Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2405287 C 2005/10/25

(11)(21) 2 405 287

(12) BREVET CANADIEN
CANADIAN PATENT

(13) C

(22) Date de dépôt/Filing Date: 1999/06/28

(41) Mise à la disp. pub./Open to Public Insp.: 1999/12/29

(45) Date de délivrance/Issue Date: 2005/10/25

(62) Demande originale/Original Application: 2 276 697

(30) Priorité/Priority: 1998/06/28 (09/106,585) US

(51) Cl.Int.^{7}/Int.Cl.^{7} G06K 11/06, G06F 3/00

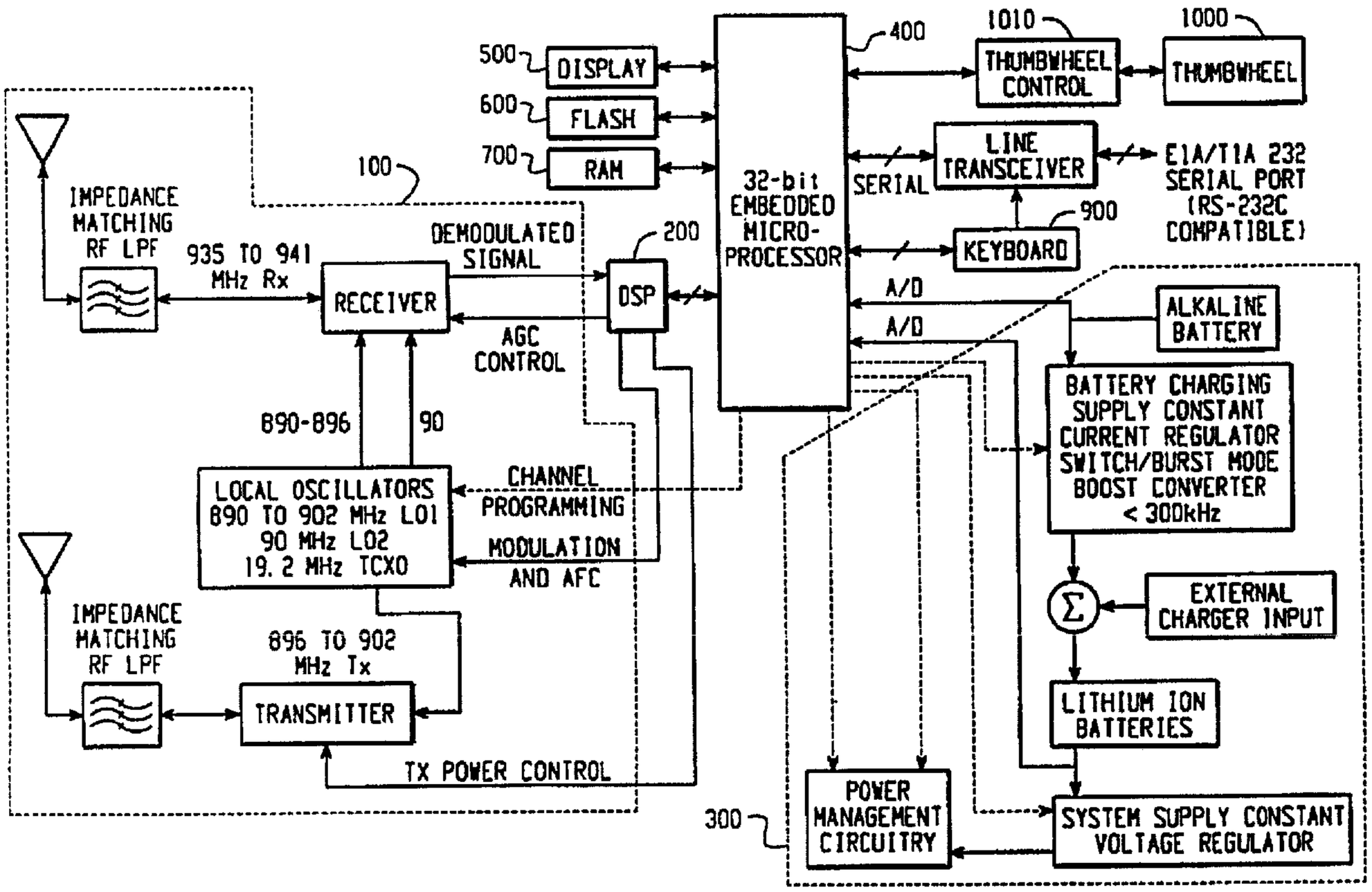
(72) Inventeurs/Inventors:
GRIFFIN, JASON T., CA;
HOLMES, JOHN A., CA;
LAZARDIS, MIHAL, CA;
LITTLE, HERB A., CA;
MAJOR, HARRY R., CA;
DUNK, CRAIG, CA;
BROWN, MICHAEL, CA;
LANG, JEROME, CA

(73) Propriétaire/Owner:
RESEARCH IN MOTION LIMITED, CA

(74) Agent: BORDEN LADNER GERVAIS LLP

(54) Titre : APPAREIL ELECTRONIQUE PORTATIF AVEC CLAVIER A UTILISATION OPTIMALE DES POUCES

(54) Title: HAND-HELD ELECTRONIC DEVICE WITH A KEYBOARD OPTIMIZED FOR USE WITH THE THUMBS



(57) Abrégé/Abstract:
A hand-held electronic device with a keyboard, thumbwheel, display and associated software is optimized for use of the device with the thumbs. A plurality of features to optimize efficient use of the limited keyboard space and encourage the use of the device by thumb-based data entry through the thumbwheel and/or through a combination of minimal number of keystrokes. Features include international character scrolling, autopunctuation, undo, character substitution and phrase substitution. In addition, the keys on the device keyboard are optimally shaped and configured for thumb-based input.

ABSTRACT

A hand-held electronic device with a keyboard, thumbwheel, display and associated software is optimized for use of the device with the thumbs. A plurality of features to optimize efficient use of the limited keyboard space and encourage the use of the device by thumb-based data entry through the thumbwheel and/or through a combination of minimal number of keystrokes. Features include international character scrolling, autopunctuation, undo, character substitution and phrase substitution. In addition, the keys on the device keyboard are optimally shaped and configured for thumb-based input.

Hand-held Electronic Device with a Keyboard Optimized for Use with the Thumbs

CROSS-REFERENCE TO RELATED APPLICATION

This application is a divisional application of Canadian Patent Application S/N 2,276,697, filed on June 28, 1999.

BACKGROUND OF THE INVENTION

The present invention is directed toward the field of small, hand-held electronic devices such as personal data assistants (PDA's), personal information managers (PIM's), two-way pagers and the like. In particular, the system and method of the present invention provide the user of the hand-held device with the ability to input data with a minimal amount of key strokes and optimized for use substantially with the thumbs.

In a two-way paging system that provides two-way, full text messaging, there is a need to permit the user to initiate messages and to respond to messages in a timely fashion and with text entirely created by the user. In order to keep the form factor of the two-way pager small enough to be worn on the body of the user, such as with a belt clip, the input device needs to be small, have a minimal number of keys and optimized for use with a minimal number of key strokes. Prior art systems have attempted to address these needs by incorporating virtual keyboards or pen-based input systems for user inputs to the device, but such systems require the user to input data in an unfamiliar manner. Additionally, in a small hand-held messaging device, such as a two-way pager, these systems prove awkward to use.

In order to provide a hand-held electronic device that permits a user the opportunity to enter data into an address book, a calendar, a task list, an email message or a similar text file that requires

user-generated data, the instant invention is directed to an input device that is oriented to be used substantially through use of the thumbs. This is accomplished first by providing a keyboard with a minimal number of keys, but with the keys representing the alphabet generally placed in the same order as they would appear on a standard keyboard, such as in a standard QWERTY or a DVORAK keyboard layout. The use of a keyboard layout that is familiar to the user enables the user to immediately use the device without having to hunt for the keys he or she wishes to use.

Although the layout is similar to a standard keyboard, the keys are placed at an orientation and in a particular shape that attempts to maximize the surface area of the thumb hitting the key and to provide the user with a comfortable position of the hands for data input. Also, the orientation encourages input by the thumbs, which the inventors of the instant invention have discovered to be faster and more accurate in small hand-held electronic devices than touch-typing or "hunting and pecking" typing.

An additional feature of the invention is the use of an additional input means for control of functions that might otherwise be controlled by a keyboard that included function keys. To encourage data entry using thumbs and again to minimize the number of keys on the keyboard, the instant invention also includes a thumb-wheel for control of menus for selection of forms and functions relevant to data input. As discussed below, one of the data forms the thumbwheel is implemented to accommodate is a special character set. The thumb-wheel is positioned in close proximity to the keyboard to enable the easily transition from thumb-based typing to thumb control of forms and functions.

In addition to hardware features that encourage optimal data entry through the use of thumbs, there are several software features that are designed to minimize keystrokes and aid in entry of data.

The features of this invention, both individually and collectively, have not, to the knowledge of the inventors, been applied to a small hand-held electronic device that requires user-generated data entry. To permit efficient operation of such devices while keeping the form factor of the device small enough to be worn on the body, there is a general need for a hand-held electronic device that can fit in the palm of the hand and that can be operated substantially with the thumbs.

There is a further need for a keyboard for a palm-size data entry device with keys placed at an angle to optimize operation of the keyboard by the use of the thumbs.

There remains another need for a keyboard with keys that are shaped and sized to maximize contact with the thumbs while minimizing the keyboard area required for such keys.

There also remains a need for an auxiliary input device that is to be operated by the thumb for data inputs forms and function control and that, in conjunction with the keyboard, encourages and permits data entry and management through input performed substantially by the thumbs.

There remains still another need for a software-implemented user interface system that is designed, at least in part, to support and encourage data entry through use of the thumbs.

There remains another need for facilitating entry of special or international characters with the thumbs while limiting the keyboard area.

There remains another need for automatically entering certain common punctuation thereby reducing repetitive key entry and increasing the speed of data entry.

There remains another need for automatically entering certain common phrases or electronic signatures without tediously typing the full phrase or electronic signature.

There remains another need for easily removing automatic features when such features are not desired.

There remains another need for executing a set of common instructions for a particular task with a minimum of user data entry.

SUMMARY OF THE INVENTION

The present invention overcomes the problems noted above and satisfies the needs in this field for a hand-held electronic device with a keyboard optimized for use with the thumbs. In one preferred embodiment of the present invention, the hand-held electronic device is a two-way paging or wireless email device that permits full-text, two-way messaging such as email messaging. The device may also include standard PDA or PIM features such as an address book, an electronic calendar, a task list and other text-based features. Such features require user input of text strings that can be lengthy and that cannot be reduced to pre-determined or "canned" strings. Thus, for such a device, the efficient entry of data in a device meant to fit into the palm of one's hand requires that two goals are achieved. First, the data entry must be relatively easy from a user perspective. This means that the user must be somewhat familiar with analogous forms of data entry and not have to be trained to use the data entry for the hand-held device. Second, the form factor does not permit a large number of keys or keys that are very large. Thus, efficient use of the keyboard space is required and functions that might be able to be performed by a standard keyboard are off-loaded to an auxiliary input device or are performed, through a minimal number of keystrokes that encourage the use of thumb-based data entry.

To accomplish these goals, the invention first optimizes the placement of the keys on the device keyboard. In order to work within the limited space available for the keyboard, it was determined that it was preferable to use keys that were oval or oblong and that were placed at angles designed to facilitate use by thumb typing. An angle for the keys on the right side of the keyboard

and a complementary angle for the keys on the left side of the keyboard are chosen based upon observation of the angle at which a user will orient his or her thumbs while thumb-typing.

The invention also minimizes the number of keys available or required for data input. In the preferred embodiment, only keys for the 26 letters of the English alphabet are available as well as a backspace key, a line feed key, an "alt" key, a "cap" key and a space bar. The alt key enables the user in conjunction the other keys to input numbers and symbols to perform certain functions. The placement of the keys is designed to enhance the user experience while typing with the thumbs by meeting two seemingly opposite goals – minimizing the keyboard footprint while maximizing the likelihood that proper keys will be struck by the thumb-typing user.

Alternative embodiments could include fewer or more keys, including, for example function keys or control keys. The form factor, however, generally limits the number of keys that may be implemented if the goal of facilitating thumb typing is to be met.

The invention also provides additional incentive for the user to use thumb input by providing an input device adjacent to the keyboard, but integral to the overall hand-held device. Although other devices can be used in an auxiliary fashion, the preferred device is a thumbwheel that registers movement of the wheel by measuring the number of indents traversed while rolling the wheel and that also registers as an input the depression or "clicking" of the wheel, which is performed by pressing the wheel toward the back of the pager. This clicking of the wheel is similar to the clicking of a mouse associated with a PC or any other input device that registers the depression of a button. The thumbwheel in the preferred embodiment is placed vertically on the two-way paging device so that the user can easily move his or her thumb from the thumbwheel to the keyboard and back for performing functions and retrieving data forms, such as an email template or address book entry template, for data entry.

Additionally, various software techniques can be implemented to enhance the thumb-typing user's experience in using the device of the instant invention. In the preferred embodiment, for example, the user can change the capitalization of a particular letter simply by keeping a key depressed for a particular length of time without an intermittent release being detected by the keyboard controller.

Therefore, according to an aspect of the present invention there is provided a method of changing the functional state of a key on a keyboard of a handheld wireless communications device comprising the steps of: receiving a predefined functional state associated with a key being depressed on the keyboard; initiating of an altering functional state condition; processing the altering functional state condition; and, modifying the predefined state of the key and submitting for display a modified functional state.

According to another aspect of the present invention there is provided a method of changing the character display of a key on a keyboard of a handheld wireless communications device comprising the steps of: receiving a first input from a depression of a keyboard key, the first input defining a first character associated with the depressed keyboard key; receiving a second input from an auxiliary input device in conjunction with the continual depression of the keyboard key; initiating of an altering character condition and processing the altering character condition; and, modifying the first character and submitting for display a second character associated with the depressed keyboard key.

According to another aspect of the present invention, there is provided a method of optimizing user input in a handheld wireless communications device comprising the steps of: receiving first user input data from a keyboard; substituting the first user input data with predefined formatted data upon the occurrence of a formatting condition and submitting the predefined

formatted data for display; and, restoring and resubmitting for display the first user input data upon the occurrence of a restore input condition from the keyboard.

According to another aspect of the present invention, there is provided a method of modifying the character display of a key on a keyboard comprising the steps of: receiving a first input from a depression of a keyboard key, the first input associating a first character associated with the depressed keyboard key; initiating of an altering character condition upon a continual depression of the keyboard key beyond a predefined period of time; processing the altering character condition; and, modifying the first character and submitting for display a second character.

According to another aspect of the present invention, there is provided a handheld two-way wireless communications device optimally configured for use with the thumbs of a user comprising: a keyboard having a plurality of keys, each key having a predefined functional state; an auxiliary input device positioned in relative proximity to the keyboard, the auxiliary input device having measurable rotatable and depressible input movements to initiate an altering functional state condition; a display positioned adjacent to the keyboard; and user interface software residing in the communications device for providing functionality to inputs received from the auxiliary input device and the keyboard.

According to another aspect of the present invention, there is provided a handheld two-way communications device optimally configured for use with the thumbs of a user comprising: a keyboard having a plurality of keys; a thumbwheel device positioned in relative proximity to the keyboard, the thumbwheel device having a corresponding thumbwheel logic circuitry in the communications device for detecting measurable rotatable and depressible input movements from the thumbwheel device; a display positioned adjacent to the keyboard; and, user interface software residing in the communications device for providing functionality to inputs received from the

thumbwheel device and the keyboard; wherein an altering functional state condition is triggered by the occurrence of input via the thumbwheel device in conjunction an input from the depression of a keyboard key.

According to another aspect of the present invention, there is provided a handheld two-way wireless communications device optimally configured for use with the thumbs of a user comprising: a keyboard having a first set of keys and a second set of keys, wherein the first set of keys is positioned a positive predetermined degree relative to a vertical reference and the second set of keys is positioned a negative predetermined degree relative to the vertical reference; an auxiliary input device positioned in relative proximity to the keyboard; a display positioned adjacent to the keyboard; software residing in the communications device for providing functionality to inputs received from the auxiliary input device and the keyboard.

According to another aspect of the present invention, there is provided a method of composing an email address in an email address field comprising the steps of: receiving a first input; receiving a first delimiter trigger signal; determining whether the first input contains a first email delimiter; and, substituting the first delimiter trigger signal with a first email delimiter upon a positive determination or with a second email delimiter upon a negative determination.

A primary advantage of the present invention is that it enables efficient and user-friendly data entry into a palm-sized electronic device by maximizing the potential for user data entry through thumb typing.

These are just a few of the many advantages of the present invention, as described in more detail below. As will be appreciated, the invention is capable of other and different embodiments and its several details are capable of modifications in various respects, all without departing from the

spirit of the invention. Accordingly, the drawings and description of the preferred embodiment set forth below are to be regarded as illustrative in nature and not restrictive.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention satisfies the needs noted above as will become apparent from the following description when read in conjunction with the accompanying drawings wherein:

FIG. 1 is a block diagram of a two-way, full-text, messaging device incorporating a keyboard and an auxiliary data entry device;

FIG. 2 is a frontal view of the hand-held device showing the shape and placement of the keys on the keyboard and the auxiliary input device;

FIG. 3 is a diagram of showing the shape, size and placement of the keys on the keyboard;

FIG. 4 is a diagram of the control circuitry for the thumbwheel;

FIG. 5 is a general flow diagram of the major steps of the keyboard-related aspect of the invention;

FIG. 6 is a general block diagram of the major subsystems involved in the process described in FIG. 5;

FIG. 7 is a general flow diagram of the major steps of the thumbwheel-related aspect of the invention; and,

FIG. 8 is a general block diagram of the major subsystems involved in the process described in FIG. 7.

DETAILED DESCRIPTION OF THE INVENTION

Referring now to the drawings, FIG. 1 is a block diagram of the major subsystems and elements comprising a palm-sized, mobile, two-way messaging device that preferably incorporates the invention. In its broadest terms, the messaging device includes a transmitter/receiver subsystem 100 connected to a DSP 200 for digital signal processing of the incoming and outgoing data transmissions, power supply and management subsystem 300, which supplies and manages power to the overall messaging device components, microprocessor 400, which is preferably an X86 architecture processor, that controls the operation of the messaging device, display 500, which is preferably a full graphic LCD, FLASH memory 600, RAM 700, serial output and port 800, keyboard 900 , thumbwheel 1000 and thumbwheel control logic 1010. In its intended use, a message comes via a wireless data network, such as the Mobitex network, into subsystem 100, where it is demodulated via DSP 200 and decoded and presented to microprocessor 300 for display on display 500. To access the display of the message, the user may choose from functions listed under a menu presented as a result of user interaction with thumbwheel 1000. If the message is an email message, the user may chose to respond to the email by selecting "Reply" from a menu presented on the display through interaction via thumbwheel 1000 or via menu selection from keyboard 900. In typing the reply, the user can use keyboard 900 to type full text message replies, or insert pre-determined or "canned" response by using either a particular keystroke pattern or through pulling down pre-determined text strings from a menu of items presented on display 500 through the use of thumbwheel 1000. When the reply to the message is composed, the user can initiate the sending of the message preferably by interaction through thumbwheel 1000, or alternatively, with less efficiency, through a combination of keyboard 900 keystrokes. When the microprocessor 300 receives an indication that the message is to be sent, it processes the message for transport and, by

directing and communicating with transmitter/receiver subsystem 100, enables the reply message to be sent via the wireless communications data network to the intended recipient. Similar interaction through I/O devices keyboard 900 and thumbwheel 1000 can be used to initiate full-text messages or to forward messages to another party. Also, the keyboard 900 and thumbwheel 1000 can be used to permit data entry to an address book resident on the messaging device, or an electronic calendar or log book, or any other function on the messaging device requiring data entry. Preferably, the thumbwheel is a thumbwheel with a push button SPST with quadrature signal outputs, such as that manufactured by Matsushita Electronic Components Co. Ltd. as part number EVQWK2001.

Figure 2 is a front view of messaging device 10 that incorporates the invention. Shown in Figure 2 are a plurality of letter keys 901, and specialized keys 902, 903, 904 and 905 and space bar 906. Also shown is thumbwheel 1000 in its vertical orientation and in association with display 500 and keyboard 900. In a preferred embodiment, 902 is the alt key, 903 is the cap key, 904 is the line feed (blank space) key and 905 is the backspace key.

Figure 3 is a view of a subset of the letter keys 901, showing the dimensions and relative position of the keys. Shown also is the point 950 that marks the center of keyboard 900, key dimensions 970, 971, 972 and 973, as well as angle 960 and the rho value 965, representing curvature of a letter key 901. In investigating optimal key placement on the keyboard, it was determined that the keys should be placed at an angle 960 relative to vertical that facilitated easy typing using thumbs. That angle is preferably positive 40 degrees relative to vertical for keys on the right side of the keyboard (where 950 is the center of the keyboard) and negative 40 degrees for the keys on the left side of the keyboard, although complementary angles ranging from 20 degrees to 70 degrees could also be used to accomplish the goal, albeit less optimally, of facilitating thumb typing.

Also as shown on Figures 2 and 3, the keys are dispersed across keyboard 900 evenly so that there is

sufficient space between the keys to decrease the opportunity for multiple keys being depressed while thumb typing. Additionally, the keys are sized appropriately given the footprint of the messaging device and the keyboard 900. In its preferred embodiment, the messaging device 10 measures across its face 64 mm by 89mm, which does not leave much room for keyboard 900 and display 500. In the preferred embodiment, keyboard 900 occupies over half of the face of the messaging device 10.

The key shape and dimensions are also key components of the invention. In order to maximize the surface area of the key that a thumb would hit, the keys are preferably oval, and have a rho 965 defining the curvature of the key of .414, although values may range higher or lower. Other rho values will lead to an acceptable, but not as optimal or aesthetically pleasing shape of keys 901. As to the key dimensions, the width 970 of the key 901 is 4.8 millimeters (971 representing the radius of half that value, 2.4 mm) and the length (or height) 972 of the key 901 is 7 millimeters (973 representing the radius of half that value, 3.5 mm).

Figure 4 is the logic circuitry 1010 associated with thumbwheel 1000. Thumbwheel 1000 outputs quadrature signals phase A 1021 and phase B 1022, which are processed by D flip-flops 1031 and 1032 to present signals 1041 W_UP and 1042 W_DN to microprocessor 400. Signals 1041 and 1042 represent, respectively, a user rolling the thumbwheel up and rolling the thumbwheel down. Preferably, another detectable input movement from the thumbwheel is desirable. One such input movement implementation would produce an additional input signal derived from pushing the thumbwheel toward the rear of the device. Hence, the thumbwheel of the present invention has preferably measurable rotatable and depressible input movements. Although the description that follows is specifically relating to a thumbwheel input device, it is to be understood that other suitable

thumb-based auxiliary input devices having multiple input detectable movements are envisioned and well within the scope and spirit of the present invention.

In a broad aspect of the invention, input via keys on the keyboard in conjunction with software residing in the flash memory 600 also facilitates the device's operation with the user's thumbs. In another broad aspect of the invention, input via the thumbwheel 1000 and a key on the keyboard in conjunction with software residing in the flash memory 600 further increases the ease, performance and functionality of the device. These two broad aspects and their specific embodiments will now be described.

Figure 5 is a general flow diagram of the major steps 2080 of the aforesaid keyboard-related aspect of the invention. The first step 2000 is the input via depressing of a key having a predefined functional state in combination with another key on a keyboard. Depending on the combination of keys depressed, an altering functional state condition is then initiated at step 2020. The altering functional state condition is processed at step 2030. Next, the predefined functional state of the key is modified at 2040. In step 2050, the modified functional state is interpreted, and information is displayed accordingly at 2060.

Figure 6 is a general block diagram of the major subsystems involved in the process described in Figure 5. In the preferred embodiment of the device, when a key is depressed its predefined functional state 901 is given to the microprocessor 400 and then to the operating system 910. The operating system is responsible for accepting and dispatching to a specialized application 915 any user input and for providing services to control the display. The modifying step is achieved by a user interface software engine 930 processing the altering functional state condition 905 received from the operating system 910 via a specialized application 915. The modified functional state 905' is then passed to the operating system 910. The operating system then acts on the

information in a predetermined manner so as to produce a display 906' representing the modified functional state of the depressed key.

In one preferred embodiment of the invention, there is a plurality of specialized applications 915 that manage different major features of the device. In such a preferred embodiment, the device would include the following specialized applications: a graphical ribbon application, message compose application, transport application, address book application, device options application, cryptographic application and database application. Other applications are well within the scope and spirit of this aspect of the invention. The applications function directly with the operating system while the user interface engine (albeit another application), in general, provides further specialized functional services to these applications. The user interface engine comprises, in essence, a plurality of modules - each module executing at least one different function, and either alone or in combination with another module, providing additional functionality to the applications that use its services. For instance, the user interface engine provides an input field that other applications such as the message compose application can use to display user input. Additionally, the user interface engine receives, stores, manages and outputs data in a consistent manner to readily perform the various specific features to be described below. This scheme has proven to be very advantageous in view of the fact the operating system and the specialized applications may then be limited in their coding overhead and hence allows them to be very robust. Some of the unique functions will be described in more detail below.

Keypad Aspect Generally

The process logic detail is now described for any keypad event resulting from user input. When any key on the device is depressed, a keypad event containing the parameters of the particular

key is communicated to the microprocessor. In this instance, "any key" includes the thumbwheel in addition to the keys on the keyboard. Preferably, the parameters indicate that a keypad event has occurred and which key has been depressed. The microprocessor communicates the keypad event information to the operating system. The operating system takes this keypad event information and notifies a specialized application of the keypad event information. Since, preferably, the specialized application is blocked waiting for such a keypad event, the application is awakened, examines the keypad event information, and passes the keypad event information to the user interface engine for further processing. In essence, the keypad event information is passed to the user interface engine from the operating system via the specialized application: the specialized application in this case acts as a conduit for information flow. In other instances, the specialized application may preprocess the keypad event information before passing the same to the user interface engine. Based on the keypad event information provided to the user interface engine, the user interface engine then acts in a predetermined functional manner. The predetermined functional manner preferably includes processing the keypad event information and updating stored display data so as to provide output instruction data. The user interface engine submits the output instruction data to the operating system. The operating system then acts according to the instructions in the instruction data, in most cases, by submitting for display a representation of the keypad event. The output instruction data would preferably include instructions as to where characters are to be displayed within the display and a character code reference value indicating which character should be displayed on the display. In other cases, a plurality of character code references may be provided for producing a display of a string of characters. The preferred character code reference is the standard Latin One Code Set. The operating system would simply execute the instructions by mapping the appropriate characters from its internal bitmap of characters into the appropriate locations on the display.

Auxiliary Input Device Aspect

Figure 7 is a general flow diagram of the major steps of the auxiliary input device-related aspect of the invention. The process 2080' begins with a first input 2000' via the depressing of a key on a keyboard having a predefined functional state. Next, a second input 2010' from the auxiliary input device in conjunction with the depression of the key is provided. This combined effect of the first and second input initiates an altering functional state condition step 2020'. The altering functional state condition is processed 2030' and the predefined functional state of the key is modified 2040'. The modified functional state is then submitted 2050' for display. Additional modified functional states are accomplished by receiving further inputs 2060' from the auxiliary input device while the depressed key is continued to be depressed. Figure 8 is a block diagram of the major subsystems involved in the process described in Figure 7. Figure 8 is similar to Figure 6 save for input from the auxiliary input device subsystem now contributing to the altering functional state condition. This aspect of the present invention has proven extremely useful in optimizing the functionality of the handheld device by operation of the device with only the user's thumbs.

International Character-Scrolling Feature

Turning now to a specific embodiment of the aforesaid auxiliary input device-related aspect of the invention, the user interface engine, the thumbwheel and at least one key on the keyboard may be utilized to insert international characters into a user input field such as the input field provided by the user interface engine. In general and preferably, the user first presses and holds down a key with the desired associated international characters while the thumbwheel is rolled. For each rolled input from the thumbwheel, a different international character associated with the depressed key is

outputted on the display as a substitute for the previous character displayed. There is preferably a plurality of international characters or symbols associated with every key on the keyboard. When the desired character is displayed, the user releases the depressed key and the desired character remains on the display. Traditional means to select such international characters require a number of additional input steps or the availability of additional keys on the keyboard – requirements undesirable with the present handheld device.

With reference to Figure 8, the international character-scrolling feature begins with the depression of a key on the keyboard 900 in conjunction with input from the thumbwheel 1010. The input from the thumbwheel is preferably accomplished by rolling the thumbwheel in either the up or down direction. The combined effect of inputs 901 from the key and the thumbwheel produce an altering functional state condition or a special keypad event 905. The parameters associated with this special keypad event 905 include data descriptors that will indicate whether the thumbwheel was either rolled up or down and which key on the keyboard was depressed. The user interface engine 930 processes the special keypad event information by indexing through an array containing references to the international characters associated with the depressed key. The processing incrementally indexes up or down the array (depending on the thumbwheel scroll direction parameter), stores the current array position where the indexing stopped and retrieves the character reference code information stored in the current array position. Preferably, at each index increment, output instruction 905' is submitted to the operating system 910 for display on the display 500. The output instruction preferably includes the reference character code and the location of the character to be displayed on the display. Based on this output instruction, the operating system outputs at the appropriate location a bitmap image 906' for a given reference code from its internal bitmap. If the user chooses to see the next international character associated with the depressed key, the user

provides an additional rolling input while continuing to depress the key. This produces a second special keypad event thereby initiating the altering functional state process. Consequently, the above process is repeated. The user interface engine processes the keypad parameters from the second special keypad event by further indexing in the appropriate direction by one element from the previously stored position within the array. Again, the user interface engine submits to the operating system sufficient information to display the next associated international character retrieved from the array. Further inputs from the thumbwheel would simply repeat the processing until all international characters associated with the depressed key are indexed through after which time the indexing returns to the top of the array.

Although the above description refers to modifying the default character of the key depressed with an international character it is to be understood that the application of this aspect of the invention can equally be applied to modifying the functional state of the depressed key. For instance, a certain special keypad event defined by a certain key depressed in conjunction with input received from the thumbwheel could trigger a macro to be executed thereby altering the functional state of the depressed key. In this instance, a macro is a single instruction provided by the scrolling feature that expands automatically into a set of instructions to perform a particular task. In another instance, the altered functional state of a key could produce any one of the following: a larger font for the display; change the default font type; change subsequent text to bold or italics; cause certain text to blink in a received email transmission; and reverse image certain text. In another instance, the altered the functional state of a key could provide the user with "canned" or pre-set messages that could be readily be used for the purposes of quickly responding to a received message. In another instance, the context of the current active display area of the device may either add, or alternatively, suppress, associated altered states. For example, when a user is in the reply-to-message screen display,

depression of the letter "M" with the scrolling input from the thumbwheel would produce an ellipsis on the display as the altered functional state. In contrast, the same key and thumbwheel sequence would instead result in a macro sequence to execute in the options screen display.

Capitalization

Turning to another software-related feature that aids in the device 10 being optimally used for thumb typing is a capitalization feature implemented via software. If a user depresses a key 901, the operating system detects a key down event. If the key is released after a period of time, the operating system detects a key up event. If after a key down event, a certain period of time elapses before a key up event is detected, the operating system determines that a key repeat event has occurred representing a situation where a user has continued to depress a key without releasing it. A key repeat event is then treated by application software residing in either flash 600 as an event that requests the capitalization of the key previously depressed. This feature disables the key repeat feature and substitutes instead a capitalization feature based on the length of a key press. The timing of the key scanning to determine whether a key has been released can be set to permit a slower keyboard response or a faster keyboard response, depending upon user experience or preferences. Although the capitalization function preferably works only to change the state of a letter to a capital, it alternatively could operate to change a capital letter to a lower case letter. The actual display image is changed by the operating system mapping appropriate bitmapped characters as instructed by the user interface engine. As alternatively implemented, the continued depressing without release of a letter key could result in a key oscillating between upper case and lower case, depending on the length of time the key is depressed.

Automatic Formatting Features

Now follows a description of another software-related aspect of the present invention that is implemented to specifically minimize the number of keystrokes required as input by the user in using the device. In some cases, two software engines handle this software-related aspect, namely the user interface engine and an autotext engine. The user interface engine by using the above-mentioned functional modules facilitates presenting common interface elements such as menus, lists and textboxes. The autotext engine is another software application that extends the user interface engine and makes use of a database to perform text insertions and other related activities. The autotext engine is preferably implemented as a user interface engine element (such as a choice box or list or standard edit element). This arrangement of being a user interface engine element allows specialized applications to incorporate a buffer that performs autotext substitution as they would a standard input field. However, because of the substantive amount of code associated with the autotext engine and the engine's need to be configured with a list of replacement strings, the engine is preferably implemented as a separate module. One specific advantage that flows from the specific embodiments of the engines as described herein is that there is a substantial reduction of input formatting by the user, since automatic formatting can be triggered by a small number of keystrokes, or by predefined key sequences entered one key at a time.

Autopunctuation

Another embodiment of the user interface engine is the "autopunctuation" feature of the device. If the user enters one of a number of predefined character sequences, the user interface engine will automatically insert a period (full stop) in the user input field. In this manner the user does not need to use the alt key to enter a period. An example of one such character sequence would

be <character><space><space>, after such character sequence the user interface engine will instruct the operating system to replace the first <space> with a full stop and to capitalize the next character entered by the user. For clarity, in this description: <space> represents a blank space user input; <character> represents any character user input; and, <period> represents user input of the character ".". Because of bandwidth considerations when the device transmits data to the wireless network, it is preferable that only one space, instead of two, is present between the period and the first character of the next sentence. Alternatively implemented, instead of replacing the first <space> in the aforesaid character sequence, a period is inserted after the <character> and the next character entered by the user is capitalized thereby producing two spaces between the period and the next character. In another embodiment of this autopunctuation feature, if the user enters one of a number of predefined character sequences that is used as a capitalization signal, such as <period><space>, the user interface engine will process the event and capitalize the next character entered by the user. Alternatively implemented, the predefined character sequence could be <period><space><space>. To further enhance this feature, autopunctuation can be undone easily, as described below.

Character Substitution

Another embodiment of the user interface engine is the "intelligent character substitution" feature of the device. In fields designed for certain types of input, the user interface engine will automatically substitute appropriate characters if inappropriate characters are entered. For example, in a field designed to hold an email address, a method of composing an email address is provided by the character substitution feature of the present invention for email address taking on the general form <characters>@<characters>.<characters>. In the instance, it is assumed blank spaces are unacceptable characters for an email address field. The method begins with the user interface engine

receiving from user input a character set representing a certain portion of the email address. If a first delimiter trigger signal, preferably a first blank space, is received from the user input, the engine confirms that no at-sign character ("@") has been entered in the already received character set. If no, the engine substitutes the first delimiter trigger signal with a first email delimiter, preferably the at-sign character. If yes, the engine substitutes the first delimiter with a second email delimiter, preferably a period character. For any subsequent blank space input received, the engine substitutes preferably another period character.

Another example of the character substitution feature occurs when entering data in a field designed to hold numeric data. If the user presses an alphabetic key whose corresponding alt character is a numeral, the user interface engine will recognize this and instruct the operating system to automatically insert the numeral into the field, even if the user had not pressed the alt key.

Phrase substitution

The first embodiment of the autotext engine provides an application service for error correction and phrase substitution. By entering pairs of strings into an autotext database, the user can associate "original strings" with "replacement strings". The autotext engine also defines an autotext input field that applications may use to access the functionality of the autotext engine. The autotext input field may be of variable length typically up to the length of an email message. As the user is entering data into an autotext input field, each time a space is entered, the autotext engine will search backwards from the current cursor position to extract the last word from the autotext input field. The autotext engine will then efficiently search through the autotext database for an original string that exactly matches the last word. If such a string is found, the last word in the autotext input field will automatically be replaced with the corresponding replacement string from the database. This

phrase substitution feature could be used to correct common typing errors (for example, "teh" could be automatically converted to "the") or to facilitate the entry of common phrases (for example, "fyi" could be automatically converted to "for your information"). This feature could also automatically insert a regularly used international character or symbol in a message thereby reducing the instances when the international character-scrolling feature needs to be used. In this latter scenario, the original string could be "jr" and the associated replacement string could be "Jürgen". The international character would be entered as described before for the international character-scrolling feature.

The phrase substitution feature could advantageously be used for electronic signatures to be appended to the end of an email message composed by the user on the device. Advantageously, a variety of different electronic signatures can be stored in the autotext database. In this manner, the user may chose the appropriate signature from his set of signatures for the particular context of the email message about to be transmitted. For instance, in one case the user in an informal message may sign off "Take Care & Cheers" with the original string "tc" while in a legally sensitive message the same user may sign off "Solicitor-Client Privileged Communication" with the original string "scp". In this manner, there is no need for any electronic signature field insertion at the gateway of the wireless network and the user may amend and modify his electronic signatures without network interaction. In the preferred embodiment, the replacement string is case sensitive in that any preset capitalization in the replacement string appears when displayed and cannot be forced into a lower case by the case status of the original string. However, if the first character of the original string is upper case, then preferably the first character of the replacement is forced into an upper case as well.

In another embodiment, the autotext engine preferably includes a macro functionality that allows a user to specify special character codes in the replacement string that will be treated specially

when the replacement string is entered into the autotext engine. These codes are preferably divided into two distinct sets: control character codes and dynamic data codes.

In the control character code situation, some character sequences in the replacement string will be converted to control characters when the original string is replaced by the substitution feature. Examples of these sequences would be "%b" and "%B" which convert to a backspace character and a delete character respectively. Therefore, for an open bracket "(" to be displayed, the original string could be defined as "br" and the replacement string as "(%B". Likewise, for a close bracket ")", the original string could be "rb" and the replacement string "%b)".

In the dynamic data code situation, certain character sequences in the replacement string will be converted to dynamic data when the original string is replaced. Examples of these sequences would be "%t" and "%d" wherein the codes are converted into the current time and current date respectively.

Like the intelligent character substitution feature, autotext replacement can be undone easily, as described below.

Undo Feature

As mentioned above, all automatic formatting features of the user interface and autotext engines such as autocapitalization, autopunctuation and phrase substitution may be overridden by what is referred to the "easy undo" feature herein. The automatic formatting features have undo capabilities that are triggered by backspacing over a significant character. As such, if the user backspaces over one of these significant characters, the formatted data is replaced with the raw data as it existed prior to automatic formatting.

For instance, the user may type "fyi" into an autotext input field such one implemented in the message compose screen. If then the user types <space> and "fyi" is in the autotext database, the autotext engine will replace "fyi" with "for your information" in the input field as earlier described in phrase substitution. If the user does not desire this substitution, the user can preferably backspace over the last character of the displayed string "for your information". Consequently, the autotext engine will restore the contents of the input field to "fyi" as they were prior to the phrase substitution formatting operation. In this instance, the significant character was the last character "n" in the phrase "for your information".

If the user types a sequence of characters that results in an automatic formatting operation, and backspaces to activate the easy undo feature, the appropriate engine will retain the fact that an undo operation took place at that significant character position. Consequently, if the user retypes the same sequence of characters, then the automatic formatting will not be applied for a second time. This allows the user to suppress an automatic formatting feature when desired.

For example the user may suppress the autopunctuation feature from inserting a full stop automatically. As described above, if the user types <space><space> in an input field, a full stop will be inserted by the user interface engine for the first <space>. If the user does not desire a full stop in that location, the user would then backspace over the full stop character and retype <space><space>. The user interface engine will not replace the full stop as it did the first time.

Navigation Aids

The device software provides a number of navigation aids for most user input tasks. The navigation aids maximize the likelihood that there will be a graphical interface that the user finds intuitive. The aids allow most input tasks to be performed completely using either the keyboard or

the thumbwheel. The navigation aids are preferably incorporated into choice boxes and menus of the device software. A choice box is graphical interface wherein only one selection is displayed at one time. The other possible selections are displayed one at a time by a navigation means. A menu, on the other hand, is a graphical interface wherein all selections for a particular input are displayed at one time. There are other instances where the navigation aids may be used and such applications are within the scope of the invention.

As a first aid, the user may cycle through all available options in a choice box by pressing the space bar. For instance, in a box containing the months of the year, pressing the space bar repeatedly will change the selection from "January" to "February" to "March", etc. A keypad event, defined by the location of the cursor being in a choice box and the depression of the space bar, would trigger the altering functional state process. As a result, the user interface engine would process the event by indexing through a choice array associated with the choice box and provide to the operating system instructions to display the next available option.

As a second aid, the user may also cycle through the data by pressing a control key and rolling the thumbwheel. The control key is preferably the "alt" key 902. Other control keys are possible and within the scope of the invention. This navigation method allows more flexibility for the user because the user can cycle through data in either direction in the choice array. This navigation is similar to the aforesaid international character-scrolling embodiment. Instead of international characters associated with the key depressed, the software displays a list of options. Preferably, in the downward scrolling direction, the user interface engine will move through the list of choices in the same direction as it would have if the space bar had been pressed. In the upward scrolling direction, the user interface engine will move backwards through the list of choices (e.g. "March" to "February" to "January", etc.). Hence, the alt key's defined function of producing for

display an alternate character for a key has been altered by the keypad event triggered by the thumbwheel and the alt key inputs. In the case of a menu display, the user can move the cursor between items in the menu by depressing a character key. If a menu item exists beginning with the character, the cursor will be positioned automatically over the first item in the menu beginning with that character. If there are multiple menu items beginning with that character, pressing the key repeatedly will cycle through those menu items. When the cursor is positioned over the desired item, the user can select it by pressing the Enter key. If the user decides that he does not want to select any menu item, but instead return to his previous position in the application, he can again select the "Hide Menu" menu item or press the backspace key, and the menu will be dismissed.

Another navigation aid is available when the cursor is positioned over a specific choice box that changes options by allowing the user to cycle through the options in the choice box by only rolling up or down without depressing the alt key. Similarly, when a menu is displayed, the user can move the cursor up or down in the menu by rolling the thumbwheel, respectively, up or down. When the cursor is positioned over the desired menu item, the user can select it by pressing the roller wheel toward the back of the device (called clicking the wheel). If the user decides that he does not want to select any menu item, but instead desires to return to his previous position in the application, preferably, he can select the "Hide Menu" item, and the menu will be dismissed. Preferably, the user can also dismiss the menu by pressing the backspace key.

Another navigation aid utilizes the uniqueness of the first character of each choice listed in a choice box. If the user depresses a key and there exists a possible choice beginning with the character represented by the key, that choice will automatically be selected without any further input required by the user. If there are multiple choices beginning with that letter, pressing the key repeatedly will cycle through the available choices. For example, in a month choice box, pressing 'j'

will first select "January", then "June" and then "July". Alternatively implemented, rapid keystrokes could result in a narrower search. For instance, rapidly pressing "ju" would result in "June" and then "July".

Having described in detail the preferred embodiments of the present invention, including the preferred methods of operation, it is to be understood that this operation could be carried out with different elements and steps. This preferred embodiment is presented only by way of example and is not meant to limit the scope of the present invention that is defined by the following claims.

THE EMBODIMENTS OF THE INVENTION IN WHICH AN EXCLUSIVE PROPERTY OR PRIVILEGE IS CLAIMED ARE DEFINED AS FOLLOWS:

1. A method of changing the character display associated with a key on a mobile device having a display, a keyboard, and a thumbwheel input device, the method comprising the steps of:

(A) selecting a key on the keyboard and displaying a first character on the display;

(B) holding down the selected key;

(C) while the key is held down, rolling the thumbwheel input device and, in response to rolling the thumbwheel input device, displaying a second character on the display in place of the first character; and

(D) while the key is held down, continuing to roll the thumbwheel input device and displaying a third character on the display in place of the second character.

2. The method of claim 1, further comprising the step of:

(E) while the key is held down, continuing to roll the thumbwheel input device and displaying a fourth character on the display in place of the third character.

3. The method of claim 2, wherein the second, third and fourth characters are international characters associated with the selected key.

4. The method of claim 1, further comprising the step of:

(E) while the key is held down, continuing to roll the thumbwheel input device and displaying, in succession, the first character on the display in place of the third character, the second

character on the display in place of the first character, and the third character on the display in place of the second character.

5. The method of claim 1, wherein the second character is an international character associated with the selected key.

6. The method of claim 1, wherein the second and third characters are international characters associated with the selected keys.

7. A method of changing the character display associated with a key on a mobile device having a display, a keyboard, and a thumbwheel input device, the method comprising the steps of:

(A) selecting a key on the keyboard and displaying a first character on the display;

(B) holding down the selected key;

(C) while the key is held down, if the thumbwheel input device is rolled in a first direction, then displaying a second character on the display in place of the first character; and

(D) while the key is held down, if the thumbwheel input device is rolled in a second direction, the displaying a third character on the display in place of the first character.

8. The method of claim 7, further comprising the steps of:

storing an array of characters in memory in the mobile device, wherein the array of characters includes the first, second, and third characters; and

in response to the rolling of the thumbwheel, scanning the array of characters in order to select a character to display.

9. A method of changing the character display associated with a key on a mobile device having a display, a keyboard, and an auxiliary input device, the method comprising the steps of:

(A) selecting a key on the keyboard and displaying a first character on the display, wherein the first character is displayed using a first display characteristic;

(B) holding down the selected key; and

(C) while the key is held down, activating the auxiliary input device and changing the display characteristic of the first character from the first display characteristic to one of a plurality of second display characteristics.

10. The method of claim 9, wherein the display characteristic is font size, and the first display characteristic is a first font size and the one of a plurality of second display characteristics is a second font size.

11. The method of claim 9, wherein the display characteristic is font type, and the first display characteristic is a first font type and the one of a plurality of second display characteristics is a second font type.

12. The method of claim 9, wherein the display characteristic is style type, and the first display characteristic is a normal style type and the one of a plurality of second display characteristics is an italic style type.

13. The method of claim 9, wherein the display characteristic is style type, and the first display characteristic is a normal style type and the one of a plurality of second display characteristics is a bold style type.

14. The method of claim 9, wherein the auxiliary input device is a thumbwheel.

15. A method of changing the character display associated with a key on a mobile device having a display, a keyboard, and a thumbwheel input device, the method comprising the steps of:

(A) selecting a key on the keyboard and displaying a first character on the display;

(B) holding down the selected key;

(C) while the key is held down, rolling the thumbwheel input device and displaying a first set of characters on the display in place of the first character; and

(D) while the key is held down, rolling the thumbwheel input device and displaying a second set of characters on the display in place of the first set of characters.

16. A method of character substitution in a handheld mobile device having a display, an auxiliary input device, and a keyboard, comprising the steps of:

associating a plurality of characters with at least one key on the keyboard;

selecting the key and displaying a first character of the plurality of characters on the display; and

while holding the selected key down, activating the auxiliary input device and substituting a second character of the plurality of characters for the first character on the display.

17. The method of claim 16, further comprising the step of:

while continuing to hold the selected key down, activating the auxiliary input device and substituting a third character of the plurality of characters for the second character on the display.

18. The method of claim 16, wherein the keyboard is a QWERTY keyboard, and a plurality of keys in the QWERTY keyboard each are associated with a plurality of characters.

19. The method of claim 16, wherein the plurality of characters includes at least one English language character and a plurality of international characters.

20. The method of claim 16, wherein the auxiliary input device is a thumbwheel input device.

21. The method of claim 16, wherein the auxiliary input device includes a directional component for scrolling through the plurality of characters and a selection component for selecting a particular character.

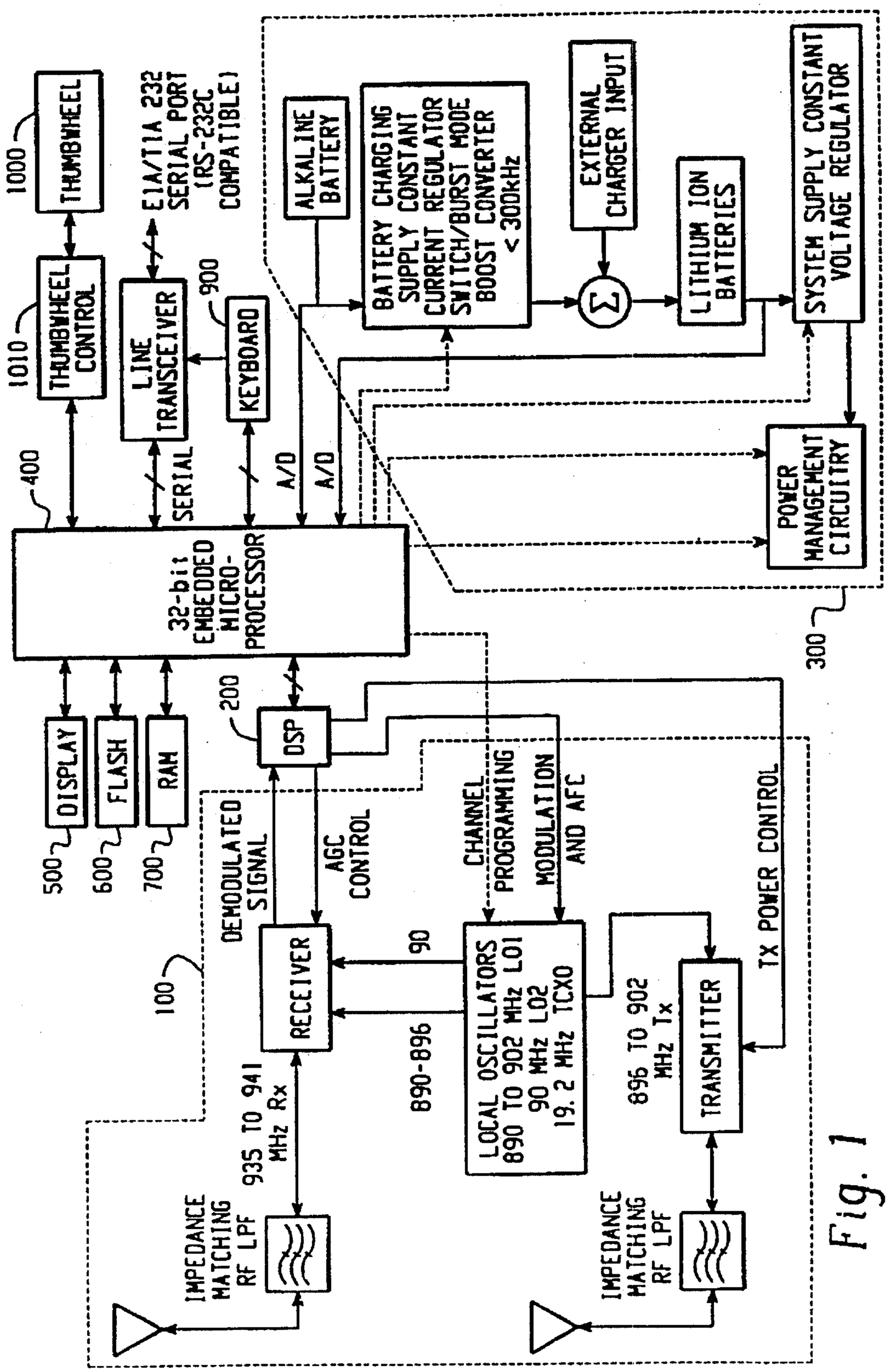

1000
THUMBWHEEL
E1A/T1A 232
SERIAL PORT
(RS-232C
COMPATIBLE)
1010
THUMBWHEEL
CONTROL
LINE
TRANSCEIVER
900
KEYBOARD
400
32-bit
EMBEDDED
MICRO-
PROCESSOR
SERIAL
A/D
A/D
ALKALINE
BATTERY
BATTERY CHARGING
SUPPLY CONSTANT
CURRENT REGULATOR
SWITCH/BURST MODE
BOOST CONVERTER
< 300kHz
EXTERNAL
CHARGER INPUT
LITHIUM ION
BATTERIES
SYSTEM SUPPLY CONSTANT
VOLTAGE REGULATOR
POWER
MANAGEMENT
CIRCUITRY
300
DISPLAY
FLASH
RAM
500
600
700
200
DSP
DEMODULATED
SIGNAL
AGC
CONTROL
CHANNEL
PROGRAMMING
MODULATION
AND AFC
TX POWER CONTROL
100
RECEIVER
90
LOCAL OSCILLATORS
890 TO 902 MHz LO1
90 MHz LO2
19.2 MHz TCXO
890-896
896 TO 902
MHz Tx
TRANSMITTER
935 TO 941
MHz Rx
IMPEDANCE
MATCHING
RF LPF
IMPEDANCE
MATCHING
RF LPF

Fig. 1

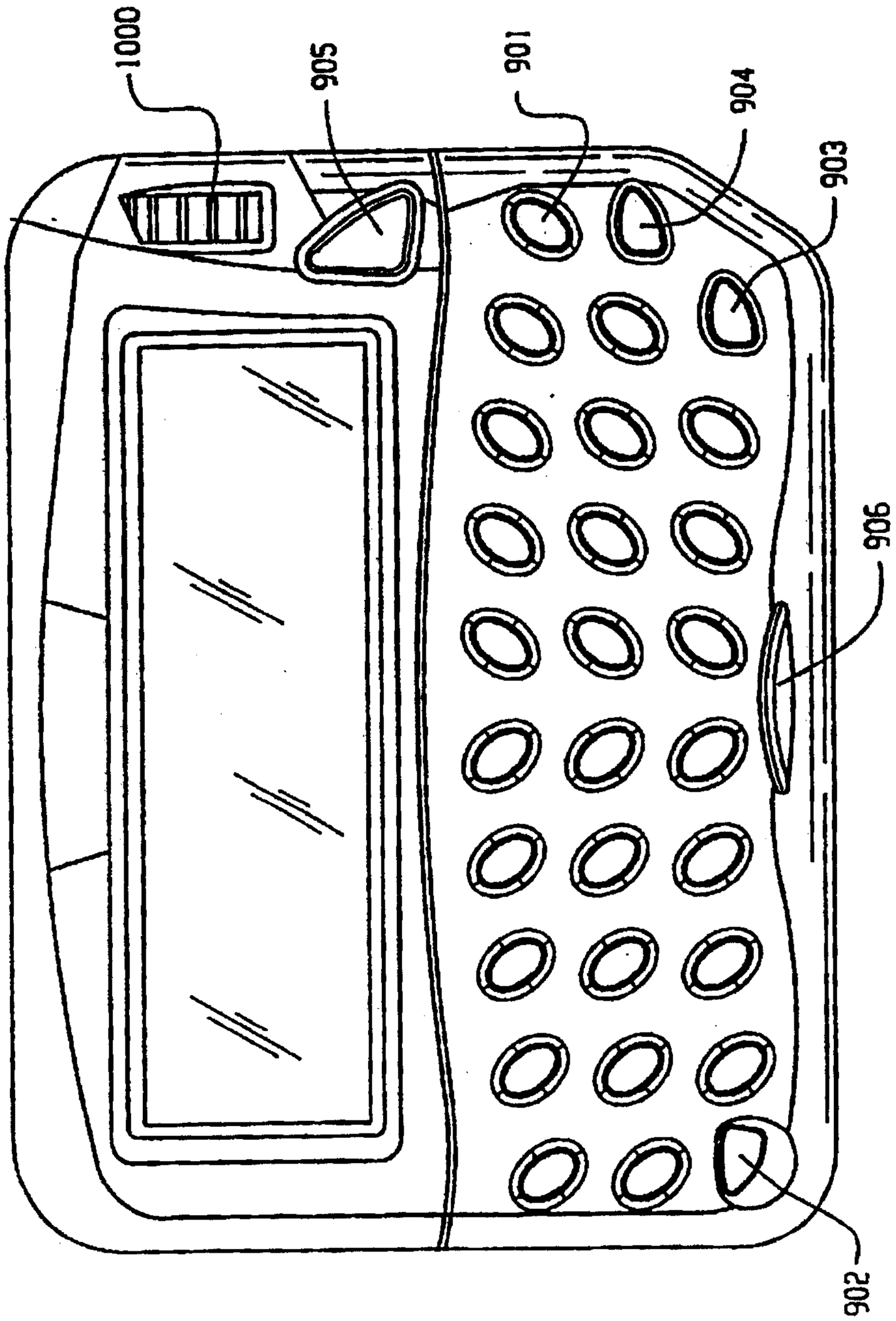


Fig. 2

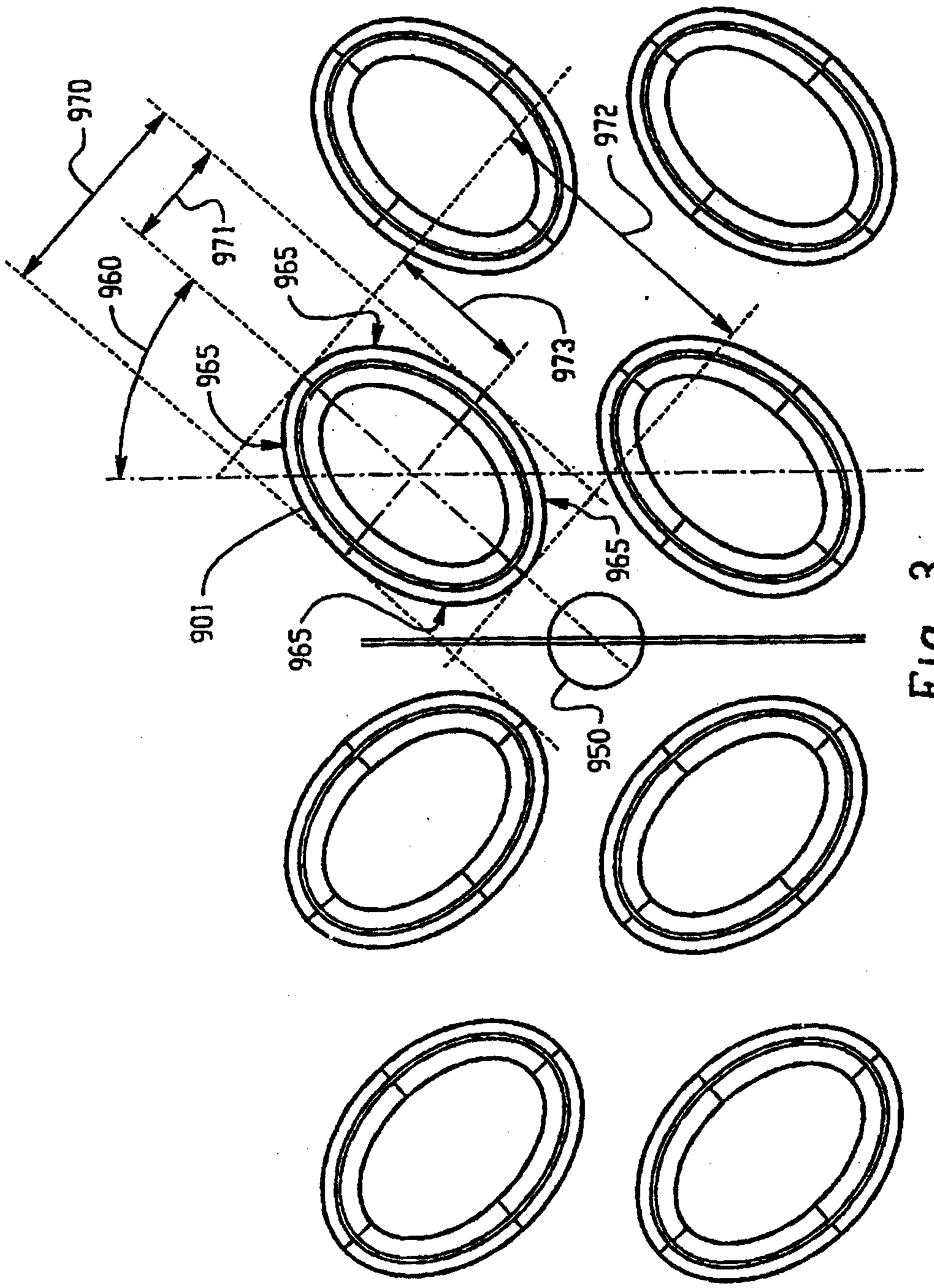


Fig. 3

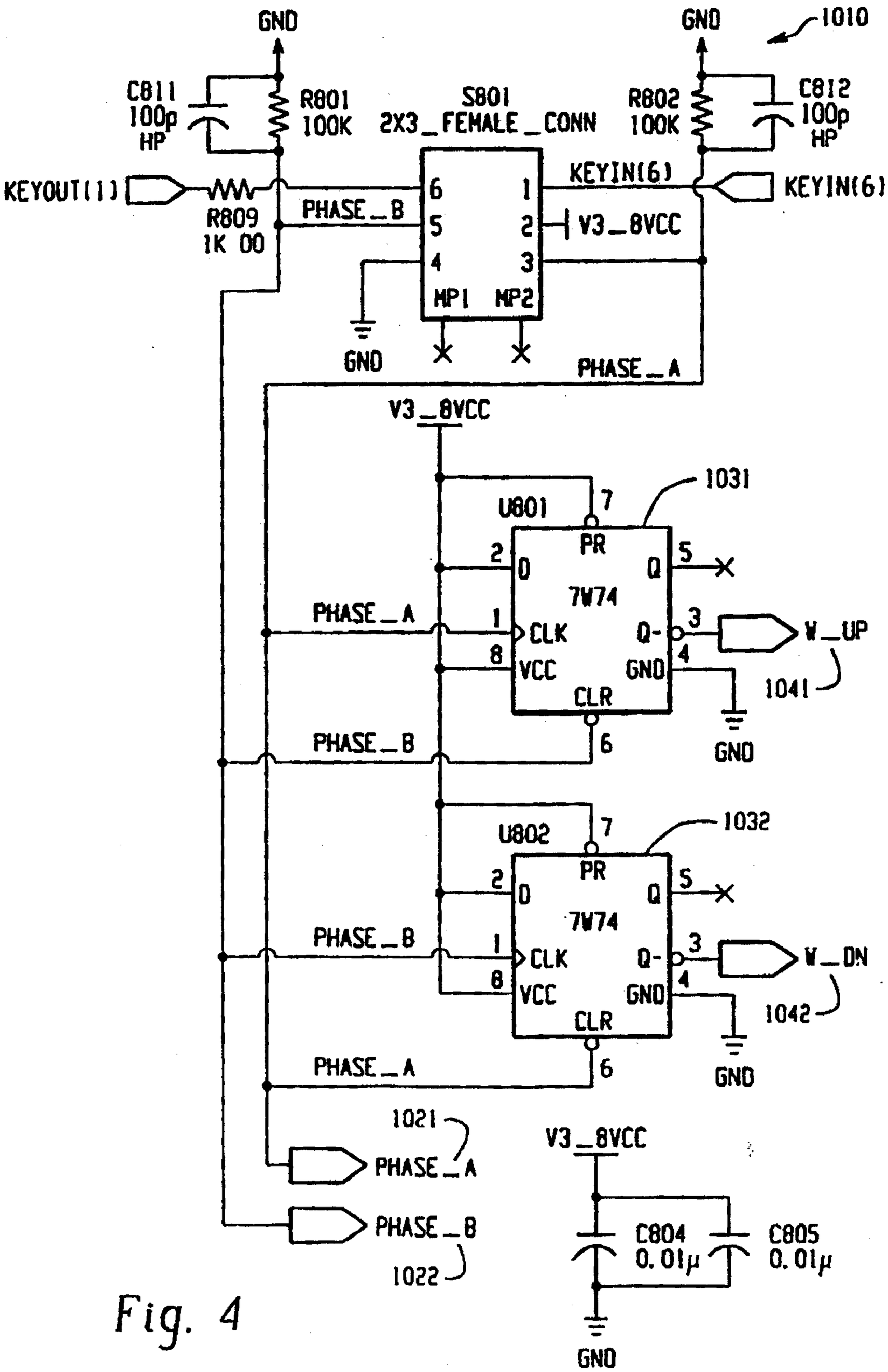

GND
C811
100p
HP
R801
100K
S801
2X3_FEMALE_CONN
GND
1010
R802
100K
C812
100p
HP
KEYOUT(1)
R809
1K 00
PHASE_B
6
5
4
MP1
MP2
1
2
3
KEYIN(6)
KEYIN(6)
V3_8VCC
GND
PHASE_A
V3_8VCC
1031
U801
7
PR
2
D
Q
5
7W74
PHASE_A
1
CLK
Q-
3
W_UP
8
VCC
GND
4
CLR
1041
GND
PHASE_B
6
U802
7
1032
PR
2
D
Q
5
7W74
PHASE_B
1
CLK
Q-
3
W_DN
8
VCC
GND
4
CLR
1042
GND
PHASE_A
6
1021
PHASE_A
PHASE_B
1022
V3_8VCC
C804
0.01μ
C805
0.01μ
GND

Fig. 4

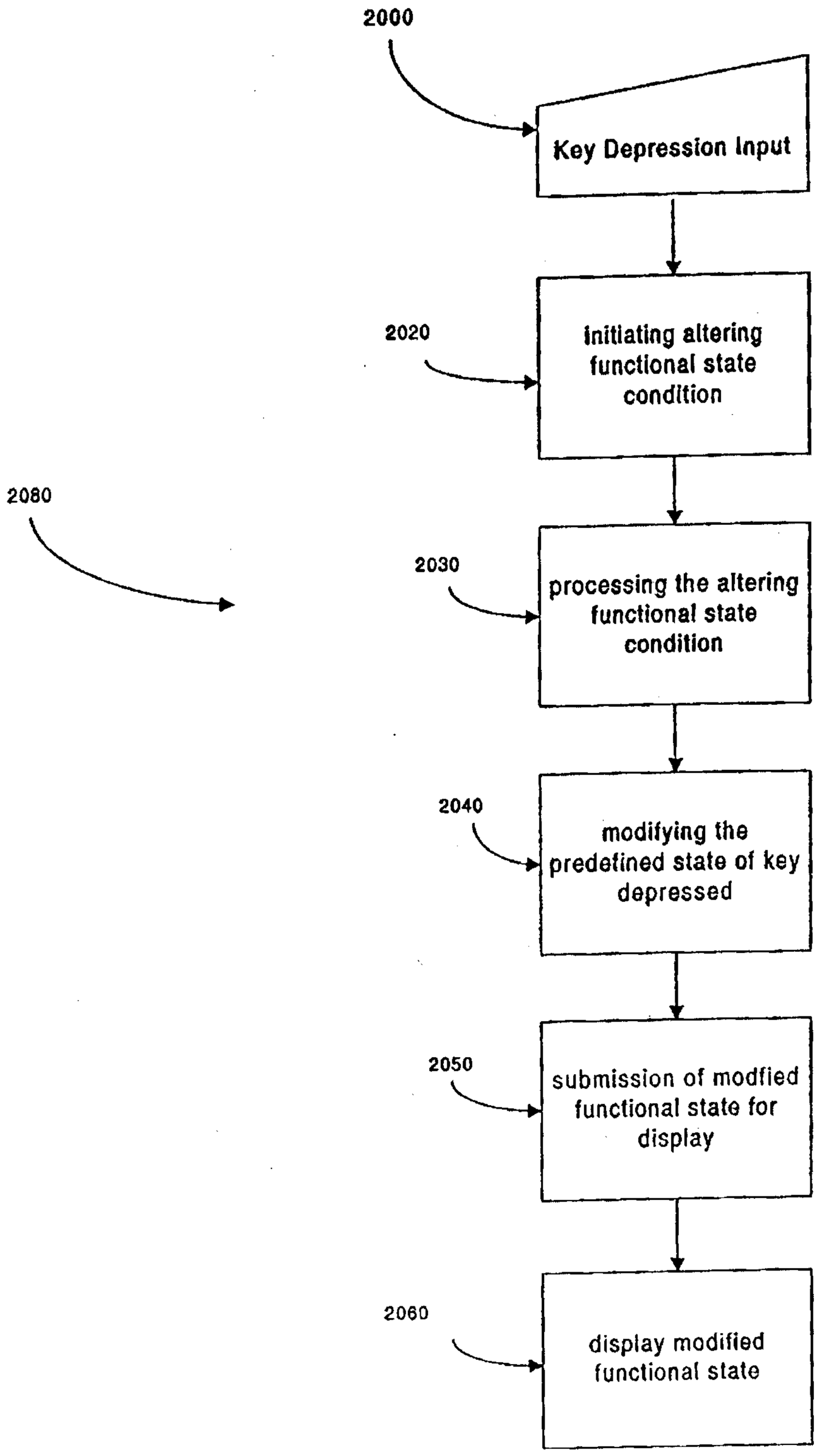

2000
Key Depression Input
2020
initiating altering functional state condition
2080
2030
processing the altering functional state condition
2040
modifying the predefined state of key depressed
2050
submission of modfied functional state for display
2060
display modified functional state

Figure 5

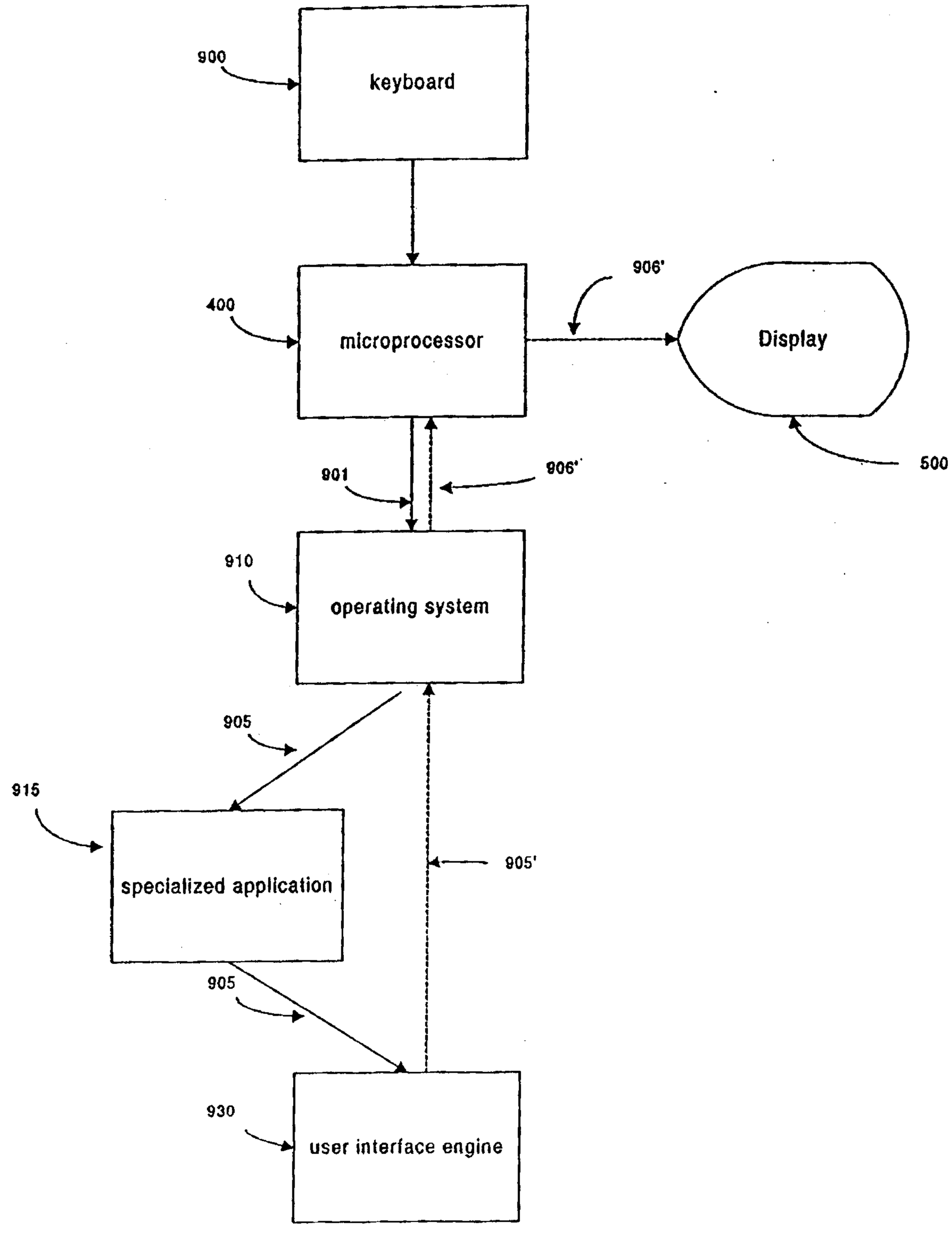

900
keyboard
400
microprocessor
906'
Display
500
901
906'
910
operating system
905
915
specialized application
905'
905
930
user interface engine

Figure 6

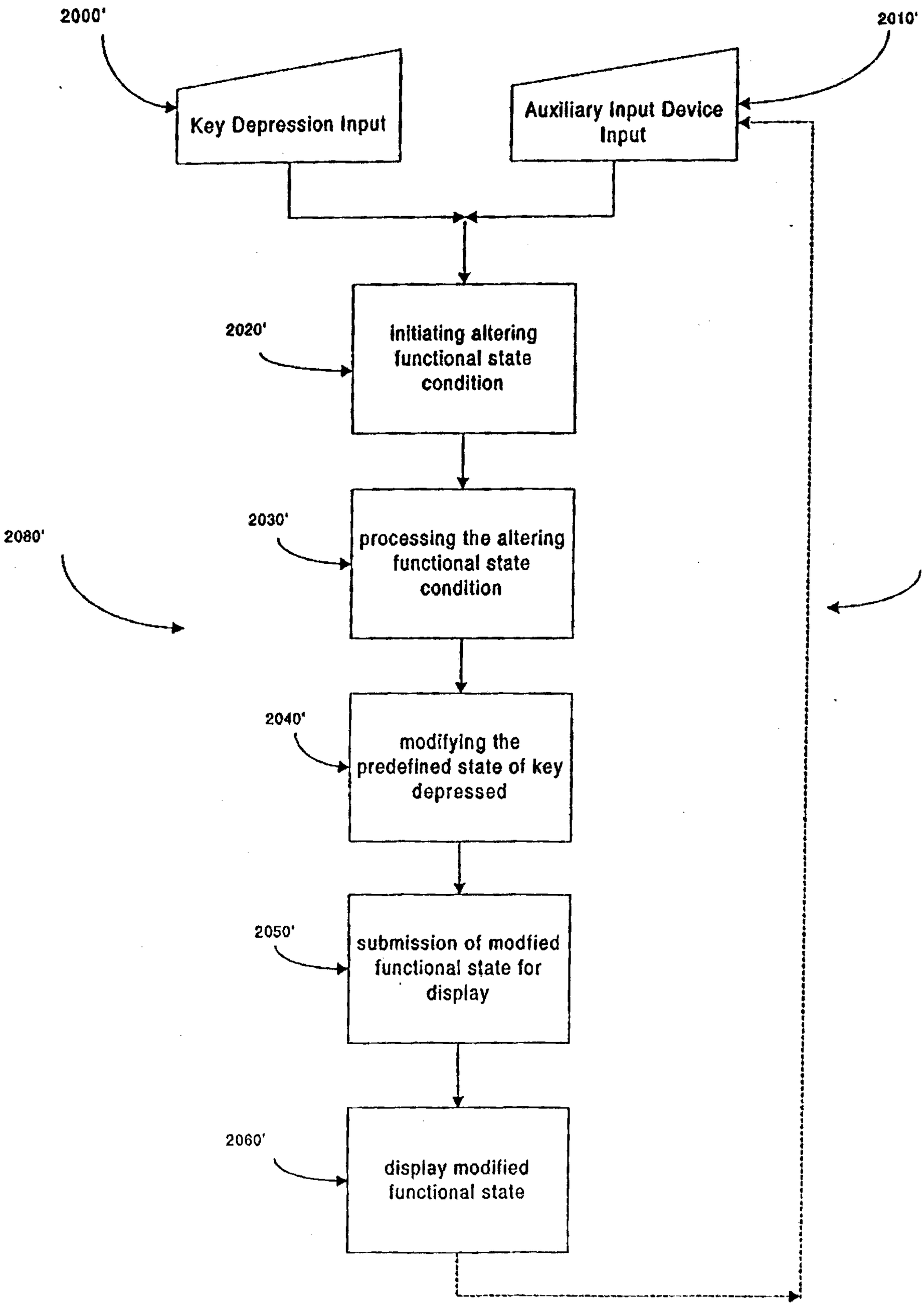

2000'
Key Depression Input
2010'
Auxiliary Input Device Input
2020'
initiating altering functional state condition
2030'
processing the altering functional state condition
2080'
2040'
modifying the predefined state of key depressed
2050'
submission of modfied functional state for display
2060'
display modified functional state
2060'

Figure 7

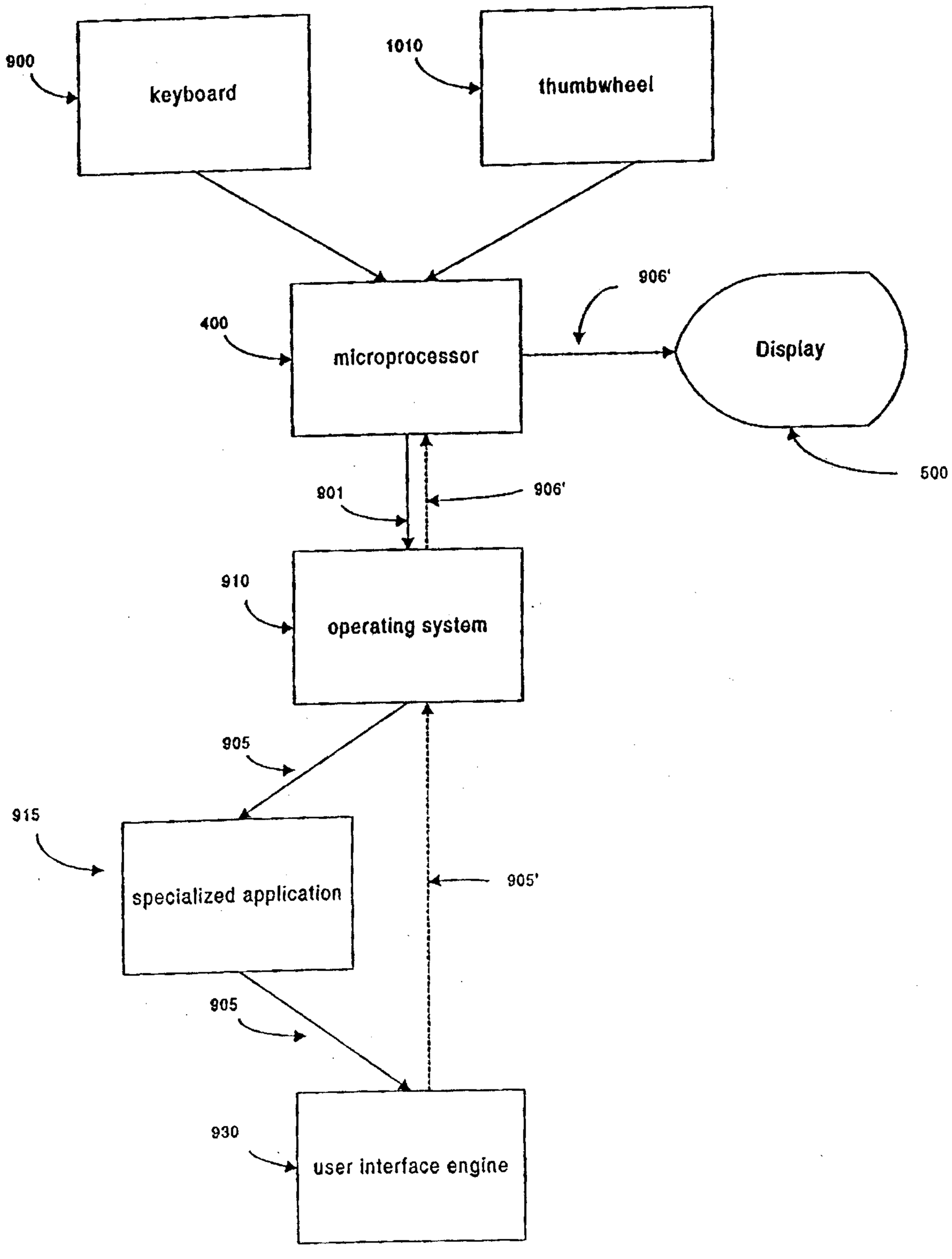
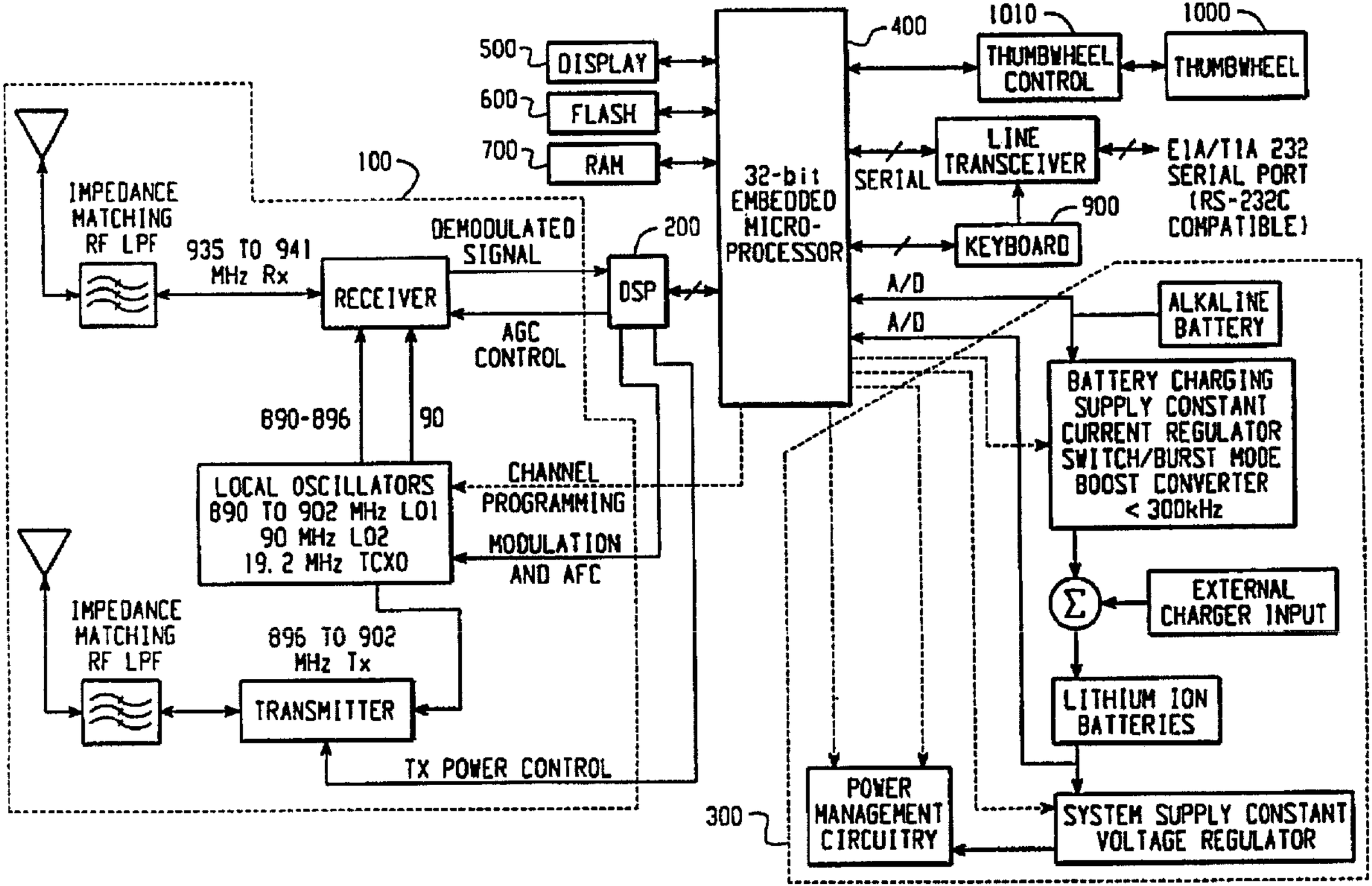

900
keyboard
1010
thumbwheel
400
microprocessor
906'
Display
500
901
906'
910
operating system
905
915
specialized application
905'
905
930
user interface engine

Figure 8


500
DISPLAY
600
FLASH
700
RAM
100
400
1010
THUMBWHEEL CONTROL
1000
THUMBWHEEL
32-bit EMBEDDED MICRO-PROCESSOR
SERIAL
LINE TRANSCEIVER
EIA/TIA 232 SERIAL PORT (RS-232C COMPATIBLE)
900
KEYBOARD
IMPEDANCE MATCHING RF LPF
935 TO 941 MHz Rx
RECEIVER
DEMODULATED SIGNAL
200
DSP
AGC CONTROL
A/D
A/D
ALKALINE BATTERY
890-896
90
BATTERY CHARGING SUPPLY CONSTANT CURRENT REGULATOR SWITCH/BURST MODE BOOST CONVERTER < 300kHz
LOCAL OSCILLATORS
890 TO 902 MHz LO1
90 MHz LO2
19.2 MHz TCXO
CHANNEL PROGRAMMING
MODULATION AND AFC
Σ
EXTERNAL CHARGER INPUT
IMPEDANCE MATCHING RF LPF
896 TO 902 MHz Tx
TRANSMITTER
LITHIUM ION BATTERIES
TX POWER CONTROL
300
POWER MANAGEMENT CIRCUITRY
SYSTEM SUPPLY CONSTANT VOLTAGE REGULATOR