

(11) 特許出願公開番号

特開2008-176453

(P2008-176453A)

(43) 公開日 平成20年7月31日(2008.7.31)

(51) Int.Cl.

G06F 9/455 (2006.01)

F 1

G O 6 F 9/44 3 1 0 D

テーマコード (参考)

審査請求 未請求 請求項の数 19 O L (全 21 頁)

(21) 出願番号 特願2007-7888 (P2007-7888)
(22) 出願日 平成19年1月17日 (2007.1.17)

(71) 出願人 302062931
NECエレクトロニクス株式会社
神奈川県川崎市中原区下沼部1753番地

(74) 代理人 100102864
弁理士 工藤 実

(72) 発明者 吉永 恵美
神奈川県川崎市中原区下沼部1753番地
NECエレクトロニクス株式会社内

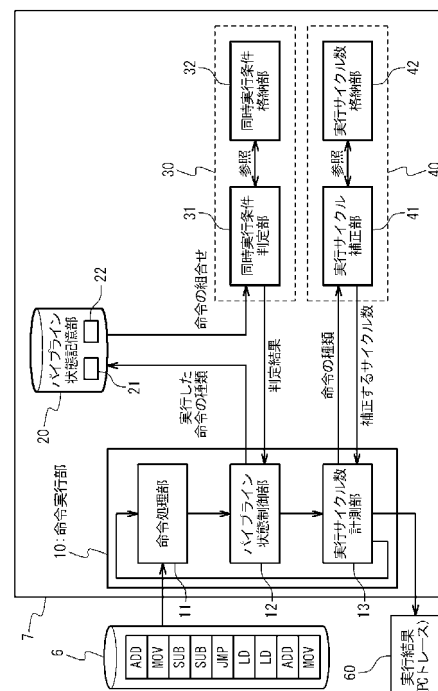
(54) 【発明の名称】 シミュレーション装置

(57) 【要約】

【課題】複数のパイプラインを考慮したプロセッサの動作をシミュレートすることができるシミュレーション装置を提供する。

【解決手段】シミュレーション装置は、命令処理部（１１）と、同時実行条件判定部（３１）と、実行マシンスイクル補正部（４１）と、を具備している。命令処理部（１１）は、プロセッサ（７０）上で実行可能な命令群で構成された解析対象プログラムを読み込んで各命令処理を行う。同時実行条件判定部（３１）は、解析対象プログラムが有する命令群のそれぞれが複数のパイプライン（７１、７２）で同時に処理可能かどうかを判断する。実行マシンスイクル補正部（４１）は、同時実行条件判定部（３１）の結果と、実行マシンスイクル数格納部（４２）に予め記憶された複数の命令の各々の実行マシンスイクル数とに基づいて、命令群の実行マシンスイクル数を算出する。

【選択図】図4



【特許請求の範囲】**【請求項 1】**

複数のパイプライン構造を有するプロセッサの動作を模擬するシミュレーション装置であって、

前記プロセッサ上で実行可能な命令群で構成された解析対象プログラムを読み込んで各命令処理を行う命令処理部と、

前記解析対象プログラムが有する命令群のそれぞれが前記複数のパイプラインで同時に処理可能かどうかを判断する同時実行条件判定部と、

前記同時実行条件判定部の結果と、実行サイクル数格納部に予め記憶された複数の命令の各々の実行マシンサイクル数とに基づいて、前記命令群の実行マシンサイクル数を算出する実行マシンサイクル補正部と、
を具備するシミュレーション装置。

10

【請求項 2】

前記同時実行条件判定部は、同時実行条件格納部に予め記憶された同時実行条件に基づいて、前記解析対象プログラム中の連続する命令群が前記複数のパイプラインで同時に処理可能かどうかを判断し、

前記同時実行条件は、前記複数のパイプラインのそれぞれが処理可能な命令に基づいて定義されている、

請求項 1 に記載のシミュレーション装置。

【請求項 3】

前記実行マシンサイクル補正部は、前記命令群のうちの、前記同時実行条件判定部によって同時実行可能と判断された実行命令群の実行マシンサイクル数を、前記実行命令群のそれぞれの実行マシンサイクル数の中で最も大きい実行マシンサイクル数とする、

請求項 2 に記載のシミュレーション装置。

20

【請求項 4】

前記複数のパイプラインのそれぞれに対応するパイプライン状態記憶部群を更に具備し、

前記同時実行条件判定部は、前記同時実行条件格納部を参照して、1 命令ずつ処理される 1 番目から最終番目までの前記命令群を 1 番目から最終番目までの複数の前記実行命令群に分割し、前記実行命令群をそれぞれ前記パイプライン状態記憶部群に格納し、

前記実行マシンサイクル補正部は、前記パイプライン状態記憶部群と実行サイクル数格納部とを参照して、前記実行命令群の実行マシンサイクル数の中から、前記最大実行マシンサイクル数を検索し、前記実行命令群の実行マシンサイクル数を前記最も大きい実行マシンサイクル数である最大実行マシンサイクル数に変更する、

請求項 3 に記載のシミュレーション装置。

30

【請求項 5】

前記パイプライン状態記憶部群に格納された実行命令の実行マシンサイクル数の合計値を含む実行結果を出力装置に出力する実行サイクル数計測部

を更に具備する請求項 4 に記載のシミュレーション装置。

【請求項 6】

前記同時実行条件判定部は、前記パイプライン状態記憶部群に格納された前記実行命令群が、同一のレジスタを用いて演算を行う演算実行命令群を表す場合、前記演算実行命令群を 1 番目から最終番目までの複数のハザード実行命令群に分割し、

前記複数のハザード実行命令群の各々は、同一のレジスタを用いていない命令の組を表し、

前記同時実行条件判定部は、前記パイプライン状態記憶部群に格納された前記演算実行命令群に代えて、前記ハザード実行命令群を前記実行命令群として 1 番目から最終番目までこの順に前記パイプライン状態記憶部群のうちの 1 つのパイプライン状態記憶部に格納する、

請求項 4 又は 5 に記載のシミュレーション装置。

40

50

【請求項 7】

前記命令群のうちの、演算命令を識別する識別子と、前記プロセッサが前記演算命令を実行するときに用いるレジスタのレジスタ名とが格納された使用レジスタ情報記憶部を更に具備し、

前記同時実行条件判定部は、前記パイプライン状態記憶部群と前記使用レジスタ情報記憶部とを参照して、前記パイプライン状態記憶部群に格納された前記実行命令群が、同一のレジスタを用いて演算を行う演算実行命令群を表す場合、前記演算実行命令群を 1 番目から最終番目までの複数のハザード実行命令群に分割し、

前記複数のハザード実行命令群の各々は、同一のレジスタを用いていない命令の組を表し、

前記同時実行条件判定部は、前記パイプライン状態記憶部群に格納された前記演算実行命令群に代えて、前記ハザード実行命令群を前記実行命令群として 1 番目から最終番目までこの順に前記パイプライン状態記憶部群のうちの 1 つのパイプライン状態記憶部に格納する、

請求項 4 又は 5 に記載のシミュレーション装置。

【請求項 8】

複数のパイプライン構造を有するプロセッサの動作をコンピュータを用いて模擬するシミュレーション方法であって、

前記プロセッサ上で実行可能な命令群で構成された解析対象プログラムを読み込んで各命令処理を行う命令処理ステップと、

前記解析対象プログラムが有する命令群のそれぞれが前記複数のパイプラインで同時に処理可能かどうかを判断する同時実行条件判定ステップと、

前記同時実行条件判定部の結果と、実行サイクル数格納部に予め記憶された複数の命令の各々の実行マシンサイクル数とに基づいて、前記命令群の実行マシンサイクル数を算出する実行マシンサイクル補正ステップと、
を具備するシミュレーション方法。

【請求項 9】

前記同時実行条件判定ステップは、同時実行条件格納ステップに予め記憶された同時実行条件に基づいて、前記解析対象プログラム中の連続する命令群が前記複数のパイプラインで同時に処理可能かどうかを判断し、

前記同時実行条件は、前記複数のパイプラインのそれぞれが処理可能な命令に基づいて定義されている、

請求項 8 に記載のシミュレーション方法。

【請求項 10】

前記実行マシンサイクル補正ステップは、前記命令群のうちの、前記同時実行条件判定ステップによって同時実行可能と判断された実行命令群の実行マシンサイクル数を、前記実行命令群のそれぞれの実行マシンサイクル数の中で最も大きい実行マシンサイクル数とする、

請求項 9 に記載のシミュレーション方法。

【請求項 11】

前記同時実行条件判定ステップは、前記同時実行条件格納部を参照して、1 命令ずつ処理される 1 番目から最終番目までの前記命令群を 1 番目から最終番目までの複数の前記実行命令群に分割し、前記実行命令群をそれぞれ前記複数のパイプラインに対応する前記パイプライン状態記憶部群に格納し、

前記実行マシンサイクル補正ステップは、前記パイプライン状態記憶部群と実行サイクル数格納部とを参照して、前記実行命令群の実行マシンサイクル数の中から、前記最大実行マシンサイクル数を検索し、前記実行命令群の実行マシンサイクル数を前記最も大きい実行マシンサイクル数である最大実行マシンサイクル数に変更する、

請求項 10 に記載のシミュレーション方法。

【請求項 12】

前記パイプライン状態記憶部群に格納された実行命令の実行マシンサイクル数の合計値を含む実行結果を出力装置に出力する実行サイクル数計測ステップ
を更に具備する請求項 1 1 に記載のシミュレーション方法。

【請求項 1 3】

前記同時実行条件判定ステップは、前記パイプライン状態記憶部群に格納された前記実行命令群が、同一のレジスタを用いて演算を行う演算実行命令群を表す場合、前記演算実行命令群を 1 番目から最終番目までの複数のハザード実行命令群に分割し、

前記複数のハザード実行命令群の各々は、同一のレジスタを用いていない命令の組を表し、

前記同時実行条件判定ステップは、前記パイプライン状態記憶部群に格納された前記演算実行命令群に代えて、前記ハザード実行命令群を前記実行命令群として 1 番目から最終番目までこの順に前記パイプライン状態記憶部群のうちの 1 つのパイプライン状態記憶部に格納する、

請求項 1 1 又は 1 2 に記載のシミュレーション方法。

【請求項 1 4】

コンピュータにインストールされ、複数のパイプライン構造を有するプロセッサの動作を前記コンピュータを用いて模擬するためのコンピュータプログラムであるシミュレーションプログラムであって、

前記プロセッサ上で実行可能な命令群で構成された解析対象プログラムを読み込んで各命令処理を行う命令処理ステップと、

前記解析対象プログラムが有する命令群のそれぞれが前記複数のパイプラインで同時に処理可能かどうかを判断する同時実行条件判定ステップと、

前記同時実行条件判定部の結果と、実行サイクル数格納部に予め記憶された複数の命令の各々の実行マシンサイクル数とに基づいて、前記命令群の実行マシンサイクル数を算出する実行マシンサイクル補正ステップと、

の各ステップを前記コンピュータに実行させるシミュレーションプログラム。

【請求項 1 5】

前記同時実行条件判定ステップは、同時実行条件格納ステップに予め記憶された同時実行条件に基づいて、前記解析対象プログラム中の連続する命令群が前記複数のパイプラインで同時に処理可能かどうかを判断し、

前記同時実行条件は、前記複数のパイプラインのそれぞれが処理可能な命令に基づいて定義されている、

請求項 1 4 に記載のシミュレーションプログラム。

【請求項 1 6】

前記実行マシンサイクル補正ステップは、前記命令群のうちの、前記同時実行条件判定ステップによって同時実行可能と判断された実行命令群の実行マシンサイクル数を、前記実行命令群のそれぞれの実行マシンサイクル数の中で最も大きい実行マシンサイクル数とする、

請求項 1 5 に記載のシミュレーションプログラム。

【請求項 1 7】

前記同時実行条件判定ステップは、前記同時実行条件格納部を参照して、1 命令ずつ処理される 1 番目から最終番目までの前記命令群を 1 番目から最終番目までの複数の前記実行命令群に分割し、前記実行命令群をそれぞれ前記複数のパイプラインに対応する前記パイプライン状態記憶部群に格納し、

前記実行マシンサイクル補正ステップは、前記パイプライン状態記憶部群と実行サイクル数格納部とを参照して、前記実行命令群の実行マシンサイクル数の中から、前記最大実行マシンサイクル数を検索し、前記実行命令群の実行マシンサイクル数を前記最も大きい実行マシンサイクル数である最大実行マシンサイクル数に変更する、

請求項 1 6 に記載のシミュレーションプログラム。

【請求項 1 8】

10

20

30

40

50

前記パイプライン状態記憶部群に格納された実行命令の実行マシンサイクル数の合計値を含む実行結果を出力装置に出力する実行サイクル数計測ステップ
を更に前記コンピュータに実行させる請求項 17 に記載のシミュレーションプログラム。

【請求項 19】

前記同時実行条件判定ステップは、前記パイプライン状態記憶部群に格納された前記実行命令群が、同一のレジスタを用いて演算を行う演算実行命令群を表す場合、前記演算実行命令群を 1 番目から最終番目までの複数のハザード実行命令群に分割し、

前記複数のハザード実行命令群の各々は、同一のレジスタを用いていない命令の組を表し、

前記同時実行条件判定ステップは、前記パイプライン状態記憶部群に格納された前記演算実行命令群に代えて、前記ハザード実行命令群を前記実行命令群として 1 番目から最終番目までこの順に前記パイプライン状態記憶部群のうちの 1 つのパイプライン状態記憶部に格納する、

請求項 17 又は 18 に記載のシミュレーションプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、プロセッサの動作をシミュレートするシミュレーション装置、シミュレーション方法及びシミュレーションプログラムに関する。

【背景技術】

【0002】

プロセッサ上で動作させるプログラムの解析やデバックのためにシミュレーション装置が利用されている。シミュレーション装置は、プロセッサの動作と同じ動作か、あるいは、それを簡略化した動作を行うように設計される。このようなシミュレーション装置としては、命令セットシミュレータ (ISS; Instruction Set Simulator) といったものが広く知られている。ISS は、パーソナルコンピュータやワークステーションなどのコンピュータ上で動作し、プロセッサの命令の動作や、その実行結果、レジスタの状態などを模擬するものである。ISS を用いることにより、プログラムの解析やデバックのために専用のハードウェアを用意する必要がないというメリットがある。

【0003】

ISS などを用いたプログラムの解析においては、プログラムの動作の正当性を表す情報だけでなく、プログラムの実行速度を表す情報が得られると有益である。より具体的には、後者の情報として、プログラム中にある命令の処理にプロセッサが何サイクル要するか、ということである。例えば、特開 2001 - 290857 号公報には、パイプライン構造を有するハードウェアと同等の動作を行うシミュレーション手法に関する技術が記載されており、1 命令毎の実行におけるパイプラインの状態を命令毎のタイミングテーブルを参照することで、パイプラインを考慮した実行マシンサイクル数を算出している。

【0004】

【特許文献 1】特開 2001 - 290857 号公報

【発明の開示】

【発明が解決しようとする課題】

【0005】

近年のプロセッサでは、命令の処理速度を上げるために、1 つのプロセッサ内に複数のパイプラインを有するようになってきた。従って、このようなプロセッサで動作させるプログラムを開発する場合においても、複数のパイプラインを考慮したシミュレーション手法が望まれる。

【0006】

しかしながら、特開 2001 - 290857 号公報には、複数のパイプラインを有するプロセッサをシミュレートする技術に関しては何ら記載されていない。複数のパイプライン

10

20

30

40

50

ンを有するプロセッサに対してこの従来技術を適用した場合、個々のパイプラインで実行されるマシンサイクル数を算出することは出来たとしても、複数のパイプラインを組み合わせて動作するプロセッサ全体のマシンサイクル数を算出することは困難である。

【課題を解決するための手段】

【 0 0 0 7 】

以下に、[発明を実施するための最良の形態] で使用する番号・符号を用いて、課題を解決するための手段を説明する。これらの番号・符号は、[特許請求の範囲] の記載と[発明を実施するための最良の形態] の記載との対応関係を明らかにするために付加されたものであるが、[特許請求の範囲] に記載されている発明の技術的範囲の解釈に用いてはならない。

10

【 0 0 0 8 】

本発明のシミュレーション装置 (4) は、

複数のパイプライン構造 (7 1 、 7 2) を有するプロセッサ (7 0) の動作を模擬するシミュレーション装置 (4) であって、

前記プロセッサ (7 0) 上で実行可能な命令群 (“ A D D 、 M O V 、 S U B 、 S U B 、 J M P 、 L D 、 L D 、 A D D 、 M O V ”) で構成された解析対象プログラムを読み込んで各命令処理を行う命令処理部 (1 1) と、

前記解析対象プログラムが有する命令群 (“ A D D 、 M O V 、 S U B 、 S U B 、 J M P 、 L D 、 L D 、 A D D 、 M O V ”) のそれぞれが前記複数のパイプライン (7 1 、 7 2) で同時に処理可能かどうかを判断する同時実行条件判定部 (3 1) と、

20

前記同時実行条件判定部 (3 1) の結果と、実行マシンサイクル数格納部 (4 2) に予め記憶された複数の命令 (L D 、 A D D 、 S U B 、 J M P 、 M O V 、 …) の各々の実行サイクル数 (1 、 2 、 2 、 6 、 4 、 …) とに基づいて、前記命令群 (“ A D D 、 M O V 、 S U B 、 S U B 、 J M P 、 L D 、 L D 、 A D D 、 M O V ”) の実行マシンサイクル数 (“ 4 ”) (“ 2 ”) (“ 6 ”) (“ 1 ”) (“ 2 ”) (“ 4 ”) を算出する実行マシンサイクル補正部 (4 1) と、
を具備している。

【発明の効果】

【 0 0 0 9 】

以上により、本発明のシミュレーション装置 (4) によれば、複数のパイプラインを考慮したプロセッサ (7 0) の動作をシミュレートすることができる。

30

【 0 0 1 0 】

その理由として、本発明のシミュレーション装置 (4) では、命令群 (“ A D D 、 M O V 、 S U B 、 S U B 、 J M P 、 L D 、 L D 、 A D D 、 M O V ”) に対して、プロセッサ (7 0) がパイプライン群 (7 1 、 7 2) により同時実行可能な命令の組として実行命令群 (“ A D D 、 M O V ” 、 “ S U B 、 S U B ” 、 “ J M P 、 _ ” 、 “ L D 、 _ ” 、 “ L D 、 A D D ” 、 “ M O V 、 _ ”) を検索し、命令群 (“ A D D 、 M O V 、 S U B 、 S U B 、 J M P 、 L D 、 L D 、 A D D 、 M O V ”) が処理される順番を考慮して、実行命令群 (“ A D D 、 M O V ” 、 “ S U B 、 S U B ” 、 “ J M P 、 _ ” 、 “ L D 、 _ ” 、 “ L D 、 A D D ” 、 “ M O V 、 _ ”) の実行マシンサイクル数 (“ 2 、 4 ” 、 “ 2 、 2 ” 、 “ 6 、 _ ” 、 “ 1 、 _ ” 、 “ 1 、 2 ” 、 “ 4 、 _ ”) を最大実行マシンサイクル数 (“ 4 ” 、 “ 2 ” 、 “ 6 ” 、 “ 1 ” 、 “ 2 ” 、 “ 4 ”) に変更する。このように、本発明のシミュレーション装置 (4) では、プロセッサ (7 0) がパイプライン群 (7 1 、 7 2) により同時実行可能な実行命令群を検索し、その実行マシンサイクル数を最大実行マシンサイクル数に変更することにより、複数のパイプライン { パイプライン群 (7 1 、 7 2) } を考慮したプロセッサ (7 0) の動作をシミュレートすることができる。

40

【発明を実施するための最良の形態】

【 0 0 1 1 】

以下に添付図面を参照して、本発明のシミュレーション装置について詳細に説明する。

【 0 0 1 2 】

50

(第1実施形態)

図1は、本発明の第1実施形態によるシミュレーション装置4の構成を示している。そのシミュレーション装置4は、ユーザインターフェース部5と、メモリである命令データ格納部6と、シミュレーションエンジン部(シミュレーションプログラム)7とを具備している。ユーザインターフェース部5は、入力装置による入力、及び、出力装置に出力する制御を行う。命令データ格納部6には、デバック対象のプログラムが格納されている。シミュレーションエンジン部7は、デバック対象のプログラムを命令データ格納部6から読み出し、デバック対象のプログラムに記述されている命令群を処理する。

【0013】

シミュレーションエンジン部7は、コンピュータにインストールされるコンピュータプログラム(シミュレーションプログラム)としても実現可能である。図2は、コンピュータの構成を示している。コンピュータは、入力装置2と、出力装置3と、コンピュータ本体1とを具備している。入力装置2と出力装置3は、コンピュータ本体1に接続されている。入力装置2は、キーボード、ポインティングデバイスを含み、出力装置3は、表示装置、プリンタを含んでいる。

【0014】

上記のコンピュータ本体1は、図示しない記憶装置と、図示しないCPU(Central Processing Unit)とを具備している。シミュレーションエンジン部7は、記憶装置にインストールされる。この場合、命令データ格納部6に格納されたデバック対象のプログラムも記憶装置にインストールされる。シミュレーションエンジン部7(シミュレーションプログラム)は、デバック対象のプログラムを記憶装置から読み出し、デバック対象のプログラムに記述されている命令群をCPU上で処理する。

【0015】

シミュレーションエンジン部7は、プロセッサ上で動作させるプログラムの解析やデバックのためのソフトウェアとして、設計者に利用される。図3は、本発明でシミュレートを実行しようとしているプロセッサ70のパイプライン部分の概略図である。プロセッサ70は、命令群とデータとが格納されたメモリ73と、複数の汎用レジスタを含むレジスタ74と、2つのパイプライン71、72とを具備している。

【0016】

プロセッサ70は、クロックに応じて動作し、更に、命令群をフェッチするIF(命令フェッチ)ステージ81と、フェッチされた命令をデコードするDC(デコード)ステージ82-1、82-2と、デコードされた命令を実行するEX(実行)ステージ83-1、83-2と、命令(メモリアクセス命令)に対してメモリ73のアクセスを行うME(メモリアクセス)ステージ84-1、84-2と、命令の実行結果をレジスタ74に書き込むWB(ライトバック)ステージ85-1、85-2とを具備している。プロセッサ70は、IFステージ81とDCステージ82-1とEXステージ83-1とMEステージ84-1とWBステージ85-1によりパイプライン71を構成し、IFステージ81とDCステージ82-2とEXステージ83-2とMEステージ84-2とWBステージ85-2によりパイプライン72を構成する。

【0017】

プロセッサ70により実行される命令群nは、例えば、“ADD、MOV、SUB、JMP、LD”といったものがあるとする。ニーモニック表記の“LD”、“ADD”、“SUB”、“JMP”、“MOV”は、それぞれ、メモリ73からレジスタ74にデータを書き込むロード命令、加算命令、減算命令、ジャンプ命令、ムーブ命令を表している。プロセッサ70上で動作するプログラムはこれらの命令で構成されることになる。

【0018】

図3に示されるプロセッサ70の場合、2つのパイプライン71、72を具備しているため、フェッチした各命令は2つのパイプライン71、72を使って並列に処理することにより1つのパイプライン71の場合よりも処理速度を上げることができる。ただし、2つのパイプライン71、72により処理される加算器、減算器といった演算器は同じ構成

10

20

30

40

50

にはならない場合がある。これは各演算器をパイプラインの数だけ用意してしまうと、処理速度を上げることはできるが、プロセッサ70が巨大になってしまうというデメリットが生じるためである。大型コンピュータに搭載されるプロセッサであれば問題にはないが、小型コンピュータ用や、家電・車両用のマイコンに搭載されるプロセッサでは、重要な要素となる。具体的には、例えば、命令“LD”、“JMP”、“MOV”を処理するための演算器が一方のパイプライン71にしか配置されていない、というプロセッサ70が想定される。この場合、例えば、プロセッサ70上で動作させようとするプログラム中に、命令“JMP、MOV”という連続する命令があった場合、プロセッサ70は、パイプラインの制約上、これらの命令を2つのパイプライン71、72で並列に実行することはできない。このため、プロセッサ70は、命令“JMP”をパイプライン71で実行した後に、命令“MOV”をパイプライン71で実行するという逐次処理となる。本実施例では、このようなプロセッサ70をシミュレートする場合の手法について述べる。なお、上記の命令“LD”、“JMP”、“MOV”を処理するための演算器が一方のパイプライン71にしか配置されていないプロセッサ70、というものは、単に説明を簡略化するために挙げたものに過ぎない。

10

20

30

40

50

【0019】

図4は、シミュレーションエンジン部7の構成を示している。シミュレーションエンジン部7は、命令実行部10と、パイプライン状態記憶装置20と、同時実行命令検索部30と、実行サイクル変更部40とを具備している。命令実行部10は、命令処理部11と、パイプライン状態制御部12と、実行サイクル数計測部13とを具備している。パイプライン状態記憶装置20は、パイプライン状態記憶部群21、22を具備している。パイプライン状態記憶部群21、22は、それぞれ、上記のパイプライン群71、72に対応している。同時実行命令検索部30は、同時実行条件判定部31と、同時実行条件格納部32とを具備している。実行サイクル変更部40は、実行サイクル補正部41と、実行サイクル数格納部42とを具備している。命令実行部10において、命令処理部11は、前述のISSが有している機能と同等のものであるため、その説明を省略する。パイプライン状態制御部12と実行サイクル数計測部13は、本発明の特徴部分であり、以下はこれらを中心に説明する。

【0020】

図5は、同時実行条件格納部32を示している。同時実行条件格納部32には、プロセッサ70がパイプライン群71、72により同時実行可能な命令の組を表す同時実行条件が予め格納されている。同時実行条件は、パイプラインの制約として予め決められている。例えば、命令の組として“ADD、MOV”、“LD、MOV”、...と、それに対して同時実行を許可することを表す判定“OK”とが同時実行条件格納部32に格納されている。また、命令の組として“JMP、MOV”、“LD、LD”、...と、それに対して同時実行を許可しないことを表す判定“NG”とが同時実行条件格納部32に格納されている。

【0021】

図6は、実行サイクル数格納部42を示している。実行サイクル数格納部42には、複数の命令と、複数の命令の各々に対してEXステージが実行されときのマシンサイクル(クロック)の数を表す実行マシンサイクル数とが予め格納されている。例えば、複数の命令として“LD、ADD、SUB、JMP、MOV、...”と、それに対する実行マシンサイクル数として“1、2、2、6、4、...”とが実行サイクル数格納部42に格納されている。ここで、実行マシンサイクル数については、本実施形態を説明する便宜上、実際よりも極端に大きい値を記載している。これらの値は、実際のプロセッサ70の処理速度によって決定される。その決定方法としては、プロセッサ70を用いて予め測定して決定する場合や、プロセッサ70の設計仕様に基づいて予め決定する場合などが挙げられる。

【0022】

命令処理部11は、設計者の入力装置2の操作により、記憶装置(命令データ格納部6に対応)からデバック対象のプログラムを読み出す。図4では、説明の簡略化のためにデ

バック対象のプログラム中の命令群だけを記載しており、デバック対象のプログラムが、“ADD、MOV、SUB、SUB、JMP、LD、LD、ADD、MOV”の順番で命令群を処理するようなプログラムであるものとする。命令処理部11は、記憶装置(命令データ格納部6に対応)から命令群“ADD、MOV、SUB、SUB、JMP、LD、LD、ADD、MOV”を順次読み出して、パイプライン状態制御部12に出力するとともに各命令に応じた処理を実行する。パイプライン状態制御部12は、同時実行可能な命令の組を検索する同時実行命令検索処理を実行させるために、その命令群を同時実行条件判定部31に出力する。

【0023】

同時実行命令検索処理において、図7Aに示されるように、同時実行条件判定部31は、同時実行条件格納部32を参照して、1命令ずつ処理される1番目から最終番目までの命令群“ADD、MOV、SUB、SUB、JMP、LD、LD、ADD、MOV”を1番目から最終番目までの複数の実行命令群“ADD、MOV”、“SUB、SUB”、“JMP、__”、“LD、__”、“LD、ADD”、“MOV、__”に分割する。“__”は、同時実行条件により実行されない命令を表している。複数の実行命令群“ADD、MOV”、“SUB、SUB”、“JMP、__”、“LD、__”、“LD、ADD”、“MOV、__”の各々は、プロセッサ70がパイプライン群71、72により同時実行可能な命令の組を表している。同時実行条件判定部31は、1番目から最終番目までこの順に実行命令群“ADD、MOV”、“SUB、SUB”、“JMP、__”、“LD、__”、“LD、ADD”、“MOV、__”をパイプライン状態記憶装置20に格納する。このとき、同時実行条件判定部31は、実行命令群“ADD、MOV”、“SUB、SUB”、“JMP、__”、“LD、__”、“LD、ADD”、“MOV、__”をそれぞれパイプライン状態記憶部群21、22に格納し、同時実行命令検索処理の終了をパイプライン状態制御部12に通知する。

【0024】

実行命令群“ADD、MOV”、“SUB、SUB”、“JMP、__”、“LD、__”、“LD、ADD”、“MOV、__”の実行マシンサイクル数は、それぞれ、“2、4”、“2、2”、“6、__”、“1、__”、“1、2”、“4、__”である。例えば、実行命令群“ADD、MOV”、“SUB、SUB”、“JMP、__”、“LD、__”、“LD、ADD”、“MOV、__”に対して、それぞれプロセッサ70のパイプライン群71、72によってIFステージ81、DCステージ82-1、82-2、EXステージ83-1、83-2、MEステージ84-1、84-2、WBステージ85-1、85-2が実行される。このとき、プロセッサ70が1番目の実行命令“ADD”と2番目の実行命令“MOV”とを同時実行した場合、実行命令“ADD”、“MOV”に対してEXステージ83-1、83-2が実行されるときの実行マシンサイクル数は“2”、“4”である。図7Bに示されるように、1番目の実行命令“ADD”と2番目の実行命令“MOV”が実行されるときを時間t0とした場合、2番目の実行命令“MOV”が実行されているときに、時間t6において1番目の実行命令“ADD”の実行が先に終了し、時間t8において2番目の実行命令“MOV”と3番目の実行命令“SUB”の実行が同時に終了する。即ち、1命令ずつ処理される1番目から最終番目までの命令群“ADD、MOV、SUB、SUB、JMP、LD、LD、ADD、MOV”に対して、3番目の実行命令“SUB”が2番目の実行命令“MOV”よりも先に実行されてしまう可能性がある。そこで、実行マシンサイクル数を補正する必要がある。

【0025】

パイプライン状態制御部12は、実行マシンサイクル数を補正する実行サイクル検索処理を実行させるために、パイプライン状態記憶部群21、22に格納された実行命令群“ADD、MOV”、“SUB、SUB”、“JMP、__”、“LD、__”、“LD、ADD”、“MOV、__”を読み出し、実行サイクル数計測部13に出力する。実行サイクル数計測部13は、その実行命令群を実行サイクル補正部41に出力する。

【0026】

10

20

30

40

50

実行サイクル検索処理において、実行サイクル補正部 4 1 は、実行サイクル数計測部 1 3 からの実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、_ ”、“ L D、_ ”、“ L D、A D D ”、“ M O V、_ ”を受け取る。又は、実行サイクル補正部 4 1 は、実行サイクル検索処理を実行させる通知を実行サイクル数計測部 1 3 から受け取ったとき、パイプライン状態記憶部群 2 1、2 2 を参照してもよい。実行サイクル補正部 4 1 は、実行マシンサイクル数格納部 4 2 を参照して、パイプライン群 7 1、7 2 によって実行される実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、_ ”、“ L D、_ ”、“ L D、A D D ”、“ M O V、_ ”の実行マシンサイクル数 “ 2、4 ”、“ 2、2 ”、“ 6、_ ”、“ 1、_ ”、“ 1、2 ”、“ 4、_ ”の中から、最も大きい実行マシンサイクル数である最大実行マシンサイクル数 “ 4 ”、“ 2 ”、“ 6 ”、“ 1 ”、“ 2 ” “ 4 ”を検索する。検索の結果、図 8 B に示されるように、実行マシンサイクル補正部 4 1 は、実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、_ ”、“ L D、_ ”、“ L D、A D D ”、“ M O V、_ ”の実行マシンサイクル数 “ 2、4 ”、“ 2、2 ”、“ 6、_ ”、“ 1、_ ”、“ 1、2 ”、“ 4、_ ”を最大実行マシンサイクル数 “ 4 ”、“ 2 ”、“ 6 ”、“ 1 ”、“ 2 ”、“ 4 ”に変更する。実行マシンサイクル補正部 4 1 は、その最大実行マシンサイクル数 “ 4 ”、“ 2 ”、“ 6 ”、“ 1 ”、“ 2 ”、“ 4 ”を命令実行部 1 0 に出力し、実行サイクル検索処理の終了を通知する。

10

【 0 0 2 7 】

このとき、図 8 A に示されるように、命令実行部 1 0 のパイプライン状態制御部 1 2 は、パイプライン状態記憶部群 2 1、2 2 に格納された実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、_ ”、“ L D、_ ”、“ L D、A D D ”、“ M O V、_ ”の実行マシンサイクル数 “ 2、4 ”、“ 2、2 ”、“ 6、_ ”、“ 1、_ ”、“ 1、2 ”、“ 4、_ ”を最大実行マシンサイクル数 “ 4 ”、“ 2 ”、“ 6 ”、“ 1 ”、“ 2 ”、“ 4 ”に更新し、命令実行部 1 0 の実行サイクル数計測部 1 3 に出力する。

20

【 0 0 2 8 】

実行サイクル数計測部 1 3 は、図 9 に示されるような実行結果 6 0 を出力装置 3 に出力する。実行結果 6 0 は、パイプライン状態記憶部群 2 1、2 2 に格納された実行命令 “ A D D、S U B、J M P、L D、L D、M O V ”、“ M O V、S U B、_、_、A D D、_ ”と、その実行マシンサイクル数 “ 4、2、6、1、2、4 ”の合計値 “ 1 9 ”を含んでいる。この場合、命令群 “ A D D、M O V、S U B、S U B、J M P、L D、L D、A D D、M O V ”の実行マシンサイクル数 “ 2、4、2、2、6、1、1、2、4 ”の合計値 “ 2 4 ”よりも小さい。また、1 クロックの周期を 1 0 [n s e c]とした場合、上記の合計値 “ 1 9 ”は、処理時間 [n s e c]に対応し、実行命令 “ A D D、S U B、J M P、L D、L D、M O V ”、“ M O V、S U B、_、_、A D D、_ ”に対する処理時間 [n s e c]は、“ 4 0、6 0、1 2 0、1 3 0、1 5 0、1 9 0 ”[n s e c]により表される。ここで、1 クロックの周期については、本実施形態を説明する便宜上、実際よりも極端に大きい値を記載している。

30

【 0 0 2 9 】

プロセッサ 7 0 では、命令の処理速度を上げるために、1 つのプロセッサ 7 0 内に複数のパイプラインとしてパイプライン群 7 1、7 2 を有している。従って、このようなプロセッサ 7 0 で動作させるプログラムを開発する場合においても、パイプライン群 7 1、7 2 を考慮したシミュレーション手法が望まれる。しかしながら、パイプライン群 7 1、7 2 がまったく同じ構成ではなく、それぞれのパイプライン 7 1、7 2 に実装されている演算器が異なる場合は、特開 2 0 0 1 - 2 9 0 8 5 7 号公報に記載されている技術を単純にパイプライン群 7 1、7 2 に適用することは困難である。

40

【 0 0 3 0 】

そこで、本発明の第 1 実施形態によるシミュレーション装置 4 では、命令群 “ A D D、M O V、S U B、S U B、J M P、L D、L D、A D D、M O V ”に対して、プロセッサ 7 0 がパイプライン群 7 1、7 2 により同時実行可能な命令の組として実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、_ ”、“ L D、_ ”、“ L D、A D D ”

50

、“MOV、_”を検索し、命令群“ADD、MOV、SUB、SUB、JMP、LD、LD、ADD、MOV”が処理される順番を考慮して、実行命令群“ADD、MOV”、“SUB、SUB”、“JMP、_”、“LD、_”、“LD、ADD”、“MOV、_”の実行マシンサイクル数“2、4”、“2、2”、“6、_”、“1、_”、“1、2”、“4、_”を最大実行マシンサイクル数“4”、“2”、“6”、“1”、“2”、“4”に変更する。このように、本発明の第1実施形態によるシミュレーション装置4では、プロセッサ70がパイプライン群71、72により同時実行可能な実行命令群を検索し、その実行マシンサイクル数を最大実行マシンサイクル数に変更することにより、複数のパイプライン（パイプライン群71、72）を考慮したプロセッサ70の動作をシミュレートすることができる。

10

【0031】

図10は、シミュレーションエンジン部7の動作を示すフローチャートである。

【0032】

まず、命令処理部11は、1番目の命令“ADD”を命令データ格納部6から読み出す。同時に、実行サイクル数計測部13は、実行サイクル数格納部42を参照して、1番目の命令“ADD”の実行マシンサイクル数“2”を取得するように、実行サイクル補正部41を制御する（ステップS1）。パイプライン状態制御部12は、同時実行命令検索処理として、1番目の命令“ADD”をパイプライン状態記憶部群21の1レコード目に格納するように、同時実行条件判定部31を制御する（ステップS2-NO）。このとき、実行サイクル数計測部13は、実行結果60として、1番目の命令“ADD”の実行マシンサイクル数“2”を、1番目の実行命令群の総実行マシンサイクル数“0”に加算する（ステップS8）。

20

【0033】

次に、命令処理部11は、2番目の命令“MOV”を命令データ格納部6から読み出す（ステップS9-NO）。同時に、実行サイクル数計測部13は、実行サイクル数格納部42を参照して、2番目の命令“MOV”の実行マシンサイクル数“4”を取得するように、実行サイクル補正部41を制御する（ステップS1）。ここで、パイプライン状態記憶部群21の1レコード目には、1番目の命令“ADD”が格納されている（ステップS2-YES）。パイプライン状態制御部12は、同時実行命令検索処理として、同時実行条件格納部32を参照するように、同時実行条件判定部31を制御する（ステップS3）。検索の結果、1番目の命令（先行命令群）“ADD”と2番目の命令“MOV”は、同時実行可能である（ステップS4-YES）。この場合、同時実行条件判定部31は、1番目の実行命令群“ADD、MOV”として、パイプライン状態記憶部群21に格納された1番目の命令“ADD”に対応付けて、2番目の命令“MOV”をパイプライン状態記憶部群21の1レコード目に格納する。1番目の命令（先行命令群）“ADD”の実行マシンサイクル数“2”は、2番目の命令（現命令）“MOV”の実行マシンサイクル数“4”よりも小さい（ステップS5-NO）。即ち、2番目の命令“MOV”の実行マシンサイクル数“4”は、最大実行マシンサイクル数“4”となる。このとき、実行サイクル数計測部13は、実行サイクル検索処理として、1番目の命令“ADD”の実行マシンサイクル数“2”を最大実行マシンサイクル数“4”に変更するように、実行マシンサイクル補正部41を制御する。同時に、実行サイクル数計測部13は、実行結果60として、最大実行マシンサイクル数“4”と1番目の命令“ADD”の実行マシンサイクル数“2”との差分“2”を、1番目の実行命令群“ADD、MOV”の総実行マシンサイクル数“2”に加算する（ステップS7）。

30

40

【0034】

一方、先行命令群の実行マシンサイクル数は、現命令の実行マシンサイクル数以上である（ステップS5-YES）。この場合、実行マシンサイクル補正部41は、先行命令群の実行マシンサイクル数を変更しない。同時に、実行サイクル数計測部13は、実行結果60として、先行命令群と現命令とを含む実行命令群の総実行マシンサイクル数を更新しない（ステップS6）。

50

【 0 0 3 5 】

シミュレーションエンジン部 7 は、1 番目から最終番目（9 番目）の命令 “ A D D、M O V、S U B、S U B、J M P、L D、L D、A D D、M O V ” が終了するまで（ステップ S 9 - Y E S）、上記の動作を実行する。

【 0 0 3 6 】

以上の説明により、本発明の第 1 実施形態によるシミュレーション装置 4 では、プロセッサ 7 0 がパイプライン群 7 1、7 2 により同時実行可能な実行命令群を検索し、その実行マシンサイクル数を最大実行マシンサイクル数に変更することにより、複数のパイプライン（パイプライン群 7 1、7 2）を考慮したプロセッサ 7 0 の動作をシミュレートすることができる。

10

【 0 0 3 7 】

（第 2 実施形態）

本発明の第 2 実施形態によるシミュレーション装置 4 では、第 1 実施形態と重複する説明を省略する。

【 0 0 3 8 】

図 1 1 は、シミュレーションエンジン部 7 の構成を示している。シミュレーションエンジン部 7 は、更に、使用レジスタ情報記憶部 5 0 を具備している。

【 0 0 3 9 】

図 1 2 は、使用レジスタ情報記憶部 5 0 を示している。使用レジスタ情報記憶部 5 0 には、演算命令を識別する識別子と、各演算命令を実行するときに用いるレジスタ名とが格納されている。レジスタ名は、シミュレート対象のプロセッサ 7 0 が備えているレジスタ 7 4 であり、デバッグ対象のプログラムで各演算命令を実行するときに用いるものである。例えば、デバッグ対象のプログラムが命令 “ A D D、M O V、S U B、S U B、J M P、L D、L D、A D D、M O V ” という様に構成されている場合に、“ A D D、S U B、S U B、A D D ” の演算が “ A D D R 1 1、R 1 2（ $R 1 1 = R 1 1 + R 1 2$ の意味）、S U B R 1、R 2（ $R 1 = R 1 - R 2$ の意味）、S U B R 3、R 1（ $R 3 = R 3 - R 1$ の意味）、A D D R 2 1、R 2 2（ $R 2 1 = R 2 1 + R 2 2$ の意味）” となっている場合、それぞれの演算を識別する識別子とレジスタ名が使用レジスタ情報記憶部 5 0 に格納される。

20

【 0 0 4 0 】

命令処理部 1 1 は、設計者の入力装置 2 の操作により、命令群 “ A D D、M O V、S U B、S U B、J M P、L D、L D、A D D、M O V ” を記憶装置（命令データ格納部 6 に対応）から読み出して、パイプライン状態制御部 1 2 に出力するとともに、各命令で使用するレジスタ名と各命令を識別するための識別子とを使用レジスタ情報記憶部 5 0 に出力する。パイプライン状態制御部 1 2 は、同時実行可能な命令の組を検索する同時実行命令検索処理を実行させるために、その命令群を同時実行条件判定部 3 1 に出力する。

30

【 0 0 4 1 】

同時実行命令検索処理において、図 1 3 に示されるように、同時実行条件判定部 3 1 は、同時実行条件格納部 3 2 を参照して、1 命令ずつ処理される 1 番目から最終番目までの命令群 “ A D D、M O V、S U B、S U B、J M P、L D、L D、A D D、M O V ” を 1 番目から最終番目までの複数の実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、__ ”、“ L D、__ ”、“ L D、A D D ”、“ M O V、__ ” に分割する。“ __ ” は、同時実行条件により実行されない命令を表している。複数の実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、__ ”、“ L D、__ ”、“ L D、A D D ”、“ M O V、__ ” の各々は、プロセッサ 7 0 がパイプライン群 7 1、7 2 により同時実行可能な命令の組を表している。同時実行条件判定部 3 1 は、1 番目から最終番目までこの順に実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、__ ”、“ L D、__ ”、“ L D、A D D ”、“ M O V、__ ” をパイプライン状態記憶装置 2 0 に格納する。このとき、同時実行条件判定部 3 1 は、実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、__ ”、“ L D、__ ”、“ L D、A D D ”、“ M O V、__ ” をそれぞれパイプライン

40

50

状態記憶部群 2 1、2 2 に格納する。

【 0 0 4 2 】

次に、同時実行条件判定部 3 1 は、パイプライン状態記憶部群 2 1、2 2 と使用レジスタ情報記憶部 5 0 とを参照して、実行命令群 “ ADD、MOV ”、“ SUB、SUB ”、“ JMP、_ ”、“ LD、_ ”、“ LD、ADD ”、“ MOV、_ ” が、同一のレジスタを用いて演算を行う演算実行命令群であるか否かを検索する。検索の結果、実行命令群 “ SUB、SUB ” が、同一のレジスタ “ R 1 ” を用いて演算 “ $R 1 = R 1 - R 2$ ”、“ $R 3 = R 3 - R 1$ ” を行う演算実行命令群である。即ち、演算実行命令群 “ SUB、SUB ” は、同一のレジスタ “ R 1 ” を用いるため、同時に実行することができない。この同一のレジスタを用いることにより同時に実行することができない命令をハザード実行命令と呼ぶ。

10

【 0 0 4 3 】

そこで、図 1 3 に示されるように、同時実行条件判定部 3 1 は、演算実行命令群 “ SUB、SUB ” を 1 番目から最終番目までの複数のハザード実行命令群 “ SUB、_ ”、“ SUB、_ ” に分割する。複数のハザード実行命令群 “ SUB、_ ”、“ SUB、_ ” の各々は、同一のレジスタ “ R 1 ” を用いていない命令の組を表し、“ _ ” は、同時実行条件により実行されない命令を表している。同時実行条件判定部 3 1 は、パイプライン状態記憶部群 2 1、2 2 に格納された演算実行命令群 “ SUB、SUB ” に代えて、ハザード実行命令群 “ SUB、_ ”、“ SUB、_ ” を実行命令群として 1 番目から最終番目までこの順にパイプライン状態記憶部群 2 1、2 2 のうちの 1 つのパイプライン状態記憶部（例えばパイプライン状態記憶部 2 1）に格納する。この場合、パイプライン状態記憶部群 2 1、2 2 には、実行命令群として、“ ADD、MOV ”、“ SUB、_ ”、“ SUB、_ ”、“ JMP、_ ”、“ LD、_ ”、“ LD、ADD ”、“ MOV、_ ” が格納されている。同時実行条件判定部 3 1 は、同時実行命令検索処理の終了をパイプライン状態制御部 1 2 に通知する。

20

【 0 0 4 4 】

パイプライン状態制御部 1 2 は、実行マシンサイクル数を補正する実行サイクル検索処理を実行させるために、パイプライン状態記憶部群 2 1、2 2 に格納された実行命令群 “ ADD、MOV ”、“ SUB、_ ”、“ SUB、_ ”、“ JMP、_ ”、“ LD、_ ”、“ LD、ADD ”、“ MOV、_ ” を読み出し、実行サイクル数計測部 1 3 に出力する。実行サイクル数計測部 1 3 は、その実行命令群を実行サイクル補正部 4 1 に出力する。

30

【 0 0 4 5 】

実行サイクル検索処理において、実行サイクル補正部 4 1 は、実行サイクル数計測部 1 3 からの実行命令群 “ ADD、MOV ”、“ SUB、_ ”、“ SUB、_ ”、“ JMP、_ ”、“ LD、_ ”、“ LD、ADD ”、“ MOV、_ ” を受け取る。又は、実行サイクル補正部 4 1 は、実行サイクル検索処理を実行させる通知を実行サイクル数計測部 1 3 から受け取ったとき、パイプライン状態記憶部群 2 1、2 2 を参照してもよい。実行サイクル補正部 4 1 は、実行マシンサイクル数格納部 4 2 を参照して、パイプライン群 7 1、7 2 によって実行される実行命令群 “ ADD、MOV ”、“ SUB、_ ”、“ SUB、_ ”、“ JMP、_ ”、“ LD、_ ”、“ LD、ADD ”、“ MOV、_ ” の実行マシンサイクル数 “ 2、4 ”、“ 2、_ ”、“ 2、_ ”、“ 6、_ ”、“ 1、_ ”、“ 1、2 ”、“ 4、_ ” の中から、最も大きい実行マシンサイクル数である最大実行マシンサイクル数 “ 4 ”、“ 2 ”、“ 2 ”、“ 6 ”、“ 1 ”、“ 2 ”、“ 4 ” を検索する。検索の結果、実行マシンサイクル補正部 4 1 は、実行命令群 “ ADD、MOV ”、“ SUB、_ ”、“ SUB、_ ”、“ JMP、_ ”、“ LD、_ ”、“ LD、ADD ”、“ MOV、_ ” の実行マシンサイクル数 “ 2、4 ”、“ 2、_ ”、“ 2、_ ”、“ 6、_ ”、“ 1、_ ”、“ 1、2 ”、“ 4、_ ” を最大実行マシンサイクル数 “ 4 ”、“ 2 ”、“ 2 ”、“ 6 ”、“ 1 ”、“ 2 ”、“ 4 ” に変更する。実行マシンサイクル補正部 4 1 は、その最大実行マシンサイクル数 “ 4 ”、“ 2 ”、“ 2 ”、“ 6 ”、“ 1 ”、“ 2 ”、“ 4 ” を命令実行部 1 0 に出力し、実行サイクル検索処理の終了を通知する。

40

50

【 0 0 4 6 】

このとき、図 1 4 に示されるように、命令実行部 1 0 のパイプライン状態制御部 1 2 は、パイプライン状態記憶部群 2 1、2 2 に格納された実行命令群 “ A D D、M O V ”、“ S U B、_ ”、“ S U B、_ ”、“ J M P、_ ”、“ L D、_ ”、“ L D、A D D ”、“ M O V、_ ”の実行マシンサイクル数 “ 2、4 ”、“ 2、_ ”、“ 2、_ ”、“ 6、_ ”、“ 1、_ ”、“ 1、2 ”、“ 4、_ ”を最大実行マシンサイクル数 “ 4 ”、“ 2 ”、“ 2 ”、“ 6 ”、“ 1 ”、“ 2 ”、“ 4 ”に更新し、命令実行部 1 0 の実行サイクル数計測部 1 3 に出力する。

【 0 0 4 7 】

実行サイクル数計測部 1 3 は、図 1 5 に示されるような実行結果 6 0 を出力装置 3 に出力する。実行結果 6 0 は、パイプライン状態記憶部群 2 1、2 2 に格納された実行命令 “ A D D、S U B、S U B、J M P、L D、L D、M O V ”、“ M O V、_、_、_、_、A D D、_ ”と、その実行マシンサイクル数 “ 4、2、2、6、1、2、4 ”の合計値 “ 2 1 ”を含んでいる。この場合、命令群 “ A D D、M O V、S U B、S U B、J M P、L D、L D、A D D、M O V ”の実行マシンサイクル数 “ 2、4、2、2、6、1、1、2、4 ”の合計値 “ 2 4 ”よりも小さい。また、1クロックの周期を 1 0 [n s e c]とした場合、上記の合計値 “ 2 1 ”は、処理時間 [n s e c]に対応し、実行命令 “ A D D、S U B、S U B、J M P、L D、L D、M O V ”、“ M O V、_、_、_、_、A D D、_ ”に対する処理時間 [n s e c]は、“ 4 0、6 0、1 2 0、1 3 0、1 5 0、1 9 0 ” [n s e c]により表される。ここで、1クロックの周期については、本実施形態を説明する便宜上、実際よりも極端に大きい値を記載している。

【 0 0 4 8 】

本発明の第 2 実施形態によるシミュレーション装置 4 では、命令群 “ A D D、M O V、S U B、S U B、J M P、L D、L D、A D D、M O V ”に対して、プロセッサ 7 0 がパイプライン群 7 1、7 2 により同時実行可能な命令の組として実行命令群 “ A D D、M O V ”、“ S U B、S U B ”、“ J M P、_ ”、“ L D、_ ”、“ L D、A D D ”、“ M O V、_ ”を検索する。上述のように、順番の規定があり、実行命令群 “ S U B、S U B ”が、同一のレジスタ “ R 1 ”を用いて演算 “ $R 1 = R 1 - R 2$ ”、“ $R 3 = R 3 - R 1$ ”を行う場合、同時に実行することができない。

【 0 0 4 9 】

そこで、本発明の第 2 実施形態によるシミュレーション装置 4 では、実行命令群（演算実行命令群） “ S U B、S U B ”に対して同一のレジスタ “ R 1 ”を用いていない命令の組として実行命令群（ハザード実行命令群） “ S U B、_ ”、“ S U B、_ ”を検索する。シミュレーション装置 4 は、命令群 “ A D D、M O V、S U B、S U B、J M P、L D、L D、A D D、M O V ”が処理される順番を考慮して、実行命令群 “ A D D、M O V ”、“ S U B、_ ”、“ S U B、_ ”、“ J M P、_ ”、“ L D、_ ”、“ L D、A D D ”、“ M O V、_ ”の実行マシンサイクル数 “ 2、4 ”、“ 2、_ ”、“ 2、_ ”、“ 6、_ ”、“ 1、_ ”、“ 1、2 ”、“ 4、_ ”を最大実行マシンサイクル数 “ 4 ”、“ 2 ”、“ 2 ”、“ 6 ”、“ 1 ”、“ 2 ”、“ 4 ”に変更する。このように、本発明の第 2 実施形態によるシミュレーション装置 4 では、プロセッサ 7 0 がパイプライン群 7 1、7 2 により実行命令群と、同一のレジスタ “ R 1 ”を用いていない実行命令群とを検索し、その実行マシンサイクル数を最大実行マシンサイクル数に変更することにより、複数のパイプライン（パイプライン群 7 1、7 2 ）を考慮したプロセッサ 7 0 の動作をシミュレートすることができる。

【 0 0 5 0 】

図 1 6 は、シミュレーションエンジン部 7 の動作を示すフローチャートである。

【 0 0 5 1 】

例えば、1 番目の実行命令群 “ A D D、M O V ”として、パイプライン状態記憶部群 2 1 の 1 レコード目に 1 番目の命令 “ A D D ”が格納され、パイプライン状態記憶部群 2 1 の 1 レコード目に 2 番目の命令 “ M O V ”が格納されている。

【 0 0 5 2 】

次に、命令処理部 1 1 は、3 番目の命令 “ S U B ” を命令データ格納部 6 から読み出す（ステップ S 9 - N O）。同時に、実行サイクル数計測部 1 3 は、実行サイクル数格納部 4 2 を参照して、3 番目の命令 “ S U B ” の実行マシンサイクル数 “ 2 ” を取得するように、実行サイクル補正部 4 1 を制御する（ステップ S 1）。パイプライン状態制御部 1 2 は、同時実行命令検索処理として、3 番目の命令 “ S U B ” をパイプライン状態記憶部群 2 1 の 2 レコード目に格納するように、同時実行条件判定部 3 1 を制御する（ステップ S 2 - N O）。このとき、実行サイクル数計測部 1 3 は、実行結果 6 0 として、3 番目の命令 “ S U B ” の実行マシンサイクル数 “ 2 ” を、2 番目の実行命令群の総実行マシンサイクル数 “ 0 ” に加算する（ステップ S 8）。

10

【 0 0 5 3 】

次に、命令処理部 1 1 は、4 番目の命令 “ S U B ” を命令データ格納部 6 から読み出す（ステップ S 9 - N O）。同時に、実行サイクル数計測部 1 3 は、実行サイクル数格納部 4 2 を参照して、4 番目の命令 “ S U B ” の実行マシンサイクル数 “ 2 ” を取得するように、実行サイクル補正部 4 1 を制御する（ステップ S 1）。ここで、パイプライン状態記憶部群 2 1 の 2 レコード目には、3 番目の命令 “ S U B ” が格納されている（ステップ S 2 - Y E S）。パイプライン状態制御部 1 2 は、同時実行命令検索処理として、同時実行条件格納部 3 2 を参照するように、同時実行条件判定部 3 1 を制御する（ステップ S 3）。検索の結果、3 番目の命令（先行命令群） “ S U B ” と 2 番目の命令 “ S U B ” は、同時実行可能である（ステップ S 4 - Y E S）。この場合、同時実行条件判定部 3 1 は、2 番目の実行命令群 “ S U B、S U B ” として、パイプライン状態記憶部群 2 1 に格納された 3 番目の命令 “ S U B ” に対応付けて、4 番目の命令 “ S U B ” をパイプライン状態記憶部群 2 1 の 2 レコード目に格納する。

20

【 0 0 5 4 】

この同時実行命令検索処理において、同時実行条件判定部 3 1 は、パイプライン状態記憶部群 2 1、2 2 と使用レジスタ情報記憶部 5 0 とを参照して、3 番目の命令（先行命令群） “ S U B ” と 4 番目の命令（現命令） “ S U B ” とが同一のレジスタを用いて演算を行うか否かを検索する（ステップ S 10）。検索の結果、3 番目の命令 “ S U B ” と 4 番目の命令 “ S U B ” は、同一のレジスタ “ R 1 ” を用いて演算 “ $R 1 = R 1 - R 2$ ”、“ $R 3 = R 3 - R 1$ ” を行うため、同時に実行することができない（ステップ S 10 - N O）。この場合、同時実行条件判定部 3 1 は、パイプライン状態記憶部群 2 1、2 2 の 2 レコード目に格納された実行命令群 “ S U B、S U B ” に代えて、2 番目の実行命令群 “ S U B、_ ” として、3 番目の命令 “ S U B ” をパイプライン状態記憶部群 2 1 の 2 レコード目に格納し、3 番目の実行命令群 “ S U B、_ ” として、4 番目の命令 “ S U B ” をパイプライン状態記憶部群 2 1 の 3 レコード目に格納する。このとき、実行サイクル数計測部 1 3 は、実行結果 6 0 として、4 番目の命令 “ S U B ” の実行マシンサイクル数 “ 2 ” を、3 番目の実行命令群の総実行マシンサイクル数 “ 0 ” に加算する（ステップ S 8）。

30

【 0 0 5 5 】

一方、先行命令群と現命令とが同一のレジスタを用いて演算を行わない場合（ステップ S 10 - Y E S）、上記のステップ S 5 以降が実行される。

40

【 0 0 5 6 】

以上の説明により、本発明の第 2 実施形態によるシミュレーション装置 4 では、プロセッサ 7 0 がパイプライン群 7 1、7 2 により同時実行可能な実行命令群と、同一のレジスタ “ R 1 ” を用いていない実行命令群とを検索し、その実行マシンサイクル数を最大実行マシンサイクル数に変更することにより、複数のパイプライン（パイプライン群 7 1、7 2）を考慮したプロセッサ 7 0 の動作をシミュレートすることができる。

【 0 0 5 7 】

本発明の第 1 及び第 2 実施形態では、その特徴を明確に説明するために 2 つのパイプライン間で同時に実行できない命令がある場合について説明したが、シミュレートする対象のプロセッサが複数のパイプラインを有し、各パイプライン間で同時に実行できない命令

50

がない場合であっても、すなわち、同時実行条件に N G のものがなくても、問題なく処理できることは言うまでもない。

【図面の簡単な説明】

【0058】

【図1】図1は、本発明のシミュレーション装置4の構成を示している。(第1実施形態、第2実施形態)

【図2】図2は、コンピュータの構成を示している。(第1実施形態、第2実施形態)

【図3】図3は、本発明でシミュレートを実行しようとしているプロセッサ70のパイプライン部分の概略図を示している。

【図4】図4は、シミュレーションエンジン部7の構成を示している。(第1実施形態)

【図5】図5は、同時実行条件格納部32を示している。(第1実施形態、第2実施形態)

【図6】図6は、実行サイクル数格納部42を示している。(第1実施形態、第2実施形態)

【図7A】図7Aは、本発明のシミュレーション装置4の動作を説明するための図である。(第1実施形態)

【図7B】図7Bは、本発明のシミュレーション装置4の動作を説明するための図である。(第1実施形態)

【図8A】図8Aは、本発明のシミュレーション装置4の動作を説明するための図である。(第1実施形態)

【図8B】図8Bは、本発明のシミュレーション装置4の動作を説明するための図である。(第1実施形態)

【図9】図9は、実行結果60を示している。(第1実施形態)

【図10】図10は、シミュレーションエンジン部7の動作を示すフローチャートである。(第1実施形態)

【図11】図11は、シミュレーションエンジン部7の構成を示している。(第2実施形態)

【図12】図12は、使用レジスタ情報記憶部50を示している。(第2実施形態)

【図13】図13は、本発明のシミュレーション装置4の動作を説明するための図である。(第2実施形態)

【図14】図14は、本発明のシミュレーション装置4の動作を説明するための図である。(第2実施形態)

【図15】図15は、実行結果60を示している。(第2実施形態)

【図16】図16は、シミュレーションエンジン部7の動作を示すフローチャートである。(第2実施形態)

【符号の説明】

【0059】

- 1 コンピュータ本体、
- 2 入力装置、
- 3 出力装置、
- 4 シミュレーション装置、
- 5 ユーザインターフェース部、
- 6 命令データ格納部、
- 7 シミュレーションエンジン部、
- 10 命令実行部、
- 11 命令処理部、
- 12 パイプライン状態制御部、
- 13 実行サイクル数計測部、
- 20 パイプライン状態記憶装置、
- 21、22 パイプライン状態記憶部、

10

20

30

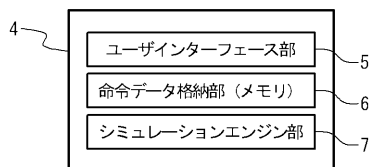
40

50

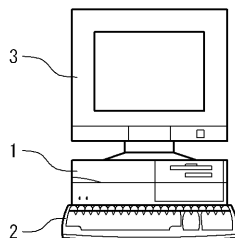
- 3 0 同時実行命令検索部、
- 3 1 同時実行条件判定部、
- 3 2 同時実行条件格納部、
- 4 0 実行サイクル変更部、
- 4 1 実行サイクル補正部、
- 4 2 実行サイクル数格納部、
- 5 0 使用レジスタ情報記憶部、
- 6 0 実行結果、
- 7 0 プロセッサ、
- 7 1、7 2 パイプライン、
- 7 3 メモリ、
- 7 4 レジスタ、
- 8 1 I F (命令フェッチ) ステージ、
- 8 2 - 1、8 2 - 2 D C (デコード) ステージ、
- 8 3 - 1、8 3 - 2 E X (実行) ステージ、
- 8 4 - 1、8 4 - 2 M E (メモリアクセス) ステージ、
- 8 5 - 1、8 5 - 2 W B (ライトバック) ステージ、

10

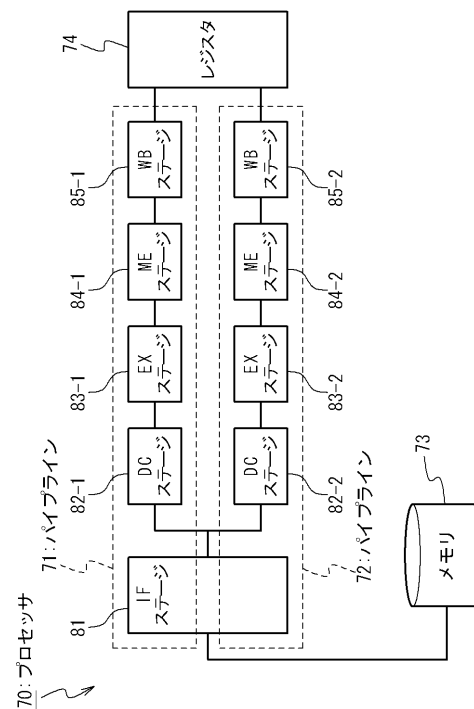
【図 1】



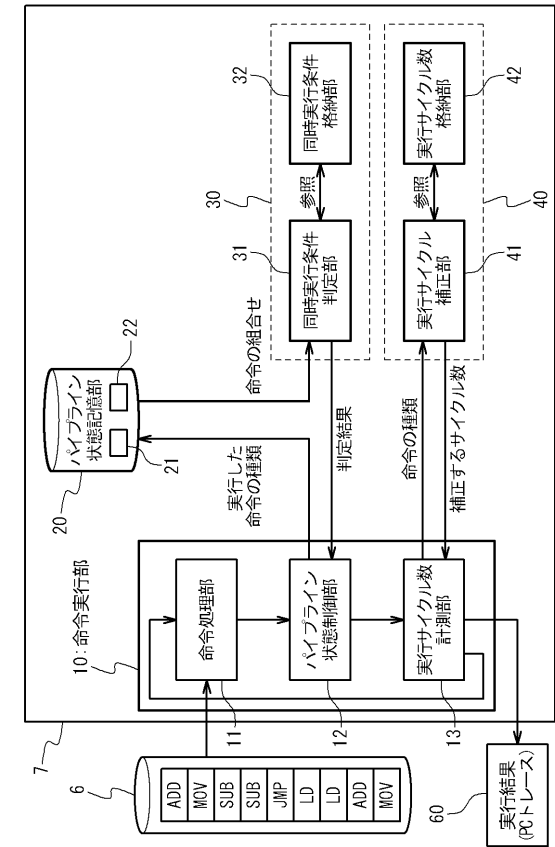
【図 2】



【図 3】



【 図 4 】



【 図 5 】

32

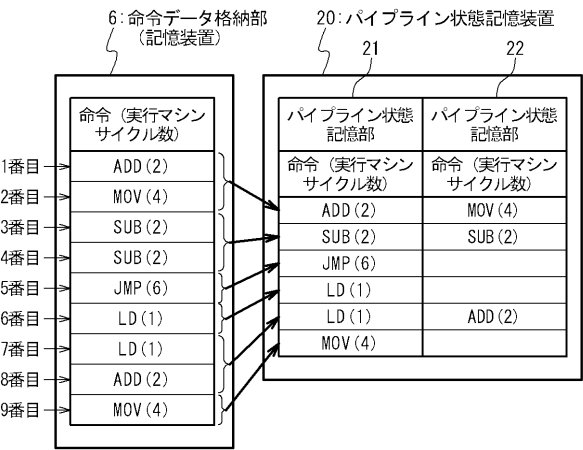
同時実行条件		
命令の組		判定
ADD	MOV	OK
LD	MOV	OK
⋮	⋮	⋮
JMP	MOV	NG
LD	LD	NG
⋮	⋮	⋮

【 図 6 】

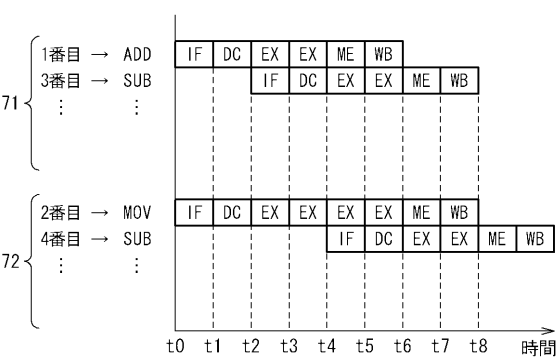
42

命令	実行サイクル数
LD	1
ADD	2
SUB	2
JMP	6
MOV	4
⋮	⋮

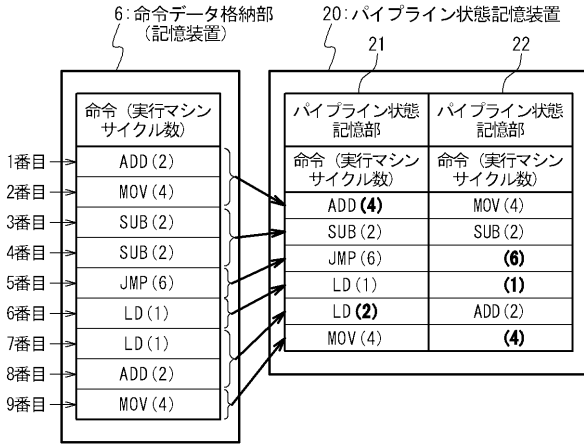
【 図 7 A 】



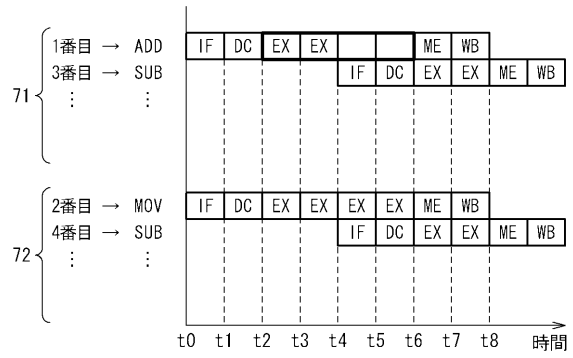
【 図 7 B 】



【図 8 A】



【図 8 B】

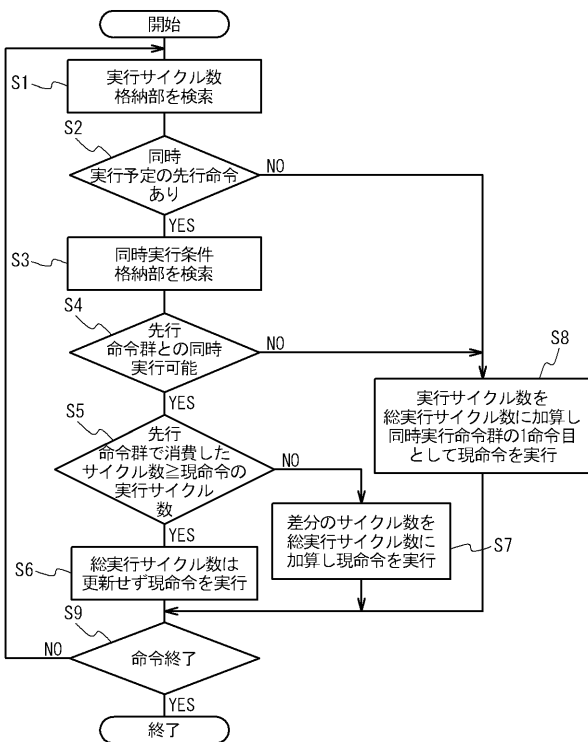


【図 9】

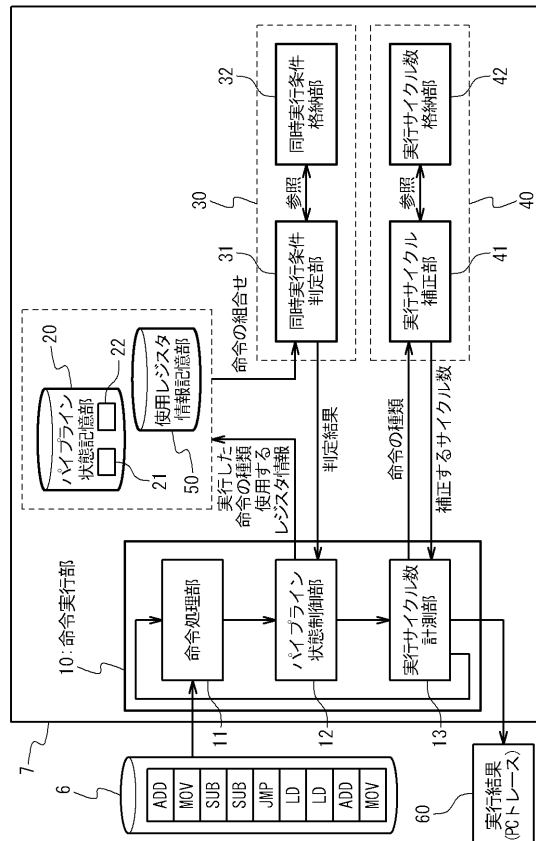
60

処理時間 [nsec]	パイプライン71 により実行される命令	パイプライン72 により実行される命令
40	ADD	MOV
60	SUB	SUB
120	JMP	
130	LD	
150	LD	ADD
190	MOV	

【図 10】



【図 11】

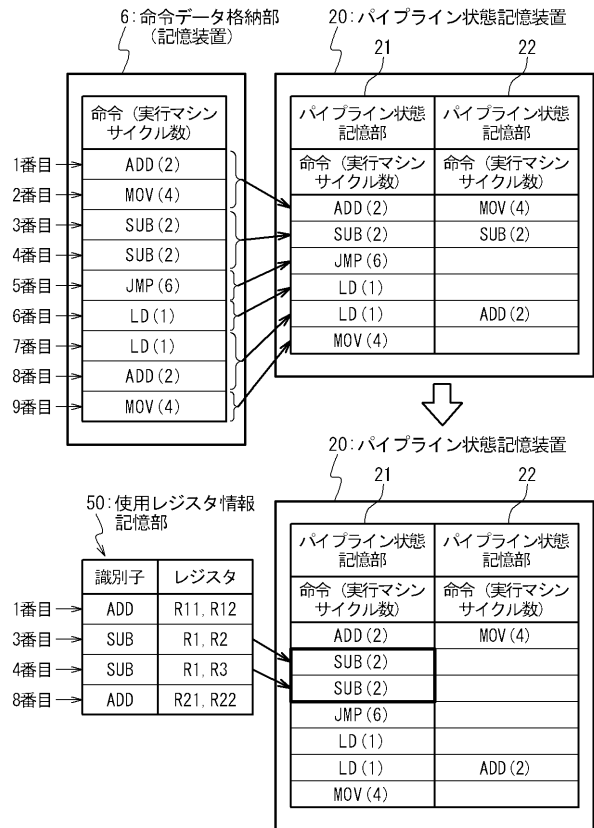


【図 1 2】

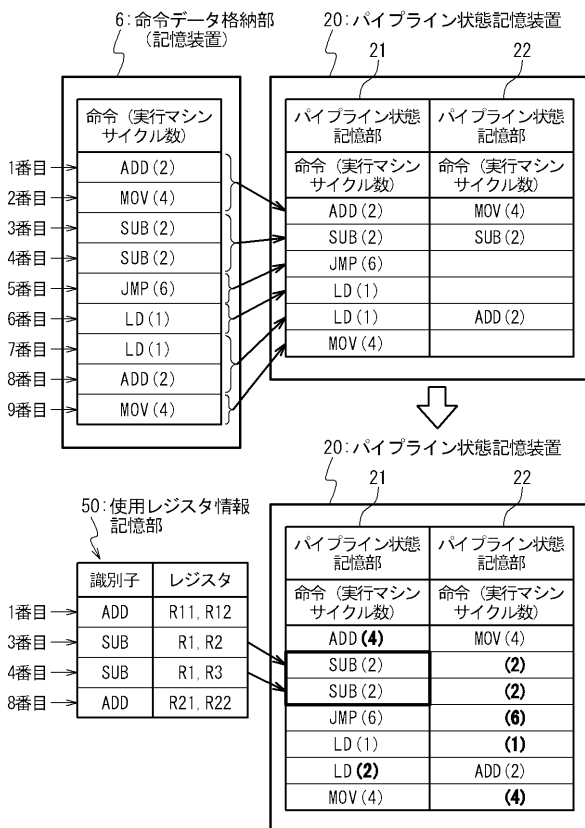
50

識別子	レジスタ
1番目の命令ADD	R11, R12 (R11=R11+R12)
3番目の命令SUB	R1, R2 (R1=R1-R2)
4番目の命令SUB	R3, R1 (R3=R3-R1)
8番目の命令ADD	R21, R22 (R21=R21+R22)

【図 1 3】



【図 1 4】



【図 1 5】

60

処理時間 [nsec]	パイプライン71により実行される命令	パイプライン72により実行される命令
40	ADD	MOV
60	SUB	
80	SUB	
140	JMP	
150	LD	
170	LD	ADD
210	MOV	

【図 16】

