

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
30 October 2003 (30.10.2003)

PCT

(10) International Publication Number  
**WO 03/090068 A2**

(51) International Patent Classification<sup>7</sup>: **G06F 9/40**

(21) International Application Number: PCT/US03/11770

(22) International Filing Date: 17 April 2003 (17.04.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:  
60/374,400 22 April 2002 (22.04.2002) US

(71) Applicant (for all designated States except US): **THOMSON LICENSING S.A.** [FR/FR]; 46, Quai A. Le Gallo, F-92648 Boulogne (FR).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **LEFEVRE, Chad, Andrew** [US/US]; 8707 Arbor Lake Drive, #1526, Indianapolis, IN 46268 (US). **COOPER, JR., Steven, Lee**

[US/US]; 4527 South 700 East, Whitestown, IN 46075 (US). **TENBARGE, James, Duane** [US/US]; 11372 Spyglass Ridge Drive, Fishers, IN 46038 (US). **LYNCH, David, Johnston** [US/US]; 12562 Glendurgan Drive, Carmel, IN 46032 (US).

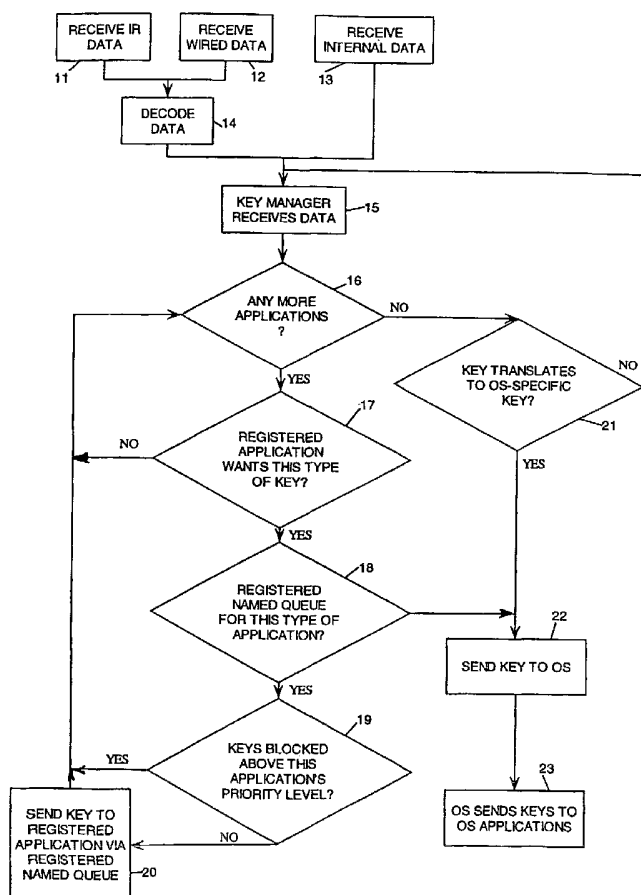
(74) Agents: **TRIPOLI, Joseph, S.** et al.; c/o THOMSON Licensing Inc., Two Independence Way, Suite #2, Princeton, NJ 08540 (US).

(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW),

[Continued on next page]

(54) Title: REMOTE KEY MANAGER



(57) Abstract: An apparatus having a memory and an operating system running a plurality of applications that are controlled by user signals, means to register the user signals required by each application, means to receive user signals, and means to send each received user signal to the application registered to require that signal. A method of passing user signals to a plurality of applications running with an operating system comprising receiving from each application information on user signals required by that application; saving in a memory the required user signals associated with each application; receiving a user signal from a user interface device; retrieving from the memory applications that require the received user signal; and sending the received user signal to each of the retrieved applications.

WO 03/090068 A2



Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),  
European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE,  
ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO,  
SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM,  
GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

*For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

**Published:**

— *without international search report and to be republished  
upon receipt of that report*

## REMOTE KEY MANAGER

### Cross-Reference to Related Applications

5           Benefit of Provisional Patent Application 60/374,400, filed April 22, 2002, is claimed.

### Technical Field of Invention

          This invention relates to the field computer operating systems, and particularly to managing user commands in devices running a plurality of applications.

### Background Art

10           Devices have become available which run a plurality of applications and receive a plurality of user control signals. For example, devices that include television, Internet, satellite television, personal video recorder (PVR), and digital versatile disk (DVD) applications, among others, controlled by a single operating  
15           system such as Windows CE, are known.

          The operating systems in such devices receive user signals from a plurality of sources, for example infrared (IR) signals from a remote control device, electronic signals from a front panel assembly (FPA), and other parts of the system. Data is received and processed from the respective IR drivers, FPA drivers, and other parts  
20           of the system. Presently available systems assume the keys are intended for the foreground window. However, sometimes the user intends to control a background application, which is not possible with present systems.

### Disclosure of Invention

25           An object of the present invention is to organize the receipt of various user keys and direction thereof to various applications. This object, and others which will become apparent from the following disclosure and drawings, is addressed by the present invention which comprises in one aspect an apparatus having a memory and an operating system running a plurality of applications which are controlled by user  
30           signals, means to register the user signals required by each application, means to receive user signals, and means to send each received user signal to the application registered to require that signal.

Another aspect of the invention is a method of passing user signals to a plurality of applications running with an operating system, the method comprising receiving from each application information on user signals required by that application; saving in a memory the required user signals associated with each application; receiving a user signal from a user interface device; retrieving from the memory applications that require the received user signal; and sending the received user signal to each of the retrieved applications.

The received user signal can be sent directly to the application or it can be translated to a second signal recognizable by the operating system, which is in turn sent to the operating system.

An example of an operating system, which is currently typical, is Windows CE, but other operating systems can, of course, be used in the invention. Typically, the applications comprise a user interface and a browser. The user signals may be received, for example, by IR from a remote control device, from front panel buttons, and/or from the system. Certain applications can "fake" a key press, for example.

Error checking and conversion of user signals can be conducted by known methods and software.

A key manager can be provided, and the key manager can be registered with an infrared driver and a front panel assembly driver.

Active application programs can be registered with the key manager and be unregistered when they become inactive. When the application becomes registered, a set of keys corresponding to each active program can be registered so that when keys are received, the key manager passes the received keys to an active application program for which the keys are registered.

A message queue corresponding to each application program can be registered, including the name and size of the queue, upon which the key manager can create one end of the message queue for passing keys to the application program. In some embodiments, all the keys are translated to match an operating system equivalent, sent to the operating system upon receipt, and thereafter passed to the top level application, and if the top level application does not want the key, the key is then passed sequentially to the next application which wants the key. In some embodiments, the system provides an option to block applications below an assigned

priority level to receive keys. In some embodiments, the keys are translated to match an operating system equivalent, sent to the operating system upon receipt, and thereafter passed to each application that responds to the key. Upon registration of the application, the type of keys required by the application can be registered by the  
5 key manager.

The apparatus to which this invention is applicable comprise a memory and an operating system adapted to run a plurality of applications which are controlled by user signals, means to register the user signals required by each application, means to receive user signals, and means to send each received user signal to the  
10 application registered to require that signal. Examples of such apparatus are televisions with Internet browser capability. The apparatus will usually have a means to display a user interface, means to receive remote control signals, means to receive front panel assembly signals, and means to receive keyboard signals.

#### Brief Description of Drawings

15 FIG. 1 is a high level flow chart of a method according to the invention.

#### Modes for Carrying Out the Invention

Referring first to Fig. 1, a process is illustrated wherein raw keys from an IR driver, keyboard, and FPA are passed to different applications and systems.

The process and apparatus of the invention will be referred to herein as the  
20 Key Manager. In the illustrated embodiment, the Key Manager takes the raw keys from the IR driver in the form of IR data 11 and Front Panel Assembly (FPA) keys in the form of wired data 12 and decodes 14 the raw key data. The Key Manager 15 also receives internal data 13 from other parts of the system, for example other applications that may "fake" keys. The Key Manager 15 manages passing the  
25 decoded key data to different applications and systems, so that each part receives what it expects, as illustrated in this example.

After receiving a key press from, for example the remote control via IR 11, the Key Manager 15 determines 16 if there is a registered application and if so, then whether 17 the registered application wants the type of key received by the Key  
30 Manager. If the first registered application does not 17 want the type of key sent by the Key Manager, the Key Manager checks 16 whether there are any more registered applications until it finds one which wants the type of key being sent 17, and then checks 18 whether there is a registered queue for that application.

If there are no more registered applications, the Key Manager checks 21 whether the key translates to an operating system specific key and if so, sends 22 the key to the operating system, otherwise the Key Manager waits for another key. The Key Manager then determines 18 if there is a registered named queue for the application which wants the type of key being sent and if so, the Key Manager checks 5 19 whether keys are blocked for that application. Each application is assigned a priority level. If the keys are not blocked 19 above that application's priority level, the Key Manager sends 20 the key to the registered application via the registered named queue, otherwise, if the keys are blocked above that application's priority level, the 10 Key Manager checks 16 for the next registered application. If the Key Manager checks 18 for whether there is a registered named queue for the first registered application that wants the type of key which was received by the Key Manager and if there is no registered named queue for the first registered application which wants that type of key, the key is sent 22 to the operating system which then sends 23 the 15 key to operating system applications.

As an example of how the Key Manager solves a problem, the user interface (UI) and the browser both need to be aware of a "Channel Up" key when the browser is showing, i.e., the browser is in the foreground. Both applications would ask for the Channel Up key in this instance. Since the key manager sends keys to 20 any known application that asks for them, and also to any known or unknown application that is using the default method to receive keys, both the UI and the browser would receive the Channel Up key press signal.

This invention takes key data from the low-level system and manages its passing to higher parts of the system. Some of the data is based upon infrared (IR) 25 protocols, some of it is retrieved from buttons of the FPA, and some of it is sent back through the system. After basic error checking and conversion on the data is done, the data is passed to the Key Manager to be handled.

The Key Manager registers itself with the IR\FPA drivers in order to receive keys from them. Applications running at higher levels are registering and 30 unregistering with the Key Manager. The Key Manager uses the registration data to govern which applications should get which keys. When the Key Manager receives data from the IR driver, the FPA driver, and other parts of the system, it

preprocesses it and then passes the data to other parts of the system based on how they registered with the Key Manager.

There are two ways that the Key Manager passes keys. The first method is via message queues. When an application registers with the Key Manager, it  
5 supplies the name and size for its message queue. The Key Manager then creates one end of that queue, which it will use to pass keys to that application. The second method is via Microsoft Windows key-passing mechanism. All keys are translated to match the Windows Virtual Key equivalent, and then doled out to the Operating System, which then passes the keys to the top-level application (if that application  
10 does not want the key, then Windows passes it down to the next application, and so on).

The Key Manager has the ability to pass keys to both "known" applications (that may or may not use the Microsoft Windows CE method of passing keys) and "unknown" applications (which use the Microsoft Windows CE method of passing  
15 keys). Since Microsoft Windows CE will only pass keys to the foreground application, the present improved system allows multiple applications to see what key had been pressed. In certain circumstances, more than one application may need to respond to a key, i.e., the browser and the UI. With the conventional Microsoft Windows CE method of passing keys, that ability is not present. The Key  
20 Manager has ability to register and unregister known applications, which is an improvement.

When an application registers with the Key Manager, it also tells the Key Manager what type of keys it wants. Its choices are remote control keys, keyboard keys, and FPA keys. The Key Manager uses this information to decide whether the  
25 application wants to receive keys from certain devices.

While the invention and the best mode known to the inventors for carrying it out has been described herein in sufficient detail for others skilled in this art to understand it and to make and use it, various alternatives, modifications, and improvements should become readily apparent without departing from the spirit and  
30 scope thereof.

**CLAIMS**

1. A method for passing user signals to a plurality of applications running with an operating system, the method comprising the steps of:
  - 5 receiving from each application information on user signals required by that application;
  - saving in a memory the required user signals associated with each application;
  - receiving a user signal from a user interface device;
  - retrieving from the memory applications that require the received user signal;
  - 10 and
  - sending the received user signal to each of the retrieved applications.
2. The method of claim 1, further comprising the step of sending the received user signal to the operating system.
- 15 3. The method of claim 1, further comprising the step of translating the received user signal into a second signal recognizable by the operating system and sending the second signal to the operating system.
- 20 4. The method of claim 1, wherein the operating system is Windows CE.
5. The method of claim 1 wherein the applications comprise a user interface and a browser.
- 25 6. The method of claim 1 wherein the user signals are received by IR from a remote control device, from front panel buttons, and/or from the system.
7. The method of claim 1, further including error checking and conversion of user signals.

8. The method of claim 1 comprising registering a key manager with an infrared driver and a front panel assembly driver, registering active application programs and unregistering inactive programs with the key manager,  
5 registering a set of keys corresponding to each active program and unregistering the set of keys upon a program becoming inactive, and passing received keys to an active application program for which the received keys are registered.
- 10 9. The method of claim 8 wherein upon registration of an application program, a name and size for a message queue corresponding to the program is stored by the key manager, upon which the key manager creates one end of the message queue for passing keys to the application program.
- 15 10. The method of claim 8 wherein all keys are translated to match an operating system equivalent, sent to the operating system, and thereafter passed to a top level application, and if the top level application does not want the key, the key is then passed sequentially to a next highest level of application which accepts the key.
- 20 11. The method of claim 8 wherein the keys are translated to match an operating system equivalent, sent to the operating system, and thereafter passed to each application that responds to the key.
- 25 12. The method of claim 8 wherein upon registration of an application program with the key manager, the type of keys required by the application program is registered.
- 30 13. The method of claim 12 wherein the type of keys is selected from the group consisting of remote control keys, keyboard keys, and front panel assembly keys.

14. An apparatus having a memory and an operating system adapted to run a plurality of applications that are controlled by user signals, means to register the user signals required by each application, means to receive user signals, and means to send each received user signal to the application registered to require that signal.

5

15. The apparatus of claim 14 comprising a television, a browser, a display, and a keyboard.

16. The apparatus of claim 14 comprising means to display a user interface, means to receive infrared remote control signals, means to receive front panel assembly signals, and means to receive keyboard signals.

10

15

1 / 1

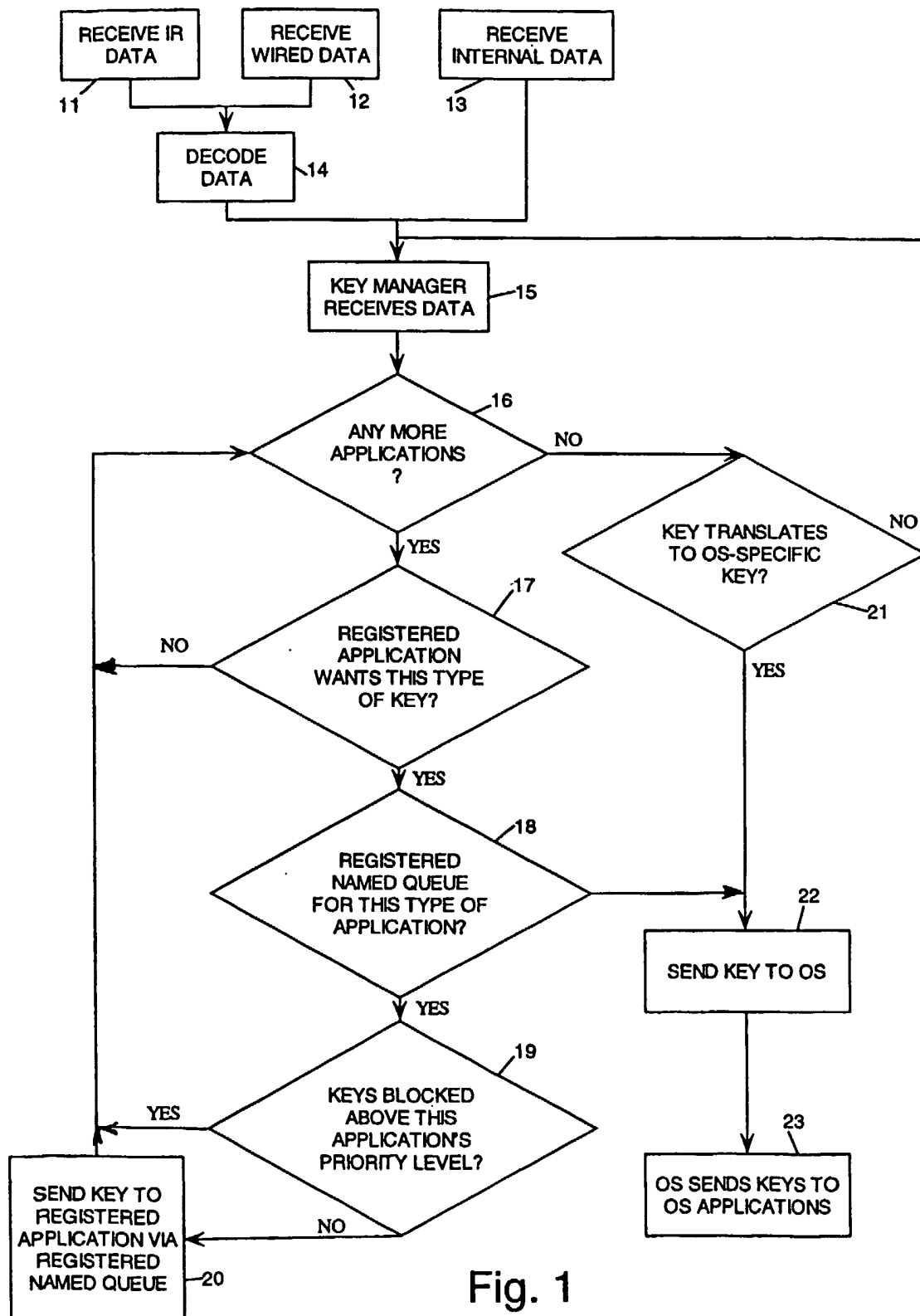


Fig. 1