



**ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ**

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: 2010112956/08, 03.10.2008

(24) Дата начала отсчета срока действия патента:
03.10.2008

Приоритет(ы):

(30) Конвенционный приоритет:
05.10.2007 US 60/978,086

(43) Дата публикации заявки: 10.10.2011 Бюл. № 28

(45) Опубликовано: 10.04.2013 Бюл. № 10

(56) Список документов, цитированных в отчете о поиске: US 6226728 B1, 01.05.2001. US 2001/0032293 A1, 18.10.2001. US 5675790 A, 07.10.1997. US 6401166 B2, 04.06.2002. EP 0880139 A2, 25.11.1998. RU 2280273 C2, 20.07.2006.

(85) Дата начала рассмотрения заявки РСТ на национальной фазе: 05.04.2010

(86) Заявка РСТ:
US 2008/078823 (03.10.2008)

(87) Публикация заявки РСТ:
WO 2009/046353 (09.04.2009)

Адрес для переписки:

119019, Москва, Гоголевский бульвар, 11,
Московское представительство фирмы
"Гоулингз Интернэшнл Инк.", В.А.Клюкину

(72) Автор(ы):

**ТОМАС Бэсил (US),
ДЖЕНСЕН Крейг (US),
СТАФФЕР Эндрю (US),
РАМАНКУТТИ Сантош (US)**

(73) Патентообладатель(и):

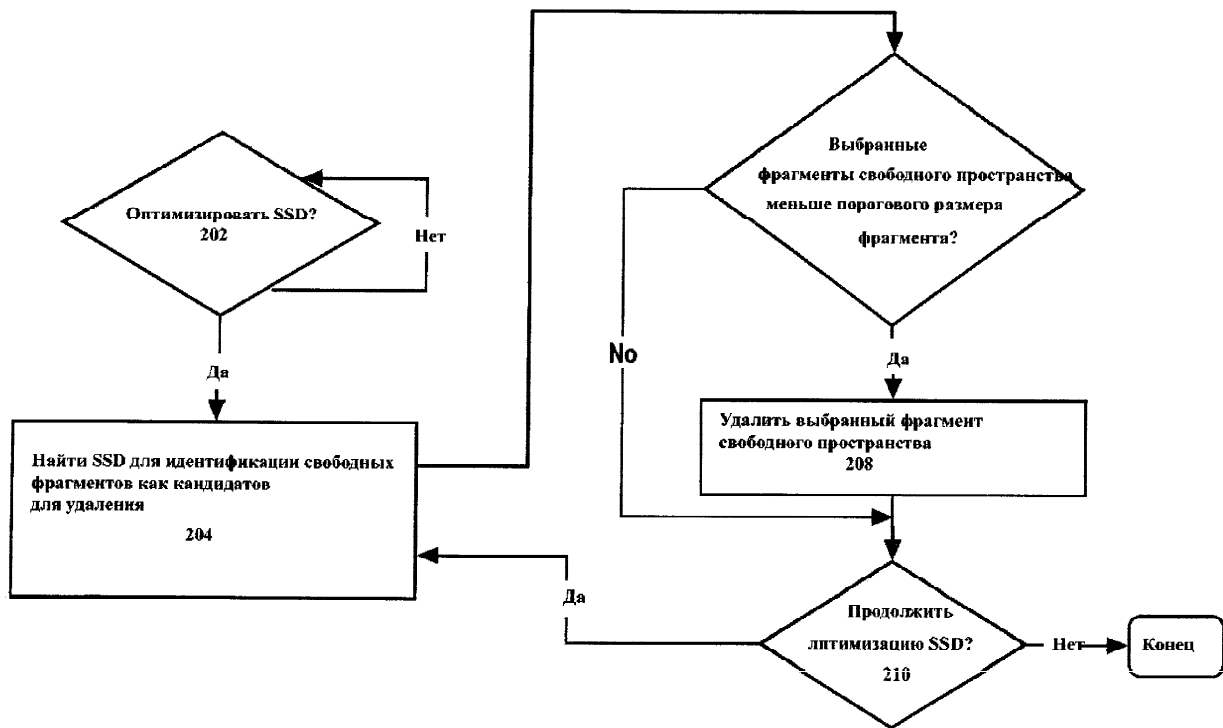
**КОНДУСИВ ТЕКНОЛОДЖИЗ
КОРПОРЕЙШН (US)**

(54) СПОСОБ ОПТИМИЗАЦИИ ТВЕРДОТЕЛЬНОГО НАКОПИТЕЛЯ И ОПТИМИЗАТОР

(57) Реферат:

Изобретение относится к вычислительной технике. Технический результат заключается в оптимизации твердотельного накопителя (SSD) путем обеспечения высокой скорости записи за счет сокращения числа операций записи. Способ оптимизации твердотельного накопителя, содержащий следующие стадии: получение порогового размера фрагмента, в котором пороговый размер фрагмента вычисляется на основании, по меньшей мере, одной или нескольких скоростей последовательной записи твердотельного

накопителя или скорости ввода-вывода (I/O) в/из SSD в секунду; определение, является ли фрагмент свободного пространства на диске SSD меньше порогового размера фрагмента; если фрагмент свободного пространства на диске SSD меньше порогового размера фрагмента, удаление фрагмента свободного пространства; и если фрагмент свободного пространства на диске SSD не меньше порогового размера фрагмента, фрагмент свободного пространства используется для хранения данных. 3 н. и 14 з.п. ф-лы, 3 ил.



Фиг. 2



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(51) Int. Cl.

G06F 12/02 (2006.01)*G06F 3/06* (2006.01)**(12) ABSTRACT OF INVENTION**(21)(22) Application: **2010112956/08, 03.10.2008**(24) Effective date for property rights:
03.10.2008

Priority:

(30) Convention priority:
05.10.2007 US 60/978,086(43) Application published: **10.10.2011 Bull. 28**(45) Date of publication: **10.04.2013 Bull. 10**(85) Commencement of national phase: **05.04.2010**(86) PCT application:
US 2008/078823 (03.10.2008)(87) PCT publication:
WO 2009/046353 (09.04.2009)

Mail address:

**119019, Moskva, Gogolevskij bul'var, 11,
Moskovskoe predstavitel'stvo firmy "Goulingz
Internehsnl Ink.", V.A.Kljukinu**

(72) Inventor(s):

**TOMAS Behsil (US),
DZhENSEN Krejg (US),
STAFFER Ehndrju (US),
RAMANKUTTI Santosh (US)**

(73) Proprietor(s):

**KONDUSIV TEKNOLODZhIZ KORPOREhJShN
(US)**

(54) METHOD OF OPTIMISING SOLID-STATE DRIVE AND OPTIMISER

(57) Abstract:

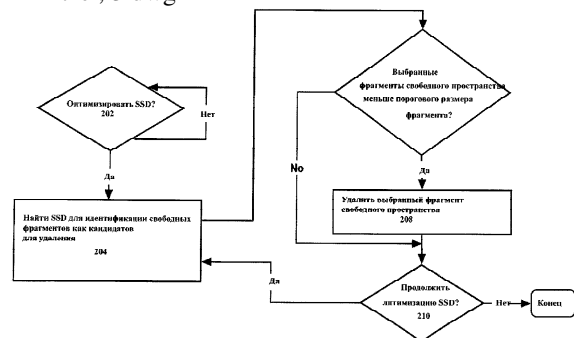
FIELD: information technology.

SUBSTANCE: method of optimising a solid-state drive, comprising the following steps: obtaining a threshold fragment size, wherein the threshold fragment size is calculated based on at least one or more serial recording speeds of the solid-state drive (SSD) or input/output (I/O) speeds into/from the SSD per second; determining whether a free space fragment on the SSD is smaller than the threshold fragment size; if the free space fragment on the SSD is smaller than the threshold fragment size, eliminating the free space fragment; and if the free space fragment on the SSD is not smaller than the threshold fragment size, the free space fragment is

used for storing data.

EFFECT: optimising a solid-state drive by providing a high recording speed by cutting the number of recording operations.

17 cl, 3 dwg



Фиг. 2

ПРИТЯЗАНИЕ НА ПРИОРИТЕТ

По настоящей заявке истребован приоритет по предварительной патентной заявке США номер 60/978,086, зарегистрированной 5 октября 2007 года, и по предварительной патентной заявке США номер 12/244,771, зарегистрированной 3 октября 2008.

ВКЛЮЧЕНИЕ ССЫЛОК

Настоящая заявка объединяется ссылкой на патентную заявку США номер 11/546,072, поданную 10 октября 2006 года, патентную заявку США номер 11/546,514, поданную 10 октября 2006 года, и патентную заявку США номер 11/471,466, поданную 19 июня 2006 года.

КРАТКОЕ СОДЕРЖАНИЕ ИЗОБРЕТЕНИЯ

Настоящее изобретение в целом относится к твердотельным накопителям. Более конкретно, изобретение относится к оптимизации твердотельных накопителей.

ПРЕДПОСЫЛКИ СОЗДАНИЯ ИЗОБРЕТЕНИЯ

Подходы, описанные в этом разделе, являются подходами, которых следовало добиваться, но они не обязательно должны быть достигнуты, как ранее задуманные или преследовались. Следовательно, если не обозначено иначе, нельзя считать, что любой из подходов, описанных в этом разделе, готовился как предшествующая технология просто на основании их включения в этот раздел.

Твердотельный накопитель (SSD), который также упоминается здесь как твердотельный диск, является запоминающим устройством, которое хранит данные, используя энергозависимую или энергонезависимую твердотельную память.

Твердотельная память состоит из электронных компонентов, которые состоят из полупроводников. Твердотельные накопители могут содержать различные типы элементов памяти, включая, без ограничения, статическую оперативную память (SRAM), динамическую оперативную память (DRAM), запоминающее устройство с произвольной выборкой PRAM, флэш-память (например, флэш-память НЕ-ИЛИ и флэш-память И-НЕ), или Фазовращательную память (PCM).

Поскольку накопители SSD не имеют никаких движущихся частей, они значительно снижают риск механических повреждений. Меньшее количество механических отказов приводит к повышению надежности системы. Кроме того, благодаря отсутствию движущихся частей накопители SSD обеспечивают меньшее время чтения и записи по сравнению со стандартными жесткими дисками со средней скоростью вращения. Соответственно, накопители SSD обеспечивают более высокую скорость считывания и записи последовательных и случайных данных.

Хотя накопитель SSD не имеет никаких подвижных частей, тем не менее возникают проблемы снижения быстродействия в результате определенных ограничивающих факторов и возможной фрагментации свободного пространства. Например, одним из ограничений по быстродействию может быть ограниченное число операций записи, которые SSD может выполнить в любом данном интервале времени. Когда SSD записывает данные во множество небольших фрагментов свободного пространства, файл должен быть фрагментирован в ряд различных участков памяти на SSD. При записи каждого фрагмента в свободное пространство могут потребоваться операции стирания и операции записи, и число операций записи, которые может выполнить накопитель SSD, может быть ограничено, причем множество небольших фрагментов свободного пространства может привести к невозможности максимального использования скорости последовательной записи, что приводит к ограничению числа операций записи на диск SSD.

При нормальном использовании компьютера современные операционные системы, такие как Windows (Windows - торговый знак изготовителя - Корпорации Майкрософт, Сигл, Вашингтона), создают множество фрагментов свободного пространства.

Например, просмотр Интернета потенциально может создать множество
5
нежелательных фрагментов свободного пространства. Web-браузер при его использовании создает много временных файлов. Большинство временных файлов являются небольшими по объему, имеют краткосрочное время существования и часто удаляются, например, во время очистки буферной памяти браузера и т.д. Однако все
10
временные файлы не удаляются одновременно или в порядке, в котором они были созданы. Это явление ускоряет создание фрагментов свободного пространства.

Кроме того, когда вновь созданный файл хранится на диске, файловые системы вообще начинают писать вновь созданный файл либо (a) в последний освобожденный
15
фрагмент свободного пространства, либо (b) в первоначально идентифицированный фрагмент свободного пространства, не рассматривая размер фрагмента свободного пространства. Если записываемый файл является большим, а идентифицированные фрагменты свободного пространства являются малыми, файл разбивается на множество фрагментов, соответствующих идентифицированным фрагментам
20
свободного пространства, и множество операций записи обеспечивают хранение файла на диске. В дополнение к операциям записи, требуемым для хранения фактического файла, требуется множество операций записи для обновления структуры файловой системы, когда файл хранится во множестве фрагментов.

КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

Настоящее изобретение описывается на не ограничивающем примере его реализации со ссылками на приложенные чертежи, в которых цифровые позиции обозначают одинаковые элементы и в которых:

Фигура 1 - блок-схема, иллюстрирующая твердотельный накопитель и оптимизатор
30
твердотельного накопителя в соответствии с примером воплощения изобретения.

Фигура 2 - технологическая схема, иллюстрирующая пример воплощения для оптимизации твердотельного накопителя.

Фигура 3 - блок-схема, иллюстрирующая компьютерную систему, которая может использоваться при осуществлении примера воплощения настоящего изобретения.

ПОДРОБНОЕ ОПИСАНИЕ

В последующем описании приведены многочисленные конкретные детали, чтобы обеспечить полное понимание настоящего изобретения. Однако специалистам понятно, что настоящее изобретение может быть осуществлено без этих конкретных
40
деталей. В других примерах известные структуры и устройства показаны в виде блок-схем, чтобы избежать излишнего затенения сущности изобретения.

Несколько признаков описаны таким образом, что каждый из них может использоваться независимо друг от друга или с любой комбинацией других признаков. Однако любой индивидуальный признак не может быть отнесен ни к одной из
45
проблем, обсужденных выше, или может быть отнесен только к одной из этих проблем. Некоторые из обсужденных выше проблем не могут быть полностью отнесены ни одному из описанных здесь признаков. Хотя предусмотрены заголовки, информация, связанная с конкретным заголовком и не найденная в разделе, имеющем этот заголовок, может быть найдена в другом месте описания.

ОБЗОР

Предлагается способ для оптимизации твердотельного накопителя (SSD). Запись файла на диск SSD может потребовать множества операций стирания и записи, если

записанные фрагменты свободного пространства являются слишком малыми для всего файла. Например, если каждый фрагмент свободного пространства на SSD, в который пишется файл, намного меньше размера файла, то файл может быть разбит на множество фрагментов и операция записи может требоваться для каждого

5 фрагмента свободного пространства, пока весь файл не будет записан на диск SSD. Соответственно, настоящее изобретение оптимизирует SSD, отделяя малые фрагменты свободного пространства, заполняя фрагменты свободного пространства данными или делая фрагменты свободного пространства большими с тем, чтобы это не влияло

10 на скорость записи. Это принуждает операционную систему записывать файлы последовательно или, по меньшей мере, в меньшем количестве фрагментов свободного пространства, что обеспечивает высокую скорость записи.

В соответствии с одним примером воплощения пороговый размер фрагмента используется как основа для решения, удалить ли фрагменты свободного

15 пространства или сохранить фрагменты свободного пространства в SSD для хранения данных. Размер фрагментов свободного пространства сравнивается с пороговым размером фрагмента. Если фрагменты свободного пространства меньше порогового размера фрагмента, фрагменты свободного пространства удаляются с тем, чтобы

20 диск SSD не использовал эти фрагменты свободного пространства для записи файла. Если фрагменты свободного пространства, по меньшей мере, равны пороговому размеру фрагмента, фрагменты свободного пространства сохраняются для хранения данных на диске SSD. Удаление фрагментов свободного пространства, имеющих

25 размер меньше пороговой величины, позволяет записывать файл в фрагменты с большим свободным пространством и, соответственно, требует меньшего числа фрагментов свободного пространства для записи файла и меньшего числа операций записи. Сокращая число операций записи за промежуток времени, меньший или

равный числу операций записи, которые могут фактически быть выполнены

30 диском SSD, улучшает его рабочие характеристики. Улучшение рабочих характеристик SSD не ограничивается числом операций записи, которые могут быть выполнены, начиная с числа операций записи, необходимых для оптимизированного диска SSD, меньше числа операций записи, которые могут быть выполнены. В

35 примере воплощения пороговый размер фрагмента определяется как основанный, по меньшей мере, на последовательной скорости записи в SSD и на числе операций ввода-вывода (I/O), осуществленных за установленный промежуток времени на диск SSD. Например, пороговый размер фрагмента может быть определен, по меньшей мере, частично, разделяя скорость последовательной записи на скорость ввода-вывода

40 в/из SSD в секунду.

В примере воплощения могут быть проверены несколько различных пороговых размеров фрагмента, используя при записи файлов пороговый размер фрагмента, который соответствует оптимальным рабочим характеристикам. Оптимизация порогового размера фрагмента может выполняться периодически либо на основании

45 predeterminedного условия, например при низком уровне скорости SSD, либо на predeterminedном числе фрагментов свободного пространства.

Хотя конкретные компоненты описаны здесь как стадии выполнения способа, в других примерах воплощения механизмы, действующие как части указанных

50 компонентов, могут выполнить все стадии способа. Далее, хотя изобретение обсуждается по отношению к компонентам единственной системы, изобретение может быть осуществлено с компонентами, расположенными во множестве систем. Кроме того, хотя изобретение обсуждается по отношению к твердотельному

накопителю (SSD), примеры воплощения изобретения могут быть применимыми к любому диску для хранения данных (например, оно может использоваться в дисководе с вращающимся диском).

Примеры воплощения изобретения также включают любую систему, которая содержит средства для выполнения стадий описанного здесь способа. Примеры воплощения изобретения также включают машинно-считываемую среду с командами, которые затем выполняются компьютером, таким образом, осуществляя стадии описываемого здесь способа.

УДАЛЕНИЕ ФРАГМЕНТОВ СВОБОДНОГО ПРОСТРАНСТВА

Фрагменты свободного пространства на уровне файловой системы являются блоками памяти или участками хранения, которые не обязательно являются хранилищем данных и доступны для распределения. Фрагменты свободного пространства могут быть созданы, когда объект, хранившийся в памяти, удален, например при удалении ссылки из пространства памяти. Поскольку объекты не могут быть удалены в том же самом порядке, в каком они были созданы, фрагменты свободного пространства, соответствующие удаленным объектам, создаются между блоками памяти, где хранятся данные. Запись в каждый фрагмент свободного пространства может потребовать операции стирания и последующей операции записи. Соответственно, если фрагменты свободного пространства являются слишком малыми, чтобы сохранить файл полностью, файл должен быть фрагментирован и записан во многие различные фрагменты свободного пространства. Чем меньше фрагменты свободного пространства, тем более высокая фрагментация необходима для записи файла. Поскольку число операций записи, которые диск может выполнить за определенный промежуток времени, ограничено, множество малых фрагментов свободного пространства могут потребовать высокого уровня фрагментации файла и предотвратить использование максимальной скорости записи диска. Соответственно, в примере воплощения изобретения фрагменты свободного пространства, которые меньше указанного порогового размера фрагмента, удаляются. Удаление этих фрагментов свободного пространства уменьшает количество фрагментов свободного пространства, которые записаны в файловой системе.

Удаление фрагмента свободного пространства может включать заполнение фрагмента свободного пространства, по меньшей мере, частью другого файла. Например, для заполнения фрагмента свободного пространства могут использоваться файлы к концу диска, редко используемые файлы или временные файлы наполнителя. Заполнение фрагмента свободного пространства при сохранении, по меньшей мере, части другого файла удаляет фрагменты свободного пространства, которые могут потенциально влиять на рабочие характеристики SSD. Удаление фрагмента свободного пространства также может включить объединение фрагмента свободного пространства с одним или несколькими смежными фрагментами свободного пространства так, чтобы объединенный фрагмент свободного пространства, по меньшей мере, был бы столь же большим, как и фрагмент порогового размера. В примере воплощения фрагмент свободного пространства удаляется, если ресурс, требуемый для удаления фрагмента свободного пространства, отвечает критерию бездействия. Например, если использование ресурса ниже предопределенного процента, критерий бездействия удовлетворен. В другом примере критерии бездействия могут включить частоту использования ресурса, где предопределенная низкая частота использования соответствует критериям бездействия. Более подробное описание критериев бездействия и ресурсов описано в патентной заявке США

номер 11/546,514, поданной 10 октября 2006 года, которая включена здесь в качестве ссылки.

АРХИТЕКТУРА СИСТЕМЫ

Хотя здесь описана определенная архитектура вычислительной системы, другие
5 примеры воплощения изобретения применимы к любой архитектуре, которая может использоваться для оптимизации твердотельного накопителя (SSD) на основе порогового размера фрагмента.

На фигуре 1 показаны накопитель SSD (100) и оптимизатор SSD (125) в
10 соответствии с одним или несколькими примерами воплощения изобретения. Как показано на фигуре 1, SSD (100) включает интерфейс (105), контроллер памяти (110) и твердотельную память (115).

ИНТЕРФЕЙС

Интерфейс (105), как правило, представляет собой любое соединение, которое
15 может использоваться для обмена информацией (например, сохраненными данными или восстановленными данными) с SSD (100). Например, интерфейс (105) может быть включен между SSD (100) и системной платой для передачи данных, Интерфейс (105) может нести слова информации параллельно или поразрядно. Примеры
20 интерфейса (105) включают, без ограничения, интерфейс АТА (например, последовательный интерфейс (SATA) и параллельный интерфейс (PATA)) или интерфейс со встроенным контроллером (IDE) или системный интерфейс персональных компьютеров (SCSI). Интерфейс (105) позволяет внешнему узлу прямо или косвенно обращаться к твердотельной памяти (115) по командам контроллера
25 памяти (110).

КОНТРОЛЛЕР ПАМЯТИ

В одном или нескольких примерах воплощения твердотельная память (115) в целом
представляет собой компонент хранения данных, который включает логику для
30 хранения цифровых данных. Твердотельная память (115) включает полупроводниковые устройства, которые содержат логику и аппаратные средства, чтобы сохранять цифровые данные. Например, твердотельная память (115) может сохранять биты в фазосдвигающей оперативной памяти, одноуровневых ячейках (SLC)
или многоуровневых ячейках (MLC). Описанный выше пороговый размер
35 фрагмента (120) может быть вычислен по другой методике на основании того, сохраняются ли биты в ячейках SLC или MLC в твердотельной памяти (115). Кроме того, пороговый размер фрагмента (120) также может быть вычислен на основе числа уровней в многоуровневых ячейках. Твердотельная память (115) может быть
40 осуществлена как статическая оперативная память (SRAM), динамическая оперативная память (DRAM), параметрическая оперативная память (PRAM), флэш-память (например, флэш-память НЕ-И, флэш-память И-НЕ), фазовращательная память (PCM) или другой тип подходящей памяти.

ОПТИМИЗАТОР ТВЕРДОТЕЛЬНОГО НАКОПИТЕЛЯ

В одном или нескольких примерах воплощения оптимизатор твердотельного
45 накопителя (125) в целом представляет собой программные и/или аппаратные средства, используемые для оптимизации SSD (100), чтобы улучшить рабочие характеристики диска SSD. Например, оптимизатор SSD (125) может соответствовать
50 приложению, выполняемому в устройстве, подключенном к SSD (100). В компьютерной системе оптимизатор твердотельного накопителя (125) может быть приложением, работающим одновременно с другими приложениями, которые хранят данные на диске SSD (100). В другом примере оптимизатор SSD (125) может

соответствовать устройству с логикой, внедренной, чтобы оптимизировать SSD (100). Хотя на фигуре 1 оптимизатор SSD (125) показан соединенным с SSD (100), оптимизатор SSD (125) может также соответствовать программному обеспечению и/или компоненту непосредственно на диске SSD (100), в соответствии с одним

5 примером воплощения изобретения. Оптимизатор SSD (125) включает логику для оптимизации SSD (100) путем удаления фрагментов свободного пространства, которые меньше порогового размера фрагмента (120), таким образом вынуждая операционную систему записывать данные или файлы последовательно, как описано выше. Оптимизатор SSD (125) может быть активизирован пользователем, может активизироваться периодически или может быть активизирован на основании predetermined условия (условий) (например, на predetermined уровне фрагментации SSD (100) или на уровне низких рабочих характеристик SSD (100)). Оптимизатор SSD (125) может также быть активизирован при низком использовании ресурса. Например, если ресурс (например, процессор) используемый оптимизатором SSD (125) при оптимизации SSD (100), имеет низкий текущий уровень использования, оптимизатор SSD (125) может быть активизирован. В одном примере воплощения могут потребоваться несколько условий одновременно для активизации оптимизатора SSD (125) (например, низкий уровень рабочих характеристик SSD (100) и низкий текущий уровень использования ресурса).

В одном примере воплощения оптимизатор SSD (125) может включать логику, чтобы планировать любые задания, выполняемые как компьютерные микрозадачи. Компьютерные микрозадачи описаны в патентной заявке США номер 11/471,466, 25 поданной 19 июня 2006 года, и патентной заявке США номер 11/546,072, поданной 10 октября 2006 года, которые включены здесь в качестве ссылки. Соответственно, задачи, поставленные оптимизатору SSD (125) для оптимизации диска SSD (100), могут выполняться в течение длительного времени как различные компьютерные микрозадачи.

ПОРОГОВЫЙ РАЗМЕР ФРАГМЕНТА

В одном или нескольких примерах воплощения пороговый размер фрагмента (120), показанный как сохраненный на оптимизаторе SSD (125), соответствует конкретному размеру фрагмента свободного пространства, используемому оптимизатором SSD (125) при выполнении вышеописанных задач. Иными словами, фрагменты свободного пространства, которые меньше порогового размера фрагмента (120), удаляются оптимизатором SSD (125), и фрагменты свободного пространства, которые больше чем или равны пороговому размеру фрагмента (120), выделяются для хранения данных оптимизатором SSD (125). Использование порогового размера фрагмента (120) при удалении фрагментов свободного пространства меньше порогового размера фрагмента (120) может вынудить файловую систему записать файл последовательно в единственный фрагмент свободного пространства или во множество фрагментов свободного пространства в пределах памяти записи 10 в секунду (IOPS) диска SSD, который не влияет на рабочие характеристики SSD, основанные на записи 10 в секунду SSD.

Хотя на чертеже показано сохранение на оптимизаторе SSD (125), пороговый размер фрагмента (120) может поддерживаться любым компонентом (например, оптимизатором SSD (125), SSD (100) или другим подходящим устройством). Пороговый размер фрагмента может поддерживаться приложением (например, оптимизатором SSD) как переменная или может быть передан в устройство по проводам. Например, пороговый размер фрагмента может быть аппаратным

решением на диске SSD (100), оптимизатором SSD (125), если оптимизатор SSD осуществлен как устройство, системой, реализующей оптимизатор SSD (125), если оптимизатор SSD осуществлен как приложение, выполняемое системой или другим подходящим устройством. Пороговый размер фрагмента (120) также может быть

получен из области конфигурации, которая может управляться пользователем (120). В одном примере воплощения пороговый размер фрагмента (120), вычисляется оптимизатором SSD (125), контроллером памяти (110), пользователем или другим подходящим объектом. В другом примере воплощения пороговый размер фрагмента (120) может быть вычислен на основе скорости последовательной записи или числа операций записи, которые SSD (100) может выполнить за определенный промежуток времени. Одна из стадий вычисления порогового размера фрагмента может включать деление последовательной скорости записи на число операций записи, которые SSD может выполнить в установленный период времени. Например, если скорость последовательной записи равна 40 МБ/с и число операций записи, которые могут быть выполнены SSD (100), составляет 10/с, то пороговый размер вычисленного фрагмента, полученный путем деления 40 на 10, равен 4 МБ. Соответственно, пороговый размер фрагмента может быть установлен как 4 МБ. Кроме того, вычисление порогового размера фрагмента может также включить дополнительные вычисления (например, умножение на константу 3 или 4, чтобы получить 12 МБ или 16 МБ) с учетом разницы между фактическими рабочими характеристиками и теоретическими рабочими характеристиками или скоростью.

Пороговый размер фрагмента (120) для SSD может быть определен динамически или может быть статическим значением, используемым оптимизатором SSD (125). Например, аппаратный пороговый размер фрагмента (120) может постоянно использоваться для оптимизации диска SSD (100). Альтернативно, каждый раз, когда диск SSD (100) оптимизируется периодически, пороговый размер фрагмента (120) может быть вычислен или получен повторно.

В одном примере воплощения могут использоваться различные пороговые размеры фрагмента, чтобы определить оптимальные рабочие характеристики. Например, изменение значений порогового размера фрагмента может использоваться для оптимизации SSD (100), удаляя фрагменты свободного пространства на диске SSD (100). После того как эксплуатационные качества могут быть определены за время, последующее за оптимизацией SSD (100). Пороговый размер фрагмента, который приводит к улучшению рабочих характеристик, является результатом оптимизации и может быть установлен как пороговый размер фрагмента (120).

В одном примере воплощения, когда установлен пороговый размер фрагмента (120), повторное испытание порогового размера фрагментов (120) может выполняться периодически. Во время повторного испытания значения выше и ниже порогового размера фрагмента (120) могут временно использоваться для оптимизации диска SSD (100). После использования более высокого и/или более низкого порогового размера фрагмента, полученные рабочие характеристики диска SSD (100) могут быть оценены, чтобы определить, должен ли пороговый размер фрагмента (120) быть увеличен, уменьшен или сохранен без изменения.

ОПТИМИЗАЦИЯ ТВЕРДОТЕЛЬНОГО НАКОПИТЕЛЯ

На фигуре 2 представлена блок-схема оптимизации твердотельного накопителя (SSD), используя пороговый размер фрагмента, в соответствии с одним или несколькими примерами воплощения изобретения. Описанные ниже стадии могут быть опущены, повторены и/или выполнены в различном порядке. Соответственно,

определенное расположение стадий, показанных на фигуре 2, не должно рассматриваться как ограничение объема изобретения.

В одном примере воплощения, описанном ниже, перед выполнением одной или нескольких стадий оптимизации твердотельного накопителя может быть проведена
5 проверка доступности ресурса. Проверка доступности ресурса может быть выполнена, чтобы гарантировать, что ресурсы для выполнения этой стадии доступны и соответствуют критериям бездействия, как описано выше. Кроме того, одна или несколько стадий, описанных ниже, могут быть спланированы как описанные выше
10 компьютерные микрозадачи.

Сначала принимается решение, оптимизировать ли SSD (стадия 202). Решение оптимизации SSD может быть основано на одном или нескольких условиях. Например, решение оптимизировать SSD может быть основано на текущих низких
15 рабочих характеристиках SSD, наличии предопределенного уровня фрагментации (например, имеется много фрагментов свободного пространства, число фрагментов свободного пространства меньше порогового размера фрагмента и т.д.), пригодности ресурса, учитывая планирование по времени (например, периодическая оптимизация), уровень фрагментации файла, или на основании любых других подходящих критериев.
20 Оптимизатор SSD также может дефрагментировать слишком фрагментированные файлы, что делает доступ к файлу более эффективным и надежным.

Затем SSD осуществляет идентификацию фрагментов свободного пространства как кандидатов на удаление (стадия 204). Поиск свободного пространства может
25 включить поиск файловой системы или API операционной системы или поиск адреса памяти, которого нет в файловой системе, или адресов, индексированных как местоположения блоков памяти, доступных для распределения. Поиск файловой системы может быть выполнен последовательно, в порядке освобождения памяти, или в любом другом подходящем порядке. В одном примере воплощения фрагменты
30 свободного пространства могут быть идентифицированы как кандидаты на удаление в порядке от наименьшего до наибольшего с тем, чтобы наименьшие фрагменты свободного пространства могли быть удалены первыми.

Затем определяется, является ли фрагмент свободного пространства, идентифицированный как кандидат на удаление, по размеру меньше, чем пороговый
35 размер фрагмента (стадия 206). Если идентифицированный фрагмент свободного пространства не меньше порогового размера фрагмента, то идентифицированный фрагмент свободного пространства подходит для записи файла и, соответственно, идентифицированный фрагмент свободного пространства сохраняется для записи.
40 Однако если фрагмент свободного пространства меньше порогового размера фрагмента, то идентифицированный фрагмент свободного пространства не подходит для хранения данных и, соответственно, удаляется, как описано выше (стадия 208). Идентифицированный фрагмент свободного пространства также может быть удален,
45 объединяя фрагмент свободного пространства с одним или несколькими смежными фрагментами свободного пространства, чтобы создать объединенный фрагмент свободного пространства, где такой фрагмент свободного пространства является, по меньшей мере, столь же большим, что и пороговый размер фрагмента.

Затем определяется, следует ли продолжить оптимизацию SSD (стадия 210). Может
50 быть принято решение продолжить оптимизацию до тех пор, пока все SSD не были проверены на наличие фрагментов свободного пространства меньше, чем пороговый размер фрагмента. В другом примере выполнения SSD может подвергаться оптимизации до тех пор, пока не будут свободно доступны все ресурсы (например, в

течение непиковых часов). SSD может также быть оптимизирован по таймеру, где блок времени настроен на оптимизацию SSD.

Хотя вышеупомянутые стадии описаны в последовательном порядке относительно удаления каждого фрагмента свободного пространства, который меньше порогового размера фрагмента, примеры воплощения изобретения включают любой подходящий способ выполнения описанных выше задач. Например, все фрагменты свободного пространства могут быть идентифицированы сначала как кандидаты на удаление. После этого каждый из идентифицированных фрагментов свободного пространства может сравниваться с пороговым размером фрагмента и удаляться, если он меньше порогового размера фрагмента.

КРАТКОЕ ОПИСАНИЕ АППАРАТНЫХ СРЕДСТВ

На фигуре 3 представлена блок-схема, иллюстрирующая компьютерную систему 300, которая может быть осуществлена как вариант воплощения изобретения. Компьютерная система 300 включает шину 302 или другой механизм связи для передачи информации и процессор 304 вместе с шиной 302 для обработки информации. Компьютерная система 300 также включает оперативную память 306, например память с произвольным доступом (ОЗУ) или другое устройство динамической памяти, соединенное с шиной 302 и используемое для хранения информации и команд, которые будут выполняться процессором 304. Оперативная память 306 также может использоваться для того, чтобы сохранить временные переменные или другую промежуточную информацию во время выполнения команд, которые будут выполняться процессором 304. Компьютерная система 300 далее включает постоянное запоминающее устройство (ROM) 308 или другое статическое запоминающее устройство, соединенное с шиной 302 для хранения статической информации и команд для процессора 304. Запоминающее устройство 310, например магнитный или оптический диск, соединено с шиной 302 и служит для хранения информации и команд.

Компьютерная система 300 может быть соединена через шину 302 с дисплеем 312 типа электронно-лучевой трубки (CRT) для отображения информации пользователю компьютера. Устройство ввода данных 314, включая алфавитно-цифровую клавиатуру, соединено с шиной 302 для передачи информации и набора команд на процессор 304. Другой тип пользовательского устройства ввода данных представляет собой устройство управления курсором 316, например мышь, шаровой указатель или управляющие клавиши курсора, используемые для передачи информации направления и команд на процессор 304 и для направления движения курсора 312. Это устройство ввода данных обычно имеет две степени свободы в двух осях, таких как первая ось (например, x) и вторая ось (например, y), которые позволяют устройству определять положения на плоскости.

Изобретение связано с использованием компьютерной системы 300 для осуществления описанного здесь способа. Согласно одному примеру воплощения изобретения этот способ выполняется компьютерной системой 300 в ответ на действия процессора 304, выполняя одну или несколько последовательностей одной или нескольких команд, содержащихся в оперативной памяти 306. Такие команды могут считываться в оперативную память 306 от другой машинно-читаемой передающей среды, такой как запоминающее устройство 310. Выполнение последовательностей команд содержащихся в оперативной памяти 306 процессоров причин 304 необходимо для выполнения описанных здесь стадий технологического процесса. В альтернативных примерах воплощения аппаратные средства могут использоваться вместо программных средств или в сочетании с ними, чтобы осуществить настоящее

изобретение. Таким образом, примеры воплощения изобретения не ограничены никакой определенной комбинацией схем аппаратных средств и программного обеспечения.

5 Термин "машинно-читаемая среда", как он используется здесь, относится к любой среде, предназначенной для передачи данных, которые обрабатываются компьютером. В одном примере воплощения, осуществляемом, используя компьютерную систему 300, используются различные машинно-читаемые носители, например, для передачи команд на процессор 304 с целью их выполнения. Такая среда 10 может иметь различные формы, включая, без ограничения, среду для хранения и среду для передачи данных. Носители данных включают как энергонезависимые носители, так и энергозависимые носители. Энергонезависимые носители включают, например, оптические или магнитные диски, такие как запоминающее устройство 310. Энергозависимые носители включают динамическую память типа оперативной 15 памяти 306. Передающая среда включает коаксиальные кабели, медную проводную и волоконную оптику, включая провода, содержащие шину 302. Передающая среда может также принять форму слухового аппарата или световых волн, например радиоволны и инфракрасное излучение, используемое для передачи данных. Все такие 20 носители должны быть физическими элементами, способными передавать команды, носимые этими средами, которые будут обнаружены физическим механизмом, который считывает эти команды и передает их компьютеру.

Стандартные формы машинно-читаемых носителей включают, например, гибкий диск, жесткий диск, магнитную ленту или любую другую магнитную среду, CD-ROM, 25 любую другую оптическую среду, перфокарты, перфоленты или другую физическую среду с перфорацией, оперативную память, ППЗУ и программируемую ПЗУ, любые другие микросхемы памяти или картриджи, несущую частоту, как описано ниже, или любую другую среду, которая может быть считана компьютером.

30 Различные формы машинно-читаемых носителей могут быть использованы для передачи одной или нескольких последовательностей одной или нескольких команд на процессор 304 для их выполнения. Например, команды первоначально могут быть записаны на магнитном диске удаленного компьютера. Удаленный компьютер может загрузить команды в свою динамическую память и послать команды по телефонной 35 линии, используя модем. Модем, встроенный в местный компьютер 300, может получить данные по телефонной линии и использовать инфракрасный передатчик, чтобы преобразовать данные в инфракрасный сигнал. Детектор инфракрасных сигналов может получать данные, которые заложены в инфракрасном сигнале, и 40 использовать схему передачи данных по шине 302. Шина 302 переносит данные в оперативную память 306, из которой процессор 304 извлекает и выполняет команды. Команды, полученные оперативной памятью 306, могут быть произвольно сохранены в запоминающем устройстве 310 до или после их выполнения процессором 304.

45 Компьютерная система 300 также включает интерфейс связи 318, соединенный с шиной 302. Интерфейс связи 318 обеспечивает дуплексную связь при передаче данных по сети 320, которая соединена с местной сетью 322. Например, интерфейс связи 318 может быть цифровой сетью с предоставлением комплексных услуг (цифровая сеть комплексного обслуживания), платой или модемом, чтобы обеспечить передачу 50 данных по телефонной линии соответствующего типа. В качестве другого примера, интерфейс связи 318 может быть карточкой местной сети (LAN), которая обеспечивает соединение при передаче данных через совместимую локальную сеть. Могут быть также беспроводные линии связи. В любой такой практической реализации интерфейс

связи 318 посылает и получает электрические, электромагнитные или оптические сигналы, которые несут цифровые потоки данных, представляющие различные типы информации.

Сетевая система 320 обычно обеспечивает передачу данных через одну или несколько сетей на другие устройства обработки данных. Например, сеть связи 320 может обеспечить соединение через местную сеть 322 с главным компьютером 324 или с оборудованием обработки данных, эксплуатируемое поставщиком услуг Интернета (провайдером) 326. Провайдер 326, в свою очередь, оказывает услуги по передаче данных через всемирную сеть передачи пакетов данных, теперь обычно называемую "Интернетом" 328. Местная сеть 322 и Интернет 328 используют электрические, электромагнитные или оптические сигналы, которые несут цифровые потоки данных. Сигналы через различные сети и сигналы сети 320, передаваемые с помощью интерфейса 318 на компьютерную систему 300, являются примером несущих, транспортирующих информацию.

Компьютерная система 300 может посылать сообщения и получить данные, включая код программы, через сеть (сети) связи 320, и эта связь осуществляется с помощью интерфейса 318. В Интернет-примере сервер 330 может передать требуемый код для прикладной программы через Интернет 328, поставщика услуг 326, местную сеть 322 и интерфейс связи 318.

Полученный код может быть выполнен процессором 304 по его получении через сеть и/или из запоминающего устройства 310 или из другой энергонезависимой памяти для более позднего выполнения. Этим способом компьютерная система 300 может получить код программы в виде несущей.

МОДИФИКАЦИИ И АЛЬТЕРНАТИВЫ

Вышеприведенные варианты воплощения изобретения были описаны на примере различных конкретных деталей, которые могут изменяться от реализации до реализации. Таким образом, единственный и исключительный индикатор того, что является изобретением и заявлено заявителями как изобретение, является перечень пунктов патентования, которые исходят из данной заявки, в конкретной форме, в которой такие пункты сформулированы, включая любое последующее изменение. Любые определения, явно сформулированные здесь для терминов, содержащихся в таких пунктах, должны управлять значениями таких терминов, как использующихся в этих пунктах. Следовательно, никакое ограничение, элемент, свойство, признак, преимущество или атрибут, который явно не приведен в пункте, в любом случае не должны ограничить объем такого пункта. Соответственно, описание и чертежи должны рассматриваться в иллюстративном, а не в ограничительном смысле.

Формула изобретения

1. Способ оптимизации твердотельного накопителя, содержащий следующие стадии:
 - получение порогового размера фрагмента, в котором пороговый размер фрагмента вычисляется на основании, по меньшей мере, одной или нескольких скоростей последовательной записи твердотельного накопителя (SSD) или скорости ввода-вывода (I/O) в/из SSD в секунду;
 - определение, является ли фрагмент свободного пространства на диске SSD меньше порогового размера фрагмента;
 - если фрагмент свободного пространства на диске SSD меньше порогового размера фрагмента, удаление фрагмента свободного пространства;
 - и, если фрагмент свободного пространства на диске SSD не меньше порогового

размера фрагмента, фрагмент свободного пространства используется для хранения данных.

2. Способ по п.1, в котором пороговый размер фрагмента вычисляется на основании, по меньшей мере, скорости последовательной записи SSD.

3. Способ по п.1, в котором пороговый размер фрагмента вычисляется, используя одну или несколько стадий, включающих деление скорости последовательной записи SSD на скорость ввода-вывода SSD в секунду.

4. Способ по п.1, в котором удаление фрагмента свободного пространства содержит:

определение, когда использование одного или нескольких ресурсов, требуемых для удаления фрагмента свободного пространства, отвечает критериям бездействия; и отклик на использование одного или некоторого количества ресурсов, отвечающих критериям бездействия, для удаления фрагмента свободного пространства.

5. Способ по п.1, в котором удаление фрагмента свободного пространства содержит:

наполнение фрагмента свободного пространства, по меньшей мере, частью файла.

6. Способ по п.5, в котором файл представляет собой один или несколько временных файлов наполнителя или редко используемый файл.

7. Способ по п.1, в котором пороговый размер фрагмента вычисляется на основании, по меньшей мере, скорости ввода-вывода в/из SSD в секунду.

8. Способ по п.1, в котором пороговый размер фрагмента определяется путем: получения порогового размера фрагмента от пользователя;

вычисления порогового размера фрагмента;

получения порогового размера фрагмента от аппаратно зашитого элемента в SSD; определения порогового размера фрагмента, проверяя различные значения порогового размера фрагмента и выбирая пороговый размер фрагмента, обеспечивающий оптимальные рабочие характеристики.

9. Оптимизатор твердотельного накопителя (SSD), содержащий:

средство для получения порогового размера фрагмента, в котором пороговый размер фрагмента вычисляется на основании, по меньшей мере, одной или нескольких скоростей последовательной записи SSD или скорости ввода-вывода (I/O) в/из SSD в секунду;

средство для определения, является ли фрагмент свободного пространства на диске SSD меньше порогового размера фрагмента;

средство для удаления фрагмента свободного пространства, если фрагмент свободного пространства на диске SSD меньше порогового размера фрагмента; и

средство для сохранения фрагмента свободного пространства для хранения данных, если фрагмент свободного пространства на диске SSD не меньше порогового размера фрагмента.

10. Оптимизатор SSD по п.9, дополнительно содержащий средство для вычисления порогового размера фрагмента на основании, по меньшей мере, скорости последовательной записи SSD.

11. Оптимизатор SSD по п.9, дополнительно содержащий средство для вычисления порогового размера фрагмента, сконфигурированный для определения порогового размера фрагмента путем деления, по меньшей мере, скорости последовательной записи SSD на скорость ввода-вывода SSD.

12. Оптимизатор SSD по п.9, в котором средство для удаления фрагмента свободного пространства содержит;

средство для определения, когда использование одного или нескольких ресурсов, требуемых для удаления фрагмента свободного пространства, отвечает критериям бездействия.

5 13. Оптимизатор SSD по п.9, в котором средство для удаления фрагмента свободного пространства содержит

средство для заполнения фрагмента свободного пространства, по меньшей мере, частью файла.

10 14. Оптимизатор SSD по п.13, в котором файл представляет собой один или несколько

временных файлов наполнителя или редко используемый файл.

15 15. Оптимизатор SSD по п.9, дополнительно содержащий средство для вычисления порогового размера фрагмента на основании, по меньшей мере, скорости ввода-вывода SSD в секунду.

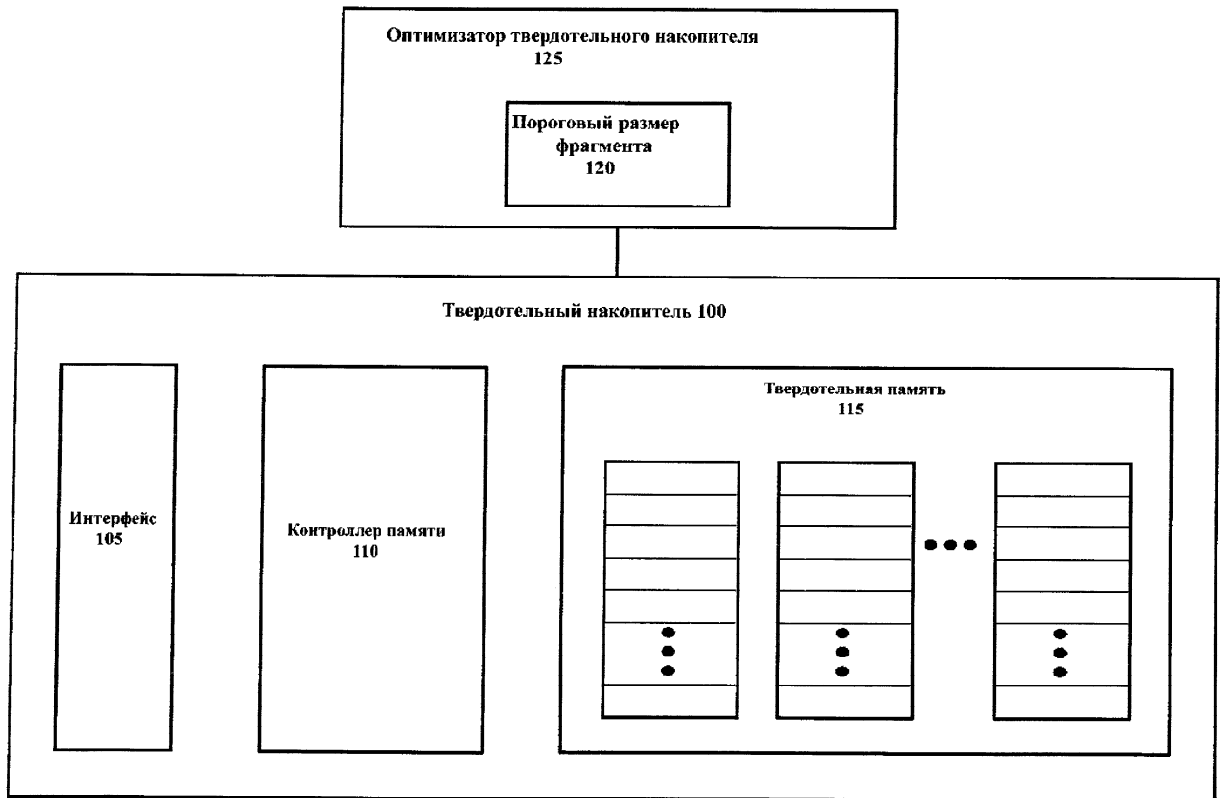
16. Оптимизатор SSD по п.9, в котором средство для получения пороговых фрагментов дополнительно содержит:

средство для получения порогового размера фрагмента от пользователя;

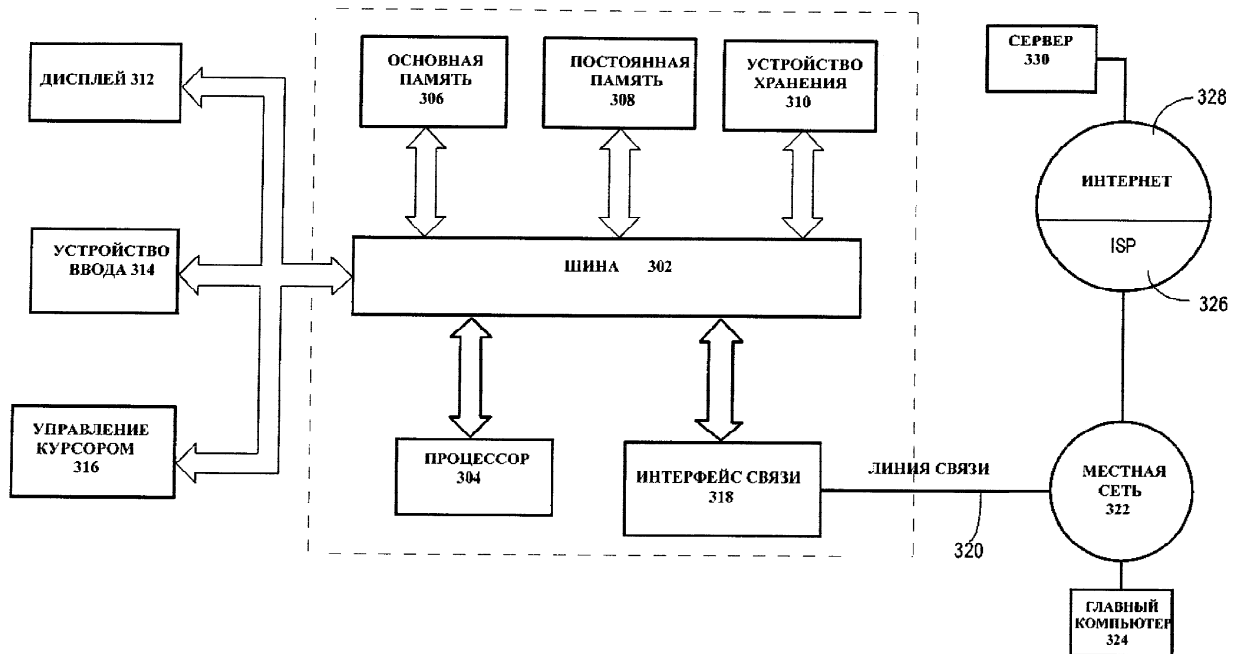
20 средство для получения порогового размера фрагмента от аппаратно зашитого элемента в SSD или

средство для определения порогового размера фрагмента, проверяя различные значения порогового фрагмента, и выбор порогового размера фрагмента, обеспечивающий оптимальные рабочие характеристики.

25 17. Машинно-считываемый носитель данных, содержащий одну или несколько последовательностей команд, при выполнении которых одним или несколькими процессорами, процессоры реализуют способ, согласно любому из пп.1-8.



Фиг. 1



Фиг. 3