

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
8 July 2010 (08.07.2010)

(10) International Publication Number
WO 2010/077787 A1

(51) International Patent Classification:
G06F 11/14 (2006.01) **G06F 1/32** (2006.01)
G06F 9/445 (2006.01)

Gunsite Road, Brittons Hill, St. Michael, Barbados 14027 (BB).

(21) International Application Number:
PCT/US2009/067767

(22) International Filing Date:
11 December 2009 (11.12.2009)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/142,502 5 January 2009 (05.01.2009) US
61/142,699 6 January 2009 (06.01.2009) US
61/143,548 9 January 2009 (09.01.2009) US
12/636,558 11 December 2009 (11.12.2009) US

(72) Inventors; and

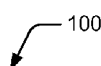
(75) Inventors/Applicants (for US only): **ZHAO, Qun** [US/US]; 1320 Oak Vista Way, Pleasanton, CA 94566 (US). **CHIANG, Hsing-Yi** [CN/—]; 5F No. 39, Ln. 265, Sec 3 Kuang-Ning Rd., Taipei (TW). **KANG, Xinhai** [SG/US]; 1940 Vai Reggio Ct., San Jose, CA 95132 (US). **CHU, Chee, Hoe** [SG/US]; 7163 Brisbane Ct., San Jose, CA 95129 (US). **CHANG, Lihan** [US/US]; 1701 Indigo Oak Lane, San Jose, CA 95121 (US). **LI, Jin-Nan** [US/US]; 448 Casa View Drive, San Jose, CA 95129 (US). **LI, Robyn, Jie** [US/US]; 19514 Greenwood Ct. #4, Cupertino, CA 95014 (US).

(71) Applicant (for all designated States except US): **MARVELL WORLD TRADE LTD.** [BB/BB]; L'Horizon,

(74) Agent: **COLBY, Michael, K.**; Sadler Breen Morasch & Colby, ps, 422 W. Riverside Avenue, Suite 424, Spokane, WA 99201 (US).

[Continued on next page]

(54) Title: METHOD AND SYSTEM FOR HIBERNATION OR SUSPEND USING A NON-VOLATILE-MEMORY DEVICE



(57) Abstract: This disclosure describes techniques for using a non-volatile-memory device such as flash memory to store memory data during hibernation or suspend. By so doing, hard drives and/or data are safer, and less power may be used.

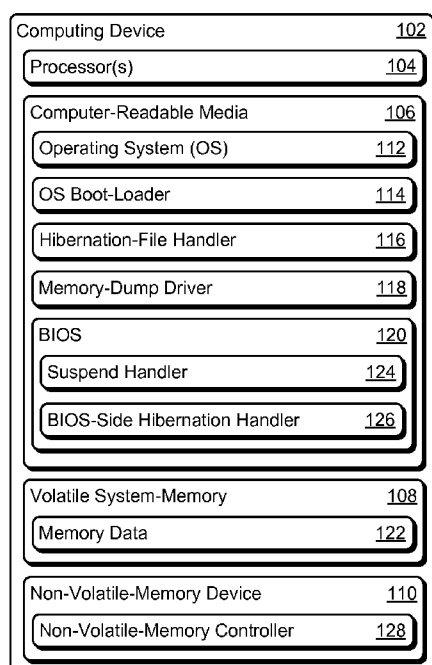


Fig. 1

WO 2010/077787 A1



(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

METHOD AND SYSTEM FOR HIBERNATION OR SUSPEND USING A NON-VOLATILE-MEMORY
DEVICE

RELATED APPLICATIONS

[0001] This application claims priority to U.S. Provisional Patent Application Serial No. 61/142,502 filed January 5th, 2009, the disclosure of which is incorporated by reference herein in its entirety. This application also claims priority to U.S. Provisional Patent Application Serial No. 61/142,699 filed January 6th, 2009, the disclosure of which is incorporated by reference herein in its entirety. This application also claims priority to U.S. Provisional Patent Application Serial No. 61/143,548 filed January 9th, 2009, the disclosure of which is incorporated by reference herein in its entirety.

BACKGROUND

[0002] Modern computing devices employ power-saving modes when not in use. Hibernation is a mode in which an operating system (OS) saves memory data from volatile system-memory to an OS boot partition of a hard drive in the form of a hibernation file, after which the computing device is shut down. When turned

back on, the basic input output system (BIOS) of the computing device posts and then loads an OS boot loader. The OS boot loader copies the memory data within the hibernation file back into the volatile system-memory. The OS boot loader then resumes operation of the operating system where the operating system was paused instead of booting the operating system as normal. This allows for currently running applications to retain their data even if the data was not saved before hibernation.

[0003] Suspend is a mode in which the operating system shuts down power to most of the devices within the computing device but not to the volatile system-memory such that the memory data is preserved. To resume full use, the operating system powers on the devices and resumes operation using the memory data that was preserved. Suspend uses significantly more power than hibernation but is much faster.

[0004] While these modes serve their respective purposes they have undesired limitations. Hibernation mode can be slow to begin and can lose data. Hibernation mode can be slow to begin because it is limited by the speed at which the hard drive can save the memory data to the hibernation file. A computer's data is often stored on a spinning-media hard-disk drive, which is spinning while the hibernation mode is beginning; this presents a data-safety issue. Any substantial movement of the computing device is potentially hazardous while the drive is still spinning. A user who selects to hibernate his laptop, closes the lid, and goes on his way may damage the hard drive and the data it contains. Furthermore, during the time that the hard drive is spinning, both it and the computing device are using

power. This is undesirable if the hibernation occurred due to a critical battery alarm because the computing device may run out of power before the hibernation is complete. Even if the device's battery does not fail, using additional power contradicts the point of a power-saving mode.

[0005] Suspend mode also has undesired limitations. While significantly faster than hibernation, it uses more power because the volatile system-memory remains powered. Furthermore, if the computing device's power source is lost while suspended, the memory data may not be recovered and any information not saved to the hard disk will likely be lost. This can easily occur, such as when the user unplugs the computing device or when a power source fails.

[0006] The background description provided herein is for the purpose of generally presenting the context of the disclosure. Work of the presently named inventors, to the extent the work is described in this background section, as well as aspects of the description that may not otherwise qualify as prior art at the time of filing, are neither expressly nor impliedly admitted as prior art against the present disclosure.

SUMMARY

[0007] This summary is provided to introduce subject matter that is further described below in the Detailed Description and Drawings. Accordingly, this Summary should not be considered to describe essential features nor used to limit the scope of the claimed subject matter.

[0008] In one embodiment, a method is described that comprises monitoring operating status of a computing device, the operating status including a suspend

mode, directing a non-volatile-memory controller to copy memory data from a volatile system-memory into a non-volatile-memory device in response to detecting a notification of the suspend mode, and directing the non-volatile-memory controller to copy the memory data from the non-volatile-memory device into the volatile system-memory in response to receiving a request to resume from the suspend mode. This embodiment may include receiving, from the non-volatile-memory controller, a notification that the memory data has been copied to the non-volatile-memory device, responsive to receiving the notification, shutting down power to the volatile system-memory, and responsive to receiving a request to wake from the suspend mode, powering up the volatile system-memory. This embodiment may include receiving, from the non-volatile-memory controller, a notification that the memory data has been copied from the non-volatile-memory device into the volatile system-memory, the notification indicating that the memory data is available for an operating system to use to resume from the suspend mode.

[0009] In another embodiment, a method is described that comprises monitoring operating status of a computing device, the operating status including a hibernation mode, receiving a notification of the hibernation mode from an operating system of the computing device, and in response to the notification, copying memory data from a volatile system-memory to a non-volatile-memory device as a hibernation file, wherein the non-volatile-memory device is not a memory device from which the operating system is booted. This embodiment may include requesting a hard drive to spin-down, the requesting performed prior to

copying the memory data from the volatile system-memory to the non-volatile-memory device.

[0010] In still another embodiment, a system is described that comprises a non-volatile-memory device, a volatile system-memory, a processor, a hibernation-file handler configured to save a hibernation file to the non-volatile-memory device instead of a hard drive or solid-state-disk from which an operating system is booted, and a BIOS-side hibernation handler configured to redirect a BIOS interrupt 13hex call and cause the hibernation file to be read from the non-volatile-memory device. This embodiment may include a non-volatile-memory controller and a suspend handler configured to direct the non-volatile-memory controller to copy memory data from the volatile system-memory into the non-volatile-memory device in response to a notification of a suspend mode and copy the memory data from the non-volatile-memory device into the volatile system-memory in response to a notification of a resume from the suspend mode. This embodiment may additionally include an on-chip accelerator located within the non-volatile-memory controller, the on-chip accelerator configured to compress the memory data when copying the memory data into the non-volatile-memory device and decompress the memory data when copying the memory data into the volatile system-memory.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] The detailed description is described with reference to the accompanying figures. In the figures, the left-most digit of a reference number identifies the figure in which the reference number first appears. The use of the

same reference numbers in different instances in the description and the figures indicate similar or identical items.

[0012] Fig. 1 illustrates an example operating environment configured to enable hibernation or suspend using a non-volatile-memory device.

[0013] Fig. 2 illustrates a method for suspending or resuming a computing device using a non-volatile-memory device.

[0014] Fig. 3 illustrates a method for hibernating a computing device using a non-volatile-memory device.

[0015] Fig. 4 illustrates a method for resuming a computing device from hibernation using a non-volatile-memory device.

DETAILED DESCRIPTION

[0016] As noted in the Background above, conventional methods of implementing hibernation and suspend modes have undesired limitations. Hibernation mode is slow and may damage hard drives. Suspend mode uses more power and memory data may not be preserved during a power failure. The present disclosure describes techniques for using a non-volatile-memory device such as flash memory to store the memory data during hibernation or suspend. By so doing, hard drives and/or data are safer, and less power may be used.

[0017] In the discussion that follows, an example operating environment is described. Example methods are also described that may be employed in the example operating environment as well as other environments. In the discussion below, reference will be made to the environment by way of example only and,

therefore, implementations described below are not limited to the example environment.

[0018] Example Operating Environment

Fig. 1 illustrates an example operating environment 100 having a computing device 102. Computing device 102 includes one or more processors 104, one or more computer-readable media 106, volatile system-memory 108, and non-volatile-memory device 110. Computer-readable media 106 may include various kinds of media, such as volatile (e.g., Static Random Access Memory, or SRAM) and non-volatile memory (e.g. flash memory, BIOS Chip, solid state disk, spinning-media hard-disk drive, or CD/DVD). Computer-readable media 106 may include volatile system-memory 108, non-volatile-memory device 110, and/or any other computer-readable media. Volatile system-memory 108 loses data when power is removed.

[0019] Non-volatile-memory device 110 retains data when power is removed. Non-volatile-memory device 110 may include non-volatile memory such as flash memory or solid state disks. Non-volatile-memory device 110 can have a storage capacity as small as the storage capacity of volatile system-memory 108 or even smaller if compression is used. In computing devices that implement hibernation, non-volatile-memory device 110 does not include an OS bootable spinning-media hard-disk drive or an OS bootable solid state disk. OS bootable memory are those that include a bootable partition for operating system 112 and OS boot-loader 114. In computing devices that implement both hibernation and

suspend, there may be a separate non-volatile-memory device for each of hibernation and suspend.

[0020] Mechanisms for interfacing non-volatile-memory device 110 with other components of computing device 102 may include, but are not limited to, the use of an internal data bus, a Small Computer System Interface (SCSI), an Internet Small Computer System Interface (iSCSI), a Peripheral Component Interconnect bus (PCI), a PCI eXtended bus (PCI-X), a Peripheral Component Interconnect Express bus (PCIe), a Universal Serial Bus (USB), a Serial ATA bus (SATA), a Serial Attached SCSI bus (SAS), or a Fiber Channel network (FC). For example, non-volatile-memory device 110 may be a flash memory device that is communicatively attached to the other components of computing device 102 through a PCIe connection.

[0021] Computer-readable media 106 is shown including operating system (OS) 112, OS boot-loader 114, hibernation-file handler 116, memory-dump driver 118, and BIOS 120. Operating system 112 is configured to operate computing device 102. OS boot-loader 114 is typically installed when installing operating system 112. During a normal boot of computing device 102, BIOS 120 loads OS boot-loader 114, which then loads operating system 112.

[0022] Hibernation-file handler 116 can be software and/or hardware that intercepts memory data 122 to be written to a hard drive and redirects it to non-volatile-memory device 110. In some cases, hibernation-file handler 116 is a SCSI miniport-driver that intercepts memory data 122 from memory-dump driver 118 when driver 118 attempts to write memory data 122 to the hard drive. In other

cases, hibernation-file handler 116 is software that intercepts a command from operating system 112 to memory-dump driver 118 to request storage of a hibernation file on a spinning-media hard-disk drive. Hibernation-file handler 116 intercepts this command and saves the hibernation file to memory device 110 instead of the hibernation file being stored on the spinning-media hard-disk drive. Alternatively, memory-dump driver 118 is configured to store the hibernation file on memory device 110. In such a case, environment 100 may not include a separate hibernation-file handler 116. In systems that do not implement a hibernation feature, environment 100 may not include hibernation-file handler 116 and memory-dump driver 118.

[0023] BIOS 120 is shown including suspend handler 124 and BIOS-side hibernation handler 126, though both may not be present if only one of hibernation or suspend is implemented. Suspend handler 124 comprises computer instructions configured to request that non-volatile-memory controller 128 copy memory data 122 in and out of volatile system-memory 108 and non-volatile-memory device 110. Suspend handler 124 may also provide non-volatile-memory controller 128 with the address or addresses of one or more locations within volatile system-memory 108.

[0024] BIOS-side hibernation handler 126 comprises computer instructions configured to redirect requests from OS boot-loader 114. The requests may include BIOS INT 13_{hex} requests, which are redirected to read from non-volatile-memory device 110 instead of a hard disk intended by the requests. This redirection causes the hibernation file to be loaded from non-volatile-memory

device 110 on system boot up. Operating system 112 then uses the loaded memory data 122 to resume from hibernation.

[0025] Volatile system-memory 108 is shown including memory data 122, which is preserved in non-volatile-memory device 110 during hibernation or suspend. Non-volatile-memory device 110 includes non-volatile-memory controller 128. Non-volatile-memory controller 128 is configured to copy memory data 122 into and out of non-volatile-memory device 110. Memory data 122 may be compressed prior to being saved to non-volatile-memory device 110 and decompressed prior to being saved to volatile system-memory 108. This compression is performed or aided by software within suspend handler 124 or by an on-chip accelerator, or a combination of both. The on-chip accelerator may be located within non-volatile-memory controller 128. Note that one or more of the entities shown in Fig. 1 may be further divided, combined, and so on. Thus, environment 100 illustrates some of many possible environments capable of employing the described techniques.

[0026] Example Methods

The present disclosure describes techniques for suspending or hibernating a computing device using a non-volatile-memory device to preserve the memory data. This allows for additional power savings as well as enhanced data safety. These techniques are described using three different methods, though they may act independently or in combination. Aspects of these methods may be implemented in hardware, firmware, software, or a combination thereof. The methods are

shown as a set of acts that specify operations performed by one or more entities and are not necessarily limited to the order shown.

[0027] Fig. 2 illustrates a method 200 for suspending or resuming a computing device using a non-volatile-memory device. At 202, an operating status of a computing device is monitored. The operating status includes a suspend mode. At 204, a notification of the suspend mode is received from an operating system (OS) of the computing device. For example, suspend handler 124 (Fig. 1) within BIOS 120 receives the notification of suspend from operating system 112.

[0028] At 206, a non-volatile-memory controller is directed to copy memory data from volatile system-memory into a non-volatile-memory device. For example, suspend handler 124 issues a request to have non-volatile-memory controller 128 copy memory data 122 from volatile system-memory 108 into non-volatile-memory device 110. The request may include one or more addresses of corresponding locations within volatile system-memory 108 to copy. In this case, controller 128 copies memory data 122 that is addressed. In alternative cases, if no addresses are provided, controller 128 copies all of memory data 122.

[0029] Storing memory data 122 in non-volatile-memory device 110 provides extra data security because non-volatile-memory device 110 retains its data during a power failure. A power failure during a conventional suspend can result in a hard shut-down of computing device 102, which results in memory data 122 being lost. Powering on from this state requires a normal system boot, which takes considerably longer than resuming from a suspended state.

[0030] At 208, a notification indicating that the memory data has been copied is received from the non-volatile-memory controller. For example, suspend handler 124 receives a notification from controller 128 that memory data 122 has been copied into non-volatile-memory device 110.

[0031] At 210, power to the volatile system-memory is shut down in response to receiving the notification at 208. Here, suspend handler 124 or other components within BIOS 120 shut down power to volatile system-memory 108. This saves additional power while computing device 102 is in suspend mode.

[0032] At 212, the volatile system-memory is powered on in response to receiving a request to resume from the suspend mode. This is implemented if 210 is implemented. The request to resume from suspend may come from various sources, such as a keyboard key press, a mouse movement or click, a system wake-up event, or a wake-on-LAN request. For example, suspend handler 124 receives a request to resume and powers on volatile system-memory 108.

[0033] At 214, the non-volatile-memory controller is directed to copy the memory data from the non-volatile-memory device into the volatile system-memory. The directing at 214 is similar to the directing at 206 except that the memory data is requested to be copied into volatile system-memory instead of out of volatile system-memory. At 214, the contents of the volatile system-memory are restored to their pre-suspend state. This will allow the operating system to continue operation from where it left off before the suspend paused its operation. Continuing the example, suspend handler 124 requests that non-volatile-memory

controller 128 copy memory data 122 from non-volatile-memory device 110 into volatile system-memory 108.

[0034] The techniques may optionally compress and decompress the memory data prior to storing it on the non-volatile-memory device and the volatile system-memory, such as through use of an on-chip accelerator added to non-volatile-memory controller 128 or use of suspend handler 124. Compressing the memory data permits use of fewer memory resources to store the contents of the volatile-system-memory. If the compression and decompression is sufficiently fast, it may speed up the storage and retrieval of the memory data because less time is spent saving and retrieving the memory data on the non-volatile-memory device.

[0035] At 216, a notification indicating that the memory data has been copied is received. After such notification, the conventional method of resuming from suspend can, but is not required to, be implemented as if a non-volatile-memory device was not used to save the memory data. For example, suspend handler 124 receives a notification from controller 128 indicating that memory data 122 is restored to volatile system-memory 108. Suspend handler 124 or other components within BIOS 120 may continue to act to resume computing device 102 using conventional methods. Operating system 112 is able to power back on any devices that were turned off during the suspend and use memory data 122 to continue operation as though no suspend occurred.

[0036] Fig. 3 illustrates a method 300 for hibernating a computing device using a non-volatile-memory device to store a hibernation file. Method 300 may be implemented on a computing device that also implements method 200.

[0037] At 302, an operating status of a computing device is monitored. The operating status includes a hibernation mode. At 304, a notification of the hibernation mode is received from an operating system (OS) of the computing device. At 306, memory data from volatile system-memory is copied to a non-volatile-memory device in the form of a hibernation file. During conventional hibernation, the operating system loads a memory-dump driver, which dumps memory data to a hibernation file on an OS boot partition located on a hard drive. In this method, the memory-dump driver is modified and/or a hibernation-file handler is used. The hibernation-file handler may comprise computer software added to a computing device, such as computing device 102 of Fig. 1.

[0038] By way of example, hibernation-file handler 116 receives a notification of hibernation. Hibernation-file handler 116 intercepts each write command from memory-dump driver 118 and redirects each write command to non-volatile-memory device 110. This causes the hibernation file to be stored on non-volatile-memory device 110 instead of a hard drive intended by the write command(s).

[0039] In another example, hibernation-file handler 116 intercepts a command from operating system 112 to memory-dump driver 118. The command is originally intended to instruct memory-dump driver 118 to save memory data 122 to a hard drive. Hibernation-file handler 116 intercepts this command and

dumps memory data 122 to a hibernation file. The hibernation file is saved to non-volatile-memory device 110 instead of the hard drive intended by the command.

[0040] In another example, memory-dump driver 118 is modified to copy memory data 122 to non-volatile-memory device 110 instead of the OS boot partition of the hard drive. In this case, the method forgoes the use of a separate hibernation-file handler 116.

[0041] At 308, the operating system is informed that the hibernation file has been saved. This allows the operating system to continue shutting down power to various components of the computing device. In the example in which hibernation-file handler 116 intercepts each write command from memory-dump driver 118, handler 116 will notify driver 118, which in turn will notify operating system 112. In the example in which hibernation-file handler 116 intercepts a command from operating system 112 to memory-dump driver 118, handler 116 sends operating system 112 a notification when done copying memory data 122. In the example in which memory-dump driver 118 has been modified, driver 118 will notify operating system 112. The notification at 308 allows operating system 112 to continue shutting down the system in hibernation mode.

[0042] Storing the hibernation file on a non-volatile-memory device that is separate from an OS boot disk allows for faster hibernation and resumption in cases where the non-volatile-memory device is faster than the OS boot disk. The OS boot disk is a disk that has a boot partition for the OS. In an example in which the OS boot disk is a solid-state-disk, this method can still be beneficial as faster memory can be used to store the hibernation file. The size requirements for the OS

boot disk are quite large and thus it can be cost prohibitive to use large amounts of the fastest memory. Furthermore, use of a solid-state-disk as the OS boot disk is often cost prohibitive because the cost per capacity is greater than with spinning-media hard-disk drives. The non-volatile-memory device of method 300 can be small enough to only store a hibernation file. In such a case it is often economical to use the fastest memory available.

[0043] Additionally the techniques may request one or more spinning-media hard-disk drives in the computing device to spin-down. This allows for additional power savings and data safety. As a separate non-volatile-memory device is used to store the hibernation file, spinning-media hard-disk drives are no longer needed as soon as the hibernation process begins and can safely be spun down. This act is performed prior to 306 if maximum power savings and data safety are desired.

[0044] Fig. 4 illustrates a method 400 for resuming a computing device from hibernation, which may be implemented on a computing device that also implements method(s) 200 and/or 300.

[0045] At 402, a request to read a hibernation file from a boot disk is received. In some conventional approaches, when hibernation is to be terminated, the OS boot loader uses BIOS interrupt 13_{hex} to read the hibernation file from the OS boot disk. In this method, a BIOS-side hibernation handler can instead receive the request. For example, BIOS-side hibernation handler 126 of Fig. 1 receives the request to read all or part of the hibernation file. The request is received from OS boot-loader 114.

[0046] At 404, the request to read the hibernation file from the OS boot disk is redirected and the hibernation file is read from a non-volatile-memory device. Continuing the example, BIOS-side hibernation handler 126 reads the hibernation file from non-volatile-memory device 110 instead of from the OS boot disk as requested. If the hibernation file is not found, BIOS-side hibernation handler 126 may then read from the OS boot disk. Once memory data 122 within the hibernation file is loaded into volatile system-memory 108, operating system 112 resumes operation using memory data 122.

[0047] Note that OS boot-loader 114 may be modified to read the hibernation file from non-volatile-memory device 110 and as such, no redirection is used. In this case, BIOS-side hibernation handler 126 is part of OS boot-loader 114, such that the modified OS boot-loader requests the hibernation file from the non-volatile-memory device and loads the memory data that is within the hibernation file into volatile system-memory.

[0048] One or more of the techniques described above can be performed by one or more programmable processors executing a computer program to perform functions by operating on input data and generating output. Generally, the techniques can take the form of an entirely hardware embodiment, an entirely software embodiment, or an embodiment containing both hardware and software components. In one implementation, the methods are implemented in software, which includes but is not limited to firmware, resident software, microcode, etc. Furthermore, the methods can take the form of a computer program product accessible from a computer-usable or computer-readable medium providing

program code for use by or in connection with a computer or any instruction execution system.

[0049] For the purposes of this description, a computer-usable or computer-readable medium can be any apparatus that can contain, store, communicate, propagate, or transport the program for use by or in connection with the instruction execution system, apparatus, or device. The medium can be an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system (or apparatus or device) or a propagation medium. Examples of a computer-readable medium include a semiconductor or solid state memory, magnetic tape, a removable computer diskette, a random access memory (RAM), a read-only memory (ROM), a rigid magnetic disk and an optical disk. Current examples of optical disks include compact disk – read only memory (CD-ROM), compact disk – read/write (CD-R/W) and DVD.

[0050] Although the subject matter has been described in language specific to structural features and/or methodological techniques and/or acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features, techniques, or acts described above, including orders in which they are performed.

CLAIMS

What is claimed is:

1. A method comprising:

monitoring operating status of a computing device, the operating status including a suspend mode;

directing a non-volatile-memory controller to copy memory data from a volatile system-memory into a non-volatile-memory device in response to detecting a notification of the suspend mode; and

directing the non-volatile-memory controller to copy the memory data from the non-volatile-memory device into the volatile system-memory in response to receiving a request to resume from the suspend mode.

2. The method as recited in claim 1, further comprising:

receiving, from the non-volatile-memory controller, a notification that the memory data has been copied to the non-volatile-memory device;

responsive to receiving the notification, shutting down power to the volatile system-memory; and

responsive to receiving a request to wake from the suspend mode, powering up the volatile system-memory.

3. The method as recited in claim 1, further comprising receiving, from the non-volatile-memory controller, a notification that the memory data has been copied from the non-volatile-memory device into the volatile system-memory, the notification indicating that the memory data is available for an operating system of the computing device to use to resume from the suspend mode.
4. The method as recited in claim 1, wherein the notification of the suspend mode is received from an operating system of the computing device and comprises:
 - a request to enter the suspend mode; or
 - a notification that the computing device is going to suspend.
5. The method as recited in claim 1, wherein the volatile system-memory includes static random access memory (SRAM).
6. The method as recited in claim 1, wherein the non-volatile-memory device includes flash memory.

7. The method as recited in claim 1, wherein:

directing the non-volatile-memory controller to copy the memory data from the volatile system-memory into the non-volatile-memory device includes communicating a source address corresponding to a location within the volatile system-memory to the non-volatile-memory controller; and

directing the non-volatile-memory controller to copy the memory data from the non-volatile-memory device into the volatile system-memory includes communicating a destination address corresponding to a location within the volatile system-memory to the non-volatile-memory controller.

8. A method comprising:

monitoring operating status of a computing device, the operating status including a hibernation mode;

receiving a notification of the hibernation mode from an operating system of the computing device; and

in response to the notification, copying memory data from a volatile system-memory to a non-volatile-memory device as a hibernation file, wherein the non-volatile-memory device is not a memory device from which the operating system is booted.

9. The method as recited in claim 8, wherein receiving the notification of the hibernation mode includes intercepting a command from the operating system to a memory-dump driver, the command comprising instructions to save the memory data from the volatile system-memory to a hard-drive.

10. The method as recited in claim 8, wherein copying the memory data from the volatile system-memory to the non-volatile-memory device as the hibernation file includes intercepting a data write command from a memory-dump driver to a hard drive and redirecting the data write command to the non-volatile-memory device.

11. The method as recited in claim 8, wherein receiving the notification includes receiving a command from the operating system, the command comprising instructions to save the memory data from the volatile system-memory to the non-volatile-memory device as the hibernation file.

12. The method as recited in claim 8, wherein the non-volatile-memory device includes flash memory.

13. The method as recited in claim 8, further comprising requesting a hard drive to spin-down, the requesting performed prior to copying the memory data from the volatile system-memory to the non-volatile-memory device.

14. The method as recited in claim 8, wherein copying the memory data from the volatile system-memory to the non-volatile-memory device as the hibernation file enables the operating system to resume from the hibernation mode using the memory data that was saved in the hibernation file.

15. The method as recited in claim 14, wherein a Basic Input Output System (BIOS) is modified to redirect a call from an OS boot loader and the modified BIOS causes the hibernation file to be read from the non-volatile-memory device instead of a hard disk.

16. The method as recited in claim 15, wherein the call that is redirected is a BIOS interrupt 13_{hex} call.

17. A computing device comprising:

a non-volatile-memory device;

a volatile system-memory;

a processor;

a hibernation-file handler configured to save a hibernation file to the non-volatile-memory device instead of a hard drive or solid-state-disk from which an operating system is booted; and

a BIOS-side hibernation handler configured to redirect a BIOS interrupt 13_{hex} call and cause the hibernation file to be read from the non-volatile-memory device.

- 18.** The computing device as recited in claim 17, further comprising:
- a non-volatile-memory controller; and
 - a suspend handler configured to direct the non-volatile-memory controller to:

- copy memory data from the volatile system-memory into the non-volatile-memory device in response to a notification of a suspend mode;
 - and

- copy the memory data from the non-volatile-memory device into the volatile system-memory in response to a notification of a resume from the suspend mode.

- 19.** The computing device as recited in claim 18, further comprising:
- an on-chip accelerator located within the non-volatile-memory controller,
- the on-chip accelerator configured to:

- compress the memory data when copying the memory data into the non-volatile-memory device; and

- decompress the memory data when copying the memory data into the volatile system-memory.

20. The computing device as recited in claim 17, wherein the hibernation-file handler is a Small-Computer-System-Interface (SCSI) miniport-driver configured to:

intercept a data write command from a memory-dump driver to the hard drive; and

redirect the data write command to the non-volatile-memory device.

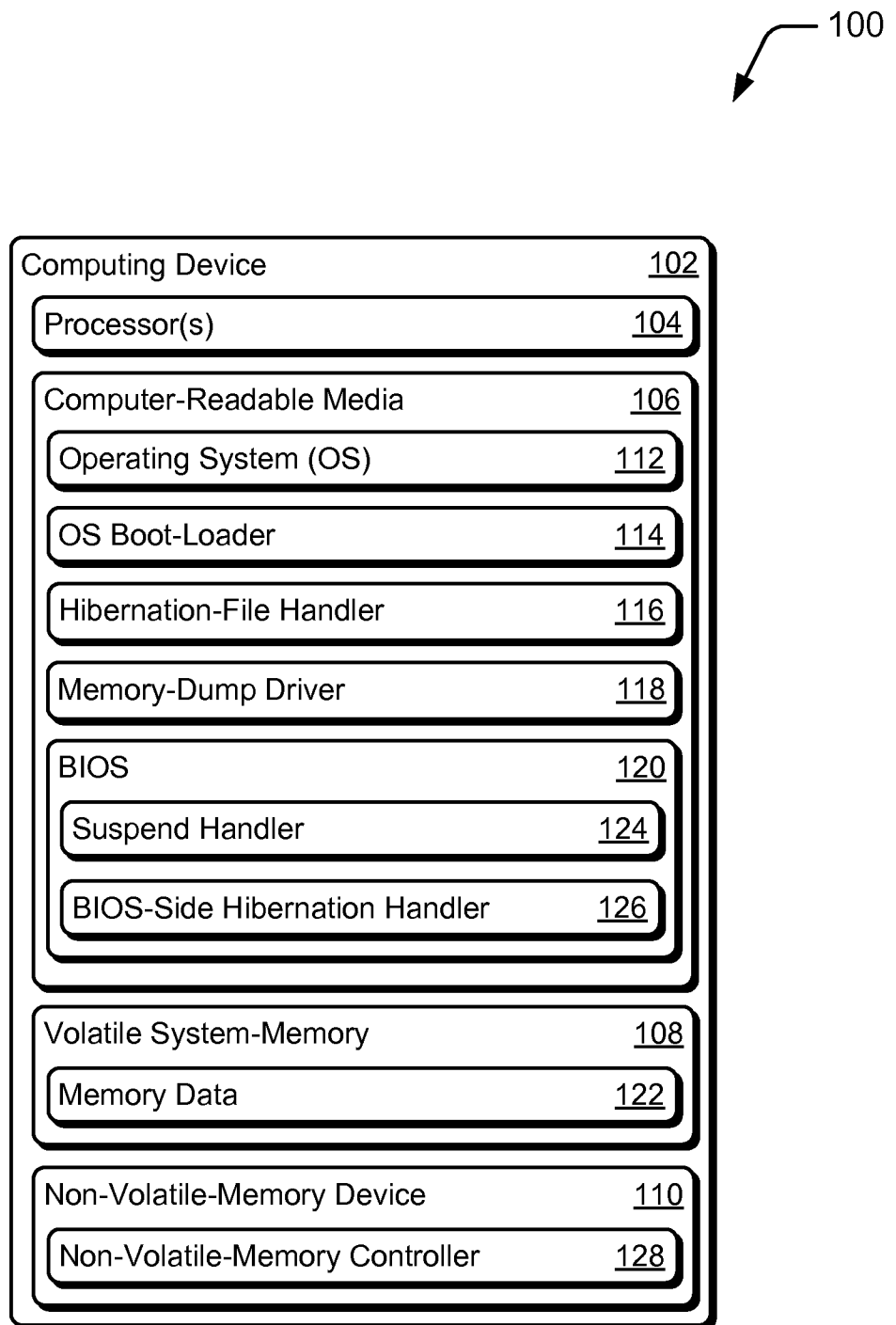
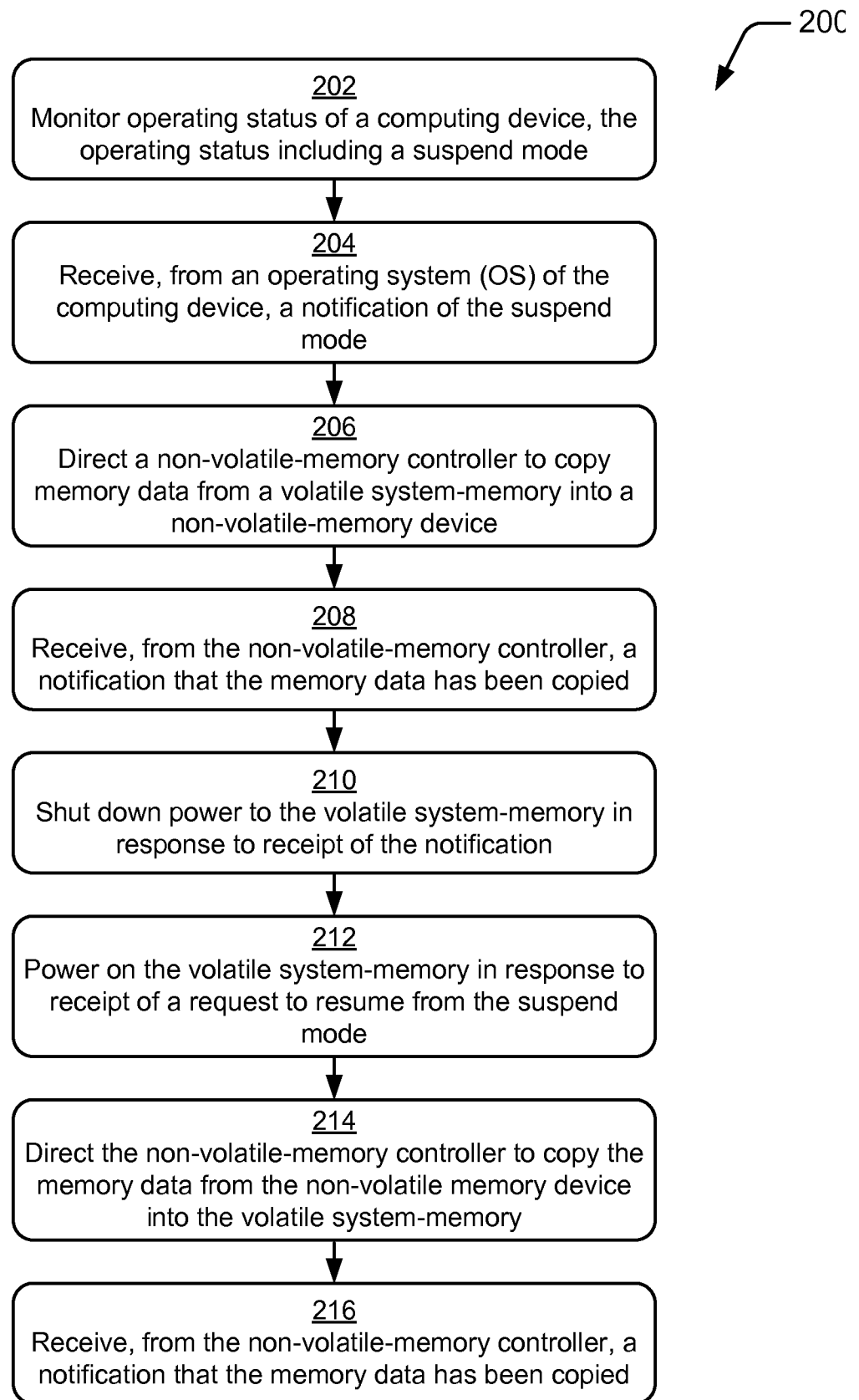
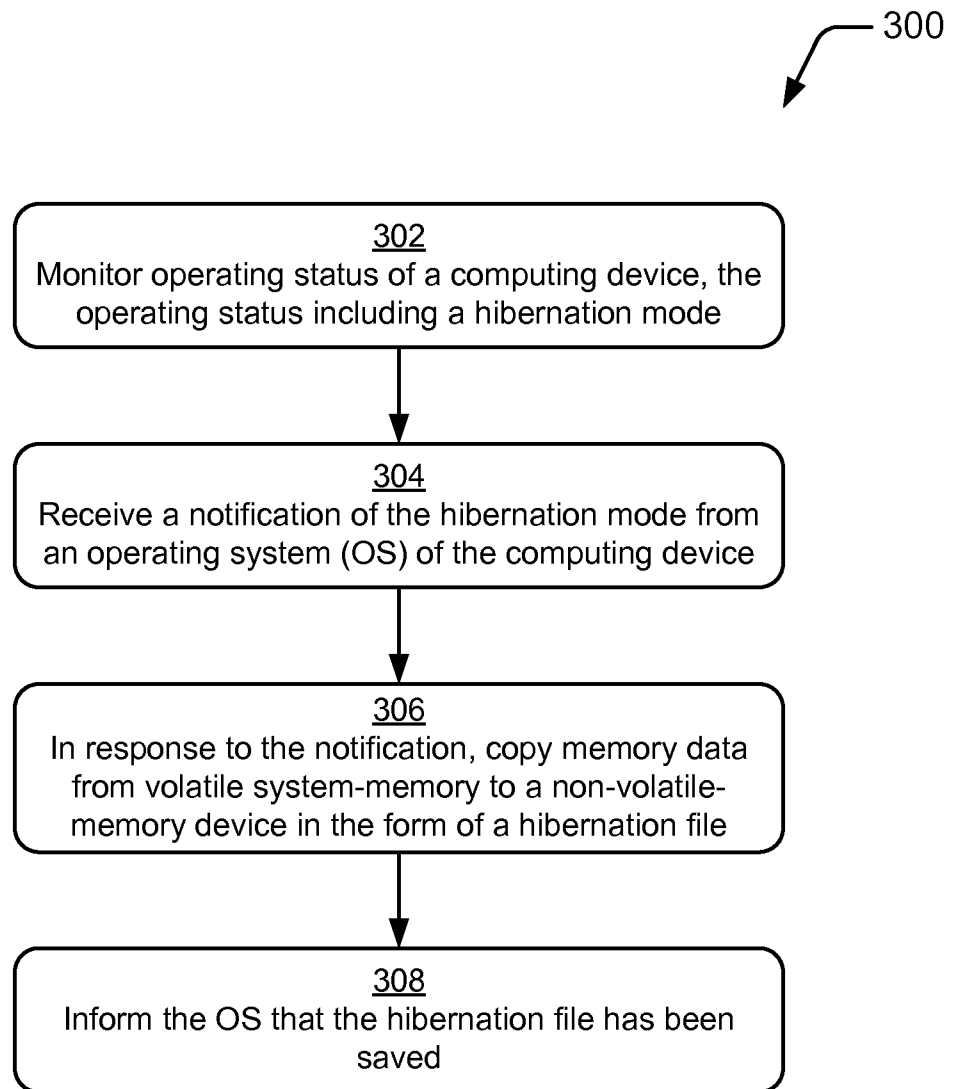
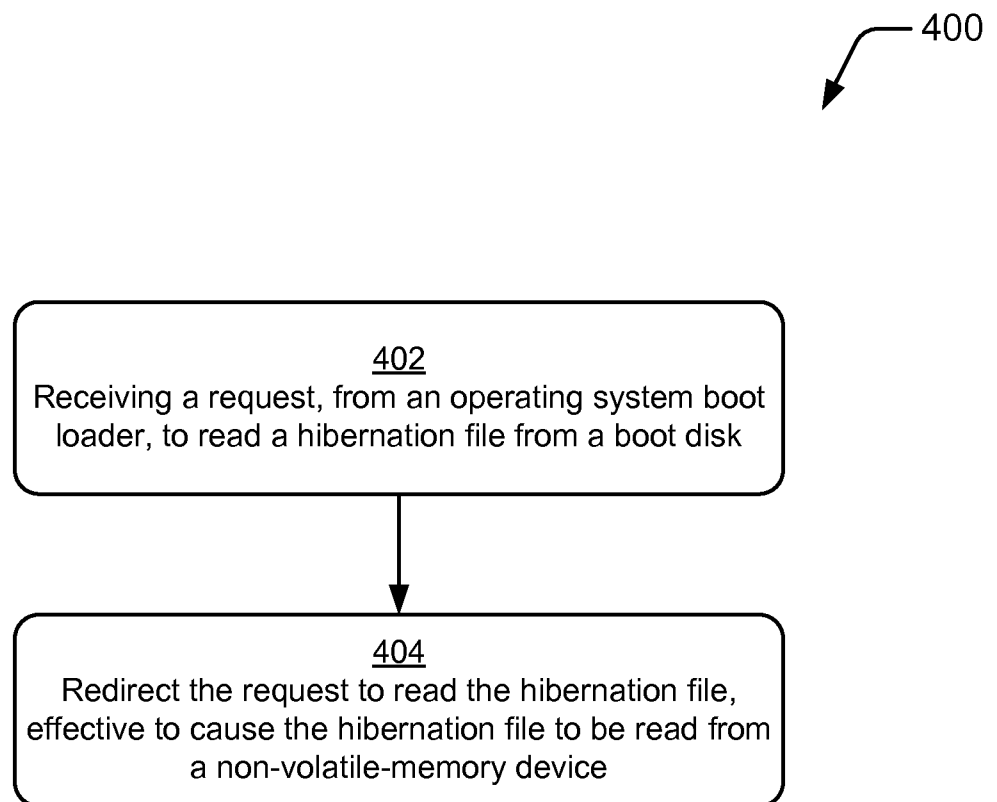


Fig. 1

**Fig. 2**

**Fig. 3**

**Fig. 4**

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2009/067767

A. CLASSIFICATION OF SUBJECT MATTER

INV. G06F11/14 G06F9/445 G06F1/32

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2007/234028 A1 (ROTHMAN MICHAEL A [US] ET AL) 4 October 2007 (2007-10-04) abstract figures 4,8 paragraphs [0081] - [0090]	1-20
X	US 2005/086551 A1 (WIRASINGHE MARCO [US] ET AL) 21 April 2005 (2005-04-21) abstract figure 1b paragraphs [0011] - [0021]	1-20
A	US 2002/087816 A1 (ATKINSON LEE W [US] ET AL) 4 July 2002 (2002-07-04) abstract paragraphs [0057], [0066], [0069], [0072], [0074] ----- -/--	1-20



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

19 March 2010

Date of mailing of the international search report

26/03/2010

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Weber, Vincent

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2009/067767

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2003/208675 A1 (BUROKAS GINTAUTAS [US] ET AL) 6 November 2003 (2003-11-06) abstract figures 1,5 paragraphs [0020], [0037], [0038], [0039], [0041], [0043] -----	1-20
A	US 2007/277051 A1 (REECE DEAN [US] ET AL) 29 November 2007 (2007-11-29) abstract figure 5 paragraphs [0009], [0010] -----	1-20

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2009/067767

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2007234028 A1	04-10-2007	CN 101320314 A EP 1983431 A2	10-12-2008 22-10-2008
US 2005086551 A1	21-04-2005	NONE	
US 2002087816 A1	04-07-2002	US 2003131206 A1	10-07-2003
US 2003208675 A1	06-11-2003	US 2006010316 A1	12-01-2006
US 2007277051 A1	29-11-2007	US 2010037076 A1	11-02-2010