

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2014-182514
(P2014-182514A)

(43) 公開日 平成26年9月29日(2014.9.29)

(51) Int.Cl.	F I	テーマコード (参考)
G06F 9/30 (2006.01)	G06F 9/30 310E	5B033
G06F 9/34 (2006.01)	G06F 9/34 330	
G06F 9/315 (2006.01)	G06F 9/30 340D	

審査請求 未請求 請求項の数 12 O L (全 24 頁)

(21) 出願番号 特願2013-55660 (P2013-55660)
(22) 出願日 平成25年3月18日 (2013.3.18)

(71) 出願人 504143441
国立大学法人 奈良先端科学技術大学院大学
奈良県生駒市高山町8916-5
(74) 代理人 110000338
特許業務法人HARAKENZO WORLD PATENT & TRADEMARK
(72) 発明者 中島 康彦
奈良県生駒市高山町8916-5 国立大学法人奈良先端科学技術大学院大学内
Fターム(参考) 5B033 AA11 BA03 BD03 DD04 DD06

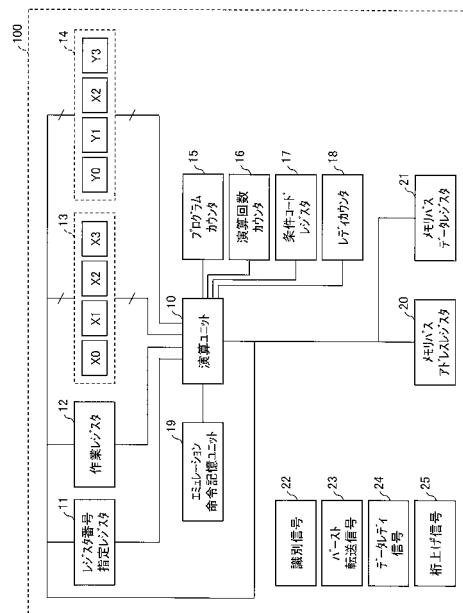
(54) 【発明の名称】 演算装置

(57) 【要約】

【課題】狭ビット幅の演算装置を用いて広ビット幅の演算装置をエミュレート可能なエミュレーション技術を提供する。

【解決手段】Nビット幅の演算ユニット10と、複数のエミュレーション命令を格納するエミュレーション命令記憶ユニット19と、演算ユニット10に入力されるデータを格納するNビット幅の入力演算レジスタ群を複数個有する入力演算レジスタ群13と、演算ユニット10の演算結果を格納するNビット幅の出力演算レジスタを複数個有する出力演算レジスタ群14とを備える演算装置100である。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

Nビット幅の演算ユニットと、
複数のエミュレーション命令を格納する記憶ユニットと、
前記演算ユニットに入力されるデータを格納するNビット幅の入力演算レジスタを複数個有する入力演算レジスタ群と、
前記演算ユニットの演算結果を格納するNビット幅の出力演算レジスタを複数個有する出力演算レジスタ群と
を備えることを特徴とする演算装置。

【請求項 2】

前記演算装置は、前記演算ユニットの演算結果を一時的に格納する作業レジスタをさらに備え、

前記エミュレーション命令は、前記記憶ユニットに含まれるNビットオペランドの値(S)を用いて、前記入力演算レジスタX1の下位(N-S)ビットと前記入力演算レジスタX0の上位Sビットとを連結し、当該連結されたNビットデータを前記作業レジスタに格納する命令であることを特徴とする請求項1に記載の演算装置。

【請求項 3】

前記演算装置は、前記演算ユニットの演算結果を一時的に格納する作業レジスタをさらに備え、

前記エミュレーション命令は、前記記憶ユニットに含まれるNビットのオフセット値を用いてNビットデータを取得し、当該取得されたNビットデータの上位N/2ビットをU、前記Nビットデータの下部N/2ビットと前記作業レジスタの下部N/2ビットの反転値との論理積をL1、前記作業レジスタの上位N/2ビットと前記作業レジスタの下部N/2ビットとの論理積をL2として、前記Uと、前記L1と前記L2との論理和とを連結し、当該連結されたNビットデータを前記作業レジスタに格納する命令であることを特徴とする請求項1に記載の演算装置。

【請求項 4】

前記演算装置は、前記演算ユニットの演算結果を一時的に格納する作業レジスタをさらに備え、

前記エミュレーション命令は、前記記憶ユニットに含まれるNビットのオフセット値を用いてNビットデータを取得し、当該取得されたNビットデータの上位N/2ビットと前記作業レジスタの下部N/2ビットの反転値との論理積をU1、前記作業レジスタの上位N/2ビットと前記作業レジスタの下部N/2ビットとの論理積をU2、前記Nビットデータの下部N/2ビットをLとして、前記U1と前記U2との論理和と、前記Lとを連結し、当該連結されたNビットデータを前記作業レジスタに格納する命令であることを特徴とする請求項1に記載の演算装置。

【請求項 5】

前記演算装置は、

メモリからのデータ転送を行なうことなく、前記演算ユニットが演算する回数をカウントし、当該カウント数を格納する演算回数カウンタと、

4つの条件コードフラグである、ネガティブ、ゼロ、キャリーおよびオーバフローを格納する条件コードレジスタと

をさらに備え、

前記エミュレーション命令は、

前記演算回数カウンタが初期値の場合には、前記出力演算レジスタY3を右に1ビットシフトして前記出力演算レジスタY3に格納し、追い出される前記出力演算レジスタY3の最下位ビットを前記条件コードレジスタに格納し、前記出力演算レジスタY3に格納される値が0であるときに前記条件コードレジスタに1をセットし、前記演算回数カウンタに1を加え、

前記演算回数カウンタが初期値+1の場合には、前記出力演算レジスタY2を右に1ビ

10

20

30

40

50

ットシフトして前記出力演算レジスタ Y 2 に格納し、追い出される前記出力演算レジスタ Y 2 の最下位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 2 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、前記演算回数カウンタに 1 を加え、

前記演算回数カウンタが初期値 + 2 の場合には、前記出力演算レジスタ Y 1 を右に 1 ビットシフトして前記出力演算レジスタ Y 1 に格納し、追い出される前記出力演算レジスタ Y 1 の最下位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 1 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、前記演算回数カウンタに 1 を加え、

前記演算回数カウンタが初期値 + 3 の場合には、前記出力演算レジスタ Y 0 を右に 1 ビットシフトして前記出力演算レジスタ Y 0 に格納し、追い出される前記出力演算レジスタ Y 0 の最下位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 0 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットする命令であることを特徴とする請求項 1 に記載の演算装置。

【請求項 6】

前記エミュレーション命令は、

前記出力演算レジスタ Y 3 を右に 1 ビットシフトして前記出力演算レジスタ Y 3 に格納するとき、前記出力演算レジスタ Y 3 の最上位ビットに 0 または前記条件コードレジスタの値を格納し、

前記出力演算レジスタ Y 2 を右に 1 ビットシフトして前記出力演算レジスタ Y 2 に格納するとき、前記出力演算レジスタ Y 2 の最上位ビットに前記条件コードレジスタの値を格納し、

前記出力演算レジスタ Y 1 を右に 1 ビットシフトして前記出力演算レジスタ Y 1 に格納するとき、前記出力演算レジスタ Y 1 の最上位ビットに前記条件コードレジスタの値を格納し、

前記出力演算レジスタ Y 0 を右に 1 ビットシフトして前記出力演算レジスタ Y 0 に格納するとき、前記出力演算レジスタ Y 0 の最上位ビットに前記条件コードレジスタの値を格納する命令であることを特徴とする請求項 5 に記載の演算装置。

【請求項 7】

前記演算装置は、

メモリからのデータ転送を行なうことなく、前記演算ユニットが演算する回数をカウントし、当該カウント数を格納する演算回数カウンタと、

4 つの条件コードフラグである、ネガティブ、ゼロ、キャリーおよびオーバフローを格納する条件コードレジスタとをさらに備え、

前記エミュレーション命令は、

前記演算回数カウンタが初期値の場合には、前記出力演算レジスタ Y 0 を左に 1 ビットシフトして前記出力演算レジスタ Y 0 に格納し、追い出される前記出力演算レジスタ Y 0 の最上位ビットを前記条件コードレジスタに格納し、前記出力演算レジスタ Y 0 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、前記演算回数カウンタに 1 を加え、

前記演算回数カウンタが初期値 + 1 の場合には、前記出力演算レジスタ Y 1 を左に 1 ビットシフトして前記出力演算レジスタ Y 1 に格納し、追い出される前記出力演算レジスタ Y 1 の最上位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 1 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、前記演算回数カウンタに 1 を加え、

前記演算回数カウンタが初期値 + 2 の場合には、前記出力演算レジスタ Y 2 を左に 1 ビットシフトして前記出力演算レジスタ Y 2 に格納し、追い出される前記出力演算レジスタ Y 2 の最上位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 2 に格納される値が 0 であるときに前記条件コードレジ

10

20

30

40

50

タに 1 をセットし、前記演算回数カウンタに 1 を加え、

前記演算回数カウンタが初期値 + 3 の場合には、前記出力演算レジスタ Y 3 を左に 1 ビットシフトして前記出力演算レジスタ Y 3 に格納し、追い出される前記出力演算レジスタ Y 3 の最上位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 3 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットする命令であることを特徴とする請求項 1 に記載の演算装置。

【請求項 8】

前記エミュレーション命令は、

前記出力演算レジスタ Y 0 を左に 1 ビットシフトして前記出力演算レジスタ Y 0 に格納するとき、前記出力演算レジスタ Y 0 の最下位ビットには 0 または前記条件コードレジスタの値を格納し、

10

前記出力演算レジスタ Y 1 を左に 1 ビットシフトして前記出力演算レジスタ Y 1 に格納するとき、前記出力演算レジスタ Y 1 の最下位ビットには前記条件コードレジスタの値を格納し、

前記出力演算レジスタ Y 2 を左に 1 ビットシフトして前記出力演算レジスタ Y 2 に格納するとき、前記出力演算レジスタ Y 2 の最下位ビットには前記条件コードレジスタの値を格納し、

前記出力演算レジスタ Y 3 を左に 1 ビットシフトして前記出力演算レジスタ Y 3 に格納するとき、前記出力演算レジスタ Y 3 の最下位ビットには前記条件コードレジスタの値を格納する命令であることを特徴とする請求項 7 に記載の演算装置。

20

【請求項 9】

前記演算装置は、前記演算ユニットが次に実行すべき命令のアドレスを格納するプログラムカウンタをさらに備え、

前記エミュレーション命令は、

前記記憶ユニットに含まれるオペランドの上位 $N/2$ ビットが第 1 の値の場合には、前記オペランドの次アドレスに、前記入力演算レジスタ X の下位 $N/2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N/2$ ビットが第 2 の値の場合には、前記オペランドの下位 $N/2$ ビットと前記入力演算レジスタ X の下位 $N/2$ ビットとの論理和を求め、

30

当該論理和の結果が $A11 - 1$ のとき、前記オペランドの次アドレスに、前記入力演算レジスタ X の下位 $N/2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記論理和の結果が $A11 - 1$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N/2$ ビットが第 3 の値の場合には、前記オペランドの下位 $N/2$ ビットと前記入力演算レジスタ X の下位 $N/2$ ビットとの論理積を求め、

当該論理積の結果が $A11 - 0$ のとき、前記オペランドの次アドレスに、前記入力演算レジスタ X の下位 $N/2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

40

前記論理積の結果が $A11 - 0$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N/2$ ビットが第 4 の値の場合には、前記オペランドの下位 $N/2$ ビットと前記入力演算レジスタ X の下位 $N/2$ ビットとの排他論理和を求め、

当該排他論理和の結果が $A11 - 0$ のとき、前記オペランドの次アドレスに、前記入力演算レジスタ X の下位 $N/2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記排他論理和の結果が $A11 - 0$ のとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加える命令であることを特徴とする請求項 1 に記載の演算装置。

50

【請求項 10】

前記演算装置は、前記演算ユニットが次に実行すべき命令のアドレスを格納するプログラムカウンタをさらに備え、

前記エミュレーション命令は、

前記記憶ユニットに含まれるオペランドの上位 $N/2$ ビットが第 1 の値の場合には、前記オペランドの次アドレスに、前記入力演算レジスタ X の上位 $N/2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N/2$ ビットが第 2 の値の場合には、前記オペランドの下位 $N/2$ ビットと前記入力演算レジスタ X の上位 $N/2$ ビットとの論理和を求め、

当該論理和の結果が $A11-1$ のとき、前記オペランドの次アドレスに Q を加えたアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記論理和の結果が $A11-1$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N/2$ ビットが第 3 の値の場合には、前記オペランドの下位 $N/2$ ビットと前記入力演算レジスタ X の上位 $N/2$ ビットとの論理積を求め、

当該論理積の結果が $A11-0$ のとき、前記オペランドの次アドレスに Q を加えたアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記論理積の結果が $A11-0$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N/2$ ビットが第 4 の値の場合には、前記オペランドの下位 $N/2$ ビットと前記入力演算レジスタ X の上位 $N/2$ ビットとの排他論理和を求め、

当該排他論理和の結果が $A11-0$ のとき、前記オペランドの次アドレスに Q を加えたアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記排他論理和の結果が $A11-0$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加える命令であることを特徴とする請求項 1 に記載の演算装置。

【請求項 11】

前記演算装置は、メモリからのバースト転送が開始すると、当該バースト転送開始後、前記メモリから到着するデータブロックをカウントし、当該カウント数を格納するレディカウンタと、

4 つの条件コードフラグである、ネガティブ、ゼロ、キャリーおよびオーバフローを格納する条件コードレジスタと

をさらに備え、

前記エミュレーション命令は、

前記記憶ユニットに含まれる N ビットのオフセット値を用いて、前記メモリから複数の N ビットデータを取得し、

前記レディカウンタが初期値かつメモリバス上に前記メモリから到着した有効なデータがあることを示す信号であるデータレディ信号がオンの場合には、前記出力演算レジスタ Y_0 と前記メモリバス上の N ビットデータ D_0 とを加算して前記出力演算レジスタ Y_0 に格納し、桁上げ信号を前記条件コードレジスタに格納し、前記出力演算レジスタ Y_0 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、

前記レディカウンタが初期値 + 1 かつ前記データレディ信号がオンの場合には、前記出力演算レジスタ Y_1 と前記メモリバス上の N ビットデータ D_1 と前記条件コードレジスタとを加算して前記出力演算レジスタ Y_1 に格納し、桁上げ信号を前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y_1 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、

前記レディカウンタが初期値 + 2 かつ前記データレディ信号がオンの場合には、前記出力演算レジスタ Y_2 と前記メモリバス上の N ビットデータ D_2 と前記条件コードレジスタ

10

20

30

40

50

とを加算して前記出力演算レジスタ Y 2 に格納し、桁上げ信号を前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 2 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、

前記レディカウンタが初期値 + 3 かつ前記データレディ信号がオンの場合には、前記出力演算レジスタ Y 3 と前記メモリバス上の N ビットデータ D 3 と前記条件コードレジスタを加算して前記出力演算レジスタ Y 3 に格納し、桁上げ信号を前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 3 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットする命令であることを特徴とする請求項 1 に記載の演算装置。

【請求項 1 2】

前記演算装置は、メモリからのバースト転送が開始すると、当該バースト転送開始後、前記メモリから到着するデータブロックをカウントし、当該カウント数を格納するレディカウンタと、

4 つの条件コードフラグである、ネガティブ、ゼロ、キャリーおよびオーバフローを格納する条件コードレジスタとをさらに備え、

前記エミュレーション命令は、

前記記憶ユニットに含まれる N ビットのオフセット値を用いて、前記メモリから複数の N ビットデータを取得し、

前記レディカウンタが初期値かつメモリバス上に前記メモリから到着した有効なデータがあることを示す信号であるデータレディ信号がオンの場合には、前記出力演算レジスタ Y 0 とメモリバス上の N ビットデータ D 0 とを論理演算して前記出力演算レジスタ Y 0 に格納し、前記出力演算レジスタ Y 0 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、

前記レディカウンタが初期値 + 1 かつ前記データレディ信号がオンの場合には、前記出力演算レジスタ Y 1 と前記メモリバス上の N ビットデータ D 1 とを論理演算して前記出力演算レジスタ Y 1 に格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 1 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、

前記レディカウンタが初期値 + 2 かつ前記データレディ信号がオンの場合には、前記出力演算レジスタ Y 2 と前記メモリバス上の N ビットデータ D 2 とを論理演算して前記出力演算レジスタ Y 2 に格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 2 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、

前記レディカウンタが初期値 + 3 かつ前記データレディ信号がオンの場合には、前記出力演算レジスタ Y 3 と前記メモリバス上の N ビットデータ D 3 とを論理演算して前記出力演算レジスタ Y 3 に格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 3 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットする命令であることを特徴とする請求項 1 に記載の演算装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、演算装置に関する。特に、本発明は、狭ビット幅の演算装置を用いて広ビット幅の演算装置をエミュレート可能なエミュレーション技術に関する。

【背景技術】

【0002】

コンピュータ製造業者は、新たな命令セットへ移行することにより新たなコンピュータにおいてより大きなアドレス空間へ遷移する傾向がある。新たな命令組へ移行することは、ユーザ及び製造業者の両方にとって潜在的に致命的な結果をもたらすことがある。ユーザの観点からは、このことは、古いマシンに対して書かれたプログラムが新しいマシンでは稼動しないことを意味している。ソフトウェアにかなりの投資をしたユーザは、ソフトウェアを変換するか又は置換するための費用を支払うか又は新しいマシンにおいて組み込

10

20

30

40

50

まれた種々の進んだものの利点を諦めるかの何れかの不快な決断に迫られる。製造業者の観点からは、このような移行は、ユーザの憤りを買ひ且つこのような新しいマシンの当初の販売が停滞することとなる蓋然性がある。

【0003】

このような状況を考慮して、例えば、特許文献1には、コンピュータアーキテクチャに関するものであって、より大きなワード寸法及びアドレス空間を持つために命令セットアーキテクチャを拡張する技術が開示されている。この特許文献1に開示されたプロセッサによれば、例えば、32ビット整数演算をサブセットとして有する64ビット整数演算を実行し、且つ、32ビットアドレッシングをサブセットとして有する64ビットアドレッシングを実行することができる。

10

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特許第3554342号公報(2004年5月14日登録)

【発明の概要】

【発明が解決しようとする課題】

【0005】

近年、特許文献1に開示された技術とは異なり、狭ビット幅の演算装置を用いて広ビット幅の命令セットを実行する技術が求められて来ている。特許文献1に開示された技術は、広ビット幅の演算装置を用いて狭ビット幅の命令セットを実行する技術であるからである。

20

【0006】

より具体的には、動作が超高速であり、消費電力も極めて小さい、次世代デバイスの開発が盛んに進められている。しかし、価格・安定性・歩留りの観点から、現状のシリコンデバイスを前提として設計された大規模回路をそのまま利用してコンピュータを構成することは極めて困難である。

【0007】

実用化を急ぐ場合、まずは極めて小規模の回路を用いてコンピュータを設計する必要がある。しかし、過去に開発された技術では、8ビットまたは4ビットの世界のアプリケーションプログラムやOS(Operating System)しか動作させることができない。従来の8ビットコンピュータや4ビットコンピュータ上で動作するソフトウェアにより、32ビットアプリケーションをエミュレートすることは可能であるものの、極めて低速なコンピュータしか実現することができない。

30

【0008】

このため、エミュレーションに特化した8ビットコンピュータを構成し、実用レベルのアプリケーションを小型CPU(Central Processing Unit)上で動作可能とする技術が必要である。

【0009】

このように具体的に説明したとおり、狭ビット幅の演算装置を用いて広ビット幅の命令セットを実行する技術、すなわち、狭ビット幅の演算装置を用いて広ビット幅の演算装置をエミュレート可能なエミュレーション技術が求められて来ている。

40

【0010】

このような技術は、上で述べたとおり、特許文献1に開示された技術と異なるものである。

【0011】

本発明は、このような課題に鑑みてなされたものであり、狭ビット幅の演算装置を用いて広ビット幅の演算装置をエミュレート可能なエミュレーション技術を提供することを目的とする。

【課題を解決するための手段】

【0012】

50

上記の課題を解決するために、本発明の一実施形態に係る演算装置は、
 Nビット幅の演算ユニットと、
 複数のエミュレーション命令を格納する記憶ユニットと、
 前記演算ユニットに入力されるデータを格納するNビット幅の入力演算レジスタを複数
 個有する入力演算レジスタ群と、
 前記演算ユニットの演算結果を格納するNビット幅の出力演算レジスタを複数個有する
 出力演算レジスタ群と
 を備える。

【0013】

前記演算装置は、前記演算ユニットの演算結果を一時的に格納する作業レジスタをさら
 に備え、

前記エミュレーション命令は、前記記憶ユニットに含まれるNビットオペランドの値(
 S)を用いて、前記入力演算レジスタX1の下位(N-S)ビットと前記入力演算レジス
 タX0の上位Sビットとを連結し、当該連結されたNビットデータを前記作業レジスタに
 格納する命令であることが好ましい。

【0014】

前記演算装置は、前記演算ユニットの演算結果を一時的に格納する作業レジスタをさら
 に備え、

前記エミュレーション命令は、前記記憶ユニットに含まれるNビットのオフセット値を
 用いてNビットデータを取得し、当該取得されたNビットデータの上位N/2ビットをU
 、前記Nビットデータの下部N/2ビットと前記作業レジスタの下部N/2ビットの反転
 値との論理積をL1、前記作業レジスタの上部N/2ビットと前記作業レジスタの下部N
 /2ビットとの論理積をL2として、前記Uと、前記L1と前記L2との論理和とを連結
 し、当該連結されたNビットデータを前記作業レジスタに格納する命令であることが好ま
 しい。

【0015】

前記演算装置は、前記演算ユニットの演算結果を一時的に格納する作業レジスタをさら
 に備え、

前記エミュレーション命令は、前記記憶ユニットに含まれるNビットのオフセット値を
 用いてNビットデータを取得し、当該取得されたNビットデータの上位N/2ビットと前
 記作業レジスタの下部N/2ビットの反転値との論理積をU1、前記作業レジスタの上部
 N/2ビットと前記作業レジスタの下部N/2ビットとの論理積をU2、前記Nビットデ
 ータの下部N/2ビットをLとして、前記U1と前記U2との論理和と、前記Lとを連結
 し、当該連結されたNビットデータを前記作業レジスタに格納する命令であることが好ま
 しい。

【0016】

前記演算装置は、

メモリからのデータ転送を行なうことなく、前記演算ユニットが演算する回数をカウン
 トし、当該カウント数を格納する演算回数カウンタと、

4つの条件コードフラグである、ネガティブ、ゼロ、キャリーおよびオーバフローを格
 納する条件コードレジスタと
 をさらに備え、

前記エミュレーション命令は、

前記演算回数カウンタが初期値の場合には、前記出力演算レジスタY3を右に1ビット
 シフトして前記出力演算レジスタY3に格納し、追い出される前記出力演算レジスタY3
 の最下位ビットを前記条件コードレジスタに格納し、前記出力演算レジスタY3に格納さ
 れる値が0であるときに前記条件コードレジスタに1をセットし、前記演算回数カウンタ
 に1を加え、

前記演算回数カウンタが初期値+1の場合には、前記出力演算レジスタY2を右に1ピ
 ットシフトして前記出力演算レジスタY2に格納し、追い出される前記出力演算レジスタ

10

20

30

40

50

Y 2 の最下位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 2 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、前記演算回数カウンタに 1 を加え、

前記演算回数カウンタが初期値 + 2 の場合には、前記出力演算レジスタ Y 1 を右に 1 ビットシフトして前記出力演算レジスタ Y 1 に格納し、追い出される前記出力演算レジスタ Y 1 の最下位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 1 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、前記演算回数カウンタに 1 を加え、

前記演算回数カウンタが初期値 + 3 の場合には、前記出力演算レジスタ Y 0 を右に 1 ビットシフトして前記出力演算レジスタ Y 0 に格納し、追い出される前記出力演算レジスタ Y 0 の最下位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 0 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットする命令であることが好ましい。

10

【 0 0 1 7 】

前記エミュレーション命令は、

前記出力演算レジスタ Y 3 を右に 1 ビットシフトして前記出力演算レジスタ Y 3 に格納するとき、前記出力演算レジスタ Y 3 の最上位ビットに 0 または前記条件コードレジスタの値を格納し、

前記出力演算レジスタ Y 2 を右に 1 ビットシフトして前記出力演算レジスタ Y 2 に格納するとき、前記出力演算レジスタ Y 2 の最上位ビットに前記条件コードレジスタの値を格納し、

20

前記出力演算レジスタ Y 1 を右に 1 ビットシフトして前記出力演算レジスタ Y 1 に格納するとき、前記出力演算レジスタ Y 1 の最上位ビットに前記条件コードレジスタの値を格納し、

前記出力演算レジスタ Y 0 を右に 1 ビットシフトして前記出力演算レジスタ Y 0 に格納するとき、前記出力演算レジスタ Y 0 の最上位ビットに前記条件コードレジスタの値を格納する命令であることが好ましい。

【 0 0 1 8 】

前記演算装置は、

メモリからのデータ転送を行なうことなく、前記演算ユニットが演算する回数をカウントし、当該カウント数を格納する演算回数カウンタと、

30

4 つの条件コードフラグである、ネガティブ、ゼロ、キャリーおよびオーバフローを格納する条件コードレジスタとをさらに備え、

前記エミュレーション命令は、

前記演算回数カウンタが初期値の場合には、前記出力演算レジスタ Y 0 を左に 1 ビットシフトして前記出力演算レジスタ Y 0 に格納し、追い出される前記出力演算レジスタ Y 0 の最上位ビットを前記条件コードレジスタに格納し、前記出力演算レジスタ Y 0 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、前記演算回数カウンタに 1 を加え、

40

前記演算回数カウンタが初期値 + 1 の場合には、前記出力演算レジスタ Y 1 を左に 1 ビットシフトして前記出力演算レジスタ Y 1 に格納し、追い出される前記出力演算レジスタ Y 1 の最上位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 1 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、前記演算回数カウンタに 1 を加え、

前記演算回数カウンタが初期値 + 2 の場合には、前記出力演算レジスタ Y 2 を左に 1 ビットシフトして前記出力演算レジスタ Y 2 に格納し、追い出される前記出力演算レジスタ Y 2 の最上位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 2 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、前記演算回数カウンタに 1 を加え、

50

前記演算回数カウンタが初期値 + 3 の場合には、前記出力演算レジスタ Y 3 を左に 1 ビットシフトして前記出力演算レジスタ Y 3 に格納し、追い出される前記出力演算レジスタ Y 3 の最上位ビットを前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y 3 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットする命令であることが好ましい。

【 0 0 1 9 】

前記エミュレーション命令は、

前記出力演算レジスタ Y 0 を左に 1 ビットシフトして前記出力演算レジスタ Y 0 に格納するとき、前記出力演算レジスタ Y 0 の最下位ビットには 0 または前記条件コードレジスタの値を格納し、

10

前記出力演算レジスタ Y 1 を左に 1 ビットシフトして前記出力演算レジスタ Y 1 に格納するとき、前記出力演算レジスタ Y 1 の最下位ビットには前記条件コードレジスタの値を格納し、

前記出力演算レジスタ Y 2 を左に 1 ビットシフトして前記出力演算レジスタ Y 2 に格納するとき、前記出力演算レジスタ Y 2 の最下位ビットには前記条件コードレジスタの値を格納し、

前記出力演算レジスタ Y 3 を左に 1 ビットシフトして前記出力演算レジスタ Y 3 に格納するとき、前記出力演算レジスタ Y 3 の最下位ビットには前記条件コードレジスタの値を格納する命令であることが好ましい。

【 0 0 2 0 】

20

前記演算装置は、前記演算ユニットが次に実行すべき命令のアドレスを格納するプログラムカウンタをさらに備え、

前記エミュレーション命令は、

前記記憶ユニットに含まれるオペランドの上位 $N / 2$ ビットが第 1 の値の場合には、前記オペランドの次アドレスに、前記入力演算レジスタ X の下位 $N / 2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N / 2$ ビットが第 2 の値の場合には、前記オペランドの下位 $N / 2$ ビットと前記入力演算レジスタ X の下位 $N / 2$ ビットとの論理和を求め、

当該論理和の結果が $A 1 1 - 1$ のとき、前記オペランドの次アドレスに、前記入力演算レジスタ X の下位 $N / 2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

30

前記論理和の結果が $A 1 1 - 1$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N / 2$ ビットが第 3 の値の場合には、前記オペランドの下位 $N / 2$ ビットと前記入力演算レジスタ X の下位 $N / 2$ ビットとの論理積を求め、

当該論理積の結果が $A 1 1 - 0$ のとき、前記オペランドの次アドレスに、前記入力演算レジスタ X の下位 $N / 2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記論理積の結果が $A 1 1 - 0$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

40

前記オペランドの上位 $N / 2$ ビットが第 4 の値の場合には、前記オペランドの下位 $N / 2$ ビットと前記入力演算レジスタ X の下位 $N / 2$ ビットとの排他論理和を求め、

当該排他論理和の結果が $A 1 1 - 0$ のとき、前記オペランドの次アドレスに、前記入力演算レジスタ X の下位 $N / 2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記排他論理和の結果が $A 1 1 - 0$ のとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加える命令であることが好ましい。

【 0 0 2 1 】

50

前記演算装置は、前記演算ユニットが次に実行すべき命令のアドレスを格納するプログラムカウンタをさらに備え、

前記エミュレーション命令は、

前記記憶ユニットに含まれるオペランドの上位 $N/2$ ビットが第 1 の値の場合には、前記オペランドの次アドレスに、前記入力演算レジスタ X の上位 $N/2$ ビットの値を Q 倍した値を加え、新たなアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N/2$ ビットが第 2 の値の場合には、前記オペランドの下位 $N/2$ ビットと前記入力演算レジスタ X の上位 $N/2$ ビットとの論理和を求め、

当該論理和の結果が $A11-1$ のとき、前記オペランドの次アドレスに Q を加えたアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記論理和の結果が $A11-1$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N/2$ ビットが第 3 の値の場合には、前記オペランドの下位 $N/2$ ビットと前記入力演算レジスタ X の上位 $N/2$ ビットとの論理積を求め、

当該論理積の結果が $A11-0$ のとき、前記オペランドの次アドレスに Q を加えたアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記論理積の結果が $A11-0$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記オペランドの上位 $N/2$ ビットが第 4 の値の場合には、前記オペランドの下位 $N/2$ ビットと前記入力演算レジスタ X の上位 $N/2$ ビットとの排他論理和を求め、

当該排他論理和の結果が $A11-0$ のとき、前記オペランドの次アドレスに Q を加えたアドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加え、

前記排他論理和の結果が $A11-0$ でないとき、前記オペランドの次アドレスから取得した Q バイトの値の一部または全部を前記プログラムカウンタに加える命令であることが好ましい。

【0022】

前記演算装置は、メモリからのバースト転送が開始すると、当該バースト転送開始後、前記メモリから到着するデータブロックをカウントし、当該カウント数を格納するレディカウンタと、

4 つの条件コードフラグである、ネガティブ、ゼロ、キャリーおよびオーバフローを格納する条件コードレジスタと

をさらに備え、

前記エミュレーション命令は、

前記記憶ユニットに含まれる N ビットのオフセット値を用いて、前記メモリから複数の N ビットデータを取得し、

前記レディカウンタが初期値かつメモリバス上に前記メモリから到着した有効なデータがあることを示す信号であるデータレディ信号がオンの場合には、前記出力演算レジスタ Y_0 と前記メモリバス上の N ビットデータ D_0 とを加算して前記出力演算レジスタ Y_0 に格納し、桁上げ信号を前記条件コードレジスタに格納し、前記出力演算レジスタ Y_0 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、

前記レディカウンタが初期値 + 1 かつ前記データレディ信号がオンの場合には、前記出力演算レジスタ Y_1 と前記メモリバス上の N ビットデータ D_1 と前記条件コードレジスタとを加算して前記出力演算レジスタ Y_1 に格納し、桁上げ信号を前記条件コードレジスタに格納し、前記条件コードレジスタの値が 1 かつ前記出力演算レジスタ Y_1 に格納される値が 0 であるときに前記条件コードレジスタに 1 をセットし、

前記レディカウンタが初期値 + 2 かつ前記データレディ信号がオンの場合には、前記出力演算レジスタ Y_2 と前記メモリバス上の N ビットデータ D_2 と前記条件コードレジスタとを加算して前記出力演算レジスタ Y_2 に格納し、桁上げ信号を前記条件コードレジスタ

10

20

30

40

50

に格納し、前記条件コードレジスタの値が1かつ前記出力演算レジスタY2に格納される値が0であるときに前記条件コードレジスタに1をセットし、

前記レディカウンタが初期値+3かつ前記データレディ信号がオンの場合には、前記出力演算レジスタY3と前記メモリバス上のNビットデータD3と前記条件コードレジスタを加算して前記出力演算レジスタY3に格納し、桁上げ信号を前記条件コードレジスタに格納し、前記条件コードレジスタの値が1かつ前記出力演算レジスタY3に格納される値が0であるときに前記条件コードレジスタに1をセットする命令であることが好ましい。

【0023】

前記演算装置は、メモリからのバースト転送が開始すると、当該バースト転送開始後、前記メモリから到着するデータブロックをカウントし、当該カウント数を格納するレディカウンタと、

4つの条件コードフラグである、ネガティブ、ゼロ、キャリーおよびオーバフローを格納する条件コードレジスタと

をさらに備え、

前記エミュレーション命令は、

前記記憶ユニットに含まれるNビットのオフセット値を用いて、前記メモリから複数のNビットデータを取得し、

前記レディカウンタが初期値かつメモリバス上に前記メモリから到着した有効なデータがあることを示す信号であるデータレディ信号がオンの場合には、前記出力演算レジスタY0とメモリバス上のNビットデータD0とを論理演算して前記出力演算レジスタY0に格納し、前記出力演算レジスタY0に格納される値が0であるときに前記条件コードレジスタに1をセットし、

前記レディカウンタが初期値+1かつ前記データレディ信号がオンの場合には、前記出力演算レジスタY1と前記メモリバス上のNビットデータD1とを論理演算して前記出力演算レジスタY1に格納し、前記条件コードレジスタの値が1かつ前記出力演算レジスタY1に格納される値が0であるときに前記条件コードレジスタに1をセットし、

前記レディカウンタが初期値+2かつ前記データレディ信号がオンの場合には、前記出力演算レジスタY2と前記メモリバス上のNビットデータD2とを論理演算して前記出力演算レジスタY2に格納し、前記条件コードレジスタの値が1かつ前記出力演算レジスタY2に格納される値が0であるときに前記条件コードレジスタに1をセットし、

前記レディカウンタが初期値+3かつ前記データレディ信号がオンの場合には、前記出力演算レジスタY3と前記メモリバス上のNビットデータD3とを論理演算して前記出力演算レジスタY3に格納し、前記条件コードレジスタの値が1かつ前記出力演算レジスタY3に格納される値が0であるときに前記条件コードレジスタに1をセットする命令であることが好ましい。

【発明の効果】

【0024】

本発明は、狭ビット幅の演算装置を用いて広ビット幅の演算装置をエミュレート可能なエミュレーション技術を提供することができるという効果を奏する。

【図面の簡単な説明】

【0025】

【図1】本発明の一実施形態に係る演算装置の概略構成を示すブロック図である。

【図2】マルチプレックス命令の概略手順を示すフローチャートである。

【図3】ロードマージLower命令の概略手順を示すフローチャートである。

【図4】ロードマージUpper命令の概略手順を示すフローチャートである。

【図5】多ビット長右1ビットシフト命令の概略手順を示すフローチャートである。

【図6】多ビット長左1ビットシフト命令の概略手順を示すフローチャートである。

【図7】第1のテーブル分岐命令の概略手順を示すフローチャートである。

【図8】第2のテーブル分岐命令の概略手順を示すフローチャートである。

【図9】多ビット長加算命令の概略手順を示すフローチャートである。

【図10】多ビット長論理積命令の概略手順を示すフローチャートである。

【発明を実施するための形態】

【0026】

以下、本発明の実施の形態について、詳細に説明する。

【0027】

(本発明の意義)

従来の8ビットコンピュータ(MC6502やZ80等)や、4ビットコンピュータ(FERICA)では、32ビットのデータを前提とするプログラムを実用的速度でエミュレートすることが極めて困難である。特に、32ビット幅の命令をデコードしたり、32ビット幅のデータを読み書きしたりする際に多くの時間を要する。このため、特に、デコードやデータの読み書きに関して、従来の8ビットコンピュータが備えていない機能を追加し、不要な機能を削除することにより、エミュレーションに最適な8ビットコンピュータを構成しなければならない。

10

【0028】

例えば8ビットを演算の基本単位とする極めて小規模のハードウェアによりCPUを構成する場合、メモリアクセスのためのアドレスバス幅は16ビット程度、データバス幅は8ビット程度とするのが一般的である。

【0029】

32ビットコンピュータのエミュレーションに特化する場合、アドレスバスは少なくとも24ビット(記憶空間では16MB相当)が必要である。単純にアドレスバス幅を増加するとピン数増加によるハードウェア量の増加を招くため、24ビットのアドレスバスと8ビットのデータバスのピンを共有し、24ビットの信号線を共用するのが効果的である。

20

【0030】

同様に、データバスも32ビット幅があれば32ビットコンピュータのエミュレーションを高速に実行することができる。しかし、CPU内部が8ビット構成であるため、データバスの拡張はあまり効果がない。むしろ、8ビットのデータ転送を連続させるほうが、ハードウェアを効率的に使用できる。

【0031】

そこで、メモリを参照する場合に、まず、24ビットにアドレス情報を載せ、引続き8ビットを複数回駆動するメモリバスサイクルを基本とするのが最適である。本発明は、このようなバスサイクルに適合し、かつ、エミュレーションに特化した後述の命令を備えるハードウェア構成方式とするものである。

30

【0032】

当然のことながら、本発明は、上述したような、8ビット構成のコンピュータを用いて、32ビットのデータを前提とするプログラムをエミュレートすることに好適なものに限られるものではない。本発明は、要は、狭ビット幅の演算装置を用いて広ビット幅の演算装置をエミュレート可能なエミュレーション技術に好適なものである。

【0033】

なお、以下では、8ビット構成のコンピュータを用いて、32ビットのデータを前提とするプログラムをエミュレートする実施形態を例として説明する。

40

【0034】

(演算装置の構成)

図1は、本発明の一実施形態に係る演算装置の概略構成を示すブロック図である。図1に示すように、本発明の一実施形態に係る演算装置100は、演算ユニット10と、作業レジスタ12と、入力演算レジスタ群13と、出力演算レジスタ群14と、プログラムカウンタ15と、演算回数カウンタ16と、条件コードレジスタ17と、レディカウンタ18と、エミュレーション命令記憶ユニット(記憶ユニット)19と、メモリバスアドレスレジスタ20と、メモリバスデータレジスタ21とを備えている。

【0035】

50

演算装置 100 は、メモリバス（図示省略）を介してメモリ（図示省略）に接続されており、メモリバスを用いて、メモリとの間におけるデータ転送を実行する。メモリ内のアドレスを指定するアドレス信号を搬送するアドレスバスと、メモリから読み出されたデータを搬送するデータバスとは、メモリバスを共用する。具体的には、アドレスバスとデータバスとは時分割でメモリバスを共用する。

【0036】

なお、演算装置 100 は、さらに、レジスタ番号指定レジスタ 11 を備えていてもよい。この場合、演算装置 100 は、エミュレーション命令記憶ユニット 19 から読み出された命令に含まれる、後述のオペランドの値及びオフセット値等を、一旦、レジスタ番号指定レジスタ 11 に格納し、レジスタ番号指定レジスタ 11 から、それらオペランドの値及びオフセット値等を読み出しても構わない。

10

【0037】

入力演算レジスタ群 13 は、演算レジスタ X0、演算レジスタ X1、演算レジスタ X2 及び演算レジスタ X3 を含んでおり、出力演算レジスタ群 14 は、演算レジスタ Y0、演算レジスタ Y1、演算レジスタ Y2 及び演算レジスタ Y3 を含んでいる。

【0038】

演算ユニット 10 は、エミュレーション命令記憶ユニット 19 に記憶されている複数のエミュレーション命令の各々を読み出し、エミュレーションを実現する。

【0039】

作業レジスタ 12 は、演算ユニット 10 の演算結果を一時的に格納するレジスタである。

20

【0040】

入力演算レジスタ群 13 は、演算ユニット 10 に入力されるデータを格納するレジスタ群であり、出力演算レジスタ群 14 は、演算ユニット 10 の演算結果を格納するレジスタ群である。

【0041】

プログラムカウンタ 15 は、演算ユニット 10 が次に実行すべき命令のアドレスを格納するレジスタであり、演算回数カウンタ 16 は、メモリバスを用いたメモリからのデータ転送を行なうことなく、演算ユニット 10 が演算する回数をカウントし、当該カウント数を格納するレジスタである。

30

【0042】

条件コードレジスタ 17 は、4つの条件コードフラグ（ネガティブ、ゼロ、キャリー、オーバフロー）を格納するレジスタである。

【0043】

レディカウンタ 18 は、メモリからのバースト転送が開始すると、バースト転送開始後、メモリから到着するデータのかたまり（ブロック）をカウントし、当該カウント数を格納するレジスタである。すなわち、レディカウンタ 18 は、バースト転送開始から何番目に到着したデータであるかを示すカウンタである。メモリからのバースト転送は、バースト転送信号 23 を用いて、制御される。

【0044】

メモリバスアドレスレジスタ 20 は、メモリバスを用いて、演算装置 100 とメモリとの間において搬送されるアドレス信号を格納するレジスタである。メモリバスデータレジスタ 21 は、メモリバスを用いて、演算装置 100 とメモリとの間において搬送されるデータを格納するレジスタである。

40

【0045】

なお、演算装置 100 は、上述したバースト転送信号 23 に加え、以下の信号も、演算ユニット 10 の演算に用いている。

【0046】

識別信号 22 は、メモリバス上で、アドレス信号が搬送されているか、または、データが搬送されているか、を識別する信号である。上述したように、メモリバスは、アドレス

50

信号を搬送するアドレスバスと、データを搬送するデータバスと、に共用されている。演算装置 100 は、この識別信号 22 を用いて、メモリバス上で、アドレス信号が搬送されているか、または、データが搬送されているか、を認識することができる。

【0047】

データレディ信号 24 は、メモリバス上にメモリから到着した有効なデータがあることを示す信号である。

【0048】

桁上げ信号 25 は、演算ユニット 10 の演算の結果、桁上げが発生した場合、その桁上げを表わす信号である。

【0049】

(エミュレーション命令)

エミュレーション命令記憶ユニット 19 は、以下の複数のエミュレーション命令を記憶している。

【0050】

(マルチプレックス命令)

エミュレーション命令記憶ユニット 19 には 8 ビットオペランドの値 (S) が書き込まれている。演算ユニット 10 は、エミュレーション命令記憶ユニット 19 からマルチプレックス命令を読み出し、そのマルチプレックス命令を実行する。図 2 は、マルチプレックス命令の概略手順を示すフローチャートである。

【0051】

図 2 に示すように、演算ユニット 10 は、エミュレーション命令記憶ユニット 19 に書き込まれている 8 ビットオペランドの値 S を読み出す (S1)。この読み出したオペランドの値 S を用いて、演算ユニット 10 は、入力演算レジスタ X1 を構成する 8 ビットのうち、下位 (8 - S) ビットを取り出す (S2)。また、演算ユニット 10 は、上述のオペランドの値 S を用いて、入力演算ユニット X0 を構成する 8 ビットのうち、上位 S ビットを取り出す (S3)。

【0052】

演算ユニット 10 は、このように取り出した、入力演算レジスタ X1 の下位 (8 - S) ビットと入力演算レジスタ X0 の上位 S ビットとを連結する (S4)。そして、演算ユニット 10 は、この連結された 8 ビットを作業レジスタ 12 に格納する (S5)。

【0053】

(ロードマージ Lower 命令)

エミュレーション命令記憶ユニット 19 には 8 ビットのオフセット値が書き込まれている。演算ユニット 10 は、エミュレーション命令記憶ユニット 19 からロードマージ Lower 命令を読み出し、そのロードマージ Lower 命令を実行する。図 3 は、ロードマージ Lower 命令の概略手順を示すフローチャートである。

【0054】

図 3 に示すように、演算ユニット 10 は、エミュレーション命令記憶ユニット 19 に書き込まれている 8 ビットのオフセット値を読み出す (S11)。この読み出したオフセット値を用いて、演算ユニット 10 は、メモリから 1 バイトデータ (D) を取得する。そして、演算ユニット 10 は、D の上位 4 ビットを U とする (S12)。

【0055】

さらに、演算ユニット 10 は、D の下位 4 ビットと作業レジスタ 12 の下位 4 ビットの反転値との論理積を L1 (S13)、および、作業レジスタ 12 の上位 4 ビットと作業レジスタ 12 の下位 4 ビットとの論理積を L2 とする (S14)。

【0056】

そして、演算ユニット 10 は、U と、L1 と L2 との論理和とを連結し (S15)、この連結された 8 ビットを作業レジスタ 12 に格納する (S16)。

【0057】

(ロードマージ Upper 命令)

10

20

30

40

50

エミュレーション命令記憶ユニット19には8ビットのオフセット値が書き込まれている。演算ユニット10は、エミュレーション命令記憶ユニット19からロードマージUpper命令を読み出し、そのロードマージUpper命令を実行する。図4は、ロードマージUpper命令の概略手順を示すフローチャートである。

【0058】

図4に示すように、演算ユニット10は、エミュレーション命令記憶ユニット19に書き込まれている8ビットのオフセット値を読み出す(S21)。この読み出したオフセット値を用いて、演算ユニット10は、メモリから1バイトデータ(D)を取得する。そして、演算ユニット10は、Dの下位4ビットをLとする(S22)。

【0059】

さらに、演算ユニット10は、Dの上位4ビットと作業レジスタ12の下位4ビットの反転値との論理積をU1(S23)、および、作業レジスタ12の上位4ビットと作業レジスタ12の下位4ビットとの論理積をU2とする(S24)。

【0060】

そして、演算ユニット10は、U1とU2との論理和と、Lとを連結し(S25)、この連結された8ビットを作業レジスタ12に格納する(S26)。

【0061】

(多ビット長右1ビットシフト命令)

演算ユニット10は、出力演算レジスタ群14の各演算レジスタY3、Y2、Y1、Y0の内容を右に1ビットシフトする多ビット長右1ビットシフト命令をエミュレーション命令記憶ユニット19から読み出し、その多ビット長右1ビットシフト命令を実行する。図5は、多ビット長右1ビットシフト命令の概略手順を示すフローチャートである。

【0062】

図5に示すように、演算回数カウンタ16が初期値の場合には、演算レジスタY3を右に1ビットシフトして演算レジスタY3に格納する(S31)。

【0063】

このシフトにおいて、演算レジスタY3の最上位ビットには0または条件コードレジスタ17の条件コードCの値を格納する。また、演算レジスタY3から追い出される最下位ビットを条件コードレジスタ17の条件コードCの値として格納する。

【0064】

そして、演算レジスタY3に格納される値が0である場合には、条件コードレジスタ17の条件コードZに1をセットし、演算回数カウンタ16に1を加える。

【0065】

演算回数カウンタ16が初期値+1の場合には、演算レジスタY2を右に1ビットシフトして演算レジスタY2に格納する(S32)。

【0066】

このシフトにおいて、演算レジスタY2の最上位ビットには条件コードレジスタ17の条件コードCの値を格納する。また、演算レジスタY2から追い出される最下位ビットを条件コードレジスタ17の条件コードCの値として格納する。

【0067】

そして、条件コードレジスタ17の条件コードZの値が1であり、かつ、演算レジスタY2に格納される値が0である場合には、条件コードレジスタ17の条件コードZに1をセットし、演算回数カウンタ16に1を加える。

【0068】

演算回数カウンタ16が初期値+2の場合には、演算レジスタY1を右に1ビットシフトして演算レジスタY1に格納する(S33)。

【0069】

このシフトにおいて、演算レジスタY1の最上位ビットには条件コードレジスタ17の条件コードCの値を格納する。また、演算レジスタY1から追い出される最下位ビットを条件コードレジスタ17の条件コードCの値として格納する。

10

20

30

40

50

【 0 0 7 0 】

そして、条件コードレジスタ 17 の条件コード Z の値が 1 であり、かつ、演算レジスタ Y 1 に格納される値が 0 である場合に条件コードレジスタ 17 の条件コード Z に 1 をセットし、演算回数カウンタ 16 に 1 を加える。

【 0 0 7 1 】

演算回数カウンタ 16 が初期値 + 3 の場合には、演算レジスタ Y 0 を右に 1 ビットシフトして演算レジスタ Y 0 に格納する (S 3 4) 。

【 0 0 7 2 】

このシフトにおいて、演算レジスタ Y 0 の最上位ビットには条件コードレジスタ 17 の条件コード C の値を格納する。また、演算レジスタ Y 0 から追い出される最下位ビットを条件コードレジスタ 17 の条件コード C の値として格納する。

10

【 0 0 7 3 】

そして、条件コードレジスタ 17 の条件コード Z の値が 1 であり、かつ、演算レジスタ Y 0 に格納される値が 0 である場合には、条件コードレジスタ 17 の条件コード Z に 1 をセットする。

【 0 0 7 4 】

(多ビット長左 1 ビットシフト命令)

演算ユニット 10 は、出力演算レジスタ群 14 の各演算レジスタ Y 3、Y 2、Y 1、Y 0 の内容を左に 1 ビットシフトする多ビット長左 1 ビットシフト命令をエミュレーション命令記憶ユニット 19 から読み出し、その多ビット長左 1 ビットシフト命令を実行する。

20

図 6 は、多ビット長左 1 ビットシフト命令の概略手順を示すフローチャートである。

【 0 0 7 5 】

図 6 に示すように、演算回数カウンタ 16 が初期値の場合には、演算レジスタ Y 0 を左に 1 ビットシフトして演算レジスタ Y 0 に格納する (S 4 1) 。

【 0 0 7 6 】

このシフトにおいて、演算レジスタ Y 0 の最下位ビットには 0 または条件コードレジスタ 17 の条件コード C の値を格納する。また、演算レジスタ Y 0 から追い出される最上位ビットを条件コードレジスタ 17 の条件コード C の値として格納する。

【 0 0 7 7 】

そして、演算レジスタ Y 0 に格納される値が 0 である場合には、条件コードレジスタ 17 の条件コード Z に 1 をセットし、演算回数カウンタ 16 に 1 を加える。

30

【 0 0 7 8 】

演算回数カウンタ 16 が初期値 + 1 の場合には、演算レジスタ Y 1 を左に 1 ビットシフトして演算レジスタ Y 1 に格納する (S 4 2) 。

【 0 0 7 9 】

このシフトにおいて、演算レジスタ Y 1 の最下位ビットには条件コードレジスタ 17 の条件コード C の値を格納する。また、演算レジスタ Y 1 から追い出される最上位ビットを条件コードレジスタ 17 の条件コード C の値として格納する。

【 0 0 8 0 】

そして、条件コードレジスタ 17 の条件コード Z の値が 1 であり、かつ、演算レジスタ Y 1 に格納される値が 0 である場合には、条件コードレジスタ 17 の条件コード Z に 1 をセットし、演算回数カウンタ 16 に 1 を加える。

40

【 0 0 8 1 】

演算回数カウンタ 16 が初期値 + 2 の場合には、演算レジスタ Y 2 を左に 1 ビットシフトして演算レジスタ Y 2 に格納する (S 4 3) 。

【 0 0 8 2 】

このシフトにおいて、演算レジスタ Y 2 の最下位ビットには条件コードレジスタ 17 の条件コード C の値を格納する。また、演算レジスタ Y 2 から追い出される最上位ビットを条件コードレジスタ 17 の条件コード C の値として格納する。

【 0 0 8 3 】

50

そして、条件コードレジスタ 17 の条件コード Z の値が 1 であり、かつ、演算レジスタ Y 2 に格納される値が 0 である場合に条件コードレジスタ 17 の条件コード Z に 1 をセットし、演算回数カウンタ 16 に 1 を加える。

【0084】

演算回数カウンタ 16 が初期値 + 3 の場合には、演算レジスタ Y 3 を左に 1 ビットシフトして演算レジスタ Y 3 に格納する (S 44)。

【0085】

このシフトにおいて、演算レジスタ Y 3 の最下位ビットには条件コードレジスタ 17 の条件コード C の値を格納する。また、演算レジスタ Y 3 から追い出される最上位ビットを条件コードレジスタ 17 の条件コード C の値として格納する。

10

【0086】

そして、条件コードレジスタ 17 の条件コード Z の値が 1 であり、かつ、演算レジスタ Y 3 に格納される値が 0 である場合には、条件コードレジスタ 17 の条件コード Z に 1 をセットする。

【0087】

(第 1 のテーブル分岐命令)

図 7 は、第 1 のテーブル分岐命令の概略手順を示すフローチャートである。図 7 に示すように、エミュレーション命令記憶ユニット 19 には 8 ビットオペランドの値 (M) が書き込まれている。演算ユニット 10 は、エミュレーション命令記憶ユニット 19 から第 1 のテーブル分岐命令を読み出し (S 51)、その第 1 のテーブル分岐命令を実行する。

20

【0088】

この第 1 のテーブル分岐命令は、オペランドの次アドレスに、演算レジスタ X 0、演算レジスタ X 1、演算レジスタ X 2 及び演算レジスタ X 3 のいずれかの下位 4 ビットの値を 1 倍または 2 倍した値を加え、新たなアドレスから取得した 1 バイトまたは 2 バイトの値の一部または全部をプログラムカウンタ 15 に加える命令である。また、4 ビット (16 エントリ) 分のオフセットを全て用意することなく、テーブルサイズを縮小するために、8 ビットオペランドの値 (M) を使用する。M の上位 4 ビットの値が 0 の場合、演算レジスタ X 0、演算レジスタ X 1、演算レジスタ X 2 及び演算レジスタ X 3 のいずれかの下位 4 ビットの値をそのまま使用する (S 52)。ここでは、演算レジスタ X 0 を使用する。

30

【0089】

M の上位 4 ビットの値が 1 の場合、M の下位 4 ビットと演算レジスタ X 0 の下位 4 ビットとの論理和を求め (S 53)、結果が A 11 - 1 の場合は、第 2 番目のオフセットを使用し、結果が A 11 - 1 でない場合は、第 1 番目のオフセットを使用する。

【0090】

ここで、第 1 番目のオフセットは、エミュレーション命令記憶ユニット 19 に書き込まれている 8 ビットオペランドの次アドレスに配置されるメモリ上の値である。すなわち、第 1 のテーブル分岐命令に引き続くデータである。また、第 2 番目のオフセットは、第 1 番目のオフセットの次アドレスに配置されるメモリ上の値である。

【0091】

M の上位 4 ビットの値が 2 の場合、M の下位 4 ビットと演算レジスタ X 0 の下位 4 ビットとの論理積を求め (S 54)、結果が A 11 - 0 の場合は、第 2 番目のオフセットを使用し、結果が A 11 - 0 でない場合は、第 1 番目のオフセットを使用する。

40

【0092】

M の上位 4 ビットの値が 3 の場合、M の下位 4 ビットと演算レジスタ X 0 の下位 4 ビットとの排他論理和を求め (S 55)、結果が A 11 - 0 の場合は、第 2 番目のオフセットを使用し、結果が A 11 - 0 でない場合は、第 1 番目のオフセットを使用する。

【0093】

このようにして、M の上位 4 ビットの値を適切に設定することにより、テーブルエントリ数を 16 から 2 に縮退できる。

50

【 0 0 9 4 】

(第 2 のテーブル分岐命令)

図 8 は、第 2 のテーブル分岐命令の概略手順を示すフローチャートである。図 8 に示すように、エミュレーション命令記憶ユニット 19 には 8 ビットオペランドの値 (M) が書き込まれている。演算ユニット 10 は、エミュレーション命令記憶ユニット 19 から第 2 のテーブル分岐命令を読み出し (S 6 1)、その第 2 のテーブル分岐命令を実行する。

【 0 0 9 5 】

この第 2 のテーブル分岐命令は、オペランドの次アドレスに、演算レジスタ X 0、演算レジスタ X 1、演算レジスタ X 2 及び演算レジスタ X 3 のいずれかの上位 4 ビットの値を 1 倍または 2 倍した値を加え、新たなアドレスから取得した 1 バイトまたは 2 バイトの値の一部または全部をプログラムカウンタ 15 に加える命令である。また、4 ビット (16 エントリ) 分のオフセットを全て用意することなく、テーブルサイズを縮小するために、8 ビットオペランドの値 (M) を使用する。M の上位 4 ビットの値が 0 の場合、演算レジスタ X 0、演算レジスタ X 1、演算レジスタ X 2 及び演算レジスタ X 3 のいずれかの下位 4 ビットの値をそのまま使用する (S 6 2)。ここでは、演算レジスタ X 0 を使用するとする。

10

【 0 0 9 6 】

M の上位 4 ビットの値が 1 の場合、M の下位 4 ビットと演算レジスタ X 0 の上位 4 ビットとの論理和を求め (S 6 3)、結果が A 1 1 - 1 の場合は、第 2 番目のオフセットを使用し、結果が A 1 1 - 1 でない場合は、第 1 番目のオフセットを使用する。

20

【 0 0 9 7 】

M の上位 4 ビットの値が 2 の場合、M の下位 4 ビットと演算レジスタ X 0 の上位 4 ビットとの論理積を求め (S 6 4)、結果が A 1 1 - 0 の場合は、第 2 番目のオフセットを使用し、結果が A 1 1 - 0 でない場合は、第 1 番目のオフセットを使用する。

【 0 0 9 8 】

M の上位 4 ビットの値が 3 の場合、M の下位 4 ビットと演算レジスタ X 0 の上位 4 ビットとの排他論理和を求め (S 6 5)、結果が A 1 1 - 0 の場合は、第 2 番目のオフセットを使用し、結果が A 1 1 - 0 でない場合は、第 1 番目のオフセットを使用する。

【 0 0 9 9 】

このようにして、M の上位 4 ビットの値を適切に設定することにより、テーブルエントリ数を 16 から 2 に縮退できる。

30

【 0 1 0 0 】

(多ビット長加算命令)

演算ユニット 10 は、出力演算レジスタ群 14 の演算レジスタ Y 3、Y 2、Y 1、Y 0 の内容に、オペランドで示されるメモリアドレスの内容 M 3、M 2、M 1、M 0 を加算して得られる 32 ビットの値を演算レジスタ Y 3、Y 2、Y 1、Y 0 に格納するとともに、条件コードレジスタ 17 の条件コード N (ネガティブ)、Z (ゼロ)、V (オーバフロー)、C (キャリー) を更新する多ビット長加算命令をエミュレーション命令記憶ユニット 19 から読み出し、その多ビット長加算命令を実行する。図 9 は、多ビット長加算命令の概略手順を示すフローチャートである。

40

【 0 1 0 1 】

エミュレーション命令記憶ユニット 19 には 8 ビットのオフセット値 (すなわち、オペランドで示されるメモリアドレス) が書き込まれている。

【 0 1 0 2 】

図 9 に示すように、演算ユニット 10 は、エミュレーション命令記憶ユニット 19 に書き込まれている 8 ビットのオフセット値を読み出す (S 7 1)。この読み出したオフセット値を用いて、演算ユニット 10 は、メモリから複数の 8 ビットデータ (D 3、D 2、D 1、D 0) を取得する (S 7 2)。そして、演算ユニット 10 は、演算レジスタ Y 3、Y 2、Y 1、Y 0 の内容に、複数の 8 ビットデータ (D 3、D 2、D 1、D 0) を加算する。

50

【0103】

具体的には、レディカウンタ18が初期値、かつ、データレディ信号24がオンの場合には、演算レジスタY0とD0とを加算して(S73)、演算レジスタY0に格納し、桁上げ信号25を条件コードレジスタ17の条件コードCに格納する。そして、演算レジスタY0に格納される値が0である場合には、条件コードレジスタ17の条件コードレジスタZに1をセットする。

【0104】

なお、レディカウンタ18は、8ビットデータ(D3、D2、D1、D0)のうちのいずれが転送されてきたかを表わすものである。

【0105】

レディカウンタ18が初期値+1、かつ、データレディ信号24がオンの場合には、演算レジスタY1とD1と条件コードレジスタ17の条件コードCとを加算して(S74)、演算レジスタY1に格納し、桁上げ信号25を条件コードレジスタ17に格納する。そして、条件コードレジスタ17の条件コードレジスタZの値が1、かつ、演算レジスタY1に格納される値が0である場合には、条件コードレジスタ17の条件コードレジスタZに1をセットする。

【0106】

レディカウンタが初期値+2、かつ、データレディ信号24がオンの場合には、演算レジスタY2とD2と条件コードCとを加算して(S75)、演算レジスタY2に格納し、桁上げ信号25を条件コードレジスタ17に格納する。そして、条件コードレジスタ17の条件コードレジスタZの値が1、かつ、演算レジスタY2に格納される値が0である場合には、条件コードレジスタ17の条件コードレジスタZに1をセットする。

【0107】

レディカウンタが初期値+3、かつ、データレディ信号24がオンの場合には、演算レジスタY3とD3と条件コードCとを加算して(S76)、演算レジスタY3に格納し、桁上げ信号25を条件コードレジスタ17に格納する。そして、条件コードレジスタZの値が1、かつ、演算レジスタY3に格納される値が0である場合には、条件コードレジスタ17の条件コードレジスタZに1をセットする。

【0108】

(多ビット長論理積命令)

演算ユニット10は、出力演算レジスタ群14の演算レジスタY3、Y2、Y1、Y0の内容に、オペランドで示されるメモリアドレスの内容M3、M2、M1、M0を論理演算して得られる32ビットの値をY3、Y2、Y1、Y0に格納するとともに、条件コードレジスタ17の条件コードN(ネガティブ)、Z(ゼロ)、V(オーバフロー)、C(キャリー)を更新する多ビット長論理積命令をエミュレーション命令記憶ユニット19から読み出し、その多ビット長論理積命令を実行する。図10は、多ビット長論理積命令の概略手順を示すフローチャートである。

【0109】

エミュレーション命令記憶ユニット19には8ビットのオフセット値が書き込まれている。

【0110】

図10に示すように、演算ユニット10は、エミュレーション命令記憶ユニット19に書き込まれている8ビットのオフセット値を読み出す(S81)。この読み出したオフセット値を用いて、演算ユニット10は、メモリから複数の8ビットデータ(D3、D2、D1、D0)を取得する(S82)。そして、演算ユニット10は、演算レジスタY3、Y2、Y1、Y0の内容に、複数の8ビットデータ(D3、D2、D1、D0)を論理演算する。

【0111】

具体的には、レディカウンタ18が初期値、かつ、データレディ信号24がオンの場合には、演算レジスタY0とD0とを論理演算して(S83)、演算レジスタY0に格納す

10

20

30

40

50

る。そして、演算レジスタ Y 0 に格納される値が 0 である場合には、条件コードレジスタ 17 の条件コードレジスタ Z に 1 をセットする。

【0112】

レディカウンタ 18 が初期値 + 1、かつ、データレディ信号 24 がオンの場合には、演算レジスタ Y 1 と D 1 とを論理演算して (S 8 4)、演算レジスタ Y 1 に格納する。そして、条件コードレジスタ 17 の条件コードレジスタ Z の値が 1、かつ、演算レジスタ Y 1 に格納される値が 0 である場合には、条件コードレジスタ 17 の条件コードレジスタ Z に 1 をセットする。

【0113】

レディカウンタ 18 が初期値 + 2、かつ、データレディ信号 24 がオンの場合には、演算レジスタ Y 2 と D 2 とを論理演算して (S 8 5)、演算レジスタ Y 2 に格納する。そして、条件コードレジスタ 17 の条件コードレジスタ Z の値が 1、かつ、演算レジスタ Y 2 に格納される値が 0 である場合には、条件コードレジスタ 17 の条件コードレジスタ Z に 1 をセットする。

【0114】

レディカウンタ 18 が初期値 + 3、かつ、データレディ信号 24 がオンの場合には、演算レジスタ Y 3 と D 3 とを論理演算して (S 8 6)、演算レジスタ Y 3 に格納する。そして、条件コードレジスタ 17 の条件コードレジスタ Z の値が 1、かつ、演算レジスタ Y 3 に格納される値が 0 である場合には、条件コードレジスタ 17 の条件コードレジスタ Z に 1 をセットする。

【0115】

本発明は上述した各実施形態に限定されるものではなく、請求項に示した範囲で種々の変更が可能であり、異なる実施形態にそれぞれ開示された技術的手段を適宜組み合わせ得られる実施形態についても本発明の技術的範囲に含まれる。

【産業上の利用可能性】

【0116】

本発明は、狭ビット幅の演算装置を用いて広ビット幅の演算装置をエミュレート可能なエミュレーション技術に利用することができる。

【符号の説明】

【0117】

- 10 演算ユニット
- 12 作業レジスタ
- 13 入力演算レジスタ群
- 14 出力演算レジスタ群
- 15 プログラムカウンタ
- 16 演算回数カウンタ
- 17 条件コードレジスタ
- 18 レディカウンタ
- 19 エミュレーション命令記憶ユニット (記憶ユニット)
- 100 演算装置

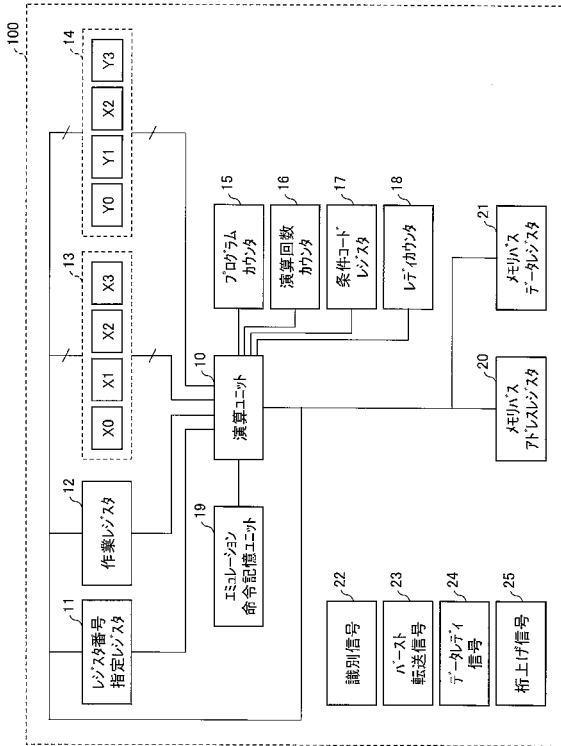
10

20

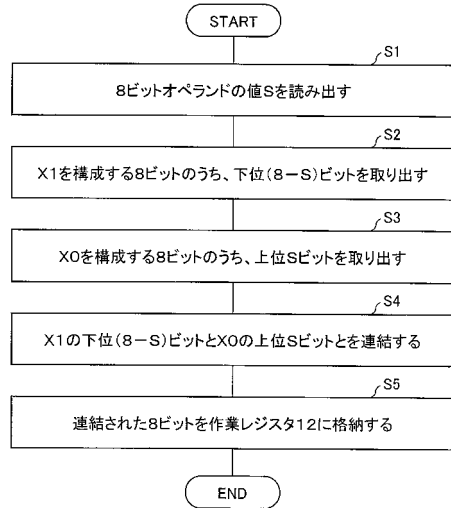
30

40

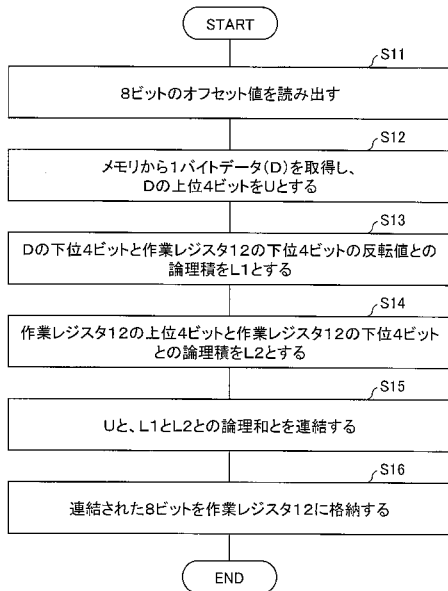
【図1】



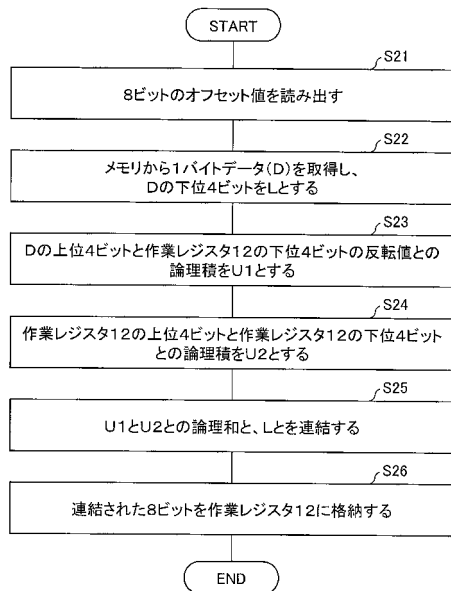
【図2】



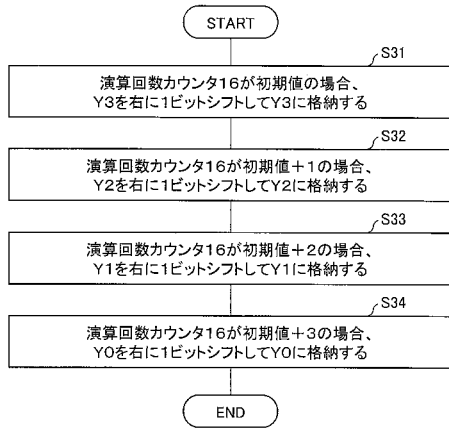
【図3】



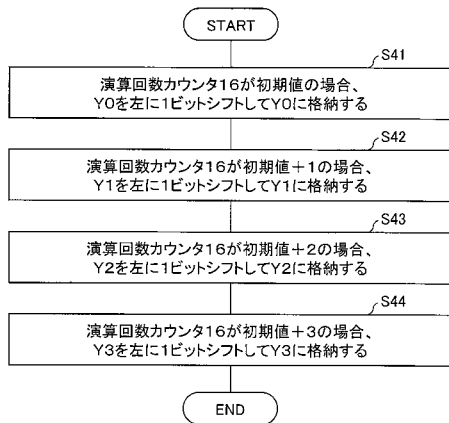
【図4】



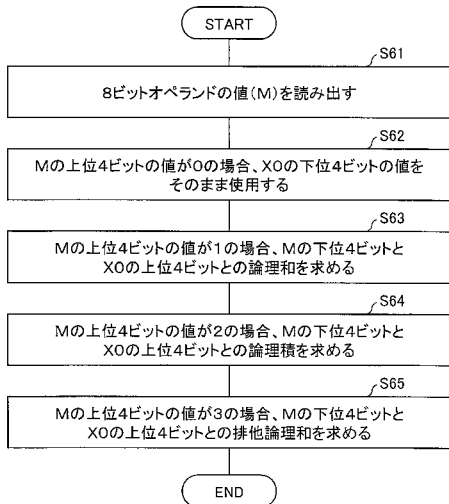
【 図 5 】



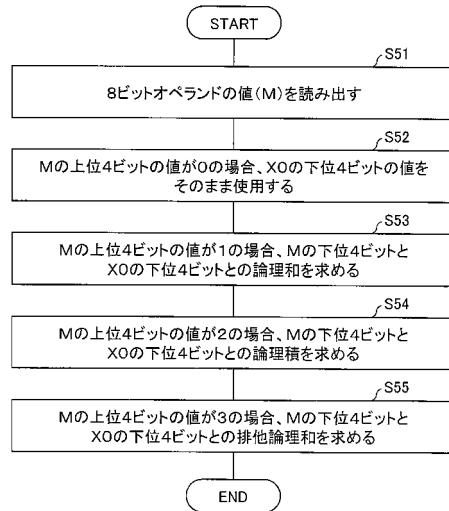
【 図 6 】



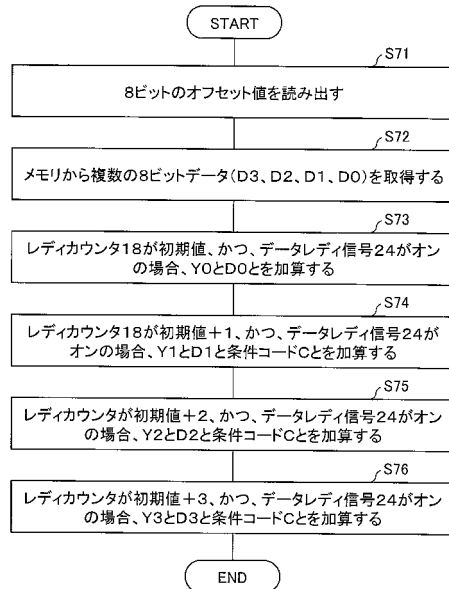
【 図 8 】



【 図 7 】



【 図 9 】



【 図 1 0 】

