US 20120272230A1

(19) **United States**

(12) **Patent Application Publication**

(10) **Pub. No.: US 2012/0272230 A1**

(43) **Pub. Date: Oct. 25, 2012**

**LEE**

(54) **POWER SAVING APPLICATION UPDATE IN A PORTABLE ELECTRONIC DEVICE**

(75) Inventor: **JAE HAN LEE**, SAN JOSE, CA (US)

(73) Assignee: **APPLE INC.**, CUPERTINO, CA (US)

(21) Appl. No.: **13/090,007**

(22) Filed: **Apr. 19, 2011**

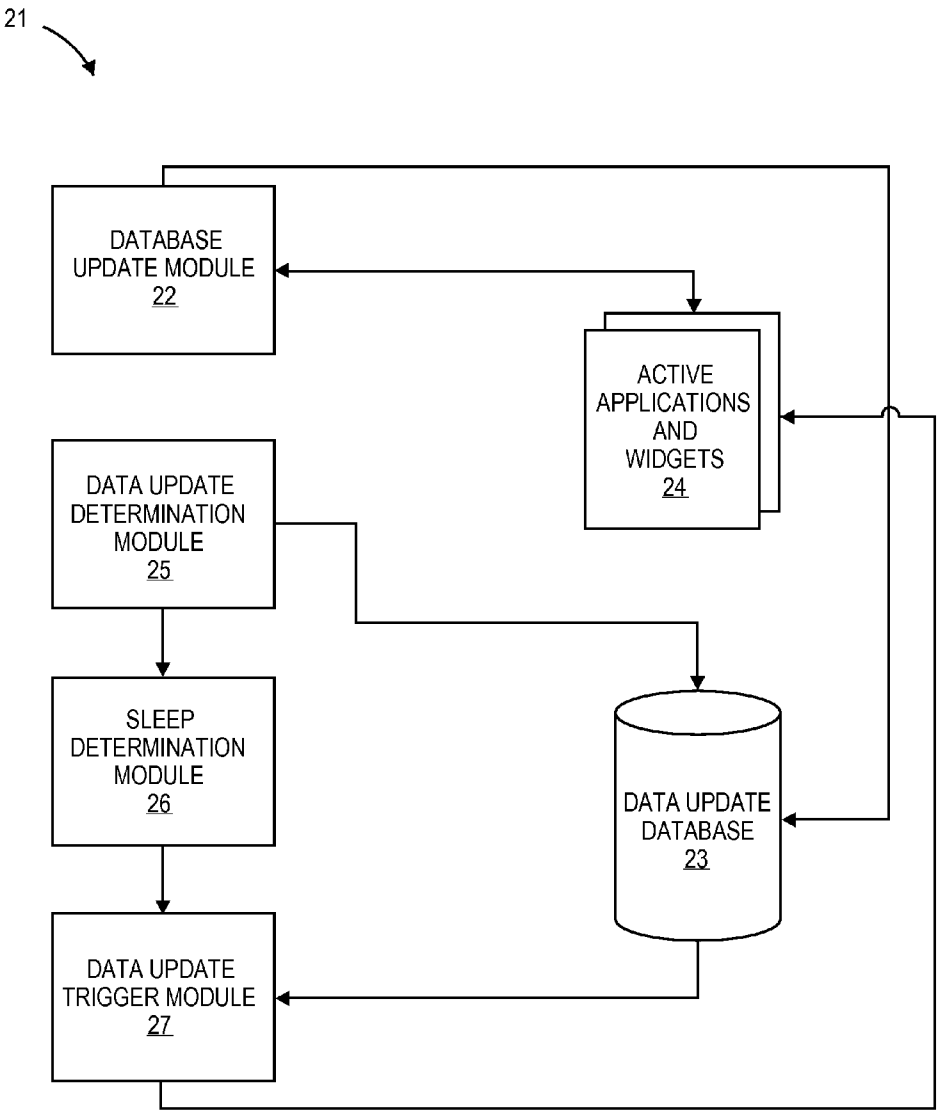
**Publication Classification**

(51) **Int. Cl.**
*G06F 9/44* (2006.01)

(52) **U.S. Cl.** ..... **717/173**; 717/171

(57) **ABSTRACT**

A power savings mode for data updates is provided. The power savings mode prevents data updates to occur while a screen of a portable electronic device is turned off and the device is in a sleep state. The power savings mode waits until the screen is turned on and the portable electronic device is in a wake state before allowing applications and widgets to update data from network repositories. By preventing applications and widgets from updating data while the portable electronic device is in a sleep state, the power savings mode conserves battery life and network bandwidth by limiting possibly unnecessary data transmissions.




**MASIMO 1070**
**Masimo v. Apple**
**IPR2023-00664**

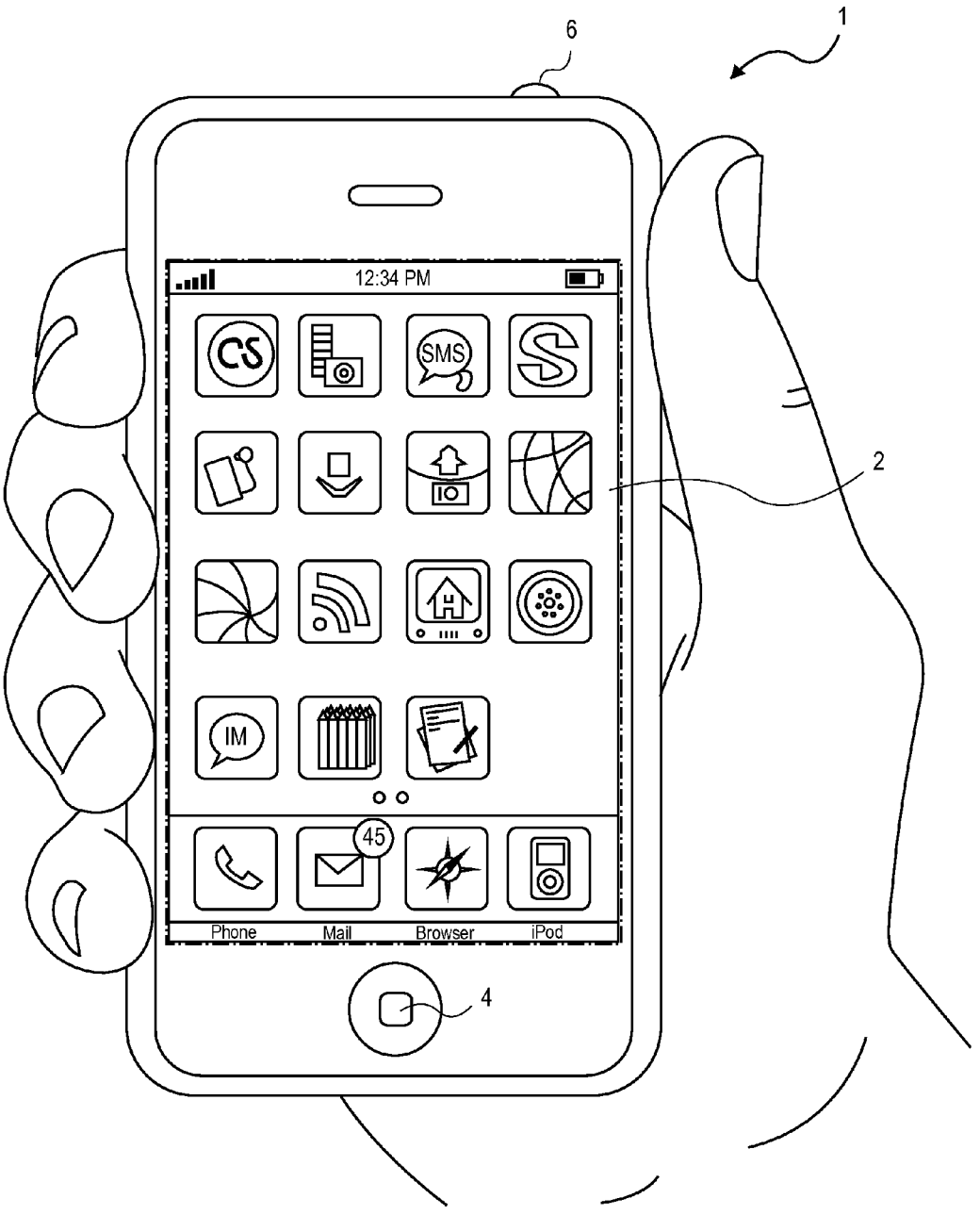



FIG. 1A

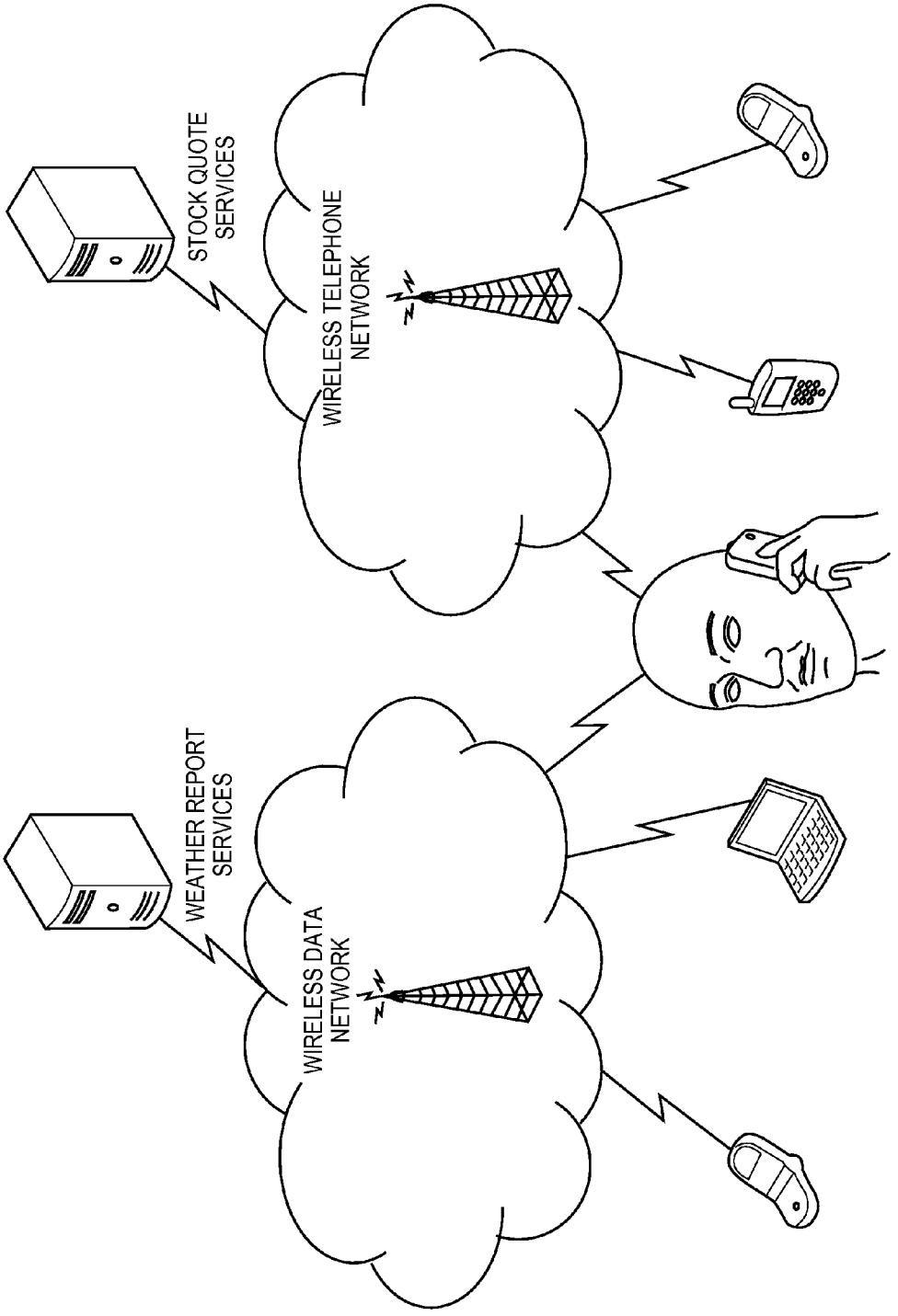

STOCK QUOTE SERVICES
WIRELESS TELEPHONE NETWORK
WEATHER REPORT SERVICES
WIRELESS DATA NETWORK

FIG. 1B

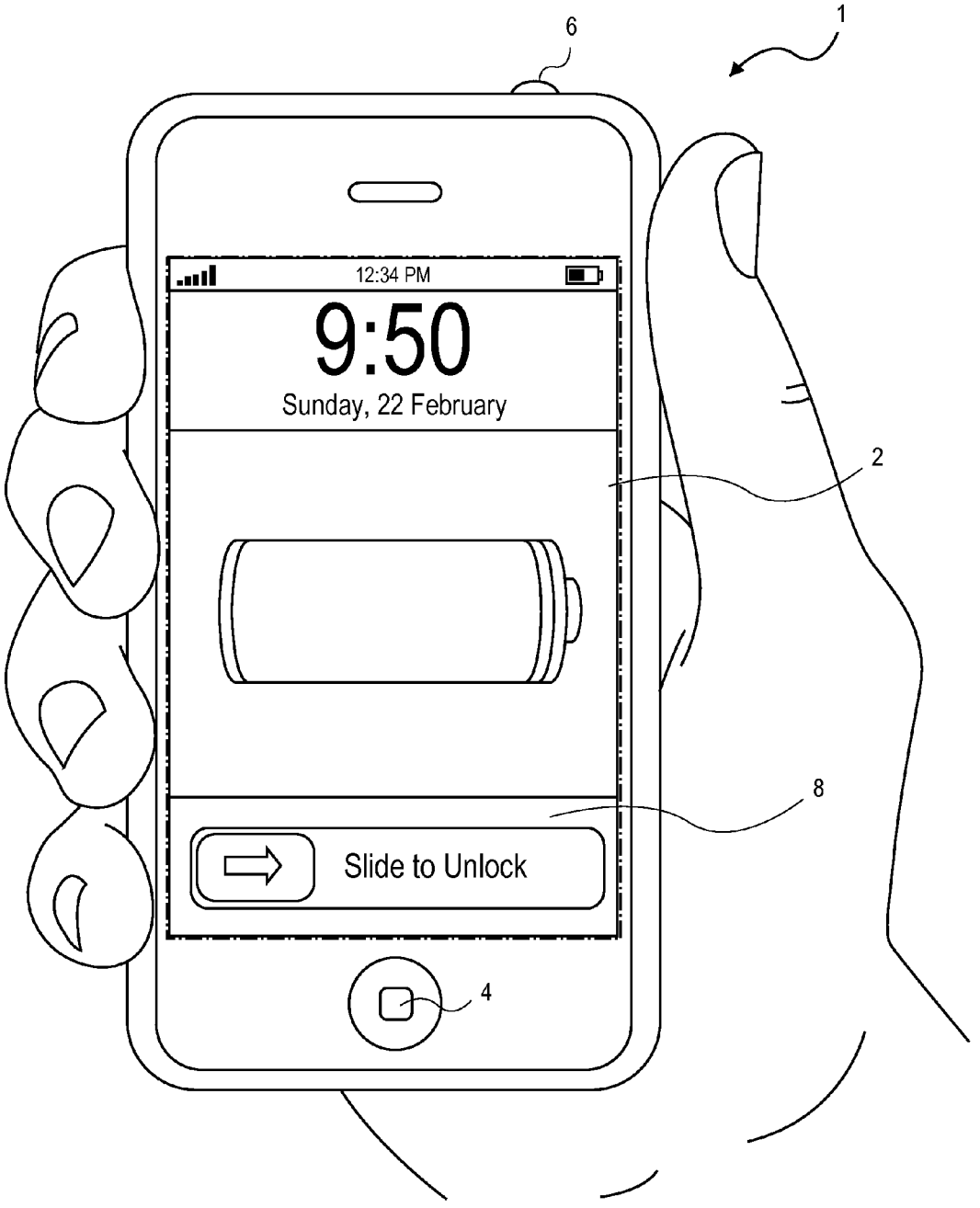



FIG. 2

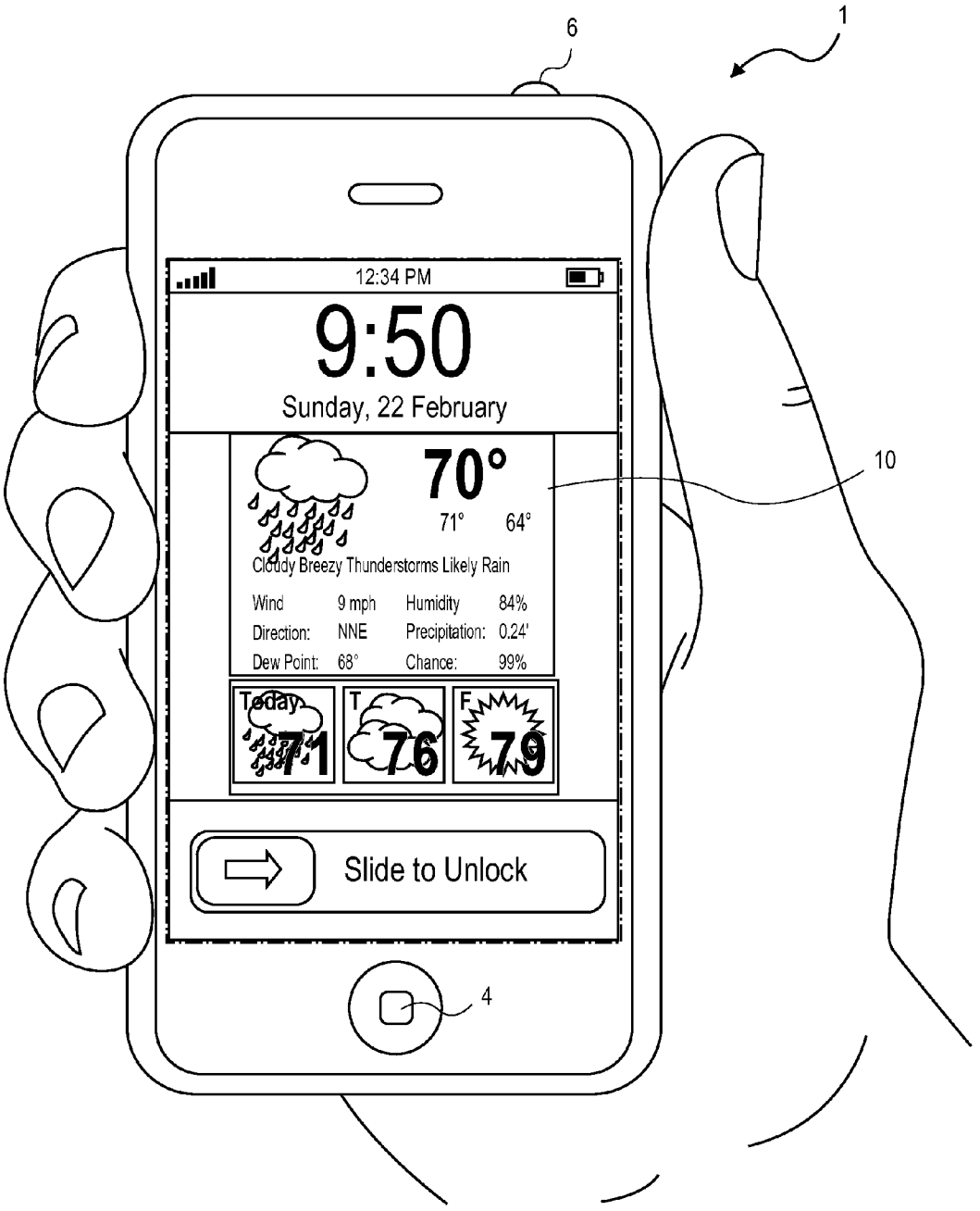

6
1
12:34 PM
9:50
Sunday, 22 February
70°
71° 64°
10
Cloudy Breezy Thunderstorms Likely Rain
Wind 9 mph Humidity 84%
Direction: NNE Precipitation: 0.24'
Dew Point: 68° Chance: 99%
Today
T
F
71
76
79
Slide to Unlock
4

FIG. 3A

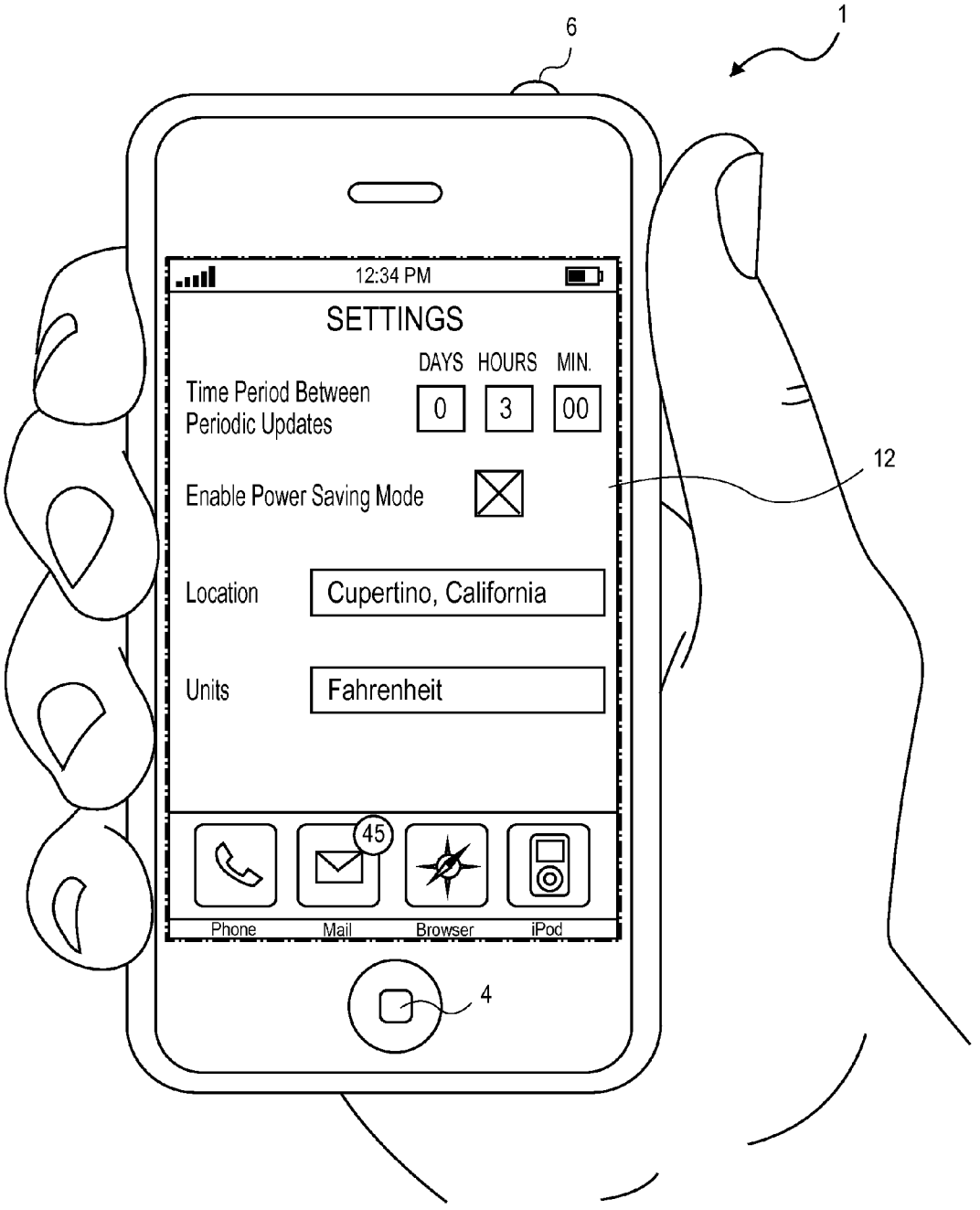

6
1
12:34 PM
SETTINGS
DAYS HOURS MIN.
Time Period Between Periodic Updates
0
3
00
Enable Power Saving Mode
12
Location
Cupertino, California
Units
Fahrenheit
45
Phone
Mail
Browser
iPod
4

FIG. 3B

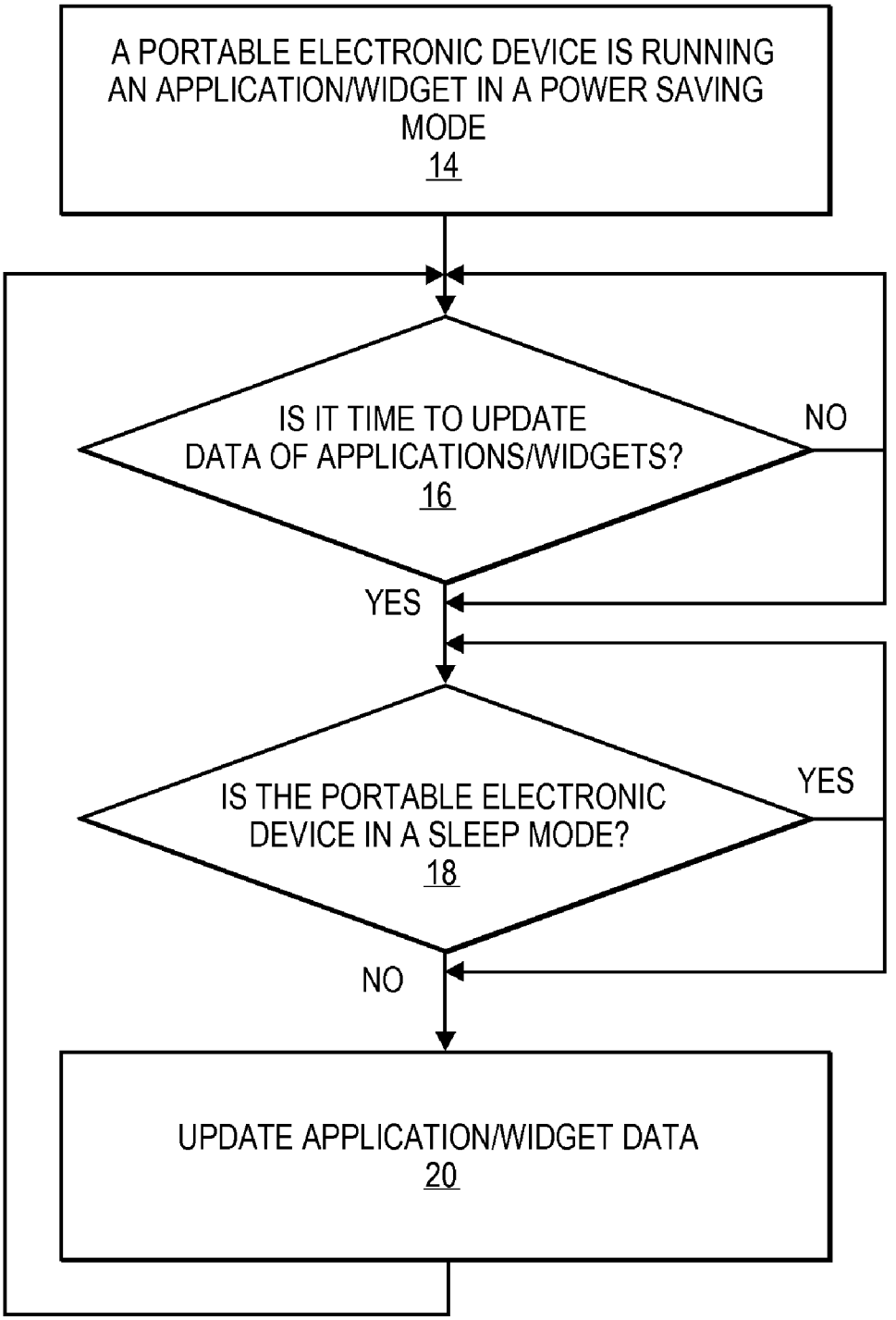



FIG. 4

MASIMO 1070
Masimo v. Apple
IPR2023-00664

6
1
12:34 PM
12:00 PM
Sunday, 22 February
70°
71° 64°
10
Cloudy Breezy Thunderstorms Likely Rain
Wind 9 mph
Direction: NNE
Dew Point: 68°
Humidity 84%
Precipitation: 0.24'
Chance: 99%
Today
T
F
71
76
79
Slide to Unlock
4

FIG. 5A

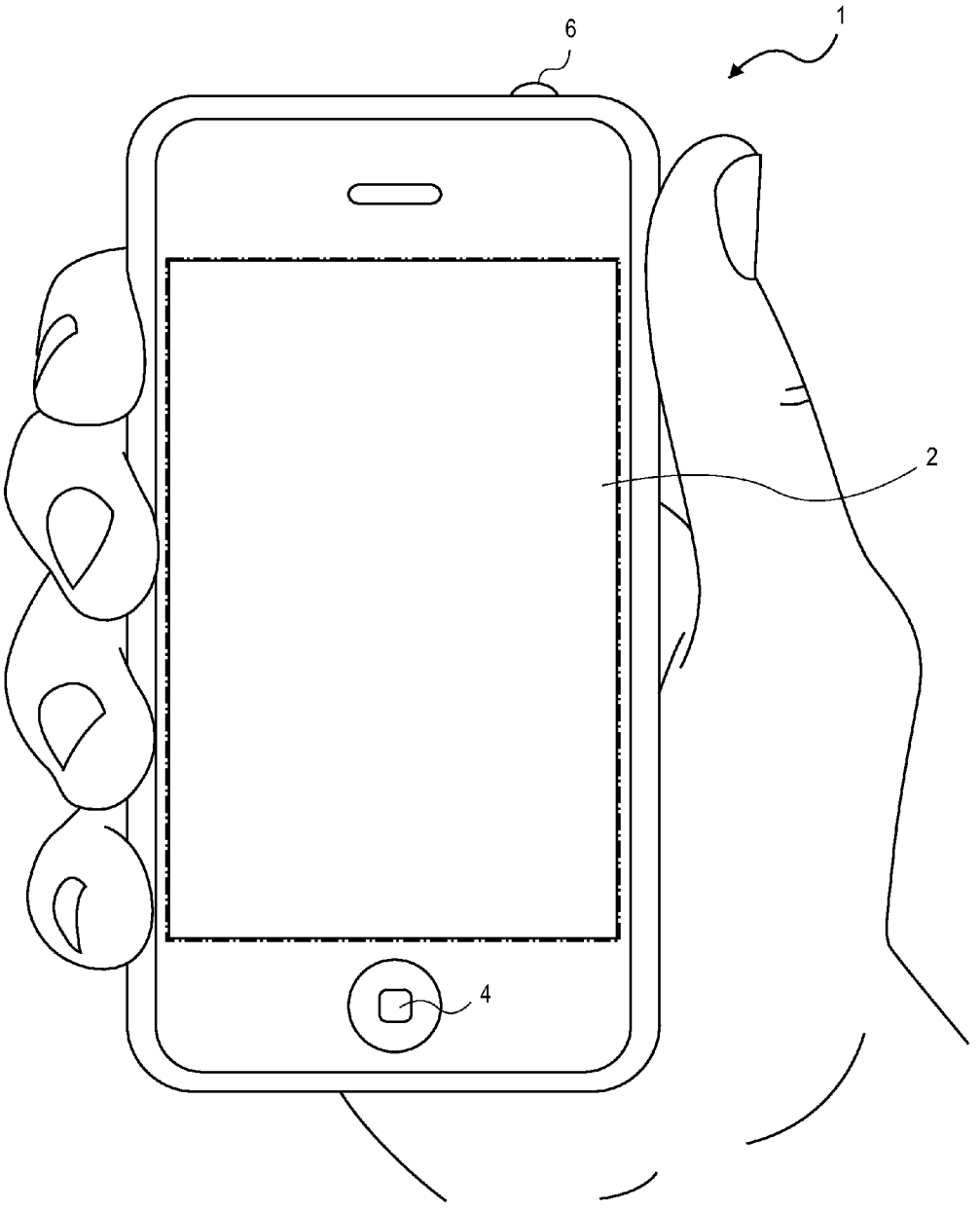



FIG. 5B

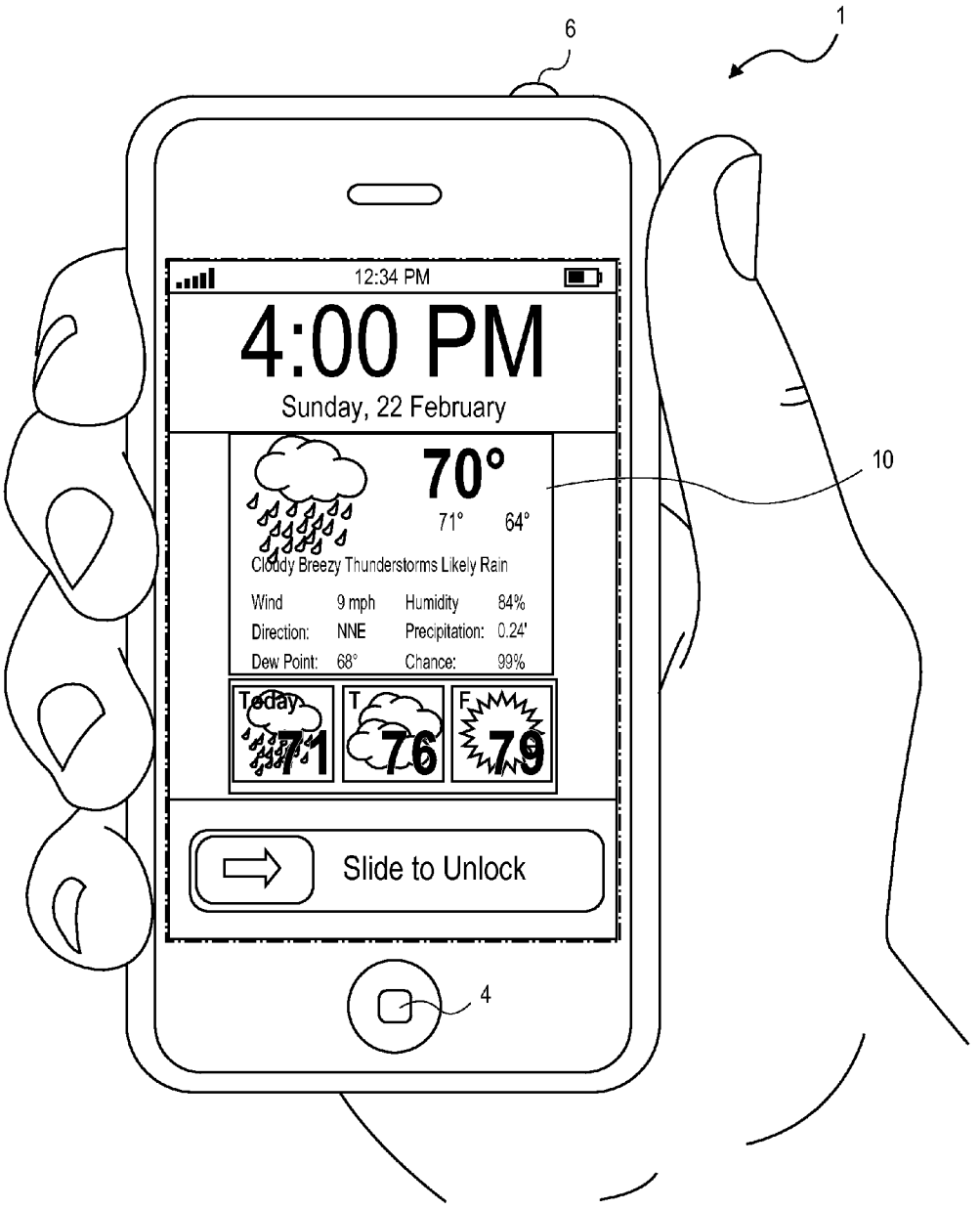

6
1
12:34 PM
4:00 PM
Sunday, 22 February
70°
71° 64°
10
Cloudy Breezy Thunderstorms Likely Rain
Wind 9 mph
Direction: NNE
Dew Point: 68°
Humidity 84%
Precipitation: 0.24'
Chance: 99%
Today
T
F
71
76
79
Slide to Unlock
4

FIG. 5C

6
1
12:34 PM
4:00 PM
Sunday, 22 February
68°
71° 64°
2
Cloudy Breezy Thunderstorms Likely Rain
Wind 11 mph Humidity 84%
Direction: NNE Precipitation: 0.24'
Dew Point: 68° Chance: 99%
Today
T
F
71
76
79
Slide to Unlock
4

FIG. 5D

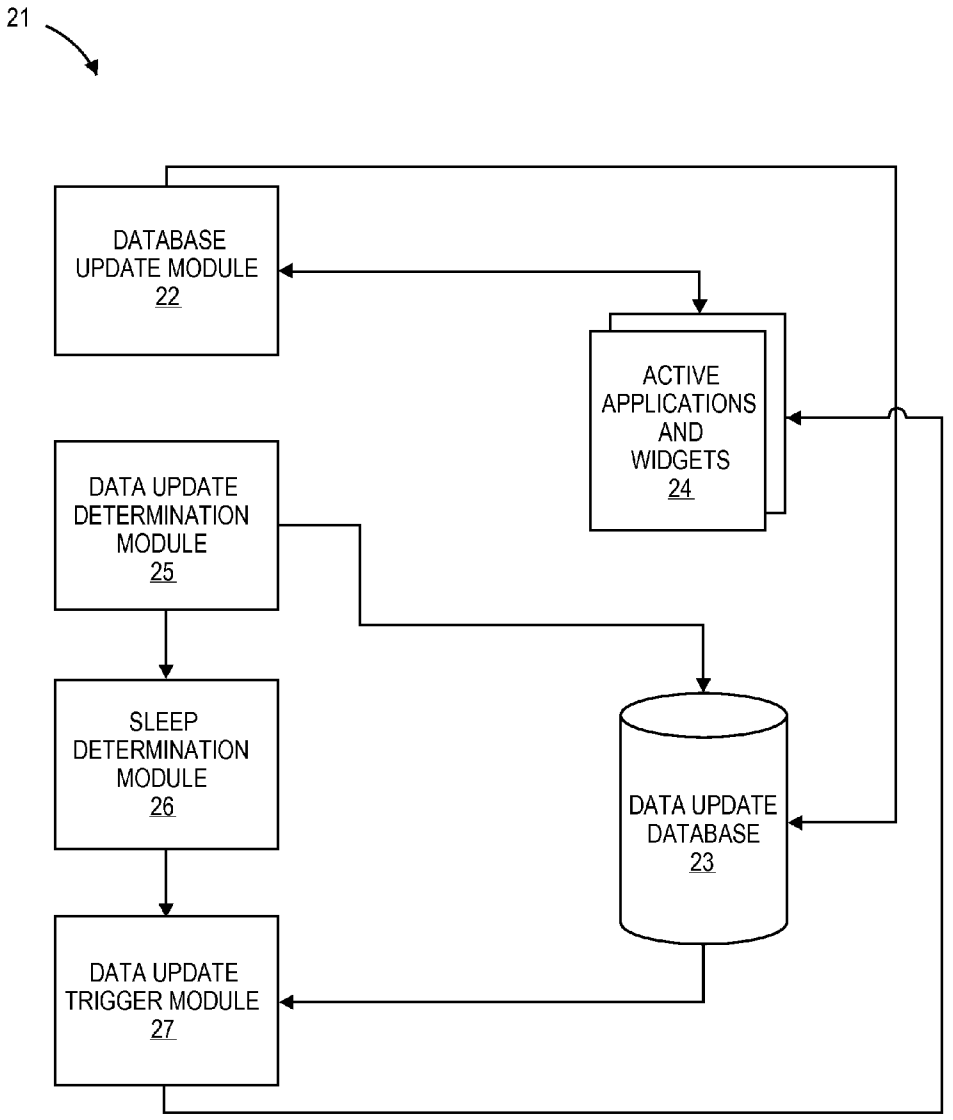



FIG. 6

MASIMO 1070
Masimo v. Apple
IPR2023-00664

# POWER SAVING APPLICATION UPDATE IN A PORTABLE ELECTRONIC DEVICE

## BACKGROUND

**[0001]** An embodiment of the invention generally relates to a power saving system and method for updating data in an application or widget running on a portable electronic device. Other embodiments are also described.

**[0002]** Portable electronic devices receive and display numerous pieces of information to users. The information may be received through the Internet or any type of data network. Conventionally, users have retrieved and viewed information from data networks through websites formatted for web browsers. However, websites are often difficult to view on smaller screens typically used on portable electronic devices. Accordingly, special purpose applications have been developed for portable electronic devices that properly format information typically viewed on a website for use on a portable electronic device's small screen. In some cases, these special purpose applications may fully encompass the viewing area of the screen such that a user can only view the contents of one active application. However, in other cases special purpose applications may be sized to encompass only a small area of a screen such that multiple applications may be viewed simultaneously. These smaller footprint applications are sometimes referred to as "widgets."

**[0003]** Whether using a web-browser to view information or a special purpose application, these applications receive information through a network interface using either a push or pull protocol. For example, a portable electronic device with Internet connectivity may retrieve weather information for a user's hometown from a weather broadcast service, such as The Weather Channel (www.weather.com), The National Weather Service (www.weather.gov), AccuWeather (www.accuweather.com), or a similar service. Upon receipt, the information is shown to a user on a screen of the portable electronic device through a visible user interface provided by a weather application.

**[0004]** In many cases, the constant changing of information results in the need to periodically update the data that is shown by the application. For example, a weather application may be configured to automatically retrieve and update its displayed data every hour. These updates take place even while the screen of the portable electronic device is shut off such that the weather application is not viewable to a user. In some instances, data displayed in the application may be refreshed several times before the screen of the portable electronic device is turned on and viewed by a user. Each unviewed data update thus consumes battery power and network bandwidth, without providing a benefit to the user since the results were never viewed.

## SUMMARY

**[0005]** Many portable electronic devices include various applications and widgets that rely on periodically obtaining data from remote network repositories. After The updated data is typically shown to a user on a screen of the portable electronic device only when the screen has been turned on and the device has exited a sleep state. There is a need for a power saving mode that efficiently controls the update of such data.

**[0006]** An embodiment of the invention provides a power savings mode for data updates. The power savings mode prevents data updates from taking place, while a screen of the portable electronic device is turned off and the device is in a sleep state. In one embodiment, the power savings mode waits until the screen is turned on and the portable electronic device is in a wake state, before allowing a given application or widget to update data from network repositories. In another embodiment, the power savings mode prevents a given application or widget from updating its data while the device is in its sleep state, until a predetermined user command is detected or received, e.g. a physical home button being pressed, an audible pattern being recognized (using a microphone), or a movement pattern being recognized (using a position, orientation or movement sensor). By preventing the application or widget from updating data while the portable electronic device is in a sleep state, the power savings mode may conserve battery energy and network bandwidth by limiting possibly unnecessary network data transmissions.

**[0007]** The above summary does not include an exhaustive list of all aspects of the present invention. It is contemplated that the invention includes all systems and methods that can be practiced from all suitable combinations of the various aspects summarized above, as well as those disclosed in the Detailed Description below and particularly pointed out in the claims filed with the application. Such combinations have particular advantages not specifically recited in the above summary.

## BRIEF DESCRIPTION OF THE DRAWINGS

**[0008]** The embodiments of the invention are illustrated by way of example and not by way of limitation in the figures of the accompanying drawings in which like references indicate similar elements. It should be noted that references to "an" or "one" embodiment of the invention in this disclosure are not necessarily to the same embodiment, and they mean at least one.

**[0009]** FIG. **1***a* shows a portable electronic device running an operating system and one or more applications or widgets, in accordance with an embodiment of the invention.

**[0010]** FIG. **1***b* shows a portable electronic device connected to a wireless data network and a wireless telephone network, in accordance with an embodiment of the invention.

**[0011]** FIG. **2** shows a lock screen application running on the portable electronic device while the portable electronic device **1** is in a wake state.

**[0012]** FIG. **3***a* shows the portable electronic device running a lock screen application and a weather widget in a wake state.

**[0013]** FIG. **3***b* shows a configuration panel for a weather widget, in accordance with an embodiment of the invention.

**[0014]** FIG. **4** shows a method for updating data on the portable electronic device using a power saving mode, in accordance with an embodiment of the invention.

**[0015]** FIGS. **5***a*-**5***d* show screen displays of the portable electronic device while the method of FIG. **4** is being performed.

**[0016]** FIG. **6** shows a power saving data update system, in accordance with an embodiment of the invention.

## DETAILED DESCRIPTION

**[0017]** Several embodiments of the invention with reference to the appended drawings are now explained. While numerous details are set forth, it is understood that some embodiments of the invention may be practiced without these details. In other instances, well-known circuits, structures,

MASIMO 1070
Masimo v. Apple
IPR2023-00664

and techniques have not been shown in detail so as not to obscure the understanding of this description.

**[0018]** FIG. **1***a* shows a portable electronic device **1** that is running an operating system and one or more applications or widgets. The portable electronic device **1** may be a personal digital assistant, a smart phone, a portable audio device, or any similar apparatus. For example, the portable electronic device **1** may be an iPhone™ device, an iPad™ device or an iPod™ device by Apple Inc. of Cupertino, Calif. The portable electronic device **1** includes a display screen **2**, a home menu button **4**, and a power on/off button **6**. In one embodiment, the display screen **2** may be a touch screen, which can detect the presence and location of a touch within the display screen **2**, which may be used for essentially all user input. In other embodiments, a physical QWERTY keypad is provided for the user input. The device **1** may also include audio volume buttons and a call ringer on/off button. An example home menu of a graphical user interface is shown in FIG. **1**, with graphic icons representing various applications stored in the device **1** and that can be executed therein.

**[0019]** The display screen **2** displays graphics and text output by applications, widgets, and the operating system running on the portable electronic device **1**. The term "application" is used generically here to encompass both full-featured applications and apps, which typically take up the entire display screen of a smart phone device when opened, and widgets which are miniature programs that have much more limited functionality and can "reside" a lock screen of the device while displaying data that is being periodically updated. A widget can also be an interactive portal to its parent app. Examples include a clock widget, a calendar widget, a stock ticker widget, and a weather widget. Multiple applications may be viewable on the display screen **2** at a single time, or a single application may be expanded to the entire viewing area of the display screen **2**.

**[0020]** The device **1** includes the necessary hardware and software to manage turning on and off the display screen **2** and the sleep and wake modes of the device, to enable efficient use of the device's battery (not shown). The display screen **2** is turned off when the portable electronic device **1** enters the sleep state while the display screen **2** is turned on when the portable electronic device **1** exits the sleep state and enters a wake state. The home button **4** may be used to turn the display screen **2** on (i.e. transition the device **1** from a sleep state to a wake state). The power button **6** may be used to toggle the display screen **2** on and off (i.e. transition the device **1** between wake and sleep states) and to toggle the entire device **1** on and off (i.e. force a power on reset and completely power down the device **1**). For example, when the display screen **2** is on (e.g. lit and in a wake state), pressing the power button **6** for a short duration (e.g. holding it for no more than 1 second) may toggle the display screen **2** off (e.g. unlit and in a sleep state) but not completely power down the device **1**. In one embodiment, when the display screen **2** is off, all touch sensitive controls on the display screen **2** are disabled. While the portable electronic device **1** can still receive data, phone calls and play music/audio when the display screen **2** is off and the device **1** is in a sleep state, user input via the display screen **2** is disabled. Additionally, pressing the power button **6** for a short duration while the display screen **2** is turned off turns the display screen **2** on and the touch sensitivity of the display screen **2** is reactivated.

**[0021]** As noted above, the power button **6** may also be used to toggle power to the entire device **1**. For example, when powered on in either a sleep or wake state, holding the power button **6** down for a relatively long duration, e.g. more than 1 second, powers down the device **1** completely. When powered down, the device **1** cannot transmit or receive data, initiate or answer phone calls, process any data, etc. When the device **1** is powered off, holding the power button **6** down for a relatively long duration causes a power-on-reset for the entire device **1**.

**[0022]** Similar to the power button **6**, the home button **4** may be configured to toggle the device **1** from sleep state to wake state, when pressed for essentially any duration.

**[0023]** The portable electronic device **1** includes an operating system that manages hardware resources and provides common services to various applications running on the operating system. For example, the operating system installed on the portable electronic device **1** may manage memory allocation, network interfaces, audio circuitry, and external interfaces. Example operating systems include iOS by Apple Inc. and Symbian by Nokia.

**[0024]** With the assistance of the installed operating system, the device **1** supports a variety of applications, such as a telephone application for receiving/placing phone calls, a lock screen application, a weather widget, a stock quote widget, a blogging widget, a web browser application, etc. An application running on the portable electronic device **1** may access a network interface through a system call to the installed operating system. The network interface provides the applications with access to networks and external sources of data using a variety of communications standards, protocols, and technologies as introduced above. For example, as shown in FIG. **1***b*, the portable electronic device **1** may include network interfaces that allow the device **1** to connect to both wireless data networks and wireless telephone networks to access weather report and stock quote services.

**[0025]** FIG. **2** shows an "unlock screen" of a lock screen application running on the portable electronic device **1** in a wake state. The display screen **2** is turned on, and the user is being prompted to unlock or enable the user interface of the device **1**. The lock screen application may be a standalone application that runs on top of the operating system, or the lock screen application may be a service that is integrated as part of the operating system.

**[0026]** The lock screen application prevents unauthorized access or inadvertent use of the portable electronic device **1**. As shown in FIG. **2**, with the assistance of the lock screen application, the user interface of the portable electronic device **1** is maintained in a locked state until the user "manually" commands the application to unlock the device **1**. In the locked state, the unlock screen contains a graphic that prompts the user to slide a finger over a predetermined path on the display screen **2** to unlock the device **1**. In other embodiments, the user may be directed to enter an alphanumeric code or other pattern (e.g. a movement pattern) to unlock the device **1**. Note that there may be a password prompt after a physical button has been pressed but before the device **1** is "unlocked", requiring the user to enter a secret password before the user interface of the device **1** can be normally used in its wake state.

**[0027]** In the locked state, the lock screen may display various pieces of data to the user. For instance, in FIG. **2**, time of day, date, data rate, and battery status are shown. The information may instead or in addition include the number of missed calls, the number of unread SMS and/or MMS messages, the number of unheard voicemail messages, etc. In one

MASIMO 1070
Masimo v. Apple
IPR2023-00664

embodiment, one or more widgets may be "placed on the lock screen", such as a weather widget or a stock ticker widget. These may display data that is periodically updated (also referred to as dynamic content) from external sources. In one embodiment, these external sources are accessed through a network interface of the portable electronic device.

**[0028]** FIG. **3***a* shows the portable electronic device **1** running a lock screen application in a wake state. The lock or unlock screen being displayed in this example includes a weather widget **10** that is positioned at a center portion of the lock screen. Although a weather widget **10** is shown and described, any other similar type of widget (e.g. a stock widget that displays up to date stock price data or news relating to a particular stock, a sports widget that displays up to date score data or professional sports news, a blog widget that displays the latest posts made to a particular blog, etc.) may be used. The use of the weather widget **10** is merely for purposes of explanation.

**[0029]** The weather widget **10** shows the current weather for a location designated by the user. For example, the weather widget **10** may show the current temperature, wind direction, wind speed, barometric pressure, humidity, and other similar pieces of data for a designated city or location. The weather widget **10** may retrieve this weather data from an external data source through a network interface using either a push or pull protocol. For example, the external source may be a weather broadcast service or server that is accessible over the Internet, such as The Weather Channel (www.weather.com), The National Weather Service (www.weather.gov), AccuWeather (www.accuweather.com), or a similar service. To access such a server over the Internet, the weather widget **10** utilizes an appropriate network interface of the portable electronic device such as a WiFi wireless local area network interface, and various cellular network interfaces such as a GSM network interface, an EDGE network interface, a HSDPA network interface, a W-CDMA network interface, a CDMA network interface, or a TDMA network interface.

**[0030]** FIG. **3***b* shows a configuration panel **12** that is being displayed once the user interface of the device **1** has been unlocked for the weather widget **10**. The configuration panel **12** allows the user to select various settings for the weather widget **10**. In one of the settings, the user may select a time period between periodic updates of weather data. A time period between periodic updates may be set to a desired number of minutes, hours, or days. Since, weather conditions are not rapidly changing, this level of granularity is sufficient. However, other widgets may require a greater frequency of updates. For example, for a stock quotes widget it may be necessary to adjust periodic updates in terms of seconds, because stock quotes/prices change extremely rapidly.

**[0031]** FIG. **3***b* also shows a configuration setting for activating/deactivating of a power saving mode. The power saving mode regulates when the weather widget **10** updates data. In one embodiment, when activated, the power saving mode restricts the weather widget **10** from downloading data while the portable electronic device is in a sleep state (e.g. the display screen is off). Accordingly, any periodic updates that are scheduled to occur while the portable electronic device **1** is in the sleep state are not performed until the portable electronic device **1** transitions from the sleep state to a wake state.

**[0032]** Although the above description refers to a widget appearing on a lock screen of the portable electronic device **1**, a stand-alone full-featured application running on the portable electronic device **1**, but that does not have a corresponding widget appearing in the lock screen, may also be configured with the power saving mode as described herein. For example, a stand-alone stock application may be running on the portable electronic device **1** in a power saving mode. While the portable electronic device **1** is in a sleep state, the stand-alone stock application is not allowed to update data from an external data source using the network interface. All such scheduled data updates wait for the portable electronic device **1** to enter a wake state before being performed.

**[0033]** FIG. **4** shows a method for updating application data on the portable electronic device **1** while operating in a power saving mode, according to one embodiment of the invention. An application may store the time of its next scheduled data update. At operation **14**, the process begins when a power saving mode for the application has been selected. Selection of the power saving mode may be universal, that is for all applications running on the portable electronic device **1**, using a user configurable setting provided by the operating system. In this embodiment, upon selection of the power saving mode, the operating system may create and update a database with the next scheduled data update of each active application. Alternatively, an application may be individually set to operate in the power saving mode, using a configurable setting provided by the application.

**[0034]** After selecting a power saving mode, the method of FIG. **4** determines if it is time to perform a data update at operation **16**. In embodiments in which each individual application controls their own power saving mode, operation **16** may be performed asynchronously by each application. An application may determine if it is time to perform a data update by comparing the current time (maintained by the operating system) with its next scheduled data update time, which was stored earlier at operation **14**.

**[0035]** In embodiments in which the operating system controls updates for each application that is in a power saving mode, operation **16** is performed synchronously by the operating system for each active application. The operating system may determine if it is time to perform a data update for a given active application by querying a database for scheduled data updates (see operation **14**) that occur at or before the current time. Based on this comparison, the operating system may create a list of applications that are scheduled to update data. Operation **16** repeats until at least one application or widget is to update its data. In one embodiment, operation **16** repeats at a set interval (e.g. every 5 seconds).

**[0036]** Upon the determination that an application or widget is to update its data now, operation **18** determines if the portable electronic device **1** is in a sleep state. As described above, the portable electronic device **1** is in a sleep state when the display screen is turned off. Conversely, when the display screen is turned on, the portable electronic device **1** is in a wake state. Operation **18** continues to operate until the portable electronic device **1** is determined to not be in a sleep state. In embodiments in which the operating system controls data updates for each application that is in a power saving mode, the method may return to operation **16** after operation **18** determines the portable electronic device **1** is in a sleep state. Returning to operation **16** allows the method to add additional applications to the list of applications that need to update their data "now".

**[0037]** In one embodiment, the device **1** determines that it is no longer in a sleep state when the operating system detects or receives a user command to exit the sleep state. For instance,

MASIMO 1070
Masimo v. Apple
IPR2023-00664

a processor interrupt may be signaled by the home menu button being pressed. This would be followed with other operations that are more formally within the definition of wake state, such as turning on the display screen and enabling user input (e.g. alphanumeric text via a touch screen or via a physical keyboard) for unlocking the user interface of the device **1**.

**[0038]** After determining that the portable electronic device **1** is not in a sleep state, operation **20** triggers the "now due" applications in its list to update their data. In embodiments in which the operating system controls data updates for each application that is in a power saving mode, the operating system sends an update message to the application to update its data. In response to the update message, the applications establish a connection with an external data source and then download its latest data. After updates have been completed at operation **20**, the method may clear the list of applications that need to update their data now and returns to operation **16**. The method continues to operate as long as a running application is in a power saving mode.

**[0039]** An example operation of the method of FIG. **4** will be described, while referring to various embodiments. FIGS. **5***a*-**5***d* show a portable electronic device **1** performing the method of FIG. **4**.

**[0040]** FIG. **5***a* shows a lock screen of the portable electronic device **1** that is in a wake state at 12:00 PM. The lock screen includes a weather widget with a power saving mode selected, and a clock widget. The weather widget is scheduled to update every three hours. In this example, the weather widget has just updated and is scheduled to update again at 3:00 PM.

**[0041]** FIG. **5***b* shows the portable electronic device **1** in a sleep state at 12:01 PM. The device may enter the sleep state by pressing the power button for a short duration as explained above. Since the device **1** is in a sleep state, the lock screen and associated widget is not viewable on the display screen **2**. Further any updates scheduled to take place while the portable electronic device **1** is in a sleep state, will not take place until the device **1** enters a wake state.

**[0042]** FIG. **5***c* shows the portable electronic device **1** at 4:00 PM immediately after entering a wake state for the first time since 12:00 PM. The portable electronic device **1** may enter a wake state by a user pressing either the power button **6** or home button **4**, in response to which the display screen **2** is turned on. As seen in FIG. **5***c*, the weather data shown in the weather widget is identical to the weather data from FIG. **5***a*. The data is identical because as explained above no data updates were performed while the portable electronic device **1** was in a sleep state. Even though an update was scheduled for 3:00 PM, this update was rescheduled/delayed until after the portable electronic device **1** entered a wake state. Thus, in this embodiment, the display screen **2** is turned on without first updating any widgets that are visible in it, showing the original state of the widget from its previous update.

**[0043]** In one embodiment, the device **1** determines if the device is in a sleep state by accessing an application programming interface (API) provided by the operating system of the device. The sleep and wake states of the device **1** may be defined using a power management array. The first element in the array contains the structure that describes the sleep state and the second element of the array contains the structure that describes the wake state of the device **1**. For example, the array and the corresponding structures may be defined in objective C as follows:

```
// Declare an array of two IOPMPowerState structures
(kMyNumberOfStates = 2).
static IOPMPowerState myPowerStates[kMyNumberOfStates];
// Zero-fill the structures.
bzero (myPowerStates, sizeof(myPowerStates));
// Fill in the information about the device's 1 sleep state:
myPowerStates[0].version = 1;
myPowerStates[0].capabilityFlags = kIOPMPowerOff;
myPowerStates[0].outputPowerCharacter = kIOPMPowerOff;
myPowerStates[0].inputPowerRequirement = kIOPMPowerOff;
myPowerStates[0].screenPower = kIOPMPowerOff;
// Fill in the information about the device's 1 wake state:
myPowerStates[1].version = 1;
myPowerStates[1].capabilityFlags = kIOPMPowerOn;
myPowerStates[1].outputPowerCharacter = kIOPMPowerOn;
myPowerStates[1].inputPowerRequirement = kIOPMPowerOn;
myPowerStates[1].screenPower = kIOPMPowerOn;
```

**[0044]** The state of the device **1** may thereafter be determined by comparing the value in the power management array against the device's **1** current state. If the device **1** is determined to be in a wake mode, a data update is performed. An example function, which may be used by a widget or standalone application to update data, is shown below:

```
void MyIOServiceDriver::powerSavingDataUpdate (IOService *
whatDevice )
{
   if ( whatDevice.powerState == myPowerStates[1].screenPower) {
      // In wake state.
      performDataUpdate( );
   } else {
      // In sleep state.
      return;
   }
}
```

**[0045]** FIG. **5***d* shows the portable electronic device **1** a short period of time (less than a minute and in general on the order of a few seconds) after the display screen **2** was turned on (in FIG. **5***c*) wake state. The weather widget has performed its data update, which was originally scheduled for 3:00 PM and the data being displayed in the weather widget is now different from the original data shown in FIG. **5***c*.

**[0046]** Note that as an alternative, the display screen **2** could be turned on (in response to the home menu button **4** being pressed) but without the original state of the weather widget being displayed. In that case, there would be a short delay after the screen is turned on, while the weather widget updates its data, before its icon showing the updated data appears on the display screen **2**.

**[0047]** Turning now to FIG. **6**, a power saving data update system **21** will be described. The power saving data update system of FIG. **6** is capable of controlling multiple applications that may be running in a portable electronic device **1**. As will be further described below, the power saving data update system may be used to perform the power saving update method of FIG. **4**.

**[0048]** The power saving data update system **21** operates when at least one application has been selected to use a power saving mode. The applications may individually select to use a power saving mode through a user configuration panel provided by each application and widget. Alternatively, a

MASIMO 1070
Masimo v. Apple
IPR2023-00664

global setting may be provided by the operating system, which requires all applications and widgets to use a power saving mode.

**[0049]** Upon selection of a power saving mode, the database management module **22** creates a data update database **23**. The data update database **23** includes a listing of each active application and details of its respective next scheduled data update, e.g. time and date. The database management module **22** retrieves this information by sending a multicast request to each active application and widget **24**.

**[0050]** After the creation of the data update database **23**, the data update determination module **25** periodically queries the database **23** to determine if a scheduled data update is due for one or more active applications or widgets **24**. The data update determination module **25** may determine a scheduled update is due by querying the data update database **23** for updates that are scheduled to occur on or before the current time. The data update determination module **25** may create a list of applications and widgets that are due to update data. The list may be separate from the data update database **23** or the list may be incorporated into the database **23** as a Boolean field. If no updates are due to occur, the data update module **25** retains control of the system **21** and continues to periodically query the data update database **23**.

**[0051]** Upon determining that a scheduled update is due to occur, the sleep determination module **27** determines if the mobile electronic device **1** is in a sleep state. As described above, a sleep state is defined as the display screen **2** of the mobile electronic device **1** being turned off. If the mobile electronic device **1** is in a sleep state, the sleep determination module **27** passes control of the system back to the data update determination module **28**.

**[0052]** Upon determining that the mobile electronic device **1** is not in a sleep state, the sleep determination module **27** passes control of the system **21** to the data update trigger module **28**. The data update trigger module **28** receives a list of applications and widgets that are due to perform data updates. The list may be separate from the data update database **23** or the list may be incorporated into the database **23** as a Boolean field. Based on the list, the data update trigger module **28** triggers active applications and widgets **24** to update their data.

**[0053]** After triggering all listed applications and widgets, control of the system **21** passes back to the database management module **22**. The database management module **22** updates the next scheduled update in the data update database **23** for each application and widget that has been updated.

**[0054]** To conclude, various aspects of a technique for updating data in applications and widgets running on a portable electronic device have been described. As explained above, an embodiment of the invention may be a machine-readable medium such as one or more solid sate memory devices having stored thereon instructions which program one or more data processing components (generically referred to here as "a processor" or a "computer system") to perform some of the operations described above. In other embodiments, some of these operations might be performed by specific hardware components that contain hardwired logic. Those operations might alternatively be performed by any combination of programmed data processing components and fixed hardwired circuit components.

**[0055]** While certain embodiments have been described and shown in the accompanying drawings, it is to be understood that such embodiments are merely illustrative of and not restrictive on the broad invention, and that the invention is not limited to the specific constructions and arrangements shown and described, since various other modifications may occur to those of ordinary skill in the art. The description is thus to be regarded as illustrative instead of limiting.

What is claimed is:

**1**. A method for updating data in an application running on a portable electronic device, comprising:
- determining whether the application is scheduled to update data from an external source;
- determining whether the portable electronic device is in a sleep state; and
- in response to determining that the portable electronic device is not in the sleep state, updating said data from the external source.

**2**. The method of claim **1**, wherein the sleep state is defined in part by a display screen of the portable electronic device being off.

**3**. The method of claim **1**, wherein updating data from the external source comprises:
- establishing a network connection between the portable electronic device and the external source; and
- downloading data from the external source using the established network connection.

**4**. The method of claim **1**, further comprising:
- creating a database of applications running on the portable electronic device and each of their associated next scheduled update times.

**5**. The method of claim **1**, wherein the external source is an Internet data repository.

**6**. The method of claim **2**, wherein the application is visible on a lock screen application that locks and unlocks the user interface of the portable electronic device and shows said data from the external source.

**7**. The method of updating the active application of claim **6**, wherein the application is a widget that operates within a framework of the lock screen application.

**8**. The method of updating the active application of claim **6**, wherein the lock screen application includes widgets that show one or more of time, date, and battery charge status in addition to data from the external source.

**9**. An article of manufacture comprising:
- a machine readable computer storage medium having stored instructions that program a portable electronic device to:
- determine whether a visible application is due to update data from an Internet source in relation to the current time;
- determine whether a screen of the device is turned on; and
- in response to determining that the screen of the device is turned on, updating data of the visible applications from the Internet source.

**10**. The article of manufacture of claim **9**, further comprising:
- creating a database of visible applications and each of their associated next scheduled update times.

**11**. The article of manufacture of claim **10**, wherein determining whether the one or more visible applications are due to update data comprises:
- querying the database for visible applications that have a next scheduled update time that is at or before the current time; and
- creating a list of visible applications that have a next scheduled update time that is at or before the current time.

MASIMO 1070
Masimo v. Apple
IPR2023-00664

**12**. The article of manufacture of claim **11**, wherein updating data of the one or more visible applications from the one or more Internet sources comprises:

sending a trigger message to the visible applications on the list of visible applications that have a next scheduled update time that is at or before the current time.

**13**. The article of manufacture of claim **9**, wherein the visible applications are visible on the screen of the portable electronic device.

**14**. A computer system that is to run a lock screen application which displays dynamic content to a user, wherein the system can be configured into a mode that allows an update of the dynamic content, from a remote server over the Internet, when the system transitions from a sleep mode to a wake mode and not before while the system is still in the sleep mode.

**15**. A computer system that is to run an unlock application which prompts a user to enter a code before unlocking a user interface of the system, wherein the unlock application displays dynamic content to the user on a display screen of the system prior to unlocking the user interface, and wherein the system can be configured into a mode that allows an update of the dynamic content, from a remote server over the Internet, when the display screen transitions from a screen off state to a screen on state and not before while the system is still in the screen off state.

**16**. A computer system that is to run an unlock application which prompts a user to enter a code before unlocking a user interface of the system, wherein the unlock application displays dynamic content to the user on a display screen of the system prior to unlocking the user interface, and wherein the system can be configured into a mode of operation that allows update of the dynamic content, from a remote server over the Internet, while the display screen is in a screen off state and the system is in a sleep state, only in response to receiving a command from the user.

**17**. The system of claim **16** wherein the command from the user is a result of the user's actuation of a physical switch of the system.

* * * * *

**MASIMO 1070**
**Masimo v. Apple**
**IPR2023-00664**