

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4483491号
(P4483491)

(45) 発行日 平成22年6月16日(2010.6.16)

(24) 登録日 平成22年4月2日(2010.4.2)

(51) Int. Cl.		F I			
G06F	9/305	(2006.01)	G06F	9/30	340A
G06F	9/32	(2006.01)	G06F	9/32	350B
G06F	7/00	(2006.01)	G06F	7/00	

請求項の数 2 (全 42 頁)

(21) 出願番号	特願2004-257057 (P2004-257057)	(73) 特許権者	000001443 カシオ計算機株式会社 東京都渋谷区本町1丁目6番2号
(22) 出願日	平成16年9月3日(2004.9.3)	(74) 代理人	100090033 弁理士 荒船 博司
(65) 公開番号	特開2005-149474 (P2005-149474A)	(74) 代理人	100093045 弁理士 荒船 良男
(43) 公開日	平成17年6月9日(2005.6.9)	(72) 発明者	仲江 哲一 東京都羽村市栄町3丁目2番1号 カシオ 計算機株式会社 羽村技術センター内
審査請求日	平成19年6月15日(2007.6.15)	(72) 発明者	伊藤 久志 東京都羽村市栄町3丁目2番1号 カシオ 計算機株式会社 羽村技術センター内
(31) 優先権主張番号	特願2003-359453 (P2003-359453)	審査官	三坂 敏夫
(32) 優先日	平成15年10月20日(2003.10.20)		最終頁に続く
(33) 優先権主張国	日本国(JP)		

(54) 【発明の名称】 十進計算装置

(57) 【特許請求の範囲】

【請求項1】

計算命令で計算桁数を設定して多桁の十進計算を行う十進計算装置であって、
 予め定められた桁数単位の桁数よりも多い桁数の数値を記憶する複数の浮動小数点型データレジスタからなり、各レジスタが各桁を十進表記した仮数部と指数部とにより構成される十進浮動小数点データ記憶手段と、

前記複数の浮動小数点型データレジスタの仮数部についての計算開始桁と計算終了桁と計算の種類とを設定している計算命令を記憶する開始終了桁命令記憶手段と、

この開始終了桁命令記憶手段に記憶された計算命令で設定されている計算開始桁から前記桁数単位毎に順次、前記十進浮動小数点データ記憶手段の複数の浮動小数点型データレジスタに記憶されたそれぞれ対応する桁数単位分の数値を、前記開始終了桁命令記憶手段に記憶された計算命令で設定されている計算の種類に従った十進計算で、計算終了桁まで計算し、この計算結果を前記十進浮動小数点データ記憶手段の複数の該当する仮数部のレジスタ部分に前記桁数単位毎に順次書き込む十進計算を行う仮数部十進計算手段と、
 を備えることを特徴とする十進計算装置。

【請求項2】

前記開始終了桁命令記憶手段は、計算開始桁を1桁毎で設定する桁毎開始桁計算命令を記憶する桁毎開始命令記憶手段を備え、

前記仮数部十進計算手段は、前記桁毎開始命令記憶手段により記憶された桁毎開始桁計算命令で設定されている桁から前記桁数単位毎に十進計算する桁毎開始桁対応十進計算手

段を備える

ことを特徴とする請求項 1 に記載の十進計算装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、CPU (Central Processing Unit) 等の演算装置に関する。

【背景技術】

【0002】

CPU等の演算装置を備えた演算システムは従来から広く普及している。普及の一因に、規格化がある。演算システムを構成する演算装置やメモリ等について様々な規格があるが、例えばバイナリ浮動小数点演算を定めた最も一般的な演算規格として、ANSI/IEEE 754 - 1985 標準(以下、単に「IEEE 754」と称する。)があり、このIEEE 754では、バイナリ浮動小数点演算で扱う数値形式(精度)を、単精度(32ビット)、倍精度(64ビット)及び拡張倍精度(96ビット)の3種類と定めている。現在普及している大部分の演算システムはこのIEEE 754に準拠しており、その演算システムを構成する演算装置やメモリも、勿論IEEE 754に準拠した製品である。

10

【0003】

図28は、従来の演算システムの論理的な階層構造の一例を示す図である。同図によれば、従来の演算システムは、演算装置であるCPUを最下位層として、下位層から順に、CPUを統括する基本ソフトウェアであるOS (Operating System)、高級言語で記述された上位層のアプリケーションプログラムを機械語プログラムに変換するコンパイラ、表計算ソフトやワープロソフト等の市販アプリケーション、そしてユーザアプリケーションといった構造となっている。

20

【0004】

IEEE 754に準拠した演算装置の一例としては、IEEE 754 - 1985 標準 (IEEE 754) に適合した掛け算・累積演算を高速に行うデータ処理装置及び方法があるが(特許文献1参照)、IEEE 754 準拠の演算装置が広く普及している好例といえる。

【特許文献1】特開2000 - 10959号公報

【発明の開示】

30

【発明が解決しようとする課題】

【0005】

しかしながら、規格化を推し進めた従来の演算システムは、例えばIEEE 754といった一定の精度(有効桁数)で演算する回路として組み上げられているため、次のような問題が発生していた。

【0006】

(1) 有効桁数の制約

従来の演算装置を具備した演算システムにおいて、演算の精度即ち有効桁数は、演算規格によって定められた精度 (IEEE 754では、3種類) に限られる。即ち、定められた有効桁数以外での演算を行うことができない。従って、必要とされるぴったりの精度で演算を行えない問題があった。

40

【0007】

(2) 2進演算による誤差

また、演算装置は2進演算を行うため、演算システム内部において数値データは2進数で表現される。このため、コンパイラで2進10進変換を行う必要があり、これによる誤差の発生が避けられなかった。例えば、収束計算等において、繰り返し演算をする度に(1)の有効桁数の制約と相まって、誤差が拡大するといった不都合があった。このため、誤差を解消するために、10進計算の精度よりも大きい精度で2進演算を行う必要があった。

【0008】

50

(3) 有効桁数の指定

そもそも、演算装置は演算規格で定められた精度で演算を行うように回路構成されている。そのため、機械語の命令コードは、演算装置の演算精度（有効桁数）で演算を行わせるように作成されるため、命令コード自体においても精度（有効桁数）は固定的であった。具体的には、例えばC言語等の高級言語において、変数を定義する際には必ず精度（例えば、double等）を特定する必要がある、定義された変数でもって演算がなされる。従って、演算途中（プログラムコードの中途）で、変数の有効桁数を任意に指定するといったプログラムを作成することは不可能であった。勿論、高級言語に限らず機械語のプログラムであっても同様である。命令コード自体において精度（有効桁数）が指定できないので、プログラムで対応する必要があり、プログラム作成が煩雑になった。

10

【0009】

上記事情に鑑み、本発明の目的は、計算する際の計算桁数を計算命令により自在に設定できるとともに、小規模な計算手段で効率よく、設定した計算桁数の計算を実行する十進計算装置を提供することである。

【課題を解決するための手段】

【0010】

上記課題を解決するために、請求項1に記載の発明は、計算命令で計算桁数を設定して多桁の十進計算を行う十進計算装置であって、
予め定められた桁数単位の桁数よりも多い桁数の数値を記憶する複数の浮動小数点型データレジスタからなり、各レジスタが各桁を十進表記した仮数部と指数部とにより構成される十進浮動小数点データ記憶手段（例えば、図9のレジスタ部360）と、

20

前記複数の浮動小数点型データレジスタの仮数部についての計算開始桁と計算終了桁と計算の種類とを設定している計算命令を記憶する開始終了桁命令記憶手段（例えば、図9のプログラムROM310）と、

この開始終了桁命令記憶手段に記憶された計算命令で設定されている計算開始桁から前記桁数単位毎に順次、前記十進浮動小数点データ記憶手段の複数の浮動小数点型データレジスタに記憶されたそれぞれ対応する桁数単位分の数値を、前記開始終了桁記憶手段に記憶された計算命令で設定されている計算の種類に従った十進計算で、計算終了桁まで計算し、この計算結果を前記十進浮動小数点データ記憶手段の複数の該当する仮数部のレジスタ部分に前記桁数単位毎に順次書き込む十進計算を行う仮数部十進計算手段（例えば、図9の演算器370）と、

30

を備えることを特徴とする十進計算装置である。

【0011】

請求項2に記載の発明は、請求項1記載の発明において、前記開始終了桁命令記憶手段は、計算開始桁を1桁毎で設定する桁毎開始桁計算命令を記憶する桁毎開始命令記憶手段を備え、

前記仮数部十進計算手段は、前記桁毎開始命令記憶手段により記憶された桁毎開始桁計算命令で設定されている桁から前記桁数単位毎に十進計算する桁毎開始桁対応十進計算手段を備えることを特徴とする十進計算装置である。

40

【発明の効果】

【0019】

請求項1記載の発明によれば、仮数部と指数部を有する複数レジスタ間の演算において、両者の仮数部同士の十進計算を行うときの計算開始桁と計算終了を指定して、計算を行うようにしたので、例えば、十進計算を公式に基づいて正確に行う際に、計算開始桁から計算終了桁まで筆算で行う計算作業をそのまま演算命令として設定記憶でき、見通し良く計算処理の指示を行うことができる。その際、十進計算対象の仮数部と計算対象でない指数部を一体として計算命令で処理できるので、一連の計算命令をシンプルに設定記憶して十進計算を行なわせることができる。

【0020】

50

請求項 2 記載の発明によれば、請求項 1 の発明と同様の効果を奏するとともに、計算開始桁を 1 桁毎で設定できて、この計算開始桁から桁数単位毎に十進計算するようにしたので、桁数単位毎に無駄なく効率のよい十進計算を容易に実行することができる。例えば、桁数単位が 4 桁単位であり、計算開始桁が 4 の倍数の桁位置以外であっても、計算開始桁位置から 4 桁づつ十進計算するので、開始桁未満は処理対象でなく正確に効率よく十進計算が行える。

【発明を実施するための最良の形態】

【 0 0 2 8 】

[第 1 の実施の形態]

以下、図面を参照して本発明を実施するための第 1 の実施の形態を説明する。

10

【 0 0 2 9 】

図 1 は、本第 1 の実施の形態における演算装置の基本構造を示す概略図である。同図によれば、本実施の形態において、演算装置は、主に、演算プログラムを格納するプログラム ROM、演算プログラムを読み出して実行する演算制御回路、演算データを一時的に格納するためのレジスタ部から構成される。

【 0 0 3 0 】

そして、レジスタ部の演算桁数（有効桁数）は可変に構成されており、加減乗除の四則演算や平方根等の基本演算、対数関数や三角関数等の初等関数演算での演算桁数の可変を実現し、また、特殊関数や統計分布関数等の高等関数演算、漸化式等の演算での収束条件の可変を実現している。更に、演算制御回路では、10 進演算を行うことで、2 進 10 進

20

変換による誤差の発生を防止している。

【 0 0 3 1 】

続いて、このような演算装置を備えた演算システム的具体例を説明する。

図 2 (a) は、本実施の形態における演算システム S の概略構成図である。同図によれば、本実施の形態において、演算システム S 1 は、P C (Personal Computer) 1 0 0 と、演算装置 2 0 0 とを備えている。P C 1 0 0 と演算装置 2 0 0 とは U S B (Universal Serial Bus) ケーブル等の通信ケーブル K で接続され、相互にデータ転送可能になっている。

【 0 0 3 2 】

P C 1 0 0 は、C P U (Central Processing Unit) や R O M (Read Only Memory) 、 R A M (Random Access memory) 、キーボードやマウス等の入力装置、ディスプレイ等の表示装置、通信ケーブル K を接続可能な接続端子を有する通信装置を備えて構成される公知の汎用コンピュータで実現され、本演算システム S とユーザとの間のマンマシンインターフェースとして機能する。

30

【 0 0 3 3 】

即ち、P C 1 0 0 は、ユーザによって任意に指定される演算桁数と、演算の種類（四則演算、初等関数、高等関数等）と、演算対象の初期データ値とに基づいて機械語プログラムを生成し、演算装置 2 0 0 に転送する装置であり、演算自体は演算装置 2 0 0 において実行される。また、演算装置 2 0 0 による演算結果を表示装置に表示する。

【 0 0 3 4 】

演算装置 2 0 0 は、P C 1 0 0 から転送されてきた機械語プログラムに従った数値演算を実行する装置であり、特に、機械語プログラムを構成する各命令それぞれを、当該命令において指定された演算桁数で演算する。そして、演算結果を P C 1 0 0 に転送する。

40

【 0 0 3 5 】

ところで、演算装置 2 0 0 で実行される数値演算は、演算桁数の指定の仕方によって、(1) 直接アドレス方式と、(2) 間接アドレス方式と、の 2 つの方式に分類される。

【 0 0 3 6 】

図 2 (b) に示すように、直接アドレス方式は、演算桁数をオペランド部にて直接的に指定する命令方式のことである。演算装置 2 0 0 は、命令のオペランド部で指定された演算桁数で該命令の数値演算を実行する。

50

【 0 0 3 7 】

一方、間接アドレス方式は、演算桁数を格納している格納先を指定（即ち、演算桁数を間接的に指定）する命令方式のことである。演算装置 2 0 0 は、命令によって指定された格納先から演算桁数とする値を読み出し、読み出した演算桁数で該命令の数値演算を実行する。

【 0 0 3 8 】

図 3 は、演算装置 2 0 0 の回路構成を示すブロック図であり、数値演算の実行に係る要部構成を示している。同図によれば、演算装置 2 0 0 は、プログラム ROM 2 1 0 と、プログラムカウンタ 2 1 2 と、ラッチ部 2 1 4 と、命令デコーダ 2 1 6 と、可変パラメータ記憶部 2 2 0 と、セレクタ 2 3 1、2 3 2、2 3 3、2 3 4 と、アドレスカウンタ 2 4 0 と、レジスタ部 2 6 0 と、演算器 2 7 0 と、終了判定回路 2 5 0 とを備えて構成される。

10

【 0 0 3 9 】

プログラム ROM 2 1 0 は、例えば電氣的に書き換え可能なメモリである E E P R O M (Electrical Erasable Programmable ROM) で構成され、P C 1 0 0 から転送されてきた機械語プログラムをプログラム命令 2 1 1 として格納する。このプログラム命令 2 1 1 は、計算の種類と計算桁数とを設定している計算命令からなり、1 以上の任意の計算命令を組み合わせて構成される。プログラム ROM 2 1 0 に格納されたプログラム命令 2 1 1 は、アドレスカウンタ 2 4 0 が示すアドレスの命令が 1 つずつ読み出されてラッチ部 2 1 4 にラッチされる。尚、プログラム ROM 2 1 0 は、電氣的に書き換え可能な不揮発性の R A M 等により構成されてもよい。ラッチ部 2 1 4 は、R A M 等の書き込み自在なメモリで構成され、プログラム ROM から読み出された 1 つの命令を保持する。

20

【 0 0 4 0 】

ここで、1 つの命令は、命令部 1 0 及びオペランド部 2 0 から構成される。命令部 1 0 は、命令コード O P 及び 3 ビット (bit) の拡張命令コード E X T を有する。また、オペランド部 2 0 は、それぞれ 6 ビットの第 1 オペランド 2 1 及び第 2 オペランド 2 2 を有しており、第 1 オペランド 2 1 は、2 ビットのデータ F u 及び 4 ビットのデータ F l から成り、第 2 オペランド 2 2 は、2 ビットのデータ S u 及び 4 ビットのデータ S l から成る。尚、これらの各データの詳細については後述する。

【 0 0 4 1 】

命令デコーダ 2 1 6 は、ラッチ部 2 1 4 にラッチされた命令コード O P を解読（デコード）し、解読した命令を演算器 2 7 0 に実行させるための演算制御信号を演算器 2 7 0 に出力するといった、演算装置 2 0 0 内の各種制御回路を実行させるための各種制御信号を出力する。

30

【 0 0 4 2 】

可変パラメータ記憶部 2 2 0 は、R A M 等で構成され、それぞれ 4 ビットデータを格納可能なレジスタ W、V を備えている。このレジスタ W、V は、間接アドレス方式の場合に演算桁数の格納先となるものであり、演算桁数となる値を変更可能に格納する。

【 0 0 4 3 】

セレクタ 2 3 1 には、ラッチ部 2 1 4 にラッチされたデータ F l 及びレジスタ W に格納されているデータ（以下、「データ w」と称する。）が入力されるとともに、拡張命令コード E X T の 3 ビット目のデータ（以下、「データ E X T [3]」と称する。）が選択制御信号として入力される。そして、セレクタ 2 3 1 は、選択制御信号即ちデータ E X T [3] の値に従って、2 つの入力データの何れか一方を選択して出力する。具体的には、データ E X T [3] の値が「1」の場合にはデータ w を選択して出力し、「0」の場合にはデータ F l を選択して出力する。

40

【 0 0 4 4 】

セレクタ 2 3 2 には、ラッチ部 2 1 4 にラッチされたデータ S l 及びレジスタ V に格納されているデータ（以下、「データ v」と称する。）が入力されるとともに、拡張命令コード E X T の 2 ビット目のデータ（以下、「データ E X T [2]」と称する。）が選択制御信号として入力される。そして、セレクタ 2 3 2 は、選択制御信号即ちデータ E X T [

50

2]の値に従って、2つの入力データの何れか一方を選択して出力する。具体的には、データEXT[2]の値が「1」の場合にはデータvを選択して出力し、「0」の場合にはデータS1を選択して出力する。

【0045】

アドレスカウンタ240は、4ビットのアップカウンタであり、セレクタ232の出力データが初期値としてセットされる。そして、内部クロックに同期してカウントアップし、現在のカウンタ値を出力する。

【0046】

セレクタ233には、セレクタ231の出力データ及びアドレスカウンタ240の出力データ(カウンタ値)が入力されるとともに、拡張命令コードEXTの1ビット目のデータ(以下、「データEXT[1]」と称する。)が選択制御信号として入力される。そして、セレクタ233は、データEXT[1]の値に従って、2つの入力データの何れか一方を選択して出力する。具体的には、データEXT[1]の値が「1」の場合にはセレクタ231の出力データを選択して出力し、「0」の場合にはアドレスカウンタ240の出力データを選択して出力する。

10

【0047】

セレクタ234には、セレクタ232の出力データ及びアドレスカウンタ240の出力データが入力されるとともに、データEXT[1]が選択制御信号として入力される。そして、セレクタ234は、データEXT[1]の値に従って、2つの入力データの何れか一方を選択して出力する。具体的には、データEXT[1]の値が「1」の場合にはセレクタ232の出力データを選択して出力し、「0」の場合にはアドレスカウンタ240の出力データを選択して出力する。

20

【0048】

終了判定回路250は、一致回路252及びORゲート254を有する。

一致回路252は、セレクタ231の出力データ及びアドレスカウンタ240の出力データが入力され、2つの入力データが一致する場合に一致信号「1」を出力し、一致しない場合には不一致信号「0」を出力する。ORゲート254は、一致回路252の出力信号及びデータEXT[1]が入力され、2つの入力信号の論理和の信号を出力する。

【0049】

従って、終了判定回路250は、アドレスカウンタ240の出力データとセレクタ231の出力データとが一致した場合、或いは、データEXT[1]の値が「1」である場合に命令終了信号「1」を出力し、これ以外の場合には命令継続信号「0」を出力する。

30

【0050】

レジスタ部260は、RAM等で構成され、BCDコード化された演算データの各桁の値を所定桁数毎にアドレス順に格納するデータ用レジスタである。また、レジスタ部260は、アドレス指定及びデータ出力をそれぞれ2ポート有する2ポート(Dual Port)方式のレジスタである。

【0051】

第1ポート(Firstポート)では、アドレス端子F u a dから入力される2ビットデータを上位アドレスとし、アドレス端子F l a dから入力される4ビットデータを下位アドレスとする6ビットのアドレスF a dに格納されている16ビットデータを、出力端子F o u tから出力する。アドレス端子F u a dには、ラッチ部214にラッチされたデータF uが入力され、アドレス端子F l a dには、セレクタ233の出力データが入力される。

40

【0052】

第2ポート(Secondポート)では、アドレス端子S u a dから入力される2ビットデータを上位アドレスとし、アドレス端子S l a dから入力される4ビットデータを下位アドレスとする6ビットのアドレスS a dに格納されている16ビットデータを、出力端子S o u tから出力する。アドレス端子S u a dには、ラッチ部214にラッチされたデータS uが入力され、アドレス端子S l a dには、セレクタ234の出力データが入力される

50

。

【0053】

また、レジスタ部260には、入力端子Finから16ビットデータが入力され、入力データは、アドレスFadに書き込まれる。

【0054】

演算器270は、16ビット演算が可能であり、終了判定回路250から命令終了信号「1」が出力されていない間（即ち、命令継続信号「0」が出力されている間）、命令デコード216から入力される演算制御信号に従った演算を行う。即ち、演算器270は、レジスタ部260の出力端子Fout、Soutそれぞれから出力される16ビットデータが入力され、2つの入力データに対する演算を行い、演算結果を16ビットデータで出力する。演算器270の出力データ（演算結果）はレジスタ部260の入力端子Finに入力され、レジスタ部260に書き込まれる。

10

【0055】

図4は、レジスタ部260の詳細構成を示す図である。同図によれば、レジスタ部260は、4本のレジスタX、Y、Z、Aを備えており、各レジスタは、それぞれ16ワード（Word）で構成されている。ここで、1ワードは16ビットであり、演算データはBCDコード化されて格納されているので、1ワードは10進表示で4桁に相当する。また、各レジスタは浮動小数点型のデータを格納し、そのフォーマットは、上位14ワード（第2～第15ワード）が仮数部、下位2ワード（第0～第1ワード）が指数部（符号を含む）となっている。

20

【0056】

レジスタ部260のアドレス指定は、2ビットの上位アドレスでレジスタを指定し、4ビットの下位アドレスでレジスタ内のワードを指定することでなされる。即ち、レジスタ部260のアドレス指定はワード単位で行われ、1ワード分のデータ即ち16ビットデータが出力される。尚、上位アドレスFuad、Suadの値とレジスタとは、図5のデータテーブルに示すように対応付けられている。また、下位アドレスFlad、Sladは、それぞれ、その値がワードを示す。

【0057】

具体的には、第1ポートでは、2ビットの上位アドレスFuad[1:0]でレジスタを指定し、4ビットの下位アドレスFlad[3:0]でレジスタ内のワードを指定して、1ワード即ち16ビットのデータFout[15:0]を出力する。また、第2ポートでは、2ビットの上位アドレスSuad[1:0]でレジスタを指定し、4ビットの下位アドレスSlad[15:0]でレジスタ内のワードを指定して、1ワード即ち16ビットのデータSout[15:0]を出力する。

30

【0058】

更に、データ入力は、1ワード即ち16ビットのデータFin[15:0]が入力され、これは、アドレスFadで指定されたワードに書き込まれる。

【0059】

ところで、本演算システムSで実行される命令は、1命令に対する演算器270の演算回数によって、(1)1ワード命令と、(2)連続ワード命令（桁数可変型演算命令）との2つの命令に分類される。

40

【0060】

1ワード命令では、1命令について各ポートで1つのワードのみを指定し、演算器270は演算を1回のみ行う。従って、1ワード命令では、アドレスFadで指定される1ワードの演算データが被演算数となり、アドレスSadで指定される1ワードの演算データが演算数となる。

【0061】

一方、連続ワード命令では、1命令について各ポートで連続する複数のワードを順に指定し、演算器270は、各ワードについての演算を順に行う。即ち、演算器270は、順に指定されるワードの演算データそれぞれに対する同一の演算を繰り返し行う。このとき

50

、下位アドレス $F1ad$ 、 $S1ad$ は、それぞれ、指定された開始ワードから終了ワードまでを順に指定する。即ち、開始ワードを n 、終了ワードを m とすると、上位アドレス $Fuad$ 、 $Suad$ で指定する各レジスタについて第 n ワードから第 m ワードまで連続する合計 $(m - n + 1)$ ワードが順に指定されて演算が行われ、演算結果が上位アドレス $Fuad$ で指定するレジスタの第 n ワードから第 m ワードに書き込まれる。即ち、10進表現で $(4 \times (m - n + 1))$ 桁での数値演算が行われる。

【0062】

従って、連続ワード命令では、アドレス $Fuad$ で指定されるレジスタの第 n ワードから第 m ワードまでの連続するワードの演算データが被演算数となり、アドレス $Suad$ で指定されるレジスタの第 n ワードから第 m ワードまでの連続するワードの演算データが演算数となる。

10

【0063】

開始ワード及び終了ワードは、命令のオペランド部 20 に含まれるデータ或いは可変パラメータ記憶部 220 に記憶されているデータによって指定される。具体的には、開始ワードは、間接アドレス方式の場合にはデータ v で指定され、直接アドレス方式の場合にはデータ $S1$ で指定される。また、終了ワードは、間接アドレス方式の場合にはデータ w で指定され、直接アドレス方式の場合にはデータ $F1$ で指定される。

【0064】

図6は、連続ワード命令における開始ワード及び終了ワードの指定例を示す図である。同図では、終了ワードを「15」と固定し、開始ワードを可変した場合を示している。同図に示すように、開始ワードを、仮数部の最下位ワードである第2ワードから最上位ワードである第15ワードまで変化させると、演算桁数は、10進表現で56桁、52桁、48桁、・・・、4桁と、1ワード即ち4桁刻みで可変する。つまり、連続ワード命令では、開始ワード及び終了ワードを適当に設定することで、所望の演算桁数での数値演算を実現することができる。

20

【0065】

この連続ワード命令 / 1ワード命令の別、及び、上述した間接 / 直接アドレス方式の別は、命令に含まれる拡張命令コード EXT で指定される。

【0066】

拡張命令コード EXT は、図7に示すように、3ビットで構成され、図中右から順に1ビット目、2ビット目、3ビット目である。そして、1ビット目 ($EXT[1]$) は連続 / 1ワード命令の別を示し、1ワード命令の場合には「1」が設定され、連続ワード命令の場合には「0」が設定される。

30

【0067】

2ビット目 ($EXT[2]$) は、開始ワードの間接 / 直接アドレス方式の別を示し、間接アドレス方式の場合には「1」が設定され、直接アドレス方式の場合には「0」が設定される。但し、1ワード命令の場合には、1ワードの演算データについて1回のみ演算がなされるため、開始ワードのデータがそのまま被演算数となる。

【0068】

また、3ビット目 ($EXT[2]$) は、終了ワードの間接 / 直接アドレス方式の別を示し、間接アドレス方式の場合には「1」が設定され、直接アドレス方式の場合には「0」が設定される。但し、1ワード命令の場合には、終了ワードのデータがそのまま演算数となる。

40

【0069】

図8は、拡張命令コード EXT が取り得る値それぞれについての命令の機能例を示す図である。尚、同図中、命令部のオペコード「ADD」は“加算”を表す命令コードであり、「ADD」に続く3ビットのデータは拡張命令コード EXT を表している。また、オペランド部中の「*」は“指定不要(任意のワード)”を表している。また、ニーモニック及び動作中の「w」はレジスタ W に格納されているデータ w を表し、「v」はレジスタ V に格納されているデータ v を表している。

50

【 0 0 7 0 】

拡張命令コード E X T の各ビットの値は、パターン (A) ~ (H) の合計 8 パターンある。このパターン (A) ~ (H) 毎の演算装置 2 0 0 の具体的な動作例について、以下、順に説明する。

【 0 0 7 1 】

(A) 拡張命令コード E X T = 「 0 0 0 」

この場合、命令は連続ワード命令であり、直接アドレス方式で実行される。例えば、命令「 A D D 0 0 0 X 1 5 Y 4 」を実行する場合、演算装置 2 0 0 は次のように動作する。

【 0 0 7 2 】

まず、ラッチ部 2 1 4 には、 F u = 「 0 0 (X) 」、 F l = 「 1 1 1 1 (1 5) 」、 S u = 「 0 1 (Y) 」、 S l = 「 0 1 0 0 (4) 」、 O P = 「 A D D 」、 E X T = 「 0 0 0 」がラッチされる。

【 0 0 7 3 】

そして、セクタ 2 3 1 では、選択制御信号として E X T [3] = 「 0 」が入力され、 F l = 「 1 1 1 1 」が選択・出力される。また、セクタ 2 3 2 では、選択制御信号として E X T [2] = 「 0 」が入力され、 S l = 「 0 1 0 0 」が選択・出力される。次いで、アドレスカウンタ 2 4 0 には、セクタ 2 3 2 の出力データ即ち S l = 「 0 1 0 0 」がセットされる。続いて、セクタ 2 3 3、2 3 4 では、ともに選択制御信号として E X T [1] = 「 0 」が入力され、アドレスカウンタ 2 4 0 の出力データ即ち S l = 「 0 1 0 0 」が選択・出力される。

【 0 0 7 4 】

従って、レジスタ部 2 6 0 のアドレス端子 F u a d には F u = 「 0 0 」が入力され、アドレス端子 F l a d にはセクタ 2 3 3 の出力データ即ち「 0 1 0 0 」が入力されて、レジスタ X の第 4 ワードに格納されている演算データが出力端子 F o u t から出力される。また、アドレス端子 S u a d には S u = 「 0 1 」が入力され、アドレス端子 S l a d にはセクタ 2 3 3 の出力データ即ち「 0 1 0 0 」が入力されて、レジスタ Y の第 4 ワードに格納されている演算データが出力端子 S o u t から出力される。

【 0 0 7 5 】

一方、命令デコーダ 2 1 6 からは加算を指示する演算制御信号が演算器 2 7 0 に対して出力され、演算器 2 7 0 では、レジスタ部 2 6 0 から出力された 2 つの演算データが加算される。そして、加算結果がレジスタ部 2 6 0 の入力端子 F i n に入力され、レジスタ X の第 4 ワードに書き込まれる (動作 : X 4 + Y 4 X 4) 。

【 0 0 7 6 】

また、終了判定回路 2 5 0 では、セクタ 2 3 1 の出力データ (S u = 「 1 1 1 1 」) とアドレスカウンタ 2 4 0 の出力データ (S l = 「 0 1 0 0 」) とが一致しないため、一致回路 2 5 2 から不一致信号「 0 」が出力されるとともに、 E X T [1] = 「 0 」であるため、命令継続信号「 0 」が出力される。

【 0 0 7 7 】

続いて、アドレスカウンタ 2 4 0 がカウントアップし、セクタ 2 3 3、2 3 4 からは、ともに「 0 1 0 1 (5) 」が出力される。従って、レジスタ部 2 6 0 のアドレス端子 F l a d、S l a d にはともに「 0 1 0 1 」が入力されて、レジスタ X の第 5 ワードに格納されている演算データが出力端子 F o u t から出力され、レジスタ Y の第 5 ワードに格納されている演算データが出力端子 S o u t から出力される。そして、これらの出力データが演算器 2 7 0 にて加算され、加算結果がレジスタ X の第 5 ワードに書き込まれる (動作 : X 5 + Y 5 X 5) 。一方、終了判定回路 2 5 0 から、命令継続信号「 0 」が出力される。

【 0 0 7 8 】

その後も、演算器 2 7 0 が加算を行う毎にアドレスカウンタ 2 4 0 がカウントアップし、レジスタ部 2 6 0 のアドレス端子 F l a d、S l a d には、「 0 1 1 0 (6) 」、「 0

10

20

30

40

50

111(7)」、・・・が順に入力される。つまり、レジスタ部260の出力端子Foutから、レジスタXの第6ワードに格納されている演算データ、第7ワードに格納されている演算データ、・・・が順に出力されるとともに、出力端子Soutから、レジスタYの第6ワードに格納されている演算データ、第7ワードに格納されている演算データ、・・・が順に出力される。そして、これらの演算データが出力順に演算器270にて加算され、加算結果がXレジスタの第6ワード、第7ワード、・・・と順に書き込まれる。

【0079】

そして、アドレスカウンタ240の出力データ(カウント値)が「1111」になると、一致回路252から一致信号「1」が出力されて、終了判定回路250から命令終了信号「1」が出力される。すると、演算器270は演算を終了し、該命令の実行が終了する。

10

【0080】

このように、命令「ADD 000 X15 Y4」では、レジスタX、Yについて、第4ワードから第15ワードまで12ワード連続して加算が実行され、演算結果がレジスタXの第4ワードから第15ワードに順に書き込まれる(動作: $X_4 \sim 15 + Y_4 \sim 15$ $X_4 \sim 15$)。即ち、有効桁数が10進表現で48(=4×12ワード)桁の数値演算が実行される。

【0081】

(B) 拡張命令コードEXT = 「010」

この場合、命令は連続ワード命令であり、間接アドレス方式で実行される。例えば、命令「ADD 010 X15 Y*」を実行する場合、演算装置200は次のように動作する。

20

【0082】

まず、ラッチ部214には、Fu = 「00」、Fl = 「1111」、Su = 「01」、データSl = 「*」、OP = 「ADD」、EXT = 「010」がラッチされる。

【0083】

そして、セレクトア231では、選択制御信号として「0」が入力され、Fl = 「1111」が選択・出力される。また、セレクトア232では、選択制御信号として「1」が入力され、データvが選択・出力されて、アドレスカウンタ240にはデータvがセットされる。次いで、セレクトア233、234では、ともに選択制御信号として「0」が入力され、データvが選択・出力される。

30

【0084】

従って、レジスタ部260のアドレス端子Foadには「00」が入力され、アドレス端子Fladにはデータvが入力されて、レジスタXの第vワードに格納されている演算データが出力端子Foutから出力される。また、アドレス端子Suadには「01」が入力され、アドレス端子Sladにはデータvが入力されて、レジスタYの第vワードに格納されている演算データが出力端子Soutから出力される。そして、レジスタ部260から出力された2つの演算データが演算器270にて加算され、加算結果がレジスタXの第vワードに書き込まれる(動作: $X_v + Y_v$ X_v)。

【0085】

40

その後、終了判定回路250から命令終了信号「1」が出力されない間、アドレスカウンタ240が演算器270による演算毎にカウントアップして、レジスタ部260の出力端子Foutから、レジスタXの第(v+1)ワードに格納されている演算データ、第(v+2)ワードに格納されている演算データ、・・・が順に出力されるとともに、出力端子Soutから、レジスタYの第(v+1)ワードに格納されている演算データ、第(v+2)ワードに格納されている演算データ、・・・が順に出力される。そして、これらの演算データが出力順に演算器270にて加算され、演算結果がレジスタXの第(v+1)ワード、第(v+2)ワード・・・に順に書き込まれる。

【0086】

そして、アドレスカウンタ240の出力データが「1111」になると、一致回路25

50

2 から一致信号「1」が出力され、終了判定回路250から命令終了信号「1」が出力されて、該命令の実行が終了する。

【0087】

このように、命令「ADD 010 X15 Y*」では、レジスタX、Yについて、第vワードから第15ワードまで連続して加算が実行され、演算結果がレジスタXの第vワードから第15ワードまで順に書き込まれる（動作： $X_{v \sim 15} + Y_{v \sim 15} \rightarrow X_{v \sim 15}$ ）。

【0088】

またこの場合、開始ワードはデータvで指定される。データvの値は変更可能であるので、所望の演算桁数に応じた値を格納することで、図6を参照して説明したように、任意の演算桁数での数値演算を演算装置200に行わせることができる。

10

【0089】

(C) 拡張命令コードEXT = 「100」

この場合、命令は連続ワード命令であり、間接アドレス方式で実行される。例えば、命令「ADD 100 X* Y4」を実行する場合、演算装置200は次のように動作する。

【0090】

先ず、ラッチ部214には、Fu = 「00」、Fl = 「*」、Su = 「01」、Sl = 「0100」、OP「ADD」、EXT = 「100」がラッチされる。

【0091】

そして、セクタ231では、選択制御信号として「1」が入力され、データwが選択・出力される。また、セクタ232では、選択制御信号として「0」が入力され、Sl = 「0100」が選択・出力されて、アドレスカウンタ240にはSl = 「0100」がセットされる。次いで、セクタ233、234では、ともに選択制御信号として「0」が入力され、「0100」が選択・出力される。

20

【0092】

従って、レジスタ部260のアドレス端子Fua dには「00」が入力され、アドレス端子Fl a dには「0100」が入力されて、レジスタXの第4ワードに格納されている演算データが出力端子Foutから出力される。また、アドレス端子Sua dには「01」が入力され、アドレス端子Sl a dには「0100」が入力されて、レジスタYの第4ワードに格納されている演算データが出力端子Soutから出力される。そして、レジスタ部260から出力された2つの演算データが演算器270にて加算され、加算結果がレジスタXの第4ワードに書き込まれる（動作： $X_4 + Y_4 \rightarrow X_4$ ）。

30

【0093】

その後、終了判定回路250から命令終了信号「1」が出力されない間、演算器270はレジスタ部260から出力される演算データに対する演算（加算）を繰り返し実行し、アドレスカウンタ240は、演算器270による演算毎にカウントアップする。そして、アドレスカウンタ240の出力データがデータwに一致すると、終了判定回路250から命令終了信号「1」が出力されて、該命令の実行が終了する。

【0094】

このように、命令「ADD 100 X* Y4」では、レジスタX、Yについて、第4ワードから第wワードまで連続して加算が実行され、演算結果がレジスタXの第4ワードから第wワードに順に書き込まれる（動作： $X_{4 \sim w} + Y_{4 \sim w} \rightarrow X_{4 \sim w}$ ）。

40

【0095】

またこの場合、終了ワードはデータwで指定される。データwの値は変更可能であるので、所望の演算桁数に応じた値を格納することで、任意の演算桁数での数値演算を演算装置200に行わせることができる。

【0096】

(D) 拡張命令コードEXT = 「110」

この場合、命令は連続ワード命令であり、間接アドレス方式で実行される。例えば、命

50

令「ADD 110 X* Y*」を実行する場合、演算装置200は次のように動作する。

【0097】

まず、ラッチ部214には、Fu = 「00」、Fl = 「*」、Su = 「01」、Sl = 「*」、OP = 「ADD」、EXT = 「110」がラッチされる。

【0098】

そして、セクタ231では、選択制御信号として「1」が入力され、データwが選択・出力される。また、セクタ232では、選択制御信号として「1」が入力され、データvが選択・出力されて、アドレスカウンタ240にはデータvがセットされる。次いで、セクタ233、234では、ともに選択制御信号として「0」が入力され、データvが選択・出力される。

10

【0099】

従って、レジスタ部260のアドレス端子F u a dには「00」が入力され、アドレス端子F l a dにはデータvが入力されて、レジスタXの第vワードに格納されている演算データが出力端子F o u tから出力される。また、アドレス端子S u a dには「01」が入力され、アドレス端子S l a dにはデータvが入力されて、レジスタYの第vワードに格納されている演算データが出力端子S o u tから出力される。そして、レジスタ部260から出力された2つの演算データが演算器270にて加算され、加算結果がレジスタXの第vワードに書き込まれる（動作： $X_v + Y_v \rightarrow X_v$ ）。

【0100】

その後、終了判定回路250から命令終了信号「1」が出力されない間、演算器270はレジスタ部260からの出力データに対する演算（加算）を繰り返し実行し、アドレスカウンタ240は、演算器270による演算毎にカウントアップする。そして、アドレスカウンタ240の出力データがデータwに一致すると、終了判定回路250から命令終了信号「1」が出力されて、該命令の実行が終了する。

20

【0101】

このように、命令「ADD 110 X* Y*」では、レジスタX、Yについて、第vワードから第wワードまで連続して加算が実行され、演算結果がレジスタXの第vワードから第wワードに順に書き込まれる（動作： $X_v \sim w + Y_v \sim w \rightarrow X_v \sim w$ ）。

【0102】

またこの場合、開始ワードはデータvで指定されるとともに、終了ワードはデータwで指定される。データv、wの値は変更可能であるので、所望の演算桁数に応じた値を格納することで、任意の演算桁数での数値演算を演算装置200に行わせることができる。

30

【0103】

(E) 拡張命令コードEXT = 「001」

この場合、命令は1ワード命令であり、直接アドレス方式で実行される。例えば、命令「ADD 001 X15 Y4」を実行する場合、演算装置200は次のように動作する。

【0104】

まず、ラッチ部214には、Fu = 「00」、Fl = 「1111」、S = 「01」、Sl = 「0100」、OP = 「ADD」、EXT = 「001」がラッチされる。

40

【0105】

そして、セクタ231では、選択制御信号として「0」が入力され、「1111」が選択・出力される。また、セクタ232では、選択制御信号として「0」が入力され、「0100」が選択・出力されて、アドレスカウンタ240には、Sl = 「0100」がセットされる。

【0106】

次いで、セクタ233では、選択制御信号として「1」が入力され、セクタ231の出力データ即ちFl = 「1111」が選択・出力される。また、セクタ234では、選択制御信号として「1」が入力され、セクタ232の出力データ即ちSl = 「0100

50

0」が選択・出力される。

【0107】

従って、レジスタ部260のアドレス端子F u a dには「00」が入力され、アドレス端子F l a dには「1111」が入力されて、レジスタXの第15ワードに格納されている演算データが出力端子F o u tから出力される。また、アドレス端子S u a dには「01」が入力され、アドレス端子S l a dには「0100」が入力されて、レジスタYの第4ワードに格納されている演算データが出力端子S o u tから出力される。

【0108】

そして、レジスタ部260から出力された2つの演算データが演算器270にて加算され、加算結果がレジスタXの第15ワードに書き込まれる（動作： $X_{15} + Y_4 \rightarrow X_{15}$ ）。

10

【0109】

一方、終了判定回路250では、セレクタ231の出力データ（S l = 「1111」）と、アドレスカウンタ240の出力データ（「0100」）とは一致しないため、一致回路252から不一致信号「0」が出力されるが、E X T [1] = 「1」であるため、終了判定回路250から命令終了信号「1」が出力され、該命令の実行が終了する。

【0110】

このように、命令「A D D 001 X15 Y4」では、レジスタXの第15ワードの値とレジスタYの第4ワードの値とが加算され、レジスタXの第15ワードに書き込まれる（動作： $X_{15} + Y_4 \rightarrow X_{15}$ ）。

20

【0111】

(F) 拡張命令コードE X T = 「011」

この場合、命令は1ワード命令であり、間接アドレス方式で実行される。例えば、命令「A D D 011 X15 Y*」を実行する場合、演算装置200は次のように動作する。

【0112】

まず、ラッチ部214には、F u = 「00」、F l = 「1111」、S u = 「01」、S l = 「*」、O P = 「A D D」、E X T = 「011」がラッチされる。

【0113】

そして、セレクタ231では、選択制御信号として「0」が入力され、F l = 「1111」が選択・出力される。また、セレクタ232では、選択制御信号として「1」が入力され、データvが選択・出力されて、アドレスカウンタ240にはデータvがセットされる。

30

【0114】

次いで、セレクタ233では、選択制御信号として「1」が入力され、「1111」が選択・出力される。また、セレクタ234では、選択制御信号として「1」が入力され、データvが選択・出力される。

【0115】

従って、レジスタ部260のアドレス端子F u a dには「00」が入力され、アドレス端子F l a dには「1111」が入力されて、レジスタXの第15ワードに格納されている演算データが出力端子F o u tから出力される。また、アドレス端子S u a dには「01」が入力され、アドレス端子S l a dにはデータvが入力されて、レジスタYの第vワードに格納されている演算データが出力端子S o u tから出力される。

40

【0116】

そして、レジスタ部260から出力された2つの演算データが演算器270にて加算され、レジスタXの第15ワードに書き込まれる（動作： $X_{15} + Y_v \rightarrow X_{15}$ ）。また、終了判定回路250から命令終了信号「1」が出力され、該命令の実行が終了する。

【0117】

このように、命令「A D D 011 X15 Y*」では、レジスタXの第15ワードの値とレジスタYの第vワードの値とが加算され、レジスタXの第15ワードに書き込ま

50

れる（動作： $X15 + Yv \rightarrow X15$ ）。この場合、演算数はデータ v で指定される。

【0118】

(G) 拡張命令コードEXT = 「101」

この場合、命令は1ワード命令であり、間接アドレス方式で実行される。例えば、命令「ADD 101 X* Y4」を実行する場合、演算装置200は次のように動作する。

【0119】

まず、ラッチ部214には、Fu = 「00」、Fl = 「*」、Su = 「01」、Sl = 「0100」、OP = 「ADD」、EXT = 「101」がラッチされる。

【0120】

そして、セクタ231では、選択制御信号として「1」が入力され、データ w が選択・出力される。また、セクタ232では、選択制御信号として「0」が入力され、Sl = 「0100」が選択・出力されて、アドレスカウンタ240には「0100」がセットされる。

【0121】

次いで、セクタ233では、選択制御信号として「1」が入力され、データ w が選択・出力される。また、セクタ234では、選択制御信号として「1」が入力され、「0100」が選択・出力される。

【0122】

従って、レジスタ部260のアドレス端子Fua dには「00」が入力され、アドレス端子Sl a dにはデータ w が入力されて、レジスタXの第 w ワードに格納されている演算データが出力端子Sou tから出力される、また、アドレス端子Sua dには「01」が入力され、アドレス端子Sl a dには「0100」が入力されて、レジスタYの第4ワードに格納されている演算データが出力端子Sou tから出力される。

【0123】

そして、レジスタ部260から出力された2つの演算データが演算器270にて加算され、レジスタXの第 w ワードに書き込まれる（動作： $Xw + Y4 \rightarrow Xw$ ）。また、終了判定回路250から命令終了信号「1」が出力され、該命令の実行が終了する。

【0124】

このように、命令「ADD 101 X* Y4」では、レジスタXの第 w ワードの値とレジスタYの第4ワードの値とが加算され、レジスタXの第 w ワードに書き込まれる（動作： $Xw + Y4 \rightarrow Xw$ ）。この場合、被演算数はデータ w で指定される。

【0125】

(H) 拡張命令コードEXT = 「111」

この場合、命令は1ワード命令であり、間接アドレス方式で実行される。例えば、命令「ADD 111 X* Y*」を実行する場合、演算装置200は次のように動作する。

【0126】

まず、ラッチ部214には、Fu = 「00」、Fl = 「*」、Su = 「01」、Sl = 「*」、OP = 「ADD」、EXT = 「111」がラッチされる。

【0127】

そして、セクタ231では、選択制御信号として「1」が入力され、データ w が選択・出力される。また、セクタ232では、選択制御信号として「1」が入力され、データ v が選択・出力されて、アドレスカウンタ240にはデータ v がセットされる。

【0128】

次いで、セクタ233では、選択制御信号として「1」が入力され、データ w が選択・出力される。また、セクタ234では、選択制御信号として「1」が入力され、データ v が選択・出力される。

【0129】

従って、レジスタ部260のアドレス端子Fus dには「00」が入力され、アドレス

10

20

30

40

50

端子 F l a d にはデータ w が入力されて、レジスタ X の第 w ワードに格納されている演算データが出力端子 F o u t から出力される。また、アドレス端子 S u a d には「 0 1 」が入力され、アドレス端子 S l a d にはデータ v が入力されて、レジスタ Y の第 v ワードに格納されている演算データが出力端子 S o u t から出力される。

【 0 1 3 0 】

そして、レジスタ部 2 6 0 から出力された 2 つの演算データが演算器 2 7 0 にて加算され、加算結果がレジスタ X の第 w ワードに書き込まれる（動作： $X w + Y v \rightarrow X w$ ）。また、終了判定回路 2 5 0 から命令終了信号「 1 」が出力され、該命令の実行が終了する。

【 0 1 3 1 】

このように、命令「 A D D 0 0 1 X w Y v 」では、レジスタ X の第 w ワードの値とレジスタ Y の第 v ワードの値とが加算され、レジスタ X の第 w ワードに書き込まれる（動作： $X w + Y v \rightarrow X w$ ）。この場合、被演算数はデータ w で指定され、演算数はデータ v で指定される。

【 0 1 3 2 】

[第 1 の実施の形態の効果]

以上のように、本第 1 の実施の形態における演算装置 2 0 0 によれば、連続ワード命令の場合、1 つの命令で開始ワード及び終了ワードを指定すると、指定された開始ワードから終了ワードまでの連続する複数ワードについての演算が演算器 2 7 0 によって行われる。従って、所望の演算桁数での演算を行うことができる。

【 0 1 3 3 】

また、開始ワード及び終了ワードは、それぞれを命令中で直接的に指定することもできるし、可変パラメータ記憶部 2 2 0 のレジスタ W、V に格納されているデータ w、v によって間接的に指定することもできる。従って、命令単位で演算桁数（有効桁数）を自在に指定できるため、プログラムの中で演算桁数を変更したり、或いは指定するといったことが可能となる。また、演算結果をレジスタ W、V に格納するといったプログラムにより、プログラムの実行途中に演算桁数を可変することが可能となる。この結果、演算桁数に対する柔軟なプログラムが可能であり、高精度な演算を簡単に実現することができる。

【 0 1 3 4 】

更に、レジスタ部 2 6 0 には、B C D コード化されたデータ値が格納されるので、従来のような 2 進 1 0 進変換による誤差は発生しない。

【 0 1 3 5 】

[第 1 の実施の形態の変形例]

尚、本発明の適用は、上述した実施の形態に限定されることなく、本発明の趣旨を逸脱しない範囲で適宜変更可能である。

【 0 1 3 6 】

(1) 可変する桁数の単位

上述した実施の形態では、1 ワード（16 ビット）を単位とし、10 進表現で 4 桁刻みの桁数可変を行うこととしたが、これ以外のビット数、具体的には $4 \times n$ （n は自然数）ビットを単位にしても良い。例えば演算器 2 7 0 が 32 ビットで演算するものであった場合には 2 ワードを単位として良い。尚、n の 4 倍としたのは、レジスタ部 2 6 0 には B C D コード化されたデータが格納されており、B C D コードでは、4 ビットが 10 進表現で 1 桁に相当するためである。この場合、10 進表現で n 桁刻みの桁数可変が可能で演算装置を実現できる。また、上述した実施の形態において、演算結果のうちの不要な桁の値をマスクすることで、1 桁単位の演算桁数の指定を可能としても良い。

【 0 1 3 7 】

(2) 指数部の演算桁数を可変

また、上述した実施の形態では、第 2 ワードから第 15 ワードの何れかを開始ワード或いは終了ワードとし、仮数部の演算桁数を可変としたが、指数部の演算桁数を可変しても良いのは勿論である。

【 0 1 3 8 】

10

20

30

40

50

(3) レジスタのワード数

また、レジスタ部260のワード数を16として説明したが、これに限られないことは勿論である。また、仮数部を14ワード、指数部を2ワードとしたが、この割合は適宜変更して良いのは勿論である。

【0139】

[第2の実施の形態]

次に、第2の実施の形態について説明する。

本第2の実施の形態における演算システムS2は、PC100と演算装置300とがUSBケーブル等の通信ケーブルK2で接続されることにより、相互にデータ転送可能なように構成されている。尚、演算システムS2の構成は、上述した第1の実施の形態における演算システムS1と略同様であるため、第1の実施の形態と同一部分については、詳細説明を省略し、同一の構成要素は、同一の符号を付して説明する。以下、本第2の実施の形態に特徴的な部分を中心に詳細に説明する。

10

【0140】

本第2の実施の形態に特徴的な構成として、演算装置300は、可変パラメータ部320に6ビットデータを収納可能なレジスタIを有し、間接アドレス方式によりレジスタのアドレスを指定する場合に、桁単位でアドレスの指定が可能となっている。これにより、演算に用いる計算桁数(有効桁数)及び計算開始桁を1桁単位で自在に変更可能な演算装置を実現する。

【0141】

図9は、演算装置300の回路構成を示すブロック図であり、数値演算の実行に係る要部構成を示している。同図によれば、演算装置300は、プログラムROM310と、プログラムカウンタ312と、ラッチ部314と、命令デコーダ316と、可変パラメータ記憶部320と、セレクトア331、332、333、334と、アドレスカウンタ340と、レジスタ部360と、演算器370と、終了判定回路350とを備えて構成される。

20

【0142】

プログラムROM310は、プログラムROM210と同様の構成であり、PC100から転送されてきた機械語プログラムであるプログラム命令311を格納し、プログラムカウンタ312が示すアドレスのプログラム命令311を1つずつ読み出して、ラッチ部314に出力する。このプログラム命令311は、計算桁数と計算の種類とを設定している計算命令からなり、1以上の任意の計算命令を組み合わせで構成される。ラッチ部314は、ラッチ部214と同様に命令部40と、オペランド部30とから構成されるが、第2の実施の形態に特徴的な構成として、命令部40は命令コードOP及び4ビットの拡張命令コードEXTを有している。

30

【0143】

図10に、拡張命令コードEXTのデータ構成例を示す。同図によれば、拡張命令コードEXTは、4ビットで構成され、図中右から順に、1ビット目、2ビット目、3ビット目、4ビット目に対応するデータが格納される。具体的に、1ビット目のデータ(EXT[1])には、1ワード/連続命令の別を示すデータが格納され、1ワード命令の場合には「1」が設定され、連続ワード命令の場合には「0」が設定される。

40

【0144】

2ビット目及び3ビット目のデータ(EXT[2]、EXT[3])には、開始ワードの間接アドレス方式(Vレジスタ指定)/間接アドレス方式(Iレジスタ指定)/直接アドレス方式の別を示すデータが格納される。具体的に、Vレジスタにより指定される間接アドレス方式の場合には「01」が設定され、Iレジスタにより指定される間接アドレス方式の場合には「10」が設定され、直接アドレス方式の場合には「00」が設定される。

【0145】

4ビット目のデータ(EXT[4])は、終了ワードの間接/直接アドレス方式の別を示すデータが格納され、間接アドレス方式の場合には「1」が設定され、直接アドレス方

50

式の場合には「0」が設定される。

【0146】

可変パラメータ記憶部320は、RAM等で構成され、それぞれ4ビットデータを格納可能なレジスタW、レジスタVと、6ビットデータを格納可能なレジスタIを有している。レジスタWは、間接アドレス方式の場合に計算終了桁をワード単位、すなわち4桁単位で指定するものであり、4ビットでレジスタを構成する15ワード中の任意のワードを指定する。また、レジスタVは、間接アドレス方式の場合に計算開始桁をワード単位、すなわち4桁単位で指定するものであり、4ビットでレジスタを構成する15ワード中の任意のワードを指定する。レジスタIは、間接アドレス方式の場合に計算開始桁を1桁単位で指定するものであり、上位4ビットでレジスタを構成する15ワード中の任意のワードを指定し、下位2ビットでその指定されたワード中の任意の桁を指定する。

10

【0147】

セクタ331には、ラッチ部314にラッチされたデータF1及びレジスタWに格納されているデータwが入力されるとともに、拡張命令コードEXTの4ビット目のデータが選択制御信号として入力される。そして、セクタ331は、選択制御信号、すなわちEXT[4]の値にしたがって、2つの入力データの何れか一方を選択して出力する。具体的には、データEXT[4]の値が「1」の場合には、データwを選択して出力し、「0」の場合には、データF1を選択して出力する。

【0148】

セクタ332には、ラッチ部314にラッチされたデータS1、レジスタVに格納されているデータv及びレジスタIに格納されているデータ(以下、「データi」と称する。)が入力される。また、セクタ332には、データEXT[2]、データEXT[3]が選択制御信号として入力される。

20

【0149】

そして、セクタ332は、データEXT[2]、データEXT[3]の値にしたがって、3つの入力データの何れか1つを選択して出力する。具体的には、データEXT[2]、データEXT[3]の値が、「01」の場合には、データvを選択して出力し、「10」の場合には、データiを選択して出力し、「00」の場合には、データS1を選択して出力する。

【0150】

アドレスカウンタ340は、第1の実施の形態のアドレスカウンタ240と同様のアップカウンタであり、演算が行われる毎にカウントアップして、現在のカウント値を出力する。

30

【0151】

レジスタ部360は、レジスタ260と同様の構成であり、アドレス指定及びデータ出力をそれぞれ2ポート有する2ポート方式のレジスタである。第1ポート(Firstポート)では、アドレス端子F u a dから入力される2ビットデータを上位アドレスとし、アドレス端子F l a dから入力される4ビットデータを下位アドレスとする6ビットのアドレスF a dに格納されている16ビットデータを出力端子F o u tから出力する。アドレス端子F u a dには、ラッチ部314にラッチされたデータF uが入力され、アドレス端子F l a dには、セクタ333の出力データが入力される。

40

【0152】

第2ポート(Secondポート)では、アドレス端子S u a dから入力される2ビットデータを上位アドレスとし、アドレス端子S l a dから入力される4ビットデータ又は6ビットデータを下位アドレスとする6ビット又は8ビットのアドレスS a dに格納されている16ビットデータを、出力端子S o u tから出力する。アドレス端子S u a dには、ラッチ部314にラッチされたデータS uが入力され、アドレス端子S l a dには、セクタ334の出力データが入力される。

【0153】

つまり、レジスタ部360のアドレス指定は、2ビットの上位アドレスでレジスタを指

50

定し、4ビットの下位アドレスでレジスタ内のワードを指定するか、6ビットの下位アドレスでレジスタ内のワード及び桁を指定することでなされる。そして、レジスタ部360のアドレス指定が、ワード単位又は桁単位で行われると、指定されたワード又は桁を計算開始ワード又は計算開始桁として1ワード分のデータ、すなわち16ビットデータが出力される。

【0154】

図11を参照して、レジスタRnのアドレス指定が桁単位で行われた場合のレジスタ部360のデータ出力方法について説明する。ここで、同図に示すように、レジスタRnは、上位56桁(14ワード)の仮数部と、下位8桁(2ワード)の指数部とからなり、終了桁である第15ワードの最上位桁はブランク桁となっている。従って、仮数部の最大計算桁数は55桁となっている。また、計算開始桁の設定は桁単位で行われ、計算終了桁の設定はワード単位(すなわち、桁数単位)で行われるものとする。更に、レジスタRnは、1ワードすなわち4桁を桁数単位として、データの出力を行う。以下、具体的に説明する。

10

【0155】

まず、同図2段目に示すように、計算桁数が55桁に指定された場合、計算開始桁は仮数部の最下位である8桁目に設定され、計算終了桁は第15ワードに設定される。そして、レジスタ部360は、計算開始桁に設定された8桁目から、桁数単位を4桁として、4桁分すなわち1ワード分(16ビット)のデータを順次出力し、13ワード分のデータを出力した後、終了ワードにおいては3桁分(12ビット)のデータを演算器370に出力する。これにより、演算器370で処理される計算桁数は55桁となる。

20

【0156】

また、同図3段目に示すように、計算桁数が54桁に指定された場合、計算開始桁は9桁目に指定され、計算終了桁は第15ワードに指定される。そして、レジスタ部360は、計算開始桁に設定された9桁目から、桁数単位を4桁として、4桁分すなわち1ワード分(16ビット)のデータを出力し、13ワード分のデータを順次出力した後、終了ワードにおいては2桁分(8ビット)のデータを演算器370に出力する。これにより、演算器370で処理される計算桁数は54桁となる。

【0157】

更に、同図最下段に示すように、計算桁数が1桁に指定された場合、計算開始桁は62桁目に設定され、計算終了桁は第15ワードに指定される。そして、レジスタ部360は、計算開始桁に設定された62桁目から1桁分のデータ(4ビット)のデータを出力する。この場合、演算器370で処理される計算桁数は1桁となる。

30

【0158】

以上のように、計算桁数が桁単位で指定された場合、プログラム命令311に設定された計算開始桁から、予め定められた桁数単位毎に、1ワード分すなわち4桁分のデータを順次出力し、最終ワードにおいて、残りの桁数分のデータを出力することにより、指定された計算桁数で高速に十進計算を行うことができる。

【0159】

次に、具体的な命令を与えられた場合の演算装置300の動作例について説明する。図12は、拡張命令コードEXTが取り得る値それぞれについての命令の機能例を示す図である。尚、同図中、命令部のOPコード「ADD」は「加算」を表す命令コードであり、「ADD」に続く4ビットのデータは拡張命令コードEXTを表している。また、ニーモニック及び動作中の「w」は、レジスタWに格納されているデータwを表し、「v」はレジスタVに格納されているデータvを表し、「i」は、レジスタIに格納されているデータiを表している。

40

【0160】

拡張命令コードEXTの各ビットの値から、命令の機能例はパターン(I)~(T)の合計12パターンある。ここで、本第2の実施の形態における拡張命令コードEXTのEXT[2],[3]と、上述した第1の実施の形態における拡張命令コードEXTのEX

50

T [2] との対応関係から、パターン (I) ~ (T) には、上述したパターン (A) ~ (H) と重複するパターンが含まれる。そこで、重複するパターンについては対応関係を示し、詳細な動作説明を省略する。すなわち、パターン (A) と (I) 、パターン (B) と (J) 、パターン (C) と (L) 、パターン (D) と (M) 、パターン (E) と (O) 、パターン (F) と (P) 、パターン (G) と (R) 、パターン (H) と (S) は同義の機能例である。以下、第 2 の実施の形態に特徴的なパターン (K) , (N) , (Q) , (T) について説明する。

【 0 1 6 1 】

(K) 拡張命令コード EXT = 「 0 1 0 0 」

この場合、命令は連続ワード命令であり、間接アドレス方式で実行される。例えば、命令「 ADD 0 1 0 0 X 1 5 Y * 」を実行する場合、演算装置 3 0 0 は次のように動作する。

【 0 1 6 2 】

まず、ラッチ部 3 1 4 には、Fu = 「 0 0 (X) 」、Fl = 「 1 1 1 1 (1 5) 」、Su = 「 0 1 (Y) 」、Sl = 「 * 」、OP = 「 ADD 」、EXT = 「 0 1 0 0 」がラッチされる。すなわち、計算命令記憶手段であるプログラム ROM 3 1 0 から計算桁数と計算の種類とを設定している計算命令であるプログラム命令 3 1 1 が読み出される。

【 0 1 6 3 】

そして、セクタ 3 3 1 では、選択制御信号として EXT [4] = 「 0 」が入力され、Fl = 「 1 1 1 1 」が選択・出力される。また、セクタ 3 3 2 では、選択制御信号として EXT [2] = 「 0 」、EXT [3] = 「 1 」が入力され、データ i が選択・出力されて、アドレスカウンタ 3 4 0 にはデータ i がセットされる。次いで、セクタ 3 3 3 、3 3 4 では、ともに選択制御信号として EXT [1] = 「 0 」が入力される。

【 0 1 6 4 】

従って、レジスタ部 3 6 0 のアドレス端子 Fu ad には Fu = 「 0 0 」が入力され、アドレス端子 Fl ad にはデータ i が入力され、レジスタ X の i ~ (i + 3) 桁目に格納されている 1 ワード分の演算データが被演算子として出力端子 Fout から出力される。また、アドレス端子 Su ad には「 0 1 」が入力され、アドレス端子 Sl ad にはデータ i が入力され、レジスタ Y の i ~ (i + 3) 桁目に格納されている 1 ワード分の演算データが被演算子として出力端子 Sout から出力される。そして、レジスタ部 3 6 0 から出力された 2 つの演算データが演算器 3 7 0 にて加算され、加算結果がレジスタ X の i ~ (i + 3) 桁目に書き込まれる (動作 : $X_{i \sim (i+3)} + Y_{i \sim (i+3)}$)。

【 0 1 6 5 】

その後、終了判定回路 3 5 0 から命令終了信号「 1 」が出力されない間、アドレスカウンタ 3 4 0 が演算器 3 7 0 による演算毎に 1 ずつカウントアップして、レジスタ部 3 6 0 の出力端子 Fout から、レジスタ X の i ~ (i + 3) 桁目に格納されている演算データ、(i + 4) ~ (i + 7) 桁目に格納されている演算データ、・・・が順に出力されるとともに、出力端子 Sout から、レジスタ Y の i ~ (i + 3) 桁目に格納されている演算データ、(i + 4) ~ (i + 7) 桁目に格納されている演算データ、・・・が順に出力される。そして、これらの演算データが出力順に演算器 3 7 0 にて加算され、演算結果がレジスタ X の i ~ (i + 3) 桁目、(i + 4) ~ (i + 7) 桁目・・・に順に書き込まれる。

【 0 1 6 6 】

そして、アドレスカウンタ 3 4 0 の出力データが「 1 1 1 1 」になると、一致回路 3 5 2 から一致信号「 1 」が出力され、終了判定回路 3 5 0 から命令終了信号「 1 」が出力されて、該命令の実行が終了する。

【 0 1 6 7 】

このように、命令「 ADD 0 1 0 0 X 1 5 Y * 」では、レジスタ X 、 Y について、i 桁目から第 1 5 ワードまで連続して加算が実行され、演算結果がレジスタ X の i 桁目

10

20

30

40

50

から第15ワードまで順に書き込まれる（動作： $X_{i \sim 15} + Y_{i \sim 15} \quad X_{i \sim 15}$ ）。

【0168】

この動作によって、計算命令記憶手段であるプログラムROM310に記憶された計算命令であるプログラム命令311で設定されている計算桁数（例えば、「 i 桁目～15ワード」）のうち、桁数単位毎（例えば、「4桁単位毎」）に順次、多桁記憶手段であるレジスタ X 、 Y に記憶されたそれぞれに対応する桁数単位分の数値を、プログラム命令311で設定されている計算の種類（例えば、「ADD」）に従った十進計算で計算し、この計算結果を多桁記憶手段であるレジスタ X 、 Y に桁数単位毎に順次書き込む機能を実現される。

10

【0169】

またこの場合、計算開始桁はデータ i で指定される。データ i の値は変更可能であるので、所望の計算桁数に応じた値を格納することで、図11を参照して説明したように、任意の計算桁数での数値演算を演算装置300に行わせることができる。

【0170】

(N)拡張命令コードEXT = 「1100」

この場合、命令は連続ワード命令であり、間接アドレス方式で実行される。例えば、命令「ADD 1100 X^* Y^* 」を実行する場合、演算装置300は次のように動作する。

【0171】

まず、ラッチ部314には、 $F_u = 「00」$ 、 $F_l = 「*」$ 、 $S_u = 「01」$ 、 $S_l = 「*」$ 、 $OP = 「ADD」$ 、 $EXT = 「1100」$ がラッチされる。すなわち、計算命令記憶手段であるプログラムROM310から計算桁数と計算の種類とを設定している計算命令であるプログラム命令311が読み出される。

20

【0172】

そして、セクタ331では、選択制御信号として「1」が入力され、データ w が選択・出力される。また、セクタ332では、選択制御信号として「10」が入力され、データ i が選択・出力されて、アドレスカウンタ340にはデータ i がセットされる。次いで、セクタ333、334では、ともに選択制御信号として「0」が入力され、データ i が選択・出力される。

30

【0173】

従って、レジスタ部360のアドレス端子 F_{uad} には「00」が入力され、アドレス端子 F_{lad} にはデータ i が入力されて、レジスタ X の $i \sim (i+3)$ 桁目に格納されている演算データが出力端子 F_{out} から出力される。また、アドレス端子 S_{uad} には「01」が入力され、アドレス端子 S_{lad} にはデータ i が入力されて、レジスタ Y の $i \sim (i+3)$ 桁目に格納されている演算データが出力端子 S_{out} から出力される。そして、レジスタ部360から出力された2つの演算データが演算器370にて加算され、加算結果がレジスタ X の $i \sim (i+3)$ 桁目に書き込まれる（動作： $X_{i \sim (i+3)} + Y_{i \sim (i+3)} \quad X_{i \sim (i+3)}$ ）。

【0174】

その後、終了判定回路350から命令終了信号「1」が出力されない間、演算器370はレジスタ部360からの出力データに対する演算（加算）を繰り返し実行し、アドレスカウンタ340は、演算器370による演算毎に1ずつカウントアップする。そして、アドレスカウンタ340の出力データの上位4ビットがデータ w に一致すると、終了判定回路350から命令終了信号「1」が出力されて、該命令の実行が終了する。

40

【0175】

このように、命令「ADD 1100 X^* Y^* 」では、レジスタ X 、 Y について、 i 桁目から第 w ワードまで連続して加算が実行され、演算結果がレジスタ X の i 桁目から第 w ワードに順に書き込まれる（動作： $X_{i \sim w} + Y_{i \sim w} \quad X_{i \sim w}$ ）。

【0176】

50

この動作によって、計算命令記憶手段であるプログラムROM310に記憶された計算命令であるプログラム命令311で設定されている計算桁数（例えば、「 i 桁目～ w ワード」）のうち、桁数単位毎（例えば、「4桁単位毎」）に順次、多桁記憶手段であるレジスタ X 、 Y に記憶されたそれぞれに対応する桁数単位分の数値を、プログラム命令311で設定されている計算の種類（例えば、「ADD」）に従った十進計算で計算し、この計算結果を多桁記憶手段であるレジスタ X 、 Y に桁数単位毎に順次書き込む機能の実現される。

【0177】

またこの場合、計算開始桁はデータ i で指定されるとともに、計算終了桁はデータ w で指定される。データ i 、 w の値は変更可能であるので、所望の計算桁数に応じた値を格納することで、任意の計算桁数での数値演算を演算装置300に行わせることができる。

10

【0178】

(Q) 拡張命令コードEXT = 「0101」

この場合、命令は1ワード命令であり、間接アドレス方式で実行される。例えば、命令「ADD 011 X15 Y*」を実行する場合、演算装置300は次のように動作する。

【0179】

先ず、ラッチ部314には、 F_u = 「00」、 F_l = 「1111」、 S_u = 「01」、 S_l = 「*」、 OP = 「ADD」、 EXT = 「0101」がラッチされる。すなわち、計算命令記憶手段であるプログラムROM310から計算桁数と計算の種類とを設定している計算命令であるプログラム命令311が読み出される。

20

【0180】

そして、セレクトア331では、選択制御信号として「0」が入力され、 F_l = 「1111」が選択・出力される。また、セレクトア332では、選択制御信号として「10」が入力され、データ i が選択・出力されて、アドレスカウンタ340にはデータ i がセットされる。

【0181】

次いで、セレクトア333では、選択制御信号として「1」が入力され、「1111」が選択・出力される。また、セレクトア334では、選択制御信号として「1」が入力され、データ i が選択・出力される。

30

【0182】

従って、レジスタ部360のアドレス端子 F_{uad} には「00」が入力され、アドレス端子 F_{lad} には「1111」が入力されて、レジスタ X の第15ワードに格納されている演算データが出力端子 F_{out} から出力される。また、アドレス端子 S_{uad} には「01」が入力され、アドレス端子 S_{lad} にはデータ i が入力されて、レジスタ Y の $i \sim (i+3)$ 桁目に格納されている演算データが出力端子 S_{out} から出力される。

【0183】

そして、レジスタ部360から出力された2つの演算データが演算器370にて加算され、レジスタ X の第15ワードに書き込まれる（動作： $X15 + Y_{i \sim (i+3)} \times 15$ ）。また、終了判定回路350から命令終了信号「1」が出力され、該命令の実行が終了する。

40

【0184】

このように、命令「ADD 0101 X15 Y*」では、レジスタ X の第15ワードの各桁の値とレジスタ Y の $i \sim (i+3)$ 桁目の値とが加算され、レジスタ X の第15ワードに書き込まれる（動作： $X15 + Y_{i \sim (i+3)} \times 15$ ）。

【0185】

この動作によって、計算命令記憶手段であるプログラムROM310に記憶された計算命令であるプログラム命令311で設定されている計算桁数（例えば、「4桁」）のうち、桁数単位毎（例えば、「4桁単位毎」）に順次、多桁記憶手段であるレジスタ X 、 Y に記憶されたそれぞれに対応する桁数単位分の数値を、プログラム命令311で設定されて

50

いる計算の種類（例えば、「ADD」）に従った十進計算で計算し、この計算結果を多桁記憶手段であるレジスタX、Yに桁数単位毎に順次書き込む機能の実現される。

【0186】

また、この場合、計算開始桁はデータiで指定される。データiの値は変更可能であるので、所望の計算開始桁に応じた値を格納することで、任意の数値を演算数として数値演算を演算装置300に行わせることができる。

【0187】

(T) 拡張命令コードEXT = 「1101」

この場合、命令は1ワード命令であり、間接アドレス方式で実行される。例えば、命令「ADD 111 X* Y*」を実行する場合、演算装置300は次のように動作する

10

【0188】

まず、ラッチ部314には、Fu = 「00」、Fl = 「*」、Su = 「01」、Sl = 「*」、OP = 「ADD」、EXT = 「1101」がラッチされる。すなわち、計算命令記憶手段であるプログラムROM310から計算桁数と計算の種類とを設定している計算命令であるプログラム命令311が読み出される。

【0189】

そして、セクタ331では、選択制御信号として「1」が入力され、データwが選択・出力される。また、セクタ332では、選択制御信号として「10」が入力され、データiが選択・出力されて、アドレスカウンタ340にはデータiがセットされる。

20

【0190】

次いで、セクタ333では、選択制御信号として「1」が入力され、データwが選択・出力される。また、セクタ334では、選択制御信号として「1」が入力され、データiが選択・出力される。

【0191】

従って、レジスタ部360のアドレス端子Fusdには「00」が入力され、アドレス端子Fladにはデータwが入力されて、レジスタXの第wワードに格納されている演算データが出力端子Foutから出力される。また、アドレス端子Sudには「01」が入力され、アドレス端子Sladにはデータiが入力されて、レジスタYのi ~ (i + 3) 桁目に格納されている演算データが出力端子Soutから出力される。

30

【0192】

そして、レジスタ部360から出力された2つの演算データを被演算子として演算器370にて加算され、加算結果がレジスタXの第wワードに書き込まれる（動作：Xw + Yi ~ (i + 3) Xw）。また、終了判定回路350から命令終了信号「1」が出力され、該命令の実行が終了する。

【0193】

このように、命令「ADD 1101 Xw Yi」では、レジスタXの第wワードの値とレジスタYのi ~ (i + 3) 桁目の値とが加算され、レジスタXの第wワードに書き込まれる（動作：Xw + Yi ~ (i + 3) Xw）。

【0194】

この動作によって、計算命令記憶手段であるプログラムROM310に記憶された計算命令であるプログラム命令311で設定されている計算桁数（例えば、「4桁」）のうち、桁数単位毎（例えば、「4桁単位毎」）に順次、多桁記憶手段であるレジスタX、Yに記憶されたそれぞれに対応する桁数単位分の数値を、プログラム命令311で設定されている計算の種類（例えば、「ADD」）に従った十進計算で計算し、この計算結果を多桁記憶手段であるレジスタX、Yに桁数単位毎に順次書き込む機能の実現される。

40

【0195】

また、この場合、計算終了桁はデータwで指定され、計算開始桁はデータiで指定される。データw、データiの値は変更可能であるので、所望の計算終了桁、計算開始桁に応じた値を格納することで、任意の数値を被演算子とした数値演算を演算装置300に行わ

50

せることができる。

【0196】

次に、上述したパターン(N)を一例として、レジスタI及びレジスタWに数値が設定された場合に行われる動作例についてフローチャートを参照して説明する。図13は、演算装置300により実行される桁指定計算処理を示すフローチャートである。同図によれば、まず、プログラムROM310のプログラムエリアからプログラムカウンタ312により指定された命令が読み出される(ステップST1)。次いで、命令中のIレジスタの値*i*が読み出され(ステップST2)、レジスタIにデータ*i*がセットされる(ステップST3)。また、命令中のWレジスタの値*w*が読み出され(ステップST4)、レジスタWにデータ*w*がセットされる(ステップST5)。

10

【0197】

次いで、*w*が最終ワード、すなわち*w* = 15であるか否かが判別される(ステップST6)。ここで、*w* = 15である場合(ステップST6; YES)、 $w \times 4 + 2$ を計算することにより計算終了桁が算出される(ステップST7)。続いて、Xレジスタの*i* ~ *i* + 3桁目の演算データが読み出されるとともに(ステップST8)、Yレジスタの*i* ~ *i* + 3桁目の演算データが読み出される(ステップST9)。そして、演算器370において、命令に従った計算が行われ、計算結果がXレジスタの*i* ~ *i* + 3桁目に格納される(ステップST10)。

【0198】

続いて、*i* + 4が計算終了桁より小さいか否かが判別され(ステップST11)、*i* + 4が計算終了桁より小さい場合(ステップST11; YES)、終了判定回路350から命令終了信号「0」が出力されて、レジスタIの値*i*に*i* + 4がセットされる(ステップST12)。そして、ステップST8に移行して、上述した処理が繰り返して実行される。一方、*i* + 4が計算終了桁より大きい場合(ステップST11; NO)、終了判定回路350から命令終了信号「1」が出力されて、本桁指定計算処理が終了される。

20

【0199】

また、ステップST6において、*w*が最終ワードでない場合、すなわち*w* = 15でない場合(ステップST6; NO)、 $w \times 4 + 3$ を計算することにより計算終了桁が算出される(ステップST13)。続いて、Xレジスタの*i* ~ *i* + 3桁目の演算データが読み出されるとともに(ステップST14)、Yレジスタの*i* ~ *i* + 3桁目の演算データが読み出される(ステップST15)。そして、演算器370において、命令に従った計算が行われ、計算結果がXレジスタの*i* ~ *i* + 3桁目に格納される(ステップST16)。

30

【0200】

続いて、*i* + 4が計算終了桁より小さいか否かが判別され(ステップST17)、*i* + 4が計算終了桁より小さい場合(ステップST17; YES)、終了判定回路350から命令終了信号「0」が出力されて、レジスタIの値*i*に*i* + 4がセットされる(ステップST18)。そして、ステップST14に移行して、上述した処理が繰り返して実行される。一方、*i* + 4が計算終了桁より大きい場合(ステップST17; NO)、*i* + 3が計算終了桁であるか否かが判別される(ステップST19)。ここで、計算終了桁が*i* + 3である場合(ステップST19; YES)、Xレジスタに格納された計算結果の演算データと計算終了桁が一致するため、終了判定回路350から命令終了信号「1」が出力されて、本桁指定計算処理が終了される。

40

【0201】

一方、計算終了桁が*i* + 3でない場合(ステップST19; NO)、Xレジスタに格納された計算結果の演算データと計算終了桁が一致しないため、Xレジスタに格納された*i* ~ *i* + 3桁目のデータのうち、*i* + 1 ~ *i* + 3桁目のデータがマスクされる(ステップST20)。つまり、レジスタ部360は、指定された*i*桁目から1ワード毎、すなわち4桁毎に演算データを出力するため、計算桁数が4の整数倍とならない場合、終了ワードの演算データに不必要なデータを含むこととなる。そこで、不必要なデータをマスクすることにより、所望する桁数の計算結果を得ることができる。そして、レジスタXの不必要

50

データがマスクされると(ステップST20)、終了判定回路350から命令終了信号「1」が出力されて、本桁指定計算処理が終了される。

【0202】

次に、具体的な計算命令の種類、計算数値、計算桁数が設定された場合に演算装置300に実行される計算方法について説明する。まず、計算命令の種類として「平方根」、計算数値として「3」、計算開始桁として「56桁(i=56)」、計算終了桁として「15ワード(w=15)」が設定された場合について図14~図17を参照して説明する。

【0203】

図14~図17中の左側の算術式は、筆算による平方根の計算方法を説明するための式であり、図14~図17中の右側の計算手順は、演算装置300により実行される平方根の計算方法を説明するための手順である。ここで、平方根を筆算で求める場合、下記式(1)に基づいて、各位の値を求めるものとする。尚、この方法は、一般的に知られているため、詳細な説明は省略する。

10

【数1】

$$(a+b+c+d+\dots)^2 = a^2 + 2ab + b^2 + 2(a+b)c + c^2 + 2(a+b+c)d + d^2 + \dots \quad (1)$$

【0204】

図14左側に示すように、上記式(1)から、数(1)は、「3」を計算数値とした場合、平方して3を超えない値「1」として得る。数(2)は、数(1)と同じ値「1」として得る。また、数(3)は、数(1)の平方値「1²」であり、これを計算数値「3」から減算した減算結果「2」を数(4)として得る。更に、数(2)に数(1)を加算した値「2」を数(5)として得て、次の計算に進む。

20

【0205】

上述した筆算による計算を演算装置300で行う場合を説明する。図14右側に示すように、演算開始時においては、レジスタX = 「3」、レジスタY = 「0」、レジスタZ = 「0」、レジスタA = 「0」が設定されている。次いで、レジスタAには「A + 1 1」(数(1)に対応)が設定され、レジスタYには「Y + 1 1」(数(2)に対応)が設定され、レジスタZには「Y × 1 1」(数(3)に対応)が設定される。続いて、レジスタIにi(計算開始桁) = 62、レジスタWにw(計算終了) = 15が設定され、演算が開始されると、レジスタXの出力データとレジスタZの出力データとが演算され、演算結果がレジスタXに「3 - 1 2」(数(4)に対応)として設定される。更に、レジスタYには「Y + 1 2」(数(5)に対応)が設定される。

30

【0206】

次に、図15左側に示すように、上記式(1)から、(20+x) × xが200以下となる最大のxを求め、x = 7から数(6)を「1.7」として得る。数(7)は、(20+x)から「27」として得る。また、「27」と「7」とを乗算した値「189」を数(8)として得て、200から数(8)を減算し、減算結果「11」を数(9)として得る。更に、数(7)に「7」を加算した値「37」を数(10)として得て、次の計算に進む。

40

【0207】

上述した筆算による計算を演算装置300で行う場合を説明する。図15右側に示すように、レジスタAには「A + 0.7 1.7」(数(6)に対応)が設定され、レジスタYには「Y + 0.7 2.7」(数(7)に対応)が設定され、レジスタZには「Y × 0.7 1.89」(数(8)に対応)が設定される。続いて、レジスタIにi(計算開始桁) = 60、レジスタWにw(計算終了ワード) = 15が設定され、演算が開始されると、レジスタXの出力データとレジスタZの出力データとが演算され、演算結果がレジスタXに「2.00 - 1.89 0.11」(数(9)に対応)として設定される。更に、レジスタYには「Y + 0.7 3.4」(数(10)に対応)が設定される。

50

【0208】

次に、図16左側に示すように、上記式(1)から、 $(340+x) \times x$ が1100以下となる最大の x を求め、 $x=3$ から数(11)を「1.73」として得る。数(12)は、 $(340+x)$ から「343」として得る。また、「343」と「3」とを乗算した値「1029」を数(13)として得て、1100から数(13)を減算して、減算結果「71」を数(14)として得る。更に、数(12)に「3」を加算した値「346」を数(5)として得て、次の計算に進む。

【0209】

上述した筆算による計算を演算装置300で行う場合を説明する。図16右側に示すように、レジスタAには「 $A+0.03 \quad 1.73$ 」(数(11)に対応)が設定され、レジスタYには「 $Y+0.03 \quad 3.43$ 」(数(12)に対応)が設定され、レジスタZには「 $Y \times 0.03 \quad 0.1029$ 」(数(13)に対応)が設定される。続いて、レジスタIに i (計算開始桁)=58, レジスタWに w (計算終了ワード)=15が設定され、演算が開始されると、レジスタXの出力データとレジスタZの出力データとが演算され、演算結果がレジスタXに「 $0.11-0.1029 \quad 0.0071$ 」(数(14)に対応)として設定される。更に、レジスタYには「 $Y+0.03 \quad 3.46$ 」(数(15)に対応)が設定される。

10

【0210】

次に、図17左側に示すように、上記式(1)から、 $(3460+x) \times x$ が7100以下となる最大の x を求め、 $x=2$ から数(16)を「1.732」として得る。数(17)は、 $(3460+x)$ から「3462」として得る。また、「3462」と「2」とを乗算した値「6924」を数(18)として得て、7100から数(18)を減算して、減算結果「176」を数(19)として得る。更に、数(17)に「2」を加算した値「3464」を数(20)として得て、計算を終了する。

20

【0211】

上述した筆算による計算を演算装置300で行う場合を説明する。図17右側に示すように、レジスタAには「 $A+0.002 \quad 1.732$ 」(数(16)に対応)が設定され、レジスタYには「 $Y+0.002 \quad 3.462$ 」(数(17)に対応)が設定され、レジスタZには「 $Y \times 0.002 \quad 0.006924$ 」(数(18)に対応)が設定される。続いて、レジスタIに i (計算開始桁)=56, レジスタWに w (計算終了ワード)=15が設定され、演算が開始されると、レジスタXの出力データと、レジスタZの出力データが演算され、演算結果がレジスタXに「 $0.007100-0.006924 \quad 0.000176$ 」(数(19)に対応)として設定される。更に、レジスタYには「 $Y+0.002 \quad 3.464$ 」(数(20)に対応)が設定される。

30

【0212】

以上のように、計算命令の種類「平方根」、計算数値「3」、計算開始桁「56桁」、計算終了桁「15ワード」が命令として与えられた場合、演算結果として「1.732」が得られる。

【0213】

次に、計算命令の種類として「立方根」、計算数値として「3」、計算開始桁として「56桁($i=56$)」、計算終了桁として「15ワード($w=15$)」が設定された場合の演算方法について図18~図20を参照して説明する。尚、以下の説明において、演算装置300のレジスタ部370は、5個のレジスタX, Y, Z, A, Bを有し、各レジスタを適宜切り替えて使用する例として説明する。

40

【0214】

図18~図20中の左側の算術式は、筆算による立方根の演算方法を説明するための式であり、図18~図20中の右側の計算手順は、演算装置300により実行される立方根の演算方法を説明するための手順である。ここで、立方根を筆算で求める場合、下記式(2)に基づいて、各位の値を求めるものとする。尚、この方法は、一般的に知られているため、詳細な説明は省略する。

50

【数 2】

$$(a + b + c + d + \dots)^3$$

$$= a^3 + 3a^2b + 3ab^2 + b^3 + (3a^2 + 6ab + 3b^2)c + (3a + 3b)c^2 + c^3 + \dots \quad (2)$$

【0215】

図18左側に示すように、上記式(2)から、数(21)は、「3」を計算数値とした場合、3乗して3を超えない値「1」として得る。また、数(21)の立方値「1³」を数(22)として得て、これを計算数値「3」から減算した減算結果「2」を数(23)として得る。また、数(24)は、数(21)の平方値「1²」と「3」とを乗算した値「3」として得る。数(25)は、数(21)と「3」とを乗算した値「3」として得て、次の計算に進む。

10

【0216】

上述した筆算による計算を演算装置300で行う場合を説明する。図18右側に示すように、演算開始時においては、レジスタX = 「3」、レジスタY = 「0」、レジスタZ = 「0」、レジスタW = 「0」、レジスタA = 「0」が設定されている。次いで、レジスタAには「(A + 1) 1」(数(21)に対応)が設定され、レジスタZには「1³ 1」(数(22)に対応)が設定される。続いて、レジスタIにi(計算開始桁) = 62、レジスタWにw(計算終了ワード) = 15が設定され、演算が開始されると、レジスタXの出力データとレジスタZの出力データとが演算され、演算結果がレジスタXに「3 - 1 2」(数(23)に対応)として設定される。更に、レジスタBには「3 × 1 × 1 3」(数(24)に対応)が設定され、レジスタYには「3 × 1 3」(数(25)に対応)が設定される。

20

【0217】

次に、図19左側に示すように、上記式(2)から、300 × x と 30 × x² と x³ とが2000以下になる最大のxを求め、x = 4から数(26)を「14」として得る。数(27)は、300 × 4から「1200」として得る。また、数(28)は、30 × 4²から「480」として得て、数(29)は、4³から「64」として得る。そして、2000から数(27) ~ (29)を減算して、減算結果「256」を数(30)として得る。更に、「3」と数(26)の平方値「196」とを乗算した値「588」を数(31)として得て、「3」と数(26)を乗算した値「42」を数(32)として得て、次の計算に進む。

30

【0218】

上述した筆算による計算を演算装置300で行う場合を説明する。図19右側に示すように、レジスタAには「A + 0.4 1.4」(数(26)に対応)が設定され、レジスタZには「3 × 0.4 1.2」(数(27)に対応)が設定される。次いで、レジスタIにi(計算開始桁) = 59、レジスタWにw(計算終了ワード) = 15が設定され、演算が開始されると、レジスタXの出力データと、レジスタZの出力データとが演算され、演算結果がレジスタXに「2.000 - 1.200 0.800」として設定される。続いて、レジスタZには「3 × 0.4² 0.48」(数(28)に対応)が設定され、レジスタXの出力データと、レジスタZの出力データとが演算され、演算結果がレジスタXに「0.800 - 0.480 0.320」として設定される。

40

【0219】

更に、レジスタZには「0.4³ 0.064」(数(29)に対応)が設定され、レジスタXの出力データと、レジスタZの出力データとが演算され、演算結果がレジスタXに「0.320 - 0.064 0.256」(数(30)に対応)として設定される。また、レジスタBには「3 × 1.4² 5.88」(数(31)に対応)が設定され、レジスタYには「3 × 1.4 4.2」(数(32)に対応)が設定される。

【0220】

次に、図20左側に示すように、上記式(2)から、58800 × x³ と、420 × x²

50

と、 4^3 とが256000以下になる最大の x を求め、 $x = 4$ から数(33)を「144」として得る。数(34)は、 58800×4 から「235200」として得る。また、数(35)は、 420×4^2 から「6720」として得て、数(36)は、 4^3 から「64」として得る。そして、256000から数(34)～(36)を減算して、減算結果「14016」を数(37)として得る。更に、「3」と数(33)の平方値「20736」を乗算した値「62208」を数(38)とし、「3」と数(33)を乗算した値「432」を数(38)として得て、計算を終了する。

【0221】

上述した筆算による計算を演算装置300で行う場合を説明する。図20右側に示すように、レジスタAには「A + 0.04 1.44」(数(33)に対応)が設定され、レジスタZには「5.88 × 0.04 0.2352」(数(34)に対応)が設定される。次いで、レジスタIに*i*(計算開始桁) = 56, レジスタWに*w*(計算終了ワード) = 15が設定され、演算が開始されると、レジスタXの出力データと、レジスタZの出力データとが演算され、演算結果がレジスタXに「0.25600 - 0.235200 0.020800」として設定される。続いて、レジスタZには「4.2 × 0.04² 0.00672」(数(35)に対応)が設定され、レジスタXの出力データと、レジスタZの出力データとが演算され、演算結果がレジスタXに「0.020800 - 0.006720 0.014080」として設定される。

【0222】

更に、レジスタZには「0.04³ 0.000064」(数(36)に対応)が設定され、レジスタXの出力データと、レジスタZの出力データとが演算され、演算結果がレジスタXに「0.014080320 - 0.000064 0.014016」(数(37)に対応)として設定される。また、レジスタBには「3 × 1.44² 6.2208」(数(38)に対応)が設定され、レジスタYには「3 × 1.44 4.32」(数(39)に対応)が設定される。

【0223】

以上のように、計算命令の種類「立方根」、計算数値「3」、計算開始桁「56桁」、計算終了桁「15ワード」が命令として与えられた場合、計算結果として「1.44」が得られる。

【0224】

上述した通り、平方根や立方根の演算のように、多桁の計算を順次繰り返して実行することにより解が求められる演算においては、順次行われる多桁の計算は桁指定されて行われる1命令に対応している場合が多い。従って、このような1命令を演算装置300で桁指定して多桁の計算を行うことができるので、平方根や立方根を演算するプログラムの作成が容易になる。また、上述した平方根や立方根の演算において、順次行われる多桁の計算は、演算装置300における処理単位と一致しているため、プログラムで設定した桁数とぴったり一致させて計算処理を行うことができる。これにより、必要とされる精度で演算結果を得ることができる。

【0225】

[第2の実施の形態の効果]

本第2の実施の形態によれば、演算装置300の可変パラメータ記憶部320に6ビットのレジスタIを備え、レジスタIに設定されるデータ*i*により、計算開始桁を1桁単位で設定可能とする。すなわち、計算開始桁を1桁単位で設定可能にし、計算開始桁から計算終了桁まで、桁数単位毎の小規模な十進計算に分けて行うことができ、効率の良い計算を行うことができる。また、ユーザ作成のプログラム等により予め計算命令で計算開始桁及び計算終了桁を設定しておくことで、プログラムROM310のプログラム命令311に設定されている計算桁数ちょうどの十進計算を行うことができ、効率の良い十進計算を容易に行うことができる。

【0226】

また、6ビットのレジスタIにより計算開始桁の桁単位指定を可能としつつ、演算器3

10

20

30

40

50

70においては4桁毎に演算を行う構成であるため、第1の実施の形態の演算装置200と略同様の回路構成により第2の実施の形態を実現することができる。つまり、計算桁数に応じたプログラムの変更等が殆ど無く、簡単な回路変更で桁単位指定を可能とすることができ、装置の低コスト化、小型化を実現することができる。

【0227】

[第2の実施の形態の変形例]

(1) 計算桁数の単位

上述した第2の実施の形態では、1ワード(16ビット)を単位とし、10進表現4桁毎に演算を行うこととしたが、計算桁数の単位は4桁に限らない。また、指定された計算桁が4の倍数でない場合、計算終了ワードにおける不要データをマスクすることとしたが、計算開始桁を含む所定の第nワードから4桁毎に演算を開始し、その計算開始ワードにおける不要データをマスクする構成としてもよい。

10

(2) 計算終了桁を可変

上述した第2の実施の形態では、計算終了桁はレジスタWにより、ワード単位で指定されることとしたが、可変パラメータ記憶部320に、6ビットのレジスタを更に設け、このレジスタにより計算終了桁を桁単位で指定可能としてもよい。

(3) 指数部の計算桁数を可変

また、上述した第2の実施の形態では、第2ワードから第15ワードのいずれかを計算開始桁又は計算開始ワード或いは計算終了桁とし、仮数部の計算桁数を可変としたが、指数部の計算桁数を可変しても良いのは勿論である。

20

(4) レジスタのワード数

また、レジスタ部360のワード数を16として説明したが、これに限られないことは勿論である。また、仮数部を14ワード、指数部を2ワードとしたが、この割合は適宜変更可能であることも勿論である。

(5) レジスタの数

更に、本第2の実施の形態においては、レジスタ部360が有するレジスタの数を4又は5として説明を行ったが、レジスタの数はこれに限定されないことは勿論である。

【0228】

[第3の実施の形態]

次に、第3の実施の形態について説明する。

30

本第3の実施の形態における演算システムは、演算器における演算処理をパイプライン処理することにより、演算速度を向上させるものである。以下では、上述した第1の実施の形態における演算システムS1の演算装置200にパイプライン処理を適用した場合について説明する。尚、第1の実施の形態と同一部分については、同一の符号を付して、図示及び詳細な説明を省略する。以下、本第3の実施の形態に特徴的な部分につき詳細に説明する。

【0229】

図21は、演算器のパイプライン処理に係る要部の構成要素ブロックを示す図である。同図によれば、パイプライン処理に係る構成要素ブロックとして、演算器270、レジスタ部260、アドレスカウンタ240を備える。ここで、レジスタ部260は、アドレス指定及びデータ出力をそれぞれ2ポート有する2ポート方式のレジスタであるため、便宜的に第1ポートを有するレジスタをFレジスタ270aとし、第2ポートを有するレジスタをSレジスタ270bとして図示するが、実際の回路においては、物理的に1つのメモリにより構成されるものであっても良い。また、アドレスカウンタ240は、レジスタ部260の2つのポートにアドレスを出力するものであり、第1ポートにアドレスを出力するアドレスカウンタ240aと、第2ポートにアドレスを出力するアドレスカウンタ240bとを有して構成される。

40

【0230】

Fレジスタ260aは、2ワード(32ビット)によるアクセスが可能であり、アドレスカウンタ240aにより指定されたアドレスから32ビット分のデータを読み出して、

50

演算器 270 に出力する。また、演算器 270 から入力される 2ワード (32ビット) のデータをアドレスカウンタ 240 a により指定されたアドレスに書き込む。つまり、Fレジスタ 260 a は、32ビットデータの読み出しと、32ビットデータの書き込みを 1クロック毎に交互に行う。

【0231】

Sレジスタ 260 b は、1ワード (16ビット) のアクセスが可能であり、アドレスカウンタ 240 b により指定されたアドレスから 16ビット分のデータを読み出して演算器 270 に出力する。

【0232】

演算器 270 は、乗算回路 270 a、ラッチ回路 270 b、Read F / F 270 c、セクタ 270 d、加算回路 270 e、Write F / F 270 f を備えて構成される。乗算回路 270 a は、Sレジスタ 260 b から入力される 16ビットのデータと、ラッチ回路 270 b から入力される 4ビットのデータとを乗算して、乗算結果を 16ビットのデータとして加算回路 270 e に出力する。ラッチ回路 270 b は、F / F (Flip Flop) により構成され、4ビットのデータを保持して乗算回路 270 a に出力する。

10

【0233】

Read F / F 270 c は、Fレジスタ 260 a から入力される 16ビットのデータを保持して、セクタ 270 d に出力する。セクタ 270 d は、Fレジスタ 260 a から入力される 16ビットのデータと、Read F / F 270 c から入力される 16ビットのデータとを 1クロック毎に交互に選択して加算回路 270 e に出力する。

20

【0234】

加算回路 270 e は、乗算回路 270 a から入力される 16ビットのデータと、セクタ 270 d から入力される 16ビットのデータとを被演算子として加算して、加算結果を 16ビットのデータとして Write F / F 270 f 及び Fレジスタ 260 a に出力する。また、加算回路 270 e は、加算結果の一部を 4ビットのデータとしてラッチ回路 270 b に出力する。Write F / F 270 f は、加算回路 270 e から出力される 16ビットのデータを保持して Fレジスタ 260 a に出力する。

【0235】

次に、演算器 270 の動作について、図 22 に示すタイミングチャートに基づいて説明する。図 22 は、計算命令「 $X_0 \sim 5 + Y_0 \sim 5$ 」に基づいて行われる積和演算のタイミングチャートを示す図である。同図中、Fレジスタ 260 a に入出力されるデータを X、Sレジスタ 260 b から出力されるデータを Y、ラッチ回路 270 b から出力されるデータを M として以下説明する。

30

【0236】

まず、1クロック目で、Fレジスタ 260 a からデータ X_0 、 X_1 が読み出され、データ X_0 はセクタ 270 d で選択されて、加算回路 270 e に出力される。データ X_1 は、Read F / F 270 c に保持される。また、Sレジスタ 260 b からデータ Y_0 が読み出され、データ Y_0 は乗算回路 270 a に出力される。

【0237】

次に、2クロック目で、乗算回路 270 a と加算回路 270 e により積和演算「 $X_0 + Y_0 \times M$ 」が行われる。また、Read F / F 270 c からデータ X_1 がセクタ 270 d に出力され、セクタ 270 d でデータ X_1 が選択され、加算回路 270 e に出力される。更に、Sレジスタ 260 b からデータ Y_1 が読み出され、データ Y_1 は乗算回路 270 a に出力される。

40

【0238】

続いて、3クロック目で、乗算回路 270 a と加算回路 270 e により積和演算「 $X_1 + Y_1 \times M$ 」が行われる。また、先に演算された「 $X_0 + Y_0 \times M$ 」の演算結果は、Write F / F 270 f に保持される。また、Fレジスタ 260 a からデータ X_2 、 X_3 が読み出され、データ X_2 はセクタ 270 d で選択されて、加算回路 270 e に出力される。データ X_3 は、Read F / F 270 c に保持される。更に、Sレジスタ 260 b が

50

らデータY2が読み出され、データY2は乗算回路270aに出力される。

【0239】

そして、4クロック目で、WriteF/F270fで保持されている演算結果「 $X_0 + Y_0 \times M$ 」と、加算回路270eから出力される演算結果「 $X_1 + Y_1 \times M$ 」とがFレジスタ260aのX0, X1に書き込まれる。また、乗算回路270aと加算回路270eにより積和演算「 $X_2 + Y_2 \times M$ 」が行われ、ReadF/F270cからデータX3がセクタ270dに出力され、セクタ270dでデータX3が選択されて、加算回路270eに出力される。更に、Sレジスタ260bからデータY3が読み出され、データY3は乗算回路270aに出力される。

【0240】

このように、計算命令「 $X_0 \sim 5 + Y_0 \sim 5$ 」に基づく積和演算は、1サイクルを4クロックで実行し、3クロック目以降、2クロックを次サイクルと重複させて順次処理を行うことにより、高速に演算処理を行うことができる。すなわち、上述した積和演算は、1サイクルで2つの演算（例えば、 $X_0 + Y_0 \times M$, $X_1 + Y_1 \times M$ ）を実行するが、「 $X_0 \sim 5 + Y_0 \sim 5$ 」には6回分の演算が含まれるため、3サイクルを要し、総クロック数は $3 \times 4 = 12$ クロックとなる。しかし、パイプライン処理により、3サイクルを8クロックで実行することができ、4クロック分の演算時間を短縮することができる。

【0241】

[第3の実施の形態の効果]

以上のように、本第3の実施の形態によれば、32ビットデータ(2ワード)のアクセスが可能なFレジスタの入力段と出力段に、一時記憶手段としてWriteF/F270f, ReadF/F270cを設け、一時記憶手段に書込データと読出データを16ビット分ずつそれぞれ一時的に保持させる。そして、演算器270において16ビットデータの計算を1クロック毎に行うと同時に、Fレジスタ260aにおいて32ビットデータの書き込み及び読み出しを1クロック毎に交互に行うことにより、パイプライン処理を実現する。これにより、演算装置の演算速度を高速化することができる。

【0242】

[第3の実施の形態の変形例]

尚、上述した計算命令「 $X_0 \sim 5 + Y_0 \sim 5$ 」は一例であり、計算開始ワード及び計算終了ワードは適宜変更可能である。例えば、計算命令「 $X_0 \sim n + Y_0 \sim n$ 」とした場合、nが奇数値であれば、クロック数 = $n + 3$ で積和演算処理を行うことができる。また、本第3の実施の形態では、構成要素ブロックに乗算回路270aと、加算回路270eとを備え、積和演算を行う場合を例として説明したが、演算処理は積和演算処理に限るものではなく、その他種々の演算処理に適用可能なことは勿論である。

【0243】

更に、第3の実施の形態は、上述した第1の実施の形態における演算装置200においてパイプライン処理する場合を例として説明したが、上述した第2の実施の形態における演算装置300においてパイプライン処理する構成であっても良い。この場合、計算開始桁を桁単位で指定して、パイプライン処理による演算を行うことができる。

【0244】

[第4の実施の形態]

次に、第4の実施の形態について説明する。

図23は、本第4の実施の形態における演算システムS3の概略構成図である。同図によれば、演算システムS3は、PC100と、USB演算装置400とが、それぞれのUSB端子U2, U1を介して接続され、相互にデータ通信可能なように構成されている。

【0245】

PC100は、キーボード等の入力装置から演算データの入力を行わせ、入力された演算データをUSB端子U2, U1を介して接続されたUSB演算装置400に送信する。尚、第4の実施の形態において、演算データとは、演算の種類(例えば、四則演算、初等関数、高等関数)、演算対象の数値データ(被演算数、演算数)、有効桁数(計算桁数)

10

20

30

40

50

を含むデータをいう。また、PC100は、USB演算装置400から演算結果を受信すると、モニタ等の表示装置に演算結果を表示させる。

【0246】

図24は、第4の実施の形態におけるUSB演算装置400の要部構成を示すブロック図である。同図によれば、USB演算装置400は、CPU401、フラッシュメモリ402、SRAM403、ROM404、通信制御部405、共有メモリ406等を備えて構成されている。

【0247】

CPU401は、FPGA(Field Programmable Gate Array)等により構成され、ROM404に格納されるシステムプログラム、フラッシュメモリ402に格納される各種処理プログラムを読み出して、装置全体の制御や各種演算処理を行う。このCPU401は、上述した第1の実施の形態の図3に示す演算装置200や第2の実施の形態の図9に示す演算装置300に相当するものである。従って、このCPU401で、指定された計算開始桁又は計算開始ワードから計算終了ワードまでの演算処理を行うことにより、所望の計算桁数で各種演算を行うことができる。

10

【0248】

フラッシュメモリ402は、電氣的に書き換え可能な不揮発性メモリにより構成され、各種演算プログラムを記憶したり、PC100から転送されてきた演算プログラムを書き換え可能に記憶する。このフラッシュメモリ402に記憶されたプログラムは、例えば、上述した第1の実施の形態においてはプログラムROM210上に展開される等して用いられるものである。

20

【0249】

SRAM403は、CPU401の作業領域等として用いられる記憶部であり、フラッシュメモリ402やROM404から読み出されたプログラム、CPU401の演算結果等が格納される。このSRAM403は、例えば、上述した第1の実施の形態の演算装置200や第2の実施の形態の演算装置300中のレジスタ部260、360に相当するものである。

【0250】

ROM404は、装置本体の初期状態を設定するためのシステムプログラムや、USBで接続された他機器との間でデータ通信を行うための通信制御処理プログラム等を記憶する。

30

【0251】

通信制御部405は、USB端子U1を備え、USB端子U1を介して接続された他機器との間で、USB規格に基づいたデータの通信を制御する。具体的に、通信制御部405は、PC100からデータを受信すると、シリアル又はパラレル変換して共有メモリ406に書き込み、CPU401により共有メモリ406に演算結果が書き込まれると、演算結果を読み出してPC100にデータを転送する通信制御処理を実行する。

【0252】

共有メモリ406は、電氣的に書き換え可能なメモリにより構成され、PC100から受信した演算データ(例えば、演算の種類、演算対象の数値データ、有効桁数)、CPU401から出力された演算結果等を記憶する。

40

【0253】

次に、本第4の実施の形態における演算システムS3の動作について説明する。図25(a)は、PC100により実行される通信処理、図25(b)は、通信制御部405により実行される通信処理、図25(c)は、CPU401により実行される通信処理を示すフローチャートである。

【0254】

まず、PC100により実行される通信処理について説明する。図25(a)に示すように、ユーザ操作に応じて、演算データが入力されると(ステップST31)、PC100は、USB端子U2を介して接続されたUSB演算装置400に、入力された演算デー

50

タを送信する（ステップST32）。

【0255】

続いて、PC100は、USB演算装置400からの演算結果の受信を待機し（ステップST33）、演算結果を受信すると（ステップST34）、受信した演算結果を表示部（図示せず）に表示させて（ステップST35）、本通信処理を終了する。

【0256】

次に、USB演算装置400の通信制御部405により実行される通信処理について説明する。図25（b）に示すように、通信制御部405は、USB端子U1を介して接続されたPC100から演算データを受信すると（ステップST41）、受信した演算データを共有メモリ406に書き込む（ステップST42）。続いて、CPU401へのリセット信号の出力を解除して、CPU401を動作させる（ステップST43）。

10

【0257】

そして、CPU401から入力されるビジー信号を監視し（ステップST44）、ビジー信号がオフになった場合（ステップST44；オフ）、リセット信号を出力してCPU401の動作を停止させる（ステップST45）。更に、共有メモリ406からCPU401により書き込まれた演算結果を読み出して（ステップST46）、演算結果をPC100に送信し（ステップST47）、本通信処理を終了する。

【0258】

次に、USB演算装置400のCPU401により実行される通信処理について説明する。図25（c）に示すように、CPU401は、通信制御部405からリセット信号の出力が解除されたか否かを判別し（ステップST51）、リセット信号の出力が解除された場合（ステップST51；YES）、ビジー信号をオンにして通信制御部405に出力する（ステップST52）。続いて、共有メモリ406から演算データを読み出して（ステップST53）、演算処理を実行する（ステップST54）。

20

【0259】

演算処理について説明する。図26は、CPU401により実行される演算処理を示すフローチャートである。図26に示すように、CPU401は、演算データに含まれる演算の種類に対応する演算プログラムをROMから読み出す（ステップST61）。そして、演算データに含まれる有効桁数に対応して演算プログラム中で計算命令の計算桁数と計算の種類を設定して、各計算命令を実行していき演算プログラムの計算を行う（ステップST62）

30

【0260】

演算処理を終了すると、CPU401は、演算結果を共有メモリ406に書き込み（ステップST55）、ビジー信号をオフにして通信制御部405への出力を停止する（ステップST56）。そして、ステップST51に移行して上述した処理を繰り返して実行する。

【0261】

[第4の実施の形態の効果]

以上のように、第4の実施の形態によれば、USB演算装置400と、PC100とをそれぞれのUSB端子U1、U2を介して接続し、PC100から演算データ（例えば、演算の種類、演算対象の数値データ、有効桁数）の入力を行わせ、USB演算装置400で演算データに基づいて所望の計算桁数にて演算を行い、演算結果をPC100に送信する。そして、演算結果をPC100の表示部に表示することで、任意の計算桁数で十進計算を行う機能を有しないPCにおいても所望の計算桁数で十進数計算を高速に行うことができる。

40

【0262】

[第4の実施の形態の変形例]

本第4の実施の形態においては、USB規格によりデータ通信を行う場合を例として説明したが、通信方式はUSBに限らず、SCSI（Small Computer System Interface）や、IrDA（Infrared Data Association）規格に準じた無線通信等であってもよい。

50

また、USB 端末装置 400 と接続される電子機器は、PC に限らず、PDA、ノート型コンピュータ、携帯端末等であっても良い。

【0263】

また、演算データとして、演算の種類、演算対象の数値データ、有効桁数を含む場合を例として説明したが、演算データはこれに限定されない。例えば、演算データに、計算開始桁又は計算開始ワードと計算終了桁とを含み、これらに基づいて計算桁数を取得する構成であっても良い。

【0264】

或いは、図 27 に示す演算システム S4 のように構成されるときもよい。すなわち、PC 200 は、演算データが入力されると、入力された演算データに基づいた演算プログラムを作成する。具体的には、プログラム中の各計算桁数が演算データに含まれる有効桁数に対応した桁数に設定する等して入力された演算の種類に応じた演算プログラムを作成する。そして、作成した演算プログラムを USB 演算装置 500 に送信する。この場合、USB 演算装置 500 は、PC 200 から受信した演算プログラムに従って、各計算命令を実行することにより十進計算を行う。この構成によれば、メモリ用両党の点で、USB 演算装置 500 より有利な PC 200 側で各種演算プログラムを作成することができるため、数多くの演算プログラムを USB 演算装置 500 に実行させることができる。

【図面の簡単な説明】

【0265】

【図 1】本実施の形態での演算装置の基本概略構成。

【図 2】第 1 の実施の形態の演算システムの概略構成図。

【図 3】第 1 の実施の形態の演算装置の要部回路構成図。

【図 4】レジスタ部の構成図。

【図 5】上位アドレス Fuad、Suad と指定されるレジスタとの対応関係。

【図 6】第 1 の実施の形態の開始ワードの変更による演算桁数の可変を示す図。

【図 7】第 1 の実施の形態の拡張命令コード EXT の構成。

【図 8】第 1 の実施の形態の拡張命令コード EXT と演算装置の機能例との対応関係。

【図 9】第 2 の実施の形態の演算装置の要部回路構成図

【図 10】第 2 の実施の形態の拡張命令コード EXT の構成。

【図 11】第 2 の実施の形態の計算開始桁の変更による計算桁数の可変を示す図。

【図 12】第 2 の実施の形態の拡張命令コード EXT と演算装置の機能例との対応関係。

【図 13】桁指定計算処理を示すフローチャート。

【図 14】第 2 の実施の形態の演算装置での平方根の演算処理を説明する図。

【図 15】第 2 の実施の形態の演算装置での平方根の演算処理を説明する図。

【図 16】第 2 の実施の形態の演算装置での平方根の演算処理を説明する図。

【図 17】第 2 の実施の形態の演算装置での平方根の演算処理を説明する図。

【図 18】第 2 の実施の形態の演算装置での立方根の演算処理を説明する図。

【図 19】第 2 の実施の形態の演算装置での立方根の演算処理を説明する図。

【図 20】第 2 の実施の形態の演算装置での立方根の演算処理を説明する図。

【図 21】第 3 の実施の形態の構成要素ブロック図。

【図 22】タイミングチャートを示す図。

【図 23】第 4 の実施の形態の演算システムの概略構成図。

【図 24】USB 演算装置の要部構成を示すブロック図。

【図 25】(a) PC により実行される通信処理を示すフローチャート、(b) 通信制御部により実行される通信処理を示すフローチャート、(c) CPU により実行される通信処理を示すフローチャート。

【図 26】CPU により実行される演算処理を示すフローチャート

【図 27】第 4 の実施の形態の変形例

【図 28】従来の演算システムの論理的な階層構造を示す図。

【符号の説明】

10

20

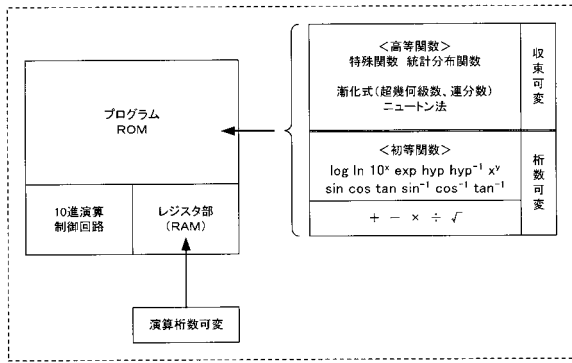
30

40

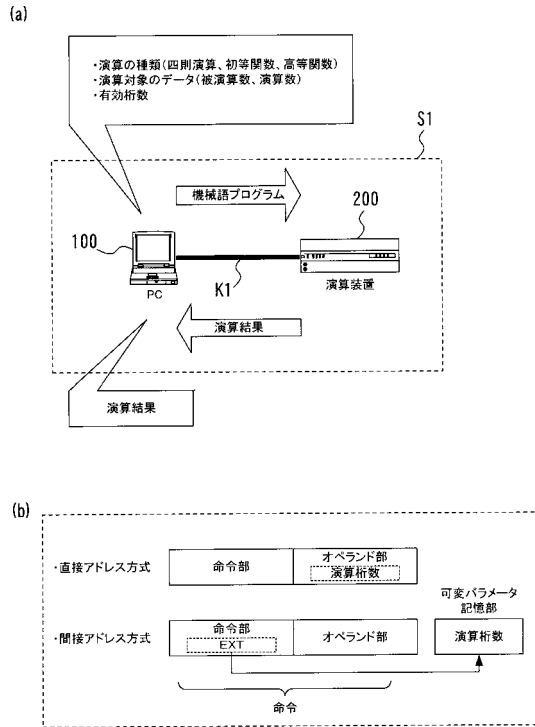
50

【 0 2 6 6 】		
1 0 0、2 0 0	P C	
2 0 0、3 0 0	演算装置	
2 1 0、3 1 0	プログラムROM	
2 1 2、3 1 2	プログラムカウンタ	
2 1 4、3 1 4	ラッチ部	
2 1 6、3 1 6	命令デコーダ	
2 2 0、3 2 0	可変パラメータ記憶部	
W、V、I	レジスタ	
2 3 1、2 3 2、2 3 3、2 3 4	セレクタ	10
3 3 1、3 3 2、3 3 3、3 3 4	セレクタ	
2 4 0、3 4 0	アドレスカウンタ	
2 5 0、3 5 0	終了判定回路	
2 5 2、3 5 2	一致回路	
2 5 4、3 5 4	ORゲート	
2 6 0、3 6 0	レジスタ部	
X、Y、Z、A、B	レジスタ	
2 7 0、3 7 0	演算器	
4 0 0、5 0 0	U S B 演算装置	
4 0 1	C P U	20
4 0 2	フラッシュメモリ	
4 0 3	S R A M	
4 0 4	R O M	
4 0 5	通信制御部	
4 0 6	共有メモリ	
K 1、K 2	通信ケーブル	
U 1、U 2	U S B 端子	

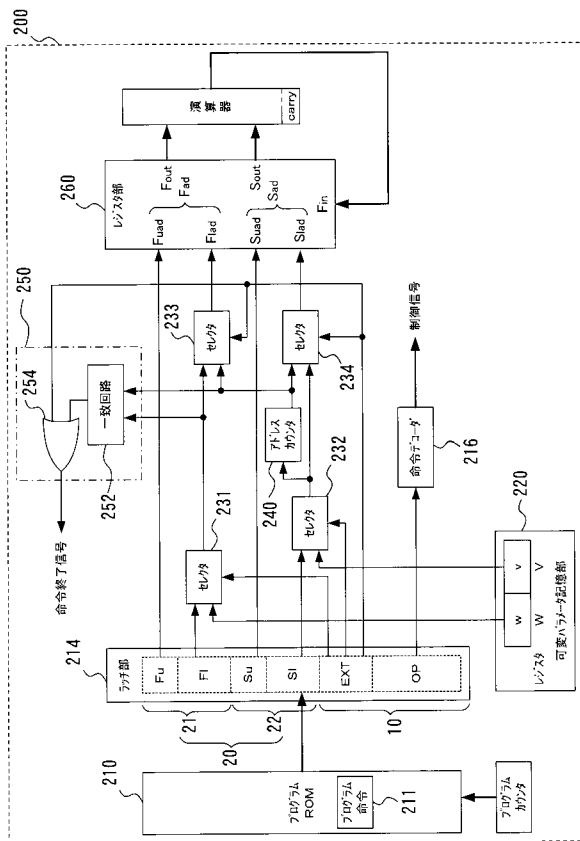
【図1】



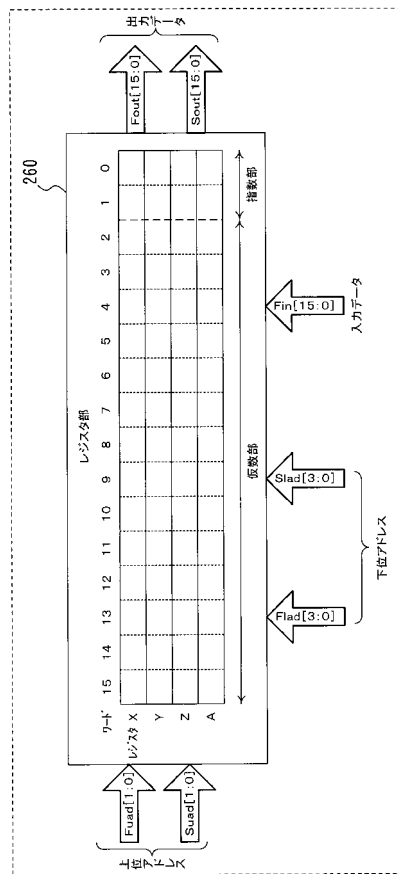
【図2】



【図3】



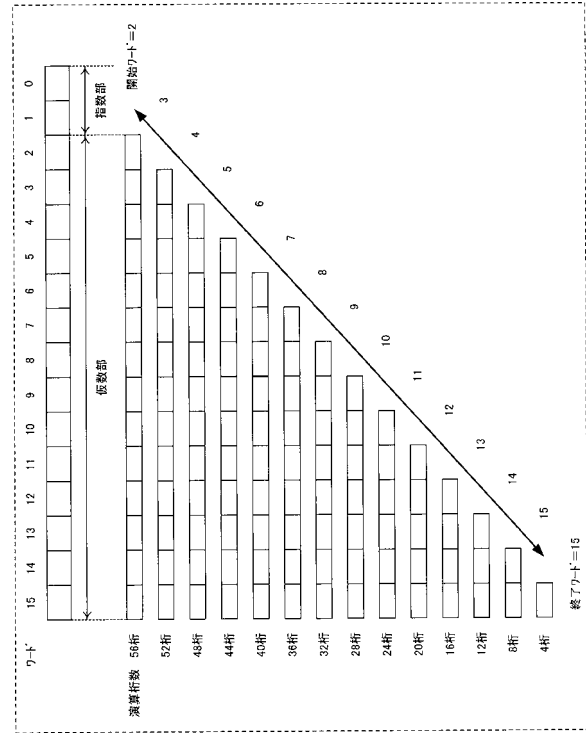
【図4】



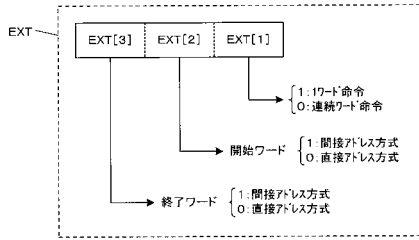
【図5】

Fuad, Suad	レジスタ
00	X
01	Y
10	Z
11	A

【図6】



【図7】

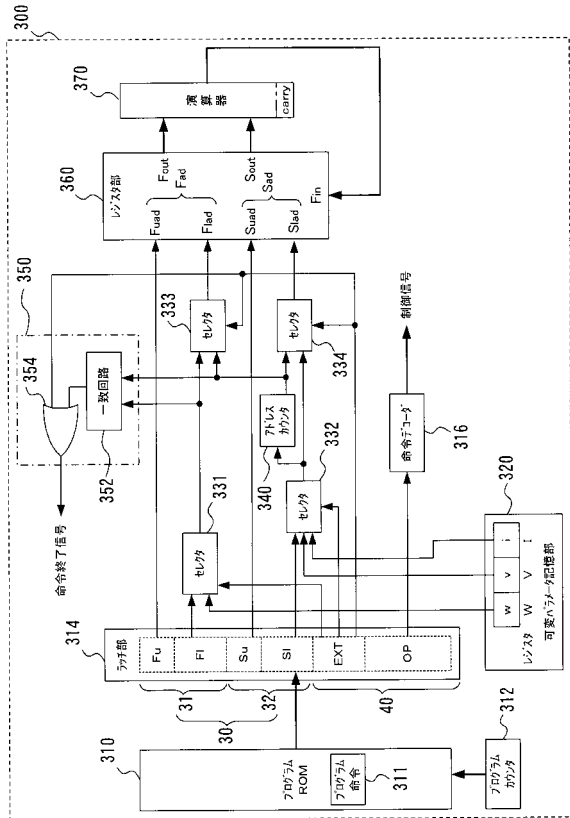


【図8】

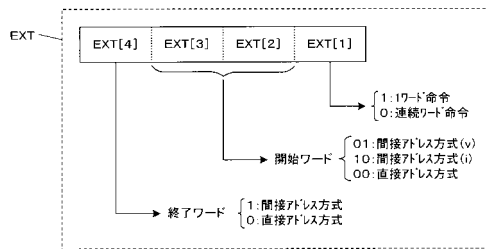
パターン	EXT		命令		機能例	
	連続ワード命令	間接アドレス方式	命令部	オペランド部	ニーモニック	動作
(A)	000	連続ワード	ADD 000	X15 Y4	X4~15+Y	X4~15+Y4~15→X4~15
(B)	010	連続ワード	ADD 010	X15 Y*	Xy~15+Y	Xy~15+Yv~15→Xv~15
(C)	100	連続ワード	ADD 100	X* Y4	X4~w+Y	X4~w+Y4~w→X4~w
(D)	110	連続ワード	ADD 110	X* Y*	Xv~w+Y	Xv~w+Yv~w→Xv~w
(E)	001	1ワード	ADD 001	X15 Y4	X15+Y4	X15+Y4→X15
(F)	011	1ワード	ADD 011	X15 Y*	X15+Yv	X15+Yv→X15
(G)	101	1ワード	ADD 101	X* Y4	Xw+Y4	Xw+Y4→Xw
(H)	111	1ワード	ADD 111	X* Y*	Xw+Yv	Xw+Yv→Xw

ADD: 加算
Rn: レジスタRnに格納されているデータw
v: レジスタRvに格納されているデータv
*: 任意の値

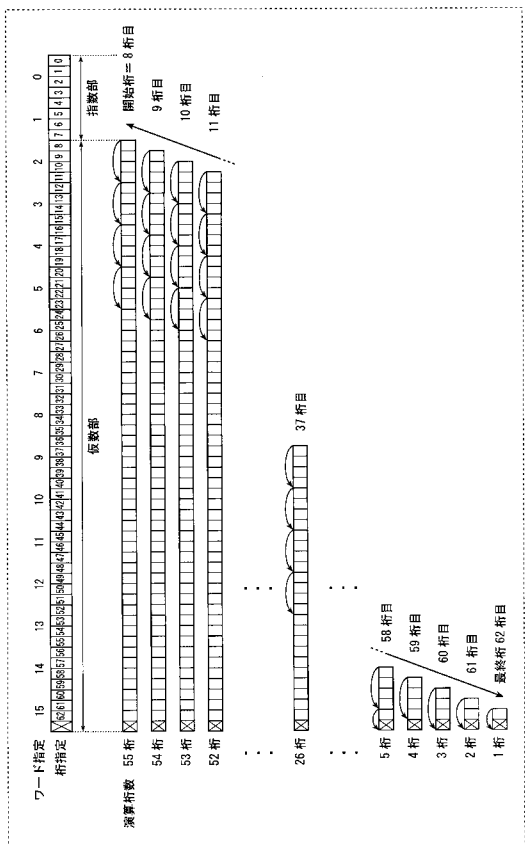
【図9】



【図10】



【図11】

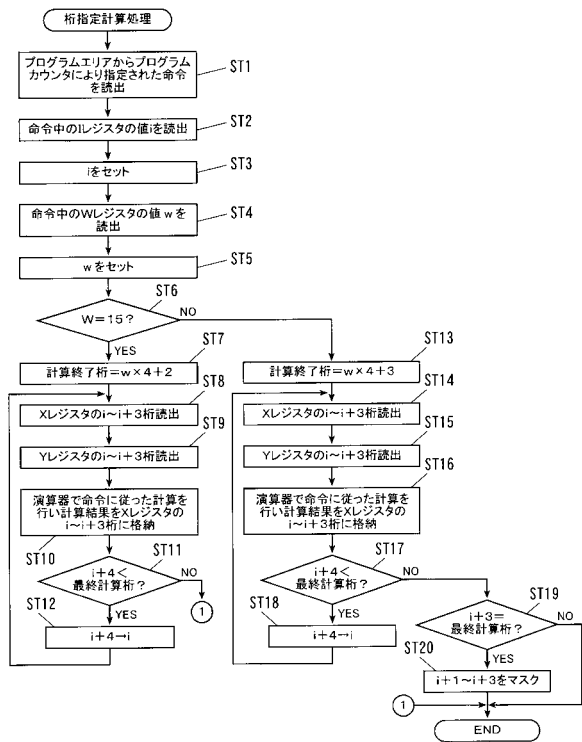


【図12】

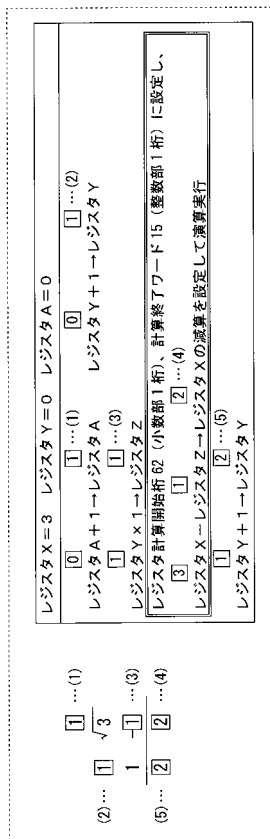
パターン	EXT	機能例		
		連続/1ワード命令	命令部	動作
(I)	0000	直接アドレス方式	X15 Y4	X4~15+Y4~15-X4~15
(J)	0010	間接アドレス	X15 Y*	X0~15+Y0~15-X0~15
(K)	0100	間接アドレス	X15 Y*	X0~15+Y0~15-X0~15
(L)	1000	間接アドレス	X* Y4	X0~w+Y0~w-X0~w
(M)	1010	間接アドレス	X* Y*	X0~w+Y0~w-X0~w
(N)	1100	間接アドレス	X* Y*	X0~w+Y0~w-X0~w
(O)	0001	直接アドレス	X15 Y4	X15+Y4-X15
(P)	0011	間接アドレス	X15 Y*	X15+Y0-X15
(Q)	0101	間接アドレス	X15 Y*	X15+Y0-X15
(R)	1001	間接アドレス	X* Y4	Xw+Y4-Xw
(S)	1011	間接アドレス	X* Y*	Xw+Y0-Xw
(T)	1101	間接アドレス	X* Y*	Xw+Y0-Xw

ADD:加算
 SUB:減算
 Rn:レジスタnの第nワード又はn桁目に格納されている値
 * :任意の値

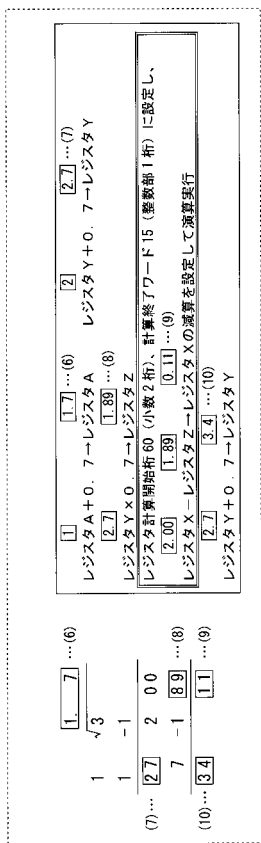
【図 13】



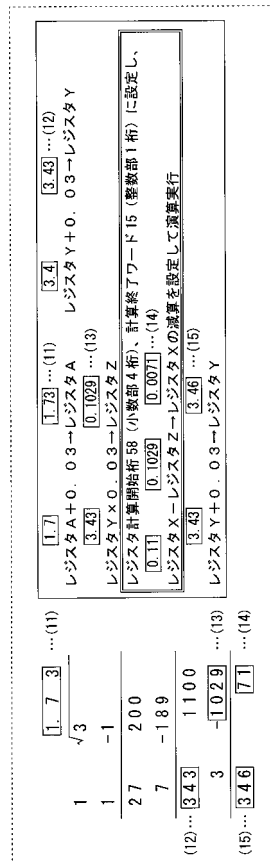
【図 14】



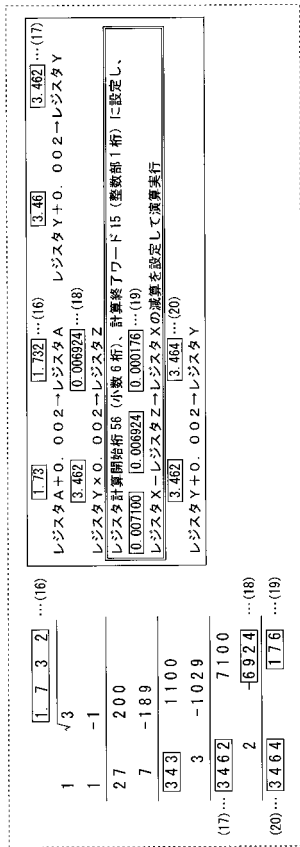
【図 15】



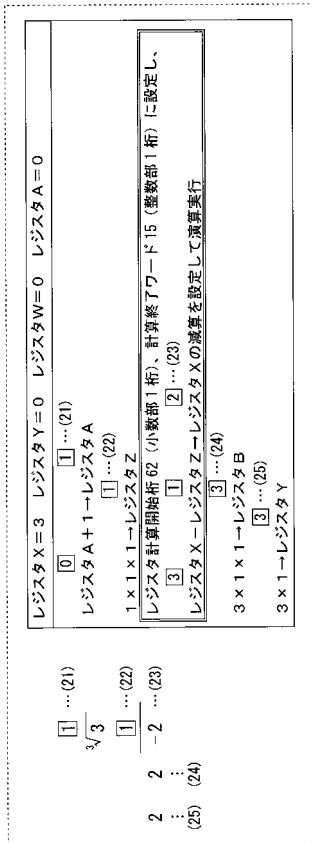
【図 16】



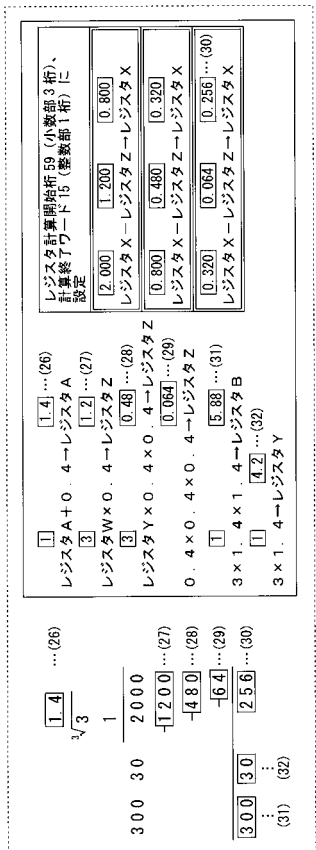
【図 17】



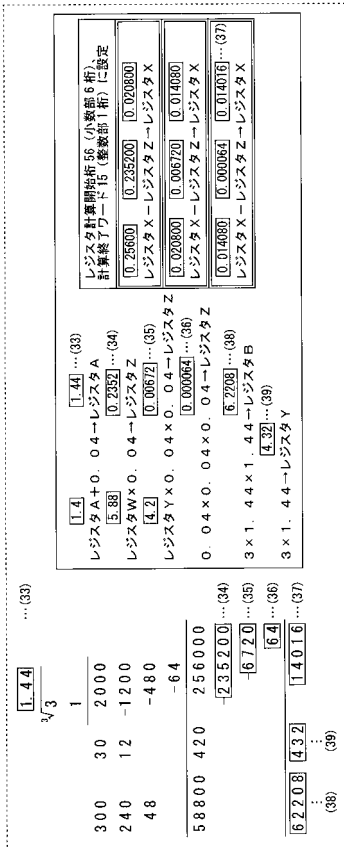
【図 18】



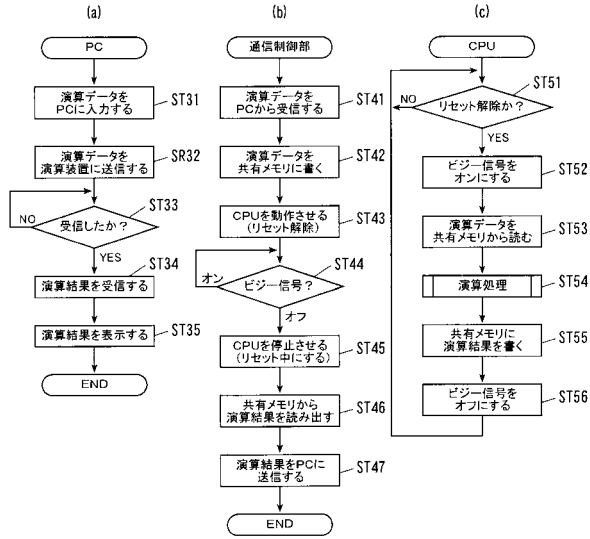
【図 19】



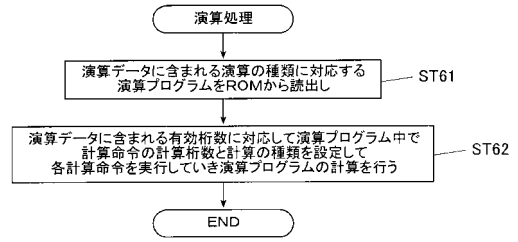
【図 20】



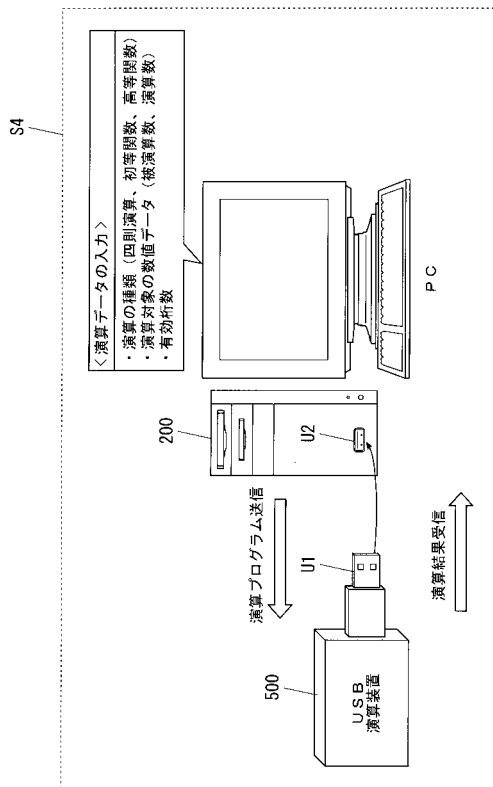
【図 25】



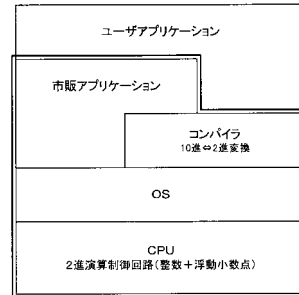
【図 26】



【図 27】



【図 28】



フロントページの続き

- (56)参考文献 特開昭62-075838(JP,A)
特開平05-089041(JP,A)
特開昭57-043239(JP,A)
特開平05-274114(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 9/305
G06F 9/32
G06F 7/00