、アクセラレータ２０８の外にあっても、かつ／あるいは、アクセラレータ２０８に対して別なふうに遠隔にあってもよい。本願で開示される更なる例では、メモリ２２２は、仮想ストレージであってもよい。図２の例では、メモリ２２２は、不揮発性ストレージ（例えば、ＲＯＭ、ＰＲＯＭ、ＥＰＲＯＭ、ＥＥＰＲＯＭ、など）である。他の例では、メモリ２２２は、不揮発性ＢＩＯＳ又はフラッシュストレージであってよい。更なる例では、メモリ２２２は、揮発性メモリであってもよい。

【００５４】

本願で開示される例では、コンフィグレーションコントローラ２２４は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。コンフィグレーションコントローラ２２４は、アクセラレータ２０８の制御ユニットとして実装される。本願で開示される例では、１つ以上のセレクタ２０４は、実行ファイル２３０を生成するためにグラフコンパイラ２０２へ設定及び制御メッセージ２３４を送信する。本願で開示されるいくつかの例では、コンフィグレーションコントローラ２２４は、実行ファイル２３０に含まれる作業負荷ノードを示す設定及び制御メッセージ（例えば、１つ以上のセレクタ２０４によって取得され及び／又はそれへ送信された設定及び制御メッセージ２３４）を識別するよう、実行ファイル２３０を取得及びパースしてよい。そのようなものとして、コンフィグレーションコントローラ２２４は、実行ファイル２３０のタスクを実行するために様々なＣＢＢへ設定及び制御メッセージ（例えば、１つ以上のセレクタ２０４によって取得され及び／又はそれへ送信された設定及び制御メッセージ２３４）を供給する。本願で開示されるかような例では、設定及び制御メッセージ２３４は、実行ファイル２３０に埋め込まれ、そのようなものとして、コンフィグレーションコントローラ２２４へ供給され、様々なＣＢＢ及び／又はカーネルバンク２３２に位置するカーネルへ送られる。例えば、コンフィグレーションコントローラ２２４は、実行可能ファイルにおいて作業負荷ノードを識別するよう実行ファイル２３０をパースし、グラフコンパイラ２０２からクレジットマネージャ２１０を介して受け取られた実行ファイル２３０及び／又は他のマシン読み出し可能な命令に如何にして応答するかを畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、カーネルバンク２３２内のカーネル、及び／又はメモリ２２２のうちの１つ以上に指示する。

【００５５】

本願で開示される例では、コンフィグレーションコントローラ２２４は、取得された実行ファイル２３０からの作業負荷ノード（例えば、設定及び制御フォーマットにある）を、識別された対応するＣＢＢへ送信する。同じく、コンフィグレーションコントローラ２２４は、クレジットの分配を開始するために作業負荷ノード（例えば、設定及び制御フォーマットにある）をクレジットマネージャ２１０へ送信してもよい。

【００５６】

図２の例では、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０は、各々のスケジューラ２３８、２４０、２４２、及び２４４を夫々含んでよい。動作中、スケジューラ２３８、２４０、２４２、及び２４４は、アクセラレータ２０８のコンフィグレーションコントローラ２２４、クレジットマネージャ２１０、及び／又は更なるＣＢＢによって畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０へ夫々割り当てられている作業負荷２０６の一部（例えば、作業負荷ノード）を夫々実行する。所与の作業負荷ノードのタスク及び／又は他の動作に応じて、作業負荷ノードはプロデューサ及び／又はコンシューマであることができる。

【００５７】

図２の例では、スケジューラ２３８、２４０、２４２、２４４のいずれも、クレジットマネージャ２１０によって供給される指示に応答して、データ（例えば、プロデューサ）をバッファ（例えば、バッファ２２８の少なくとも１つ）に書き込むことを対応するＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０のいずれか）に示す作業負荷ノードに関連したクレジット値を受け取り及び／又は別なふうにメモリにロードしてよい。例えば、実行ファイル２３０が、プロデューサとして動作してデータの３つのビットをバッファ（例えば、バッファ２２８の１つ）に書き込むようにＲＮＮエンジン２１８に対して指示する場合に、次いで、スケジューラ２４２は、３つのクレジット値をＲＮＮエンジン２１８にロードし得る。その上、かような例では、実行ファイル２３０は、ＭＭＵ２１６がＲＮＮエンジン２１８によって前に書き込まれた３つのビットを読み出すべきであることを示してもよい。そのようなものとして、スケジューラ２４２（又はＲＮＮエンジン２１８）は、３つのクレジットを、使用されると、ＭＭＵ２１６へＣｎＣファブリック２１２及びクレジットマネージャ２１０を介して送信する。

【００５８】

動作中、スケジューラ２３８、２４０、２４２、２４４、及び／又はＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０のいずれか）は、漸増的に及び／又は任意の適切な方法でクレジットを送信してよい。他の例では、第１ＣＢＢは、第１作業負荷ノードを実行するために供給された第１クレジット値を有してよい。かような例では、第１作業負荷ノードを実行することに応答して、第１ＣＢＢは、データをメモリ２２２内の第１バッファ（例えば、バッファ２２８の１つ）に書き込み、第２クレジット値をクレジットマネージャ２１０に送信する。第２クレジット値は、データを第１バッファ（例えば、バッファ２２８の１つ）に書き込むために使用される第１クレジット値の量を表す。例えば、第１クレジット値が３であり、第１ＣＢＢがバッファ（例えば、バッファ２２８の１つ）の２つのスロットに書き込む場合に、次いで、第１ＣＢＢは、２つのクレジットをクレジットマネージャ２１０へ送信する。これに応えて、クレジットマネージャ２１０は、第２クレジット値（例えば、２つのクレジット）を第２ＣＢＢへ送信し、第２ＣＢＢは、第２クレジット値（例えば、２つのクレジット）を利用して、バッファ（例えば、バッファ２２８の１つ）の２つのスロット内のデータを読み出す。そのようなものとして、第２ＣＢＢは、次いで、第２作業負荷ノードを実行することができる。本願で開示される例では、バッファ２２８は、データの読み出し及び／又は書き込みにおいて使用される任意の適切な数のデータスロットを含む循環バッファにより実装される。

【００５９】

図２の表される例では、カーネルバンク２３２は、１つ以上のカーネルを含むデータ構造である。カーネルバンク２３２のカーネルは、例えば、ＤＳＰ２２０での高スループットのためにコンパイルされたルーチンである。本願で開示される他の例では、各ＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０のいずれか）は、各々のカーネルバンクを含んでよい。カーネルは、例えば、アクセラレータ２０８で実行されるべき実行ファイルの実行可能なサブセションに対応する。図２の例では、アクセラレータ２０８は、ＶＰＵを実装し、クレジットマネージャ２１０、ＣｎＣファブリック２１２、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及びメモリ２２２、並びにコンフィグレーションコントローラ２２４を含むが、一方で、アクセラレータ２０８は、図２に表されているものに対して付加的な又は代替のＣＢＢを含んでもよい。本願で開示される更なる及び／又は代替の例では、カーネルバンク２３２は、グラフコンパイラ２０２による使用のために抽象化されるよう１つ以上のセレクタ２０４へ結合される。

【００６０】

図２の例では、データファブリック２３３が、クレジットマネージャ２１０、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、メモリ２２２、

コンフィグレーションコントローラ２２４、及びＣｎＣファブリック２１２へ結合されている。データファブリック２３３は、クレジットマネージャ２１０、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、メモリ２２２、及び／又はコンフィグレーションコントローラ２２４のうちの１つ以上がデータを交換することを可能にする配線及び少なくとも１つの論理回路の回路網である。例えば、データファブリック２３３は、プロデューサＣＢＢがデータのタイルをメモリ、例えば、メモリ２２２及び／又は畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及びＤＳＰ２２０のうちの１つ以上に位置するメモリのバッファに書き込むことを可能にする。その上、データファブリック２３３は、コンシューマＣＢＢが、メモリ、例えば、メモリ２２２及び／又は畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及びＤＳＰ２２０のうちの１つ以上に位置するメモリのバッファからデータのタイルを読み出すことを可能にする。データファブリック２３３は、データのパッケージにおいて提供される情報に応じてメモリとの間でデータを転送する。例えば、データは、パッケージのメソッドによって転送可能であり、パケットはヘッダ、ペイロード、及びトレイラを含む。パケットのヘッダは、データのあて先アドレス、データの発信元アドレス、データ送信されているプロトコルのタイプ、及びパケット番号である。ペイロードは、ＣＢＢが生成又は消費するデータである。データファブリック２３３は、意図されたあて先アドレスを解析することによってパケットのヘッダに基づいてＣＢＢ間のデータ交換を助け得る。本願で開示されるいくつかの例では、データファブリック２３３及びＣｎＣファブリック２１２は、単一及び／又は複数のコンピューティングファブリックを用いて実装されてよい。

【００６１】

図３は、図２の１つ以上のセレクタ２０４の中の例となるセレクタ３００を表す例となるブロック図である。セレクタ３００は、特定の作業負荷ノードについて図２のグラフコンパイラ２０２によって生成されるセレクタの例を表す。かような例では、セレクタ３００は、図２の特定のＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０のいずれか）及び／又はカーネルバンク２３２内のカーネルと通じるよう生成され得る。セレクタ３００は、図２の作業負荷２０６内の個々の作業負荷ノードについて実装されてよい。その上、個別のセレクタが、作業負荷２０６内の夫々の個々の作業負荷ノードについて実装されてもよい。図３に表されているセレクタ３００は、例となるＣＢＢアナライザ３０２、例となるカーネルアナライザ３０４、及び例となるコンパイラインターフェイス３０６を含む。動作中、ＣＢＢアナライザ３０２、カーネルアナライザ３０４、及び／又はコンパイラインターフェイス３０６のいずれも、例となる通信バス３０８を介して通信してよい。図３で、通信バス３０８は、任意の適切な通信方法及び／又は装置（例えば、Ｂｌｕｅｔｏｏｔｈ（登録商標）通信、ＬＡＮ通信、ＷＬＡＮ通信、など）により実装されてよい。本願で開示されるいくつかの例では、セレクタ３００は、１つ以上のセレクタ２０４の中の例となるセレクタを表し、図２のグラフコンパイラ２０２に含まれてよい。

【００６２】

図３で表されている例では、ＣＢＢアナライザ３０２は、計算要素を解析する手段、又は計算要素解析手段である。図３の例では、カーネルアナライザ３０４は、カーネルを解析する手段、又はカーネル解析手段である。図３の例では、コンパイラインターフェイス３０６は、コンパイラ通信のための手段、又はコンパイラ通信手段である。

【００６３】

図３で表されている例では、ＣＢＢアナライザ３０２は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。動作中、ＣＢＢアナライザ３０２は、作業負荷に関連したＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２

１８、及び／又はＤＳＰ２２０のいずれか）の入力及び出力条件を識別するよう構成される。図３のＣＢＢアナライザ３０２は、標準の入力要件（例えば、データ構造、入力の数、など）に対応し、かつ、作業負荷ノードを実行すると識別されたＣＢＢに関連する入力条件のタイプを識別するよう構成される。その上、ＣＢＢアナライザ３０２は、標準の結果（例えば、出力の数、結果のタイプ、など）に対応し、かつ、作業負荷を実行すると識別されたＣＢＢに関連する出力条件のタイプを識別するよう構成される。このようにして、識別された入力及び出力条件は、ＣＢＢアナライザ３０２によって識別され、グラフコンパイラ２０２による使用のために標準のフォーマットで供給される。

【００６４】

本願開示される他の例では、ＣＢＢアナライザ３０２は、動作要件を識別するよう、関連するＣＢＢと通信してよい。例えば、ＣＣＢが、例となる作業負荷ノードを実行するために特定の割合のメモリ割り当てを必要とする場合に、かような要件は、ＣＢＢアナライザ３０２によって決定され、コンパイラインターフェイス３０６を介してグラフコンパイラ２０２へ送られ得る。

【００６５】

本願で開示されるいくつかの例では、ＣＢＢアナライザ３０２は、内部知識及び／又は、関連するＣＢＢの現在及び／又は前のモデリングを利用することによって、関連するＣＢＢと間接的に通信する。例となる内部知識及び／又は現在及び／又は前のモデリングは、ＣＢＢ動作要件の知識を含んでよい。更に、ＣＢＢアナライザ３０２は、関連する作業負荷に対してノード解析を実行して、ノードタイプを識別し得る。かような例となる解析は、セレクタ３００に位置するノードアナライザを用いて実行されてよい。更に、かような例では、識別されたノードタイプは、グラフコンパイラ２０２によって通信され、供給され、及び／又は別なふうに利用されてよい。このようにして、セレクタ３００は、対応する作業負荷ノードをマッピングするターゲットであり得る対応するＣＢＢ及び／又は複数のＣＢＢに関する知識を得る。例えば、乗算を実行することを示す作業負荷ノードが存在してよい。そのようなものとして、図２のグラフコンパイラ２０２は、（例えば、識別されたノードタイプを解析することに基づいて）乗算に関する知識を有しているセレクタ３００を呼び出し及び／又はそれと別なふうに通信し、作業負荷ノードの関連パラメータをセレクタ３００へ供給し得る。セレクタ３００のＣＢＢアナライザ３０２は、マッピングにおいて使用される作業負荷ノードを実行するＣＢＢを識別する。本願で開示されるいくつかの例では、ＣＢＢアナライザ３０２は、対応する作業負荷ノードを、対応するＣＢＢへマッピングしてよい。

【００６６】

図３で、例となるカーネルアナライザ３０４は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。動作中、カーネルアナライザ３０４は、カーネル（例えば、図２のカーネルバンク２３２に含まれるカーネル）の入力及び出力条件を識別するよう構成される。例えば、カーネルアナライザ３０４は、標準の入力要件（例えば、データ構造、入力の数、など）に対応し、かつ、作業負荷ノードを実行すると識別されたカーネルに関連する入力条件のタイプを識別するよう構成される。その上、カーネルアナライザ３０４は、標準の結果（例えば、出力の数、結果のタイプ、など）に対応し、かつ、作業負荷を実行すると識別されたカーネルに関連する出力条件のタイプを識別するよう構成される。このようにして、識別された入力及び出力条件は、グラフコンパイラ２０２による使用のために標準のフォーマットで供給される。本願で開示される例では、カーネルアナライザ３０４は、アクセラレータ２０８に含まれるいずれかのカーネル（例えば、アクセラレータにダウンロードされた新しいカーネル、など）の入力及び／又は出力条件のタイプを識別し得る。

【００６７】
本願で開示される他の例では、カーネルアナライザ３０４は、動作要件を識別するよう、関連するカーネルと通信してよい。例えば、カーネルが、例となる作業負荷ノードを実行するために特定の割合のメモリ割り当てを必要とする場合に、かような要件は、カーネルアナライザ３０４によって決定され、コンパイラインターフェイス３０６を介してグラフコンパイラ２０２へ送られ得る。
【００６８】
本願で開示されるいくつかの例では、カーネルアナライザ３０４は、内部知識及び／又は、関連するカーネルの現在及び／又は前のモデリングを利用することによって、関連するカーネルと間接的に通信する。例となる内部知識及び／又は現在及び／又は前のモデリングは、カーネル動作要件の知識を含んでよい。更に、カーネルアナライザ３０４は、関連する作業負荷に対してノード解析を実行して、ノードタイプを識別し得る。かような例となる解析は、セレクタ３００に位置するノードアナライザを用いて実行されてよい。更に、かような例では、識別されたノードタイプは、グラフコンパイラ２０２によって通信され、供給され、及び／又は別なふうに利用されてよい。例えば、乗算を実行することを示す作業負荷ノードが存在してよい。そのようなものとして、図２のグラフコンパイラ２０２は、（例えば、識別されたノードタイプに基づいて）乗算に関する知識を有しているセレクタ３００を呼び出し及び／又はそれと別なふうに通信し、作業負荷ノードの関連パラメータをセレクタ３００へ供給し得る。セレクタ３００のカーネルアナライザ３０４は、マッピングにおいて使用される作業負荷ノードを実行するＣＢＢを識別する。本願で開示されるいくつかの例では、カーネルアナライザ３０４は、対応する作業負荷ノードを、対応するカーネルへマッピングしてよい。
【００６９】
本願で開示される例では、ＣＢＢアナライザ３０２及び／又はカーネルアナライザ３０４のいずれも、識別された制約及び／又は要件を、コンパイラインターフェイス３０６を介してグラフコンパイラ２０２へ伝えてよい。
【００７０】
図３で表されている例では、コンパイラインターフェイス３０６は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。本願で開示されるいくつかの例では、コンパイラインターフェイス３０６は、ハードウェア回路構成で実行可能なソフトウェア・アプリケーションプログラミングインターフェイス（Application Programming Interface，ＡＰＩ）を用いて実装されてよい。かような例となるコンパイラインターフェイス３０６は、セレクタ３００と図２のグラフコンパイラ２０２との間の通信を可能にする。その上、コンパイラインターフェイス３０６は、Ｅｔｈｅｒｎｅｔ（登録商標）インターフェイス、ユニバーサルシリアルバス（Universal Serial Bus，ＵＳＢ）、Ｂｌｕｅｔｏｏｔｈ（登録商標）インターフェイス、近距離通信（Near Field Communication，ＮＦＣ）インターフェイス、及び／又はＰＣＩエクスプレスインターフェイスなどの如何なるタイプのインターフェイス標準によっても実装されてよい。コンパイラインターフェイス３０６は、ＣＢＢアナライザ３０２及び／又はカーネルアナライザ３０４のいずれかから入力及び出力条件を取得し、入力及び出力条件をグラフコンパイラ２０２へ送るよう構成される。更に、又は代替的に、コンパイラインターフェイス３０６は、ＣＢＢアナライザ３０２及び／又はカーネルアナライザ３０４によって決定された要件をグラフコンパイラ２０２へ送るよう構成されてもよい。
【００７１】
図４は、図２のグラフコンパイラ２０２を表すブロック図の例である。グラフコンパイラ２０２は、図４に表されるように、例となるグラフインターフェイス４０２、例となる

セレクタインターフェイス４０４、例となる作業負荷アナライザ４０６、例となる実行ファイル生成部４０８、例となるデータストア４１０、及び図２のプラグイン２３６を含む。動作中、グラフインターフェイス４０２、セレクタインターフェイス４０４、作業負荷アナライザ４０６、実行ファイル生成部４０８、データストア４１０、及び／又はプラグイン２３６のいずれも、例となる通信バス４１２を介して通信してよい。図４で、通信バス４１２は、任意の適切な通信方法及び／又は装置（例えば、Ｂｌｕｅｔｏｏｔｈ（登録商標）通信、ＬＡＮ通信、ＷＬＡＮ通信、など）を用いて実装されてよい。

【００７２】

図４で表されている例では、グラフインターフェイス４０２は、グラフ通信のための手段、又はグラフ通信手段である。図４の例では、セレクタインターフェイス４０４は、セレクタ通信のための手段、又はセレクタ通信手段である。図４で表されている例では、作業負荷アナライザ４０６は、作業負荷を解析する手段、又は作業負荷解析手段である。図４の例では、プラグイン２３６は、変換する手段、又は変換手段である。図４の例では、実行ファイル生成部４０８は、実行ファイル生成のための手段、又は実行ファイル生成手段である。図４の例では、データストア４１０は、データを記憶する手段、又はデータ記憶手段である。

【００７３】

図４で表されている例では、グラフインターフェイス４０２は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。その上、グラフインターフェイス４０２は、Ｅｔｈｅｒｎｅｔインターフェイス、ユニバーサルシリアルバス（ＵＳＢ）、Ｂｌｕｅｔｏｏｔｈ（登録商標）インターフェイス、近距離通信（ＮＦＣ）インターフェイス、及び／又はＰＣＩエクスプレスインターフェイスなどの如何なるタイプのインターフェイス標準によっても実装されてよい。グラフインターフェイス４０２は、作業負荷（例えば、図２の作業負荷２０６）が受け取られるかどうかを判定するよう構成される。本願で開示される例では、作業負荷２０６が利用可能である場合に、グラフインターフェイス４０２は、作業負荷２０６をデータストア４１０に格納し得る。

【００７４】

図４で、例となるセレクタインターフェイス４０４は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。その上、セレクタインターフェイス４０４は、Ｅｔｈｅｒｎｅｔインターフェイス、ユニバーサルシリアルバス（ＵＳＢ）、Ｂｌｕｅｔｏｏｔｈ（登録商標）インターフェイス、近距離通信（ＮＦＣ）インターフェイス、及び／又はＰＣＩエクスプレスインターフェイスなどの如何なるタイプのインターフェイス標準によっても実装されてよい。セレクタインターフェイス４０４は、作業負荷２０６を取得することに応答して、作業負荷２０６内の作業負荷ノードごとに１つ以上のセレクタ２０４を生成及び／又は別なふうに供給するよう構成される。その上、セレクタインターフェイス４０４は、１つ以上のセレクタ２０４から入力及び／又は出力条件を取得し及び／又は別なふうに受け取るよう構成される。例えば、セレクタインターフェイス４０４は、アクセラレータ２０８における各ＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０のいずれか）の入力及び／又は出力条件を取得するよう構成される。かような動作において、セレクタインターフェイス４０４は、ＣＢＢを動作させる入力及び出力条件を特定するＣＢＢのジェネリックリストを取得する。他の例では、セレクタインターフェイス４０４は、アクセラレータ２０８における

各カーネル（例えば、カーネルバンク２３２内のいずれかのカーネル、及び／又は任意の適切なカーネル）の入力及び出力条件を取得するよう構成される。かような動作において、セレクタインターフェイス４０４は、カーネルを動作させる入力及び出力条件を特定するカーネルのジェネリックリストを取得する。動作中、セレクタインターフェイス４０４は、１つ以上のセレクタ２０４によって識別された入力及び／又は出力条件をデータストア４１０に格納する。

【００７５】

図４で表されている例では、作業負荷アナライザ４０６は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。作業負荷アナライザ４０６は、作業負荷（例えば、図６の作業負荷２０６）に含まれる作業負荷ノードをパースする。作業負荷アナライザ４０６は、作業負荷ノードを実行するために使用される入力及び出力条件を識別するよう作業負荷ノードをパースする。作業負荷アナライザ４０６は、パースされた作業負荷ノードを、１つ以上のセレクタ２０４による使用のためにセレクタインターフェイス４０４へ、及び／又はプラグイン２３６による使用のためにデータストア４１０へ送信し得る。

【００７６】

図４の例では、プラグイン２３６は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。動作中、プラグイン２３６は、作業負荷アナライザ４０６によって識別された作業負荷ノードをＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０のいずれか）へマッピングするために、セレクタインターフェイス４０４、作業負荷アナライザ４０６、及びデータストア４１０に記憶されているデータと通じるよう構成される。例えば、プラグイン２３６は、識別された入力及び／又は出力条件に基づいて、作業負荷を、アクセラレータ２０８におけるＣＢＢ及び／又はカーネルにマッピングし及び／又は割り当てる。更に、かような例では、プラグイン２３６は、作業負荷ノードを実装するための入力及び出力条件を取得し、かような作業負荷ノードを、同じか又は実質的に類似した入力及び／又は出力条件を同様に含むデバイス（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はカーネルバンク２３２に位置するカーネルのいずれか）に基づいて、実行されるよう割り当てる。このように、プラグイン２３６は、作業負荷ノードを割り当てられている特定のデバイス（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、ＤＳＰ２２０、及び／又はカーネルバンク２３２に位置するカーネルのいずれか）の直接的な知識を有さない。

【００７７】

本願で開示されるいくつかの例では、プラグイン２３６は、どのＣＢＢ及び／又はカーネルが特定の作業負荷ノードを割り当てられ得るかを鑑み及び／又は予測するために、適切なＡＩ技術を用いて実装されてよい。例えば、プラグイン２３６が、データをバックアップすることを示す作業負荷ノードを特定のＣＢＢへ以前に割り当てている場合に、かような作業負荷が今後割り当てられるべきであったならば、プラグインは、それを、データストア４１０に記憶されているデータを解析することから独立した特定のＣＢＢへ割り当て得る。

【００７８】

図４で、例となる実行ファイル生成部４０８は、例えば、ハードウェアプロセッサなどの論理回路によって実装される。なお、如何なる他のタイプの回路構成も、更に、又は代

替的に、使用されてよく、例えば、１つ以上のアナログ又はデジタル回路、論理回路、プログラム可能プロセッサ、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、デジタル信号プロセッサ（ＤＳＰ）などがある。プラグイン２３６が、同様の入力及び／又は出力条件を含むデバイスへ作業負荷ノードを割り当てた後、実行ファイル生成部４０８は、アクセラレータ２０８によって実行されるべき図２の実行ファイル２３０を生成するよう構成される。実行ファイル生成部４０８は更に、実行ファイル２３０をコンフィグレーションコントローラ２２４へ送る。その上、実行ファイル生成部４０８は、アクセラレータ２０８によって実行されるべき１つ以上の実行ファイルを生成してもよい。

【００７９】

図４で表されている例では、データストア４１０は、例えば、フラッシュメモリ、磁気媒体、光学媒体、などのような、データを記憶するための如何なるデバイスによっても実装されてよい。更に、例となるデータストア４１０に記憶されるデータは、例えば、バイナリデータ、コンマ区切りデータ、タブ区切りデータ、構造化されたクエリ言語（ＳＱＬ）構造、などのような、如何なるデータフォーマットにあってもよい。図４で、データストア４１０は、セレクタインターフェイス４０４から取得された入力及び／又は出力条件、グラフインターフェイス４０２から取得された作業負荷（例えば、図２の作業負荷２０６）、及び／又は作業負荷ノードを実行するための入力及び／又は出力条件（例えば、作業負荷アナライザ４０６によって識別された入力及び／又は出力条件）を記憶するよう構成される。データストア４１０は、グラフインターフェイス４０２、セレクタインターフェイス４０４、作業負荷アナライザ４０６、プラグイン２３６、及び／又は実行ファイル生成部４０８のいずれによっても書き込まれ及び／又は読み出されてよい。

【００８０】

図５は、例となる第１ＣＢＢ５０２及び例となる第２ＣＢＢ５０４を用いて実行される作業負荷を表すパイプライン５００を例示する図である。第１ＣＢＢ５０２及び／又は第２ＣＢＢ５０４は、図２の例となるＣＢＢ（例えば、畳み込みエンジン２１４、ＭＭＵ２１６、ＲＮＮエンジン２１８、及び／又はＤＳＰ２２０）であってよい。代替的に、第１ＣＢＢ５０２及び／又は第２ＣＢＢ５０４は、任意の適切なカーネル（例えば、カーネルバンク２３２に位置するカーネル）を用いて実装されてもよい。図５の例では、第１ＣＢＢ５０２は、プロデューサであり、第２ＣＢＢ５０４は、コンシューマである。例となるパイプライン５００は、例となる第１作業負荷ノード５０６、及び例となる第２作業負荷ノード５０８を含む。図５の例では、第１ＣＢＢ５０２は、第１作業負荷ノード５０６を実行するよう構成される。同様に、第２ＣＢＢ５０４は、第２作業負荷ノード５０８を実行するよう構成される。動作中、例となるクレジットマネージャ５１０は、第１作業負荷ノード５０６を実行するために第１クレジット値を第１ＣＢＢ５０２へ供給するよう構成される。例えば、第１クレジット値は、５つのクレジット（バッファ５１２で最初に利用可能なデータスロット）であり、そのようなものとして、第１作業負荷ノード５０６の実行を開始する指示を第１ＣＢＢ５０２に与える。図５で、バッファ５１２は循環バッファである。

【００８１】

図５で表されている例では、第１作業負荷ノード５０６は、バッファ５１２の２つのスロット（データスロットのサブセット）に書き込むことによって実行される。そのようなものとして、第１ＣＢＢ５０２は、バッファ５１２の最初の２つの利用可能なスロットに書き込む。これに応えて、第１ＣＢＢ５０２は、２つのクレジットをクレジットマネージャ５１０へ送る。クレジットマネージャ５１０は、利用可能になると、２つのクレジットを第２ＣＢＢ５０４へ送る。第２ＣＢＢ５０４へ供給された２つのクレジットは、第２作業負荷ノード５０８の実行を開始することを第２ＣＢＢ５０４に示すよう動作する。図５で、第２作業負荷ノード５０８は、バッファ５１２内の次の２つのスロットを先入先出法（First-In First-Out，ＦＩＦＯ）で読み出すことによって実行される。

【００８２】

例となるグラフコンパイラ２０２、例となる１つ以上のセレクタ２０４、例となるセレクタ３００、及び／又は図２のアクセラレータ２０８を実装する方法の例が図３及び／又は図４に表されている一方で、図２、図３及び／又は図４で表されている要素、プロセス、及び／又はデバイスの１つ以上は、組み合わされ、分割され、再配置され、省略され、削除され、及び／又は如何なる他の方法でも実装されてよい。更に、例となるＣＢＢアナライザ３０２、例となるカーネルアナライザ３０４、例となるコンパイラインターフェイス３０６、及び／又はより一般的に、図２及び／又は図３の例となるセレクタ３００及び／又は例となる１つ以上のセレクタ２０４、例となるグラフインターフェイス４０２、例となるセレクタインターフェイス４０４、例となる作業負荷アナライザ４０６、例となる実行ファイル生成部４０８、例となるデータストア４１０、例となるプラグイン２３６、及び／又はより一般的に、図２及び／又は図４の例となるグラフコンパイラ２０２、及び／又は例となるクレジットマネージャ２１０、例となるＣｎＣファブリック２１２、例となる畳み込みエンジン２１４、例となるＭＭＵ２１６、例となるＲＮＮエンジン２１８、例となるＤＳＰ２２０、例となるメモリ２２２、例となるコンフィグレーションコントローラ２２４、例となるカーネルバンク２３２、及び／又はより一般的に、図２の例となるアクセラレータ２０８は、ハードウェア、ソフトウェア、又はファームウェアによって、かつ／あるいは、ハードウェア、ソフトウェア、及び／又はファームウェアの任意の組み合わせによって実装されてよい。よって、例えば、例となるＣＢＢアナライザ３０２、例となるカーネルアナライザ３０４、例となるコンパイラインターフェイス３０６、及び／又はより一般的に、図２及び／又は図３の例となるセレクタ３００及び／又は例となる１つ以上のセレクタ２０４、例となるグラフインターフェイス４０２、例となるセレクタインターフェイス４０４、例となる作業負荷アナライザ４０６、例となる実行ファイル生成部４０８、例となるデータストア４１０、例となるプラグイン２３６、及び／又はより一般的に、図２及び／又は図４の例となるグラフコンパイラ２０２、及び／又は例となるクレジットマネージャ２１０、例となるＣｎＣファブリック２１２、例となる畳み込みエンジン２１４、例となるＭＭＵ２１６、例となるＲＮＮエンジン２１８、例となるＤＳＰ２２０、例となるメモリ２２２、例となるコンフィグレーションコントローラ２２４、例となるカーネルバンク２３２、及び／又はより一般的に、図２の例となるアクセラレータ２０８のいずれも、１つ以上のアナログ若しくは又はデジタル回路、論理回路、プログラム可能プロセッサ、プログラム可能コントローラ、グラフィクス処理ユニット（ＧＰＵ）、デジタル信号プロセッサ（ＤＳＰ）、特定用途向け集積回路（ＡＳＩＣ）、プログラム可能論理デバイス（ＰＬＤ）、及び／又はフィールドプログラマブルゲートアレイ（ＦＰＧＡ）によって実装されてよい。純粋にソフトウェア及び／又はファームウェアの実施をカバーするよう本発明の装置又はシステム請求項のいずれかを読む場合に、例となるＣＢＢアナライザ３０２、例となるカーネルアナライザ３０４、例となるコンパイラインターフェイス３０６、及び／又はより一般的に、図２及び／又は図３の例となるセレクタ３００及び／又は例となる１つ以上のセレクタ２０４、例となるグラフインターフェイス４０２、例となるセレクタインターフェイス４０４、例となる作業負荷アナライザ４０６、例となる実行ファイル生成部４０８、例となるデータストア４１０、例となるプラグイン２３６、及び／又はより一般的に、図２及び／又は図４の例となるグラフコンパイラ２０２、及び／又は例となるクレジットマネージャ２１０、例となるＣｎＣファブリック２１２、例となる畳み込みエンジン２１４、例となるＭＭＵ２１６、例となるＲＮＮエンジン２１８、例となるＤＳＰ２２０、例となるメモリ２２２、例となるコンフィグレーションコントローラ２２４、例となるカーネルバンク２３２、及び／又はより一般的に、図２の例となるアクセラレータ２０８のうちの少なくとも１つは、ソフトウェア及び／又はファームウェアを含むメモリ、デジタルバーサタイルディスク（ＤＶＤ）、コンパクトディスク（ＣＤ）、ブルーレイディスク、などのような非一時的なコンピュータ可読記憶デバイス又は記憶ディスクを含むようこれをもって明示的に定義される。また更に、図２、図３及び／又は図４の例となるグラフコンパイラ２０２、例となる１つ以上のセレクタ２０４、例となるセレクタ３００、及び／又はアクセラレータ２０８は、図２、図３及び／又は図４

で表されているものに加えて、又はそれらの代わりに、１つ以上の要素、プロセス及び／又はデバイスを含んでよく、かつ／あるいは、表されている要素、プロセス、及びデバイスのいずれか又は全ての１つ以上を含んでよい。本願で使用されるように、「～と通信している」との表現及びその変形は、直接的な通信及び／又は１つ以上の中間コンポーネントを介した間接的な通信を包含し、直接的な物理（例えば、有線）通信及び／又は継続的な通信を必要とせず、むしろ、周期的なインターバル、スケジューリングされたインターバル、不規則なインターバル、及び／又は一回限りでの選択的な通信を更に含む。

【００８３】

例となるグラフコンパイラ２０２、例となる１つ以上のセレクタ２０４、例となるセレクタ３００、及び／又はアクセラレータ２０８を実装するための例となるハードウェアロジック、マシン読み出し可能な命令、ハードウェア実装による状態マシン、及び／又はそれらの任意の組み合わせを表すフローチャートは、図６及び／又は図７に示される。マシに読み出し可能な命令は、図８に関連して以下で説明されるプロセッサプラットフォーム８００で例として示されるプロセッサ８１０及び／又はアクセラレータ８１２などのコンピュータプロセッサによって実行される１つ以上の実行可能プログラム又は実行可能プログラムの部分であってよい。プログラムは、ＣＤ－ＲＯＭ、フロッピー（登録商標）ディスク、ハードドライブ、ＤＶＤ、ブルーレイディスク、あるいは、プロセッサ８１０及び／又はアクセラレータ８１２に付随したメモリなどの非一時的なコンピュータ可読記憶媒体に記憶されたソフトウェアにおいて具現されてよいが、プログラム全体及び／又はその部分は、代替的に、プロセッサ８１０やアクセラレータ８１２以外の他のデバイスによって実行され、かつ／あるいは、ファームウェア又は専用ハードウェアにおいて具現されてもよい。更に、例となるプログラムが、図４で表されているフローチャートを参照して記載されるが、例となるグラフコンパイラ２０２、例となる１つ以上のセレクタ２０４、例となるセレクタ３００、及び／又はアクセラレータ２０８を実装する多くの他の方法が代替的に使用されてもよい。例えば、ブロックの実行の順序は変更されてよく、かつ／あるいは、記載されているブロックのいくつかは、変更され、削除され、又は結合されてもよい。更に、又は代替的に、ブロックのいずれか又は全ては、対応する動作をソフトウェア又はファームウェアを実行せずに実行するよう構造化された１つ以上のハードウェア回路（例えば、ディスクリート及び／又は集積アナログ及び／又はデジタル回路構成、ＦＰＧＡ、ＡＳＩＣ、コンパレータ、演算増幅器（ｏｐ－ａｍｐ）、論理回路、など）によって実装されてもよい。

【００８４】

本願で記載されるマシン読み出し可能な命令は、圧縮されたフォーマット、暗号化されたフォーマット、断片化されたフォーマット、コンパイルされたフォーマット、実行可能なフォーマット、パッケージ化されたフォーマット、などのうちの１つ以上で記憶されてよい。本願で記載されるマシン読み出し可能な命令は、マシン読み出し可能な命令を作成し、製造し、及び／又は生成するために利用され得るデータ（例えば、命令の部分、コード、コードの表現、など）として記憶されてよい。例えば、マシン読み出し可能な命令は、１つ以上の記憶デバイス及び／又はコンピュータデバイス（例えば、サーバ）で断片化及び記憶されてよい。マシン読み出し可能な命令は、それらをコンピュータデバイス及び／又は他のマシンによって直接に読み出し可能、解釈可能、及び／又は実行可能にするために、インストール、変更、適応、更新、結合、補足、設定、解読、解凍、アンパッケージ、分配、再配置、コンパイル、などのうちの１つ以上を必要とすることがある。例えば、マシン読み出し可能な命令は、別々のコンピュータデバイスで個々に圧縮、暗号化、及び記憶される複数の部分において記憶されてよく、それらの部分は、解読、解凍、及び結合されるときに、本願で記載されるようなプログラムを実装する実行可能命令の組を形成する。

【００８５】

他の例では、マシン読み出し可能な命令は、それらがコンピュータによって読み出され得る状態で記憶されるが、特定のコンピュータデバイス又は他のデバイスで命令を実行す

るために、ライブラリ（例えば、動的リンクライブラリ（Dynamic Link Library，ＤＬＬ））、ソフトウェア開発キット（Software Development Kit，ＳＤＫ）、アプリケーションプログラミングインターフェイス（ＡＰＩ）などの追加を必要とすることができる。他の例では、マシン読み出し可能な命令は、マシン読み出し可能な命令及び／又は対応するプログラムが全体として又は部分的に実行可能である前に、設定される必要があることがある（例えば、設定記憶、データ入力、ネットワークアドレス記録、など）。よって、開示されるマシン読み出し可能な命令及び／又は対応するプログラムは、記憶されているときに又は別なふうに格納された状態か若しくは伝送中にマシン読み出し可能な命令及び／又はプログラムの特定のフォーマット又は状態に関わらず、かようなマシン読み出し可能な命令及び／又はプログラムを包含することが意図される。

【００８６】

本願で記載されるマシン読み出し可能な命令は、あらゆる過去、現在、又は将来の命令言語、スクリプト言語、プログラミング言語、などによって表現可能である。例えば、マシン読み出し可能な命令は、次の言語：Ｃ、Ｃ＋＋、Ｊａｖａ（登録商標）、Ｃ＃、Ｐｅｒｌ、Ｐｙｔｈｏｎ、ＪａｖａＳｃｒｉｐｔ（登録商標）、ＨｙｐｅｒＴｅｘｔ Ｍａｒｋｕｐ Ｌａｇｕａｇｅ（ＨＴＭＬ）、構造化されたクエリ言語（ＳＱＬ）、Ｓｗｉｆｔ、などのいずれかを用いて表現され得る。

【００８７】

上記の通り、図６及び／又は図７の例となるプロセスは、情報が任意の期間（長期間、永久に、束の間、一時バッファリングのために、及び／又は情報のキャッシングのために）保存されるハードディスクドライブ、フラッシュメモリ、リードオンリーメモリ、コンパクトディスク、デジタルバーサタイルディスク、キャッシュ、ランダムアクセスメモリ及び／又はあらゆる他の記憶デバイス若しくは記憶ディスクなどの非一時的なコンピュータ及び／又はマシン読み出し可能な媒体に記憶された実行可能命令（例えば、コンピュータ及び／又はマシン読み出し可能な命令）を用いて実装されてよい。本願で使用されるように、「非一時的なコンピュータ可読媒体」との語は、あらゆるタイプのコンピュータ可読記憶デバイス及び／又は記憶デバイスを含むよう、かつ、電波信号を除外し、伝送媒体を除外するよう明示的に定義される。

【００８８】

「含む」（including）及び「有する」（comprising）（並びにそれらの全ての形及び時制）は、非限定的な語であるよう本願では使用される。よって、請求項が、プリアンブルとして又はあらゆる種類のクレーム記載の中で「含む」又は「有する」のいかなる形（例えば、有する（comprises）、含む（includes）、有している（comprising）、含んでいる（including）、持っている（having）、など）を用いるとしても、対応する請求項又は記載の範囲から出ることなしに、追加の要素、項目などが存在し得ることが理解されるべきである。本願で使用されるように、「少なくとも」（at least）との表現が、例えば、請求項のプリアンブルにおいて、移行語（transition term）として使用される場合に、それは、「有する」及び「含む」との語が非限定的であるのと同じように非限定的である。「及び／又は」（and/or）は、例えば、Ａ、Ｂ及び／又はなどの形で使用される場合に、（１）Ａのみ、（２）Ｂのみ、（３）Ｃのみ、（４）ＡとＢ、（５）ＡとＣ、（６）ＢとＣ、及び（７）ＡとＢとＣといったＡ、Ｂ、Ｃの任意の組み合わせ又は部分集合を指す。構造、構成要素、項目、対象、及び／又は物事を記載する文脈において本願で使用されるように、「Ａ及びＢの少なくとも１つ」との表現は、（１）少なくとも１つのＡ、（２）少なくとも１つのＢ、及び（３）少なくとも１つのＡ及び少なくとも１つのＢ、のいずれかを含む実施を指すよう意図される。同様に、構造、構成要素、項目、対象、及び／又は物事を記載する文脈において本願で使用されるように、「Ａ又はＢの少なくとも１つ」との表現は、（１）少なくとも１つのＡ、（２）少なくとも１つのＢ、及び（３）少なくとも１つのＡ及び少なくとも１つのＢ、のいずれかを含む実施を指すよう意図される。プロセス、命令、動作、活動及び／又はステップの実施または実行を記載する文脈において本願で使用されるように、「Ａ及びＢの少なくとも１つ」との表現は、（１）少な

くとも１つのＡ、（２）少なくとも１つのＢ、及び（３）少なくとも１つのＡ及び少なくとも１つのＢ、のいずれかを含む実施を指すよう意図される。同様に、プロセス、命令、動作、活動及び／又はステップの実施または実行を記載する文脈において本願で使用されるように、「Ａ又はＢの少なくとも１つ」との表現は、（１）少なくとも１つのＡ、（２）少なくとも１つのＢ、及び（３）少なくとも１つのＡ及び少なくとも１つのＢ、のいずれかを含む実施を指すよう意図される。

【００８９】

本願で使用されるように、単数参照（例えば、「１つ」（a）、「１つ」（an）、「第１」、「第２」、など）は複数を除外しない。「１つ」（a又はan）との語は、そのエンティティの１つ以上を指す。「１つ」（a又はan）、「１つ以上」、及び「少なくとも１つ」との語は、道義的に本願では使用され得る。更に、たとえ個々に挙げられているとしても、複数の手段、要素、又は方法動作は、例えば、単一ユニット又はプロセッサによって、実装されてよい。その上、たとえ個々の特徴が異なる例又は請求項に含まれることがあるとしても、それらは、場合により組み合わされてもよく、異なる例又は請求項における包含は、特徴の組み合わせが実現可能及び／又は有利でないことを暗示しない。

【００９０】

図６は、図２の実行ファイル２３０を生成するために図２、図３及び／又は図４のグラフコンパイラ２０２、セレクタ３００、及び／又は１つ以上のセレクタ２０４を実装するよう実行され得るプロセス６００を表すフローチャートである。図６の表される例で、グラフインターフェイス４０２（図４）は、作業負荷２０６が受け取られる及び／又は別なふうに利用可能であるかどうかを判定する（ブロック６０２）。作業負荷２０６が受け取られない及び／又は別なふうに利用可能でないとグラフインターフェイス４０２が決定する（例えば、ブロック６０２の制御が否定（ＮＯ）の結果を返す）ことに応答して、プロセス６００は待ち続ける。代替的に、作業負荷２０６が受け取られる及び／又は別なふうに利用可能であるとグラフインターフェイス４０２が決定する（例えば、ブロック６０２の制御が肯定（ＹＥＳ）の結果を返す）場合に、次いで、作業負荷アナライザ４０６（図４）は、作業負荷ノードを識別するよう作業負荷２０６をパースする（ブロック６０４）。

【００９１】

これに応えて、セレクタインターフェイス４０４（図４）は、作業負荷ノードごとにセレクタ（例えば、図２の１つ以上のセレクタ２０４）を生成する（ブロック６０６）。ＣＢＢアナライザ３０２（図３）は更に、関連するＣＢＢの入力及び出力条件を取得及び／又は別なふうに識別する（ブロック６０８）。これに応えて、セレクタインターフェイス４０４は、生成された全てのセレクタが各々の入力及び／又は出力条件を供給したかどうかを判定し、そのようなものとして、解析すべき更なるＣＢＢがあるかどうかを判定する（ブロック６１０）。解析すべき更なるＣＢＢがあるとセレクタインターフェイス４０４が決定する（ブロック６１０の制御が肯定の結果を返す）場合に、次いで、制御はブロック６０８に戻る。代替的に、解析すべき更なるＣＢＢがないとセレクタインターフェイス４０４が決定する（ブロック６１０の制御が否定の結果を返す）場合に、次いで、カーネルアナライザ３０４（図３）は更に、関連するカーネルの入力及び出力条件を取得及び／又は別なふうに識別する（ブロック６１２）。これに応えて、セレクタインターフェイス４０４は、生成された全てのセレクタが各々の入力及び／又は出力条件を供給したかどうかを判定し、そのようなものとして、解析すべき更なるカーネルがあるかどうかを判定する（ブロック６１４）。解析すべき更なるカーネルがあるとセレクタインターフェイス４０４が決定する（例えば、ブロック６１４の制御が肯定の結果を返す）場合に、次いで、制御はブロック６１２に戻る。代替的に、解析すべき更なるカーネルがないとセレクタインターフェイス４０４が決定する（例えば、ブロック６１４の制御が否定の結果を返す）場合に、次いで、プラグイン２３６（図２及び／又は図４）は、セレクタ（例えば、図１の１つ以上のセレクタ２０４）によって識別された入力及び出力条件に基づいて、作業負荷ノードをＣＢＢ及び／又はカーネルにマッピングする（ブロック６１６）。

【００９２】

実行ファイル生成部４０８（図４）は、次いで、実行ファイル２３０を生成する（ブロック６１８）。実行ファイル生成部４０８は更に、実行ファイル２３０をコンフィグレーションコントローラ２２４へ送る（ブロック６２０）。本願で開示される他の例では、ブロック６１８の実行に応答して、実行ファイル生成部４０８は、外部及び／又は内部デプロイメントシステム（例えば、図１のシステム１００）での後の使用のために、実行ファイル２３０をデータストア４１０に格納してよい。図６の表される例で、グラフコンパイラ２０２は、動作を続けるべきかどうかを判断する（ブロック６２２）。動作を続けるとグラフコンパイラ２０２が決定する（例えば、ブロック６２２の制御が肯定の結果を返す）場合に、次いで、制御はブロック６０２に戻り、グラフインターフェイス４０２は、作業負荷２０６が受け取られる及び／又は別なふうに利用可能であるかどうかを判定する。例えば、グラフコンパイラ２０２は、更なる作業負荷が利用可能である場合に、かつ／あるいは、新しいＣＢＢ及び／又はカーネルがアクセラレータ２０８に含まれる場合に、動作し続けると決定してよい。

【００９３】

代替的に、動作が続くべきでないとグラフコンパイラ２０２が決定する（ブロック６２２の制御が否定の結果を返す）場合に、次いで、図６のプロセス６００は終了する。すなわち、プロセス６００は、これ以上作業負荷が利用可能でない場合に停止し得る。

【００９４】

図７は、図２の実行ファイル２３０の実行を助けるために図２のクレジットマネージャ２１０及び／又はコンフィグレーションコントローラ２２４を実装するよう実行され得るプロセス７００を表すフローチャートである。図７で、コンフィグレーションコントローラ２２４（図２）は、実行ファイル２３０がグラフコンパイラ２０２から受け取られる及び／又は別なふうに利用可能であるかどうかを判定する（ブロック７０２）。実行ファイル２３０が受け取られない及び／又は別なふうに利用可能でないとコンフィグレーションコントローラ２２４が決定する（例えば、ブロック７０２の制御が否定の結果を返す）場合に、次いで、プロセス７００は待ち続ける。代替的に、実行ファイル２３０が受け取られる及び／又は別なふうに利用可能であるとコンフィグレーションコントローラ２２４が決定する（ブロック７０２の制御が肯定の結果を返す）場合に、次いで、コンフィグレーションコントローラ２２４は、生成作業負荷ノード及び消費作業負荷ノードを実行するよう各々のＣＢＢを識別するために、生成作業負荷ノード及び消費作業負荷ノードを識別するよう実行ファイル２３０をパースする（ブロック７０４）。これに応えて、コンフィグレーションコントローラ２２４は、生成作業負荷ノードを第１の選択されたＣＢＢ（例えば、畳み込みエンジン２１４）へ送る（ブロック７０６）。同様に、コンフィグレーションコントローラ２２４は、消費作業負荷ノードを第２の選択されたＣＢＢ（例えば、ＤＳＰ２２０）へ送る（ブロック７０８）。

【００９５】

応答して、又は並行して、クレジットマネージャ２１０は、生成作業負荷ノードの実行を開始するよう第１の選択されたＣＢＢ（例えば、畳み込みエンジン２１４）へクレジットを分配する（ブロック７１０）。本願で開示されるいくつかの例では、ブロック７０６、７０８、及び／又は７１０の動作は、全ての生成作業負荷ノード及び／又は消費作業負荷ノードに対して作用する。例えば、クレジットマネージャ２１０は、全ての生成作業負荷ノードに対応するクレジットを全ての対応する生成ＣＢＢへ分配する。かような例では、ランタイム中の同期化は、対応するＣＢＢ及び／又はクレジットマネージャ２１０の間の通信に基づいて達成される。クレジットはクレジットマネージャ２１０へ及びそれから送出されるので、クレジットマネージャ２１０は、クレジットが第１の選択されたＣＢＢ（例えば、畳み込みエンジン２１４）から受け取られるかどうかを判定する（ブロック７１２）。クレジットが第１の選択されたＣＢＢ（例えば、畳み込みエンジン２１４）から取得も送出されていないとクレジットマネージャ２１０が決定する（例えば、ブロック７１２の制御が否定の結果を返す）場合に、次いで、プロセス７００は待ち続ける。代替的に、クレジットが第１の選択されたＣＢＢ（例えば、畳み込みエンジン２１４）から取得

及び／又は送出されたとクレジットマネージャ２１０が決定する（例えば、ブロック７１２の制御が肯定の結果を返す）場合に、クレジットマネージャ２１０は、消費作業負荷ノードの実行を開始するようクレジットを第２の選択されたＣＢＢ（例えば、ＤＳＰ２２０）へ分配する（ブロック７１４）。

【００９６】

これに応えて、クレジットマネージャ２１０は、クレジットが第２の選択されたＣＢＢ（例えば、ＤＳＰ２２０）から受け取られるかどうかを判定する（ブロック７１６）。クレジットが第２の選択されたＣＢＢ（例えば、ＤＳＰ２２０）から取得も送出もされていないとクレジットマネージャ２１０が決定する（例えば、ブロック７１６の制御が否定の結果を返す）場合に、次いで、プロセス７００は待ち続ける。代替的に、クレジットが第２の選択されたＣＢＢ（例えば、ＤＳＰ２２０）から取得及び／又は送出されたとクレジットマネージャ２１０が決定する（ブロック７１６の制御が肯定の結果を返す）場合に、次いで、クレジットマネージャ２１０は、生成作業負荷の実行を続けるようクレジットを第１の選択されたＣＢＢ（例えば、畳み込みエンジン２１４）へ分配する（ブロック７１８）。

【００９７】

クレジットマネージャ２１０は、作業負荷ノード（例えば、生成作業負荷ノード又は消費作業負荷ノード）の実行が完了しているかどうかを判定する（ブロック７２０）。本願で開示されるいくつかの例では、クレジットマネージャ２１０は、バッファの生成されたクレジットを数えることに基づいて、作業負荷ノードの実行が完了しているかどうかを判定してよい。例えば、クレジットマネージャ２１０は、プロデューサとして振る舞うＣＢＢ（例えば、図５の第１ＣＢＢ５０２）が、対応する作業負荷ノードを実行及び／又は別なふうに処理する間に、５０個のクレジットを生成すべきであることを実行ファイル２３０から知り得る。従って、クレジットマネージャ２１０は、生成作業負荷ノード（例えば、第１ＣＢＢ５０２）から５０個のクレジットを取得し及び／又は別なふうに受け取ることに応答して、作業負荷の実行が完了していると決定し得る。作業負荷ノード（例えば、生成作業負荷ノード又は消費作業負荷ノード）の実行が完了してないとクレジットマネージャ２１０が決定する（ブロック７２０の制御が否定の結果を返す）場合に、次いで、制御はブロック７１２に戻り、クレジットマネージャ２１０は、クレジットが第１の選択されたＣＢＢ（例えば、畳み込みエンジン２１４）から受け取られるかどうかを判定する。本願で開示される他の例では、作業負荷ノード（例えば、生成作業負荷ノード又は消費作業負荷ノード）の実行が完了してないと（ブロック７２０の制御が否定の結果を返す）、かつ、生成作業負荷ノードの実行が完了しているとクレジットマネージャ２１０が決定する場合に、次いで、制御は、消費作業負荷ノードの実行を完了するためにブロック７１４へ進んでよい。

【００９８】

代替的に、作業負荷ノード（例えば、生成作業負荷ノード又は消費作業負荷ノード）の実行が完了しているとクレジットマネージャ２１０が決定する（ブロック７２０の制御が肯定の結果を返す）場合に、次いで、コンフィグレーションコントローラ２２４は、更なる生成及び消費作業負荷ノードが利用可能であるかどうかを判定する（ブロック７２２）。更なる生成及び消費作業負荷ノードが利用可能であるとコンフィグレーションコントローラ２２４が決定する（例えば、ブロック７２２の制御が肯定の結果を返す）場合に、制御はブロック７０４に戻る。代替的に、利用可能な更なる生成又は消費作業負荷ノードがないとコンフィグレーションコントローラ２２４が決定する（例えば、ブロック７２２の制御が否定の結果を返す）場合に、次いで、プロセス７００は停止する。

【００９９】

図８は、図２、図３及び／又は図４の例となるグラフコンパイラ２０２、例となる１つ以上のセレクタ２０４、例となるセレクタ３００、及び／又はアクセラレータ２０８を実装するように図６及び／又は図７の命令を実行するよう構造化されたプロセッサプラットフォーム８００（例えば、コンパイル結合型デプロイメントシステム）の例のブロック図

である。代替的に、本願で開示されるいくつかの例では、例となるグラフコンパイラ２０２、例となる１つ以上のセレクタ２０４、及び／又は例となるセレクタ３００は、例となるアクセラレータ２０８に比べて図６の命令を実行するよう構造化された別個のコンパイルシステム（例えば、コンパイルプロセッサ）で動作可能であってよい。かような例となる分離システム動作では、アクセラレータ２０８は、コンパイルシステムに比べて図７の命令を実行するよう構造化された別個のデプロイメントシステム（例えば、デプロイメントプロセッサ）で実行ファイルを実行するよう動作可能であってよい。プロセッサプラットフォーム８００は、例えば、サーバ、パーソナルコンピュータ、ワークステーション、自己学習マシン（例えば、ニューラルネットワーク）、モバイル機器（例えば、携帯電話機、スマートフォン、ｉＰａｄ（登録商標）などのタブレット）、パーソナルデジタルアシスタント（ＰＤＡ）、インターネットアプライアンス、ゲーム機、パーソナルビデオレコーダ、セットトップボックス、ヘッドセット若しくは他のウェアラブル機器、又はあらゆる他のタイプのコンピュータデバイスであることができる。

【０１００】

表されている例のプロセッサプラットフォーム８００は、プロセッサ８１０及びアクセラレータ８１２を含む。表されている例のプロセッサ８１０は、ハードウェアである。例えば、プロセッサ８１０は、１つ以上の集積回路、論理回路、マイクロプロセッサ、ＧＰＵ、ＤＳＰ、又は何らかの望ましいファミリ若しくは製造業者からのコントローラによって実装可能である。ハードウェアプロセッサは、半導体ベース（例えば、シリコンベース）のデバイスであってよい。その上、アクセラレータ８１２は、例えば、１つ以上の集積回路、論理回路、マイクロプロセッサ、ＧＰＵ、ＤＳＰ、ＦＰＧＡ、ＶＰＵ、コントローラ、及び／又は何らかの望ましいファミリ若しくは製造業者からの他のＣＢＢによって実装可能である。表されている例のアクセラレータ８１２は、ハードウェアである。ハードウェアアクセラレータは、半導体ベース（例えば、シリコンベース）のデバイスであってよい。この例では、アクセラレータ８１２は、例となるクレジットマネージャ２１０、例となるＣｎＣファブリック２１２、例となる畳み込みエンジン２１４、例となるＭＭＵ２１６、例となるＲＮＮエンジン２１８、例となるＤＳＰ２２０、例となるメモリ２２２、例となるコンフィグレーションコントローラ２２４、及び／又は例となるカーネルバンク２３２を実装する。この例では、プロセッサは、例となるＣＢＢアナライザ３０２、例となるカーネルアナライザ３０４、例となるコンパイラインターフェイス３０６、及び／又はより一般的に、図２及び／又は図３の例となるセレクタ３００及び／又は例となる１つ以上のセレクタ２０４、例となるグラフインターフェイス４０２、例となるセレクタインターフェイス４０４、例となる作業負荷アナライザ４０６、例となる実行ファイル生成部４０８、例となるデータストア４１０、例となるプラグイン２３６、及び／又はより一般的に、図２及び／又は図４の例となるグラフコンパイラ２０２、及び／又は例となるクレジットマネージャ２１０、例となるＣｎＣファブリック２１２、例となる畳み込みエンジン２１４、例となるＭＭＵ２１６、例となるＲＮＮエンジン２１８、例となるＤＳＰ２２０、例となるメモリ２２２、例となるコンフィグレーションコントローラ２２４、例となるカーネルバンク２３２、及び／又はより一般的に、図２の例となるアクセラレータ２０８を実装する。

【０１０１】

表されている例のプロセッサ８１０は、ローカルメモリ８１１（例えば、キャッシュ）を含む。表されている例のプロセッサ８１０は、バス８１８を介して、揮発性メモリ８１４及び不揮発性メモリ８１６を含むメインメモリと通信している。更に、表されている例のアクセラレータ８１２は、ローカルメモリ８１３（例えば、キャッシュ）を含む。表されている例のアクセラレータ８１２は、バス８１８を介して、揮発性メモリ８１４及び不揮発性メモリ８１６を含むメインメモリと通信している。揮発性メモリ８１４は、同期型動的ランダムアクセスメモリ（ＳＤＲＡＭ）、動的ランダムアクセスメモリ（ＤＲＡＭ）、ＲＡＭＢＵＳ（登録商標）動的ランダムアクセスメモリ（ＲＤＲＡＭ（登録商標））及び／又はあらゆる他のタイプの欄無アクセスメモリデバイスによって実装されてよい。不

揮発性メモリ８１６は、フラッシュメモリ及び／又はあらゆる他の望ましいタイプのメモリデバイスによって実装されてよい。メインメモリ８１４、８１６へのアクセスは、メモリコントローラによって制御される。

【０１０２】

表されている例のプロセッサプラットフォーム８００はまた、インターフェイス回路８２０を含む。インターフェイス回路８２０は、Ｅｔｈｅｒｎｅｔインターフェイス、ユニバーサルシリアルバス（ＵＳＢ）、Ｂｌｕｅｔｏｏｔｈ（登録商標）インターフェイス、近距離通信（ＮＦＣ）インターフェイス、及び／又はＰＣＩエクスプレスインターフェイスなどの如何なるタイプのインターフェイスによっても実装されてよい。

【０１０３】

表されている例では、１つ以上の入力デバイス８２２がインターフェイス回路８２０へ接続されている。入力デバイス８２２は、ユーザがプロセッサ８１０及び／又はアクセラレータ８１２にデータ及び／又はコマンドを入力することを可能にする。入力デバイスは、例えば、オーディオセンサ、マイクロホン、カメラ（静止若しくはビデオ）、キーボード、ボタン、マウス、タッチスクリーン、トラックパッド、トラックボール、アイソポイント（isopoint）及び／又は音声認識システムによって実装可能である。

【０１０４】

１つ以上の出力デバイス８２４も、表されている例のインターフェイス回路８２０へ接続されている。出力デバイス８２４は、例えば、表示デバイス（例えば、発光ダイオード（ＬＥＤ）、有機発光ダイオード（ＯＬＥＤ）、液晶ディスプレイ（ＬＣＤ）、陰極線管ディスプレイ（ＣＲＴ）、インプレイススイッチング（In-Place Switching，ＩＰＳ）ディスプレイ、タッチスクリーン、など）、触覚出力デバイス、プリンタ及び／又はスピーカによって実装可能である。表されている例のインターフェイス回路８２０は、よって、通常は、グラフィクスドライバカード、グラフィクスドライバチップ及び／又はグラフィクスドライバプロセッサを含む。

【０１０５】

表されている例のインターフェイス回路８２０はまた、ネットワーク８２６を介した外部のマシン（例えば、あらゆる種類のコンピュータデバイス）とのデータの交換を助けるよう、送信器、受信器、トランシーバ、モデム、住宅用ゲートウェイ、無線アクセスポイント、及び／又はネットワークインターフェイスなどの通信デバイスを含む。通信は、例えば、Ｅｔｈｅｒｎｅｔ接続、デジタル加入者回線（ＤＳＬ）接続、電話回線接続、同軸ケーブルシステム、衛星システム、ラインオブサイト（Line-of-Site）無線システム、セルラー電話システム、などを介することができる。

【０１０６】

表されている例のプロセッサプラットフォーム８００はまた、ソフトウェア及び／又はデータを記憶する１つ以上の大容量記憶デバイス８２８を含む。かような大容量記憶デバイス８２８の例には、フロッピーディスクドライブ、ハードドライブディスク、コンパクトディスクドライブ、ブルーレイディスクドライブ、レイド（ＲＡＩＤ，Redundant Array of Independent Disks）システム、及びデジタルバーサタイルディスク（ＤＶＤ）ドライブがある。

【０１０７】

図６及び／又は図７のマシン実行可能な命令８３２は、大容量記憶デバイス８２８に、揮発性メモリ８１４に、不揮発性メモリ８１６に、及び／又はＣＤ若しくはＤＶＤなどのリムーバブル式非一時コンピュータ可読記憶媒体に記憶されてよい。

【０１０８】

以上より、例となる方法、装置及び製品が開示されており、それらはアクセラレータにおいてヘテロジニアスコンポーネントを設定することが理解されるだろう。開示されている方法、装置及び製品は、作業負荷内の作業負荷ノードごとにセレクタを生成及び／又は別なふうに提供することによって、コンピュータデバイスを使用する効率を改善する。そのようなものとして、開示されている方法、装置及び製品は、グラフコンパイラが、アク

セラレータにおける夫々のヘテロジニアス計算構築ブロック及び／又はカーネルについて個別的に設定される必要なしに、実行ファイルを生成することを可能にする。その上、本願で開示される例は、アクセラレータにおけるヘテロジニアス計算構築ブロック及び／又はカーネルからクレジットを分配し及び／又は受け取るクレジットマネージャを含む。このようにして、計算構築ブロック及び／又はカーネルは、センターファブリック及びクレジットマネージャを通じて他のヘテロジニアス計算構築ブロック及び／又はカーネルと通信することができる。本願で開示される例は、グラフコンパイラがアクセラレータにおける任意の数のヘテロジニアス計算構築ブロック及び／又はカーネルについて作業負荷（例えば、受け取られグラフ）を効率的にマッピングすることを可能にする。本願で開示される例は、同様に、グラフジェネレータが、更なる計算構築ブロック及び／又はカーネルが後にアクセラレータに含まれる場合に、あるいは、現在の計算構築ブロック及び／又はカーネルが変更又は調整される場合に、受け取られる作業負荷（例えば、グラフ）を効率的にマッピングすることを可能にする。開示されている方法、装置及び製品は、従って、コンピュータの機能の１つ以上の改善を対象としている。

【０１０９】

アクセラレータにおいてヘテロジニアスコンポーネントを設定するための方法及び装置に対する例となる方法、装置、システム、及び製品が本願では開示されている。更なる例及びそれらの組み合わせは、次を含む。

【０１１０】

例１は、アクセラレータにおいてヘテロジニアスコンポーネントを設定する装置であって、

作業負荷において作業負荷ノードを識別し、該作業負荷ノードについてセレクタを生成するグラフコンパイラと、

計算構築ブロックの入力条件及び出力条件を識別する前記セレクタと

を有し、

前記グラフコンパイラは、前記識別された入力条件及び出力条件を前記セレクタから取得することに応答して、前記作業負荷ノードを前記計算構築ブロックにマッピングする、

装置を含む。

【０１１１】

例２は、例１の装置であって、前記グラフコンパイラが、前記作業負荷において第２作業負荷ノードを識別し、該第２作業負荷ノードについて第２セレクタを生成する、装置を含む。

【０１１２】

例３は、例２の装置であって、前記第２セレクタが、カーネルの第２入力条件及び第２出力条件を識別する、装置を含む。

【０１１３】

例４は、例１の装置であって、前記作業負荷が、前記グラフコンパイラによって取得された前記作業負荷ノードを含むグラフである、装置を含む。

【０１１４】

例５は、例１の装置であって、前記入力条件が、前記計算構築ブロックの入力要件に対応し、前記出力条件が、前記計算構築ブロックの実行の結果に対応する、装置を含む。

【０１１５】

例６は、例１の装置であって、前記グラフコンパイラが、前記作業負荷ノードを前記計算構築ブロックにマッピングすることに応答して、実行ファイルを生成する、装置を含む。

【０１１６】

例７は、例１の装置であって、前記グラフコンパイラが、前記計算構築ブロックに対する前記作業負荷ノードのマッピングを可能にするよう、前記識別された入力条件及び出力条件に基づいて、前記作業負荷ノードと前記計算構築ブロックとの間の変換レイヤを形成するプラグインを更に含む、装置を含む。

【０１１７】

例８は、実行される場合に、少なくとも１つのプロセッサに、少なくとも
作業負荷において作業負荷ノードを識別することと
前記作業負荷ノードについて、該作業負荷ノードを実行するための計算構築ブロックに関連したセレクタを生成することと、
前記計算構築ブロックの入力条件及び出力条件を識別することと、
前記識別された入力条件及び出力条件を取得することに応答して、前記作業負荷ノードを前記計算構築ブロックにマッピングすることと
を実行させる命令を有する少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。

【０１１８】

例９は、例８の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記命令が、実行される場合に、前記少なくとも１つのプロセッサに、更に
前記作業負荷において第２作業負荷ノードを識別することと、
前記第２作業負荷ノードについて第２セレクタを生成することと
を実行させる、少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。

【０１１９】

例１０は、例９の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記命令が、実行される場合に、前記少なくとも１つのプロセッサに、更に
カーネルの第２入力条件及び第２出力条件を識別することを実行させる、
少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。

【０１２０】

例１１は、例８の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記作業負荷が、前記作業負荷ノードを含むグラフである、少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。

【０１２１】

例１２は、例８の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記入力条件が、前記計算構築ブロックの入力要件に対応し、前記出力条件が、前記計算構築ブロックの実行の結果に対応する、少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。

【０１２２】

例１３は、例８の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記命令が、実行される場合に、前記少なくとも１つのプロセッサに、更に
前記作業負荷ノードを前記計算構築ブロックにマッピングすることに応答して実行ファイルを生成することを実行させる、
少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。

【０１２３】

例１４は、例８の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記命令が、実行される場合に、前記少なくとも１つのプロセッサに、更に
前記計算構築ブロックに対する前記作業負荷ノードのマッピングを可能にするよう、前記識別された入力条件及び出力条件に基づいて、前記作業負荷ノードと前記計算構築ブロックとの間の変換レイヤを形成することを実行させる、
少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。

【０１２４】

例１５は、
作業負荷において作業負荷ノードを識別し、該作業負荷ノードについて、該作業負荷ノードを実行するための計算構築ブロックに関連した選択手段を生成するコンパイル手段と、
前記計算構築ブロックの入力条件及び出力条件を識別する前記選択手段と
を有し、
前記コンパイル手段は更に、前記識別された入力条件及び出力条件を取得することに応答して、前記作業負荷ノードを前記計算構築ブロックにマッピングする、

装置を含む。
【０１２５】
例１６は、例１５の装置であって、前記コンパイル手段が更に、前記作業負荷において第２作業負荷ノードを識別し、該第２作業負荷ノードについて第２選択手段を生成する、装置を含む。
【０１２６】
例１７は、例１６の装置であって、前記第２選択手段が更に、カーネルの第２入力条件及び第２出力条件を識別する、装置を含む。
【０１２７】
例１８は、例１５の装置であって、前記作業負荷が、前記作業負荷ノードを含むグラフである、装置を含む。
【０１２８】
例１９は、例１５の装置であって、前記入力条件が、前記計算構築ブロックの入力要件に対応し、前記出力条件が、前記計算構築ブロックの実行の結果に対応する、装置を含む。
【０１２９】
例２０は、例１５の装置であって、前記コンパイル手段が更に、前記作業負荷ノードを前記計算構築ブロックにマッピングすることに応答して実行ファイルを生成する、装置を含む。
【０１３０】
例２１は、例１５の装置であって、前記コンパイル手段が更に、前記計算構築ブロックに対する前記作業負荷ノードのマッピングを可能にするよう、前記識別された入力条件及び出力条件に基づいて、前記作業負荷ノードと前記計算構築ブロックとの間の変換レイヤを形成する、装置を含む。
【０１３１】
例２２は、アクセラレータにおいてヘテロジニアスコンポーネントを設定する方法であって、
作業負荷において作業負荷ノードを識別することと、
前記作業負荷ノードについて、該作業負荷ノードを実行するための計算構築ブロックに関連したセレクタを生成することと、
前記計算構築ブロックの入力条件及び出力条件を識別することと、
前記識別された入力条件及び出力条件を取得することに応答して前記作業負荷ノードを前記計算構築ブロックにマッピングすることと
を有する方法を含む。
【０１３２】
例２３は、例２２の方法であって、
前記作業負荷において第２作業負荷ノードを識別することと、
前記第２作業負荷ノードについて第２セレクタを生成することと
を更に含む方法を含む。
【０１３３】
例２４は、例２３の方法であって、カーネルの第２入力条件及び第２出力条件を識別することを更に含む方法を含む。
【０１３４】
例２５は、例２２の方法であって、前記作業負荷が、前記作業負荷ノードを含むグラフである、方法を含む。
【０１３５】
例２６は、例２２の方法であって、前記入力条件が、前記計算構築ブロックの入力要件に対応し、前記出力条件が、前記計算構築ブロックの実行の結果に対応する、方法を含む。
【０１３６】
例２７は、例２２の方法であって、前記作業負荷ノードを前記計算構築ブロックにマッピングすることに応答して実行ファイルを生成することを更に含む方法を含む。

【０１３７】
例２８は、例２２の方法であって、前記計算構築ブロックに対する前記作業負荷ノードのマッピングを可能にするよう、前記識別された入力条件及び出力条件に基づいて、前記作業負荷ノードと前記計算構築ブロックとの間の変換レイヤを形成することを更に含む方法を含む。
【０１３８】
例２９は、ヘテロジニアスコンポーネントを動作させる装置であって、
多数のデータスロットを含むバッファと、
クレジットマネージャと、
第１クレジット値を有し、第１作業負荷ノードを実行し、該第１作業負荷ノードを実行することに応答して、前記多数のデータスロットのサブセットにデータを書き込み、前記第１クレジット値よりも小さい第２クレジット値を前記クレジットマネージャへ送る第１計算構築ブロックと、
前記クレジットマネージャから前記第２クレジット値を受け取ることに応答して、前記多数のデータスロットの前記サブセット内のデータを読み出し、第２作業負荷ノードを実行する第２計算構築ブロックと
を有する装置を含む。
【０１３９】
例３０は、例２９の装置であって、前記第１作業負荷ノードを供給するよう制御メッセージ及び設定メッセージを前記第１計算構築ブロックへ送るコントローラを更に含む装置を含む。
【０１４０】
例３１は、例３０の装置であって、前記コントローラが、前記第１作業負荷ノードを前記第１計算構築ブロックへ送り、前記第２作業負荷ノードを前記第２計算構築ブロックへ送る、装置を含む。
【０１４１】
例３２は、例２９の装置であって、前記クレジットマネージャが更に、前記第１作業負荷ノードの実行が完了しているかどうかを判定する、装置を含む。
【０１４２】
例３３は、例２９の装置であって、前記第２計算構築ブロックが更に、前記第２クレジット値よりも小さい第３クレジット値を前記クレジットマネージャへ送る、装置を含む。
【０１４３】
例３４は、例３３の装置であって、前記クレジットマネージャが更に、前記第３クレジット値を前記第１計算構築ブロックへ送る、装置を含む。
【０１４４】
例３５は、実行される場合に、少なくとも１つのプロセッサに、少なくとも
第１作業負荷ノードを実行することと、
前記第１作業負荷ノードを実行することに応答して、第１クレジット値を用いて多数のデータスロットにデータを書き込むことと、
前記第１クレジット値よりも小さい第２クレジット値をクレジットマネージャへ送ることと、
前記クレジットマネージャから前記第２クレジット値を受け取ることに応答して、前記第２クレジット値を用いて前記多数のデータスロット内のデータを読み出すことと、
第２作業負荷ノードを実行することと
を実行させる命令を有する少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。
【０１４５】
例３６は、例３５の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記命令が、実行されるときに更に、前記少なくとも１つのプロセッサに、前記第１作業負荷ノードを供給するよう制御メッセージ及び設定メッセージを送らせる、少なくとも１

つの非一時的なコンピュータ可読記憶媒体を含む。
【０１４６】
例３７は、例３６の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記命令が、実行されるときに更に、前記少なくとも１つのプロセッサに、前記第１作業負荷ノードを第１計算構築ブロックへ送らせ、第２作業負荷ノードを前記第２計算構築ブロックへ送らせる、少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。
【０１４７】
例３８は、例３５の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記命令が、実行されるときに更に、前記少なくとも１つのプロセッサに、前記第１作業負荷ノードの実行が完了しているかどうかを判定させる、少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。
【０１４８】
例３９は、例３５の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記命令は、実行されるときに更に、前記少なくとも１つのプロセッサに、前記第２クレジット値よりも小さい第３クレジット値を前記クレジットマネージャへ送らせる、少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。
【０１４９】
例４０は、例３９の少なくとも１つの非一時的なコンピュータ可読記憶媒体であって、前記命令は、実行されるときに更に、前記少なくとも１つのプロセッサに、前記第３クレジット値を計算構築ブロックへ送らせる、少なくとも１つの非一時的なコンピュータ可読記憶媒体を含む。
【０１５０】
例４１は、
第１作業負荷ノードを実行し、該第１作業負荷ノードを実行することに応答して、第１クレジット値を用いて多数のデータスロットにデータを書き込み、前記第１クレジット値よりも小さい第２クレジット値をクレジット管理手段へ送る第１計算手段と、
前記クレジット管理手段から前記第２クレジット値を受け取ることに応答して、前記第２クレジット値を用いて前記多数のデータスロット内のデータを読み出し、第２作業負荷ノードを実行する第２計算手段と
を有する装置を含む。
【０１５１】
例４２は、例４１の装置であって、前記第１作業負荷ノードを供給するよう制御メッセージ及び設定メッセージを前記第１計算手段へ送る制御手段を更に含む装置を含む。
【０１５２】
例４３は、例４２の装置であって、前記制御手段が更に、前記第１作業負荷ノードを前記第１計算手段へ送り、前記第２作業負荷ノードを前記第２計算手段へ送る、装置を含む。
【０１５３】
例４４は、例４１の装置であって、前記クレジット管理手段が更に、前記第１作業負荷ノードの実行が完了しているかどうかを判定する、装置を含む。
【０１５４】
例４５は、例４１の装置であって、前記第２計算手段が更に、前記第２クレジット値よりも小さい第３クレジット値を前記クレジット管理手段へ送る、装置を含む。
【０１５５】
例４６は、例４５の装置であって、前記クレジット管理手段が更に、前記第３クレジット値を前記第１計算手段へ送る、装置を含む。
【０１５６】
例４７は、ヘテロジニアスコンポーネントを動作させる方法であって、
第１作業負荷ノードを実行することと、
前記第１作業負荷ノードを実行することに応答して、第１クレジット値を用いて多数のデータスロットにデータを書き込むことと、

前記第１クレジット値よりも小さい第２クレジット値をクレジットマネージャへ送ることと、
前記クレジットマネージャから前記第２クレジット値を受け取ることに応答して、前記多数のデータスロット内のデータを読み出すことと、
第２作業負荷ノードを実行することと
を有する方法を含む。

【０１５７】

例４８は、例４７の方法であって、前記第１作業負荷ノードを供給するよう制御メッセージ及び設定メッセージを計算構築ブロックへ送ることを更に含む方法を含む。

【０１５８】

例４９は、例４７の方法であって、前記第１作業負荷ノードを第１計算構築ブロックへ送り、前記第２作業負荷ノードを第２計算構築ブロックへ送ることを更に含む方法を含む。

【０１５９】

例５０は、例４７の方法であって、前記第１作業負荷ノードの実行が完了しているかどうかを判定することを更に含む方法を含む。

【０１６０】

例５１は、例４７の方法であって、前記第２クレジット値よりも小さい第３クレジット値を前記クレジットマネージャへ送ることを更に含む方法を含む。

【０１６１】

例５２は、例５１の方法であって、前記第３クレジット値を計算構築ブロックへ送ることを更に含む方法を含む。

【０１６２】

特定の例となる方法、装置及び製品が本願で開示されてきたが、本特許がカバーする範囲はそれらに制限されない。対照的に、本特許は、本特許の請求の範囲に含まれる全ての方法、装置及び製品をカバーする。

【０１６３】

これによって、続く特許請求の範囲は、この参照によってこの詳細な説明に組み込まれ、各請求項は、本開示の別個の実施形態として独立している。

【符号の説明】

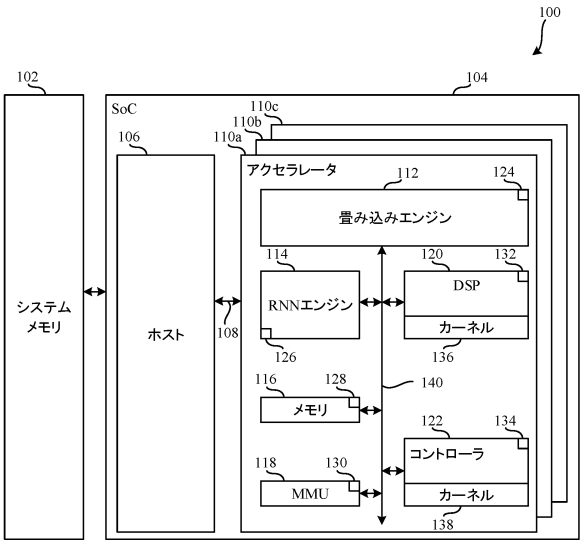
【０１６４】

１００，２００	コンピュータシステム
１０２	システムメモリ
１０４	ヘテロジニアスシステム
１０６	ホストプロセッサ
１０８，１４０，３０８，４１２	通信バス
１１０，２０８，８１２	アクセラレータ
１１２，２１４	畳み込みエンジン
１１４，２１８	ＲＮＮエンジン
１１６，２２２	メモリ
１１８，２１６	メモリ管理ユニット（ＭＭＵ）
１２０，２２０	ＤＳＰ
１２２	コントローラ
１２４～１３４，２３８～２４４	スケジューラ
１３６，１３８	カーネルライブラリ
２０２	グラフコンパイラ
２０４，３００	セレクタ
２０６	作業負荷
２１０，５１０	クレジットマネージャ
２１２	制御及び設定（ＣｎＣ）ファブリック
２２４	コンフィグレーションコントローラ

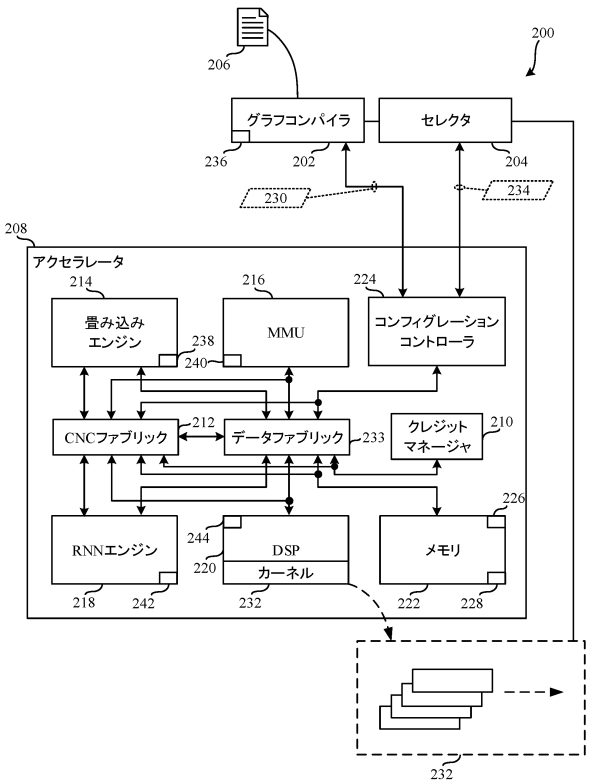
２２６	ＤＭＡユニット
２２８，５１２	バッファ
２３０	実行ファイル
２３２	カーネルバンク
２３３	データファブリック
２３４	設定及び制御メッセージ
２３６	プラグイン
３０２	ＣＢＢアナライザ
３０４	カーネルアナライザ
３０６	コンパイラインターフェイス
４０２	グラフインターフェイス
４０４	セレクタインターフェイス
４０６	作業負荷アナライザ
４０８	実行ファイル生成部
４１０	データストア
５００	パイプライン
５０２，５０５	ＣＢＢ
５０６，５０８	作業負荷ノード
８００	プロセッサプラットフォーム
８１０	プロセッサ

【図面】

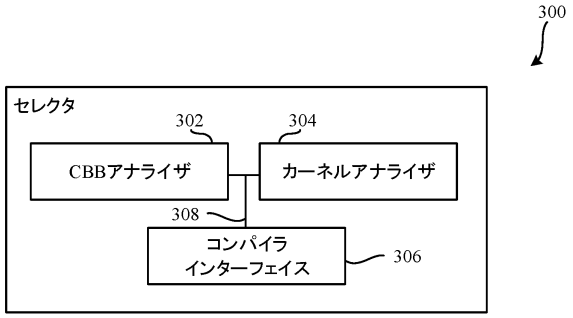
【図１】



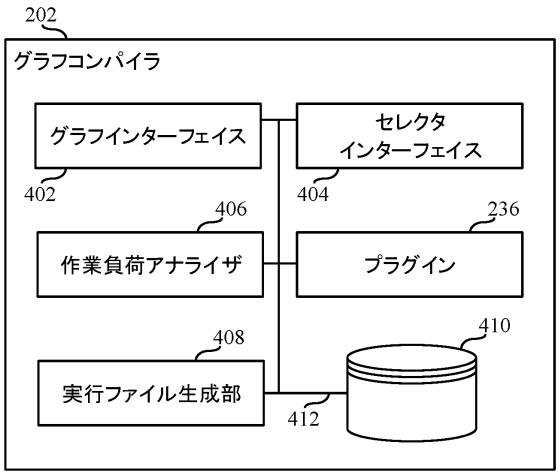
【図２】



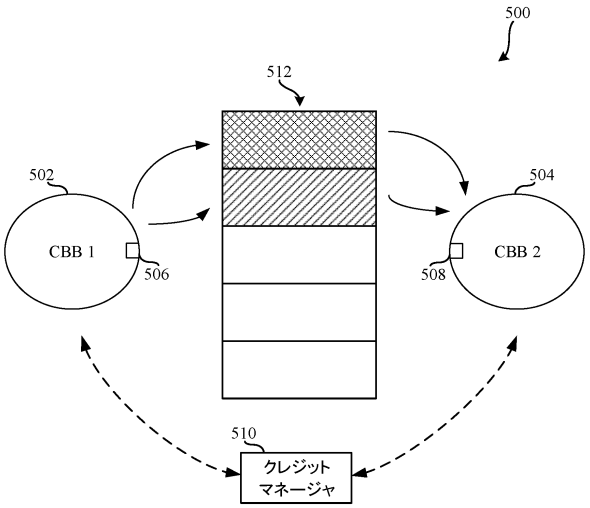
【図３】


300
セレクタ
302
304
CBBアナライザ
カーネルアナライザ
308
コンパイラ
インターフェイス
306

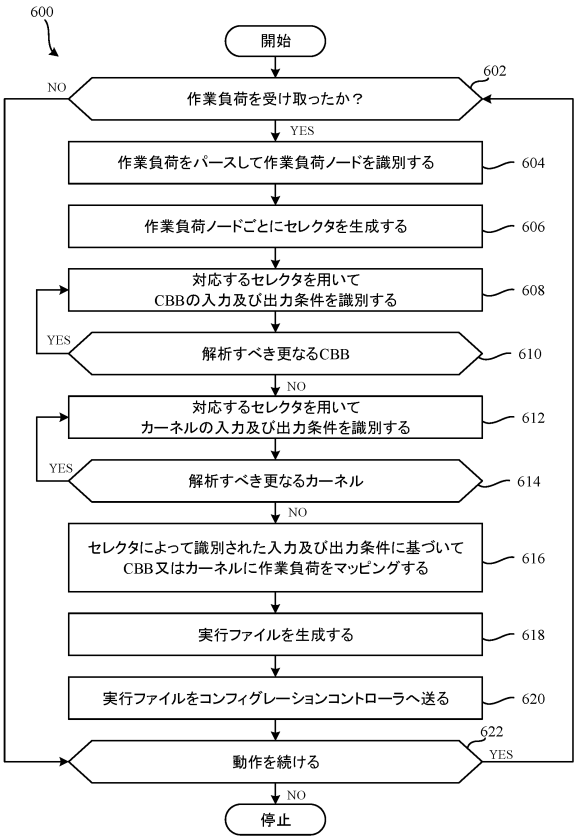
【図４】


202
グラフコンパイラ
グラフインターフェイス
セレクタ
インターフェイス
402
404
406
236
作業負荷アナライザ
プラグイン
410
408
実行ファイル生成部
412

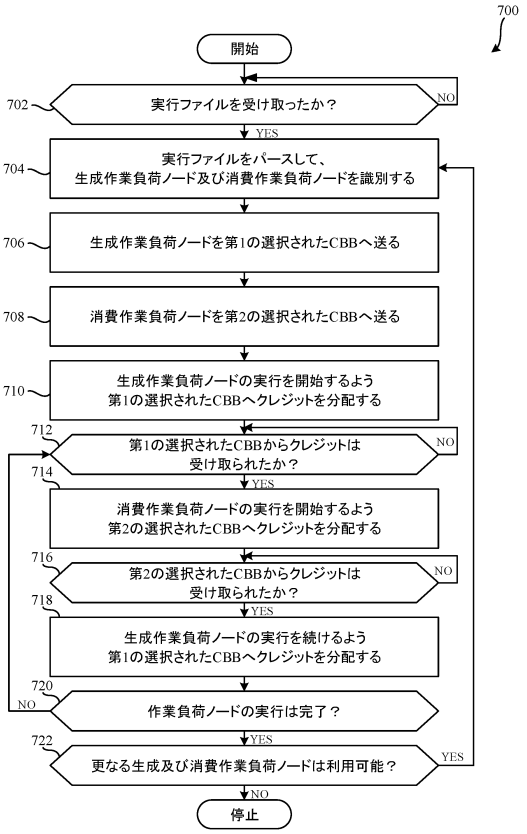
【図５】


500
512
502
504
CBB 1
506
CBB 2
508
510
クレジット
マネージャ

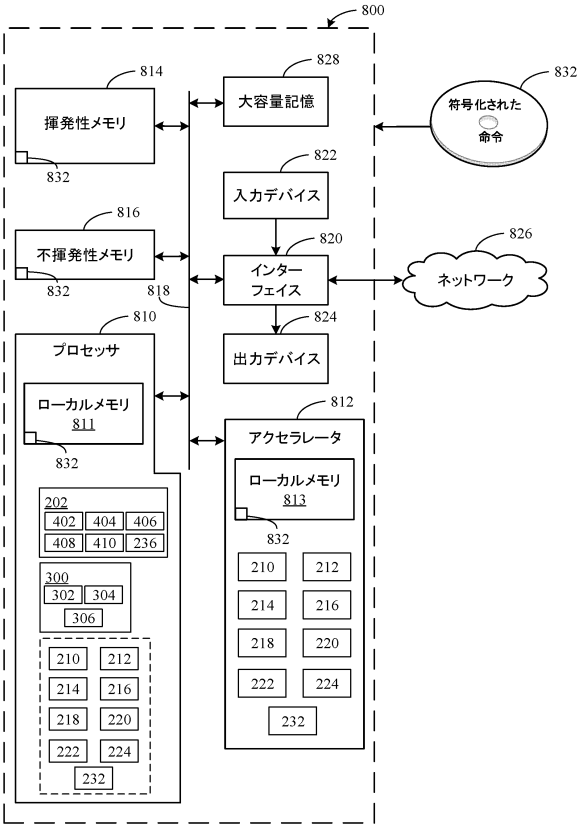
【図６】


600
開始
602
NO
作業負荷を受け取ったか？
YES
作業負荷をパースして作業負荷ノードを識別する
604
作業負荷ノードごとにセレクタを生成する
606
対応するセレクタを用いて
CBBの入力及び出力条件を識別する
608
YES
解析すべき更なるCBB
610
NO
対応するセレクタを用いて
カーネルの入力及び出力条件を識別する
612
YES
解析すべき更なるカーネル
614
NO
セレクタによって識別された入力及び出力条件に基づいて
CBB又はカーネルに作業負荷をマッピングする
616
実行ファイルを生成する
618
実行ファイルをコンフィグレーションコントローラへ送る
620
622
動作を続ける
YES
NO
停止

【図７】


700
開始
実行ファイルを受け取ったか？
NO
YES
702
実行ファイルをパースして、
生成作業負荷ノード及び消費作業負荷ノードを識別する
704
生成作業負荷ノードを第1の選択されたCBBへ送る
706
消費作業負荷ノードを第2の選択されたCBBへ送る
708
生成作業負荷ノードの実行を開始するよう
第1の選択されたCBBへクレジットを分配する
710
第1の選択されたCBBからクレジットは
受け取られたか？
712
714
消費作業負荷ノードの実行を開始するよう
第2の選択されたCBBへクレジットを分配する
第2の選択されたCBBからクレジットは
受け取られたか？
716
718
生成作業負荷ノードの実行を続けるよう
第1の選択されたCBBへクレジットを分配する
作業負荷ノードの実行は完了？
720
更なる生成及び消費作業負荷ノードは利用可能？
722
停止

【図８】


800
814
揮発性メモリ
832
828
大容量記憶
832
符号化された
命令
822
入力デバイス
816
不揮発性メモリ
832
820
インター
フェイス
826
ネットワーク
818
810
プロセッサ
824
出力デバイス
ローカルメモリ
811
832
812
アクセラレータ
ローカルメモリ
813
832
202
402
404
406
408
410
236
300
302
304
306
210
212
214
216
218
220
222
224
232
210
212
214
216
218
220
222
224
232

フロントページの続き

コーポレイション 内
(72)発明者 モシェ マオール
アメリカ合衆国 ９５０５４ カリフォルニア州 サンタ クララ ミッション カレッジ ブールバード ２２００ インテル コーポレイション 内
(72)発明者 ローネン ガバイ
アメリカ合衆国 ９５０５４ カリフォルニア州 サンタ クララ ミッション カレッジ ブールバード ２２００ インテル コーポレイション 内
(72)発明者 ロニ ロスナー
アメリカ合衆国 ９５０５４ カリフォルニア州 サンタ クララ ミッション カレッジ ブールバード ２２００ インテル コーポレイション 内
(72)発明者 ジギ ワルター
アメリカ合衆国 ９５０５４ カリフォルニア州 サンタ クララ ミッション カレッジ ブールバード ２２００ インテル コーポレイション 内
(72)発明者 オーレン アガム
アメリカ合衆国 ９５０５４ カリフォルニア州 サンタ クララ ミッション カレッジ ブールバード ２２００ インテル コーポレイション 内

審査官 武田 広太郎
(56)参考文献 米国特許出願公開第２０１０／０２１１７６０（ＵＳ，Ａ１）
特開平１１－０３１１３４（ＪＰ，Ａ）
特開２００７－３２８４１６（ＪＰ，Ａ）
特開２０１１－１９７８０３（ＪＰ，Ａ）
(58)調査した分野 (Int.Cl.，ＤＢ名)
Ｇ０６Ｆ ８／４１