

(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2004-206545
(P2004-206545A)

(43) 公開日 平成16年7月22日(2004.7.22)

(51) Int. Cl.⁷
G06F 3/00

F I
G06F 3/00 654C

テーマコード(参考)
5E501

審査請求 未請求 請求項の数 6 O L (全 44 頁)

(21) 出願番号 特願2002-376565 (P2002-376565)
(22) 出願日 平成14年12月26日(2002.12.26)

(71) 出願人 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
(74) 代理人 100090033
弁理士 荒船 博司
(74) 代理人 100093045
弁理士 荒船 良男
(72) 発明者 三改木 里美
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
Fターム(参考) 5E501 AA11 AC17 BA05 CA04 CB02
CB05 CB09 CB11 EA05 EA08
EA14 EB13 EB14 FA06 FA14
FA25 FA43

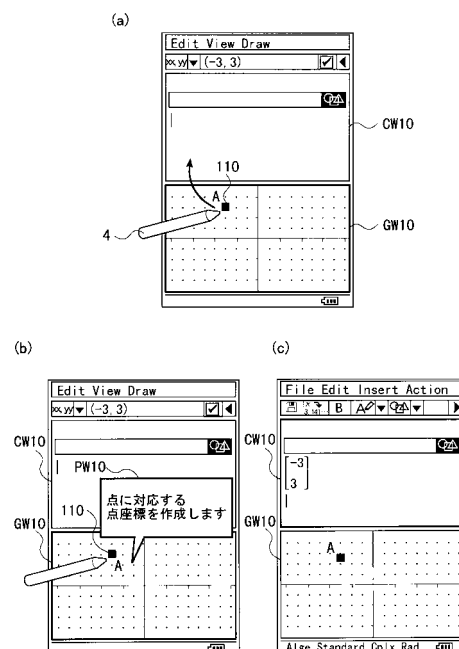
(54) 【発明の名称】 電子機器及びプログラム

(57) 【要約】

【課題】 ユーザによる操作入力の簡易性を損なうことなく、当該操作入力に応じてなされる動作を容易に理解することができる電子機器を実現すること。

【解決手段】 幾何ウィンドウGW10において、入力ペン4を用いて点オブジェクト110を選択し、ドラッグ&ドロップ操作によって計算ウィンドウCW10へ移動させる。すると、この操作に応じてなされる動作内容を表す動作説明メッセージ「点に対応する点座標を作成します」が、ポップアップウィンドウPW10に表示される。その後、点オブジェクト110に対応する点座標(-3, 3)を示す1×2行列が、計算ウィンドウCW10に表示される。

【選択図】 図10



【特許請求の範囲】**【請求項 1】**

それぞれ異なる表示形態でデータを表示する複数の表示画面を有する表示部と、前記複数の表示画面の内、一の表示画面に対する複写操作がなされた後、他の表示画面に対する貼付操作がなされた場合に、当該貼付操作がなされた他の表示画面の表示形態を判別する判別手段と、

この判別手段によって判別された表示形態に応じた変換内容を説明するメッセージを表示させる制御を行うメッセージ表示制御手段と、

このメッセージ表示制御手段によるメッセージの表示制御が行われた後、前記複写操作によって指定されたデータの表示形態を前記判別手段によって判別された表示形態に変換して、前記他の表示画面に表示させる制御を行う変換表示制御手段と、

を備えることを特徴とする電子機器。

10

【請求項 2】

第 1 表示画面、及び、異なる複数の表示形態でデータ表示可能な第 2 表示画面、を有する表示部と、

前記第 1 表示画面に対する複写操作がなされた後、前記第 2 表示画面に対する貼付操作がなされた場合に、当該第 2 表示画面に表示可能な複数の表示形態を判別する判別手段と、この判別手段によって判別された複数の表示形態を候補として表示させる制御を行う候補表示制御手段と、

この候補表示制御手段によって表示制御された複数の表示形態の候補の内から、何れか一つを選択する選択手段と、

この選択手段によって選択された表示形態に応じた変換内容を説明するメッセージを表示させる制御を行うメッセージ表示制御手段と、

このメッセージ表示制御手段によるメッセージの表示制御が行われた後、前記複写操作によって指定されたデータを前記選択手段によって選択された表示形態に変換して、前記第 2 表示画面に表示させる制御を行う変換表示制御手段と、

を備えることを特徴とする電子機器。

20

【請求項 3】

前記候補表示制御手段は、

前記複写操作によって指定されたデータを表示することができる表示形態を、前記判別手段によって判別された複数の表示形態の内から選択する手段と、

この手段によって選択された表示形態を候補として表示させる制御を行う手段と、

を備えることを特徴とする請求項 2 に記載の電子機器。

30

【請求項 4】

前記表示部は、変換内容表示画面を更に有し、

前記メッセージ表示手段は、前記メッセージを前記変換内容表示画面に表示させる制御を行う手段を有することを特徴とする請求項 1 ~ 3 の何れか一項に記載の電子機器。

【請求項 5】

それぞれ異なる表示形態でデータを表示する複数の表示画面を有する表示部に接続される、又は当該表示部を備えるコンピュータに、

前記複数の表示画面の内、一の表示画面に対する複写操作がなされた後、他の表示画面に対する貼付操作がなされた場合に、当該貼付操作がなされた他の表示画面の表示形態を判別する判別機能と、

この判別機能によって判別された表示形態に応じた変換内容を説明するメッセージを表示する制御を行うメッセージ表示制御機能と、

このメッセージ表示制御機能によるメッセージの表示制御が行われた後、前記複写操作によって指定された表示形態のデータを前記判別機能によって判別された表示形態に変換して、前記他の表示画面に表示させる制御を行う変換表示制御機能と、

を実現させるためのプログラム。

40

【請求項 6】

50

第1表示画面、及び、異なる複数の表示形態でデータ表示が可能な第2表示画面、を有する表示部に接続される、又は当該表示部を備えるコンピュータに、前記第1表示画面に対する複写操作がなされた後、前記第2表示画面に対する貼付操作がなされた場合に、当該第2表示画面に表示可能な複数の表示形態を判別する判別機能と、この判別機能によって判別された複数の表示形態を候補として表示させる制御を行う候補表示制御機能と、この候補表示制御機能によって表示制御された複数の表示形態の候補の内から、何れか一つを選択する選択機能と、この選択機能によって選択された表示形態に応じた変換内容を説明するメッセージを表示させる制御を行うメッセージ表示制御機能と、このメッセージ表示制御機能によるメッセージの表示制御が行われた後、前記複写操作によって指定されたデータを前記選択手段によって選択された表示形態に変換して、前記第2表示画面に表示させる制御を行う変換表示制御機能と、を実現させるためのプログラム。

10

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、電子機器及びプログラムに関する。

【0002】

【従来の技術】

20

従来から、グラフの表示機能を備えたグラフ関数電卓と呼ばれる小型の電子機器が知られている。また、グラフ関数電卓には、計算機能や図形描画機能等の様々な機能を搭載したものがあ

【0003】

このようなグラフ関数電卓では、例えば、計算機能によって求めた各種演算結果をグラフ化して表示させたり、幾何図形の式を入力して対応する幾何図形を表示させることができる。このため、グラフ関数電卓は、エンジニアの技術計算や教育の現場において活用されている。

【0004】

グラフ関数電卓の操作方法としては、グラフ関数電卓が備える各種機能が割り当てられた操作キー（例えば、数字キーや実行キー、グラフキー等）を操作する方法が一般的である（例えば、特許文献1参照）。また、近年では、更に、入力ペンやマウス等のポインティングデバイス等を用いて表示画面上のデータを指定し、所望の位置まで移動させる、いわゆるドラッグ&ドロップ操作によって様々な操作入力を行えるものもある。

30

【0005】

【特許文献1】

特開昭61-261780号公報

【0006】

【発明が解決しようとする課題】

しかしながら、上述したドラッグ&ドロップ操作による操作入力では、操作入力の簡易性を実現してはいるが、そのためには、ドラッグ&ドロップ操作によるユーザの操作内容（例えば、ドラッグしたデータや、そのドラッグ先等）と、この操作入力に応じて行われる関数電卓の動作と、の関係性を理解しておく必要がある。

40

【0007】

即ち、グラフ関数電卓の操作を使い慣れている（上記関係性を良く理解している）ユーザにとっては非常に便利な操作入力方法であるが、不慣れな（上記関係性を良く理解していない）ユーザにとっては、“この操作によってどのような動作（処理）がなされるのか”といったことが解りづらいという問題があった。

【0008】

上記問題に鑑み、本発明は、ユーザによる操作入力の簡易性を損なうことなく、当該操作

50

入力に応じてなされる動作を容易に理解することができる電子機器の実現を目的としている。

【0009】

【課題を解決するための手段】

上記課題を解決するために、請求項1記載の発明は、

それぞれ異なる表示形態でデータを表示する複数の表示画面を有する表示部（例えば、図2の表示部40）と、

前記複数の表示画面の内、一の表示画面に対する複写操作がなされた後、他の表示画面に対する貼付操作がなされた場合に、当該貼付操作がなされた他の表示画面の表示形態を判別する判別手段（例えば、図3のCPU10；図9のステップS12）と、

この判別手段によって判別された表示形態に応じた変換内容を説明するメッセージを表示させる制御を行うメッセージ表示制御手段（例えば、図3のCPU10；図9のステップS13）と、

このメッセージ表示制御手段によるメッセージの表示制御が行われた後、前記複写操作によって指定された表示形態のデータを前記判別手段によって判別された表示形態に変換して、前記他の表示画面に表示させる制御を行う変換表示制御手段（例えば、図3のCPU10；図9のステップS14～S16）と、

を備えることを特徴とする電子機器である。

【0010】

また、請求項5に記載の発明は、

それぞれ異なる表示形態でデータを表示する複数の表示画面を有する表示部（例えば、図2の表示部40）に接続される、又は当該表示部を備えるコンピュータに、

前記複数の表示画面の内、一の表示画面に対する複写操作がなされた後、他の表示画面に対する貼付操作がなされた場合に、当該貼付操作がなされた他の表示画面の表示形態を判別する判別機能（例えば、図9のステップS12）と、

この判別手段によって判別された表示形態に応じた変換内容を説明するメッセージを表示させる制御を行うメッセージ表示制御機能（例えば、図9のステップS13）と、

このメッセージ表示制御機能によるメッセージの表示制御機能が行われた後、前記複写操作によって指定された表示形態のデータを前記判別機能によって判別された表示形態に変換して、前記他の表示画面に表示させる制御を行う変換表示制御機能（例えば、図9のステップS14～16）と、

を実現させるためのプログラムである。

【0011】

この請求項1又は5に記載の発明によれば、それぞれ異なる表示形態でデータを表示する複数の表示画面の内、一の表示画面に対する複写操作が行われた後、他の表示画面に対する貼付操作が行われた場合に、貼り付け操作がなされた他の表示画面の表示形態を判別して、この判別した表示形態に応じた変換内容を説明するメッセージを表示制御することができる。その後、複写操作によって指定されたデータを、判別した表示形態に変換して、他の表示画面に表示制御することができる。即ち、複写操作及び貼付操作を行った際に、当該操作による表示制御がなされる前に変換内容を説明するメッセージが表示制御されるため、電子機器の操作の簡易性を損なうことなく、電子機器によってなされる動作を容易に理解することができる。

尚ここで、電子機器とは、例えばグラフ関数電卓のような図形表示制御装置であっても良い。

【0012】

請求項2に記載の発明は、

第1表示画面、及び、異なる複数の表示形態でデータ表示が可能な第2表示画面、を有する表示部（例えば、図2の表示部40）と、

前記第1表示画面に対する複写操作がなされた後、前記第2表示画面に対する貼付操作がなされた場合に、当該第2表示画面に表示可能な複数の表示形態を判別する判別手段（例

10

20

30

40

50

えば、図3のCPU10；図30のステップS32）と、
この判別手段によって判別された複数の表示形態を候補として表示させる制御を行う候補表示制御手段（例えば、図3のCPU10；図30のステップS33）と、
この候補表示制御手段によって表示制御された複数の表示形態の候補の内から、何れか一つを選択する選択手段（例えば、図3のCPU10；図30のステップS34）と、
この選択手段によって選択された表示形態に応じた変換内容を説明するメッセージを表示させる制御を行うメッセージ表示制御手段（例えば、図3のCPU10；図30のステップS35）と、
このメッセージ表示制御手段によるメッセージの表示制御が行われた後、前記複写操作によって指定されたデータを前記選択手段によって選択された表示形態に変換して、前記第2表示画面に表示させる制御を行う変換表示制御手段（例えば、図3のCPU10；図30のステップS36～S38）と、
を備えることを特徴とする電子機器である。

10

【0013】

また、請求項6に記載の発明は、
第1表示画面、及び、異なる複数の表示形態でデータ表示が可能な第2表示画面、を有する表示部に接続される、又は当該表示部を備えるコンピュータに、
前記第1表示画面に対する複写操作がなされた後、前記第2表示画面に対する貼付操作がなされた場合に、当該第2表示画面に表示可能な複数の表示形態を判別する判別機能（例えば、図30のステップS32）と、
この判別機能によって判別された複数の表示形態を候補として表示させる制御を行う候補表示制御機能（例えば、図30のステップS33）と、
この候補表示制御機能によって表示制御された複数の表示形態の候補の内から、何れか一つを選択する選択機能（例えば、図30のステップS34）と、
この選択機能によって選択された表示形態に応じた変換内容を説明するメッセージを表示させる制御を行うメッセージ表示制御機能（例えば、図30のステップS35）と、
このメッセージ表示制御機能によるメッセージの表示制御が行われた後、前記複写操作によって指定されたデータを前記選択機能によって選択された表示形態に変換して、前記第2表示画面に表示させる制御を行う変換表示制御機能（例えば、図30のステップS36～S38）と、
を実現させるためのプログラムである。

20

30

【0014】

この請求項2又は6に記載の発明によれば、第1の表示画面に対する複写操作がなされた後、第2表示画面に対する貼付操作がなされた場合に、第2画面に表示可能な複数の表示形態を判別して、この判別した複数の表示形態を候補として表示制御することができる。そして、これらの候補の内から何れか一つが選択されると、選択された表示形態に応じた変換内容を説明するメッセージを表示制御し、その後、複写操作によって指定されたデータを、上記選択された表示形態に変換して、第2表示画面に表示制御することができる。

【0015】

また、請求項3に記載の発明のように、請求項2に記載の電子機器において、
前記候補表示制御手段は、
前記複写操作によって指定されたデータを表示することができる表示形態を、前記判別手段によって判別された複数の表示形態の内から選択する手段と、
この手段によって選択された表示形態を候補として表示させる制御を行う手段と、
を有するように構成してもよい。

40

【0016】

この請求項3に記載の発明によれば、請求項2に記載の発明と同様の効果を奏するとともに、複写操作によって指定されたデータを表示することができる表示形態を、判別された複数の表示形態の内から自動的に選択して表示制御することができる。換言すると、複写操作によって指定されたデータを表示不可能な表示形態は、候補として表示されることは

50

ない。従って、操作に不慣れなユーザにとっても、所望の操作入力を間違えることなく、確実に行うことができる。

【0017】

また、請求項4に記載の発明のように、請求項1～3の何れか一項に記載の電子機器において、

前記表示部は、変換内容表示画面を更に有し、

前記メッセージ表示制御手段は、前記メッセージを前記変換内容表示画面に表示させる制御を行う手段を有するように構成しても良い。

【0018】

この請求項4に記載の発明によれば、請求項1～3の何れか一項に記載の発明と同様の効果を奏するとともに、複写操作によって指定されたデータに対する変換内容を説明するメッセージを、表示部が有する変換内容表示画面に表示制御することができる。

10

【0019】

【発明の実施の形態】

以下、図面を参照して、本発明の実施の形態を説明する。尚、以下においては、本発明をグラフ関数電卓に適用した場合について説明するが、本発明の適用はこれに限られない。

【0020】

図1は、本発明を適用したグラフ関数電卓（以下、単に「関数電卓」という。）1の外観例を示す図である。同図によれば、関数電卓1は、ディスプレイ2、キー群3、入力ペン4、を備えて構成されている。キー群3を構成するキーには、それぞれ固有の機能が割り当てられており、ユーザは、これらのキーを押下して関数電卓1を操作する。また、ディスプレイ2には、後述するタブレット（タッチパネル）30が一体的に構成されており、ユーザは、入力ペン4を使用したディスプレイ2上のタッチ操作により、関数電卓1を操作することも可能である。

20

【0021】

また、関数電卓1は、計算機能（計算機能付きワープロを含む）を実現するための計算アプリケーションプログラム（以下、適宜「計算アプリケーション」という。）、幾何図形描画機能を実現するための幾何アプリケーションプログラム（以下、適宜「幾何アプリケーション」という。）、グラフ描画機能を実現するためのグラフアプリケーションプログラム（以下、適宜「グラフアプリケーション」という。）、数表（テーブル）機能を実現するためのテーブルアプリケーションプログラム（以下、適宜「テーブルアプリケーション」という。）、統計演算機能を実現するための統計アプリケーションプログラム（以下、適宜「統計アプリケーション」という。）、を搭載している。

30

以下、この関数電卓1の2つの実施の形態について、順に説明する。

【0022】

[第1の実施の形態]

先ず、図2～図26を参照して、第1の実施の形態について説明する。

本第1の実施の形態において、関数電卓1は、データの表示形態がそれぞれ異なる2つのウィンドウ（画面）をディスプレイ2に表示する。そして、一方のウィンドウに対するコピー操作（複写操作）が行われた後、他方のウィンドウに対するペースト操作（貼付操作）が行われると、ペースト操作が行われたウィンドウの表示形態に応じて、コピー操作によって指定されたデータに対する変換内容を説明する動作説明メッセージを表示する。その後、コピー操作によって指定されたデータを、ペースト操作が行われた他方のウィンドウの表示形態に変換して、表示する。

40

【0023】

ここで、コピー操作とは、画面上の所望のデータ（オブジェクトや文字列、グラフ式、テーブル、リスト等）を、入力ペン4やマウスといったポインティングデバイス等で指定して、コピーコマンドを実行する操作である。また、ペースト操作とは、上記コピー操作の後、データの貼り付け位置を指定して、ペーストコマンドを実行する操作である。そして、この一連の操作（コピー操作及びペースト操作）を、コピー&ペースト操作と称する。

50

【0024】

本実施の形態においては、上記コピー&ペースト操作を、入力ペン4を用いたドラッグ&ドロップ操作によって実現する。ドラッグ操作とは、入力ペン4を画面に当接させたまま、当該入力ペン4を画面上で摺動させる操作である。また、ドロップ操作とは、ドラッグ操作の後、当接させている入力ペン4を画面から離す操作である。そして、ドラッグ&ドロップ操作とは、この一連の操作を称したものである。

【0025】

即ち、コピー操作は、入力ペン4を用いて画面上の所望のデータを指定した後、当該指定したデータをドラッグすることによって実現される。また、ペースト操作は、当該ドラッグしたデータを、ドロップ操作によって所望の画面上に移動させることによって実現される。そして、ここでのドロップ位置が、ペースト操作によって指定された貼り付け位置に該当する。

10

【0026】

尚、コピー操作及びペースト操作は、ドラッグ&ドロップ操作ではなく、他の操作によって実現することとしても良い。例えば、コピー元の画面でコピーメニューを選択し、コピー先の画面でペーストメニューを選択することとしても良い。

【0027】

また、関数電卓1が表示可能なウィンドウには種々有るが、代表的なものとして、計算ウィンドウCW、幾何ウィンドウGW、グラフウィンドウRW1、グラフ式ウィンドウRW2、テーブルウィンドウTW、統計グラフウィンドウSW、リストウィンドウ、等がある。

20

【0028】

計算ウィンドウCWは、計算アプリケーションの実行によって表示されるウィンドウであり、データの表示形態を“文字列”として表示する。

【0029】

幾何ウィンドウGWは、幾何アプリケーションの実行によって表示されるウィンドウであり、データの表示形態を“幾何図形”として表示する。ここで、幾何図形とは、点、線（線分、直線を含む）、ベクトル、円（円弧を含む）、多角形、関数グラフ等の線図として表される描画物である。そして、これらの描画（表示）された幾何図形の単位を「オブジェクト」という。例えば、円の幾何図形が描画されている場合には、その描画されている線図（円）のことを「円オブジェクト」という。また、円の幾何図形と直線の幾何図形とが描画されている場合には、円に係る線図の部分を「円オブジェクト」といい、直線に係る線図の部分を「直線オブジェクト」という。

30

【0030】

グラフウィンドウRW1とグラフ式ウィンドウRW2とは、グラフアプリケーションの実行によって表示されるウィンドウである。グラフアプリケーションの実行によって、グラフを描画するためのグラフ表示画面と、描画するグラフのグラフ式を入力するためのグラフ式入力画面が表示されるが、グラフウィンドウRW1が前者の画面を表示し、グラフ式ウィンドウRW2が後者の画面を表示する。グラフウィンドウRW1では、データの表示形態を“グラフ”として表示し、グラフ式ウィンドウRW2では、データの表示形態を“グラフ式”として表示する。ここで、グラフは、グラフウィンドウRW1上に設定された座標レンジ（X軸、Y軸）に従って、1表示ドット毎のX座標値をグラフ式に代入し、Y座標値を算出してグラフ化した表示形態であり、即ち、ビットマップデータが表示されたものである。

40

【0031】

テーブルウィンドウTWは、テーブルアプリケーションの実行によって表示されるウィンドウであり、データの表示形態を“テーブル（数表）”として表示する。

【0032】

統計グラフウィンドウSWとリストウィンドウとは、統計アプリケーションの実行によって表示されるウィンドウである。統計グラフウィンドウSWでは、データの表示形態を“

50

リスト”として表示し、リストウィンドウでは、データの表示形態を“統計グラフ”として表示する。ここで、統計グラフは、グラフウィンドウRWの“グラフ”と同様に、ビットマップデータが表示されたものである。

【0033】

図2は、ドラッグ&ドロップ操作によるデータの表示形態の変換に際して表示される動作説明メッセージを示す図である。尚、同図においては、説明の簡明のため、全てのウィンドウ間に対するデータの表示形態の変換については図示していない。

【0034】

同図によれば、幾何ウィンドウGWから計算ウィンドウCWへのデータの表示形態の変換の際には(1)、「{幾何種別}に対応する{文字列種別}を作成します」といった動作説明メッセージMS1が表示される。また、計算ウィンドウCWから幾何ウィンドウGWへのデータの表示形態の変換の際には(2)、「{文字列種別}に対応する{幾何種別}を描画します」といった動作説明メッセージMS2が表示される。ここで、{幾何種別}には、該当するオブジェクトの種別が入り、また、{文字列種別}には、該当する文字列が示すデータの種別が入る。詳細には、図5、図6を参照して後述する。

【0035】

グラフ式ウィンドウRW2のグラフ式入力画面からグラフウィンドウRW1のグラフ表示画面へのデータの表示形態の変換の際には(3)、10「グラフ式に対応するグラフを描画します」といった動作説明メッセージMS3が表示される。また、グラフウィンドウRW1のグラフ表示画面からグラフ式ウィンドウRW2のグラフ式入力画面へのデータの表示形態の変換の際には(4)、20「グラフに対応するグラフ式を作成します」といった動作説明メッセージMS4が表示される。

【0036】

テーブルウィンドウTWから幾何ウィンドウGWへのデータの表示形態の変換の際には(5)、30「テーブルに対応する{幾何種別}を描画します」といった動作説明メッセージMS5が表示される。また、幾何ウィンドウGWからテーブルウィンドウTWへのデータの表示形態の変換の際には(6)、40「{幾何種別}に対応するテーブルを作成します」といった動作説明メッセージMS6が表示される。ここで、{幾何種別}には、該当するオブジェクトの種別が入る。

【0037】

テーブルウィンドウTWからグラフウィンドウRW1のグラフ表示画面へのデータの表示形態の変換の際には(7)、30「テーブルに対応するグラフを描画します」といった動作説明メッセージMS7が表示される。また、グラフウィンドウRW1のグラフ表示画面からテーブルウィンドウTWへのデータの表示形態の変換の際には(8)、40「グラフに対応するテーブルを作成します」といった動作説明メッセージMS8が表示される。詳細には、図7、図8を参照して後述する。

【0038】

このように、ウィンドウ間のデータのデータの表示形態の変換の際には、双方のウィンドウの種類に応じた変換内容を説明するメッセージ(動作説明メッセージ)MSが表示される。40

【0039】

〔内部構成〕

図3は、第1の実施の形態における関数電卓1の内部構成の一例を示すブロック図である。同図によれば、関数電卓1は、CPU10、入力部20、タブレット30、位置検出回路31、表示部40、表示駆動回路41、RAM50、ROM60、を備えて構成される。

【0040】

CPU10は、入力される指示に応じて所定のプログラムに基づいた処理を実行して、各機能部への指示やデータの転送等を行い、関数電卓1を統括的に制御する。具体的には、CPU10は、入力部20又はタブレット30から入力される操作信号に応じてROM650

0に格納されたプログラムを読み出し、当該プログラムに従って処理を実行する。そして、処理結果をRAM50に保存するとともに、当該処理結果を表示するための表示信号を適宜表示駆動回路41に出力して、対応する表示情報を表示部40に表示させる。

【0041】

入力部20は、数値や数式等の入力、機能選択等に必要なキーを備えた入力装置であり、押下されたキーの押下信号等をCPU10に出力する。この入力部20におけるキー入力により、例えば、各種アプリケーションの起動指示、図形描画処理の実行、数式の入力、演算処理等の実行、処理の終了やモードの解除、各種ポインタやメニュー画面におけるカーソル等の移動、各種選択操作や当該選択操作の確定指示等の入力手段を実現する。この入力部20は、図1に示すキー群3に相当するものである。

10

【0042】

また、関数電卓1は、入力装置として、タッチパネルであるタブレット30を備える。このタブレット30は、表示部40における位置を指示する指示ペン(図1に示す入力ペン4に相当)等の装置と、指示された表示部40の位置を感知する装置とが組み合わされた入力装置である。タブレット30に接続される位置検出回路31は、タブレット30により指示された位置座標を検出する。このタブレット30を使用すれば、表示部40における位置を細かく指定することができ、タブレット30を使用した表示部40のタッチ操作により、上述した入力部20における入力手段を実現することができる。

【0043】

表示駆動回路41は、CPU10から入力される表示信号に基づいて表示部40を制御して各種画面を表示させるものである。表示部40は、LCD(LiquidCrystal Display)やELD(Electronic Luminescent Display)等で構成され、例えば、図10~図26に示す表示画面を表示する。この表示部40は、図1のディスプレイ2に相当するものであり、タブレット30と一体的に形成される。

20

【0044】

RAM50は、CPU10が実行する各種処理プログラムや、これらのプログラムの実行に係るデータ等を一時的に保持する格納領域(メモリ)を備える。特に、本実施の形態においては、幾何ウィンドウGWに描画されるオブジェクトに関するデータである幾何ウィンドウデータ51、計算ウィンドウCWに表示される計算式に関するデータである計算ウィンドウデータ52、グラフウィンドウRWに描画されるグラフに関するデータであるグラフウィンドウデータ53、を保持する格納領域(メモリ)、コピー操作によって指定されたデータを保持するコピーバッファ54、を備えている。また、グラフウィンドウデータ53には、グラフを描画するグラフ式のデータ(グラフ式データ)53aが含まれている。

30

【0045】

図4は、幾何ウィンドウデータ51のデータ構成の一例を示す図である。同図によれば、幾何ウィンドウデータ51は、幾何ウィンドウGW上のオブジェクトを識別する識別ID毎に、当該オブジェクトの種別(幾何種別)と、特定情報と、を対応付けて格納したデータテーブルである。

【0046】

特定情報とは、オブジェクトを幾何ウィンドウGW上で特定するための情報であり、例えば、特定点座標や関数式等である。具体的には、幾何種別が“点”の場合には、当該点が描画されている位置の座標(点座標;第1特定点座標)を、特定情報として保持する。また、幾何種別が“ベクトル”の場合には、当該ベクトルの終点の座標(ベクトル座標;第1特定点座標)を、特定情報として保持する。また、幾何種別が“直線”の場合には、当該直線上の2点の座標(第1、第2特定点座標)を、特定情報として保持する。また、幾何種別が“多角形(n角形)”の場合には、当該n角形の頂点座標に該当するn個の座標(第1~第n特定点座標)を、特定情報として保持する。また、幾何種別が“円”の場合には、当該円の中心の座標(第1特定点座標)と、円周上の1点の座標(第2特定点座標)とを、特定情報として保持する。また、幾何種別が“楕円”の場合には、当該楕円の中

40

50

心の座標（第1特定点座標）と、短半径方向の円周上の座標（第2特定点座標）と、長半径方向の円周上の座標（第3特定点座標）とを、特定情報として保持する。また、幾何種別が“関数グラフ”の場合には、該当する関数式を、特定情報として保持する。

【0047】

ROM60には、各種初期設定やハードウェアの検査、必要なプログラムのロードを行うための初期プログラムが格納される。CPU10は、関数電卓1の電源投入時においてこの初期プログラムを実行することにより、関数電卓1の動作環境を設定する。

【0048】

また、ROM60には、計算アプリケーションや幾何アプリケーション、グラフアプリケーション等の各種アプリケーションプログラム、各種設定処理や各種演算処理等の関数電卓1の動作に係る各種処理プログラム、関数電卓1の備える種々の機能を実現するためのプログラムやデータ等が格納される。特に、本実施の形態においては、幾何/計算メッセージテーブル61、計算/幾何メッセージテーブル62、テーブル/グラフメッセージテーブル63、グラフ/テーブルメッセージテーブル64、動作説明メッセージ表示プログラム65、が格納されている。

10

【0049】

幾何/計算メッセージテーブル61は、幾何ウィンドウGWから計算ウィンドウCWへのドラッグ&ドロップ操作がなされた際に表示される動作説明メッセージを管理するためのテーブルであり、図5に、そのデータ構成の一例を示す。

【0050】

図5は、幾何/計算メッセージテーブル61のデータ構成の一例を示す図である。同図によれば、幾何/計算メッセージテーブル61には、幾何ウィンドウGW上のオブジェクトの種別（幾何種別）毎に、動作説明メッセージを対応付けたデータが格納されている。幾何ウィンドウGWから計算ウィンドウCWへのドラッグ&ドロップ操作がなされると、CPU10は、この幾何/計算メッセージテーブル61を参照して、指定されたオブジェクトの種別に対応する動作説明メッセージを選択して表示させる。

20

【0051】

計算/幾何メッセージテーブル62は、計算ウィンドウCWから幾何ウィンドウGWへのドラッグ&ドロップ操作がなされた際に表示される動作説明メッセージを管理するためのテーブルであり、図6に、そのデータ構成の一例を示す。

30

【0052】

図6は、計算/幾何メッセージテーブル62のデータ構成の一例を示す図である。同図によれば、計算/幾何メッセージテーブル62には、計算ウィンドウCW上の文字列のデータの種別（データ種別）毎に、動作説明メッセージを対応付けたデータが格納されている。計算ウィンドウCWから幾何ウィンドウGWへのドラッグ&ドロップ操作がなされると、CPU10は、この計算/幾何メッセージテーブル62を参照して、指定された文字列のデータ種別に対応する動作説明メッセージを選択して表示させる。

【0053】

テーブル/グラフメッセージテーブル63は、テーブルウィンドウTWからグラフウィンドウRW又はグラフ式ウィンドウRWへのドラッグ&ドロップ操作がなされた際に表示される動作説明メッセージを管理するためのテーブルであり、図7に、そのデータ構成の一例を示す。

40

【0054】

図7は、テーブル/グラフメッセージテーブル63のデータ構成の一例を示す図である。同図によれば、テーブル/グラフメッセージテーブル63には、テーブルウィンドウTW上のデータの種別（データ種別）毎に、動作説明メッセージを対応付けたデータが格納される。テーブルウィンドウTWからグラフウィンドウRW又はグラフ式ウィンドウRWへのドラッグ&ドロップ操作がなされると、CPU10は、このテーブル/グラフメッセージテーブル63を参照して、指定されたデータの種別に対応した動作説明メッセージを選択して表示させる。

50

【 0 0 5 5 】

グラフ/テーブルメッセージテーブル 6 4 は、グラフウィンドウ R W 又はグラフ式ウィンドウ R W からテーブルウィンドウ T W のドラッグ&ドロップ操作がなされた際に表示される動作説明メッセージを管理するためのテーブルであり、図 8 に、そのデータ構成の一例を示す。

【 0 0 5 6 】

図 8 は、グラフ/テーブルメッセージテーブル 6 4 のデータ構成の一例を示す図である。同図によれば、グラフ/テーブルメッセージテーブル 6 4 には、グラフウィンドウ R W 上のデータの種別（データ種別）毎に、動作説明メッセージを対応付けたデータを格納している。グラフウィンドウ R W 又はグラフ式ウィンドウ R W からテーブルウィンドウ T W へのドラッグ&ドロップ操作がなされると、C P U 1 0 は、このグラフ/テーブルメッセージテーブル 6 4 を参照して、指定されたデータの種別に対応する動作説明メッセージを選択して表示させる。

10

【 0 0 5 7 】

動作説明メッセージ表示プログラム 6 5 は、本第 1 の実施の形態における動作説明メッセージ表示処理（図 9 参照）を実現するためのプログラムであり、C P U 1 0 がこの動作説明メッセージ表示プログラム 6 5 を実行することで、動作説明メッセージ表示処理が実現される。

【 0 0 5 8 】

〔動作〕

次に、本第 1 の実施の形態における動作を説明する。

図 9 は、動作説明メッセージ表示処理に係る関数電卓 1 を説明するためのフローチャートであり、図 1 0 ~ 図 2 6 は、それぞれ、図 9 のフローチャートに従った動作 1 ~ 1 7 において、表示部 4 0 に表示される表示画面の遷移例を示す図である。

20

【 0 0 5 9 】

（動作 1）

まず、動作 1 を説明する。

動作 1 は、幾何ウィンドウ G W の点オブジェクトを、計算ウィンドウ C W にドラッグ&ドロップする際の動作である。

【 0 0 6 0 】

図 9 において、まず C P U 1 0 は、計算アプリケーションの実行を開始して、計算ウィンドウ C W 1 0 を表示部 4 0 に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウ G W 1 0 を表示部 4 0 に表示させる。そして、ユーザは、例えば入力ペン 4 を用いることで点描画コマンドを指定し、幾何ウィンドウ G W 1 0 上の所望の位置をタッチする。すると、C P U 1 0 は、タッチされた位置に点オブジェクトを描画するとともに、当該描画した点オブジェクトに該当するデータを、幾何ウィンドウデータ 5 1 に追加格納する。

30

【 0 0 6 1 】

図 1 0 (a) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウ C W 1 0、幾何ウィンドウ G W 1 0、が表示されている。また、幾何ウィンドウ G W 1 0 には、点オブジェクト 1 1 0 が描画されている。このとき、幾何ウィンドウデータ 5 1 には、点オブジェクト 1 1 0 に割り当てられた識別 I D に、幾何種別“点”と、点オブジェクト 1 1 0 の点座標 A (- 3、3) と、を対応付けたデータが格納されている。この表示画面において、ユーザは、入力ペン 4 を用いて点オブジェクト 1 1 0 を指定して、ドラッグ&ドロップ操作によって、計算ウィンドウ C W 1 0 へ移動させる。

40

【 0 0 6 2 】

ドラッグ&ドロップ操作を検知すると（ステップ S 1 1 : Y E S ）、C P U 1 0 は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“幾何

50

ウィンドウ”であり、ドラッグされたデータは点オブジェクト110であり、ドロップ先のウィンドウは“計算ウィンドウ”である、と判断する(ステップS12)。

【0063】

次いで、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、幾何/計算メッセージテーブル61を参照して、点オブジェクト110の幾何種別“点”に対応する動作説明メッセージ“点に対応する点座標を作成します”を表示させる(ステップS13)。

【0064】

図10(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「点に対応する点座標を作成します」という動作説明メッセージが、ポップアップウィンドウPW10に表示されている。

10

【0065】

続いて、CPU10は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、幾何ウィンドウデータ51から、点オブジェクト110の特定点座標(点座標)を読み出し、読み出した座標(-3, 3)を表す1×2行列の文字列を生成して、コピーバッファ54に格納する(ステップS14)。

【0066】

その後、CPU10は、ポップアップウィンドウPW10の表示を終了し(ステップS15)、コピーバッファ54に格納されている文字列を、計算ウィンドウCW10のカーソル位置に表示させる(ステップS16)。

20

【0067】

図10(c)は、この段階での表示画面の一例を示す図である。同図によれば、計算ウィンドウCW10には、点オブジェクト110の点座標(-3, 3)を表す1×2行列が表示されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0068】

(動作2)

次に、動作2を説明する。

動作2は、計算ウィンドウCW12の点座標を、幾何ウィンドウGW12にドラッグ&ドロップする際の動作である。

30

【0069】

図9において、先ずCPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW12を表示部40に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウGW12を表示部40に表示させる。そして、ユーザは、例えば入力ペン4を用いることで、計算ウィンドウCW12のカーソル位置に点座標を表す1×2行列を入力する。

【0070】

図11(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウCW12、幾何ウィンドウGW12、が表示されている。また、計算ウィンドウCW12には、点座標(2, 2)を表す1×2行列が表示されている。

40

【0071】

この表示画面において、ユーザは、入力ペン4を用いて、計算ウィンドウCW12の点座標(2, 2)を表す1×2行列を含む文字列表示領域T10を指定して反転表示させ、ドラッグ&ドロップ操作によって、幾何ウィンドウGW12へ移動させる。

【0072】

ドラッグ&ドロップ操作を検知すると(ステップS11: YES)、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“計算ウィンドウ”であり、ドラッグされたデータは点座標を表す1×2行列であり、ドロップ先のウィンドウは“幾何ウィンドウ”である、と判断する(ステップS12)。

50

【0073】

次いで、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、計算/幾何メッセージテーブル62を参照して、データ種別“点座標”に対応するメッセージ“点座標に対応する点を描画します”を表示させる(ステップS13)。

【0074】

図11(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「点座標に対応する点を描画します」という動作説明メッセージが、ポップアップウィンドウPW12に表示されている。

【0075】

続いて、CPU10は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、点座標(2,2)を表す1×2行列を点座標(2,2)に変換して、コピーバッファ54に格納する(ステップS14)。

【0076】

その後、CPU10は、ポップアップウィンドウPW12の表示を終了し(ステップS15)、コピーバッファ54に格納されている点座標(2,2)に基づいて、幾何ウィンドウGW12に点オブジェクト112を描画する。それとともに、当該点オブジェクト112に割り当てた識別IDに、幾何種別“点”と、点オブジェクト112を描画した位置の座標(2,2)と、を対応付けたデータを、幾何ウィンドウデータ51に追加格納する(ステップS16)。

【0077】

図11(c)は、この段階での表示画面の一例を示す図である。同図によれば、幾何ウィンドウGW12には、座標(2,2)の位置に点オブジェクト112が描画されている。以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0078】

(動作3)

次に、動作3を説明する。

動作3は、幾何ウィンドウGW14のベクトルオブジェクトを、計算ウィンドウCW14にドラッグ&ドロップする際の動作である。

【0079】

図9において、先ずCPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW14を表示部40に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウGW14を表示部40に表示させる。そして、ユーザは、例えば入力ペン4を用いることでベクトル描画コマンドを指定し、幾何ウィンドウGW14上の所望の位置をタッチする。すると、CPU10は、幾何ウィンドウGW14の原点を始点に、タッチされた位置を終点とするベクトルオブジェクトを描画するとともに、当該描画したベクトルオブジェクトに該当するデータを、幾何ウィンドウデータ51に追加格納する。

【0080】

図12(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウCW14、幾何ウィンドウGW14、が表示されている。また、幾何ウィンドウGW14には、ベクトルオブジェクト120が表示されている。このとき、幾何ウィンドウデータ51には、ベクトルオブジェクト120に割り当てられた識別IDに、幾何種別“ベクトル”と、ベクトルオブジェクト120のベクトル座標(4,2)と、を対応付けたデータが格納されている。

【0081】

この表示画面において、ユーザは、入力ペン4を用いてベクトルオブジェクト120を指定して、ドラッグ&ドロップ操作によって、計算ウィンドウCW14へ移動させる。

【0082】

ドラッグ&ドロップ操作を検知すると(ステップS11:YES)、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウ

10

20

30

40

50

ウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“幾何ウィンドウ”であり、ドラッグされたデータはベクトルオブジェクト120であり、ドロップ先のウィンドウは“計算ウィンドウ”である、と判断する(ステップS12)。

【0083】

続いて、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、幾何/計算メッセージテーブル61を参照して、ベクトルオブジェクト120の幾何種別“ベクトル”に対応する動作説明メッセージ“ベクトルに対応するベクトル座標を作成します”を表示させる(ステップS13)。

【0084】

図12(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「ベクトルに対応するベクトル座標を作成します」という動作説明メッセージが、ポップアップウィンドウPW14に表示されている。 10

【0085】

続いて、CPU10は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、幾何ウィンドウデータ51から、ベクトルオブジェクト120の特定点座標(ベクトル座標)を読み出し、読み出したベクトル座標(4、2)を表す1×2行列の文字列を生成して、コピーバッファ54に格納する(ステップS14)。

【0086】

その後、CPU10は、ポップアップウィンドウPW14の表示を終了し(ステップS15)、コピーバッファ54に格納されている文字列を、計算ウィンドウCW14のカーソル位置に表示させる(ステップS16)。 20

【0087】

図12(c)は、この段階での表示画面の一例を示す図である。同図によれば、計算ウィンドウCW14には、ベクトルオブジェクト120のベクトル座標(4、2)を表す1×2行列が表示されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0088】

(動作4)

次に、動作4を説明する。

動作4は、計算ウィンドウCW16のベクトル座標を、幾何ウィンドウGW16にドラッグ&ドロップする際の動作である。 30

【0089】

図9において、先ずCPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW16を表示部40に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウGW16を表示部40に表示させる。そして、ユーザは、例えば入力ペン4を用いることで、計算ウィンドウCW16のカーソル位置にベクトル座標を表す1×2行列を入力する。

【0090】

図13(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウCW16、幾何ウィンドウGW16、が表示されている。また、計算ウィンドウCW16には、ベクトル座標(-4、2)を表す1×2行列が表示されている。 40

【0091】

この表示画面において、ユーザは、入力ペン4を用いて、計算ウィンドウCW16のベクトル座標(-4、2)を表す1×2行列を含む文字列表示領域T20を指定して反転表示させ、ドラッグ&ドロップ操作によって、幾何ウィンドウGW16へ移動させる。

【0092】

ドラッグ&ドロップ操作を検知すると(ステップS11: YES)、CPU10は、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“計算ウィンドウ”であり 50

、ドラッグされたデータはベクトル座標を表す 1×2 行列であり、ドロップ先のウィンドウは“幾何ウィンドウ”である、と判断する（ステップ S 1 2）。

【0093】

続いて、CPU 10 は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、計算/幾何メッセージテーブル 6 2 を参照して、データ種別“ベクトル座標”に対応するメッセージ“ベクトル座標に対応するベクトルを描画します”を表示させる（ステップ S 1 3）。

【0094】

図 1 3 (b) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「ベクトル座標に対応するベクトルを描画します」という動作メッセージが、ポップアップウィンドウ PW 1 6 に表示されている。 10

【0095】

続いて、CPU 10 は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、ベクトル座標 (- 4 , 2) を表す 1×2 行列をベクトル座標 (- 4 , 2) に変換して、コピーバッファ 5 4 に格納する（ステップ S 1 4）。

【0096】

その後、CPU 10 は、ポップアップウィンドウ PW 1 6 の表示を終了し（ステップ S 1 5）、コピーバッファ 5 4 に格納されているベクトル座標 (- 4 , 2) に基づいて、幾何ウィンドウ GW 1 6 にベクトルオブジェクト 1 2 2 を描画する。それとともに、当該ベクトルオブジェクト 1 2 2 に割り当てた識別 ID に、幾何種別“ベクトル”と、ベクトル座標 (- 4 , 2) と、を対応付けたデータを、幾何ウィンドウデータ 5 1 に追加格納する（ステップ S 1 6）。 20

【0097】

図 1 3 (c) は、この段階での表示画面の一例を示す図である。同図によれば、幾何ウィンドウ GW 1 6 には、ベクトル座標 (- 4 , 2) に対応するベクトルオブジェクト 1 2 2 が描画されている。

以上の処理を行うと、CPU 10 は、本動作説明メッセージ表示処理を終了する。

【0098】

（動作 5）

次に、動作 5 を説明する。 30

動作 5 は、幾何ウィンドウ GW 1 8 の四角形オブジェクトを、計算ウィンドウ CW 1 8 にドラッグ&ドロップする際の動作である。

【0099】

図 9 において、まず CPU 10 は、計算アプリケーションの実行を開始して、計算ウィンドウ CW 1 8 を表示部 4 0 に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウ GW 1 8 を表示部 4 0 に表示させる。そして、ユーザは、例えば入力ペン 4 を用いることで四角形描画コマンドを指定して、幾何ウィンドウ GW 1 8 上の所望の 4 点をタッチする。すると、CPU 10 は、タッチされた 4 点の位置を頂点座標とする四角形オブジェクトを描画するとともに、当該描画した四角形オブジェクトに該当するデータを、幾何ウィンドウデータ 5 1 に追加格納する。 40

【0100】

図 1 4 (a) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウ CW 1 8、幾何ウィンドウ GW 1 8、が表示されている。また、幾何ウィンドウ GW 1 8 には、四角形オブジェクト 1 3 0 が描画されている。このとき、幾何ウィンドウデータ 5 1 には、四角形オブジェクト 1 3 0 に割り当てられた識別 ID に、幾何種別“ n 角形 ($n = 4$)”と、四角形オブジェクト 1 3 0 の 4 つの頂点座標 A (0 , 0)、B (1 , 2)、C (5 , 3)、D (4 , 1) と、を対応付けたデータが格納されている。

【0101】

この表示画面において、ユーザは、入力ペンを用いて四角形オブジェクト 1 3 0 を指定し 50

て、ドラッグ&ドロップ操作によって、計算ウィンドウCW18へ移動させる。

【0102】

ドラッグ&ドロップ操作を検知すると(ステップS11: YES)、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“幾何ウィンドウ”であり、ドラッグされたデータは四角形オブジェクト130であり、ドロップ先のウィンドウは“計算ウィンドウ”である、と判断する(ステップS12)。

【0103】

続いて、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、幾何/計算メッセージテーブル61を参照して、四角形オブジェクト130の幾何種別“n角形(n=4)”に対応する動作説明メッセージ“四角形の頂点座標の行列を作成します”を表示させる(ステップS13)。

【0104】

図14(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「四角形の頂点座標の行列を作成します」という動作説明メッセージが、ポップアップウィンドウPW18に表示されている。

【0105】

続いて、CPU10は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、幾何ウィンドウデータ51から、四角形オブジェクト130の特定座標(頂点座標)を読み出し、読み出した4つの頂点座標(0、0)、(1、2)、(5、3)、(4、1)を表す4×2行列の文字列を生成して、コピーバッファ54に格納する(ステップS14)。

【0106】

その後、CPU10は、ポップアップウィンドウPW18の表示を終了し(ステップS15)、コピーバッファ54に格納されている文字列を、計算ウィンドウCW18のカーソル位置に表示させる(ステップS16)。

【0107】

図14(c)は、この段階での表示画面の一例を示す図である。同図によれば、計算ウィンドウCW18には、四角形オブジェクト130の4つの頂点座標(0、0)、(1、2)、(5、3)、(4、1)を表す4×2行列が表示されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0108】

(動作6)

次に、動作6を説明する。

動作6は、計算ウィンドウCW20の四角形の頂点座標を、幾何ウィンドウGW20にドラッグ&ドロップする際の動作である。

【0109】

図9において、先ずCPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW20を表示部40に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウGW20を表示部40に表示させる。そして、ユーザは、例えば入力ペン4を用いることで、計算ウィンドウCW20のカーソル位置に四角形の頂点座標を表す4×2行列を入力する。

【0110】

図15(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウCW20、幾何ウィンドウGW20、が表示されている。また計算ウィンドウCW20には、四角形の4つの頂点座標(0、0)、(-2、1)、(-3、5)、(-1、4)を表す4×2行列が表示されている。

【0111】

この表示画面において、ユーザは、入力ペン4を用いて、計算ウィンドウCW20の四角形の頂点座標(0、0)、(-2、1)、(-3、5)、(-1、4)を表す4×2行列

を含む文字列表示領域 T 3 0 を指定して反転表示させ、ドラッグ & ドロップ操作によって、幾何ウィンドウ G W 2 0 へ移動させる。

【 0 1 1 2 】

ドラッグ & ドロップ操作を検知すると (ステップ S 1 1 : Y E S)、C P U 1 0 は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは “ 計算ウィンドウ ” であり、ドラッグされたデータは四角形の頂点座標を表す 4×2 行列であり、ドロップ先のウィンドウは “ 幾何ウィンドウ ” である、と判断する (ステップ S 1 2)。

【 0 1 1 3 】

次いで、C P U 1 0 は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、計算 / 幾何メッセージテーブル 6 2 を参照して、データ種別 “ n 角形 ($n = 4$) の頂点座標 ” に対応するメッセージ “ 頂点座標に対応する四角形を描画します ” を表示させる (ステップ S 1 3)。

【 0 1 1 4 】

図 1 5 (b) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「頂点座標に対応する四角形を描画します」という動作説明メッセージが、ポップアップウィンドウ P W 2 0 に表示されている。

【 0 1 1 5 】

続いて、C P U 1 0 は、ドラッグされたデータを、操作内容に応じて変換する。即ち、四角形の頂点座標を表す 4×2 行列を 4 つの頂点座標 (0 , 0)、(- 2 , 1)、(- 3 , 5)、(- 1 , 4) に変換して、コピーバッファ 5 4 に格納する (ステップ S 1 4)。

【 0 1 1 6 】

その後、C P U 1 0 は、ポップアップウィンドウ P W 2 0 の表示を終了し (ステップ S 1 5)、コピーバッファ 5 4 に格納されている頂点座標 (0 , 0)、(- 2 , 1)、(- 3 , 5)、(- 1 , 4) に基づいて、幾何ウィンドウ G W 2 0 に四角形オブジェクト 1 3 2 を描画する。それとともに、当該四角形オブジェクト 1 3 2 に割り当てた識別 I D に、幾何種別 “ n 角形 ($n = 4$) ” と、頂点座標 (0 , 0)、(- 2 , 1)、(- 3 , 5)、(- 1 , 4) と、を対応付けたデータを、幾何ウィンドウデータ 5 1 に追加格納する (ステップ S 1 6)。

【 0 1 1 7 】

図 1 5 (c) は、この段階での表示画面の一例を示す図である。同図によれば、幾何ウィンドウ G W 2 0 には、頂点座標 A (0 , 0)、B (- 2 , 1)、C (- 3 , 5)、D (- 1 , 4) に対応する四角形オブジェクト 1 3 2 が描画されている。

以上の処理を行うと、C P U 1 0 は、本動作説明メッセージ表示処理を終了する。

【 0 1 1 8 】

(動作 7)

次に、動作 7 を説明する。

動作 7 は、幾何ウィンドウ G W 2 2 の直線オブジェクトを、計算ウィンドウ C W 2 2 にドラッグ & ドロップする際の動作である。

【 0 1 1 9 】

図 9 において、先ず C P U 1 0 は、計算アプリケーションの実行を開始し、計算ウィンドウ C W 2 2 を表示部 4 0 に表示させるとともに、幾何アプリケーションの実行を開始し、幾何ウィンドウ G W 2 2 を表示部 4 0 に表示させる。そして、ユーザは、例えば入力ペン 4 を用いることで直線描画コマンドを指定し、幾何ウィンドウ G W 2 2 上の所望の 2 点をタッチする。すると、C P U 1 0 は、タッチされた 2 点の位置を結ぶ直線オブジェクトを描画するとともに、当該描画した直線オブジェクトに該当するデータを、幾何ウィンドウデータ 5 1 に追加格納する。

【 0 1 2 0 】

図 1 6 (a) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面

10

20

30

40

50

には、計算ウィンドウCW22、幾何ウィンドウGW22、が表示されている。また、幾何ウィンドウGW22には、直線オブジェクト140が描画されている。このとき、幾何ウィンドウデータ51には、直線オブジェクト140に割り当てられた識別IDに、幾何種別“直線”と、直線オブジェクト140上の2点の座標A(1、-1)、B(-1、-2)と、を対応付けたデータが格納されている。

この表示画面において、ユーザは、入力ペン4を用いて直線オブジェクト140を指定し、ドラッグ&ドロップ操作によって、計算ウィンドウCW22へ移動させる。

【0121】

ドラッグ&ドロップ操作を検知すると(ステップS11: YES)、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“幾何ウィンドウ”であり、ドラッグされたデータは直線オブジェクト140であり、ドロップ先のウィンドウは“計算ウィンドウ”である、と判断する(ステップS12)。

10

【0122】

次いで、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、幾何/計算メッセージテーブル61を参照して、直線オブジェクト140の幾何種別“直線”に対応する動作説明メッセージ“直線に対応する直線式を作成します”を表示させる(ステップS13)。

【0123】

図16(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「直線に対応する直線式を作成します」という動作説明メッセージが、ポップアップウィンドウPW22に表示されている。

20

【0124】

続いて、CPU10は、ドラッグされたデータを、操作内容に基づいて変換する。即ち、幾何ウィンドウデータ51から、直線オブジェクト140の特定点座標を読み出し、読み出した2つの座標(1、-1)、(-1、-2)を結ぶ直線式“ $y = 0.5 * x - 1.5$ ”を表す文字列を生成して、コピーバッファ54に格納する(ステップS14)。

【0125】

その後、CPU10は、ポップアップウィンドウPW22の表示を終了し(ステップS15)、コピーバッファ54に格納されている文字列を、計算ウィンドウCW22のカーソル位置に表示させる(ステップS16)。

30

【0126】

図16(c)は、この段階での表示画面の一例を示す図である。同図によれば、計算ウィンドウCW22には、直線オブジェクト140に対応する直線式“ $y = 0.5 * x - 1.5$ ”が表示されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0127】

(動作8)

次に、動作8を説明する。

動作8は、計算ウィンドウCW24の直線式を、幾何ウィンドウGW24にドラッグ&ドロップする際の動作である。

40

【0128】

図9において、先ずCPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW24を表示部40に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウGW24を表示部40に表示させる。そして、ユーザは、例えば入力ペン4を用いることで、計算ウィンドウCW24のカーソル位置に直線式を入力する。

【0129】

図17(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウCW24、幾何ウィンドウGW24、が表示されている。また、計算ウィンドウCW24には、直線式を表す文字列“ $y = 0.5 * x + 1.5$ ”が表示され

50

ている。

【0130】

この表示画面において、ユーザは、入力ペン4を用いて、計算ウィンドウCW24の文字列「 $y = 0.5 * x + 1.5$ 」を含む文字列表示領域T40を指定して反転表示させ、ドラッグ&ドロップ操作によって、幾何ウィンドウGW24へ移動させる。

【0131】

ドラッグ&ドロップ操作を検知すると(ステップS11: YES)、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“計算ウィンドウ”であり、ドラッグされたデータは直線式“ $y = 0.5 * x + 1.5$ ”を表す文字列であり、ドロップ先のウィンドウは“幾何ウィンドウ”である、と判断する(ステップS12)。

【0132】

次いで、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、計算/幾何メッセージテーブル62を参照して、データ種別“直線式”に対応する動作説明メッセージ“直線式に対応する直線を描画します”を表示させる(ステップS13)。

【0133】

図17(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「直線式に対応する直線を描画します」という動作説明メッセージが、ポップアップウィンドウPW24に表示されている。

【0134】

続いて、CPU10は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、文字列「 $y = 0.5 * x + 1.5$ 」を直線式“ $y = 0.5 * x + 1.5$ ”に変換して、コピーバッファ54に格納する(ステップS14)。

【0135】

その後、CPU10は、ポップアップウィンドウPW24の表示を終了し(ステップS15)、コピーバッファ54に格納されている直線式“ $y = 0.5 * x + 1.5$ ”に基づいて、幾何ウィンドウGW24に直線オブジェクト142を描画する。それとともに、当該直線オブジェクト142に割り当てた識別IDに、幾何種別“直線”と、直線オブジェクト142上の任意の2点の座標(第1、第2特定点座標)と、を対応付けたデータを、幾何ウィンドウデータ51に追加格納する(ステップS16)。

【0136】

図17(c)は、この段階での表示画面の一例を示す図である。同図によれば、幾何ウィンドウGW24には、直線式“ $y = 0.5 * x + 1.5$ ”に対応する直線オブジェクト142が描画されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0137】

(動作9)

次に、動作9を説明する。

動作9は、幾何ウィンドウGW26の円オブジェクトを、計算ウィンドウCW26にドラッグ&ドロップする際の動作である。

【0138】

図9において、先ずCPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW26を表示部40に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウGW26を表示部40に表示させる。そして、ユーザは、例えば入力ペン4を用いることで円描画コマンドを指定し、幾何ウィンドウGW26上の所望の2点をタッチする。すると、CPU10は、タッチされた2点の位置に基づく円オブジェクトを描画するとともに、当該描画した円オブジェクトに該当するデータを、幾何ウィンドウデータ51に追加格納する。

10

20

30

40

50

【0139】

図18(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウCW26、幾何ウィンドウGW26、が表示されている。また、幾何ウィンドウGW26には、円オブジェクト150が描画されている。このとき、幾何ウィンドウデータ51には、円オブジェクト150に割り当てられた識別IDに、幾何種別“円”と、円オブジェクト150を特定するための座標A(0,0)、B(2,0)と、を対応付けたデータが格納されている。

この表示画面において、ユーザは、入力ペン4を用いて円オブジェクト150を指定し、ドラッグ&ドロップ操作によって、計算ウィンドウCW26へ移動させる。

【0140】

ドラッグ&ドロップ操作を検知すると(ステップS11: YES)、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“幾何ウィンドウ”であり、ドラッグされたデータは円オブジェクト150であり、ドロップ先のウィンドウは“計算ウィンドウ”である、と判断する(ステップS12)。

【0141】

次いで、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、幾何/計算メッセージテーブル61を参照して、円オブジェクト150の幾何種別“円”に対応する動作説明メッセージ“円に対応する方程式を作成します”を表示させる(ステップS13)。

【0142】

図18(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「円に対応する方程式を作成します」という動作説明メッセージが、ポップアップウィンドウPW26に表示されている。

【0143】

続いて、CPU10は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、幾何ウィンドウデータ51から、円オブジェクト150の特定座標を読み出し、読み出した2つの座標(0,0)、(2,0)に基づいて、円の方程式“ $x^2 + y^2 - 4 = 0$ ”を表す文字列を生成して、コピーバッファ54に格納する(ステップS14)。

【0144】

その後、CPU10は、ポップアップウィンドウPW26の表示を終了し(ステップS15)、コピーバッファ54に格納されている文字列を、計算ウィンドウCW26のカーソル位置に表示させる(ステップS16)。

【0145】

図18(c)は、この段階での表示画面の一例を示す図である。同図によれば、計算ウィンドウCW26には、円オブジェクト150に対応する円の方程式“ $x^2 + y^2 - 4 = 0$ ”が表示されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0146】

(動作10)

次に、動作10を説明する。

動作10は、計算ウィンドウCW28の円の方程式を、幾何ウィンドウGW28にドラッグ&ドロップする際の動作である。

【0147】

図9において、先ずCPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW28を表示部40に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウGW28を表示部40に表示させる。そして、ユーザは、例えば入力ペン4を用いることで、計算ウィンドウCW28のカーソル位置に円の方程式を入力する。

10

20

30

40

50

【0148】

図19(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウCW28、幾何ウィンドウGW28、が表示されている。また、計算ウィンドウCW28には、円の方程式を表す文字列「 $x^2 + y^2 - 9 = 0$ 」が表示されている。

【0149】

この表示画面において、ユーザは、入力ペン4を用いて、計算ウィンドウCW28の文字列「 $x^2 + y^2 - 9 = 0$ 」を含む文字列表示領域T50を指定して反転表示させ、ドラッグ&ドロップ操作によって、幾何ウィンドウGW28へ移動させる。

【0150】

ドラッグ&ドロップ操作を検知すると(ステップS11: YES)、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“計算ウィンドウ”であり、ドラッグされたデータは円の方程式“ $x^2 + y^2 - 9 = 0$ ”を表す文字列であり、ドロップ先のウィンドウは“幾何ウィンドウ”である、と判断する(ステップS12)。

【0151】

次いで、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、計算/幾何メッセージテーブル62を参照し、データ種別“円の方程式”に対応するメッセージ“円の方程式に対応する円を描画します”を表示させる(ステップS13)。

【0152】

図19(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「円の方程式に対応する円を描画します」という動作説明メッセージが、ポップアップウィンドウPW28に表示されている。

【0153】

続いて、CPU10は、ドラッグされたデータをドロップ先のウィンドウの種類に基づいて変換する。即ち、文字列「 $x^2 + y^2 - 9 = 0$ 」を円の方程式“ $x^2 + y^2 - 9 = 0$ ”に変換して、コピーバッファ54に格納する(ステップS14)。

【0154】

その後、CPU10は、ポップアップウィンドウPW28の表示を終了し(ステップS15)、コピーバッファ54に格納されている円の方程式“ $x^2 + y^2 - 9 = 0$ ”に基づいて、幾何ウィンドウGW28に円オブジェクト152を描画する。それとともに、当該円オブジェクト152に割り当てた識別IDに、幾何種別“円”と、円オブジェクト152を特定する任意の2つの特定点座標と、を対応付けたデータを、幾何ウィンドウデータ51に追加格納する(ステップS16)。

【0155】

図19(c)は、この段階での表示画面の一例を示す図である。同図によれば、幾何ウィンドウGW28には、円の方程式“ $x^2 + y^2 - 9 = 0$ ”に対応する円オブジェクト152が描画されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0156】

(動作11)

次に、動作11を説明する。

動作11は、幾何ウィンドウGW30の楕円オブジェクトを、計算ウィンドウCW30にドラッグ&ドロップする際の動作である。

【0157】

図9において、まずCPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW30を表示部40に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウGW30を表示部40に表示させる。そして、ユーザは、例えば入力

10

20

30

40

50

ペン4を用いることで楕円描画コマンドを指定し、幾何ウィンドウ上の所望の3点をタッチする。すると、CPU10は、タッチされた3点の位置に基づく楕円オブジェクトを描画するとともに、当該描画した楕円オブジェクトに該当するデータを、幾何ウィンドウデータ51に追加格納する。

【0158】

図20(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウCW30、幾何ウィンドウGW30、表示されている。また、幾何ウィンドウGW30には、楕円オブジェクト160が描画されている。このとき、幾何ウィンドウデータ51には、楕円オブジェクト160に割り当てられた識別IDに、幾何種別“楕円”と、楕円オブジェクト160を特定するための座標A(0、0)、B(2、0)、C(0、1)と、を対応付けたデータが格納されている。

10

この表示画面において、ユーザは、入力ペン4を用いて楕円オブジェクト160を選択し、ドラッグ&ドロップ操作によって、計算ウィンドウCW30へ移動させる。

【0159】

ドラッグ&ドロップ操作を検知すると(ステップS11: YES)、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“幾何ウィンドウ”であり、ドラッグされたデータは楕円オブジェクト160であり、ドロップ先のウィンドウは“計算ウィンドウ”である、と判断する(ステップS12)。

【0160】

20

次いで、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、幾何/計算メッセージテーブル61を参照し、楕円オブジェクト160の種別“楕円”に対応する動作説明メッセージ“楕円に対応する方程式を作成します”を表示させる(ステップS13)。

【0161】

図20(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には「楕円に対応する方程式を作成します」という動作説明メッセージが、ポップアップウィンドウPW30に表示されている。

【0162】

続いて、CPU10は、ドラッグされたデータを、操作内容に基づいて変換する。即ち、幾何ウィンドウデータ51から、楕円オブジェクト160の特定点座標を読み出し、読み出した3つの座標(0、0)、(2、1)、(0、1)に基づいて、楕円の方程式“ $x^2/4 + y^2 - 1 = 0$ ”を表す文字列を生成して、コピーバッファ54に格納する(ステップS14)。

30

【0163】

その後、CPU10は、ポップアップウィンドウPW30の表示を終了し(ステップS15)、コピーバッファ54に格納されている文字列を、計算ウィンドウCW30のカーソル位置に表示させる(ステップS16)。

【0164】

図20(c)は、この段階での表示画面の一例を示す図である。同図によれば、計算ウィンドウCW30には、楕円オブジェクト160に対応する楕円の方程式“ $x^2/4 + y^2 - 1 = 0$ ”が表示されている。

40

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0165】

(動作12)

次に、動作12を説明する。

動作12は、計算ウィンドウCW32の楕円の方程式を、幾何ウィンドウGW32にドラッグ&ドロップする際の動作である。

【0166】

図9において、先ずCPU10は、計算アプリケーションの実行を開始して、計算ウィン

50

ドウC W 3 2を表示部4 0に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウG W 3 2を表示部4 0に表示させる。そして、ユーザは、例えば入力ペン4を用いることで、計算ウィンドウC W 3 2のカーソル位置に楕円の方程式を入力する。

【0 1 6 7】

図2 1 (a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウC W 3 2、幾何ウィンドウG W 3 2、が表示されている。また、計算ウィンドウC W 3 2には、楕円の方程式を表す文字列「 $x^2 / 4 + y^2 / 9 - 4 = 0$ 」が表示されている。

【0 1 6 8】

この表示画面において、ユーザは、入力ペン4を用いて、計算ウィンドウC W 3 2の文字列「 $x^2 / 4 + y^2 / 9 - 4 = 0$ 」を含む文字列表示領域T 6 0を指定して反転表示させ、ドラッグ&ドロップ操作によって、幾何ウィンドウG W 3 2へ移動させる。

【0 1 6 9】

ドラッグ&ドロップ操作を検知すると(ステップS 1 1 : Y E S)、C P U 1 0は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“計算ウィンドウ”であり、ドラッグされたデータは楕円の方程式“ $x^2 / 4 + y^2 / 9 - 4 = 0$ ”を表す文字列であり、ドロップ先のウィンドウは“幾何ウィンドウ”である、と判断する(ステップS 1 2)。

【0 1 7 0】

次いで、C P U 1 0は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、計算/幾何メッセージテーブル6 2を参照し、データ種別“楕円の方程式”に対応するメッセージ“楕円の方程式に対応する楕円の描画します”を表示させる(ステップS 1 3)。

【0 1 7 1】

図2 1 (b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「楕円の方程式に対応する楕円を描画します」という動作説明メッセージが、ポップアップウィンドウP W 3 2に表示されている。

【0 1 7 2】

続いて、C P U 1 0は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、文字列「 $x^2 / 4 + y^2 / 9 - 4 = 0$ 」を、楕円の方程式“ $x^2 / 4 + y^2 / 9 - 4 = 0$ ”に変換して、コピーバッファ5 4に格納する(ステップS 1 4)。

【0 1 7 3】

その後、C P U 1 0は、ポップアップウィンドウP W 3 2の表示を終了し(ステップS 1 5)、コピーバッファ5 4に格納されている楕円の方程式“ $x^2 / 4 + y^2 / 9 - 4 = 0$ ”に基づいて、幾何ウィンドウG W 3 2に楕円オブジェクト1 6 2を描画する。それとともに、当該楕円オブジェクト1 6 2に割り当てた識別IDに、幾何種別“楕円”と、楕円オブジェクト1 6 2を特定する3点の座標と、を対応付けたデータを、幾何ウィンドウデータ5 1に追加格納する(ステップS 1 6)。

【0 1 7 4】

図2 1 (c)は、この段階での表示画面の一例を示す図である。同図によれば、幾何ウィンドウG W 3 2には、楕円の方程式“ $x^2 / 4 + y^2 / 9 - 4 = 0$ ”に対応する楕円オブジェクト1 6 2が表示されている。

以上の処理を行うと、C P U 1 0は、本操作説明メッセージ表示処理を終了する。

【0 1 7 5】

(動作1 3)

次に、動作1 3を説明する。

動作1 3は、幾何ウィンドウG W 3 4の関数グラフオブジェクトを、計算ウィンドウC W 3 4にドラッグ&ドロップする際の動作である。

10

20

30

40

50

【0176】

図9において、先ずCPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW34を表示部40に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウGW34を表示部40に表示させる。そして、ユーザは、例えば入力ペン4を用いることで関数グラフ描画コマンドを指定し、更に一般式を指定して、幾何ウィンドウGW34上の所望の数点をタッチする。すると、CPU10は、タッチされた数点（指定された一般式によって異なる）の位置に基づくグラフ関数オブジェクトを描画するとともに、当該描画した関数グラフオブジェクトに該当するデータを、幾何ウィンドウデータ51に追加格納する。

【0177】

図22(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウCW34、幾何ウィンドウGW34、が表示されている。また、幾何ウィンドウGW34には、関数グラフオブジェクト170が描画されている。このとき、幾何ウィンドウデータ51には、関数グラフオブジェクト170に割り当てられた識別IDに、幾何種別“関数グラフ”と、関数グラフオブジェクト170に対応する関数式“ x^2 ”と、を対応付けたデータが格納されている。

この表示画面において、ユーザは、入力ペン4を用いて関数グラフオブジェクト170を指定し、ドラッグ&ドロップ操作によって、幾何ウィンドウGW34へ移動させる。

【0178】

ドラッグ&ドロップ操作を検知すると（ステップS11：YES）、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“幾何ウィンドウ”であり、ドラッグされたデータは関数グラフオブジェクト170であり、ドロップ先のウィンドウは“計算ウィンドウ”である、と判断する（ステップS12）。

【0179】

続いて、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、幾何/計算メッセージテーブル61を参照し、関数グラフオブジェクト170の幾何種別“関数グラフ”に対応する動作説明メッセージ“関数グラフに対応する関数式を作成します”を表示させる（ステップS13）。

【0180】

図22(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「関数グラフに対応する関数式を作成します」という動作説明メッセージが、ポップアップウィンドウPW34に表示されている。

【0181】

続いて、CPU10は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、幾何ウィンドウデータ51から、関数グラフオブジェクト170の関数式“ x^2 ”を読み出し、読み出した関数式“ x^2 ”を表す文字列を生成して、コピーバッファ54に格納する（ステップS14）。

【0182】

その後、CPU10は、ポップアップウィンドウPW34の表示を終了し（ステップS15）、コピーバッファ54に格納されている文字列を、計算ウィンドウCW34のカーソル位置に表示させる（ステップS16）。

【0183】

図22(c)は、この段階での表示画面の一例を示す図である。同図によれば、計算ウィンドウCW34には、関数グラフオブジェクト170を表す関数式「 x^2 」が表示されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0184】

（動作14）

次に、動作14を説明する。

10

20

30

40

50

動作 1 4 は、計算ウィンドウ C W 3 6 の関数式を、幾何ウィンドウ G W 3 6 にドラッグ & ドロップする際の動作である。

【 0 1 8 5 】

図 9 において、先ず C P U 1 0 は、計算アプリケーションの実行を開始して、計算ウィンドウ C W 3 6 を表示部 4 0 に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウ G W 3 6 を表示部 4 0 に表示させる。そして、ユーザは、入力ペン 4 を用いることで、計算ウィンドウ C W 3 6 のカーソル位置に関数式を入力する。

【 0 1 8 6 】

図 2 3 (a) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウ C W 3 6、幾何ウィンドウ G W 3 6、が表示されている。また、計算ウィンドウ C W 3 6 には、関数式を表す文字列「 $x^2 - 3$ 」が表示されている。 10

【 0 1 8 7 】

この表示画面において、ユーザは、入力ペン 4 を用いて、計算ウィンドウ C W 3 6 の文字列「 $x^2 - 3$ 」を含む文字列表示領域 T 7 0 を指定して反転表示させ、ドラッグ & ドロップ操作によって、幾何ウィンドウ G W 3 6 へ移動させる。

【 0 1 8 8 】

ドラッグ & ドロップ操作を検知すると (ステップ S 1 1 : Y E S)、C P U 1 0 は、当該検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“計算ウィンドウ”であり、ドラッグされたデータは関数式“ $x^2 - 3$ ”を表す文字列であり、ドロップ先のウィンドウは“幾何ウィンドウ”である、と判断する (ステップ S 1 2) 20

【 0 1 8 9 】

次いで、C P U 1 0 は、当該判断内容に基づく動作説明メッセージを表示させる。即ち、同図においては、計算 / 幾何メッセージテーブル 6 2 を参照し、データ種別“関数式”に対応するメッセージ“関数式に対応する関数グラフを描画します”を表示させる (ステップ S 1 3)。

【 0 1 9 0 】

図 2 3 (b) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「関数式に対応する関数グラフを描画します」という動作説明メッセージが、ポップアップウィンドウ P W 3 6 に表示されている。 30

【 0 1 9 1 】

続いて、C P U 1 0 は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、文字列「 $x^2 - 3$ 」を関数式“ $x^2 - 3$ ”に変換して、コピーバッファ 5 4 に格納する (ステップ S 1 4)。

【 0 1 9 2 】

その後、C P U 1 0 は、ポップアップウィンドウ P W 3 6 の表示を終了し (ステップ S 1 5)、コピーバッファ 5 4 に格納されている関数式“ $x^2 - 3$ ”に基づいて、幾何ウィンドウ G W 3 6 に関数グラフオブジェクト 1 7 2 を描画する。それとともに、当該関数グラフオブジェクト 1 7 2 に割り当てた識別 ID に、幾何種別“関数グラフ”と、関数グラフオブジェクト 1 7 2 の関数式“ $x^2 - 3$ ”とを対応付けたデータを、幾何ウィンドウデータ 5 1 に追加格納する (ステップ S 1 6)。 40

【 0 1 9 3 】

図 2 3 (c) は、この段階での表示画面の一例を示す図である。同図によれば、幾何ウィンドウ G W 3 6 には、関数式“ $x^2 - 3$ ”に対応する関数グラフオブジェクト 1 7 2 が描画されている。

以上の処理を行うと、C P U 1 0 は、本動作説明メッセージ表示処理を終了する。

【 0 1 9 4 】

(動作 1 5)

次に、動作 1 5 を説明する。

動作 15 は、幾何ウィンドウ GW38 の 2 つの直線オブジェクトを、計算ウィンドウ CW38 へドラッグ&ドロップする際の動作である。

【0195】

図 9 において、先ず CPU10 は、計算アプリケーションの実行を開始して、計算ウィンドウ CW38 を表示部 40 に表示させるとともに、幾何アプリケーションの実行を開始して、幾何ウィンドウ GW38 を表示部 40 に表示させる。そして、ユーザは、例えば入力ペン 4 を用いることで直線描画コマンドを指定し、幾何ウィンドウ GW38 上の所望の 2 点をタッチした後、再度直線描画コマンドを指定し、幾何ウィンドウ GW38 上の所望の 2 点をタッチする。すると、CPU10 は、先にタッチされた 2 点を結ぶ直線オブジェクト、後にタッチされた 2 点を結ぶ直線オブジェクト、の 2 つの直線オブジェクトを描画するとともに、当該描画した 2 つの直線オブジェクトに該当するデータを、幾何ウィンドウデータ 51 に追加格納する。

10

【0196】

図 24 (a) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、計算ウィンドウ CW38、幾何ウィンドウ GW38、が表示されている。また、幾何ウィンドウ GW38 には、直線オブジェクト 180、182、が描画されている。このとき、幾何ウィンドウデータ 51 には、直線オブジェクト 180 に割り当てた識別 ID と、幾何種別“直線”と、直線オブジェクト 180 上の 2 点の座標 A (-1、-3)、B (2、0) と、を対応付けたデータが格納されているとともに、直線オブジェクト 182 に割り当てた識別 ID と、幾何種別“直線”と、直線オブジェクト 182 上の 2 点の座標 C (-2、6)、D (-1、4) と、を対応付けたデータが格納されている。

20

この表示画面において、ユーザは、入力ペン 4 を用いて直線オブジェクト 180、182 を指定し、ドラッグ&ドロップ操作によって、計算ウィンドウ CW38 へ移動させる。

【0197】

ドラッグ&ドロップ操作を検知すると(ステップ S11: YES)、CPU10 は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“幾何ウィンドウ”であり、ドラッグされたデータは直線オブジェクト 180、182 であり、ドロップ先のウィンドウは“計算ウィンドウ”である、と判断する(ステップ S12)。

【0198】

次いで、CPU10 は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、幾何/計算メッセージテーブル 61 を参照し、幾何種別が“複数の直線”に対応する動作説明メッセージ“直線に対応する連立方程式を作成します”を表示させる(ステップ S13)。

30

【0199】

図 24 (b) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「直線に対応する連立方程式を作成します」という動作説明メッセージが、ポップアップウィンドウ PW38 に表示されている。

【0200】

次いで、CPU10 は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、幾何ウィンドウデータ 51 から、直線オブジェクト 180 の特定点座標を読み出し、読み出した 2 つの座標 A (-1、-3)、B (2、0) を結ぶ直線式“ $y = x - 2$ ”を生成する。それとともに、幾何ウィンドウデータ 51 から、直線オブジェクト 182 の特定点座標を読み出し、読み出した 2 つの座標 C (-2、6)、D (-1、4) を結ぶ直線式“ $y = -2 * x + 2$ ”を生成する。そして、生成した 2 つの直線式を連立させた連立方程式を表す文字列を生成して、コピーバッファ 54 に格納する(ステップ S14)。

40

【0201】

その後、CPU10 は、ポップアップウィンドウ PW38 の表示を終了し(ステップ S15)、コピーバッファ 54 に格納されている文字列を、計算ウィンドウ CW38 のカーソ

50

ル位置に表示させる（ステップS16）。

【0202】

図24(c)は、この段階での表示画面の一例を示す図である。同図によれば、計算ウィンドウCW38には、2つの直線オブジェクト180、182に対応する直線式を連立させた連立方程式が表示されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0203】

（動作16）

次に、動作16を説明する。

動作16は、テーブルウィンドウTW40のテーブルを、グラフ式ウィンドウRW40へドラッグ&ドロップする際の動作である。 10

【0204】

図9において、先ずCPU10は、グラフアプリケーションの実行を開始して、グラフ式ウィンドウRW40を表示部40に表示させるとともに、テーブルアプリケーションの実行を開始して、テーブルウィンドウTW40を表示部40に表示させる。そして、ユーザは、入力ペン4を用いてテーブル作成コマンドを指定し、テーブルウィンドウTW40にテーブルデータを入力する。

【0205】

図25(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、グラフ式ウィンドウRW40、テーブルウィンドウTW40、が表示されている。 20

また、グラフ式ウィンドウRW40には、グラフ式を入力するためのグラフ式入力画面が表示され、テーブルウィンドウTW40には、テーブルデータが表示されている。

この表示画面において、ユーザは、入力ペン4を用いて、テーブルウィンドウTW40のテーブルデータを含む文字列表示領域T80を指定して反転表示させ、ドラッグ&ドロップ操作によって、グラフ式ウィンドウRW40へ移動させる。

【0206】

ドラッグ&ドロップ操作を検知すると（ステップS11：YES）、CPU10は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“テーブルウィンドウ”であり、ドラッグされたデータはテーブルデータであり、ドロップ先のウィンドウは“グラフ式ウィンドウ”である、と判断する（ステップS12）。 30

【0207】

次いで、CPU10は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、テーブル/グラフメッセージテーブル63を参照し、データ種別“テーブル”に対応する動作説明メッセージ“テーブルに対応するグラフ式を作成します”を表示させる（ステップS13）。

【0208】

図25(b)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「テーブルに対応するグラフ式を作成します」という動作説明メッセージが、ポップアップウィンドウPW40に表示されている。 40

【0209】

続いて、CPU10は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、テーブルデータを表す文字列をテーブルデータに変換し、変換後のテーブルデータをグラフ式“ $x - 3$ ”に変換して、コピーバッファ54に格納する（ステップS14）。

【0210】

その後、CPU10は、ポップアップウィンドウPW40の表示を終了し（ステップS15）、コピーバッファ54に格納されているグラフ式を、グラフ式ウィンドウRW40のグラフ式入力画面に表示させる（ステップS16）。

【0211】

図 2 5 (c) は、この段階での表示画面の一例を示す図である。同図によれば、グラフ式ウィンドウ R W 4 0 のグラフ式入力画面には、テーブルデータに対応するグラフ式 “ $y = x - 3$ ” が表示されている。

以上の処理を行うと、C P U 1 0 は、本動作説明メッセージ表示処理を終了する。

【 0 2 1 2 】

(動作 1 7)

次に、動作 1 7 を説明する。

動作 1 7 は、グラフウィンドウ R W 4 2 のグラフ式を、テーブルウィンドウ T W 4 2 へドラッグ&ドロップする際の動作である。

【 0 2 1 3 】

図 9 において、先ず C P U 1 0 は、グラフアプリケーションの実行を開始して、グラフウィンドウ R W 4 2 を表示部 4 0 に表示させるとともに、計算アプリケーションの実行を開始して、計算ウィンドウ C W 4 2 を表示部 4 0 に表示させる。そして、ユーザは、入力ペン 4 を用いることで、グラフウィンドウ R W 4 2 のグラフ式入力画面にグラフ式を入力する。すると、C P U 1 0 は、入力されたグラフ式のグラフを、グラフウィンドウ R W 4 2 のグラフ表示画面に描画する。

【 0 2 1 4 】

図 2 6 (a) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、グラフウィンドウ R W 4 2、テーブルウィンドウ T W 4 2 が表示されている。グラフウィンドウ R W 4 2 には、グラフを表示するためのグラフ表示画面が表示されている。そして、グラフ表示画面には、グラフ 1 9 0、1 9 2 が表示され、グラフ式表示エリア R A 4 2 には、グラフ 1 9 2 のグラフ式を表す文字列 “ $y = x - 3$ ” が表示されている。

【 0 2 1 5 】

この表示画面において、ユーザは、入力ペン 4 を用いて、グラフ式表示エリア R A 4 2 の文字列 “ $y = x - 3$ ” を含む文字列領域 T 9 0 を指定して反転表示させ、ドラッグ&ドロップ操作によって、テーブルウィンドウ T W 4 2 へ移動させる。

【 0 2 1 6 】

ドラッグ&ドロップ操作を検知すると (ステップ S 1 1 : Y E S)、C P U 1 0 は、この検知内容から、ドラッグ元のウィンドウの種類、ドラッグされたデータ、ドロップ先のウィンドウの種類、を判断する。即ち、同図においては、ドラッグ元のウィンドウは “ グラフウィンドウ ” であり、ドラッグされたデータはグラフ式を表す文字列 “ $y = x - 3$ ” であり、ドロップ先のウィンドウは “ テーブルウィンドウ ” である、と判断する (ステップ S 1 2)。

【 0 2 1 7 】

次いで、C P U 1 0 は、当該判断内容に該当する動作説明メッセージを表示させる。即ち、同図においては、グラフ/テーブルメッセージテーブル 6 4 を参照して、データ種別 “ グラフ式 ” に対応する動作説明メッセージ “ グラフ式に対応するテーブルを作成します ” を表示させる (ステップ S 1 3)。

【 0 2 1 8 】

図 2 6 (b) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、 “ グラフ式に対応するテーブルを作成します ” という動作説明メッセージが、ポップアップウィンドウ P W 4 2 に表示されている。

【 0 2 1 9 】

続いて、C P U 1 0 は、ドラッグされたデータを、ドロップ先のウィンドウの種類に基づいて変換する。即ち、文字列 “ $y = x - 3$ ” が表すグラフ式 “ $y = x - 3$ ” をテーブルデータに変換して、コピーバッファ 5 4 に格納する (ステップ S 1 4)。

【 0 2 2 0 】

その後、C P U 1 0 は、ポップアップウィンドウ P W 4 2 の表示を終了し (ステップ S 1 5)、コピーバッファ 5 4 に格納されているテーブルデータを、テーブルウィンドウ T W 4 2 のカーソル位置に表示させる (ステップ S 1 6)。

10

20

30

40

50

【0221】

図26(c)は、この段階での表示画面の一例を示す図である。同図によれば、テーブルウィンドウTW42には、グラフ式“ $y = x - 3$ ”に対応するテーブルデータが表示されている。

以上の処理を行うと、CPU10は、本動作説明メッセージ表示処理を終了する。

【0222】

以上のように、本第1の実施の形態によれば、入力ペン4を用いてウィンドウ上の所望のデータを指定し、ドラッグ&ドロップ操作によって他のウィンドウまで移動させると、先ず、当該ドラッグ&ドロップ操作に応じてなされる動作を説明するメッセージ(動作説明メッセージ)が、ポップアップウィンドウPWに表示される。その後、当該操作に応じた動作、即ち、ドラッグさせたデータの、ドロップ先のウィンドウに応じた変換・表示が行われる。

10

【0223】

[第2の実施の形態]

次に、図27~図33を参照して、第2の実施の形態を説明する。

本第2の実施の形態は、上述した第1の実施の形態と、次の点で異なる。即ち、本第2の実施の形態において、関数電卓1は、一方のウィンドウに対するコピー操作が行われた後、他方のウィンドウに対するペースト操作が行われると、コピー操作によって指定されたデータについて変換可能な表示形態を、選択可能な候補とする動作選択メッセージを表示する。そして、これらの内から選択された表示形態に応じた動作説明メッセージを表示し、その後、コピー操作によって指定されたデータの表示形態を、当該選択された表示形態に変換して、表示する。

20

【0224】

[内部構成]

本第2の実施の形態における関数電卓1の構成は、上述した第1の実施の形態において図3に示した関数電卓1のROM60を、図27に示すROM80に置き換えた構成と同様である。このため、以下の説明においては、上述した第1の実施の形態と同一要素については同符号を付し、詳細な説明を省略する。

【0225】

図27は、第2の実施の形態におけるROM80の構成を示す図である。同図によれば、ROM80には、幾何変換候補テーブル81、計算変換候補テーブル82、グラフ変換候補テーブル83、幾何メッセージテーブル84、計算メッセージテーブル85、グラフメッセージテーブル86、動作選択メッセージ表示プログラム87、が格納される。

30

【0226】

幾何変換候補テーブル81は、幾何ウィンドウGWから他のウィンドウへのドラッグ&ドロップ操作がなされた際に表示される動作選択メッセージを管理するためのテーブルであり、幾何ウィンドウGW上のオブジェクトの種別(幾何種別)毎に、変換可能な表示形態の候補(変換候補)を対応付けたデータが格納されている。幾何ウィンドウGWから他のウィンドウへのドラッグ&ドロップ操作がなされると、CPU10は、この幾何変換候補テーブル81を参照して、幾何ウィンドウGW上のオブジェクトの幾何種別に対応する変換候補を全て選択して、これらを選択肢とする動作説明メッセージを表示させる。

40

【0227】

計算変換候補テーブル82は、計算ウィンドウCWに対するコピー操作がなされた際に表示される動作選択メッセージを管理するためのテーブルであり、計算ウィンドウCW上のデータの種別(データ種別)毎に、変換可能な表示形態の候補(変換候補)を対応付けたデータが格納されている。計算ウィンドウCWから他のウィンドウへのドラッグ&ドロップ操作がなされると、CPU10は、この計算変換候補テーブル82を参照して、計算ウィンドウCW上のデータの種別(データ種別)に対応する変換候補を全て選択して、これらを選択肢とする動作説明メッセージを表示させる。

【0228】

50

グラフ変換候補テーブル 8 3 は、グラフウィンドウ R W に対するコピー操作がなされた際に表示される動作選択メッセージを管理するためのテーブルであり、図 2 8 に、そのデータ構成の一例を示す。

【 0 2 2 9 】

図 2 8 は、グラフ変換候補テーブル 8 3 のデータ構成の一例を示す図である。同図によれば、グラフ変換候補テーブル 8 3 には、グラフウィンドウ R W 上のデータの種別（データ種別）毎に、変換可能な表示形態の候補（変換候補）を対応付けたデータが格納されている。また、この変換候補は、変換可能な表示形態を示す変換内容、当該変換を実行するために要するアプリケーションの名前、を対応付けたデータである。

【 0 2 3 0 】

例えば、データ種別“グラフ式”には、NO. 1 から NO. 5 までの 5 つの変換候補が対応付けられており、具体的には、指定されたグラフ式に対応するグラフの描画を示す“指定グラフ描画”、全てのグラフ式に対応するグラフの描画を示す“全グラフ描画”、グラフ式に対応するテーブルデータの作成を示す“テーブル作成”グラフ式から対応するリストへの変換を示す“リスト変換”、グラフ式から対応する行列への変換を示す“行列変換”、が対応付けられている。

【 0 2 3 1 】

グラフウィンドウ R W 又はグラフ式ウィンドウ R W から他のウィンドウへのドラッグ&ドロップ操作がなされると、CPU 1 0 は、このグラフ変換候補テーブル 8 3 を参照して、グラフウィンドウ R W 上のデータの種別（データ種別）に対応する変換候補を選択肢とする動作説明メッセージを表示させる。

【 0 2 3 2 】

幾何メッセージテーブル 8 4 は、幾何ウィンドウ G W から他のウィンドウへのドラッグ&ドロップ操作がなされた際に表示される動作説明メッセージを管理するためのテーブルであり、幾何ウィンドウ G W 上のオブジェクトの種別（幾何種別）毎に、変換名毎と、動作説明メッセージと、を対応付けた複数のデータを格納している。幾何ウィンドウ G W から他のウィンドウへのドラッグ&ドロップ操作がなされると、CPU 1 0 は、この幾何メッセージテーブル 8 4 を参照して、指定された幾何種別及び変換名に対応する動作説明メッセージを選択して表示させる。

【 0 2 3 3 】

計算メッセージテーブル 8 5 は、計算ウィンドウ C W から他のウィンドウへのドラッグ&ドロップ操作がなされた際に表示される動作説明メッセージを管理するためのテーブルであり、計算ウィンドウ C W 上のデータの種別（データ種別）毎に、変換名毎と、動作説明メッセージと、を対応付けた複数のデータを格納している。計算ウィンドウ C W から他のウィンドウへのドラッグ&ドロップ操作がなされると、CPU 1 0 は、この計算メッセージテーブル 8 5 を参照して、指定されたデータ種別及び変換名に対応する動作説明メッセージを選択して表示させる。

【 0 2 3 4 】

グラフメッセージテーブル 8 6 は、グラフウィンドウ R W から他のウィンドウへのドラッグ&ドロップ操作がなされた際に表示される動作説明メッセージを管理するためのテーブルであり、図 2 9 に、そのデータ構成の一例を示す。

【 0 2 3 5 】

図 2 9 は、グラフメッセージテーブル 8 6 のデータ構成の一例を示す図である。同図によれば、グラフメッセージテーブル 8 6 は、グラフウィンドウ R W 又はグラフ式ウィンドウ R W 上のデータの種別（データ種別）毎に、変換名と、動作説明メッセージと、を対応付けた複数のデータが格納されている。グラフウィンドウ R W 又はグラフ式ウィンドウ R W から他のウィンドウへのドラッグ&ドロップ操作がなされると、CPU 1 0 は、このグラフメッセージテーブル 8 6 を参照して、指定されたデータ種別及び変換名に対応する動作説明メッセージを選択して表示させる。

【 0 2 3 6 】

10

20

30

40

50

動作選択メッセージ表示プログラム 87 は、本第 2 の実施の形態における動作選択メッセージ表示処理（図 30 参照）を実現するためのプログラムであり、CPU 10 がこの動作選択メッセージ表示プログラム 87 を実行することで、動作選択メッセージ表示処理が実現される。

【0237】

〔動作〕

次に、本第 2 の実施の形態における動作を説明する。

図 30 は、動作選択メッセージ表示処理に係る関数電卓 1 の動作を説明するためのフローチャートであり、図 31 ~ 図 35 は、表示部 40 に表示される表示画面の遷移例を示す図である。

【0238】

図 30 において、先ず CPU 10 は、グラフアプリケーションの実行を開始して、グラフ式ウィンドウ RW50 を表示部 40 に表示させる。そして、ユーザは、入力ペン 4 を用いることで、グラフ式ウィンドウ RW50 のグラフ式入力画面にグラフ式を入力する。

【0239】

図 31 (a) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、グラフ式ウィンドウ RW50、が表示されている。グラフ式ウィンドウ RW50 には、グラフ式入力画面が表示され、これには、グラフ式「 $y_1 = x^2 - x - 2$ 」、「 $y_2 = x - 3$ 」が表示されている。

この表示画面において、ユーザは、入力ペン 4 を用いてグラフ式ウィンドウ RW50 を指定し、ドラッグ&ドロップ操作によって、グラフ式ウィンドウ RW50 以外の表示画面へ移動させる。

【0240】

ドラッグ&ドロップ操作を検知すると（ステップ S31：YES）、CPU 10 は、この検知内容から、ドラッグ元のウィンドウの種類、当該ウィンドウ上のデータの種別、を判断する。即ち、同図においては、ドラッグ元のウィンドウは“グラフ式ウィンドウ”であり、データの種別は“グラフ式”である、と判断する（ステップ S32）。

【0241】

次いで、CPU 10 は、当該判断内容に該当する動作選択メッセージを表示させる。即ち、グラフ変換候補テーブル 83 を参照して、データ種別“グラフ式”に対応する全ての変換候補の変換名を、選択肢として表示させる（ステップ S33）。

【0242】

図 30 (b) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「A. 指定グラフ描画?」、「B. 全グラフ描画?」、「C. テーブル作成?」、「D. リスト変換?」、「E. 行列変換?」といった 5 つの選択肢を含む動作選択メッセージが、ポップアップウィンドウ PW50 に表示されている。

【0243】

この表示画面において、ユーザは、入力ペン 4 を用いることで、これらの選択肢の内から所望のものを指定する。そして、選択肢の指定を検知すると（ステップ S34：YES）、CPU 10 は、指定された変換候補に基づく以下の処理を実行する。

【0244】

即ち、「A. 指定グラフ描画?」が指定された場合には、CPU 10 は、グラフメッセージテーブル 86 を参照し、データ種別“グラフ式”及び変換名“指定グラフ描画”に対応する動作説明メッセージ“指定グラフ式のグラフを描画します。グラフ式を指定して下さい”を表示させる（ステップ S35）。

【0245】

図 31 (c) は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「指定グラフ式のグラフを描画します。グラフ式を指定して下さい」という動作説明メッセージが、ポップアップウィンドウ PW52 に表示されている。このメッセージに回答して、ユーザは、入力ペン 4 を用いて、グラフウィンドウ RW50 上のグラフ式“y

10

20

30

40

50

$2 = x - 3$ ”を指定する。

【0246】

グラフ式の指定を検知すると、CPU10は、指定されたグラフ式“ $y^2 = x - 3$ ”をコピーバッファ54に格納した後(ステップS36)、ポップアップウィンドウPW52の表示を終了する(ステップS37)。そして、グラフを描画するためのグラフ表示画面Rを表示させたグラフウィンドウRW52を表示させて、コピーバッファ54に格納されているグラフ式“ $y^2 = x - 3$ ”のグラフを描画する(ステップS38)。

【0247】

図31(d)は、この段階での表示画面の一例を示す図である。同図によれば、グラフウィンドウRW52のグラフ表示画面に、グラフ式“ $y^2 = x - 3$ ”に対応するグラフ210が表示されている。 10

以上の処理を行うと、CPU10は、本動作選択メッセージ表示処理を終了する。

【0248】

また、ステップS34において、「B.全グラフ描画?」が指定された場合には、CPU10は、グラフメッセージテーブル86を参照し、データ種別“グラフ式”及び変換名“全グラフ描画”に対応する動作説明メッセージ“全グラフ式のグラフを描画します”を表示させる(ステップS35)。

【0249】

図32(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「全グラフ式のグラフを描画します」という動作説明メッセージが、ポップアップウィンドウPW54に表示されている。 20

【0250】

続いて、CPU10は、グラフ式“ $y^1 = x^2 - x - 2$ ”、“ $y^2 = x - 3$ ”を、コピーバッファ54に格納した後(ステップS36)、ポップアップウィンドウPW54の表示を終了し(ステップS37)、グラフ表示画面を表示させたグラフウィンドウRW54を表示させる。そして、このグラフウィンドウRW52のグラフ表示画面に、コピーバッファ54に格納されているグラフ式“ $y^1 = x^2 - x - 2$ ”のグラフ212、続いて、グラフ式“ $y^2 = x - 2$ ”のグラフ210を、描画する(ステップS38)。

【0251】

図32(b)は、この段階での表示画面の一例を示す図である。同図によれば、グラフウィンドウRW54のグラフ表示画面54には、グラフ式“ $y^1 = x^2 - x - 2$ ”に対応するグラフ212、グラフ式“ $y^2 = x - 3$ ”に対応するグラフ210、が表示されている。 30

以上の処理を行うと、CPU10は、本動作選択メッセージ表示処理を終了する。

【0252】

また、ステップS34において、「Cテーブル作成?」が指定された場合には、CPU10は、グラフメッセージテーブル86を参照し、データ種別“グラフ式”及び変換名“テーブル作成”に該当する動作説明メッセージ“グラフ式に対応するテーブルを作成します。グラフ式を指定して下さい”を表示させる(ステップS35)。

【0253】

図33(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「グラフ式に対応するテーブルを作成します。グラフ式を指定して下さい」という動作説明メッセージが、ポップアップウィンドウPW56に表示されている。このメッセージに回答して、ユーザは、入力ペン4を用いて、グラフウィンドウRW50上のグラフ式“ $y^2 = x - 3$ ”を指定する。 40

【0254】

グラフ式の指定を検知すると、CPU10は、テーブルアプリケーションの実行を開始して、テーブルウィンドウTW50を表示部40に表示させる。そして、指定されたグラフ式“ $y^2 = x - 3$ ”に対応するテーブルデータを作成し、作成したテーブルデータを表す文字列を生成して、コピーバッファ54に格納する(ステップS36)。 50

【0255】

その後、CPU10は、ポップアップウィンドウPW56の表示を終了し(ステップS37)、コピーバッファ54に格納されている文字列を、テーブルウィンドウTW50のカーソル位置に表示させる(ステップS38)。

【0256】

図33(b)は、この段階での表示画面の一例を示す図である。同図によれば、テーブルウィンドウTW50には、グラフ式“ $y_2 = x - 3$ ”に対応するテーブルデータが表示されている。

以上の処理を行うと、CPU10は、本動作選択メッセージ表示処理を終了する。

【0257】

また、ステップS34において、「D.リスト変換?」が指定された場合には、CPU10は、グラフメッセージテーブル86を参照し、データ種別“グラフ式”及び変換名“リスト変換”に対応する動作説明メッセージ“グラフ式に対応するリストに変換します”を表示させる(ステップS35)。

【0258】

図34(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「グラフ式に対応するリストに変換します」という動作説明メッセージが、ポップアップウィンドウPW58に表示されている。

【0259】

続いて、CPU10は、統計アプリケーションの実行を開始して、リストウィンドウSW52を表示部40に表示させる。そして、グラフ式“ $y_1 = x^2 - x - 2$ ”についてのリスト(List1)を算出して、コピーバッファ54に格納するとともに、グラフ式“ $y_2 = x - 3$ ”についてのリスト(List2)を同様に算出して、コピーバッファ54に格納する(ステップS36)。

【0260】

その後、CPU10は、ポップアップウィンドウPW58の表示を終了し(ステップS37)、コピーバッファ54に格納されているリスト(List1、List2)のデータを、リストウィンドウSW52に表示させる(ステップS38)。

【0261】

図34(b)は、この段階での表示画面の一例を示す図である。同図によれば、リストウィンドウSW52には、グラフ式“ $y_1 = x^2 - x - 2$ ”に対応するリスト(List1)のデータ、グラフ式“ $y_2 = x - 3$ ”に対応するリスト(List2)のデータ、が表示されている。

以上の処理を行うと、CPU10は、本動作選択メッセージ表示処理を終了する。

【0262】

また、ステップS34において、「E.行列変換?」が指定された場合には、CPU10は、グラフメッセージテーブル86を参照し、データ種別“グラフ式”及び変換名“行列変換”に対応する動作説明メッセージ“グラフ式に対応する行列に変換します”を表示させる(ステップS35)。

【0263】

図35(a)は、この段階での表示画面の一例を示す図である。同図によれば、表示画面には、「グラフ式に対応する行列に変換します」というメッセージが、ポップアップウィンドウPW60に表示されている。

【0264】

続いて、CPU10は、計算アプリケーションの実行を開始して、計算ウィンドウCW54を表示部40に表示させる。そして、グラフ式“ $y_1 = x^2 - x - 2$ ”及び“ $y_2 = x - 3$ ”に対応する行列に変換し、変換した行列を表す文字列を生成して、コピーバッファ54に格納する(ステップS36)。

【0265】

その後、CPU10は、ポップアップウィンドウPW60の表示を終了し(ステップS3

10

20

30

40

50

7)、コピーバッファ54に格納されている文字列を、計算ウィンドウCW54のカーソル位置に表示させる(ステップS38)。

【0266】

図34(b)は、この段階での表示画面の一例を示す図である。同図によれば、計算ウィンドウCW54には、グラフ式“ $y_1 = x^2 - x - 2$ ”及び“ $y_2 = x - 3$ ”に対応する行列が表示されている。

以上の処理を行うと、CPU10は、本動作選択メッセージ表示処理を終了する。

【0267】

以上のように、本第2の実施の形態によれば、入力ペン4を用いてウィンドウ上の所望のデータを指定し、ドラッグ&ドロップ操作によって他のウィンドウまで移動させると、先
10
ず、当該ドラッグしたデータに対して可能な動作内容の候補を選択させるメッセージ(動作選択メッセージ)が、ポップアップウィンドウPWに表示される。そして、これらの候補の内から所望の動作内容を選択すると、選択した動作内容に応じて、上記ドロップさせたデータが変換され、表示される。

【0268】

[変形例]

尚、本発明が適用可能な実施の形態は、上述した第1及び第2の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。

【0269】

例えば、上述した第1及び第2の実施の形態においては、動作説明メッセージ及び動作選
20
択メッセージは、ポップアップウィンドウPWに表示させることとしたが、表示画面上に専用のメッセージ表示エリアを設け、ここに表示させることとしても良い。

【0270】

また、動作説明メッセージ及び動作選択メッセージは、文字列で表示(テキスト表示)することとしたが、これをアイコン(絵文字)で表示することとしても良い。

【0271】

【発明の効果】

請求項1又は5に記載の発明によれば、それぞれ異なる表示形態でデータを表示する複数の表示画面の内、一の表示画面に対する複写操作が行われた後、他の表示画面に対する貼付操作が行われた場合に、貼り付け操作がなされた他の表示画面の表示形態を判別して、
30
この判別した表示形態に応じた変換内容を説明するメッセージを表示制御することができる。その後、複写操作によって指定されたデータを、判別した表示形態に変換して、他の表示画面に表示制御することができる。

【0272】

また、請求項2又は6に記載の発明によれば、第1の表示画面に対する複写操作がなされた後、第2表示画面に対する貼付操作がなされた場合に、第2画面に表示可能な複数の表示形態を判別して、この判別した複数の表示形態を候補として表示制御することができる。そして、これらの表示形態の候補の内から何れか一つが選択されると、選択された表示形態に応じた変換内容を説明するメッセージを表示制御し、その後、複写操作によって指定されたデータを、上記選択された表示形態に変換して、第2表示画面に表示させる制
40
御を行うことができる。

【0273】

また、請求項3に記載の発明によれば、複写操作によって指定されたデータに応じて、表示可能な表示形態の候補を選択して表示制御することができる。

【0274】

また、請求項4に記載の発明によれば、複写操作によって指定されたデータに対する変換内容を説明するメッセージを、表示部が有する変換内容表示画面に表示させる制御を行うことができる。

【0275】

従って、本発明によれば、複写操作及び貼付操作を行った際に、当該操作に応じた変換内
50

容を説明するメッセージが表示制御なされた後、指定したデータに対する変換が行われるため、電子機器の操作の簡易性を損なうことなく、電子機器によってなされる動作を容易に理解することができる。

【図面の簡単な説明】

【図 1】本発明を適用した関数電卓の外観例を示す図。

【図 2】データの表示形態の変換の際に表示される動作説明メッセージを示す図。

【図 3】第 1 の実施の形態における関数電卓の内部構成例を示すブロック図。

【図 4】幾何ウィンドウデータのデータ構成例を示す図。

【図 5】幾何 / 計算メッセージテーブルのデータ構成例を示す図。

【図 6】計算 / 幾何メッセージテーブルのデータ構成例を示す図。

10

【図 7】テーブル / グラフメッセージテーブルのデータ構成例を示す図。

【図 8】グラフ / テーブルメッセージテーブルのデータ構成例を示す図。

【図 9】動作説明メッセージ表示処理を説明するためのフローチャート。

【図 10】動作 1 における表示画面の遷移例を示す図。

【図 11】動作 2 における表示画面の遷移例を示す図。

【図 12】動作 3 における表示画面の遷移例を示す図。

【図 13】動作 4 における表示画面の遷移例を示す図。

【図 14】動作 5 における表示画面の遷移例を示す図。

【図 15】動作 6 における表示画面の遷移例を示す図。

【図 16】動作 7 における表示画面の遷移例を示す図。

20

【図 17】動作 8 における表示画面の遷移例を示す図。

【図 18】動作 9 における表示画面の遷移例を示す図。

【図 19】動作 10 における表示画面の遷移例を示す図。

【図 20】動作 11 における表示画面の遷移例を示す図。

【図 21】動作 12 における表示画面の遷移例を示す図。

【図 22】動作 13 における表示画面の遷移例を示す図。

【図 23】動作 14 における表示画面の遷移例を示す図。

【図 24】動作 15 における表示画面の遷移例を示す図。

【図 25】動作 16 における表示画面の遷移例を示す図。

【図 26】動作 17 における表示画面の遷移例を示す図。

30

【図 27】第 2 の実施の形態における ROM の構成例を示す図。

【図 28】グラフ変換候補テーブルのデータ構成例を示す図。

【図 29】グラフメッセージテーブルのデータ構成例を示す図。

【図 30】動作選択メッセージ表示処理を説明するためのフローチャート。

【図 31】第 2 の実施の形態における表示画面の遷移例を示す図。

【図 32】第 2 の実施の形態における表示画面の遷移例を示す図。

【図 33】第 2 の実施の形態における表示画面の遷移例を示す図。

【図 34】第 2 の実施の形態における表示画面の遷移例を示す図。

【図 35】第 2 の実施の形態における表示画面の遷移例を示す図。

【符号の説明】

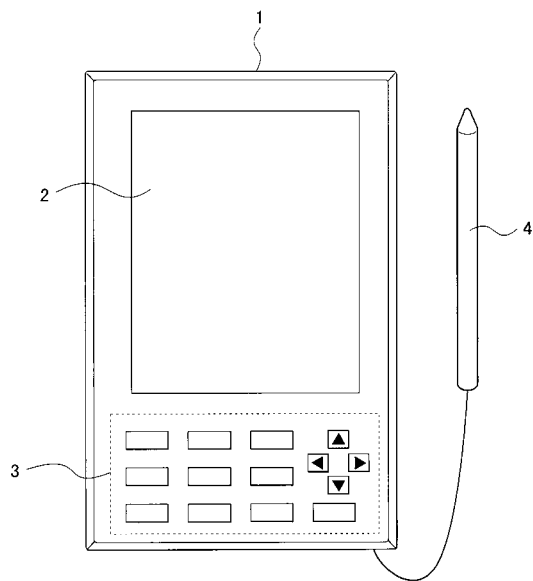
40

- 1 関数電卓
- 10 CPU
- 20 入力部
- 30 タブレット
- 31 位置検出回路
- 40 表示部
- 41 表示駆動回路
- 50 ROM
- 51 幾何ウィンドウデータ
- 52 計算ウィンドウデータ

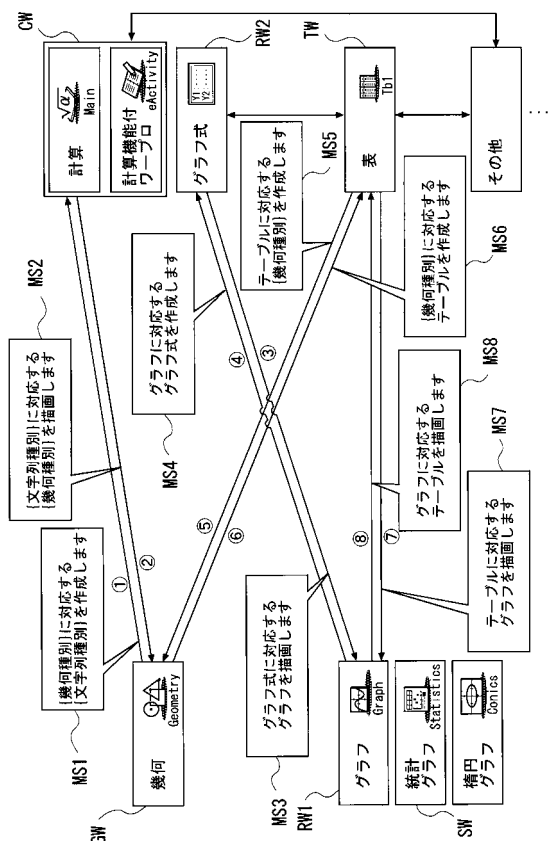
50

- 5 3 グラフウィンドウデータ
- 5 3 a グラフ式データ
- 5 4 コピーバッファ
- 6 0、8 0 R A M
- 6 1 幾何 / 計算メッセージテーブル
- 6 2 計算 / 幾何メッセージテーブル
- 6 3 テーブル / グラフメッセージテーブル
- 6 4 グラフ / テーブルメッセージテーブル
- 6 5 動作説明メッセージ表示プログラム
- 8 1 幾何変換候補テーブル
- 8 2 計算候補変換テーブル
- 8 3 グラフ変換候補テーブル
- 8 4 幾何メッセージテーブル
- 8 5 計算メッセージテーブル
- 8 6 グラフメッセージテーブル
- 8 7 動作選択メッセージ表示プログラム

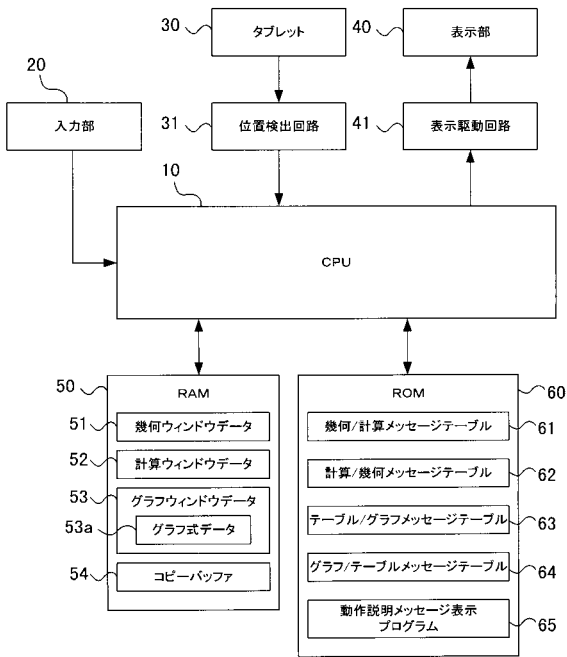
【 図 1 】



【 図 2 】



【 図 3 】



【 図 4 】

51 幾何ウィンドウデータ

識別ID	幾何種別	特定情報
ID0012	点	(-3,3)
ID0013	ベクトル	(4,2)
ID0014	{n}角形(n=4)	(0,0), (1,2), (5,3), (4,1)
ID0015	直線	(1,-1), (-1,-2)
ID0016	円	(0,0), (2,0)
ID0017	楕円	(0,0), (2,0), (0,1)
ID0018	関数グラフ	x^2
§	§	§

【 図 5 】

61 幾何/計算メッセージテーブル

幾何種別	動作説明メッセージ
点	点に対応する点座標を作成します
ベクトル	ベクトルに対応するベクトル座標を作成します
{n}角形	{n}角形の頂点座標を作成します
直線	直線に対応する直線式を作成します
直線(複数)	複数の直線に対応する連立方程式を作成します
円	円の方程式を作成します
楕円	楕円の方程式を作成します
関数グラフ	関数グラフに対応する関数式を作成します
§	§

【 図 6 】

62 計算/幾何メッセージテーブル

データ種別	動作説明メッセージ
点座標	点座標に対応する点を描画します
ベクトル座標	ベクトル座標に対応するベクトルを描画します
{n}角形の頂点座標	頂点座標に対応する{n}角形を描画します
直線式	直線式に対応する直線を描画します
円の方程式	円の方程式に対応する円を描画します
楕円の方程式	楕円の方程式に対応する楕円を描画します
関数式	関数式に対応する関数グラフを描画します
§	§

【 図 7 】

63 テーブル/グラフメッセージテーブル

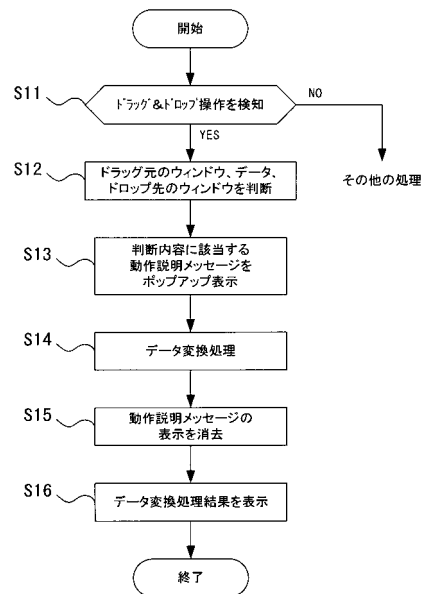
データ種別	動作説明メッセージ
テーブル	テーブルに対応するグラフ式を作成します
§	§

【 図 8 】

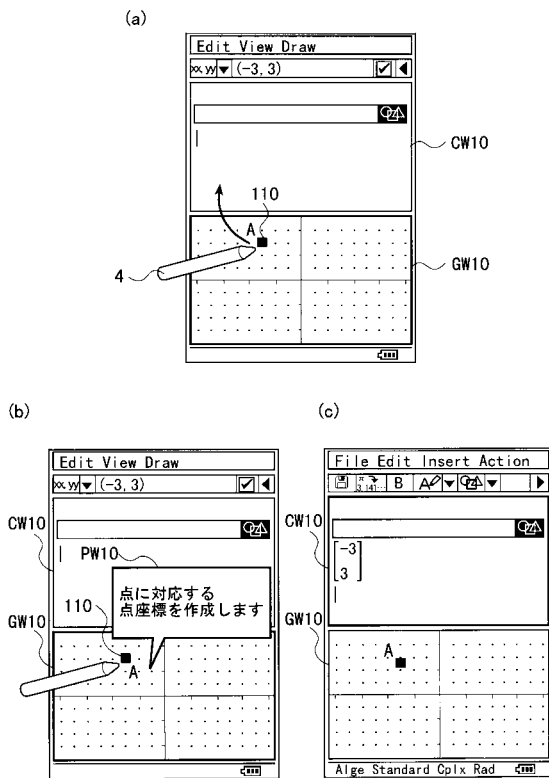
64 グラフ/テーブルメッセージテーブル

データ種別	動作説明メッセージ
グラフ式	グラフ式に対応するテーブルデータを作成します
§	§

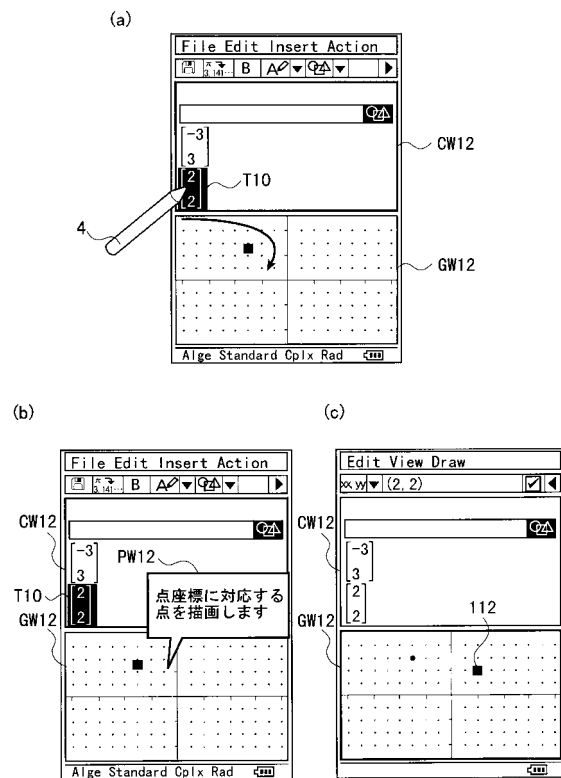
【 図 9 】



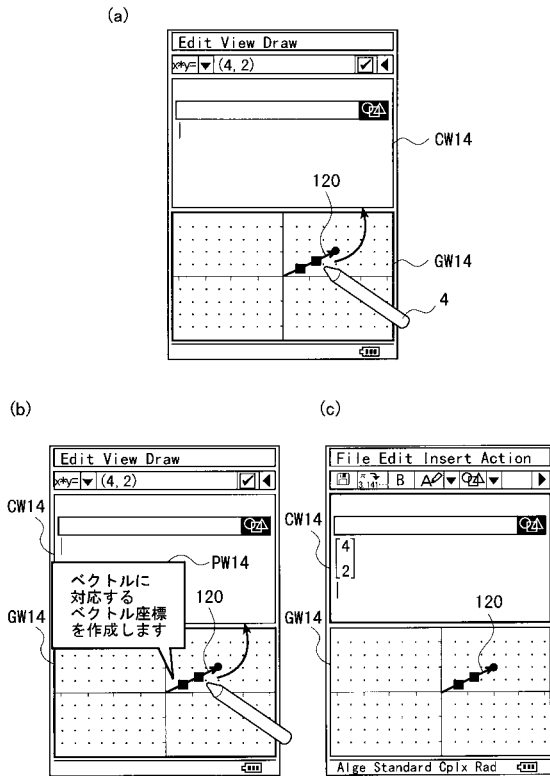
【 図 10 】



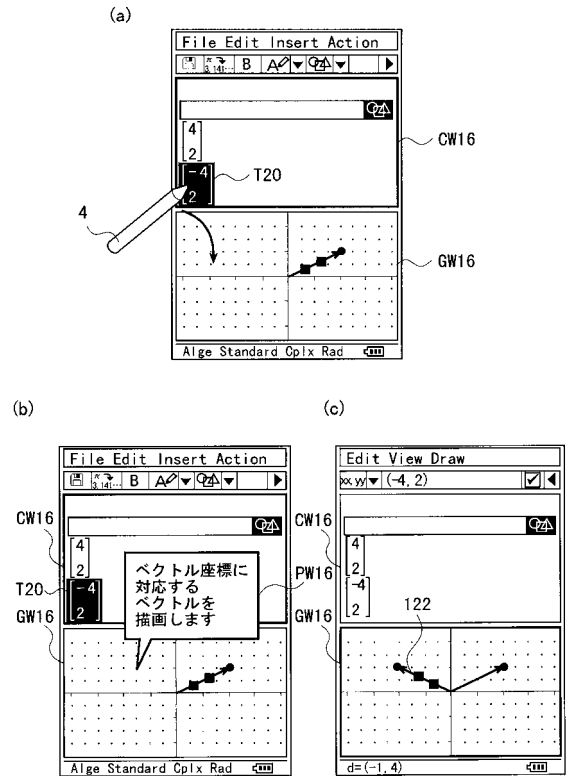
【 図 11 】



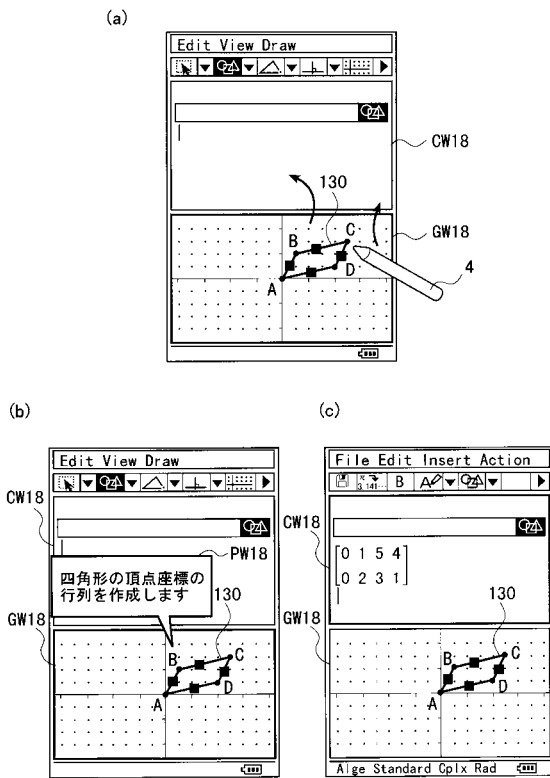
【 図 1 2 】



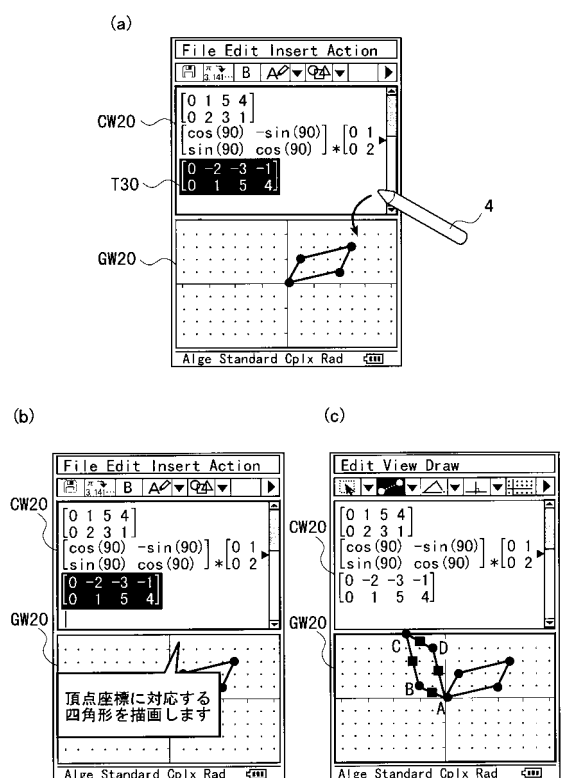
【 図 1 3 】



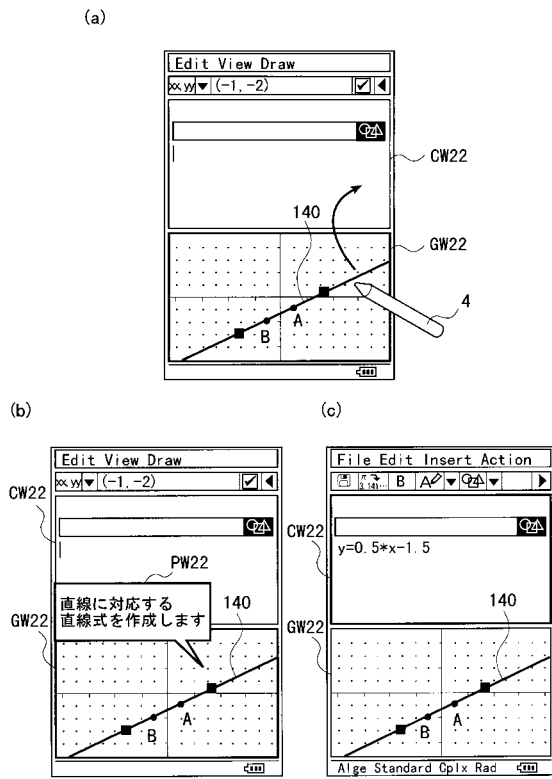
【 図 1 4 】



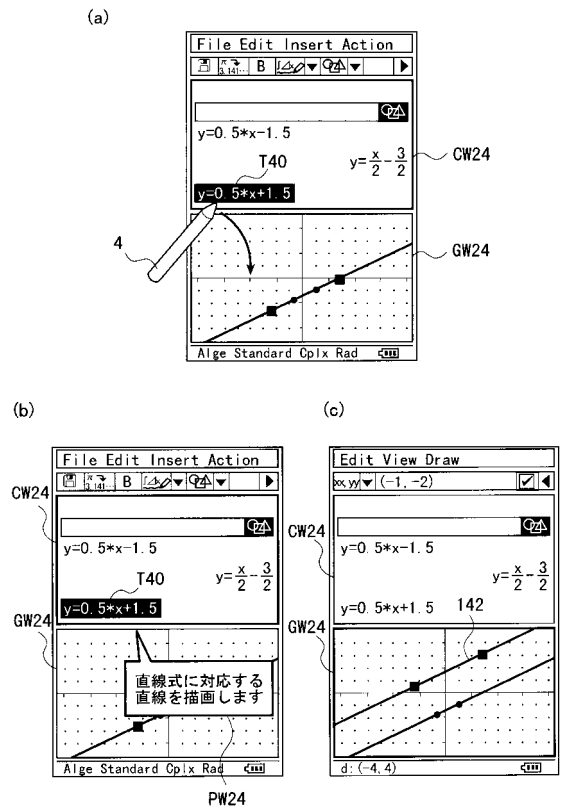
【 図 1 5 】



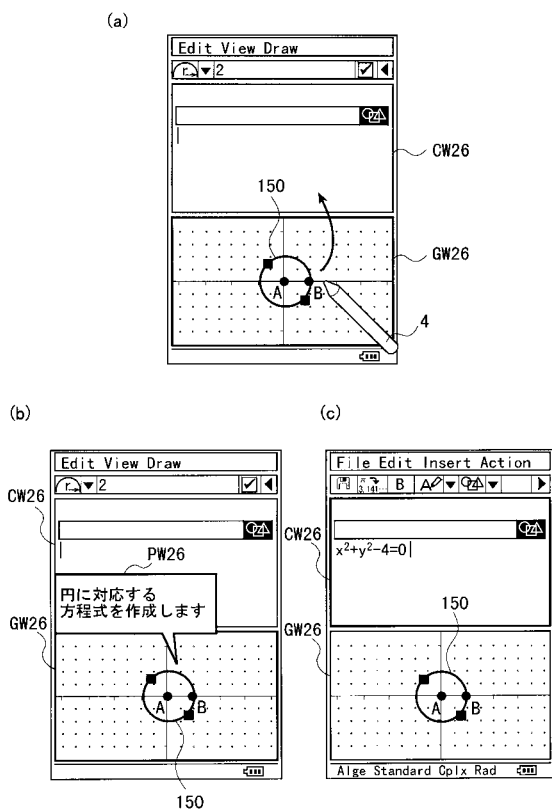
【 図 16 】



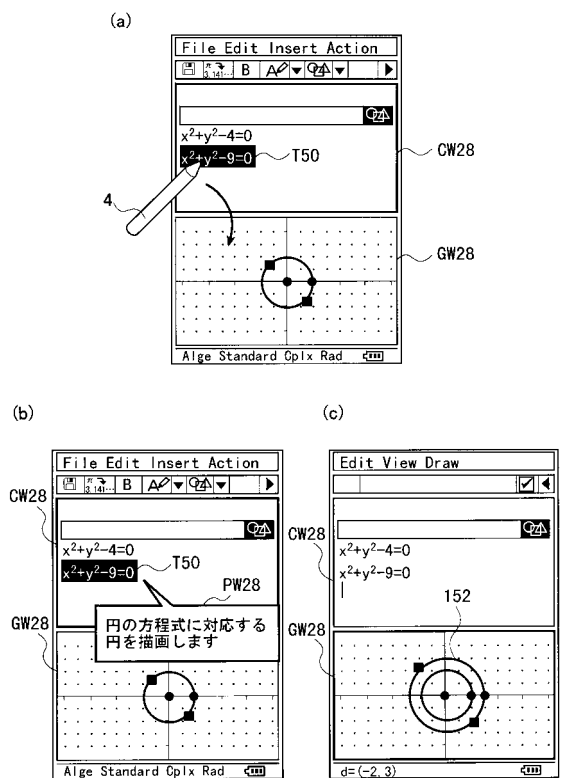
【 図 17 】



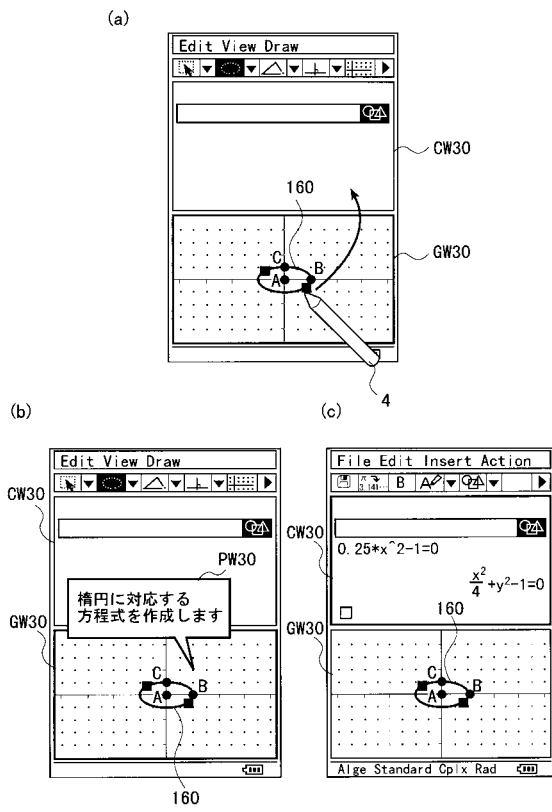
【 図 18 】



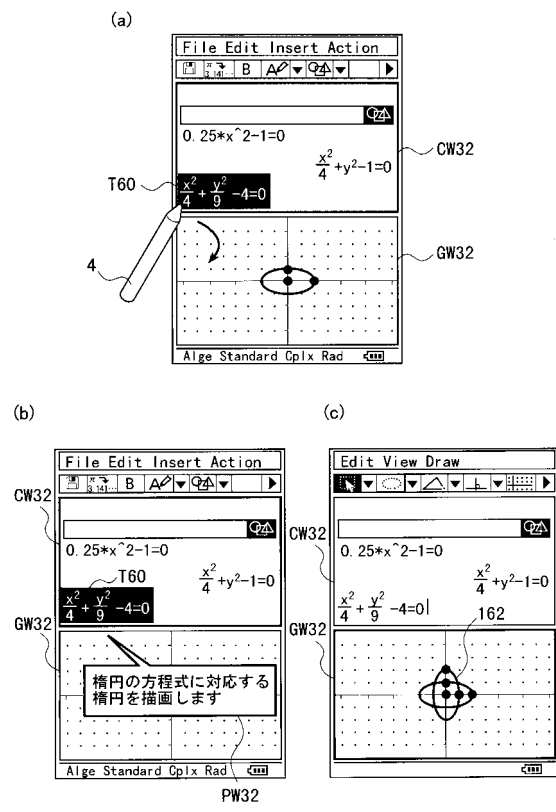
【 図 19 】



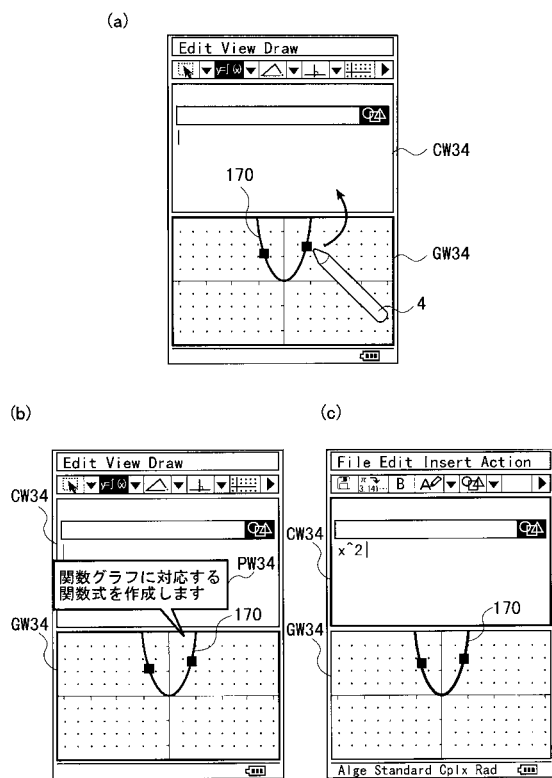
【 図 2 0 】



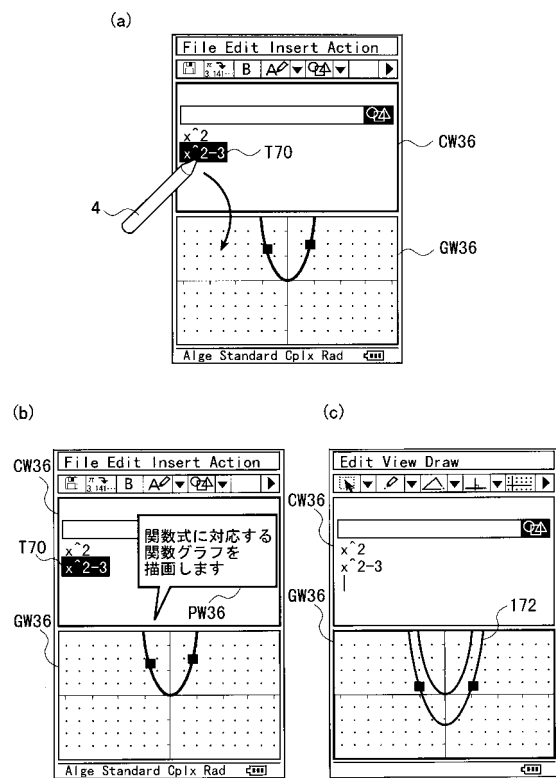
【 図 2 1 】



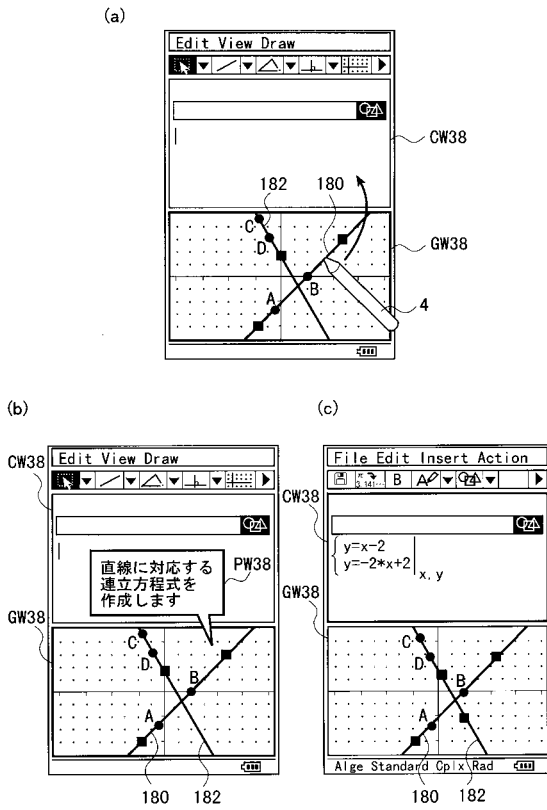
【 図 2 2 】



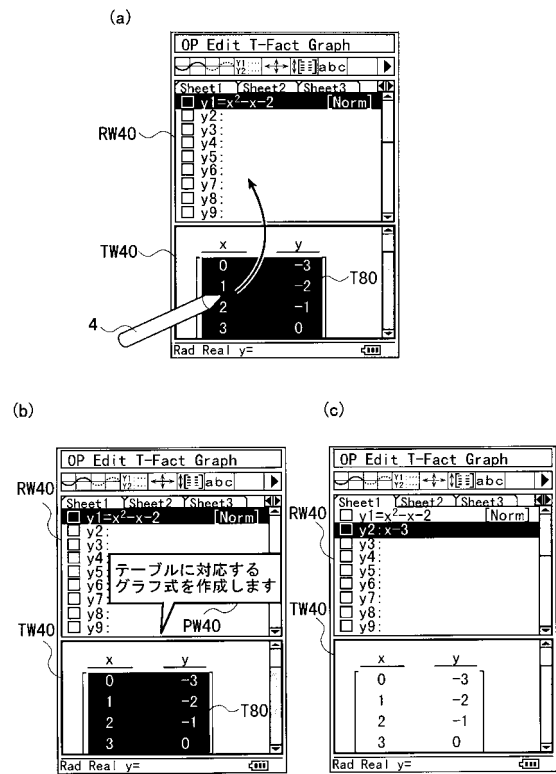
【 図 2 3 】



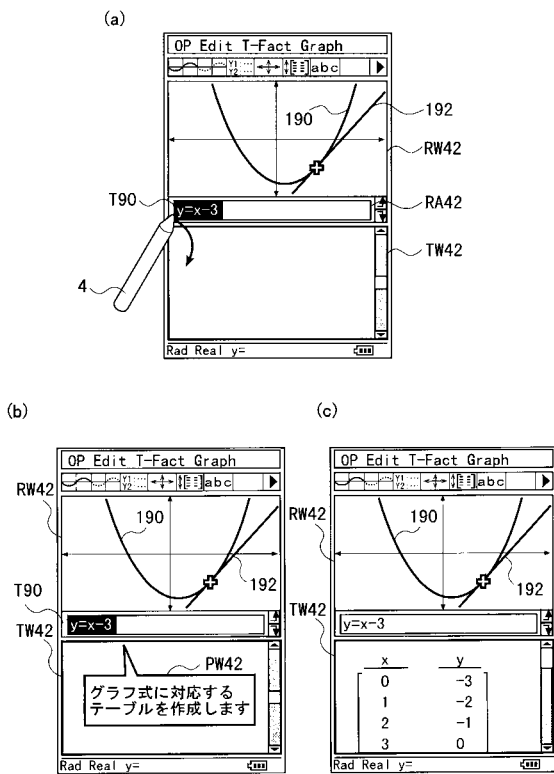
【 図 2 4 】



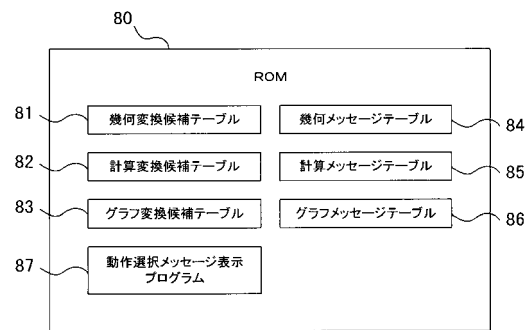
【 図 2 5 】



【 図 2 6 】



【 図 2 7 】



【 図 2 8 】

83 グラフ変換候補テーブル

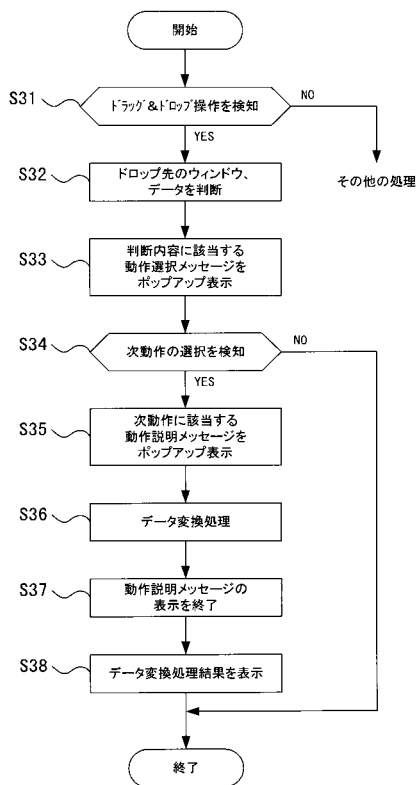
データ種別	変換候補		
	NO.	変換名	アプリケーション
グラフ式	1	指定グラフ描画	グラフアプリケーション
	2	全グラフ描画	グラフアプリケーション
	3	テーブル作成	テーブルアプリケーション
	4	リスト変換	統計アプリケーション
	5	行列変換	計算アプリケーション
§	§	§	§

【 図 2 9 】

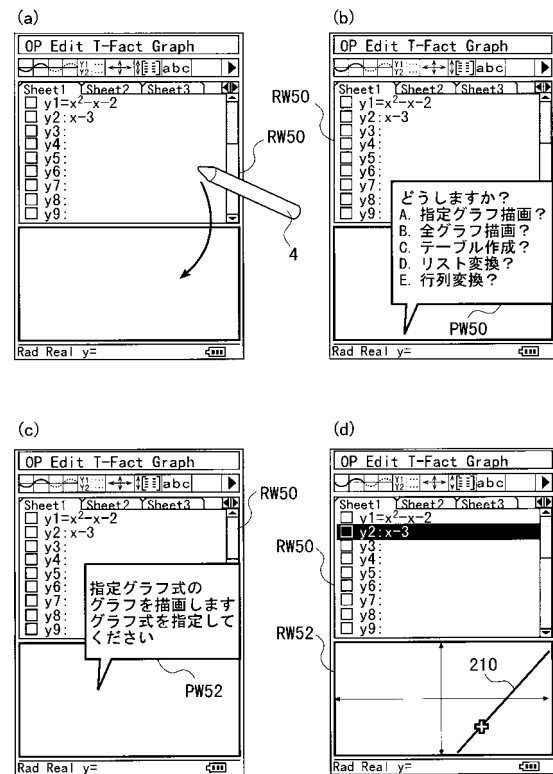
86 グラフメッセージテーブル

データ種別	変換名	動作説明メッセージ
グラフ式	指定グラフ描画	指定グラフ式のグラフを描画します。グラフ式を指定して下さい
	全グラフ描画	全グラフ式のグラフを描画します
	テーブル作成	グラフ式に対応するテーブルを作成します。グラフ式を指定して下さい
	リスト変換	グラフ式に対応するリストに変換します
	行列変換	グラフ式に対応する行列に変換します
§	§	§

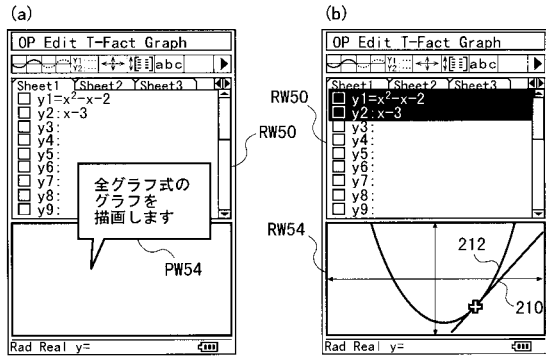
【 図 3 0 】



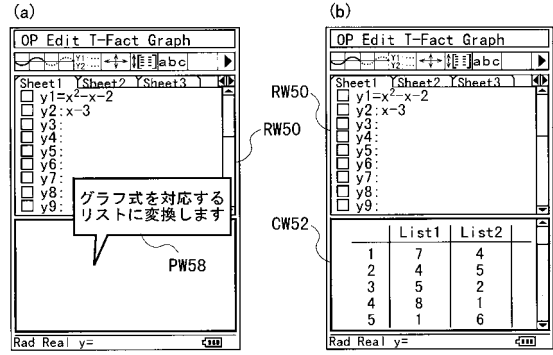
【 図 3 1 】



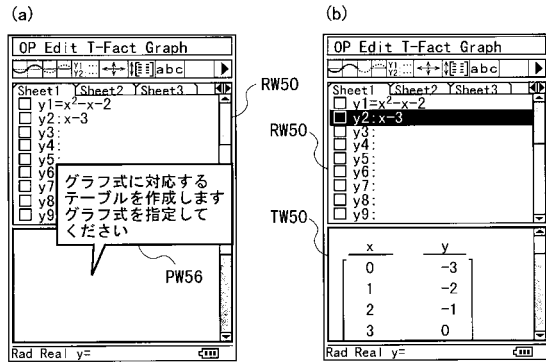
【 図 3 2 】



【 図 3 4 】



【 図 3 3 】



【 図 3 5 】

