

(12) 发明专利申请

(10) 申请公布号 CN 102163151 A

(43) 申请公布日 2011. 08. 24

(21) 申请号 201010510204. 1

(22) 申请日 2010. 10. 13

(30) 优先权数据

12/579, 022 2009. 10. 14 US

(71) 申请人 中正大学

地址 中国台湾嘉义县

(72) 发明人 罗习五 蔡玮轩

(74) 专利代理机构 北京科龙寰宇知识产权代理

有限责任公司 11139

代理人 孙皓晨

(51) Int. Cl.

G06F 9/445 (2006. 01)

G06F 1/32 (2006. 01)

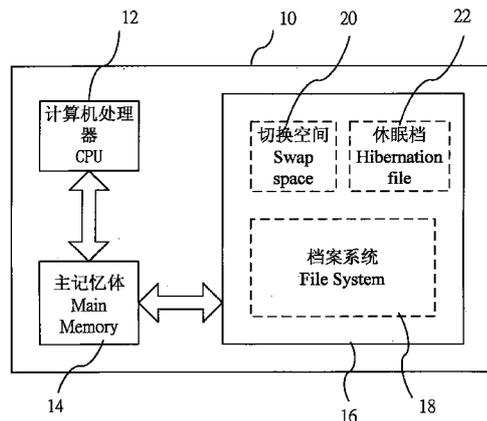
权利要求书 1 页 说明书 5 页 附图 2 页

(54) 发明名称

快速开关机的方法

(57) 摘要

本发明揭露一种快速开关机的方法,其是在电脑系统进入休眠时,以减少存储器转换存放至第二储存装置的数量,进而加快电脑系统从休眠重新启动至全速工作的速度。本发明可应用于各式电脑系统中,且能搭配采用随机存取处理技术写入与载回资料,此外,本发明能防止电脑系统因突然丧失电力资料的遗失与受损程度。



1. 一种快速开关机的方法,其特征在于,包括下列步骤:

一电脑系统进入休眠时,该电脑系统将至少一存储器区分为多个清白存储页面与多个非清白存储页面,且该清白存储页面换出放弃储存,该非清白存储页面写入休眠档储存于至少一第二储存装置的一切换空间中或一档案系统中;

该电脑系统重新启动,该电脑系统由该切换空间与该档案系统读取该休眠档,并将该休眠档回复储存至该存储器;以及

该电脑系统从该第二储存装置读取资料,并载入该资料至该存储器。

2. 如权利要求 1 所述的快速开关机的方法,其特征在于,在该电脑系统进入休眠的步骤中,该清白存储页面系换出完成后,将释放该存储器。

3. 如权利要求 2 所述的快速开关机的方法,其特征在于,在该电脑系统进入休眠的步骤中,将以一核心程序配置该存储器,控制该清白存储页面系换出,且于该清白存储页面系换出完成后,控制释放该存储器。

4. 如权利要求 3 所述的快速开关机的方法,其特征在于,该核心程序能够计算需换出的该清白存储页面的数量,并依据该数量配置该存储器。

5. 如权利要求 1 所述的快速开关机的方法,其特征在于,该第二储存装置为一快闪存储器。

6. 如权利要求 1 所述的快速开关机的方法,其特征在于,该第二储存装置为高速随机存取的存储器装置。

7. 如权利要求 1 所述的快速开关机的方法,其特征在于,在该电脑系统进入休眠的步骤中,该电脑系统能够作为区分该存储器的单位。

8. 如权利要求 1 所述的快速开关机的方法,其特征在于,该第二储存装置具有与该清白存储页面相一致的备份存放。

9. 如权利要求 1 所述的快速开关机的方法,其特征在于,在该电脑系统进入休眠的步骤中,欲换出的该非清白存储页面将依据位址排序,且将位址连续的该非清白存储页面合并成一单一写入命令,组合写入该第二储存装置。

快速开关机的方法

技术领域

[0001] 本发明涉及一种电脑系统开关机方法,特别涉及一种应用于电脑系统在进入省电模式后,可以快速开关机的方法。

背景技术

[0002] 一般电脑系统工作的状态,除了正在执行作业系统与各种应用程式的模式外,更包含了省电模式,电脑系统进入休眠的目地一方面为节省电能的消耗,另一方面为降低系统工作的杂讯。

[0003] 依据高级配置与电源接口 (Advanced Configuration & Power Interface, ACPI) 的设计分类,最常见的休眠将为两种型态,一种为 Suspend To RAM (STR),称为 S3 状态, S3 状态为只对部份的易失性存储器持续供电,包括绘图缓冲存储器 (Frame Buffer) 与主存储器 (Main Memory) 等,其余部分一律断电。S3 状态具有两个优点,一为电脑系统回返到全速运作的时间较快;另一为安全性较佳,当使用者的应用与操作属于个人私密资料不希望回存至硬碟内时,即采用 S3 状态。另一种为 spend To Disk (STD),称为 S4 状态, S4 状态是将工作中的资料全部皆写入到非易失性存储器内储存,然后系统将全部断电, S4 的优点为其更为省电。

[0004] 承上所述两种休眠 S3 状态与 S4 状态,皆不尽完善,在进入 S3 状态后,仍需提供电力给电脑系统中易失性存储器,以维持系统储存的状态;假设改采用 S4 状态虽相较于 S3 状态省电,但回返至电脑系统全速运作的速度较则相对于 S3 状态缓慢许多。

[0005] 有鉴于此,本发明针对上述的缺失,改良以上所述的休眠,提出一种快速开关机的方法,能够于进入休眠时能减少目前存储器使用量,进而增进电脑系统重新启动至全速运作的反应效率。

发明内容

[0006] 本发明的主要目的在于提供一种快速开关机的方法,其在电脑系统进入休眠时大量缩减目前系统中使用的存储器数量,以减少主存储器资料转换存放至第二储存装置的资料量,进而增进电脑系统恢复至全速工作的速度。

[0007] 本发明的另一目的在于提供一种快速开关机的方法,其可预防电脑系统意外丧失电力时资料的遗失与受损的程度。

[0008] 本发明的再一目的在于提供一种快速开关机的方法,其资料写入或载回皆采用随机存取处理的技术,将大幅提升电脑系统进入休眠或重新启动彼此转换时资料存取的速度。

[0009] 为达到上述的目的,本发明提出的快速开关机的方法,在一电脑系统进入休眠时,将区分至少一存储器为多个清白存储页面与多个非清白存储页面,清白存储页面换出放弃储存,非清白存储页面写入休眠档储存于至少一第二储存装置的一切换空间中或一档案系统中。当电脑系统重新启动,电脑系统由切换空间与档案系统载回休眠档回复至存储器,并

且可从第二储存装置读取资料,载回至存储器。如此将缩减存储器转换存放至第二储存装置的资料量,加速电脑系统重新启动恢复至全速工作的速率。

[0010] 较佳的是,在该电脑系统进入休眠的步骤中,该清白存储页面系换出完成后,将释放该存储器。

[0011] 较佳的是,在该电脑系统进入休眠的步骤中,将以一核心程序配置该存储器,控制该清白存储页面系换出,且于该清白存储页面系换出完成后,控制释放该存储器。

[0012] 较佳的是,该核心程序可计算需换出的该清白存储页面的数量,并依据该数量配置该存储器。

[0013] 较佳的是,该第二储存装置可为一快闪存储器。

[0014] 较佳的是,该第二储存装置可为高速随机存取的存储器装置。

[0015] 较佳的是,在该电脑系统进入休眠的步骤中,该电脑系统能够作为区分该存储器的单位。

[0016] 较佳的是,该第二储存装置具有与该清白存储页面相一致的备份存放。

[0017] 较佳的是,在该电脑系统进入休眠的步骤中,欲换出的该非清白存储页面将依据位址排序,且将位址连续的该非清白存储页面合并成一单一写入命令,组合写入 (Writing combing) 该第二储存装置。

[0018] 底下通过具体实施例配合所附的图式详加说明,当更容易了解本发明的目的、技术内容、特点及其所达成的功效。

附图说明

[0019] 图 1 为本发明的架构示意图;

[0020] 图 2 为本发明的处理步骤流程图。

[0021] 附图标记说明:10- 电脑系统;12- 中央处理器;14- 主存储器;16- 第二储存装置;18- 档案系统;20- 切换空间;22- 休眠档。

具体实施方式

[0022] 本发明提出一种快速开关机的方法,减少电脑系统进入休眠时存储器装置的使用数量,降低写入第二储存装置资料量,以增进电脑系统进入休眠或重新启动恢复至全速工作的速度,底下则将以较佳实施例详述本发明的技术特征。如图 1 所示,一电脑系统 10 包含一中央处理器 (CPU) 12、一主存储器 (Main Memory) 14 与一第二储存装置 (Secondary Storage) 16。第二储存装置 16 可为高速随机存取存储器装置,如一快闪存储器,主要用于储存档案系统 (file system) 18、切换空间 (swap-space) 20 及休眠档 (hibernationfile) 22。

[0023] 图 2 所示为本发明的快速开关机的方法流程图,请同时参考图 1 的架构示意图。首先,如步骤 S30,当电脑系统 10 的操作系统 (Operating System, OS) 进入休眠模式,将以区或页为单位,区分主存储器 14 为清白存储页面与非清白存储页面,由于第二储存装置 16 存放具有与清白存储页面相一致的备份,故换出的清白存储页面将放弃储存,将通过一核心程序配置主存储器 14,控制该清白存储页面换出,核心程序可计算依据需换出的清白存储页面的数量配置主存储器 14,并在该清白存储页面换出完成后,控制释放该存储器。为核心

状态 (machine status) 的非清白存储页面将储存至休眠档 22, 并存放于第二储存装置 16 的切换空间 20 中或档案系统 18 中。其中非清白存储页面可依照位址排序, 以合并连续位址的非清白存储页面成一单一写入命令。之后, 如步骤 S32, 当电脑系统 10 重新启动, 电脑系统 10 从切换空间 20 与档案系统 18 读取休眠档 22, 回复储存休眠档至主存储器 14, 恢复系统状态。最后, 如步骤 S34, 电脑系统 10 将从第二储存装置 16 读取资料载回至主存储器 14。

[0024] 当执行中止至为快闪存储器的第二储存装置 16 快闪存储器暂停 (Suspend-to-flash) 的速度主要取决于需要写入到非挥发存储器 (nonvolatile memory) 的资料量的多寡。进入休眠之前换出 (Swapping-before-hibernating) 利用操作操作系统 (Operating System, OS) 中现有的存储器管理技术, 将大部分的存储器页 (memory page) 直接舍弃而不做任何的写入动作, 因此可以加速中止 (suspend) 的速度。回复 (resume) 的时候资料 (data) 将从三个地方取回, 分别是休眠档 (hibernation file) 22、切换空间 (swap-space) 20 及档案系统 (file system) 18, 系统隶属于核心 (kernel) 部分的主要资料 (data) 及编码 (code), 在重新开机时立刻取回, 而剩余的资料则使用需求页面调度 (paging-on-demand) 的方式取回使用者现阶段所需的部分。由于使用者所需要的资料量通常远小于系统中止前存储器中的所有资料量, 因此可以加速回复 (resume) 的速度。

[0025] 存储页面 (memory page) 可分成三种, 分别为自由页面 (free page)、匿名页面 (anonymous page) 及指定页面 (named page)。自由页面 (free page) 是系统当下未使用到的存储器, 自由页面 (free page) 中的内容对系统而言毫无意义。匿名页面 (anonymous page) 是程式在执行时期动态分配的存储器, 这部分的存储器主要包含了堆迭 (stack) 和堆积 (heap) 二种。指定页面 (named page) 则是档案在主存储器 14 中的副本。形式上它的行为类似于第二储存装置 16 的快取 (cache)。指定页面 (named page) 主要包含了执行档 (executable file)、动态连结库 (dynamic-link library) 在存储器中的对应, 或者程式使用存储器对应至档案 (memory-mapped-file) 将一个档案映射到存储器。

[0026] 匿名页面 (anonymous page) 及指定页面 (named page) 都可能是清白存储页面 (clean page) 或非清白存储页面 (dirty page)。非清白匿名页面 (dirty anonymous page) 被换出 (swap-out) 时必须写入到切换空间 (swap-space) 20。而非清白指定页面 (dirty named page) 则被换出 (swap-out) 到档案系统 (filesystem) 18。当操作系统 (OS) 需要换出 (swap-out) 一个清白存储页面时, 因为此页面在第二储存装置 16 中必定有一个内容完全一致的副本, 因此操作系统 (OS) 可以直接舍弃此页面。

[0027] 在系统开始执行快闪记忆体暂停 (suspend-to-flash) 时, 大部分的页面不需要做任何的写回第二储存装置 16 的动作, 这是由于大部分的页面都是清白存储页面。部分的页面需要写回到切换空间 20 或者档案系统 18 中。剩下的页面是非换出存储器 (non-swappable memory), 这些页面主要是操作系统 (OS) 的核心 (kernel), 或者程式为了效能的因素, 将某些页面设定为非换出 (non-swappable)。非换出存储器最后将被写入到休眠档 (hibernation file) 22。

[0028] 重新开机时, 系统在做完例行性硬体初始化动作及载入操作系统装填器 (OS loader) 后, 系统会判断是否要从休眠 (hibernation) 的状态回复或执行一般的开机动作。休眠档 (hibernation file) 22 中的资料将首先载入到主存储器 14 中, 完

成了这个动作,操作系统(OS)即完成基本的开机动作。随后依照使用者的需要,从档案系统 18 及切换空间 20 中将之前的系统状态载回。若再次执行进入休眠之前换出 (swapping-before-hibernating),由于部分非清白存储页面在上一次执行进入休眠之前换出 (swapping-before-hibernating) 时已经写入切换空间 20,因此这个页面就变成了清白存储页面,并且之后中央处理器 (CPU) 12 不再对此页面作写入的动作,故这个页面不需要再次的写入切换空间 20。因此在第二次以后的进入休眠之前换出 (swapping-before-hibernating) 执行速度将更加快速。

[0029] 以快闪存储器 (flash) 作为第二储存装置 16 通过最佳化技术,可在随机存取 (random access) 上有近似于连续存取 (sequential access) 的表现。通过组合写入 (Write-combining) 主要将存储器页在换出时所产生的小的随机写入 (small random write),在实际写入快闪存储器前先排列 (queue) 起来,首先针对这些要写出页面的实体位址 (physical address) 做排序,排序后便可知道这些要被写出的页面中是否存在着连续范围的写出,之后便可以一个较大的写入需要 (write request) 取代数个较小的写入。以此提升进入休眠之前换出 (swapping-before-hibernating) 的速度。

[0030] 将换出页面 (swappable pages) 写回第二储存装置 16 可经向操作系统 (OS) 要求大量的存储器,为了避免操作系统 (OS) 在分配存储器时的最佳化动作影响到实际配置存储器的数量。每配置到一个页面随即会在此页面写入一个字节 (byte) 的资料,以强制 OS 须立即配置实体记忆体 (physical memory) 给该程序 (program)。由此强迫操作系统 (OS) 将大部分的存储器换出。随后执行一连串的释放 (free) 动作,将之前向系统要求的存储器全数归还给操作系统 (OS)。经过这二个步骤以后,操作系统 (OS) 中大部分的页面都会变成自由页面 (free page),而自由页面 (free page) 不需要写回到休眠档 (hibernation file) 22。

[0031] 此外,可直接在操作系统 (OS) 的核心 (kernel) 中要求配置存储器,将通过核心程序 (kernel function) 立即配置所需的页面,将不需要额外的写入以强制操作系统 (OS) 真实的配发存储器。在配置完存储器后立即执行释放的动作,系统将产生大量的自由页面 (free page)。由于通过核心程序 (kernel function) 配置存储器不需要额外的写入动作。因此可以事先计算换出页面的数量,以此推算出需要配置多少存储器。

[0032] 另外,可直接操控操作系统 (OS) 中的换出器 (swapper) 机制。以 Linux 为例,在核心 (kernel) 中具有一存储器管理程序,其为 shrink_all_memory 程序 (一种核心函式,可以将多数非立即需求且不影响系统稳定的存储器资料丢弃),此程序将回收页面存储页面,呼叫其时传入欲回收的页面数量,其能够在不影响系统稳定、使用者使用程式的情况下释放出页面存储页面。shrink_all_memory 程序的机制为在核心 (kernel) 中有两近来最少使用目录 (least-recently-used list, LRU list) 分别为活动目录 (active list) 及闲置目录 (inactive list),活动目录 (active list) 包含了最近被存取过的页面,闲置目录 (inactive list) 则为有段时间没有被存取的页面。shrink_all_memory 程序会先从闲置目录 (inactive list) 开始回收,接着再回收活动目录 (active list)。利用 shrink_all_memory 程序,可以在不影响到系统核心的运作、使用者的程式下,尽可能的换出。

[0033] 在 Linux 的核心 (kernel) 中,第二储存装置 16 会被视为一般的区块输入 / 输出装置 (I/O device),而在我们将页面存储页面换出到切换空间 20 时,将呼叫 submit_bio

程序（函式的一种，主要提交 bio 给区块设备层进行 I/O 操作）其所接受的参数包含了要写入的区段数量 (sector number)、读取 / 写入指令 (r/w command)、存储器位址 (memory address)、存储器长度 (memorylength) 等资讯。可以通过拦截 submit_bio 程序对即将被换出的页面再进行处理。

[0034] 上述的即将被换出的页面是由 shrink_all_memory 程序所决定，决定的依据为此页面的使用情况而非此页面在主存储器 14 中的实体位址。我们将 shrink_all_memory 程序所产生的写入需求先排列。再依照此页面的实体存储器位址进行排序，并进行合并，以减少存取输入 / 输出 (I/O) 的次数并增加此输入 / 输出 (I/O) 的需求大小 (request size)。需求大小 (request size) 越大，将使得为快闪存储器 (flash) 的第二储存装置 16 的写入效能越好。

[0035] 当一个页面存储页面被换出时，核心 (kernel) 需要纪录此页面在切换空间 20 中所摆放的位置。使核心 (kernel) 能够重新将此页面从第二储存装置 20 载入。由于组合写入 (write-combining) 将重新改写 shrink_all_memory 程序，当每次执行完 shrink_all_memory 程序后，必须于确定了每个页面写入至切换空间 20 的位置后，再将资讯填入到页面表格 (page table) 中的换出页面鉴别符 (swapped-out page identifier)。

[0036] 以上所述的实施例仅为说明本发明的技术思想及特点，其目的在使本领域技术人员能够了解本发明的内容并据以实施，当不能以的限定本发明的专利范围，即大凡依本发明所揭示的精神所作的均等变化或修饰，仍应涵盖在本发明的专利范围内。

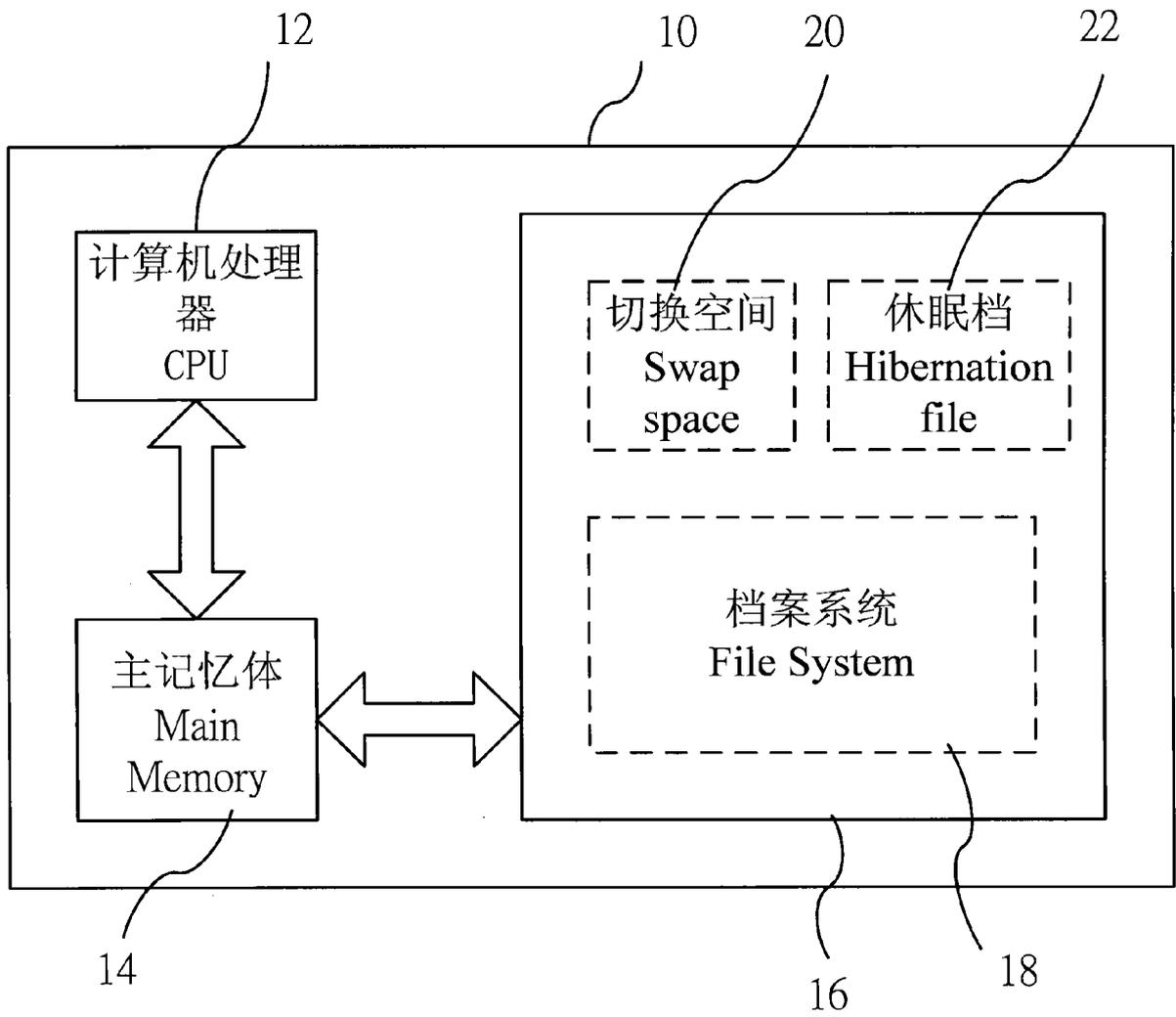


图 1

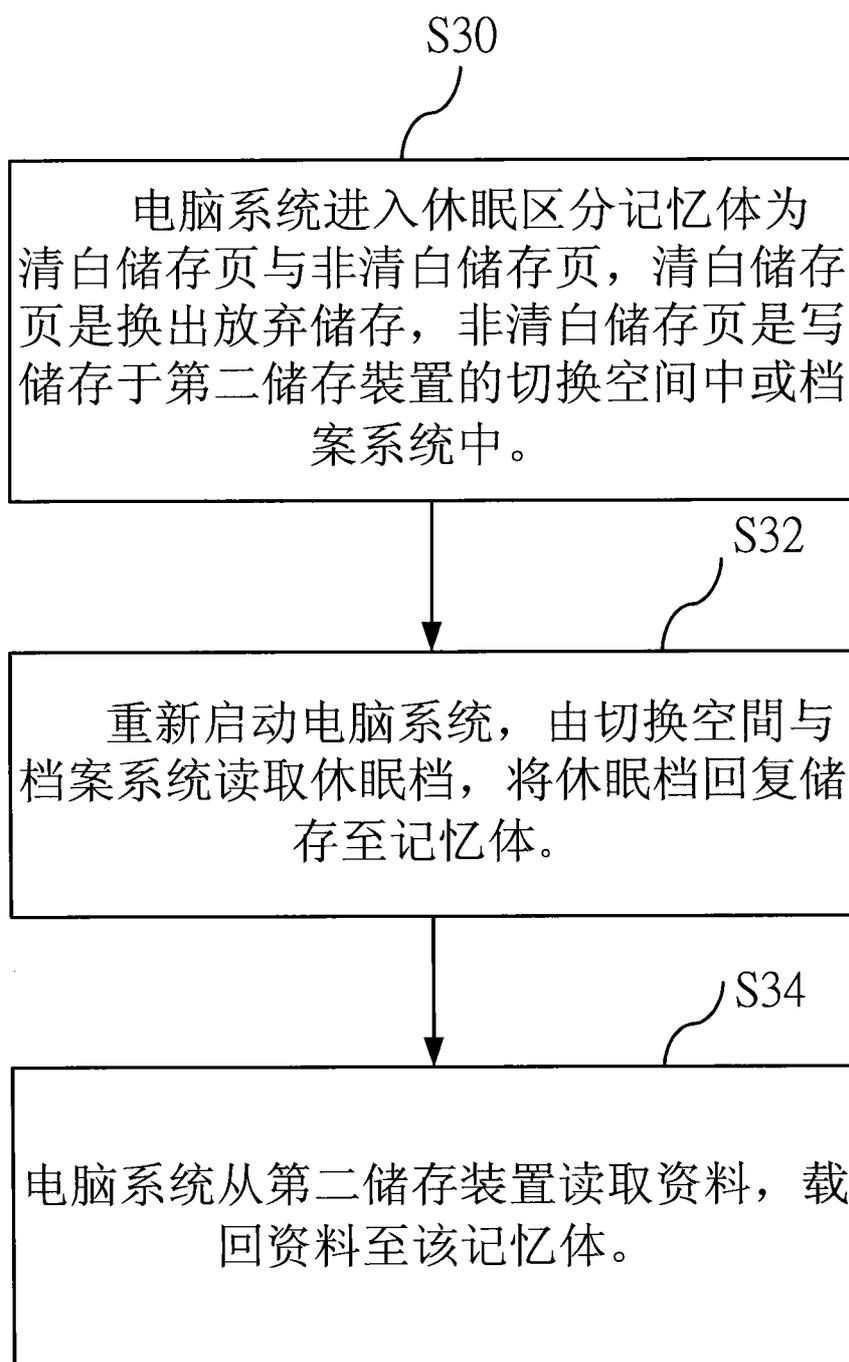


图 2