



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: 2012150100/08, 22.05.2012

(24) Дата начала отсчета срока действия патента:
22.05.2012

Приоритет(ы):

(30) Конвенционный приоритет:
10.06.2011 US 13/157,704

(43) Дата публикации заявки: 27.05.2014 Бюл. № 15

(45) Опубликовано: 20.12.2015 Бюл. № 35

(56) Список документов, цитированных в отчете о
поиске: US 2011/0060887 A1, 10.03.2011. US
2007/0294490 A1, 20.12.2007. US 2010/0124105
A1, 20.05.2010. US 2011/0093505 A1, 21.04.2011.
RU 2420019 C1, 27.05.2011.(85) Дата начала рассмотрения заявки РСТ на
национальной фазе: 23.11.2012(86) Заявка РСТ:
IB 2012/052558 (22.05.2012)

Адрес для переписки:

105082, Москва, Спартаковский пер., д. 2, стр. 1,
секция 1, этаж 3, "ЕВРОМАРКПАТ"

(72) Автор(ы):

Чарлз У. ГЕЙНИ Джр. (US),
Кеннет Джеймс ОКС (US),
Томас Брайан МАТИАС (US),
Питер Кеннет ШВЕД (US),
Питер Дейна ДРАЙЭВЕР (US),
Питер Гримм САТТОН (US),
Харри ЮДЕНФРЕНД (US),
Эльпида ТЗОРТЗАТОС (US),
Стивен Гарднер ГЛАССЕН (US)

(73) Патентообладатель(и):

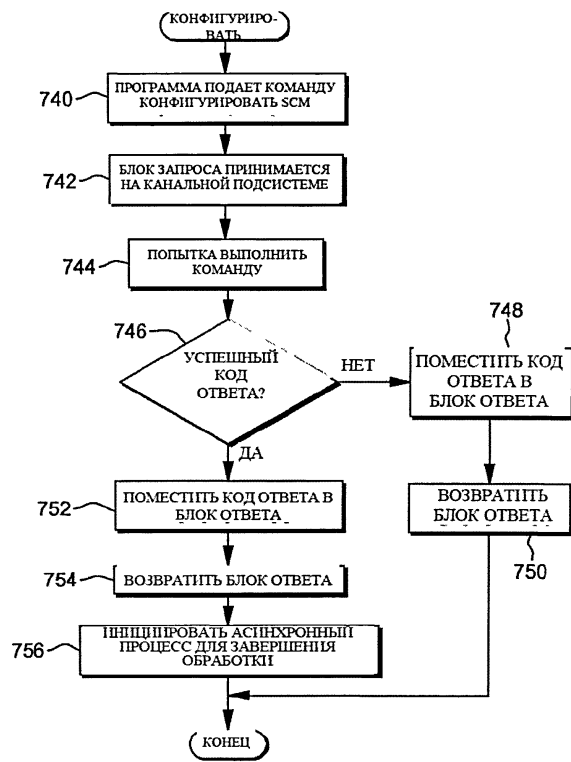
ИНТЕРНЭШНЛ БИЗНЕС МАШИНЗ
КОРПОРЕЙШН (US)

(54) КОМАНДА КОНФИГУРИРОВАНИЯ ТВЕРДОТЕЛЬНОГО ЗАПОМИНАЮЩЕГО УСТРОЙСТВА

(57) Реферат:

Изобретение относится к вычислительной технике. Технический результат заключается в обеспечении конфигурирования твердотельного запоминающего устройства для конкретного раздела или виртуализированного гостя в условиях, когда управляющая программа изначально не знает о конфигурации твердотельного запоминающего устройства. Способ выполнения команды для выполнения команды конфигурирования твердотельного запоминающего устройства (Configure Storage Class Memory) в вычислительной системе, имеющей доступ к основному запоминающему устройству и твердотельному запоминающему устройству, включает: получение подсистемой ввода-вывода информации о конфигурации

твердотельного запоминающего устройства; сообщение подсистемой ввода-вывода управляющей команде по меньшей мере части полученной информации о конфигурации твердотельного запоминающего устройства; получение подсистемой ввода-вывода блока запроса, содержащего командный код, указывающий команду конфигурирования твердотельного запоминающего устройства и значение общего размера для определения запрашиваемого количества приращений твердотельного запоминающего устройства; инициирование процесса конфигурирования для конфигурирования твердотельного запоминающего устройства. 3 н. и 9 з.п. ф-лы, 16 ил.



ФИГ. 7В



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) **ABSTRACT OF INVENTION**

(21)(22) Application: 2012150100/08, 22.05.2012

(24) Effective date for property rights:
22.05.2012

Priority:

(30) Convention priority:
10.06.2011 US 13/157,704

(43) Application published: 27.05.2014 Bull. № 15

(45) Date of publication: 20.12.2015 Bull. № 35

(85) Commencement of national phase: 23.11.2012

(86) PCT application:
IB 2012/052558 (22.05.2012)

Mail address:

105082, Moskva, Spartakovskij per., d. 2, str. 1,
sektija 1, ehtazh 3, "EVROMARKPAT"

(72) Inventor(s):

Charlz U. GEJNI Dzhr. (US),
Kennet Dzhejms OKS (US),
Tomas Brajan MATIAS (US),
Piter Kennet ShVED (US),
Piter Dejna DRAJEhVER (US),
Piter Grimm SATTON (US),
Kharri JuDENFREND (US),
Ehl'pida TZORTZATOS (US),
Stiven Gardner GLASSEN (US)

(73) Proprietor(s):

INTERNEhShNL BIZNES MASHINZ
KORPOREJShN (US)

(54) **SOLID-STATE MEMORY CONFIGURATION INSTRUCTION**

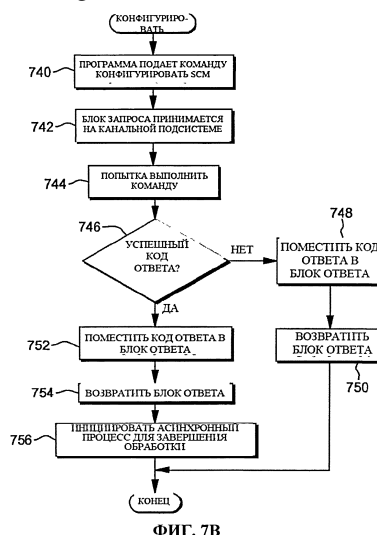
(57) Abstract:

FIELD: physics, computation hardware.

SUBSTANCE: invention relates to computer engineering. Execution of instruction on Configure Storage Class Memory in computation system with access to the main memory and solid-stage memory. This process the steps that follow. Data I/O subsystem receives data on solid-state memory configuration. Data I/O subsystem dispatches to control instruction, at least, a portion of aforesaid received data. Said data I/O subsystem receives request unit including command code that specifies solid-state configuration instruction and magnitude of total size for determination of requested amount of increments for solid-state memory. Configuration process is initiated for solid-stage memory configuring.

EFFECT: configuration for definite section or virtualised guest when control program is not aware of solid-state memory configuration.

12 cl, 16 dwg



ОБЛАСТЬ ИЗОБРЕТЕНИЯ

Одна или несколько особенностей настоящего изобретения относятся, в общем, к вспомогательному запоминающему устройству вычислительной среды и, в частности, к особенностям управления вспомогательным запоминающим устройством.

ПРЕДШЕСТВУЮЩИЙ УРОВЕНЬ ТЕХНИКИ ИЗОБРЕТЕНИЯ

Вычислительная среда может содержать основное запоминающее устройство (также известное как основная память), а также вспомогательное запоминающее устройство. Основное запоминающее устройство - это запоминающее устройство, доступное для процессора, которое является произвольно адресуемым посредством, например, абсолютного адреса. Основное запоминающее устройство рассматривается как запоминающее устройство с быстрой выборкой по сравнению с таким вспомогательным запоминающим устройством, как запоминающие устройства прямого доступа (DASD) или твердотельное запоминающее устройство. Кроме того, адресация основного запоминающего устройства считается более простой, чем адресация DASD или твердотельного запоминающего устройства.

Твердотельное запоминающее устройство, также называемое твердотельным устройством хранения данных, которое представляет собой объем внешнего запоминающего устройства снаружи классического основного запоминающего устройства, обеспечивает более быстрый доступ, чем запоминающие устройства прямого доступа. В отличие от DASD, твердотельное запоминающее устройство, как правило, не реализовано как вращающиеся диски с механической рукой, но, вместо этого, - как немеханические твердотельные части. Как правило, твердотельное запоминающее устройство реализовано как группы твердотельных устройств, соединенные с вычислительной системой посредством нескольких адаптеров ввода-вывода (I/O), которые применяют для отображения технологии устройства ввода-вывода на шину памяти центрального процессора (процессоров).

КРАТКОЕ ОПИСАНИЕ ИЗОБРЕТЕНИЯ

Для устранения недостатков уровня техники предложен способ выполнения команды для выполнения команды конфигурирования твердотельного запоминающего устройства (Configure Storage Class Memory) в вычислительной системе, имеющей доступ к основному запоминающему устройству и твердотельному запоминающему устройству, причем твердотельное запоминающее устройство включает множество твердотельных устройств, включающий: получение подсистемой ввода-вывода, осуществляемое на основании запроса управляющей программы, выполняющейся внутри раздела системы или виртуализированного гостя системы и не имеющей непосредственного доступа к твердотельному запоминающему устройству, информации о конфигурации твердотельного запоминающего устройства; сообщение подсистемой ввода-вывода управляющей команде по меньшей мере части полученной информации о конфигурации твердотельного запоминающего устройства, причем по меньшей мере часть полученной информации содержит максимальное количество приращений твердотельного запоминающего устройства, которое может быть сконфигурировано в запрашивающую конфигурацию, включающую раздел системы или виртуализированного гостя системы; получение подсистемой ввода-вывода, на основании вышеупомянутого сообщения управляющей программе, блока запроса, содержащего командный код, указывающий команду конфигурирования твердотельного запоминающего устройства и значение общего размера для определения запрашиваемого количества приращений твердотельного запоминающего устройства, подлежащего конфигурированию в адресное пространство твердотельного запоминающего устройства из доступного для

системы пула твердотельного запоминающего устройства, причем запрошенное количество приращений конфигурируют для запрашивающей конфигурации, а адресное пространство твердотельного запоминающего устройства представляет собой одно адресное пространство, перекрывающее множество твердотельных устройств и

5 заполняемое приращениями твердотельного запоминающего устройства для запрашивающей конфигурации, и скрывает подробности физической реализации твердотельного запоминающего устройства; на основании командного кода - инициирование процесса конфигурирования для конфигурирования твердотельного

10 запоминающего устройства, при этом процесс сконфигурирован для выделения адресному пространству твердотельного запоминающего устройства запрашиваемого количества приращений твердотельного запоминающего устройства, определенного в общем значении размера, причем инициирование процесса конфигурирования включает выполнение одной или нескольких проверок достоверности; продолжение осуществления

15 процесса конфигурирования твердотельного запоминающего устройства в ответ на положительный результат одной или нескольких проверок достоверности; причем одна или несколько проверок достоверности включают определение того, что запрашиваемое общее значение размера не превышает количество приращений твердотельного

20 запоминающего устройства в инициализированном состоянии, а продолжение осуществления процесса конфигурирования включает изменение внутреннего элемента управления для каждого конфигурируемого приращения, обеспечивающее возможность конфигурирования приращений таким образом, чтобы быть доступным для запросов перемещения ввода-вывода запрашивающей конфигурации; и сохранение кода ответа

25 в поле кода ответа блока ответа, указывающего на то, был ли инициирован процесс конфигурирования, при этом блок ответа содержит код длины, указывающий длину блока ответа и поле кода ответа.

Объектами изобретения являются также соответствующие компьютерная система и машиночитаемый носитель данных. Технический результат, достигаемый при осуществлении изобретения, заключается в создании инструмента для конфигурирования

30 твердотельного запоминающего устройства для конкретного раздела или виртуализированного гостя в условиях, когда управляющая программа изначально не знает о конфигурации твердотельного запоминающего устройства.

КРАТКОЕ ОПИСАНИЕ ГРАФИЧЕСКИХ МАТЕРИАЛОВ

Предпочтительные варианты осуществления изобретения будут описаны только в виде примера с отсылкой к следующим графическим материалам, на которых:

35 фиг. 1А иллюстрирует один вариант осуществления вычислительной среды, в которой содержится и используется одна или несколько особенностей настоящего изобретения;

фиг. 1Б иллюстрирует другой вариант осуществления вычислительной среды, в которой содержится и используется одна или несколько особенностей настоящего изобретения;

40 фиг. 2А иллюстрирует один вариант осуществления блока запроса операций расширенной схемы асинхронного перемещения данных, используемого согласно одной из особенностей настоящего изобретения;

фиг. 2Б иллюстрирует один вариант осуществления блока операций расширенной схемы асинхронного перемещения данных, используемого согласно одной из

45 особенностей настоящего изобретения;

фиг. 2В иллюстрирует один вариант осуществления блока запроса операций расширенной схемы асинхронного перемещения данных, используемого согласно одной из особенностей настоящего изобретения;

фиг.2Г иллюстрирует один вариант осуществления блока ответа расширенной схемы асинхронного перемещения данных, используемого согласно одной из особенностей настоящего изобретения;

5 фиг.2Д иллюстрирует один вариант осуществления блока спецификации перемещения расширенной схемы асинхронного перемещения данных, используемого согласно одной из особенностей настоящего изобретения;

фиг.2Е иллюстрирует один вариант осуществления косвенного адресного слова данных расширенной схемы асинхронного перемещения данных, используемого согласно одной из особенностей настоящего изобретения;

10 фиг.3А иллюстрирует один вариант осуществления команды Start Subchannel, используемой согласно одной из особенностей настоящего изобретения;

фиг.3Б иллюстрирует один вариант осуществления логики, связанной с командой Start Subchannel, согласно одной из особенностей настоящего изобретения;

15 фиг.3В иллюстрирует один вариант осуществления команды Clear Subchannel, используемой согласно одной из особенностей настоящего изобретения;

фиг.3Г иллюстрирует один вариант осуществления команды Test Subchannel, используемой согласно одной из особенностей настоящего изобретения;

фиг.3Д иллюстрирует один вариант осуществления команды Modify Subchannel, используемой согласно одной из особенностей настоящего изобретения;

20 фиг.3Е иллюстрирует один вариант осуществления команды Store Subchannel, используемой согласно одной из особенностей настоящего изобретения;

фиг.4А иллюстрирует один вариант осуществления блока информации о субканале расширенной схемы асинхронного перемещения данных, используемого согласно одной из особенностей настоящего изобретения;

25 фиг.4Б иллюстрирует один вариант осуществления управляющего слова управления трактами расширенной схемы асинхронного перемещения данных, используемого согласно одной из особенностей настоящего изобретения;

фиг.4В иллюстрирует один вариант осуществления слова статуса субканала, используемого согласно одной из особенностей настоящего изобретения;

30 фиг.4Г иллюстрирует один вариант осуществления расширенного слова статуса расширенной схемы асинхронного перемещения данных, используемого согласно одной из особенностей настоящего изобретения;

35 фиг.5А иллюстрирует один вариант осуществления блока запроса сохранения информации о твердотельном запоминающем устройстве, используемого согласно одной из особенностей настоящего изобретения;

фиг.5Б иллюстрирует один вариант осуществления блока ответа сохранения информации о твердотельном запоминающем устройстве, используемого согласно одной из особенностей настоящего изобретения;

40 фиг.5В иллюстрирует один вариант осуществления записи списка адресов запоминающего устройства, используемого согласно одной из особенностей настоящего изобретения;

фиг.5Г иллюстрирует один вариант осуществления логики, связанной с командой Store Storage Class Memory Information, согласно одной из особенностей настоящего изобретения;

45 фиг.6А иллюстрирует один пример диаграммы состояний, отображающей различные состояния твердотельного запоминающего устройства, согласно одной из особенностей настоящего изобретения;

фиг.6Б иллюстрирует один вариант осуществления диаграммы состояний,

показывающей состояния операций и состояния данных, согласно одной из особенностей настоящего изобретения;

фиг.7А иллюстрирует один вариант осуществления блока запроса конфигурирования твердотельного запоминающего устройства, используемого согласно одной из особенностей настоящего изобретения;

фиг.7Б иллюстрирует один вариант осуществления блока ответа конфигурирования твердотельного запоминающего устройства, используемого согласно одной из особенностей настоящего изобретения;

фиг.7В-7Г иллюстрируют один вариант осуществления логики, связанной с командой Configure Storage Class Memory, используемой согласно одной из особенностей настоящего изобретения;

фиг.8А иллюстрирует один пример блока ответа на уведомление, используемого согласно одной из особенностей настоящего изобретения;

фиг.8Б иллюстрирует один вариант осуществления блока запроса сохранения информации о событиях, используемого согласно одной из особенностей настоящего изобретения;

фиг.8В иллюстрирует один вариант осуществления блока ответа сохранения информации о событиях, используемого согласно одной из особенностей настоящего изобретения;

фиг.9А иллюстрирует один вариант осуществления блока запроса деконфигурирования твердотельного запоминающего устройства, используемого согласно одной из особенностей настоящего изобретения;

фиг.9Б иллюстрирует один вариант осуществления записи списка запроса приращений твердотельного запоминающего устройства, используемого согласно одной из особенностей настоящего изобретения;

фиг.9В иллюстрирует один вариант осуществления блока ответа деконфигурирования твердотельного запоминающего устройства, используемого согласно одной из особенностей настоящего изобретения;

фиг.9Г-9Д иллюстрируют один вариант осуществления логики, связанной с командой Deconfigure Storage Class Memory, используемой согласно одной из особенностей настоящего изобретения;

фиг.10 иллюстрирует один вариант осуществления компьютерного программного продукта, в котором содержится одна или несколько особенностей настоящего изобретения;

фиг.11 иллюстрирует один вариант осуществления хост-компьютерной системы, в которой содержится и используется одна или несколько особенностей настоящего изобретения;

фиг.12 иллюстрирует дополнительный пример компьютерной системы, в которой содержится и используется одна или несколько особенностей настоящего изобретения;

фиг.13 иллюстрирует другой пример компьютерной системы, содержащей компьютерную сеть, в которой содержится и используется одна или несколько особенностей настоящего изобретения;

фиг.14 иллюстрирует один вариант осуществления различных элементов компьютерной системы, в которой содержится и используется одна или несколько особенностей настоящего изобретения;

фиг.15А иллюстрирует один вариант осуществления блока выполнения компьютерной системы согласно фиг.14, в которой содержится и используется одна или несколько особенностей настоящего изобретения;

фиг.15Б иллюстрирует один вариант осуществления блока перехода компьютерной системы согласно фиг.14, в которой содержится и используется одна или несколько особенностей настоящего изобретения;

фиг.15В иллюстрирует один вариант осуществления блока загрузки/сохранения компьютерной системы согласно фиг.14, в которой содержится и используется одна или несколько особенностей настоящего изобретения; и

фиг.16 иллюстрирует один вариант осуществления эмулированной хост-компьютерной системы, в которой содержится и используется одна или несколько особенностей настоящего изобретения.

ПОДРОБНОЕ ОПИСАНИЕ ПРЕДПОЧТИТЕЛЬНЫХ ВАРИАНТОВ ОСУЩЕСТВЛЕНИЯ

Согласно одному или нескольким вариантам осуществления изобретения для твердотельного запоминающего устройства предусмотрена абстракция, которая скрывает подробности реализации твердотельного запоминающего устройства от программы (например, от операционной системы) и предусматривает стандартный программный интерфейс канала для осуществления определенных действий, таких как управление перемещением данных между основным запоминающим устройством и твердотельным запоминающим устройством.

В одном примере предусмотрено средство, называемое средством расширенного асинхронного перемещения данных (EADM), которое позволяет программам запрашивать передачу блоков данных между основным запоминающим устройством и твердотельным запоминающим устройством, а также запрашивать другие операции. Ниже подробно описываются особенности данного средства.

Также связанным со средством EADM является необязательное средство, называемое средством освобождения EADM. Средство освобождения EADM, если оно установлено, предоставляет программе средство для указания того, что она больше не нуждается в хранении данных в одном или нескольких блоках твердотельного запоминающего устройства. То, поддерживается ли операция освобождения для всех блоков твердотельного запоминающего устройства или только для подмножества твердотельных запоминающих устройств, зависит от модели.

Когда блок твердотельного запоминающего устройства освобожден, программа может передавать в блок новые данные, в противном случае последующие запросы на передачу данных из блока в основное запоминающее устройство будут неудачными.

Один вариант осуществления вычислительной среды, который содержит и/или использует одну или несколько особенностей настоящего изобретения, описывается согласно фиг.1А. Вычислительная среда 100 основана, например, на архитектуре z/Architecture[®], предлагаемой International Business Machines Corporation (IBM[®]), Армонк, штат Нью-Йорк. Один вариант осуществления архитектуры z/Architecture[®] описан в публикации IBM[®] под названием "z/Architecture Principles of Operation", IBM Publication No. SA22-7832-08, August, 2010 г., которая ссылкой полностью включена в настоящее описание. В одном примере вычислительная среда, основанная на архитектуре z/Architecture[®], содержит систему zEnterprise 196 (z196), предлагаемую International Business Machines Corporation. IBM[®] и z/Architecture[®] являются зарегистрированными товарными знаками, и zEnterprise 196 и z196 являются товарными знаками International Business Machines Corporation, Армонк, Нью-Йорк, США. Другие названия, используемые в заявке, могут являться зарегистрированными товарными знаками, товарными знаками или названиями продуктов International Business Machines Corporation или других

компаний.

В одном примере вычислительная среда 100 содержит один или несколько центральных процессоров 102, связанных с основной памятью 104 посредством одной или нескольких шин 106. Один или несколько центральных процессоров могут выполнять

5 операционную систему 108, такую как операционная система z/OS[®], предлагаемую International Business Machines Corporation. В других примерах один или несколько центральных процессоров могут выполнять другие операционные системы или не выполнять ни одной операционной системы. z/OS[®] является зарегистрированным

10 товарным знаком International Business Machines Corporation, Армонк, Нью-Йорк, США. Центральные процессоры 102 и основная память 104 также могут быть связаны с концентратором 120 ввода-вывода посредством одного или нескольких соединений 122 (например, шин или других соединений). Концентратор ввода-вывода обеспечивает соединяемость с одним или несколькими адаптерами 130 ввода-вывода, которые

15 дополнительно связаны с одним или несколькими твердотельными устройствами 140. Адаптеры и твердотельные устройства представляют собой реализацию твердотельного запоминающего устройства (например, флэш-памяти). Концентратор ввода-вывода является частью подсистемы 145 ввода-вывода, содействующей одной или нескольким особенностям одного варианте осуществления изобретения.

20 В дополнительном варианте осуществления изобретения, как проиллюстрировано на фиг.1Б, вычислительная среда 150 может содержать центральный процессорный комплекс (CPC) 152, который основан на архитектуре z/Architecture[®], предлагаемой International Business Machines Corporation. Центральный процессорный комплекс 152 содержит, например, один или несколько разделов 154, гипервизор 156, один или

25 несколько центральных процессоров 158 и один или несколько компонентов подсистемы 160 ввода-вывода. В данном примере разделы 154 представляют собой логические разделы (например, LPAR), каждый из которых содержит набор аппаратных ресурсов системы, виртуализированный как отдельная система.

Каждый логический раздел 154 способен функционировать как отдельная система. 30 То есть каждый логический раздел может быть независимо переустановлен, при необходимости, в исходном положении в него может быть загружена операционная система или другой управляющий код, и он может работать с разными программами. Оказывается, что операционная система, или прикладная программа, выполняемая в логическом разделе, имеет доступ к системе во всей полноте, но на самом деле доступной 35 является только ее часть. Сочетание аппаратного обеспечения и лицензионного внутреннего кода (LIC), называемого аппаратно-программным обеспечением, защищает программу в одном логическом разделе от создания помех какой-либо программы в другом логическом разделе. Это позволяет нескольким различным логическим разделам работать на одном или нескольких физических процессорах способом квантования

40 времени. Используемое в данном описании программно-аппаратное обеспечение содержит, например, микрокод, милликод и/или макрокод процессора (или сущности, осуществляющей обработку данных). Оно содержит, например, команды аппаратного уровня и/или структуры данных, используемые при реализации высокоуровневого машинного кода. В одном варианте осуществления изобретения оно содержит, например, 45 собственный код, который, как правило, поставляют как микрокод, который содержит выверенное программное обеспечение, или микрокод, характерный для базового аппаратного обеспечения, и управляет доступом операционной системы к аппаратному обеспечению системы.

В данном примере некоторые из логических разделов содержат резидентную операционную систему (OS) 170, которая может отличаться для одного или нескольких логических разделов. В одном варианте осуществления изобретения по меньшей мере один логический раздел выполняет операционную систему z/OS[®], предлагаемую International Business Machines Corporation, Армонк, Нью-Йорк.

Логические разделы 154 администрируются гипервизором 156, который реализован посредством аппаратно-программного обеспечения, выполняемого на центральных процессорах 158. Каждый из логических разделов 154 и гипервизор 156 содержат одну или несколько программ, постоянно находящихся в соответствующих частях основной памяти 159, связанной с центральными процессорами. Одним примером гипервизора 156 является Processor Resource/Systems Manager (PR/SM[™]), предлагаемый International Business Machines Corporation, Армонк, Нью-Йорк.

Центральные процессоры 158 представляют собой физические процессорные ресурсы, которые выделены логическим разделам. Например, логический раздел 154 содержит один или несколько логических процессоров, каждый из которых отображает весь выделенный разделу физический процессорный ресурс 158 или его часть. Логические процессоры конкретного раздела 154 могут быть или выделены разделу, и тогда базовый процессорный ресурс является зарезервированным для этого раздела, или они могут использоваться совместно с другим разделом, и тогда базовый процессорный ресурс потенциально является доступным и для другого раздела.

Подсистема 160 ввода-вывода (только часть которой проиллюстрирована) обеспечивает соединяемость с твердотельным запоминающим устройством 180. В данном примере для твердотельного запоминающего устройства предусмотрено адресное пространство, которое представляет память как линейную, скрывающую от программы подробности физической реализации. В одном примере для твердотельного запоминающего устройства существует одно адресное пространство в масштабе системы, но с точки зрения конфигурации (например, LPAR или в другом варианте осуществления изобретения виртуализированного гостя) существует одно адресное пространство, которое для каждой конфигурации системы заполняется приращениями твердотельного запоминающего устройства. Адресное пространство твердотельного запоминающего устройства является отдельным и не зависит от адресного пространства основного запоминающего устройства.

В одном частном примере архитектуры z/Architecture[®], подсистема ввода-вывода содержит канальную подсистему, которая не только направляет поток информации между периферийными блоками управления (и устройствами) вводом-выводом и основной памятью, но также и между твердотельным запоминающим устройством и основной памятью. Однако подсистема ввода-вывода может представлять собой и иную подсистему, чем канальная подсистема.

В случае канальной подсистемы для осуществления операций EADM используют субканалы. Эти субканалы называются субканалами, относящимися к типу схемы асинхронного перемещения данных (ADM), и они связаны с операциями EADM, а не с устройствами ввода-вывода, как другие субканалы, относящиеся к типу ввода-вывода. Субканал ADM-типа не содержит ни номер устройства, ни информацию о канальном тракте. Количество субканалов ADM-типа, предусматриваемое для некоторой конфигурации, зависит от модели. Субканалы ADM-типа адресуются словом идентификации подсистемы (SID).

Субканалы ADM-типа используются средством расширенного асинхронного перемещения данных, которое представляет собой расширение канальной подсистемы.

Как описывается в данном описании, средство EADM позволяет программе запрашивать передачу блоков данных между основным запоминающим устройством и твердотельным запоминающим устройством, а также осуществление других операций, таких как сброс блока твердотельного запоминающего устройства или освобождение блока

5 твердотельного запоминающего устройства. В одном варианте осуществления изобретения, когда средство EADM установлено:

- предусмотрен один или несколько субканалов ADM-типа, которые используются для операций EADM.

- Операции EADM указаны указанным блоком операций EADM (AOB). AOB содержит блок запроса EADM (ARQB) и блок ответа EADM (ARSB) и указывает список блоков спецификации перемещения EADM (MSB). Для операции перемещения блоки MSB содержат такую информацию о блоках данных, которые подлежат перемещению, как размер блоков, исходное местоположение и конечное местоположение блоков, и направление перемещения данных.

15 Максимальное количество блоков MSB, которое может указываться посредством AOB, зависит от модели. Максимальное число блоков, которое MSB может указывать, как подлежащие перемещению или выполнению операций на них, также зависит от модели.

- Программа инициирует операции EADM, выдавая команду Start Subchannel, которая

20 указывает субканал ADM-типа и блок запроса операций (ORB) EADM. В свою очередь, EADM ORB указывает AOB. Команда передает содержимое EADM ORB в указанный субканал.

- Когда для инициирования операций EADM выдается команда Start Subchannel, канальная подсистема асинхронно осуществляет указанную операцию.

25 - Программа следит, что твердотельное запоминающее устройство при зависящем от модели минимальном размере блоков кажется параллельным по блокам. Эта зависящая от модели величина называется размером параллельности блоков SCM.

- Когда операции EADM завершены, прерывание ввода-вывода для субканала ADM-типа, на котором были инициированы операции, делается ожидающим программу.

30 EADM ORB содержит спецификацию ключа субканала и адрес AOB, подлежащего использованию. AOB указывается, например, на границе 4 Кбайт и может иметь размер до 4 Кбайт. Если требуется больше блоков MSB, чем может поместиться в 4 Кбайта AOB, AOB может указывать список блоков MSB, который расширяется на дополнительные области хранения путем использования блоков MSB, которые вместо

35 указания области памяти, подлежащей использованию для передачи данных, указывают адрес следующего MSB в списке.

Первая операция EADM запускается канальной подсистемой, использующей информацию в указанном EADM ORB и ARQB в указанном AOB для извлечения MSB. MSB содержит информацию, которая указывает операцию EADM, подлежащую

40 обработке, и управляет этой операцией.

Каждая операция EADM отображается одним MSB. MSB может указывать, например, передачу блоков данных из основного запоминающего устройства в твердотельное запоминающее устройство; передачу блоков данных из твердотельного запоминающего устройства в основное запоминающее устройство; сброс блоков твердотельного

45 запоминающего устройства; и освобождение блоков твердотельного запоминающего устройства.

Если блоки, подлежащие передаче, не являются непрерывными в основном запоминающем устройстве, может использоваться новый MSB, или MSB может

использовать косвенную адресацию путем определения списка косвенных адресных слов данных EADM (AIDAW) для указания блоков, не являющихся непрерывными.

Так как MSB определяет передачу данных только в одном направлении, в случае, когда происходит изменение в направлении передачи, следует использовать новый MSB.

Завершение операции EADM обычно указывается комбинированными ситуациями статусов окончания работы канала и окончания работы устройства. Данная комбинация статусов отображает комбинацию первичного и вторичного статусов, указывающую, что после сброса статуса субканал является доступным для другой функции запуска.

Операция EADM может быть преждевременно завершена командой Clear Subchannel. Выполнение команды Clear Subchannel завершает выполнение АОВ в субканале, сбрасывает субканал от указателей выполнения АОВ и асинхронно осуществляет функцию сброса.

Дополнительные подробности, относящиеся к EADM ORB и связанным с ним управляющим структурам, описаны ниже со ссылкой на фиг.2А-2Е. Со ссылкой, в первую очередь, на фиг.2А описывается один вариант осуществления EADM ORB.

В одном примере EADM ORB 200 содержит:

параметр 202 прерывания: данное поле сохраняется неизменным в субканале до тех пор, пока оно не будет замещено последующей командой Modify Subchannel или Start Subchannel. Указанные биты помещены в код прерывания ввода-вывода, когда для субканала происходит прерывание ввода-вывода, и когда запрос прерывания сбрасывается путем выполнения, например, команды Test Pending Interruption.

Ключ 204 субканала: данное поле образует ключ субканала для операций EADM, определяемых посредством ARQB, и применяется для извлечения ARQB, извлечения блоков MSB, сохранения ARSB и для получения доступа к основному запоминающему устройству для передачи данных. Значение этого поля является определенным значением; в противном случае, или канальной подсистемой распознается ситуация проверки программы, или распознается исключительная ситуация операнда.

Управление (X) 205 расширением ORB: данное поле указывает, является ли ORB расширенным. Данное поле представляет собой определенное значение, когда субканал ADM-типа является указанным; в противном случае, распознается или исключительная ситуация операнда, или ситуация проверки программы.

Адрес 206 блока операций EADM (АОВ): данное поле указывает адрес блока операций EADM (АОВ). Если некоторые биты данного поля не включают определенное значение, распознается или исключительная ситуация операнда, или ситуация проверки программы.

Если данное поле указывает местоположение, защищенное от извлечения, или указывает местоположение за пределами конфигурации, функция запуска не инициируется. В этом случае субканал является ожидающим статус с первичным, вторичным и предупреждающим статусом.

Приоритет 208 канальной подсистемы (CSS): данное поле содержит указатель приоритета канальной подсистемы, который назначен указанному субканалу и используется для упорядочивания выбора субканалов ADM-типа тогда, когда функция запуска подлежит инициированию для одного или нескольких субканалов, которые являются ожидающими запуск.

Приоритет 210 твердотельного запоминающего устройства (SCM): данное поле содержит указатель приоритета твердотельного запоминающего устройства (SCM), который указывает уровень приоритета, который применяется ко всем операциям

EADM, связанным с функцией запуска.

То, распознается ли содержимое поля приоритета SCM средством EADM, зависит от модели. На моделях, которые не распознают это поле, содержимое поля игнорируется, и всем операциям EADM, связанным с функцией запуска, назначается неявный указатель приоритета.

Формат 212 (FMT): данное поле указывает структуру ORB. Данное поле предназначено для содержания определенного значения тогда, когда указан субканал ADM-типа; в противном случае, распознается исключительная ситуация операнда или устанавливается код особой ситуации.

Блок операций EADM (AOB), указываемый адресом 206 AOB EADM в EADM ORB, содержит информацию, используемую для вызова операций EADM. AOB выделяется, в одном примере, на границе 4 Кбайт и имеет переменную длину.

В одном примере, как показано на фиг.2Б, AOB 220 EADM содержит три секции: блок (ARQB) 222 запроса EADM; блок (ARSB) 224 ответа EADM; и область 226 MSB, содержащую блоки MSB до определенного количества (например, 124). ARQB может указывать использование большего количества блоков MSB, чем определенное, однако только при использовании перехода MSB (то есть путем использования перехода к флагу следующего MSB в MSB).

Один вариант осуществления блока (ARQB) 222 запроса EADM описан со ссылкой на фиг.2В. В одном примере ARQB 222 содержит:

формат 230 (FMT): данное поле указывает структуру ARQB. Значение этого поля представляет собой определенное значение; в противном случае, канальной подсистемой распознается ситуация проверки программы, или в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка кода команды.

Код 232 команды: это поле предназначено для указания команды перемещения блоков EADM; в противном случае, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка кода команды.

Счетчик 234 блоков MSB: данное поле указывает счетчик блоков MSB, которые составляют запрос EADM. Максимальное число блоков MSB, которое может быть указано, зависит от модели. Значение этого поля должно быть больше нуля и меньше или равно зависящему от модели максимальному значению счетчика блоков MSB; в противном случае, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка счетчика MSB.

В дополнение к блоку запроса EADM, EADM ORB также указывает блок ответа EADM (ARSB). В данном варианте осуществления изобретения, блок ответа EADM является значимым только тогда, когда распознается исключительная ситуация. Точнее, ARSB является значимым только тогда, когда в слове статуса субканала EADM (SCSW) присутствует предупреждающий статус, расширенное слово статуса EADM (ESW) является значимым, и блок ответа EADM, хранящий (R) битов, равен единице в расширенном слове отчета EADM (ERW), каждое из них описывается ниже. Когда ARSB не является значимым, содержимое ARSB в AOB является непредсказуемым.

Если программа сохраняется в ARSB, в то время как связанный субканал является действующим субканалом, могут возникать непредсказуемые результаты.

Когда ARSB сохраняется, объем данных, которые были переданы, если это имеет место, является непредсказуемым.

Один вариант осуществления блока ответа EADM описывается со ссылкой на фиг.2Г.

В одном примере ARSB 224 содержит:

формат 240 (FMT): это поле указывает структуру ARSB. Когда ARSB сохраняется, значение этого поля сохраняется как определенное значение.

Флаги 242 исключительных ситуаций (EF): когда ARSB сохраняется, это поле, если оно установлено, указывает причину исключительной ситуации, по которой сохраняется ARSB. Примеры причин исключительной ситуации включают:

проверку программы: обнаружена ошибка программирования.

Проверку защиты: доступу к запоминающему устройству препятствует механизм защиты. Защита распространяется на извлечение ARQB, MSB, AIDAW и данные, подлежащие передаче в твердотельное запоминающее устройство, и на сохранение информации в ARSB и данные, передаваемые из твердотельного запоминающего устройства.

Проверку канальных данных: неисправленная ошибка запоминающего устройства была обнаружена в отношении данных, которые содержатся в основном запоминающем устройстве, и в настоящее время используются для выполнения операции EADM. В случае обнаружения, данная ситуация может указываться, даже если данные не используются при предварительном извлечении. Проверка канальных данных указывается тогда, когда данные, или связанный ключ, имеют недостоверный код проверки блока (CBC) в основном запоминающем устройстве, когда к этим данным обращается канальная подсистема.

Проверку управления каналом: проверка управления каналом вызывается любым машинным сбоем, оказывающим влияние на элементы управления канальной подсистемы. Ситуация включает недостоверный CBC на ARQB, ARSB, MSB, AIDAW или соответствующем связанном ключе. Ситуация может быть указана, когда недостоверный CBC обнаруживается на подвергнутом предварительному извлечению ARQB, MSB, AIDAW или соответствующем связанном ключе, даже если этот ARQB, MSB или AIDAW не используется.

Проверку средства расширенного асинхронного перемещения данных: неисправленная ошибка была обнаружена в отношении данных, которые содержатся в твердотельном запоминающем устройстве, и в настоящее время используются для выполнения операции EADM.

Идентификаторы 244 блоков управления исключительными ситуациями (ECBI): когда ARSB сохраняется, данное поле представляет собой маску из нескольких битов, биты которой, когда они установлены, указывают любой один из следующих компонентов, которые связаны с распознанной исключительной ситуацией, указываемого полем EF, или их сочетание:

- блок или блоки управления.
- Область данных основного запоминающего устройства.
- Твердотельное запоминающее устройство.

Биты, которые могут быть установлены, отображают, например, блок спецификации перемещения EADM, косвенный адрес данных EADM, данные в основном запоминающем устройстве и/или данные в твердотельном запоминающем устройстве.

Биты в поле ECBI описывают компоненты, связанные с единичной исключительной ситуацией. Если для исключительной ситуации нельзя идентифицировать ни одного компонента, данное поле содержит, например, нули.

Флаг 246 достоверности полей (FVF): когда ARSB сохраняется, это поле содержит маску из нескольких битов, биты которой указывают достоверность некоторых полей в ARSB. Если бит достоверности установлен, соответствующее поле было сохранено

и является пригодным для целей восстановления. Биты, которые могут быть установлены, отображают, например, поле неверного адреса MSB, поле неверного AIDAW, поле неверного адреса основного запоминающего устройства и/или поле неверного адреса твердотельного запоминающего устройства.

- 5 Код 248 квалификатора исключительных ситуаций (EQC): если ARSB сохраняется, данное поле содержит кодовое значение, которое дополнительно описывает исключительную ситуацию, указываемую полем флагов исключительных ситуаций. Кодовые значения могут отображать, в качестве примеров, следующее.

- 10 Не предоставлено дополнительное описание. В этом случае поле идентификаторов блоков управления исключительными ситуациями (ECBI) и те поля, достоверность которых подтверждена полем флагов достоверности полей, могут идентифицировать блоки управления, для которых распознана исключительная ситуация.

- 15 Ошибка формата: формат, указанный полем формата, является зарезервированным. В этом случае поле идентификаторов блоков управления исключительными ситуациями (ECBI) и те поля, достоверность которых подтверждена полем флагов достоверности полей, могут идентифицировать блоки управления, для которых распознана исключительная ситуация.

Ошибка кода команды: не распознано значение, определенное в поле команды блока ARQB.

- 20 Ошибка счетчика блоков MSB: значение, указанное в поле счетчика блоков MSB в блоке ARQB, равно нулю или превышает зависящий от модели максимум блоков MSB, который может быть указан.

- 25 Ошибка флагов: биты флагов, указанные полем флагов, являются зарезервированными. В этом случае поле идентификаторов блоков управления исключительными ситуациями (ECBI) и те поля, достоверность которых подтверждена полем флагов достоверности полей, могут идентифицировать блоки управления, для которых распознана исключительная ситуация.

- 30 Ошибка кода операции: указывается зарезервированное значение кода операции. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация.

- 35 Ошибка размера блоков: указывается зарезервированное значение размера блоков. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация.

- 40 Ошибка счетчика блоков: значение, указываемое в поле счетчика блоков MSB, равно нулю или превышает зависящее от модели максимальное число блоков, которые могут указываться посредством MSB. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация.

- 45 Ошибка указания адреса основного запоминающего устройства: адрес основного запоминающего устройства указан на неправильной границе. Такой адрес мог быть указан посредством блока MSB или AIDAW. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является достоверным, поле содержит адрес AIDAW, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса

основного запоминающего устройства является достоверным, поле содержит адрес основного запоминающего устройства, для которого распознана исключительная ситуация.

5 Ошибка указания адреса твердотельного запоминающего устройства: адрес твердотельного запоминающего устройства указан на неправильной границе. Такой адрес указывается посредством MSB. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является достоверным, поле содержит
10 адрес AIDAW, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса твердотельного запоминающего устройства является достоверным, поле содержит адрес твердотельного запоминающего устройства, для которого распознана исключительная ситуация.

Исключительная ситуация адреса основного запоминающего устройства: средство
15 EADM попыталось использовать адрес, который является недоступным в данной конфигурации или охватывает максимальный адрес запоминающего устройства. Такой адрес мог быть указан MSB или является результатом приращения адресов основного запоминающего устройства в ходе передачи данных. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит
20 адрес MSB, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является достоверным, поле содержит адрес AIDAW, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса основного запоминающего устройства является достоверным, поле содержит адрес
25 основного запоминающего устройства, для которого распознана исключительная ситуация.

Исключительная ситуация адреса твердотельного запоминающего устройства: средство EADM попыталось использовать адрес твердотельного запоминающего устройства, который является недоступным в данной конфигурации. Такой адрес мог
30 быть указан посредством MSB или являться результатом приращения адресов твердотельного запоминающего устройства в ходе передачи данных. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является
35 достоверным, поле содержит адрес AIDAW, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса твердотельного запоминающего устройства является достоверным, поле содержит адрес твердотельного запоминающего устройства, для которого распознана исключительная ситуация.

40 Ошибка основного запоминающего устройства: обнаружена неисправленная ошибка основного запоминающего устройства. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является
45 достоверным, поле содержит адрес AIDAW, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса основного запоминающего устройства является достоверным, поле содержит адрес основного запоминающего устройства, для которого распознана исключительная

ситуация.

Ошибка списка блоков MSB: AOB указывает список блоков MSB, который пересекает границу 4 Кбайт без указания перехода к следующему MSB (BNM) для пересечения границы, или MSB представляет собой последний MSB в указанном списке MSB, и BNM указывается этим MSB. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация.

Ошибка перехода MSB: AOB указывает список MSB, в котором один из блоков MSB (исходный MSB перехода) указывает переход к следующему MSB (BNM), и MSB, который является целью перехода, также указывает BNM. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес исходного MSB перехода, для которого распознана исключительная ситуация.

Ошибка списка AIDAW: MSB указывает список косвенных адресных слов данных EADM (AIDAW), который пересекает границу 4Кбайт без указания перехода к следующему AIDAW (BNA) для пересечения границы. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является достоверным, поле содержит адрес AIDAW, для которого распознана исключительная ситуация.

Ошибка перехода AIDAW: MSB указывает список косвенных адресных слов данных EADM (AIDAW), в котором AIDAW (исходное AIDAW перехода) указывает переход к исходному AIDAW (BNA), и AIDAW, которое является целью перехода, также указывает BNA. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является достоверным, поле содержит адрес исходного AIDAW перехода, для которого распознана исключительная ситуация.

Временная ошибка твердотельного запоминающего устройства: обнаружена восстанавливаемая ошибка твердотельного запоминающего устройства. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является достоверным, поле содержит адрес AIDAW, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса твердотельного запоминающего устройства является достоверным, поле содержит адрес твердотельного запоминающего устройства, для которого распознана исключительная ситуация.

Ошибка отсутствия поддержки операции освобождения: операция освобождения указана для твердотельного запоминающего устройства, для которого операция освобождения не поддерживается. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является достоверным, поле содержит адрес AIDAW, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса твердотельного запоминающего устройства является достоверным, поле содержит адрес твердотельного запоминающего устройства, для которого распознана исключительная ситуация.

Ошибка чтения освобожденных данных: операция чтения указана для твердотельного запоминающего устройства, для которого последней успешной операцией являлась операция освобождения. Если поле флагов достоверности полей указывает, что поле неверного адреса MSB является достоверным, поле содержит адрес MSB, для которого
 5 распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса AIDAW является достоверным, поле содержит адрес AIDAW, для которого распознана исключительная ситуация. Если поле флагов достоверности полей указывает, что поле неверного адреса твердотельного запоминающего устройства является достоверным, поле содержит адрес твердотельного запоминающего устройства,
 10 для которого распознана исключительная ситуация.

Неверный адрес 250 MSB: если установлен указанный бит поля флагов достоверности полей, данное поле указывает адрес MSB, для которого распознана исключительная ситуация.

Неверный адрес 252 AIDAW: если установлен указанный бит поля флагов
 15 достоверности полей, данное поле указывает адрес AIDAW, для которого распознана исключительная ситуация.

Неверный адрес 254 основного запоминающего устройства: если установлен указанный бит поля флагов достоверности полей, данное поле указывает адрес блока основного запоминающего устройства, для которого распознана исключительная
 20 ситуация.

Неверный адрес 256 твердотельного запоминающего устройства: когда установлен указанный бит поля флагов достоверности полей, данное поле содержит адрес (например, 64-битный) SCM блока твердотельного запоминающего устройства, для которого распознана исключительная ситуация.

В дополнение к блоку запроса EADM и блоку ответа EADM блок операций EADM (AOB) также указывает один или несколько блоков спецификации перемещения EADM (MSB). AOB может содержать, например, до 124 блоков MSB. Однако программа может указывать более чем 124 блока MSB путем указания большего количества блоков MSB в поле счетчика ARQB и путем использования флага перехода к следующему MSB
 25 (BNM) для перехода к продолжению списка MSB. Возможно множество продолжений списка MSB, но, в одном варианте осуществления изобретения, ни одно из этих продолжений не может пересекать границу 4 Кбайт без использования BNM для пересечения этой границы. От продолжений списка MSB не требуется, чтобы они были непрерывными. Каждое продолжение списка MSB, если указано его начало на границе
 30 4Кбайт, может содержать до 128 блоков MSB. Общее количество блоков MSB в списке MSB указывается полем счетчика блоков MSB в ARQB.

Блок спецификации перемещения EADM описывает, например, блоки запоминающего устройства, которые подлежат перемещению между основным запоминающим устройством и твердотельным запоминающим устройством, или операцию, подлежащую
 40 выполнению на блоке запоминающего устройства. Один вариант осуществления MSB 226 описывается со ссылкой на фиг.2Д, и содержит, например:

формат 260 (FMT): данное поле указывает структуру MSB. Значение этого поля представляет собой определенное значение; в противном случае, в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка формата MSB.

Код 262 операции (OC): данное поле указывает операцию, подлежащую выполнению. В одном примере, когда указанный бит в описываемом ниже поле флагов, представляет собой определенное значение, данное поле указывает операцию, подлежащую
 45 осуществлению. Примеры операций включают:

сброс твердотельного запоминающего устройства: Когда указывается этот код, сбрасывается (например, устанавливается на ноль) твердотельное запоминающее устройство, указываемое адресом твердотельного запоминающего устройства, размером блока и полем счетчика блоков. Поле адреса основного запоминающего устройства

5 не имеет значения для данной операции.

Чтение: когда указывается этот код, данные указываются как подлежащие передаче из твердотельного запоминающего устройства в основное запоминающее устройство. Поля счетчика блоков, размера блоков, адреса основного запоминающего устройства и адреса твердотельного запоминающего устройства не имеют значения для данной

10 операции.

Запись: когда указывается этот код, данные указываются как подлежащие передаче из основного запоминающего устройства в твердотельное запоминающее устройство. Поля счетчика блоков, размера блоков, адреса основного запоминающего устройства и адреса твердотельного запоминающего устройства не имеют значения для данной

15 операции.

Освобождение: когда указывается этот код, и установлено средство освобождения EADM, освобождается твердотельное запоминающее устройство, указываемое полями адреса твердотельного запоминающего устройства, размером блоков и счетчика блоков. Поле адреса основного запоминающего устройства не имеет значения для данной

20 операции. Когда средство освобождения EADM не установлено, данный код резервируется.

Если указывается зарезервированное значение, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка кода операции.

25 Когда бит указанного флага не является определенным значением, данное поле является зарезервированным и не проверяется.

Флаги 264: данное поле идентифицирует один или несколько флагов, которые могут быть установлены. Биты флагов, которые являются зарезервированными, устанавливаются на определенное значение; в противном случае, в поле кода

30 квалификатора исключительных ситуаций ARSB указывается ошибка флагов. Примеры флагов включают:

- флаг перехода к следующему MSB (BNM): данный флаг, если он установлен, указывает, что MSB не указывает операцию EADM и не используется для передачи данных. Вместо этого поле адреса основного запоминающего устройства указывает

35 адрес следующего MSB, подлежащего использованию для указания операции EADM.

Если установлен флаг BNM (например, один), и поле адреса основного запоминающего устройства указывает MSB, в котором также установлен флаг BNM, канальной подсистемой распознается ситуация проверки программы, адрес исходного MSB перехода сохраняется в поле неверного адреса MSB в ARSB, и в поле кода

40 квалификатора исключительных ситуаций ARSB указывается ошибка перехода MSB.

Если MSB заканчивается на границе 4 Кбайт, АОВ указывает дополнительные блоки MSB, и флаг BNM не установлен, канальной подсистемой распознается ситуация проверки программы, адрес этого MSB сохраняется в поле неверного адреса MSB в ARSB, и в поле кода квалификатора исключительных ситуаций ARSB указывается

45 ошибка списка MSB.

Если MSB является последним MSB в указанном списке блоков MSB, и установлен флаг BNM, канальной подсистемой распознается ситуация проверки программы, адрес этого MSB сохраняется в поле неверного адреса MSB в ARSB, и в поле кода

квалификатора исключительных ситуаций ARSB указывается ошибка списка блоков MSB.

Когда установлен флаг BNM, остальные флаги, поле кода операции, поле размера блоков, поле счетчика блоков в наборе и адрес твердотельного запоминающего устройства не имеют значения.

- Флаг косвенной адресации: данный флаг, если он установлен, указывает, что поле адреса основного запоминающего устройства указывает адрес косвенного адресного слова данных EADM (AIDAW) или первого AIDAW из списка AIDAW, которое указывает местоположение или, соответственно, местоположения в основном запоминающем устройстве, подлежащие использованию для передачи данных. Данный флаг, если он не установлен, указывает, что поле адреса основного запоминающего устройства указывает адрес местоположения в основном запоминающем устройстве, подлежащего использованию для передачи данных.

Размер 266 блоков (BS): данное поле указывает размер и границу подлежащих передаче блоков данных, как для основного запоминающего устройства, так и для твердотельного запоминающего устройства, или размер и границу блока твердотельного запоминающего устройства, подлежащего выполнению операции на нем (например, сбросу или освобождению). Примеры включают:

4К: когда кодовым значением является одно определенное значение, подлежащие передаче блоки данных находятся на границе 4 Кбайт и имеют размер 4 Кбайт.

1М: когда кодовым значением является другое определенное значение, подлежащие передаче блоки данных находятся на границе 1 Мбайт и имеют размер 1 Мбайт.

Если указывается зарезервированное значение, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка размера блоков.

Когда установлен флаг BNM, данное поле игнорируется.

Счетчик 268 блоков: данное поле содержит счетчик блоков данных, подлежащих передаче или операциям на них (например, сбросу или освобождению). На основе поля размера блоков, данный счетчик представляет собой счетчик блоков размером 4 Кбайт или 1 Мбайт.

Значение данного поля должно быть больше нуля и меньше или равно зависящему от модели максимальному числу блоков, которое может указываться посредством MSB; в противном случае, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка счетчика блоков.

Когда установлен флаг BNM, данное поле является зарезервированным и не проверяется.

Адрес 270 основного запоминающего устройства: когда поле кода операции имеет значение (например, флаг BNM не установлен) и указывает или операцию чтения, или операцию записи, и указатель косвенной адресации в поле флагов не установлен, данное поле содержит адрес основного запоминающего устройства, подлежащий использованию для передачи данных, и применяется следующее:

- когда поле размера блоков указывает блоки размера 4 Кбайт, указанные биты этого поля содержат определенное значение (например, ноль); в противном случае, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка спецификации основного запоминающего устройства.

- Когда поле размера блоков указывает блоки размера 1 Мбайт, указанные биты

этого поля содержат определенное значение (например, нули); в противном случае, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка спецификации основного запоминающего устройства.

5 Когда поле кода операции имеет смысл и указывает или операцию чтения, или операцию записи, и в поле флагов установлен указатель косвенной адресации, поле содержит адрес основного запоминающего устройства AIDAW или первое из списка AIDAW, подлежащее использованию для передачи данных. В этом случае определенные биты данного поля содержат определенное значение; в противном случае, канальной
10 подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка спецификации основного запоминающего устройства.

Когда установлен флаг BNM, код операции не имеет смысла, и MSB не указывает операцию EADM. Вместо этого, данное поле содержит адрес основного запоминающего
15 устройства следующего MSB, который указывает операцию EADM. В этом случае указанные биты данного поля содержат определенное значение; в противном случае, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка спецификации основного запоминающего устройства.

20 Адрес 272 твердотельного запоминающего устройства: когда поле кода операции имеет смысл, данное поле содержит адрес (например, 64-битный) твердотельного запоминающего устройства, подлежащий использованию для передачи данных или для операций на нем (например, для сброса, освобождения), и применяется следующее:

- когда поле размера блоков указывает блоки размера 4 Кбайт, указанные биты
25 этого поля содержат определенное значение (например, нули); в противном случае, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка спецификации твердотельного запоминающего устройства.

- Когда поле размера блоков указывает блоки размера 1 Мбайт, указанные биты
30 этого поля содержат определенное значение; в противном случае, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка спецификации твердотельного запоминающего устройства.

35 Когда установлен флаг BNM, данное поле является зарезервированным и не проверяется.

Как указано выше, может быть указано косвенное адресное слово данных EADM. Косвенное адресное слово данных EADM (AIDAW) позволяет программе указывать передачу блоков данных между твердотельным запоминающим устройством и блоками
40 основного запоминающего устройства, которые не являются непрерывными. Одно из слов AIDAW или список слов AIDAW указывается посредством одного из блоков MSB, когда в MSB установлен флаг косвенной адресации.

Объем данных, передаваемый единственным AIDAW, указывается в MSB в поле размера блоков. Количество слов AIDAW в списке слов AIDAW представляет собой сумму количества, указанного в MSB полем счетчика блоков, плюс количество слов
45 AIDAW, которые указывают переход к следующему AIDAW. Передачи данных могут обрабатываться в порядках, которые отличаются от таких, указываемых в списке слов AIDAW. Кроме того, передачи данных, указанные несколькими словами AIDAW в списке слов AIDAW, могут обрабатываться параллельно.

AIDAW размещается, например, на границе учетверенного слова. Список слов AIDAW может иметь любую длину, но, в одном примере, он не пересекает границу 4 Кбайт, если для пересечения границы не указан переход к следующему AIDAW (BNA).

Требование того, чтобы AIDAW, которое является целью перехода, было непрерывным с AIDAW, указывающим BNA, отсутствует. Однако программа должна создавать список AIDAW как можно в меньшем количестве блоков размера 4 Кбайт; в противном случае, может проявляться ухудшение производительности.

Согласно фиг.2Е, в одном примере, косвенное адресное слово 280 данных EADM содержит:

10 флаги 282: пример флага содержит:

флаг перехода к следующему AIDAW (BNA): этот флаг, когда он установлен, указывает, что поле адреса основного запоминающего устройства не указывает адрес основного запоминающего устройства, подлежащий использованию для передачи данных. Вместо этого, поле адреса основного запоминающего устройства указывает 15 адрес следующего AIDAW, подлежащего использованию для передачи данных.

Когда установлен флаг BNA, и поле адреса основного запоминающего устройства указывает AIDAW, в котором также установлен флаг BNA, адрес MSB сохраняется в поле неверного адреса MSB в ARSB, адрес исходного AIDAW перехода сохраняется в поле неверного адреса основного запоминающего устройства в ARSB, и в поле кода 20 квалификатора исключительных ситуаций в ARSB указывается ошибка перехода AIDAW.

Если AIDAW заканчивается на границе 4 Кбайт, MSB указывает дополнительные AIDAW, и флаг BNA не установлен, адрес MSB сохраняется в поле неверного адреса MSB в ARSB, адрес AIDAW сохраняется в поле неверного адреса основного запоминающего устройства в ARSB, и в поле кода квалификатора исключительных 25 ситуаций в ARSB указывается ошибка списка слов AIDAW.

Адрес 284 основного запоминающего устройства: когда поле BNA не установлено, данное поле содержит адрес в основном запоминающем устройстве, подлежащий использованию для передачи данных, и применяется следующее:

- Когда поле размера блоков в MSB указывает блоки размера 4 Кбайт, указанные 30 биты этого поля содержат определенное значение (например, ноль); в противном случае, канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка спецификации основного запоминающего устройства.

- Когда поле размера блоков в MSB указывает блоки размера 1 Мбайт, указанные 35 биты этого поля содержат определенное значение (например, нули); в противном случае канальной подсистемой распознается ситуация проверки программы, и в поле кода квалификатора исключительных ситуаций ARSB указывается ошибка спецификации основного запоминающего устройства.

- Когда установлен флаг BNA, данное поле содержит адрес основного запоминающего 40 устройства следующего AIDAW, подлежащего использованию для передачи данных.

[0001] Как описано выше, операции EADM указываются командой Start Subchannel. То есть программа инициирует операции EADM, выдавая команду Start Subchannel, которая указывает субканал ADM-типа и EADM ORB. Выполнение указанной команды передает содержимое EADM ORB в назначенный субканал. EADM ORB содержит 45 спецификацию ключа субканала (используемого для защиты) и адрес AOB, подлежащего использованию.

В одном примере, как проиллюстрировано на фиг.3А, команда 300 Start Subchannel содержит код 302 операции, указывающий функцию Start Subchannel, первый операнд

304, который представляет собой подразумеваемый операнд, расположенный, например, в регистре 1 общего назначения, который содержит идентификатор подсистемы, указывающий субканал ADM-типа, который подлежит запуску; и второй операнд 306, который представляет собой логический адрес EADM ORB. EADM ORB указывает параметры, используемые в управлении функцией запуска. Содержимое EADM ORB помещается в назначенный субканал в ходе выполнения команды Start Subchannel перед установкой кода ситуации. Если выполнение команды Start Subchannel приводит к установке иного кода ситуации, чем код, указывающий успех, содержимое EADM ORB не помещается в назначенный субканал.

Вслед за выполнением команды Start Subchannel для субканала ADM-типа канальная подсистема асинхронно осуществляет функцию запуска для инициирования средством EADM операций EADM. Функция запуска, например, предусматривает:

1. извлечение ARQB из AOB.
2. Выполнение операций EADM так, как это указано ARQB и блоками MSB, которые он обозначает.
3. Условное сохранение информации о завершении в ARSB в AOB.
4. Вынуждение субканала ADM-типа к тому, чтобы он был сделан ожидающим статус, что указывает завершение функции запуска.

В одном варианте осуществления изобретения со ссылкой на фиг.3Б когда выполнена команда Start Subchannel и SID указывает субканал ADM-типа, и второй операнд указывает EADM ORB, на этапе 320 указывается операция EADM. Параметры в EADM ORB передаются в назначенный субканал на этапе 322, и на этапе 324 канальная подсистема запрашивается на выполнение средством EADM функции запуска. Канальная подсистема асинхронно осуществляет функцию запуска путем использования информации в субканале, в том числе информации, переданной в ходе выполнения команды Start Subchannel, для инициализации операций EADM на этапе 326.

Выполнение операции EADM (например, первой операции) включает использование информации из EADM ORB для получения AOB на этапе 328, и использование информации в AOB для получения блока запроса EADM (ARQB) и указания одного или нескольких блоков спецификации перемещения EADM (MSB) на этапе 330. Один или несколько указанных блоков MSB затем подвергаются извлечению из основного запоминающего устройства на этапе 332, и информация, указываемая в блоках MSB, используется для управления запрошенной операцией EADM. Первая операция считается запущенной, когда канальная подсистема предпринимает попытку инициирования передачи данных или попытку операции сброса или освобождения.

Канальная подсистема осуществляет операцию (операции), запрашиваемую в блоке (блоках) MSB, на этапе 334. Данная обработка данных является асинхронной для выполнения команды запуска.

Когда операции EADM, инициированные командой Start Subchannel заканчиваются на этапе 336, канальная подсистема на этапе 338 генерирует ситуации статуса. Генерирование указанных ситуаций на этапе 339 доводится до сведения программы посредством прерывания ввода-вывода. Программа также может запрашивать указанные ситуации путем выполнения команды Test Pending Interruption.

Генерируемые ситуации статуса представляются программе в форме слова статуса субканала EADM (SCSW). Указанное SCSW EADM сохраняется как часть блока ответа на прерывание (IRB) EADM путем выполнения команды Test Subchannel.

[0002] Когда операции EADM, указываемые блоком операций EADM (AOB), заканчиваются, канальная подсистема генерирует первичный и вторичный статусы

прерывания. Операции EADM могут быть завершены командой Clear Subchannel или ненормальной ситуацией, распознанной при осуществлении функции запуска.

Обработка MSB канальной подсистемой для операции перемещения управляет потоком единицы информации в основное запоминающее устройство или из основного запоминающего устройства. Для изменения направления перемещения данных в ходе обработки MSB необходим новый MSB. ARQB указывает число блоков MSB, которые включают запрос.

Каждая операция EADM отображается одним MSB. MSB может указывать любое из следующего:

- передачу блоков данных из основного запоминающего устройства в твердотельное запоминающее устройство.
- Передачу блоков данных из твердотельного запоминающего устройства в основное запоминающее устройство.
- Сброс блоков твердотельного запоминающего устройства.
- Освобождение блоков твердотельного запоминающего устройства, когда установлено средство освобождения EADM.

Если блоки запоминающего устройства, подлежащие передаче для операции перемещения, не являются непрерывными в основном запоминающем устройстве, может использоваться новый MSB, или MSB может использовать косвенную адресацию путем указания списка косвенных адресных слов данных EADM (AIDAW) для указания блоков, не являющихся непрерывными. Поскольку MSB может указывать передачу данных только в одном направлении, когда происходит изменение в направлении передачи данных, должен использоваться новый MSB.

Характеристиками передач данных EADM являются следующие характеристики:

- передачи данных могут обрабатываться вне очереди относительно порядка указанного списка MSB.
- Передачи данных, указываемые несколькими блоками MSB в списке MSB, могут обрабатываться параллельно.
- Передачи данных могут обрабатываться вне очереди относительно порядка указываемого списка AIDAW.
- Передачи данных, указываемые несколькими AIDAW в списке AIDAW, могут обрабатываться параллельно.
- Доступы к основному запоминающему устройству и к твердотельному запоминающему устройству необязательно являются обращениями одиночного доступа и необязательно осуществляются в направлении слева направо, как наблюдается программой и другими ЦП.

- Если две или более операций EADM в настоящий момент действуют и адресуют одни и те же местоположения SCM, местоположение в основном запоминающем устройстве или те и другие местоположения, операции могут осуществляться параллельно, и содержимое из различных операций может чередоваться; однако:

для операций ввода - данные, сохраненные средством EADM в каждый блок основного запоминающего устройства, то есть размер, равный размеру параллелизма блоков SCM, состоит из данных, переданных из твердотельного запоминающего устройства посредством только одной из параллельных операций EADM.

Для операций вывода - каждый блок твердотельного запоминающего устройства, который находится на границе и имеет размер, равный размеру параллелизма блоков SCM, содержит данные, указанные посредством только одной из параллельных операций EADM.

Вышесказанное верно независимо от того, указываются ли операции EADM единичным АОВ и являются ли обрабатываемыми одним и тем же вариантом функции запуска, или операции EADM указываются различными блоками АОВ и являются обрабатываемыми различными вариантами функции запуска.

- 5 - Когда для субканала являются действующими операции EADM, является непредсказуемым то, будут ли наблюдаться средством EADM изменения, сделанные программой в ARQB, блоках MSB, словах AIDAW и передаваемых данных, связанные с действующей операцией.

- 10 Когда все блоки, указанные всеми блоками MSB, указанными посредством АОВ, были переданы или сброшены, или освобождены, субканал генерирует статус, который сохраняется в субканале, и запрашивает для субканала ADM-типа прерывание ввода-вывода.

- 15 Завершение операции EADM обычно указывается комбинированными ситуациями статусов окончания работы канала и окончания работы устройства. Данная комбинация статусов отображает комбинацию первичного и вторичного статусов, указывающую, что после сброса статуса субканал является доступным для другой функции запуска.

Как описано выше, MSB может указывать операцию передачи данных, операцию сброса или операцию освобождения, каждая из которых описывается ниже.

- 20 Когда запрашивается операция перемещения, один или несколько блоков данных перемещаются между основным запоминающим устройством и твердотельным запоминающим устройством. Например, для операции чтения данные SCM получают из SSD, которые доставляют содержимое, обозначаемое указанным адресом SCM, и затем это содержимое сохраняется в основную память. Для операции записи процесс является обратным. Сохранение осуществляет адаптер (адаптеры), управляющий SSD.
- 25 Более подробно, для осуществления операции перемещения системное программно-аппаратное обеспечение, в первую очередь, транслирует данный адрес SCM в адрес адаптера (например, адрес логического тома, в котором логический том содержит один или несколько SSD). Например, используется таблица трансляции, которая коррелирует адрес SCM с адресом адаптера. Системное программно-аппаратное обеспечение затем
- 30 подает одному или нескольким адаптерам ввода-вывода одну или несколько соответствующих адаптерных команд перемещения (например, чтения или записи). Адаптерная команда перемещения содержит адрес основного запоминающего устройства, адрес адаптера и размер передачи. Затем адаптер использует другую таблицу трансляции для определения местоположения одного или нескольких физических адресов
- 35 SSD, которые соответствуют адресу адаптера. Адаптер ввода-вывода выполняет операцию перемещения или путем извлечения данных из основного запоминающего устройства и их сохранения на SSD, или путем извлечения данных с SSD и их сохранения в основную память. Дополнительные подробности также предоставлены ниже в разделе «Управление программно-аппаратным обеспечением твердотельного запоминающего
- 40 устройства».

Когда осуществляется операция сброса, указанные приращения твердотельного запоминающего устройства сбрасываются путем установки содержимого на ноль.

- 45 Дополнительно, когда осуществляется операция освобождения, на SSD, содержащий соответствующие адреса SCM, может подаваться одна или несколько стандартных команд TRIM. Команда TRIM позволяет программе подсказывать использование блоков, позволяющее лучше выполнять очистку от ненужных данных посредством SSD. Команда TRIM позволяет операционной системе информировать SSD о том, какие блоки запоминающего устройства больше не считаются находящимися в употреблении

и могут быть стерты.

Операция EADM может быть преждевременно завершена командой Clear Subchannel. Выполнение команды Clear Subchannel завершает выполнение АОВ в субканале, сбрасывает субканал от указателей выполнения АОВ и асинхронно осуществляет функцию сброса. Когда осуществляется функция сброса, перед тем, как субканал становится ожидающим статус, передача данных завершается, и объем переданных данных является непредсказуемым. Выполнение функции сброса не приводит к генерированию статуса, но вынуждает канальную подсистему сделать прерывание ввода-вывода ожидающим.

В одном варианте осуществления изобретения, со ссылкой на фиг.3 В, команда 350 Clear Subchannel содержит код 352 операций, указывающий функцию Clear Subchannel. Субканал, подлежащий сбросу, указывается словом идентификации подсистемы, например, в регистре общего назначения.

Функция сброса для субканала ADM-типа включает:

1. Проверку того, что текущая передача данных завершена.
2. Модификацию полей в субканале и, условно, в ARSB. Например, слово статуса субканала модифицируется так, чтобы оно указывало функцию сброса в поле управления функциями и в поле управления действиями. ARSB может быть модифицирован для отражения любых обнаруживаемых ошибок.
3. Вызов того, что субканал делается ожидающим статус, что указывает на завершение функции сброса.

Также могут выдаваться другие команды, которые могут указывать субканал ADM-типа, в том числе команды Test Subchannel, Modify Subchannel и Store Subchannel, каждая из которых описана ниже.

Со ссылкой на фиг.3Г, в одном примере, команда 360 Test Subchannel включает, например, код 362 операций, указывающий функцию проверки субканала; первый операнд 364, который представляет собой подразумеваемый операнд, расположенный, например, в регистре 1 общего назначения, который содержит слово идентификации подсистемы, указывающее субканал, который подлежит проверке; и второй операнд 366, который представляет собой логический адрес блока ответа на информацию (IRB), в котором хранится информация.

Когда выполняется команда Test Subchannel, указывающая субканал ADM-типа, субканал является ожидающим статус, и информация сохранена в указанном IRB (блоке ответа на прерывание) EADM, устанавливается код указанной ситуации. Когда субканал не является ожидающим статус, и информация о статусе сохранена в указанном IRB EADM, устанавливается код определенной ситуации. Когда субканал не предоставлен или не является деблокированным, никакие действия не предпринимаются.

В одном примере IRB EADM содержит слово статуса субканала (SSW) и расширенное слово статуса (ESW), а также расширенное слово управления, которые могут обеспечивать дополнительную зависящую от модели информацию, описывающую ситуации, которые могут существовать в средстве. Каждое из указанных слов в дальнейшем описывается ниже после описания различных команд, которые могут указывать субканал ADM-типа.

Согласно фиг.3Д, в одном варианте осуществления изобретения команда 370 Modify Subchannel содержит код 372 операции, указывающий функцию Modify Subchannel, первый операнд 374, который представляет собой подразумеваемый операнд, расположенный, например, в регистре 1 общего назначения, который содержит слово идентификации подсистемы, указывающее субканал, который подлежит модификации;

и второй операнд 376, который представляет собой логический адрес блока информации о субканале (SCHIB), связанный с субканалом.

Когда выполняется команда Modify Subchannel, указывающая субканал ADM-типа, и информация из указанного блока информации о субканале (SCHIB) размещена в субканале, устанавливается специальный код ситуации. Когда субканал является ожидающим статуса, никакие действия не предпринимаются, и устанавливается код определенной ситуации. Когда субканал занят для функции запуска или сброса, никакие действия не предпринимаются.

Согласно фиг.3Е описан один пример команды Store Subchannel. В одном примере команда 380 Store Subchannel содержит код 382 операции, идентифицирующий функцию Store Subchannel; первый операнд 384, который представляет собой подразумеваемый операнд, расположенный, например, в регистре 1 общего назначения, который содержит слово идентификации подсистемы, указывающее субканал, для которого сохраняется информация; и второй операнд 386, который представляет собой логический адрес SCHIB.

Когда выдается команда Store Subchannel, указывающая субканал ADM-типа, и SCHIB является сохраненным, устанавливается код указанной ситуации. Когда указанный субканал не предусматривается в канальной подсистеме, никакие действия не предпринимаются.

Один пример блока информации о субканале для субканала ADM-типа (SCHIB EADM) описывается со ссылкой на фиг.4А. В одном примере SCHIB 400 EADM содержит зависящую от модели область 401, которая содержит зависящую от модели информацию. Дополнительно SCHIB 400 содержит управляющее слово 402 управления трактами (PMCW) и слово 404 статуса субканала (SCSW), каждое из которых будет описано ниже.

В одном примере PMCW 402 EADM содержит, например, следующие поля, как показано на фиг.4Б:

Параметр 410 прерывания: данное поле содержит параметр прерывания, который хранится в коде прерывания ввода-вывода. Параметр прерывания может быть установлен на любое значение посредством команд Start Subchannel и Modify Subchannel.

Исходное значение поля параметра прерывания в субканале равно нулю.

Подкласс 412 прерывания (ISC): данное поле содержит ряд битов, которые представляют собой беззнаковое двоичное целое число в указанном интервале, которое соответствует позиции бита маски подкласса прерывания ввода-вывода в указанном управляющем регистре для каждого ЦП в конфигурации. Установка бита маски в управляющем регистре ЦП управляет распознаванием запросов прерывания, относящихся к субканалу этим ЦП. ISC может быть установлено на некоторое значение командой Modify Subchannel. Исходное значение поля ISC в субканале равно, например, нулю.

Поле 414 деблокирования (E): данное поле, когда оно установлено, указывает, что субканал деблокирован для всех функций EADM.

Тип 416 субканала (ST): данное поле указывает тип субканала. В зависимости от модели и конфигурации может предусматриваться один или несколько следующих типов субканалов: субканал ввода-вывода или субканал ADM.

Такое значение данного поля определяется, когда субканал является сконфигурированным и не может быть изменен командой Modify Subchannel.

Когда команда Modify Subchannel выполняется и указывает субканал ADM-типа, ST предназначено для указания субканала ADM; в противном случае распознается исключительная ситуация операнда.

Согласно фиг.4А блок информации о субканале также содержит слово 404 статуса субканала. Слово статуса субканала EADM (SCSW) обеспечивает для программы указатели, которые описывают статус субканала ADM-типа и связанные операции EADM. В одном примере, как показано на фиг.4В, слово 404 статуса субканала содержит:

ключ 420 субканала (Key): когда в поле управления функциями (описываемом ниже) установлен индикатор функции запуска EADM, данное поле содержит ключ доступа к запоминающему устройству, используемый канальной подсистемой. Указанные биты идентичны ключу, указываемому в EADM ORB при выполнении команды Start Subchannel.

Формат 422 расширенного слова статуса (L): когда установлен индикатор ожидания статуса в поле элементов управления статусами (описываемом ниже), данное поле, если оно установлено, указывает, что был сохранен ESW с форматом 0. ESW с форматом 0 сохраняется, когда командой Test Subchannel сбрасывается ситуация прерывания, содержащая любой из следующих указателей:

- Проверка программы
- Проверка защиты
- Проверка канальных данных
- Проверка управления каналом
- Проверка средства расширенного асинхронного перемещения данных (EADMF).

Код 424 отложенной ситуации (CC): когда индикатор функции запуска EADM установлен, и индикатор ожидания статуса также установлен, данное поле указывает общую причину того, что субканал являлся ожидающим статус, когда выполнялись команды Test Subchannel или Store Subchannel. Код отложенной ситуации имеет смысл, когда субканал является ожидающим статус с любой комбинацией статусов, и только тогда, когда установлен индикатор функции запуска в поле управления функциями в SCSW.

Код отложенной ситуации, если он установлен, используется для указания того, встречались ли какие-либо ситуации, которые препятствовали тому, чтобы субканал становился действующим субканалом, в то время как субканал являлся ожидающим запуск.

Примеры кодов отложенной ситуации включают следующее:

- Было представлено нормальное прерывание ввода-вывода.
- В SCSW EADM присутствует статус, который был сгенерирован канальной подсистемой для ситуаций, которые препятствовали тому, чтобы функция запуска EADM была успешно инициирована. То есть субканал не перешел в действующее состояние субканала.

Формат 426 CCW (F): когда установлен индикатор функции запуска EADM, данное поле сохраняется как определенное значение.

Предварительное извлечение 428 (P): когда установлен индикатор функции запуска EADM, данное поле сохраняется как определенное значение.

Расширенное управление 430 (E): Данное поле, когда оно установлено, указывает, что зависящая от модели информация сохранена в расширенном управляющем слове EADM (ECW).

Управление 432 функциями (FC): поле управления функциями указывает функции EADM, которые указаны в субканале. Примеры функций включают следующее:

- функцию запуска EADM: когда она установлена, она указывает, что функция запуска EADM была запрошена, и она или является ожидающей, или в настоящий момент выполняется в субканале ADM-типа. Функция запуска EADM указывается в субканале,

когда для команды Start Subchannel установлен код успешной ситуации. Функция запуска EADM в субканале сбрасывается, когда выполняется команда Test Subchannel, а субканал является ожидающим статус. Функция запуска EADM в субканале также сбрасывается в ходе выполнения функции Clear Subchannel.

- 5 - Функцию сброса EADM: когда она установлена, указывает, что была запрошена функция сброса EADM, и она или является ожидающей, или в настоящий момент выполняется в субканале ADM-типа. Функция сброса EADM указывается в субканале, когда для команды Clear Subchannel установлен код успешной ситуации. Когда выполняется команда Test Subchannel, указатель функции сброса EADM в субканале
10 сбрасывается, и субканал является ожидающим статус.

Управление 434 действиями (AC): поле управления действиями указывает текущий прогресс функции EADM, ранее принятой в субканале.

- Все ситуации, которые отображаются битами в поле управления действиями, сбрасываются в субканале ADM-типа, когда выполняется команда Test Subchannel, и
15 субканал является ожидающим статус.

Примеры действий включают следующее:

- ожидание запуска: при установке указывает, что субканал является ожидающим запуск. Канальная подсистема может находиться или может не находиться в процессе осуществления функции запуска EADM. Субканал становится ожидающим запуск,
20 когда для команды Start Subchannel установлен код успешной ситуации. Субканал остается ожидающим запуск, когда осуществляется функция запуска EADM, и канальная подсистема определяет существующие ситуации, которые препятствуют осуществлению извлечения ARQB.

- Субканал более не является ожидающим запуск, когда происходит любое из
25 следующего:

- канальная подсистема предпринимает попытку инициирования первой передачи данных, указанной посредством АОВ.
- Выполняется команда Clear Subchannel.
- Команда Test Subchannel сбрасывает ситуацию статуса в субканале.

- 30 Ожидание сброса: когда оно установлено, субканал является ожидающим сброс. Канальная подсистема может находиться или может не находиться в процессе осуществления функции сброса EADM. Субканал становится ожидающим сброс, когда для команды Clear Subchannel установлен код указанной ситуации.

- Субканал больше не является ожидающим сброс, когда происходит любое из
35 следующего:

- Осуществляется функция сброса EADM.
- Команда Test Subchannel сбрасывает только ситуацию ожидания статуса.

- Субканал действует: когда оно установлено, указывает, что субканал ADM-типа является действующим субканалом. Субканал ADM-типа считается действующим
40 субканалом, когда канальная подсистема предпринимает попытку инициировать первую передачу данных, указываемую посредством АОВ, или осуществляет первую операцию (в зависимости от того, что происходит в первую очередь).

Субканал больше не является действующим субканалом, когда происходит любое из следующего:

- 45 - Субканал становится ожидающим статус.
- Выполняется команда Clear Subchannel.

Управление 436 статусами (SC): поле управления статусами снабжает программу указателем обобщенного уровня для ситуации прерывания, описываемой информацией

в полях статуса субканала и статуса устройства. В результате ситуаций в субканале может устанавливаться больше одного индикатора управления статусами.

Примеры элементов управления статусами включают следующее:

Предупреждающий статус: когда он установлен, указывает, что существует ситуация предупреждающего прерывания. Ситуация предупреждающего прерывания распознается, когда в субканале присутствует предупреждающий статус. Предупреждающий статус генерируется канальной подсистемой в любой из следующих ситуаций:

- Субканал ADM-типа является ожидающим запуск, и ситуация статуса препятствует инициированию первой передачи данных.

- Субканал является действующим субканалом, и ненормальная ситуация, которая была сообщена как статус субканала, завершила операции EADM.

Когда выполняется команда Test Subchannel или Clear Subchannel, предупреждающий статус в субканале сбрасывается.

Первичный статус: когда он установлен, указывает, что существует ситуация первичного прерывания. Ситуация первичного прерывания распознается, когда в субканале присутствует первичный статус. Ситуация первичного прерывания представляет собой ситуацию обусловленного прерывания, которая, когда она сопровождается ситуацией вторичного прерывания, указывает завершение функции запуска EADM в субканале.

Когда выполняется команда Test Subchannel или Clear Subchannel, ситуация первичного прерывания в субканале сбрасывается.

Вторичный статус: когда он установлен, указывает существование ситуации вторичного прерывания. Ситуация вторичного прерывания распознается, когда в субканале присутствует вторичный статус. Ситуация вторичного прерывания представляет собой ситуацию обусловленного прерывания, которая, когда она сопровождается ситуацией первичного прерывания, указывает завершение функции запуска EADM в субканале.

Когда выполняется команда Test Subchannel или Clear Subchannel, ситуация вторичного прерывания в субканале сбрасывается.

Ожидание статуса: когда оно установлено, указывает, что субканал является ожидающим статус, и что доступна информация, описывающая причину ситуации прерывания. Когда выполняется команда Test Subchannel, сохраняющая SCSW EADM с набором битов ожидания статуса, все указатели SCSW EADM в субканале сбрасываются, помещая субканал в состояние бездействия. Ситуация ожидания статуса также сбрасывается в субканале в ходе выполнения команды Clear Subchannel.

Когда установлено ожидание статуса, ARSB завершил все доступы к основному запоминающему устройству и твердотельному запоминающему устройству для субканала.

Статус 438 субканала: ситуации статуса субканала ADM-типа обнаруживаются и сохраняются канальной подсистемой в поле статуса субканала. Поле статуса субканала является значимым, когда субканал является ожидающим статус. За исключением ситуаций, вызванных сбоями оборудования, статус субканала может возникать только тогда, когда канальная подсистема включена в обработку функции EADM.

Примеры ситуаций статуса включают следующее:

- Проверку программы: проверка программы происходит, когда канальной подсистемой обнаруживаются ошибки программирования.

- Проверку защиты: проверка защиты происходит, когда канальная подсистема

предпринимает попытку доступа к запоминающему устройству, который запрещен механизмом защиты. Защита распространяется на извлечение ARQB, блоков MSB, слов AIDAW, данных, подлежащих передаче в твердотельное запоминающее устройство, на сохранение информации в ARSB и данные, передаваемые из твердотельного

5 запоминающего устройства.

- Проверку канальных данных: проверка канальных данных указывает, что в отношении извлечения данных из основного запоминающего устройства или сохранения данных в основное запоминающее устройство была обнаружена неисправленная ошибка запоминающего устройства.

10 - Проверку управления каналом: проверка управления каналом указывает, что в отношении извлечения или сохранения AOB, блоков MSB или слов AIDAW была обнаружена неисправленная ошибка запоминающего устройства, или что канальная подсистема столкнулась с машинным сбоем, и сбой повлиял на операции EADM.

- Проверку средства расширенного асинхронного перемещения данных (EADM):
15 проверка средства EADM указывает, что средством EADM была обнаружена ошибка в отношении передачи данных в твердотельное запоминающее устройство или из твердотельного запоминающего устройства, или в отношении осуществления операции на твердотельном запоминающем устройстве.

Когда канальной подсистемой распознается ситуация проверки программы, проверки
20 защиты, проверки канальных данных, проверки управления каналом или проверки средства EADM, операции EADM завершаются, и канал делается ожидающим статус с первичным, вторичным и предупреждающим статусом.

Адрес 440 блока операций EADM: данное поле содержит адрес блока операций EADM.

25 Статус 442 устройства: данное поле содержит окончание работы устройства или окончание работы канала.

Когда формат 422 ESW в слове статуса субканала установлен, и субканал является
ожидающим статуса, предусматривается расширенное слово статуса субканала EADM (ESW EADM), которое указывает дополнительную информацию о субканале ADM-
30 типа.

В одном примере, со ссылкой на фиг.4Г, расширенное слово 450 статуса субканала EADM содержит следующее:

Отключение 452 субканала EADM: отключение субканала EADM включает, в одном
примере:

35 - расширенные флаги статусов (ESF): поле, биты которого, если они присутствуют, указывает, что канальной подсистемой была обнаружена ошибка.

Примеры расширенных флагов статусов включают:

проверку ключа: когда она установлена, указывает, что канальная подсистема
обнаружила недостоверный код проверки блока (CBC) на связанном ключе хранения
40 при обращении к данным в блоке операций EADM (AOB), в блоке спецификации перемещения EADM (MSB) или в косвенном адресном слове данных EADM (AIDAW).

- Достоверность адреса AOB: когда она установлена, указывает, что адрес, хранящийся в поле адреса AOB в SCSW, применим для целей восстановления.

Расширенное слово 454 отчета EADM, которое содержит, например:

45 индикатор ошибки блока операций EADM (B), который, когда он установлен, указывает, что статус исключительной ситуации, сохраненный в SCSW EADM, связан с указанным блоком операций EADM (AOB); и индикатор сохраненного блока ответа EADM (R), который, когда он установлен, указывает, что блок ответа EADM (ARSB)

сохранен.

Когда установлен расширенный индикатор управления SCSW и индикатор формата расширенного слова статуса SCSW, расширенное слово управления EADM обеспечивает дополнительную информацию о зависящей от модели сущности, которая описывает ситуации, которые могут существовать в средстве EADM.

Кроме того, для субканалов ADM-типа могут сообщаться следующие слова отчета о каналах (CRW): инициализированы установленные параметры субканала; восстановлены установленные параметры субканала; субканалы доступны; ожидание информации о событиях в субканалах.

Необусловленные события и сбои, которые происходят в средстве EADM, могут сообщаться посредством CRW ожидания информации о событиях в субканалах.

Выше было подробно описано средство расширенного асинхронного перемещения данных, используемое для перемещения блоков данных между основным запоминающим устройством и твердотельным запоминающим устройством и для других операций на твердотельном запоминающем устройстве. В одном варианте осуществления изобретения информацию о средстве EADM и твердотельном запоминающем устройстве получают с использованием команды EADM. В частности, поскольку твердотельное запоминающее устройство не является непосредственно доступным, обеспечивается средство для определения того, выделено ли твердотельное запоминающее устройство, и, если оно выделено, для получения информации касательно конфигурации. В частности, в одном примере, предусматривается средство для связи с управляющей программой (например, с операционной системой) о том, выделено или нет твердотельное запоминающее устройство, и если оно выделено, то в каком объеме и в каких местоположениях. Возможность определения доступного твердотельного запоминающего устройства в данном описании называется отысканием, и один пример функции отыскания предусматривается командой Store Storage Class Memory (SCM) Information (SSI).

Команда Store Storage Class Memory Information (SSI) используется для получения информации о твердотельном запоминающем устройстве и о средстве расширенного асинхронного перемещения данных. Команда SSI выдает следующую информацию, полученную, например, из канальной подсистемы. Указанная информация более подробно описывается ниже:

1. Характеристики средства EADM, включающие:

- максимальное число блоков спецификации перемещения (MSB), приходящихся на один АОВ.

- Максимальное число блоков, приходящихся на один MSB.

2. Характеристики твердотельного запоминающего устройства, включающие:

- размер приращений SCM.

- Список приращений адресов SCM в пределах адресного пространства SCM.

- Зависящий от модели максимальный адрес SCM.

Выполнение команды Store SCM Information, которое является синхронным, не изменяет какую-либо информацию, содержащуюся в канальной подсистеме.

Дополнительные подробности, относящиеся к команде SSI, описываются согласно фиг.5А-5Г. В начале, согласно фиг.5А, в одном варианте осуществления изобретения блок 500 запроса команды для команды Store SCM Information содержит, например: длину 502: значение, указывающее длину блока запроса команды.

Код 504 команды: значение, которое указывает код команды для команды Store SCM Information.

Формат 506 (FMT): значение, которое указывает формат блока запроса команды.

Маркер 508 продолжения: значение, которое может запрашивать точку продолжения для возобновления относительно предшествующего ответа, который не был завершен. Если значение маркера продолжения равно нулю, осуществляется запуск с начала.

5 Если значение маркера продолжения является ненулевым и нераспознанным, осуществляется запуск с начала.

Один вариант осуществления блока 520 ответа на команду SSI описывается согласно фиг.5Б, и содержит, например:

длину 522: значение, указывающее длину в байтах блока ответа на команду.

10 Код 524 ответа: значение, которое описывает результаты попытки выполнения команды SSI. Значение кода ответа обуславливает длину блока ответа. Например, если сохраняется выбранный код ответа, длина указывает $96+N \times 16$ байт, где N - количество записей в списке адресов твердотельного запоминающего устройства, которые описываются ниже. В одном примере, N находится в диапазоне $1 \leq N \leq 248$.

15 Формат 526 (FMT): значение, которое указывает формат блока ответа на команду. Значение этого поля равно, например, нулю.

RQ 528: значение квалификатора ответа, как определено ниже: квалификации ответа не существует.

20 Указанный маркер продолжения не распознается и обрабатывается так, будто бы был указан ноль.

Поле 530 максимального счетчика блоков, приходящихся на один MSB (MBC): значение, которое указывает максимальное значение, которое может использоваться в поле счетчика блоков блока спецификации перемещения (MSB).

25 Максимальный адрес SCM (MSA) 532: значение, которое указывает зависящий от модели максимальный адрес SCM. Оно представляет собой адрес SCM последнего байта в наивысшем адресуемом приращении SCM.

Размер 534 приращений SCM (IS): значение, которое отображает размер каждого приращения SCM в списке адресов SCM и представляет собой, например, степень двойки.

30 Максимальный счетчик 536 блоков MSB (MMC): значение, которое указывает максимальное число блоков спецификации перемещения (MSB), которое может быть указано в блоке операций EADM (AOB).

Максимальные конфигурируемые приращения SCM (MCI) 538: значение, которое представляет собой максимальное количество приращений SCM, которое может конфигурироваться в запрашиваемую конфигурацию.

35 В одном варианте осуществления изобретения, MCI не превышает $2^{(64-IS)}$. Например, для размера приращения SCM 16 Гбайт $MCI \leq 2^{(64-34)}$, поскольку все приращения SCM по 16 Гбайт должны быть адресуемыми в пределах ограничения адресации 64-битового адреса. Кроме того, $((MCI+1) \times IS) - 1$ не превышает зависящий от модели максимальный адрес SCM.

40 Количество конфигурируемых приращений SCM (NCI) уменьшает общий размер (TS), который может указываться, когда выполняется описываемая ниже успешная команда Configure Storage Class Memory, так что $TS \leq (MCI - NCI)$. Однако, основываясь на пропускной способности системы в целом и выделениях, уже сделанных для других конфигураций, приращения SCM в инициализированном состоянии могут быть или 45 могут не быть способны полностью удовлетворять запрос на в противном случае достоверное конфигурирование вплоть до предела MCI для запрашиваемого конфигурирования.

Поле 540 общего количества инициализированных приращений SCM в CPC: значение,

которое отображает количество приращений SCM в инициализированном состоянии для системы (например, для центрального процессорного комплекса (CPC)). Если система логически разбита, оно представляет собой количество приращений, доступное в целом для выделения разделам/разделами. Поле 542 общего количества
 5 неинициализированных приращений SCM в CPC: значение, которое отображает количество приращений SCM в неинициализированном состоянии для системы (например, для CPC).

Размер 544 блока измерения SCM: значение, которое представляет собой размер блока (BS) в байтах для блока измерения SCM. В одном примере оно представляет
 10 собой степень 2, и максимальный размер блока измерения SCM составляет, например, 4096 байт.

Максимальное количество частей 546 ресурса SCM: значение, которое представляет собой максимальное количество частей ресурса (RP) SCM (например, адаптеров ввода-вывода) на CPC. Каждое приращение SCM связано с одной из частей ресурса SCM.

15 Каждый ресурс SCM (например, один или несколько адаптеров ввода-вывода и один или несколько SSD) содержит одну или несколько частей. Максимальное количество частей ресурса SCM составляет, например, 509.

В одном примере термин «часть ресурса» определен для использования для получения информации измерений, относящейся к твердотельному запоминающему устройству.

20 Каждое приращение SCM может распределяться по нескольким адаптерам, и каждый адаптер содержит для передачи некоторые данные об использовании/измерении. Так, каждый выдаваемый блок измерения идентифицируется кортежем, состоящим из идентификатора приращения плюс идентификатор ресурса.

Размер 548 единицы данных SCM: зависящее от модели значение, которое указывает
 25 количество байтов, которые включены в единицу данных SCM. В одном примере единица данных определяется для использования при получении информации измерений, относящейся к твердотельному запоминающему устройству. Число, которое сообщается, представляет собой число единиц данных, а не число байтов.

Маркер 550 продолжения: зависящее от модели значение, посредством которого
 30 последующая выдача команды SSI может продолжаться в точке продолжения, отображаемой маркером. Содержимое маркера продолжения зависит от модели.

Список 552 адресов твердотельного запоминающего устройства: начало списка адресов SCM. Когда сохраненный код ответа представляет собой предварительно
 35 определенное значение, сохраняется ряд записей списка адресов SCM (SALE) (например, сохраняется (Length - 96)/16 записей SALE).

В одном примере сохраняемое количество записей SALE зависит от количества приращений SCM в конфигурации запросчика, от статуса каждой из них при выполнении команды, и от модели канальной подсистемы. Сохраняется нулевое или большее
 40 количество записей SALE, и фактическое сохраненное количество определяется, например, путем вычитания 96 из размера блока ответа (длина), а затем деления этого результата на 16.

Каждая запись в списке адресов SCM (SALE) отображает одно приращение SCM, которое занимает диапазон адресов SCM. Начальный адрес SCM приращения SCM, отображаемый посредством SALE, содержится в SALE и в адресе SCM первого байта
 45 соответствующего приращения SCM. Конечный адрес вычисляется, в одном примере, путем добавления размера приращения SCM в байтах к начальному адресу SCM и затем вычитания единицы. Он представляет собой адрес SCM последнего байта приращения SCM. Твердотельное запоминающее устройство, отображаемое посредством SALE,

представляет собой непрерывный набор местоположений байтов SCM, который, в одном варианте осуществления изобретения, начинается на естественной границе 2^{1S} байт.

SALE сохраняется, когда соответствующее приращение SCM находится в сконфигурированном состоянии, и в блоке ответа доступно пространство для SALE. Если пространство в списке адресов SCM блока ответа исчерпано, значение сохраняется в маркере продолжения, и выполнение завершается со специальным кодом ответа.

Две или большее количество записей SALE сохраняются в порядке возрастания их адресов SCM.

[0003] Один вариант осуществления SALE описывается согласно фиг.5B. В одном примере SALE 552 содержит, например:

адрес 560 SCM (SA): значение, которое представляет собой начальный адрес SCM байта 0 соответствующего приращения SCM в адресном пространстве SCM,

выровненного по естественной границе, определяемой размером приращения SCM (2^{1S} байт).

Атрибут 562 сохраняемости (P): значение, которое указывает текущее правило сохраняемости, применимое к приращению SCM. Правилу сохраняемости следует любое местоположение в пределах приращения SCM. Возможные правила сохраняемости включают:

правило 1 - сохранять данные при отключении питания.

Правило 2 - сохранять данные до сброса по включению питания или IML.

Состояние 564 операций: значение, которое указывает состояние операций приращения твердотельного запоминающего устройства, отображаемого посредством SALE. Состояние операций достоверно только тогда, когда связанное приращение SCM находится в сконфигурированном состоянии.

Примеры состояний операций включают:

рабочее (Op): твердотельное запоминающее устройство, отображаемое посредством SALE, доступно для всех операций ввода-вывода. Вход в рабочее состояние происходит при успешном конфигурировании и может происходить повторно при выходе из состояния временной или постоянной ошибки.

Временная ошибка (TE): твердотельное запоминающее устройство, отображаемое посредством SALE, не доступно ни для каких операций ввода-вывода. Состояние данных недостоверно, но содержимое данных сохранено при переходе из рабочего состояния к временной ошибке. Вход в состояние временной ошибки происходит из рабочего состояния, когда отсутствует доступ к приращению SCM.

Постоянная ошибка (PE): твердотельное запоминающее устройство, отображаемое посредством SALE, не доступно ни для каких операций ввода-вывода. Состояние данных недостоверно, и данные потеряны. Вход в состояние постоянной ошибки происходит из состояния временной ошибки или из рабочего состояния, когда распознается неисправимая ошибочная ситуация.

Когда операция завершается с указателем постоянной ошибки, установленным в коде квалификатора исключительных ситуаций блока ответа EADM, по меньшей мере это соответствующее приращение SCM вошло в состояние постоянной ошибки. Однако в состояние постоянной ошибки могло войти больше одного приращения SCM.

Когда приращение SCM не находится в рабочем состоянии, операция ввода-вывода, которая обращается к местоположению в приращении, распознает проверку средства расширенного асинхронного перемещения данных или с временной, или с постоянной ошибкой, установленной в коде квалификатора исключительных ситуаций блока ответа

EADM.

Состояние 566 данных: значение, которое указывает состояние данных содержимого приращения твердотельного запоминающего устройства, отображаемого посредством SALE. Состояние данных является достоверным, когда связанное приращение SCM находится в сконфигурированном и рабочем состояниях.

Примеры состояний данных включают:

установлено на ноль - все содержимое приращения SCM представляет собой нули.

Достоверно - содержимое приращения SCM представляет собой накопление всех успешных операций, относящихся к типу записи. Еще не записанные местоположения в приращении остаются или установленными на ноль, или непредсказуемыми.

Непредсказуемое - содержимое приращения SCM перед любой операцией ввода-вывода, относящейся к типу записи, неизвестно. После того, как осуществлена одна или несколько операций, относящихся к типу записи, содержимое данных других незаписанных местоположений остается непредсказуемым, даже если успешно записанные местоположения приводят к состоянию данных приращения SCM, которое становится достоверным.

Переход или от установленного на ноль, или от непредсказуемого состояния в достоверное состояние происходит с первой успешной записью. Из-за любой разности между размером записанных данных и размером целевого приращения SCM, изменение в достоверное состояние не описывает фактическую ситуацию для любого еще не записанного местоположения данных. Указанное местоположение, еще не доступное для записи, по-прежнему эффективно описывается как установленное на ноль или непредсказуемое.

Ранг 568: значение, которое указывает концептуальное качество приращения твердотельного запоминающего устройства, отображаемого посредством SALE. Ранг является достоверным только тогда, когда связанное приращение SCM находится в сконфигурированном и рабочем состояниях. Нулевое значение означает, что ранга не существует. Ненулевое значение в указанном диапазоне означает, что ранг существует. В данном примере значение ранга, равное единице, представляет собой высший, или лучший, ранг. Значение ранга, равное пятнадцати, представляет собой низший, или худший, ранг. При прочих равных условиях, SCM с более высоким рангом должна являться предпочтительной перед SCM с менее высоким рангом.

R 570: данное поле указывает, что приращение SCM распознает операцию освобождения. С этим связаны следующие поведения:

1. Освобожденный блок является первым подлежащим записи перед тем, как он будет прочитан, в противном случае, если чтение предшествует записи, распознается ошибка операции чтения. При такой ошибке приращение SCM остается в рабочем состоянии.

2. При начальной конфигурации состояние данных установлено на ноль.

3. Программа может осуществлять специальную операцию, называемую освобождением, которая помещает указанный блок в освобожденную ситуацию.

ID 572 ресурса: ненулевое значение представляет собой идентификатор ресурса (RID) для ресурса, который обеспечивает приращение SCM, отображаемое посредством SALE. Когда RID равен нулю, не указан ни один ID ресурса. В одном из частных примеров RID отображает столько адаптеров и SSD, сколько обеспечивают для приращения SCM запоминающее устройство. Поскольку могут применяться некоторые алгоритмы RAID, или чередование с целью улучшения производительности путем допуска параллельных операций ввода-вывода на нескольких адаптерах/SSD, RID может отображать составную

сущность.

Ниже описываются дополнительные подробности, относящиеся к состояниям конфигурации твердотельного запоминающего устройства и к состояниям операций и данных списка адресов твердотельного запоминающего устройства.

5 Со ссылкой, в первую очередь, на ФИГ 6А описываются состояния конфигурации и события/действия, которые приводят к переходам в этих состояниях. Как показано, состояния SCM представляют сконфигурированное состояние, состояние простоя и зарезервированное состояние. SCM может помещаться в состояние простоя из зарезервированного состояния, а затем из состояния простоя в сконфигурированное состояние. Из сконфигурированного состояния SCM может быть деконфигурирована и введена в зарезервированное состояние.

10 Согласно фиг.6Б показаны состояния операций и события, которые приводят к переходам в указанных состояниях. Для того чтобы стать сконфигурированным, приращение SCM должно находиться в состоянии простоя, и при успешном завершении действия по конфигурированию оно находится в рабочем состоянии. Первая запись в приращение SCM в установленном на ноль состоянии переводит его в достоверное состояние. Промежуточное отключение питания, а затем включение питания на приращении SCM, которое не указано, как обладающее сохраняемостью по правилу 1, переводит приращение SCM в непредсказуемое состояние.

20 Ошибка (E) может вызывать переход в состояние временной ошибки (TE) или в состояние постоянной ошибки (PE) в зависимости от зависящей от модели специфики ошибки. Приобретение (A) соединяемости может приводить к переходу из состояния временной ошибки в рабочее (Op) состояние. Деконфигурирование приращения SCM может происходить независимо от его состояния операций.

25 Фиг.6Б также иллюстрирует состояния данных при нахождении в рабочем состоянии согласно тому, каким образом происходил вход в рабочее состояние. Состояние данных является достоверным и распространяется на соответствующее приращение SCM, когда оно является сконфигурированным и находится в рабочем состоянии. Достоверными состояниями данных являются: установленное на ноль, непредсказуемое и достоверное состояния. Следующее представляет собой возможные состояния данных при различных входах в рабочее состояние:

- Из состояния простоя - установленное на ноль (z)
- Из состояния временной ошибки - достоверное (v)
- Из состояния постоянной ошибки - непредсказуемое (u) или установленное на ноль
- 35 (z)
- Из рабочего состояния - достоверное (v) - первая запись
- Из рабочего состояния - непредсказуемое (u) - силовой цикл и сохраняемость не по правилу 1.

В нерабочем состоянии состояние данных является недостоверным.

40 При первом конфигурировании и перед первой записью данные приращения SCM находятся в установленном на ноль состоянии, что означает, что все его содержимое представляет собой нули.

Несмотря на то, что содержание данных приращения SCM не изменяется при переходе в состояние временной ошибки, приращение не является доступным. Таким образом, высказывание о том, что данные являются достоверными, может быть описательным, но не слишком значимым из-за отсутствия доступности программы. Поэтому состояние данных в таком сценарии является недостоверным. Также на основе ошибки, которая вызывает переход из рабочего состояния к временной ошибке, если затрагивается

целостность данных, происходит вход в состояние постоянной ошибки, состояние данных остается недостоверным, и данные являются потерянными. Если параллельное исправление может перевести приращение SCM, находящееся в состоянии постоянной ошибки, в рабочее состояние как без деконфигурирования, так и без повторного конфигурирования, оригинальные данные по-прежнему являются потерянными, и то, является состояние данных непредсказуемым или установленным на ноль, зависит от модели.

Сохраняемость приращения SCM и его характеристики RAS (надежность, доступность и пригодность к эксплуатации) также могут определять изменение состояния данных от достоверного к непредсказуемому. Если сохраняемость превышает, ожидается, что состояние данных перейдет из достоверного состояния в непредсказуемое.

Переход или из установленного на ноль, или из непредсказуемого состояния данных в достоверное состояние данных происходит с первой успешной записью. Из-за любой разности между размером записанных данных и размером целевого приращения SCM, изменение в достоверное состояние не описывает фактическую ситуацию для любого еще не записанного местоположения данных. Такое местоположение перед первым доступом для записи фактически по-прежнему описывается как установленное на ноль или непредсказуемое.

После того, как приращение SCM сконфигурировано, незапрашиваемое уведомление тогда делается ожидающим, когда происходит любое одно или несколько событий, которые являются наблюдаемыми в ответ на команду Store SCM Information. Примеры:

1. Состояние операций изменяется от рабочего к временной ошибке или постоянной ошибке, но не сообщается в неверной операции.

2. Состояние операций изменяется от временной ошибки к рабочему.

3. Состояние операций изменяется от временной ошибки к постоянной ошибке.

4. Изменение ранга.

Примеры, когда незапрашиваемые уведомления не делаются ожидающими, включают следующее:

1. Состояние данных изменяется от установленного на ноль или непредсказуемого к достоверному.

2. Завершена команда Configure Storage Class Memory.

3. Завершена команда Deconfigure Storage Class Memory.

Когда уведомление является ожидающим, программа изучает уведомление и может выдать команду Store SCM Information для получения информации. Команда Store SCM Information также может выдаваться в другие моменты времени, в которые программа хотела бы получить информацию о SCM и/или о SALE.

В одном примере уведомление содержит выданное программе прерывание автоматической проверки с соответствующим CRW, указывающим отчет о событии. Программа выдает команду CHSC Store Event Information и получает блок ответа с кодом содержимого, содержащим сигнал уведомления об изменении твердотельного запоминающего устройства.

В одном варианте осуществления изобретения команда Store SCM Information (SSI) представляет собой команду канальной подсистемы, выдаваемую программой (например, операционной системой) для получения информации о твердотельном запоминающем устройстве и/или о записи списка адресов SCM. В одном примере программа выдает команду Channel Subsystem Call, и команда SSI указывается в командном блоке команды, которая направляется канальной подсистеме. Команда осуществляется в канальной подсистеме, и ответ выдается в блоке ответа, который

представляет собой остающуюся часть блока управления размера 4 Кбайт (т.е. запрашиваемая информация сохраняется в области основного запоминающего устройства, указанной для блока ответа). Дополнительные подробности, относящиеся к работе данной команды, описываются со ссылкой на фиг.5Г.

- 5 В начале на этапе 580 программа генерирует вышеуказанный блок запроса для запроса команды Store SCM Information. Блок запроса принимается канальной подсистемой на этапе 582, и в отношении достоверности блока запроса (например, достоверного поля длины, достоверного формата блока запроса команды, установленной команды и т.д.) делается одна или несколько проверок достоверности.
- 10 Если на этапе запроса 584 запрос не является достоверным, то на этапе 586 в блок ответа помещается код ответа, указывающий на проблему, и на этапе 592 выдается блок ответа.

- Однако если на этапе запроса 584 запрос является достоверным, то канальная подсистема на этапе 588 получает информацию от машины (например, процессоров и т.д.), и на этапе 590 заполняет блок ответа. На этапе 592 выдается блок ответа.
- 15 Например, информация содержится в энергонезависимом запоминающем устройстве машины и загружается программно-аппаратным обеспечением в основное запоминающее устройство, только доступное для программно-аппаратного обеспечения в ходе инициализации системы. Канальная подсистема (т.е., в данном случае,
- 20 программно-аппаратное обеспечение) получает информацию путем ее считывания из основного запоминающего устройства, только доступного для программно-аппаратного обеспечения, и заполняет блок ответа.

- С учетом получения информации о твердотельном запоминающем устройстве или в противном случае, может быть принято решение об изменении конфигурации
- 25 твердотельного запоминающего устройства. Решение может приниматься вручную или автоматически при помощи программы или другой сущности. Конфигурацию можно изменить путем добавления приращений или удаления приращений так, как это описывается ниже.

- В одном примере для конфигурирования твердотельного запоминающего устройства
- 30 используется команда Configure Storage Class Memory. Эта команда запрашивает объем твердотельного запоминающего устройства, подлежащий конфигурированию, из доступного пула системы. Объем указывается как размер, кодируемый как число приращений SCM.

- Если не определено иное, данное количество приращений SCM, используемых для
- 35 удовлетворения запроса, находится в инициализированном состоянии. Если запрашиваемое количество приращений SCM может вызывать превышение предела максимума конфигурируемых приращений SCM, предусматривается специальный код ответа.

- Содержимое каждого приращения составляют нули с достоверным СВС. Применимое
- 40 правило сохранения, связанное с каждым конфигурируемым приращением SCM, устанавливается, например, при помощи элементов ручного управления.

- Один вариант осуществления блока запроса команды для команды Configure Storage Class Memory проиллюстрирован на фиг.7А. В одном примере блок 700 запроса
- конфигурирования твердотельного запоминающего устройства содержит:

- 45 длину 702: значение, которое указывает длину блока запроса команды.
- Код 704 команды: значение, которое указывает код команды для команды Configure Storage Class Memory.

Формат 706 (FMT): значение, которое указывает формат блока запроса команды.

Общий размер (TS) 708: значение, которое указывает размер запрашиваемого твердотельного запоминающего устройства, кодируемый как число приращений SCM. Число уже сконфигурированных приращений SCM плюс TS не должно превышать предел максимума конфигурируемых приращений SCM (MCI). Если количество приращений SCM в инициализированном состоянии меньше указанного общего размера, предусматривается специальный код ответа.

Коррелятор 710 асинхронного завершения (ACC): значение, которое выдается в описываемом ниже поле уведомления об асинхронном завершении ответа на уведомление. Коррелятор служит для продолжения оригинального потока, который инициировал запрос.

Один вариант осуществления блока ответа на команду для команды Configure Storage Class Memory проиллюстрирован на ФИГ.7Б. В одном варианте осуществления изобретения блок 730 ответа на команду содержит:

длину 732: значение, которое указывает длину блока ответа команды.

Код 734 ответа: значение, которое описывает результаты попытки выполнения команды Configure Storage Class Memory.

Если в поле кода ответа сохраняется определенный код ответа, для окончания обработки команды инициируется асинхронный процесс. Если в поле кода ответа сохраняется иной код ответа, чем определенный код, ни одно приращение SCM не конфигурируется, ни один асинхронный процесс не инициируется и не выполняется никакого последующего уведомления. Завершение асинхронного процесса указывается в ответе на уведомление.

Формат 736 (FMT): значение, которое указывает формат блока ответа на команду.

Команда Configure Storage Class Memory выдается программой для запроса объема твердотельного запоминающего устройства, подлежащей конфигурированию в адресное пространство SCM. Один вариант осуществления логики, используемой для конфигурирования SCM, описывается со ссылкой на фиг.7В.

Вначале на этапе 740 программа выдает команду Channel Subsystem Call, которая содержит команду Configure SCM. Блок запроса команды Configure SCM на этапе 742 принимается канальной подсистемой, и канальная подсистема на этапе 744 предпринимает попытку выполнения команды. Если на этапе запроса 746 попытка выполнения указанной команды генерирует код ответа, который не указывает на успех, то код ответа на этапе 748 помещается в блок ответа на команду Configure SCM, и на этапе 750 выдается блок ответа.

Если на этапе запроса 746 указывается успешный код ответа, то на этапе 752 код ответа помещается в блок ответа, и на этапе 754 выдается блок ответа. В данном примере успешный код ответа указывает, что поле длины блока запроса является достоверным; команда является доступной в системе; блок запроса команды имеет достоверный формат; канальная подсистема способна осуществить команду (т.е. не занята); запрошенный общий размер не превышает предел максимума конфигурируемых приращений SCM запрашиваемой конфигурации; и запрошенный общий размер не превышает количество приращений SCM в инициализированном состоянии.

Кроме того, на этапе 756 для завершения конфигурирования инициируется асинхронный процесс. Дополнительные подробности, относящиеся к данной обработке, описываются со ссылкой на фиг.7Г.

В одном варианте осуществления изобретения асинхронная обработка на этапе 760 осуществляет конфигурирование для выделения одного или нескольких приращений. Например, для каждого сконфигурированного приращения SCM внутренние элементы

управления изменяются так, чтобы позволить вновь сконфигурированному приращению быть доступным для запросов перемещения ввода-вывода в данный раздел. В частности, с учетом получения канальной подсистемой команды CHSC Configure программно-аппаратное обеспечение канальной подсистемы опрашивает внутренние таблицы для подтверждения того, что имеется достаточно приращений для удовлетворения запроса, и для гарантии того, что запрос не превышает максимум конфигурируемых приращений SCM для данной конфигурации. Если запрос является достоверным, программно-аппаратное обеспечение обновляет одну или несколько таблиц для выделения приращения (приращений) конфигурации и помещения приращения (приращений) в рабочее состояние для конфигурирования. Приращения затем являются доступными для запросов перемещения ввода-вывода (описанных выше) из конфигурации. Завершение асинхронного процесса указывается на этапе 762 в ответе на уведомление.

Данные ответа на уведомление для команды Configure Storage Class Memory выдаются в блоке ответа на команду сохранения информации о событиях (SEI). Один вариант осуществления формата блока ответа на уведомление, используемый для команды Configure Storage Class Memory описывается со ссылкой на фиг.8А.

В одном варианте осуществления изобретения блок 800 ответа на уведомление на команду Configure Storage Class Memory содержит:

длину 802: значение, которое указывает длину блока ответа на команду.

Код 804 ответа: значение, которое описывает результаты попытки выполнения команды Store Event Information CHSC.

Формат 806 (FMT): значение, которое указывает формат блока ответа на команду.

Тип 808 уведомления: значение, которое указывает, что данное уведомление является относящимся к EADM.

Р 810: если установлен, указывает, что канальная подсистема содержит ожидающую информацию о событиях в дополнение к информации, предоставляемой в ответ на данную команду CHSC.

V 812: когда установлен, указывает, что канальная подсистема распознала ситуацию переполнения, и информация о событиях была потеряна.

Код 814 содержимого: значение, которое указывает, что оно представляет собой ответ на завершение выполнения асинхронного процесса, инициированного запросом команды Configure Storage Class Memory.

Вторичный код 816 ответа: значение, которое дополнительно описывает результаты попытки выполнения команды Configure SCM.

Когда вторичный код ответа представляет собой определенное значение, полный объем твердотельного запоминающего устройства был сконфигурирован, как запрашивалось первоначально. В противном случае могут предусматриваться коды ответа, которые указывают, например, на недостоверное поле длины, на то, что не установлена команда Configure SCM, блок команды Configure SCM имеет недостоверный формат, общий запрашиваемый размер может превышать предел MCI, общий запрашиваемый размер превышает количество приращений SCM в инициализированном состоянии, канальная подсистема занята.

Коррелятор 818 асинхронного завершения (ACC): значение, которое первоначально указывается в соответствующем блоке запроса команды.

Один вариант осуществления команды Store Event Information, используемой для выдачи блока ответа на уведомление для команды Configure Storage Class Memory, описывается со ссылкой на фиг.8Б-8 В.

Команда Store Event Information используется для сохранения информации о событиях,

которая канальной подсистемой была сделана ожидающей. Обычно, данная команда выполняется в результате того, что программа получила отчет канала об ожидающей информации о некотором событии.

5 Выполнение команды Store Event Information может изменять информацию, содержащуюся в канальной подсистеме. Команда Store Event Information выполняется синхронно.

Один вариант осуществления блока запроса команды для команды Store Event Information описывается со ссылкой на фиг.8Б. В одном примере блок 830 запроса содержит:

10 длину 832: данное поле указывает длину блока запроса команды.

Код 834 команды: данное поле указывает команду Store Event Information.

Формат 836 (FMT): значение, которое указывает формат блока запроса команды.

15 Маска 838 выбора типа уведомления (NTSM): маска, где каждая позиция бита соответствует значению селектора логического процессора (LPS) с тем же численным значением. В одном примере бит 0 игнорируется и полагается равным единице. Когда позиция бита в указанном диапазоне, начиная с 1, равна нулю, тип уведомления, соответствующий этой позиции бита, не сохраняется в блоке ответа, и, если распознается как ожидающий, отбрасывается. Когда указанный бит равен единице, тип уведомления, соответствующий этой позиции бита, может сохраняться в блоке ответа.

20 В одном варианте осуществления изобретения, со ссылкой на фиг.8 В, ниже описывается блок 850 ответа для команды Store Event Information:

длина 852: значение, которое указывает исходную длину блока ответа на команду. Длина по завершении зависит от кода ответа, который сохраняется в результате попытки выполнения команды Store Event Information.

25 Если в поле кода ответа сохраняется иной код ответа, чем код, указывающий на успех, информация не сохраняется в области данных ответа.

Если в поле кода ответа сохраняется код ответа, указывающий на успех, информация о событии сохраняется в области данных ответа.

30 Код 854 ответа: значение, которое описывает результаты попытки выполнения команды Store Event Information.

Для команды Store Event Information область данных ответа содержит часть фиксированной длины и часть переменной длины.

35 Для ответа указанного формата, когда NT является ненулевым, формат зависит от конкретного типа уведомления, и формат поля, зависящего от кода содержимого, зависит от конкретного типа уведомления и поля кода содержимого (CC) вместе взятых.

Формат 856 (FMT): значение, которое указывает формат блока ответа на команду.

Тип 858 уведомления (NT): значение, которое указывает тип уведомления (NT). Для команды Configure SCM предусматривается специальное значение.

40 Флаг 860 P: если установлен, указывает, что канальная подсистема содержит ожидающую информацию о событиях в дополнение к информации, предоставляемой в ответ на данную команду CHSC. Программа может получать дополнительную информацию путем повторного выполнения команды Store Event Information. Если не установлена, данный флаг указывает, что канальная подсистема не содержит дополнительной ожидающей информации о событиях.

45 Флаг 862 V: когда установлен, указывает, что канальная подсистема распознала ситуацию переполнения, и информация о событиях была потеряна. Ситуация переполнения была распознана в то время, как информация о событиях, не содержащаяся в области данных ответа, являлась самой недавней ожидающей

информацией. Переполнение не оказывает влияния на информацию, содержащуюся в области данных ответа.

Код 864 содержимого (CC): значение, которое описывает тип информации, которая содержится в области данных ответа. В одном примере данное значение указывает уведомление об изменении твердотельного запоминающего устройства, в котором одно или несколько приращений SCM изменили состояние или статус.

Поле 866, зависящее от кода содержимого: данное поле может содержать дополнительную информацию, относящуюся к событию.

Успешное уведомление об изменении конфигурации может подсказать программе выдачу команды Store SCM Information для получения подробностей, относящихся к конфигурации.

Помимо увеличения памяти твердотельного запоминающего устройства, память твердотельного запоминающего устройства может быть уменьшена. Команда Deconfigure Storage Class Memory запрашивает объем твердотельного запоминающего устройства, подлежащий удалению из адресного пространства SCM в запрашиваемой конфигурации. Приращение SCM, подлежащее деконфигурированию, должно находиться в сконфигурированном состоянии.

Приращения SCM, подлежащие деконфигурированию, указываются в записи списка запроса приращений SCM, описываемого в данном описании. В записи списка запроса приращений SCM (SIRLE) может указываться одно или несколько непрерывных приращений SCM. Для каждого списка приращений (также известного как экстенс), который не является непрерывным с любым другим списком приращений, может указываться отдельная SIRLE.

Независимо от правил сохраняемости успешное деконфигурирование приращения SCM помещает приращение в неинициализированное состояние. Когда завершается установка на ноль, приращение SCM переходит из неинициализированного состояния в инициализированное состояние.

Один вариант осуществления блока запроса команды для команды Deconfigure Storage Class Memory проиллюстрирован на фиг.9А. Блок 900 запроса команды для команды Deconfigure Storage Class Memory содержит, например:

длину 902: значение, которое указывает длину блока запроса команды. В одном примере длина равна $32+(N \times 16)$ байт, где N - число записей списка запроса приращений SCM (записей SIRLE). Достоверная длина в данном примере без остатка делится на 16 и находится в диапазоне $(32+1 \times 16) \leq L1 \leq (32+253 \times 16)$.

Код 904 команды: значение, которое указывает код команды для команды Deconfigure Storage Class Memory.

Формат 906 (FMT): значение, которое указывает формат блока запроса команды.

Коррелятор 908 асинхронного завершения (ACC): значение, которое выдается в вышеописанном уведомлении об асинхронном завершении.

Список 910 запроса приращений SCM: данное поле содержит список запроса приращений SCM (SIRL). Список запроса приращений SCM содержит одну или несколько записей (SIRLE). Длина SIRL определяется из значения поля длины.

Запись списка запроса приращений SCM (SIRLE) указывает размер и местоположение указанного экстенс твердотельного запоминающего устройства (например, списка приращений). Экстенс, или экстенс SCM, представляет собой указанный размер твердотельного запоминающего устройства.

Согласно фиг.9Б, в одном примере, SIRLE 920 содержит:

общий размер 922 (TS): значение, которое указывает размер твердотельного

запоминающего устройства, подлежащий деконфигурированию, кодируемый как число приращений SCM.

Начальный адрес 924 SCM (SA): значение, которое представляет собой адрес SCM и является местоположением в адресном пространстве SCM, из которого следует удалить первое, или единственное, приращение SCM, деконфигурируемое посредством SIRLE. Положения младших битов, которые могут составлять смещение в пределах первого приращения SCM, в данном примере игнорируются и полагаются равными нулю.

Когда общий размер больше единицы, каждое дополнительное приращение SCM за первым приращением располагается по адресу SCM, который делится без остатка на размер приращения SCM, которое содержит сконфигурированное приращение SCM, и положение которого является непрерывным с последним байтом предыдущего приращения SCM. Иными словами, в следующем, непрерывном местоположении.

Если пространство, описываемое начальным адресом и общим размером вместе взятыми, не полностью заполнено сконфигурированными приращениями SCM, предусматривается указанный код ответа, не деконфигурируется ни одно приращение SCM, не инициируется ни один асинхронный процесс, и не появляется ни одного последующего уведомления.

При успешном завершении, каждое деконфигурированное приращение SCM входит в зарезервированное состояние и затем устанавливается на ноль перед помещением в состояние простоя.

Блок ответа на команду для команды Deconfigure Storage Class Memory проиллюстрирован на фиг.9В. В одном варианте осуществления изобретения блок 950 ответа на команду содержит:

длину 952: значение, которое указывает длину блока ответа на команду.

Код 954 ответа: значение, которое описывает результаты попытки выполнения команды Deconfigure Storage Class Memory.

Если в поле кода ответа сохраняется код ответа с указанным значением, для окончания обработки команды инициируется асинхронный процесс. Если в поле кода ответа сохраняется иной код ответа, чем указанное значение, ни одно приращение SCM не деконфигурируется, не инициируется никакой асинхронный процесс, и не делается никакого последующего уведомления. Завершение асинхронного процесса указывается в ответе на уведомление.

Перед тем, как синхронная часть команды Deconfigure Storage Class Memory завершается с указанным кодом ответа, все записи в списке запроса приращений SCM опрашиваются с тем, чтобы гарантировать, что все указанные приращения SCM находятся в сконфигурированном состоянии.

Формат 956 (FMT): значение, которое указывает формат блока ответа на команду. Один вариант осуществления логики, связанной с командой Deconfigure SCM, описывается со ссылкой на фиг.9Г-9Д.

В начале на этапе 970 программа выдает команду Channel Subsystem Call, которая содержит команду (Deconfigure SCM). Блок запроса команды Deconfigure SCM принимается канальной подсистемой на этапе 972, и канальная подсистема на этапе 974 предпринимает попытку выполнения этой команды. Если попытка выполнения команды на этапе запроса 976 дает код ответа, который не указывает на успех, то на этапе 978 код ответа помещается в блок ответа на команду Deconfigure SCM, и на этапе 980 выдается блок ответа.

Если на этапе запроса 976 указывается успешный код ответа, то на этапе 982 код ответа помещается в блок ответа, и на этапе 984 выдается блок ответа. В данном

примере успешный код ответа указывает, что поле длины блока запроса является достоверным; команда в данной системе является доступной; блок запроса команды имеет достоверный формат; канальная подсистема способна осуществить команду (т.е. не занята); и приращения SCM изначально находятся в сконфигурированном состоянии.

5 Кроме того, для завершения деконфигурирования на этапе 986 инициируется асинхронный процесс. Дополнительные подробности, относящиеся к данной обработке, описываются со ссылкой на фиг.9Д.

В одном варианте осуществления изобретения деконфигурирование на этапе 990 осуществляет асинхронная обработка. Например, освобождается одно или несколько
10 приращений. Приращение SCM переводится из сконфигурированного состояния в зарезервированное состояние. За входом в зарезервированное состояние следует процесс установки на ноль, и, после его завершения, приращение SCM переходит в состояние простоя. Завершение асинхронного процесса указывается на этапе 992 в ответе на уведомление.

15 Данные ответа на уведомление для команды Deconfigure Storage Class Memory выдаются в блоке ответа на команду Store Event Information CHSC (SEI). Один пример такого блока ответа описан со ссылкой на фиг.8А. Однако, в данном примере код содержимого указывает, что это ответ на завершение выполнения асинхронного процесса, инициированного запросом команды Deconfigure Storage Class Memory.

20 Аналогично, вторичный код ответа дополнительно описывает результаты попытки выполнения команды Deconfigure Storage Class Memory.

В одном из дополнительных вариантов осуществления изобретения выделение и освобождение приращений может запрашиваться посредством панели, предоставленной пользователю. Например, для обеспечения графического интерфейса, через который
25 пользователь может указывать системе параметры, может использоваться сервисный элемент. Для твердотельного запоминающего устройства панель, называемая панелью распределения твердотельного запоминающего устройства, позволяет пользователю осуществлять следующие операции:

1. Указывать для данной конфигурации максимум конфигурируемых приращений
30 (МСІ);
2. Выделять приращения для конфигурации;
3. Освобождать приращения из конфигурации.

Панель также позволяет просматривать выделенные конфигурации приращения и МСІ, а также количество приращений в доступном, недоступном и
35 неинициализированном пулах. Когда по причине действия SE распределение приращений изменяется, или когда изменяется размер одного из пулов, конфигурациям отправляется уведомление.

Выше подробно было описано средство для администрирования твердотельного запоминающего устройства. Оно предусматривает абстракцию, позволяющую
40 программе получать доступ к памяти без знания специфики этой памяти. Согласно одной или нескольким особенностям настоящего изобретения предложено средство для перемещения данных между основным запоминающим устройством и SCM; для сброса или освобождения SCM; для конфигурирования и деконфигурирования SCM; и для отыскания конфигурации SCM. Другие средства также предусмотрены.

45 В одном варианте осуществления изобретения твердотельное запоминающее устройство представляется для программ пользовательского уровня как область линейной памяти, не зависящая от ее физической реализации для ряда устройств и адаптеров ввода-вывода.

Подробности, относящиеся к канальным подсистемам и/или средству ADM, описаны в патенте США №5377337 под названием "Method and Means for Enabling Virtual Addressing Control By Software Users Over A Hardware Page Transfer Control Entity", Antognini et al., выданном 27 декабря 1994 г.; патенте США №5442802 под названием "Asynchronous Co-Processor Data Mover Method and Means", Brent et al., выданном 15 августа 1995 г.; и в патенте США №5526484 под названием "Method and System for Pipelining the Processing of Channel Command Words", выданном 11 июня 1996 г., каждый из которых ссылкой полностью включен в данное описание.

Кроме того, дополнительная информация, относящаяся к канальной подсистеме и командам, связанным с ней (для конкретной реализации архитектуры z/Architecture[®]), представлена ниже.

Ввод-вывод (I/O)

Термины «ввод» и «вывод» используются для описания передачи данных между устройствами ввода-вывода и основным запоминающим устройством. Операция, включающая такой тип передачи, называется операцией ввода-вывода. Средства, используемые для управления операциями ввода вывода, в совокупности называются канальной подсистемой. (Устройства ввода-вывода и их устройства управления присоединяются к канальной подсистеме.)

Канальная подсистема

Канальная подсистема направляет поток информации между устройствами ввода-вывода и основным запоминающим устройством. Она освобождает центральные процессоры от задачи прямой связи с устройствами ввода-вывода и позволяет обработке данных происходить параллельно с обработкой ввода-вывода. При управлении потоком информации в устройства ввода-вывода или из них канальная подсистема использует в качестве линии связи один или несколько канальных трактов. Как часть обработки ввода-вывода канальная подсистема также осуществляет операцию управления трактами путем проверки доступности канального тракта, выбирает доступный канальный тракт и инициирует осуществление устройством операции ввода-вывода.

Внутри канальной подсистемы находятся субканалы. Для каждого устройства ввода-вывода, доступного для программы посредством канальной подсистемы, предусматривается и предназначается один субканал.

Средство нескольких наборов субканалов является необязательным средством. Когда оно установлено, субканалы разбиваются на несколько наборов субканалов, и каждый набор субканалов может обеспечивать для устройства ввода-вывода один выделенный субканал. В зависимости от используемой модели и интерфейса для некоторых устройств ввода-вывода доступ разрешается только через определенные наборы субканалов.

Каждый субканал обеспечивает информацию, касающуюся связанного устройства ввода-вывода и его присоединения к канальной подсистеме. Субканал также обеспечивает информацию, касающуюся операций ввода-вывода и других функций, включаемых в связанное устройство ввода-вывода. Субканал представляет собой средства, при помощи которых канальная подсистема доставляет информацию о связанных устройствах ввода-вывода центральным процессорам, которые получают эту информацию путем выполнения команд ввода-вывода. Фактическое количество предоставляемых субканалов зависит от модели и конфигурации; максимальная адресуемость в каждом наборе субканалов составляет 0-65,535.

Устройства ввода-вывода присоединяются к канальной подсистеме через устройства управления посредством канальных трактов. Устройства управления могут

присоединяться к канальной подсистеме посредством более чем одного канального тракта, а устройство ввода-вывода может присоединяться более чем к одному устройству управления. В зависимости от модели и конфигурации, в целом, отдельное устройство ввода-вывода может быть доступно для канальной подсистемы через субканал

5 посредством вплоть до восьми различных канальных трактов. Общее количество канальных трактов, предусматриваемое канальной подсистемой, зависит от модели и конфигурации; максимальная адресуемость составляет 0-255.

Производительность канальной подсистемы зависит от ее применения и от модели системы, в которой она реализуется. Канальные тракты снабжаются различными

10 средствами передачи данных, и устройство ввода-вывода, рассчитанное на передачу данных только с конкретной скоростью (например, запоминающее устройство на магнитной ленте, дисковое запоминающее устройство) может действовать только на канальном тракте, который способен приспособливаться, по меньшей мере, к такой скорости передачи данных.

Канальная подсистема содержит общие средства управления операциями ввода-вывода. Когда указанные средства предусматриваются в форме отдельного, автономного оборудования, рассчитанного специально на управление устройствами

15 ввода-вывода, операции ввода-вывода полностью совмещаются с действиями в центральных процессорах. Единственными циклами работы основного запоминающего устройства, требуемыми канальной подсистемой в ходе операций ввода-вывода, являются те из них, которые необходимы для передачи данных и управляющей информации из конечных местоположений в основном запоминающем устройстве или

20 в эти конечные местоположения, наряду с теми циклами работы, которые могут потребоваться канальной подсистеме для доступа к субканалам, когда они реализуются как часть неадресуемого основного запоминающего устройства. Указанные циклы не задерживают программы ЦП за исключением случаев, когда и ЦП, и канальная подсистема параллельно предпринимают попытки обращения к одной и той же области

25 основного запоминающего устройства.

Наборы субканалов

Когда средство нескольких наборов субканалов установлено, субканалы разбиваются

30 на несколько наборов субканалов. Может существовать до четырех наборов субканалов, каждый из которых идентифицируется по идентификатору набора субканалов (SSID). Когда средство нескольких наборов субканалов не установлено, имеется только один набор субканалов с SSID, равным нулю. Когда средство нескольких наборов субканалов

35 не разблокировано, видимым для программы является только нулевой набор субканалов.

Субканалы

Субканал доставляет программе логический внешний вид устройства и содержит информацию, требуемую для поддержания единичной операции ввода-вывода. Субканал

40 состоит из внутреннего запоминающего устройства, которое содержит информацию в форме канально-программного маркера, идентификатора канального тракта, номера устройства, счетчик, указатели статусов, код подкласса прерывания ввода-вывода, а также информацию о доступности тракта и функциях, которые являются ожидающими или осуществляемыми. Операции ввода-вывода инициируются устройством путем

45 выполнения команд ввода-вывода, которые указывают субканал, связанный с этим устройством.

Каждое устройство является доступным посредством одного субканала в каждой канальной подсистеме, которой оно назначено в ходе конфигурирования во время

установки. Устройство может представлять собой физически идентифицируемую единицу, или оно может монтироваться в внутри корпуса устройства управления. Например, в некоторых дисковых запоминающих устройствах каждый привод, используемый при поиске данных, рассматривается как устройство. Во всех случаях устройство, с точки зрения канальной подсистемы, представляет собой сущность, которая однозначно связана с одним субканалом и которая отвечает на выбор канальной подсистемой путем использования протоколов связи, определенных для типа канального тракта, посредством которого обеспечивается доступ к ним.

На некоторых моделях субканалы предусматриваются в блоках. На таких моделях может предусматриваться больше субканалов, чем имеется присоединенных устройств. Субканалы, которые предусмотрены, но не содержат назначенных им устройств, не используются канальной подсистемой для осуществления каких-либо функций, и указываются путем сохранения связанного бита достоверности номера устройства как ноля в блоке информации о субканале в субканале.

Количество субканалов, предусматриваемых канальной подсистемой, не зависит от количества канальных трактов к связанным устройствам. Например, устройство, доступное через альтернативные канальные тракты, по-прежнему отображается как единственный субканал. Когда установлено средство наборов субканалов, каждый субканал адресуется с использованием 16-битного бинарного номера субканала и двухбитного SSID.

После обработки ввода-вывода в субканале, запрошенной путем выполнения команды START SUBCHANNEL, ЦП освобождается для другой работы, а канальная подсистема собирает или разбирает данные и синхронизирует передачу байтов данных между устройством ввода-вывода и основным запоминающим устройством. Для выполнения этой задачи канальная подсистема поддерживает и обновляет адрес и число, которые описывают адрес назначения или источника данных в основном запоминающем устройстве. Аналогично, когда устройства ввода-вывода создает сигналы, которые следует довести до внимания программы, канальная подсистема преобразовывает сигналы в информацию о статусе и сохраняет эту информацию в субканале, где она может быть отыскана программой.

Присоединение устройств ввода-вывода

Канальные тракты

Канальная подсистема связывается с устройствами ввода-вывода посредством канальных трактов между канальной подсистемой и устройствами управления.

Устройство управления может быть доступно для канальной подсистемы посредством более чем одного канального тракта. Аналогично, устройство ввода-вывода может быть доступно для канальной подсистемы через более чем одно устройство управления, каждое из которых содержит один или несколько канальных трактов к канальной подсистеме.

Устройства, которые присоединяются к канальной подсистеме посредством нескольких канальных трактов, сконфигурированных в субканал, могут быть доступны для канальной подсистемы с использованием любого из доступных канальных трактов. Аналогично, устройство, содержащее средство динамической перестройки соединений и действующее в многотрактном режиме, может быть инициализировано для работы таким образом, чтобы устройство могло выбирать любой из доступных канальных трактов, сконфигурированных в субканал, при логической перестройке соединений с канальной подсистемой для продолжения цепочки операций ввода-вывода.

Канальная подсистема может содержать больше одного типа канальных трактов.

Примерами типов канальных трактов, используемых канальной подсистемой, являются: интерфейс ввода-вывода ESCON, интерфейс ввода-вывода FICON, преобразованный интерфейс ввода-вывода FICON и интерфейс ввода-вывода IBM System/360 и System/370. Термин «интерфейс последовательного ввода-вывода» используется для обращения к интерфейсу ввода-вывода ESCON, интерфейсу ввода-вывода FICON и преобразованному интерфейсу ввода-вывода FICON. Термин «интерфейс параллельного ввода-вывода» используется для обращения к интерфейсу ввода-вывода IBM System/360 and System/370.

Интерфейс ввода-вывода ESCON описан в публикации системной библиотеки IBM Enterprise Systems Architecture/ 390 ESCON I/O Interface, SA22-7202, которая ссылкой полностью включена в данное описание.

Интерфейс ввода-вывода FICON описан в документе стандартов ANSI Fibre Channel - Single-Byte Command Code Sets-2 (FC-SB-2).

Интерфейс ввода-вывода IBM System/360 и System/370 описан в публикации системной библиотеки IBM System/ 360 and System/370 I/O Interface Channel to Control Unit OEMI, GA22-6974, которая ссылкой полностью включена в данное описание.

В зависимости от типа канального тракта средств, обеспечиваемых канальным трактом и устройством ввода-вывода, и устройства ввода-вывода, операция ввода-вывода может происходить в одном из трех режимов: кадровый мультиплексный режим, пакетно-монополюсный режим или байтовый мультиплексный режим.

В кадровом мультиплексном режиме устройство ввода-вывода может оставаться логически соединенным с канальным трактом в течение всей длительности выполнения канальной программы. Средства канального тракта, допускающие работу в кадровом мультиплексном режиме, могут совместно использоваться несколькими параллельно работающими устройствами ввода-вывода. В этом режиме информация, требуемая для завершения операции ввода-вывода, разделяется на кадры, которые могут чередоваться с кадрами из других устройств ввода-вывода. В ходе указанного периода времени логически соединенными с канальным трактом считаются несколько устройств ввода-вывода.

В пакетно-монополюсном режиме устройство ввода-вывода монополизует канальный тракт и остается логически соединенным с канальным трактом в течение передачи пакета информации. Ни одно другое устройство не может связываться по канальному тракту в течение времени передачи пакета. Пакет может состоять из нескольких байтов, всего блока данных, последовательности блоков со связанной управляющей информацией и информацией о статусе (длина блока может быть равной нулю) или из информации о статусе, которая монополизует канальный тракт. Средства канального тракта, допускающие работу в пакетно-монополюсном режиме, могут совместно использоваться несколькими параллельно работающими устройствами ввода-вывода.

Некоторые канальные тракты в ходе пакетно-монополюсного режима могут допускать отсутствие передачи данных в течение приблизительно половины минуты, что, например, происходит, когда считывается длинный промежуток на магнитной ленте. Когда отсутствие передачи данных превышает установленный предел, может указываться сбой оборудования.

В байтовом мультиплексном режиме устройство ввода-вывода остается логически соединенным с канальным трактом только в течение короткого промежутка времени. Средства канального тракта, допускающие работу в байтовом мультиплексном режиме, могут совместно использоваться несколькими параллельно работающими устройствами

ввода-вывода. В данном режиме все операции ввода-вывода разделяются на короткие промежутки времени, в ходе которых по каналному тракту передается только некоторый сегмент информации. В течение указанного промежутка с каналным трактом логически соединено только одно устройство и связанный субканал.

5 Промежутки, связанные с параллельной работой нескольких устройств ввода-вывода, выстраиваются в последовательность в ответ на потребности устройств. Средство канальной подсистемы, связанное с субканалом, применяет свои управляющие элементы для любой операции только в течение времени, требуемого для передачи сегмента информации. Сегмент может состоять из единственного байта данных, нескольких
10 байтов данных, отчета о статусе из устройства или из управляющей последовательности, используемой для инициирования новой операции.

Обычно устройства с высокими требованиями к скоростям передачи данных работают с каналными трактами в кадровом мультиплексном режиме, более медленные устройства работают в пакетно-монопольном режиме, и самые медленные устройства
15 работают в байтовом мультиплексном режиме. Некоторые устройства управления содержат ручной переключатель для установки желаемого режима работы.

Операция ввода-вывода, которая происходит на каналном тракте с параллельным типом интерфейса ввода-вывода, может происходить или в пакетно-монопольном режиме, или в байтовом мультиплексном режиме в зависимости от средств,
20 обеспечиваемых каналным трактом и устройством ввода-вывода. Для увеличения производительности некоторые каналные тракты и устройства управления оснащаются средствами для высокоскоростной и потоковой передачи данных. Описание двух таких средств см. в публикации System Library publication IBM System/360 and System/ 370 I/O Interface Channel to Control Unit OEMI, GA22-6974, которая ссылкой полностью включена
25 в данное описание.

Операция ввода-вывода, которая происходит на каналном тракте с последовательным типом интерфейса ввода-вывода, может происходить или в кадровом мультиплексном режиме, или в пакетно-монопольном режиме. Для увеличения
производительности некоторые устройства управления, присоединяемые к
30 последовательному интерфейсу ввода-вывода, предусматривают средство доставки программе данных ощущений параллельно с предоставлением статуса проверки устройства, если такое действие допускается программой.

В зависимости от устройства управления или канальной подсистемы доступ к устройству через субканал может быть ограничен типом с единственным каналным
35 трактом.

Вышеописанные режимы и признаки оказывают влияние только на протокол, используемый для передачи информации по каналному тракту, и на скорость передачи. ЦП или каналными программами не могут наблюдаться никакие воздействия в отношении способов, которыми осуществляются указанные программы.

40 Устройства управления

Устройство управления обеспечивает логические мощности, необходимые, чтобы приводить в действие и управлять устройством ввода-вывода, и приспособливает характеристики каждого устройства так, что оно может отвечать на стандартную форму управления, обеспечиваемую канальной подсистемой.

45 Связь между устройством управления и канальной подсистемой происходит по каналному тракту. Устройство управления принимает сигналы управления от канальной подсистемы, управляет распределением по времени передачи данных по каналному тракту и предоставляет показания относительно статуса устройства.

Устройство ввода-вывода, подключенное к устройству управления, может быть предназначено выполнять только определенные ограниченные операции, или оно может выполнять множество различных операций. Обычной операцией является перемещение среды записи и данных записи. Чтобы выполнять свои операции, устройство нуждается в подробных сигнальных последовательностях, специфических для своего типа устройства. Устройство управления декодирует команды, полученные от канальной подсистемы, интерпретирует их для конкретного типа устройства, и предоставляет сигнальную последовательность, необходимую для выполнения операции.

Устройство управления может размещаться отдельно, или оно может быть физически и логически объединено с устройством ввода-вывода, канальной подсистемой или ЦП. Для большинства электромеханических устройств существует хорошо определенный интерфейс между устройством и устройством управления из-за различия в типе оборудования, требуемого устройством управления и устройством. Эти электромеханические устройства часто бывают устройствами типа, в котором из группы, подключенной устройству управления для передачи данных в момент времени (например, устройства на магнитной ленте или устройства доступа к диску), требуется только одно устройство, и элемент управления используется совместно некоторым числом устройств ввода-вывода. С другой стороны, в некоторых электронных устройствах ввода-вывода, таких как адаптер «канал-канал», устройство управления не имеет собственной сущности.

С точки зрения программиста большинство функций, выполняемых устройством управления, могут быть объединены с функциями, выполняемыми устройством ввода-вывода. Следовательно, как правило, в этом описании не делается конкретного упоминания об устройстве управления; выполнение операций ввода-вывода описывается, как если бы устройства ввода-вывода сообщались непосредственно с канальной подсистемой. Ссылка на устройство управления делается только при особом упоминании функции, выполняемой им, или при описании того, как совместное использование устройства управления рядом устройств влияет на производительность операций ввода-вывода.

Устройства ввода-вывода

Устройство ввода-вывода (I/O) обеспечивает внешнее запоминающее устройство, средства связи между системами обработки данных или средства связи между системой и ее средой. Устройства ввода-вывода включают такое оборудование как устройства на магнитных лентах, запоминающие устройства прямого доступа (например, диски), устройства отображения, клавиатурные печатающие устройства, принтеры, устройства телеобработки и сенсорное оборудование. Устройство ввода-вывода может быть физически отдельным оборудованием или оно может совместно использовать оборудование с другими устройствами ввода-вывода.

Большинство типов устройств ввода-вывода, такие как принтеры, или ленточные устройства, используют внешние среды, и эти устройства физически отличимы и идентифицируемы. Другие типы являются исключительно электронными и прямо не работают с физическими средами записи. Адаптер «канал-канал», например, обеспечивает передачу данных между двумя канальными трактами, и данные никогда не достигают физической среды записи за пределами основного запоминающего устройства. Аналогично, контроллеры связи могут управлять передачей информации между системой обработки данных и удаленной станцией, и ее входом и выходом являются сигналы на линии передачи.

В простейшем случае устройство ввода-вывода подключается к одному устройству

управления и доступно из одного канального тракта. Переключающее оборудования доступно для того, чтобы делать некоторые устройства достижимыми из одного или нескольких канальных трактов путем переключения устройств между устройствами управления и переключения устройств управления между канальными трактами. Такое переключающее оборудование предоставляет множество трактов, по которым можно получить доступ к устройству ввода-вывода. Множество канальных трактов к устройству ввода-вывода предоставляются, чтобы улучшить производительность или доступность ввода-вывода, или и то, и другое, в системе. Управление многими канальными трактами к устройствам находится под контролем канальной подсистемы и устройства, но канальные тракты могут косвенно управляться программой.

Адресация ввода-вывода

Четыре различных типа адресации ввода-вывода обеспечиваются канальной подсистемой для необходимой адресации различных компонентов: идентификаторы канального тракта, номера субканалов, номера устройств и, хотя невидимые для программ, адреса, зависящие от типа канального тракта. Когда установлено средство нескольких наборов субканалов, в адресации ввода-вывода также используется идентификатор набора субканалов (SSID).

Идентификатор набора субканалов

Идентификатор набора субканалов (SSID) представляет собой двубитное значение, приписанное к каждому предоставленному набору субканалов.

Идентификатор канального тракта

Идентификатор канального тракта (CHPID) представляет собой уникальное для системы восьмибитное значение, приписанное к каждому установленному канальному тракту системы. CHPID используется для адресации канального тракта. CHPID определяется адресом второго операнда RESET CHANNEL PATH и используется, чтобы указывать канальный тракт, который необходимо восстановить. Канальные тракты, посредством которых доступно устройство, идентифицируются в блоке информации субканала (SCHIB), каждый своим связанным CHPID, когда выполняется STORE SUBCHANNEL. CHPID также может использоваться в сообщениях оператора, когда необходимо идентифицировать конкретный канальный тракт. Модель системы может обеспечивать целых 256 канальных трактов. Максимальное число канальных трактов и назначение идентификаторов CHPID для канальных трактов зависит от модели системы.

Номер субканала

Номер субканала представляет собой уникальное для системы 16-битное значение, используемое для адресации субканала. Это значение является уникальным в наборе субканалов канальной подсистемы. Адресация к субканалу осуществляется с помощью восьми команд ввода-вывода: CANCEL SUBCHANNEL, CLEAR SUBCHANNEL, HALT SUBCHANNEL, MODIFY SUBCHANNEL, RESUME SUBCHANNEL, START SUBCHANNEL, STORE SUBCHANNEL и TEST SUBCHANNEL. Все функции ввода-вывода относительно конкретного устройства ввода-вывода определяются программой путем указания субканала, приписанного к устройству ввода-вывода. Субканалам в каждом наборе субканалов всегда приписываются номера субканалов из одного диапазона смежных чисел. Субканалом с самым малым номером является субканал 0. Субканалом с самым большим номером в канальной подсистеме является субканал с номером, равным числу на единицу меньше числа предоставленных субканалов. В каждом наборе субканалов может предоставляться не более 65536 субканалов. Как правило, номера субканалов используются только при связи между программой ЦП

и канальной подсистемой.

Номер устройства

Каждый субканал, который имеет назначенное устройство ввода-вывода, также содержит параметр, называемый номером устройства. Номер устройства представляет собой 16-битное значение, которое приписывается в качестве одного из параметров субканала, когда субканалу назначается устройство. Номер устройства уникальным образом идентифицирует устройство для программы.

Номер устройства обеспечивает способ идентификации устройства, независимый от каких-либо ограничений, накладываемых системной моделью, конфигурацией или протоколами канального тракта. Номер устройства используется в процессах передачи информации, касающихся устройства, которые имеют место между системой и оператором системы. Например, номер устройства вводится оператором системы, чтобы указывать устройство ввода, которое надо использовать для начальной загрузки программы.

Замечание для программирования: Номер устройства назначается во время установки устройства и может иметь любое значение. Однако пользователь должен принимать во внимание любые ограничения на назначение номера устройства, которые могут накладываться управляющей программой, вспомогательными программами или конкретным устройством управления или устройством ввода-вывода.

Идентификатор устройства

Идентификатор устройства представляет собой адрес, невидимый программе, который используется канальной подсистемой для осуществления связи с устройствами ввода-вывода. Тип используемого идентификатора устройства зависит от типа конкретного канального тракта и обеспечиваемых протоколов. Каждый субканал содержит один или несколько идентификаторов устройства.

Для канального тракта интерфейса ввода-вывода параллельного типа идентификатор устройства называется адресом устройства и имеет восьмибитное значение. Для интерфейса ввода-вывода ESCON идентификатор устройства состоит из четырехбитного адреса устройства управления и восьмибитного адреса устройства. Для интерфейса ввода-вывода FICON идентификатор устройства состоит из восьмибитного ID образа устройства управления и восьмибитного адреса устройства. Для FICON-преобразованного интерфейса ввода-вывода идентификатор устройства состоит из четырехбитного адреса устройства управления и восьмибитного адреса устройства.

Адрес устройства идентифицирует конкретное устройство ввода-вывода (и, на параллельном интерфейсе ввода-вывода, - устройство управления), связанное с субканалом. Адрес устройства может идентифицировать, например, конкретный накопитель на магнитной ленте, устройство доступа к диску или линию передачи данных. В качестве адреса устройства может быть назначено любое число в диапазоне 0-255.

Расширение Fibre-Channel

Средства расширения Fibre Channel (FCX) представляют собой необязательные средства, которые обеспечивают формирование канальной программы, которая состоит из слова (TCW) управления передачей, которое указывает блок (TCCB) управления командами передачи и блок (TSB) статуса передачи. TCCB включает область (TCA) команд передачи, которая содержит список до 30 команд ввода-вывода, которые имеют форму слов (DCW) команд устройств. TCW и его TCCB могут определять операцию или чтения, или записи. В дополнение к IRB, TSB содержит статус завершения и другую информацию, относящуюся к канальной программе TCW.

Средства FCX обеспечивают возможность прямо или косвенно указывать один или все TCCB, область памяти входных данных и область памяти выходных данных. Когда область памяти указана прямо, TCW определяет местоположение одного непрерывного блока памяти. Когда область памяти указана косвенно, TCW указывает местоположение списка одного или нескольких слов (TIDAW) косвенного адреса данных передачи. Списки TIDAW и область памяти, указанная каждым TIDAW в списке, не могут превышать границ 4 кбайт.

Средства FCX также обеспечивают операцию опроса, которая может инициироваться командой CANCEL SUBCHANNEL, чтобы определять состояние операции ввода-вывода.

Слова команд ввода-вывода

Слово команды ввода-вывода определяет команду и содержит информацию, связанную с командой. Когда средства FCX установлены, имеется две простейших формы слов команд ввода-вывода, а именно командное слово (CCW) канала и командное слово (DCW) устройства.

CCW имеет размер 8 байт и определяет команду, подлежащую выполнению. Для команд, которые инициируют определенные операции, CCW также указывает область памяти, связанную с операцией, подсчет байтов данных, действие, которое необходимо предпринять, когда команда завершается, и другие настройки. Все устройства ввода-вывода распознают слова CCW.

DCW имеет размер 8 байт и определяет команду, подлежащую выполнению, подсчет байтов данных и другие настройки. Устройства ввода-вывода, которые поддерживают FCX, распознают слова DCW.

Командное слово (TCW) передачи

TCW указывает блок (TCCB) управления командами передачи, который содержит список команд, которые необходимо передать и на устройство ввода-вывода и выполнить на нем. TCW также указывает области памяти для команд в TCCB, а также блок (TSB) статуса передачи, который должен содержать статус операции ввода-вывода.

Организация канальной программы

Когда средства FCX не установлены, имеется единственная форма канальной программы, а именно канальная программа CCW. Когда средства FCX установлены, имеется дополнительная форма канальной программы, а именно канальная программа TCW. Обе формы канальных программ описываются ниже.

Канальная программа CCW

Канальная программа, состоящая из одного или нескольких слов CCW, называется канальной программой CCW (CCP). Такая канальная программа содержит одно или несколько слов CCW, которые логически связаны и упорядочены для последовательного выполнения канальной подсистемой.

Канальная программа TCW

Канальная программа, которая состоит из одного слова TCW, называется канальной программой TCW. TCW указывает блок (TCCB) управления командами передачи, который содержит от 1 до 30 слов DCW. Слова DCW в TCCB логически связаны и упорядочены для последовательного выполнения. Для слов DCW, которые определяют информацию управления, TCCB также содержит информацию управления для этих команд. TCW также указывает область или области памяти для слов DCW, которые определяют передачу данных от устройства или на устройство и местоположение блока (TSB) статуса передачи для статуса завершения. TCCB и области памяти для передачи данных могут определяться или как непрерывная, или как прерывная память.

TCW также указывает TSB для статуса завершения.

Выполнение операций ввода-вывода

Операции ввода-вывода инициируются и управляются посредством информации с четырьмя типами форматов: командой START SUBCHANNEL, словами команд передачи, словами команд ввода-вывода и указаниями. Команда START SUBCHANNEL
5 выполняется ЦП и является частью программы ЦП, которая осуществляет управление потоком запросов для операций ввода-вывода от других программ, которые управляют или обрабатывают данные ввода-вывода.

Когда выполняется START SUBCHANNEL, параметры передаются на целевой субканал, запрашивая, чтобы канальная подсистема выполнила функцию запуска с
10 устройством ввода-вывода, связанным с субканалом. Канальная подсистема выполняет функцию запуска, используя информацию на субканале, включая информацию, переданную во время выполнения команды START SUBCHANNEL, чтобы найти доступный канальный тракт к устройству. Когда устройство было выбрано, выполнение операции ввода-вывода совершается путем декодирования и выполнения CCW
15 канальной подсистемой и устройством ввода-вывода, для канальных программ CCW или, для канальных программ TCW, - путем передачи TCCB на устройство ввода-вывода канальной подсистемой и декодирования и выполнения DCW устройством. Слова команд ввода-вывода и слова команд передачи извлекаются из основного запоминающего устройства, хотя модифицирующие биты в командном коде CCW DCW
20 могут определять зависимые от устройства ситуации для выполнения операции на устройстве.

Операции, характерные для устройства, такие как перемотка пленки или расположение устройства доступа на дисковом накопителе, определяются указаниями, которые декодируются и выполняются устройствами ввода-вывода. Указания могут
25 передаваться на устройство в виде модифицирующих битов в командном коде команды управления, могут передаваться на устройство в виде данных во время операции управления или записи, или могут получать доступ к устройству другими способами.

Инициация функции запуска

Программы ЦП инициируют операции ввода-вывода командой START
30 SUBCHANNEL. Эта команда передает содержимое блока (ORB) запроса операции на субканал. Если ORB определяет канальную программу CCW, содержимое ORB включает ключ субканала, адрес первого CCW, подлежащего выполнению, и спецификацию формата слов CCW. CCW определяет команду, подлежащую выполнению, и область памяти, если имеется, которую необходимо использовать. Если ORB определяет
35 канальную программу TCW, содержимое ORB включает ключ субканала и адрес слова TCW, подлежащего выполнению. TCW указывает TCCB, который содержит команды, подлежащие передаче на устройство для выполнения, область или области памяти, если имеются, которые надо использовать для передачи данных, и TSB, который должен содержать статус операции ввода-вывода.

40 Когда содержимое ORB было передано на субканал, выполнение START SUBCHANNEL завершается. Результаты выполнения команды обозначаются набором кодов ситуаций в слове статуса программы.

Когда средства становятся доступными, и ORB определяет канальную программу CCW, канальная подсистема извлекает первое CCW и декодирует его в соответствии с
45 битом формата, определенным в ORB. Если бит формата равен нулю, определяется формат-0 слов CCW. Если бит формата равен единице, определяется формат-1 слов CCW. Слова CCW формата-0 и формата-1 содержат одинаковую информацию, но поля в CCW формате-1 организованы иным образом, так что 31-битные адреса могут

определяться непосредственно в CCW. Когда средства становятся доступными, и ORB определяет канальную программу TCW, канальная подсистема извлекает указанное TCW и передает указанный TCCB на устройство. Области памяти, указанные TCW для передачи данных на устройство или из устройства, являются 64-битными адресами.

5 Режимы работы субканалов

10 Имеется два режима работы субканалов. Субканал входит в режим передачи, когда средство FCX установлено, и функция запуска установлена на субканале в результате выполнения команды START SUBCHANNEL, которая определяет канальную программу TCW. Субканал остается в режиме передачи, пока функция запуска в субканале не будет перезапущена. Во всех остальных случаях субканал находится в командном режиме.

Управление трактами

15 Если ORB определяет канальную программу CCW, и первое CCW проходит определенные проверки достоверности и не имеет флага приостановки, заданного в единицу, или если ORB определяет канальную программу TCW, и указанное TCW проходит определенные проверки достоверности, канальная подсистема предпринимает выбор устройства путем выбора канального тракта из группы канальных трактов, которые доступны для выбора. Устройство управления, которое распознает идентификатор устройства, логически соединяется с канальным трактом и отвечает на его выбор.

20 Если ORB определяет канальную программу CCW, канальная подсистема отправляет часть CCW с кодом команд по канальному тракту, и устройство отвечает байтом статуса, указывающим, может ли команда быть выполнена. Устройство управления в это время может логически отключаться от канального тракта или может оставаться подключенным, чтобы инициировать передачу данных.

25 Если ORB определяет канальную программу TCW, канальная подсистема использует информацию в указанном TCW, чтобы передавать TCCB на устройство управления. Содержимое TCCB игнорируется канальной подсистемой и имеет значение только для устройства управления и устройства ввода-вывода.

30 Если предпринятый выбор не происходит то ли из-за указания занятости, то ли из-за ситуации нерабочего тракта, канальная подсистема пытается выбрать устройство по альтернативному канальному тракту, если таковой доступен. Если выбор был предпринят по всем доступным для выбора трактам, а ситуация занятости сохраняется, операция остается в ожидании, пока тракт не освободится. Если ситуация нерабочего тракта обнаруживается на одном или нескольких канальных трактах, на которых 35 предпринимался выбор устройства, программа получает предупреждение последующим прерыванием ввода-вывода. Прерывание ввода-вывода возникает или после выполнения канальной программы (в предположении, что устройство было выбрано по альтернативному канальному тракту) или в результате того, что выполнение было прекращено из-за обнаружения ситуаций нерабочего тракта на всех канальных трактах, 40 на которых был предпринят выбор устройства.

Выполнение канальной программы

Если команда иницируется на устройстве, и выполнение команды не требует передачи каких-либо данных на устройство или от него, устройство может подавать сигнал о завершении операции непосредственно по получении командного кода. В 45 операциях, которые включают передачу данных, субканал устанавливается так, что канальная подсистема отвечает на служебные запросы от устройства и предполагает дальнейшее управление операций.

Операция ввода-вывода может включать передачу данных в или из одной области

памяти, указанных одним CCW или TCW, или в или из ряда прерывных областей памяти. В последнем случае обычно для выполнения операции ввода-вывода используется список слов CCW, причем каждое CCW указывает непрерывную область памяти, и слова CCW объединяются сцеплением данных. Сцепление данных определяется флагом в CCW и заставляет канальную подсистему извлекать другое CCW после исчерпания или заполнения области памяти, указанной текущим CCW. Область памяти, указанная CCW, извлеченным посредством сцепления данных, относится к уже происходящей операции ввода-вывода на устройстве ввода-вывода, и устройство ввода-вывода при извлечении нового CCW не уведомляется.

Для программиста в формате CCW дается возможность задавать, чтобы при декодировании CCW канальная подсистема запрашивала прерывание ввода-вывода как можно быстрее, тем самым уведомляя программу ЦП, что сцепление продвинулось по меньшей мере до этого CCW в канальной программе.

Для дополнения динамической трансляции адресов в центральных процессорах предоставляются косвенная адресация данных CCW и модифицированная косвенная адресация данных CCW.

Когда ORB определяет канальную программу CCW, и используется косвенная адресация данных CCW, флаг в CCW указывает, что для указания областей памяти для этого CCW должен использоваться список косвенных адресов данных. Каждый раз, когда достигается граница блока памяти, производится обращение к списку, чтобы определить следующий блок памяти, который надо использовать. ORB определяет, равен размер каждого блока памяти 2 кб или 4 кб.

Когда ORB определяет канальную программу CCW, и используется модифицированная косвенная адресация данных CCW, флаг в ORB и флаг в CCW указывают, что для указания областей памяти для этого CCW должен использоваться список модифицированных косвенных адресов данных. Каждый раз, когда достигается итоговое число байтов, определенное для блока памяти, осуществляется обращение к списку, чтобы определить следующий блок памяти, который надо использовать. В отличие от случая использования косвенной адресации данных, блок может определяться на любой границе и размером до 4 кб, при условии, что не определяется передача данных через границу 4 кб.

Когда ORB определяет канальную программу TCW и используется косвенная адресация передачи данных, флаги в TCW указывают, будет ли список передачи косвенных адресов данных использоваться, чтобы указывать области памяти, содержащие TCCB, и используется ли список передачи косвенных адресов данных, чтобы указывать области запоминающего устройства для данных, связанные со словами DCW в TCCB. Каждый раз, когда достигается итоговое число байтов, определенное для блока памяти, осуществляется обращение к соответствующему списку передачи косвенных адресов данных, чтобы определить следующий блок памяти, который надо использовать.

Косвенная адресация данных CCW и модифицированная косвенная адресация данных CCW позволяет использовать по существу те же последовательности CCW для программы, работающей с динамической трансляцией адреса, активной в ЦП, которые использовались бы, если бы ЦП работал с эквивалентным непрерывным реальным запоминающим устройством. Косвенная адресация данных CCW позволяет программе обозначать блоки данных, имеющие абсолютные адреса памяти до $2^{64}-1$, независимо от того, был ли определен в ORB формат-0 или формат-1 CCW. Косвенная адресация данных CCW позволяет программе обозначать блоки данных, имеющие абсолютные

адреса памяти до $2^{64}-1$, независимо от того, был ли определен в ORB формат-0 или формат-1 CCW.

В целом, выполнение операции ввода-вывода или цепочки операций включает целых три уровня участия:

1. За исключением эффектов, вызванных интеграцией ЦП и оборудования канальной подсистемы, ЦП занят в течение выполнения START SUBCHANNEL, которое длится, пока адресуемому субканалу не передано содержимое ORB.

2. Субканал занимается новой командой START SUBCHANNEL от получения содержимого ORB до того, как первичная ситуация прерывания не будет сброшена на субканале.

3. Устройство ввода-вывода занято от инициации первой операции на устройстве до момента, когда или субканал приостанавливается, или на субканал накладывается вторичная ситуация прерывания. В случае приостановленного субканала устройство снова становится занятым, когда выполнение приостановленной канальной программы возобновляется.

Завершение операций ввода-вывода

На завершение операции ввода-вывода, как правило, указывают две ситуации статуса: окончание работы канала и окончание работы устройства. Ситуация окончания работы канала указывает, что устройство ввода-вывода получило или предоставило все данные, связанные с операцией, и больше не нуждается в средствах канальной подсистемы. Эта ситуация называется первичной ситуацией прерывания, и окончание работы канала в этом случае является первичным статусом. Как правило, первичная ситуация прерывания - это любая ситуация прерывания, которая относится к операции ввода-вывода и которая сигнализирует о завершении операции ввода-вывода или цепочки операций ввода-вывода на субканале.

Сигнал окончания работы устройства указывает, что устройство ввода-вывода завершило выполнение и готово выполнять другую операцию. Эта ситуация называется вторичной ситуацией прерывания, и окончание работы устройства в этом случае является вторичным статусом. Как правило, вторичная ситуация прерывания - это любая ситуация прерывания, которая относится к операции ввода-вывода и которая сигнализирует о завершении операции ввода-вывода или цепочки операций на устройстве. Вторичная ситуация прерывания может возникать одновременно с первичной ситуацией прерывания или позже нее.

Параллельно с первичной или вторичной ситуациями прерывания и канальная подсистема, и устройство ввода-вывода могут предоставлять указания на необычные ситуации.

Ситуации, сигнализирующие о завершении операции ввода-вывода, могут доводиться до сведения программы прерываниями ввода-вывода или, когда центральные процессоры отключены для прерываний ввода-вывода, - программным опросом канальной подсистемы. В первом случае эти ситуации вызывают сохранение кода прерывания ввода-вывода, который содержит информацию, касающуюся источника прерывания. В последнем случае код прерывания сохраняется в результате выполнения TEST PENDING INTERRUPTION.

Когда распознается первичная ситуация прерывания, канальная подсистема пытается уведомить программу, посредством запроса прерывания, что субканал содержит информацию, описывающую завершение операции ввода-вывода на субканале. Для прерываний командного режима информация идентифицирует последнее использованное CCW и может предоставлять его остаточный счетчик байтов, таким образом описывая

размер использованной памяти основного запоминающего устройства. Для прерываний транспортного режима информация идентифицирует текущее TCW и TSB, связанный с канальной программой, который содержит дополнительный статус об операции ввода-вывода, такой как остаточный счетчик байтов. В дополнение к информации о канальной программе и канальная подсистема, и устройство ввода-вывода могут предоставлять дополнительные указания на необычные ситуации в качестве части или первичной, или вторичной ситуации прерывания.

Информация, содержащаяся на субканале, может сохраняться выполнением TEST SUBCHANNEL или выполнением STORE SUBCHANNEL. Сохраненная, эта информация называется словом статуса субканала (SCSW).

Сцепление при использовании канальной программы CCW

Когда ORB определяет канальную программу CCW, предоставляются средства для инициации программой выполнения цепочки операций ввода-вывода одной командой START SUBCHANNEL. Когда текущее CCW определяет сцепление команд, и во время операции не было обнаружено никаких необычных ситуаций, получение сигнала окончания работы устройства заставляет канальную подсистему извлекать новое CCW. Если CCW проходит определенные проверки достоверности, и флаг приостановки в новом CCW не задан в единицу, на устройстве инициируется выполнение новой команды. Если CCW не проходит проверки достоверности, новая команда не инициируется, сцепление команд подавляется, и статус, связанный с новым CCW, приводит к генерации ситуации прерывания. Если флаг приостановки задан в единицу, и это значение является достоверным по причине единичного значения в контроле приостановки, бит 4 слова 1 связанного ORB, выполнение новой команды не инициируется, и сцепление команд завершается.

Выполнение новой команды инициируется канальной подсистемой тем же образом, что и в предыдущей операции. Сигналы окончания, возникающие при завершении операции, вызванном CCW, определяющим сцепление команд, не делаются доступными программе. Когда сцеплением команд инициируется другая операция ввода-вывода, канальная подсистема продолжает выполнение канальной программы. Однако если была обнаружена необычная ситуация, сцепление команд подавляется, канальная программа прерывается, генерируется ситуация прерывания, а сигналы окончания, вызывающие прерывание, делаются доступными программе.

Функция приостановки и возобновления обеспечивает программу контролем над выполнением канальной программы. Инициация функции приостановки контролируется установкой бита контроля приостановки в ORB. Сигнал о функции приостановки подается на канальную подсистему во время выполнения канальной программы, когда бит контроля приостановки в ORB равен единице и флаг приостановки в первом CCW или в CCW, извлеченном во время сцепления команд, равен единице.

Приостановка возникает, когда канальная подсистема извлекает CCW с флагом приостановки правомерно (по причине единичного значения бита контроля приостановки в ORB) установленным в единицу. Команда в этом CCW не отправляется на устройство ввода-вывода, и устройству передается сигнал о том, что цепочка команд завершается. Последующая команда RESUME SUBCHANNEL информирует канальную подсистему, что CCW, которое вызвало приостановку, могло быть изменено и что канальная подсистема должна повторно извлечь CCW и проверить текущую установку флага приостановки. Если обнаруживается, что флаг приостановки в CCW равен нулю, канальная подсистема возобновляет выполнение цепочки команд с устройством ввода-вывода.

Сцепление при использовании канальной программы TCW

Когда ORB определяет канальную программу TCW, также предоставляются средства для инициации программой выполнения цепочки операций устройства одной командой START SUBCHANNEL. Сцепление команд может определяться для тех слов DCW, которые указаны одним TCW. Когда текущее DCW определяет сцепление команд и во время операции не было обнаружено никаких необычных ситуаций, распознавание успешного выполнения DCW приводит к обработке следующего DCW в текущем TCCB.

Если следующее DCW проходит определенные проверки достоверности, на устройстве инициируется выполнение новой команды, и это DCW становится текущим DCW. Если DCW не проходит проверки достоверности, новая команда не инициируется, сцепление команд подавляется, канальная программа прерывается, и статус, связанный с новым DCW, приводит к генерации ситуации прерывания.

Выполнение новой команды инициируется тем же образом, что и в предыдущей операции. Сигналы окончания, возникающие при завершении операции, вызванном DCW, которое не является последним определенным DCW, не делаются доступными программе. Когда сцеплением команд инициируется другая операция ввода-вывода, выполнение канальной программы продолжается. Однако если была обнаружена необычная ситуация, сцепление команд подавляется, канальная программа прерывается, генерируется условие прерывания, и программе делается доступным статус, идентифицирующий необычную ситуацию.

Преждевременное завершение операций ввода-вывода

Выполнение канальной программы может преждевременно прерываться посредством CANCEL SUBCHANNEL, HALT SUBCHANNEL или CLEAR SUBCHANNEL. Выполнение CANCEL SUBCHANNEL вынуждает канальную подсистему прерывать функцию запуска на субканале, если канальная программа не была инициирована на устройстве. Когда функция запуска прерывается выполнением CANCEL SUBCHANNEL, в ответ на команду CANCEL SUBCHANNEL канальная подсистема устанавливает код 0 ситуации.

Выполнение HALT SUBCHANNEL вынуждает канальную подсистему подавать сигнал остановки на устройство ввода-вывода и прерывать выполнение канальной программы на субканале. Когда выполнение канальной программы прерывается выполнением HALT SUBCHANNEL, программа уведомляется о завершении запросом прерывания ввода-вывода. Когда субканал находится в командном режиме, запрос прерывания генерируется, когда устройство передает статус для прерванной операции. Когда субканал находится в режиме передачи, запрос прерывания генерируется

немедленно. Однако если сигнал остановки был подан на устройство во время сцепления команд после приема окончания работы устройства, но до того, как на устройство была передана следующая команда, запрос прерывания генерируется после того, как сигнал был подан на устройство. В последнем случае поле статуса устройства SCSW содержит нули. Выполнение CLEAR SUBCHANNEL очищает субканал от признаков выполняющейся канальной программы, вынуждает канальную подсистему подавать сигнал сброса на устройство ввода-вывода и заставляет канальную подсистему генерировать запрос прерывания ввода-вывода, чтобы уведомить программу о завершении функции сброса.

Прерывания ввода-вывода

Ситуации, вызывающие запросы прерывания ввода-вывода, являются асинхронными относительно активности в центральных процессорах, и одновременно может возникать более одной ситуации. Ситуации сохраняются на субканалах, пока не сбрасываются посредством TEST SUBCHANNEL или CLEAR SUBCHANNEL, или не

переустанавливаются перезапуском системы ввода-вывода.

Когда ситуация прерывания была распознана канальной подсистемой и указана на субканале, запрос прерывания ввода-вывода переводится в ожидание для подкласса прерывания ввода-вывода, определенного на субканале. Подкласс прерывания ввода-вывода, которого ожидает прерывание, находится под программным управлением посредством использования MODIFY SUBCHANNEL. Ожидающее прерывание ввода-вывода может приниматься любым ЦП, который активирован для прерываний от своего подкласса прерываний ввода-вывода. Каждый ЦП имеет восемь битов маски, в управляющем регистре 6, которые управляют активированием этого ЦП для каждого из восьми подклассов прерываний ввода-вывода, где маска ввода-вывода, 6 бит в PSW, является главной маской прерываний ввода-вывода для ЦП.

Когда на ЦП возникает прерывание ввода-вывода, код прерывания ввода-вывода сохраняется в области связи ввода-вывода этого ЦП, и запрос прерывания ввода-вывода сбрасывается. Код прерывания ввода-вывода идентифицирует субканал, для которого прерывание было ожидающим. Ситуации, вызывающие генерацию запроса прерывания, затем могут быть явно извлечены из субканала посредством TEST SUBCHANNEL или STORE SUBCHANNEL.

Ожидающий запрос прерывания ввода-вывода также может быть сброшен посредством TEST PENDING INTERRUPTION, когда соответствующий подкласс прерывания ввода-вывода активирован, но у PSW прерывания ввода-вывода отключены, или посредством TEST SUBCHANNEL, когда ЦП отключен для прерываний ввода-вывода от соответствующего подкласса прерываний ввода-вывода. Ожидающий запрос прерывания ввода-вывода также может сбрасываться CLEAR SUBCHANNEL. Как CLEAR SUBCHANNEL, так и TEST SUBCHANNEL сбрасывают сохраненную ситуацию прерывания также и на субканале.

Как правило, если только запрос прерывания не сбрасывается CLEAR SUBCHANNEL, программа запускает TEST SUBCHANNEL, чтобы получить информацию относительно выполнения операции.

CLEAR SUBCHANNEL

Указанный субканал сбрасывается, текущая функция запуска или остановки, если имеется, прерывается на указанном субканале, и канальной подсистеме подается сигнал асинхронно выполнять функцию сброса на указанном субканале и на связанном устройстве.

Регистр 1 общего назначения содержит слово (SID) идентификации подсистемы, которое указывает субканал, который необходимо сбросить.

Если функция запуска или остановки выполняется, она прерывается на субканале.

Субканал выводится из ожидания статуса. Вся активность, как указано в поле контроля активности слова SCSW, на субканале сбрасывается, только субканал приводится в ожидание сброса. Любые работающие функции, как указано в поле контроля функций слова SCSW, на субканале сбрасываются, кроме функции сброса, которая должна быть выполнена вследствие выполнения этой команды.

Когда субканал работает в режиме передачи и установлен код 2 ситуации, ЦП может подавать сигнал канальной подсистеме асинхронно выполнять функцию опроса и завершать команду.

Канальной подсистеме подается сигнал асинхронно выполнять функцию сброса. Функция сброса кратко описывается ниже в разделе "Связанные функции", а более подробно описывается впоследствии.

Устанавливается код 0 ситуации, чтобы указывать, что действия, описанные выше,

были предприняты.

Связанные функции

Следом за выполнением CLEAR SUBCHANNEL, канальная подсистема асинхронно выполняет функцию сброса. Если позволяет ситуация, канальная подсистема выбирает канальный тракт и пытается подать сигнал сброса на устройство, чтобы прервать операцию ввода-вывода, если таковая имеет место. Затем субканал становится ожидающим статус. Ситуации, с которыми сталкивается канальная подсистема и которые препятствуют подаче сигнала сброса на устройство, не мешают субканалу становиться ожидающим статус.

Когда субканал становится ожидающим статус в результате выполнения функции сброса, передача данных, если имеет место, со связанным устройством была прервана. Хранимое SCSW, когда результирующий статус сбрасывается посредством TEST SUBCHANNEL, имеет бит функции сброса, установленный в единицу. Если канальная подсистема может определить, что сигнал сброса был подан на устройство, бит ожидания сброса в SCSW сохраняется как ноль. Иначе бит ожидания сброса сохраняется как единица, и предоставляются другие указания, более подробно описывающие ситуацию, с которой пришлось столкнуться.

Данные измерений не накапливаются, и время соединения с устройством не сохраняется в слове расширенного статуса для субканала, для функции запуска, которая прерывается посредством CLEAR SUBCHANNEL.

Особые ситуации

Код 3 ситуации устанавливается, и никакое другое действие не предпринимается, когда субканал не является рабочим для CLEAR SUBCHANNEL. Субканал не является рабочим для CLEAR SUBCHANNEL, когда субканал не предоставляется в канальной подсистеме, не имеет приписанного к нему достоверного номера устройства или не активирован.

CLEAR SUBCHANNEL может сталкиваться с программными исключительными ситуациями, описанными или перечисленными ниже. Когда средство нескольких наборов субканалов не установлено, биты 32-47 регистра 1 общего назначения должны содержать шестнадцатеричное значение 0001; иначе распознается исключительная ситуация операнда.

Когда средство нескольких наборов субканалов установлено, биты 32-44 регистра 1 общего назначения должны содержать нули, биты 45-46 должны содержать достоверное значение, а бит 47 должен содержать единицу; иначе распознается исключительная ситуация операнда.

Результирующий код ситуации:

0 Функция инициирована

1 -

2 -

3 Нерабочий

Программные исключительные ситуации:

- Операнда

- Приоритетной операции

Функция сброса

Следом за выполнением CLEAR SUBCHANNEL, канальная подсистема выполняет функцию сброса. Выполнение функции сброса состоит в (1) выполнении операции управления трактами, (2) изменении полей на субканале, (3) подаче сигнала сброса на связанное устройство и (4) переводе субканала в ожидание статуса, что указывает на

завершение функции сброса.

Управление трактами в функции сброса

Операция управления трактами выполняется как часть функции сброса, чтобы проверить ситуации канального тракта для связанного субканала и попробовать выбрать доступный канальный тракт, по которому связанному устройству можно подать сигнал сброса.

Ситуации канального тракта проверяются в следующем порядке:

1. Если канальная подсистема активно осуществляет связь или пытается установить активную связь с устройством, на которое необходимо передать сигнал, выбирается используемый канальный тракт.

2. Если канальная подсистема находится в процессе получения указания на завершение занятости (которое не приведет к распознаванию ситуации прерывания) от устройства, на которое необходимо передать сигнал, и связанный субканал не привязан ни к какому канальному тракту, выбирается используемый канальный тракт.

3. Если связанный субканал имеет назначенную привязку для канального тракта, выбирается этот канальный тракт.

4. Если связанный субканал имеет работающую привязку для одного или нескольких канальных трактов, выбирается один из этих канальных трактов.

5. Если связанный субканал не имеет привязки ни для какого канального тракта, если указан последний использованный канальный тракт и если этот канальный тракт доступен для выбора, выбирается этот канальный тракт. Если этот канальный тракт недоступен для выбора, то либо не выбирается никакой канальный тракт, либо канальный тракт выбирается из набора канальных трактов, если таковые имеются, которые доступны для выбора (как будто не был указан последний использованный канальный тракт).

6. Если связанный субканал не имеет привязки ни для какого канального тракта, если не указан последний использованный канальный тракт и если существуют один или несколько канальных трактов, которые доступны для выбора, выбирается один из этих канальных трактов.

Если ни одна из перечисленных выше ситуаций для канальных трактов не имеет места, не выбирается никакой канальный тракт.

Для пункта 4, для пункта 5 при указанных условиях, и для пункта 6, канальная подсистема выбирает канальный тракт из набора канальных трактов. В этих случаях канальная подсистема может пытаться выбрать канальный тракт при условии, что следующие ситуации не имеют места:

1. Для канального тракта существует терминальная ситуация канального тракта.

2. Для параллельного канального тракта или канального тракта ESCON: Другой субканал имеет активную привязку для канального тракта.

Для канального тракта FICON: Канальный тракт в настоящее время используется, чтобы активно поддерживать связь с максимальным числом субканалов, которые могут поддерживать одновременные активные процессы связи.

3. Устройство, на которое необходимо передать сигнал, подключено к устройству управления типа-1, и субканал для другого устройства, подключенного к тому же устройству управления, имеет привязку к тому же канальному тракту, кроме случаев, когда привязка является работающей привязкой и первичный статус был принят этим субканалом.

4. Устройство, на которое необходимо передать сигнал, подключено к устройству управления типа-3, и субканал для другого устройства, подключенного к тому же

устройству управления, имеет назначенную привязку к тому же каналному тракту.

Изменение субканалов функцией сброса

Контрольные указания управления трактами на субканале изменяются во время выполнения функции сброса. Фактически, это изменение происходит после попытки
5 выбрать каналный тракт, но до попытки выбрать устройство для подачи сигнала сброса. Изменяемые контрольные указания управления трактами следующие:

1. Состояние всех восьми возможных каналных трактов на субканале устанавливается в рабочее для субканала.

2. Указание последнего использованного тракта переустанавливается, чтобы не
10 указывать никакого последнего использованного каналного тракта.

3. Ситуации нерабочего тракта, если имеются, - переустанавливаются.

Передача сигналов и завершение в функции сброса

Следом за попыткой выбрать каналный тракт и изменением контрольных полей управления трактами, каналная подсистема, если позволяет ситуация, пытается выбрать
15 устройство, чтобы подать сигнал сброса. Ситуации, связанные с субканалом и выбранным каналным трактом, если имеется, влияют на то, (1) делается ли попытка подать сигнал сброса, и (2) является ли попытка подать сигнал сброса успешной.

Независимо от этих ситуаций субканал затем устанавливается в ожидание статуса, и выполнение функции сброса завершается. Эти ситуации и их влияние на функцию сброса
20 описываются ниже:

Попытка подать сигнал сброса не предпринимается: Канальная подсистема не пытается подать сигнал сброса на устройство, если существует любая из следующих ситуаций:

1. Никакой каналный тракт не был выбран

25 2. Выбранный каналный тракт больше недоступен для выбора.

3. Для выбранного каналного тракта существует терминальная ситуация каналного тракта.

4. Для параллельных каналных трактов и каналных трактов ESCON: Выбранный каналный тракт в настоящее время используется для активной поддержки связи с
30 другим устройством. Для каналных трактов FICON: Выбранный каналный тракт в настоящее время используется для активной поддержки связи с максимальным количеством устройств, которые могут поддерживать одновременно активные сеансы связи.

5. Устройство, на которое необходимо передать сигнал, подключено к устройству
35 управления типа-1, и субканал для другого устройства, подключенного к тому же устройству управления, имеет привязку к тому же каналному тракту, кроме случаев, когда привязка является работающей привязкой и первичный статус был принят этим субканалом.

6. Устройство, на которое необходимо передать сигнал, подключено к устройству
40 управления типа-3, и субканал для другого устройства, подключенного к тому же устройству управления, имеет назначенную привязку к тому же каналному тракту.

Если любая из вышеуказанных ситуаций имеет место, субканал остается ожидающим сброс и устанавливается ожидающим статус, и выполнение функции сброса завершается.

Попытка подать сигнал сброса безуспешна: Когда каналная подсистема пытается
45 подать на устройство сигнал сброса, попытка может быть безуспешной из-за следующих ситуаций:

1. Когда каналная подсистема пытается выбрать устройство, чтобы подать сигнал сброса, устройство управления или устройство подает сигнал о ситуации занятости.

2. Когда канальная подсистема пытается выбрать устройство, чтобы подать сигнал сброса, распознается ситуация нерабочего тракта.

3. Когда канальная подсистема пытается подать сигнал сброса, встречается ситуация ошибки.

5 Если любая из вышеуказанных ситуаций имеет место, и канальная подсистема или определяет, что попытка подать сигнал сброса была безуспешной, или не может определить, была ли попытка успешной, субканал остается ожидающим сброс и устанавливается ожидающим статус, и выполнение функции сброса завершается.

10 Попытка подать сигнал сброса успешна: Когда канальная подсистема определяет, что попытка подать сигнал сброса была успешной, субканал больше не является ожидающим сброс и устанавливается ожидающим статус, и выполнение функции сброса завершается. Когда субканал становится ожидающим статус, операция ввода-вывода, если таковая имеется, со связанным устройством была прервана.

15 Замечание для программирования: Вслед за выполнением функции сброса, любой ненулевой статус, кроме единственно окончания работы устройства управления, которое передается канальной подсистеме устройством, передается программе как незапрашиваемый предупреждающий статус. Незапрашиваемый статус, состоящий из единственно окончания работы устройства управления или нулевого состояния, программе не передается.

20 **MODIFY SUBCHANNEL**

Информация, содержащаяся в блоке (SCHIB) информации субканала, располагается в программно-изменяемых полях на субканале. В результате программа влияет, для этого субканала, на определенные особенности обработки ввода-вывода что касается функций сброса, остановки, возобновления и запуска и определенные вспомогательные функции ввода-вывода.

25 Регистр 1 общего назначения содержит слово идентификации подсистемы (SID), которое указывает субканал, который подлежит изменению как указано определенными полями блока SCHIB. Адрес второго операнда является логическим адресом SCHIB и должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации.

Операции канальной подсистемы, на которые может оказывать влияние расположение информации SCHIB в субканале, таковы:

- Обработка ввода-вывода (поле E)
- Обработка прерываний (параметр прерываний и поле ISC)
- 35 - Управление трактами (поля D, LPM и POM)
- Отслеживание и проверка адресных ограничений (показатель блока измерения, поля LM и MM)
- Контроль формата блока измерения (поле F)
- Включение режима расширенного слова измерения (поле X)
- 40 - Средство обнаружения одновременности (поле S)
- Адрес блока измерения (MBA)

Биты 0, 1, 6 и 7 слова 1 и биты 0-28 слова 6 операнда SCHIB должны быть нулями, а биты 9 и 10 слова 1 не должны быть оба единицами. Когда расширенное средство блока измерения ввода-вывода установлено и определен блок измерения формата-1, биты 26-31 слова 11 должны быть установлены в нули. Когда расширенное средство блока измерения ввода-вывода не установлено, бит 29 слова 6 должен быть установлен в ноль; иначе распознается исключительная ситуация операнда. Когда расширенное средство слова измерения ввода-вывода не установлено, или установлено, но не

активировано, бит 30 слова 6 должен быть установлен в ноль; иначе распознается исключительная ситуация операнда. Остальные поля SCHIB игнорируются и не оказывают влияния на обработку MODIFY SUBCHANNEL.

Устанавливается код 0 ситуации, чтобы указывать, что информация из SCHIB была размещена в программно-изменяемых полях на субканале, за исключением того, что когда бит (V) достоверности номера устройства на указанном субканале равен нулю, то устанавливается код 0 ситуации, а информация из SCHIB не размещается в программно-изменяемых полях.

Особые ситуации

Код 1 ситуации устанавливается и никакое другое действие не производится, когда субканал ожидает статус.

Код 2 ситуации устанавливается и никакое другое действие не производится, когда на субканале выполняется функция сброса, остановки или запуска.

Код 3 ситуации устанавливается, и никакое другое действие не предпринимается, когда субканал не является рабочим для MODIFY SUBCHANNEL. Субканал является нерабочим для MODIFY SUBCHANNEL, когда субканал не предоставляется в канальной подсистеме.

MODIFY SUBCHANNEL может сталкиваться с программными исключительными ситуациями, описанными или перечисленными ниже.

В слове 1 SCHIB биты 0, 1, 6 и 7 должны быть нулями и, когда установлено средство проверки адресных ограничений, биты 9 и 10 не должны оба быть единицами. В слове 6 SCHIB биты 0-28 должны быть нулями. Иначе распознается исключительная ситуация операнда.

Когда расширенное средство блока измерения ввода-вывода установлено и определен блок измерения формата-1, биты 26-31 слова 11 должны быть установлены в нули; иначе распознается исключительная ситуация операнда. Когда расширенное средство блока измерения ввода-вывода не установлено, бит 29 слова 6 должен быть установлен в ноль; иначе распознается исключительная ситуация операнда. Когда расширенное средство слова измерения ввода-вывода не установлено, или установлено, но не активировано, бит 30 слова 6 должен быть установлен в ноль; иначе распознается исключительная ситуация операнда.

Когда средство нескольких наборов субканалов не установлено, биты 32-47 регистра 1 общего назначения должны содержать шестнадцатеричное значение 0001; иначе распознается исключительная ситуация операнда.

Когда средство нескольких наборов субканалов установлено, биты 32-44 регистра 1 общего назначения должны содержать нули, биты 45-46 должны содержать достоверное значение, а бит 47 должен содержать единицу; иначе распознается исключительная ситуация операнда.

Второй операнд должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации. Выполнение MODIFY SUBCHANNEL подавляется во всех исключительных ситуациях адресации и защиты.

Результирующий код ситуации:

0 Функция завершена

1 Ожидание статуса

2 Занят

3 Нерабочий

Программные исключительные ситуации:

- Доступ (извлечение, операнд 2)

- Операнда
- Приоритетной операции
- Спецификации

Замечания для программирования:

5 1. Если устройство подает сигнал предупреждения об ошибке ввода-вывода, в то время как связанный субканал отключен, канальная подсистема подает сигнал сброса на устройство и отвергает указание предупреждения об ошибке ввода-вывода без генерации ситуации прерывания ввода-вывода.

10 2. Если устройство передает незапрошенный статус, в то время как связанный субканал отключен, это состояние отвергается канальной подсистемой без генерации ситуации прерывания ввода-вывода. Однако если переданный статус содержит проверку блока, канальная подсистема подает сигнал сброса для связанного субканала и не генерирует ситуации прерывания ввода-вывода. Это необходимо принимать во внимание, когда программа использует MODIFY SUBCHANNEL для активации субканала.

15 Например, среда на связанном устройстве, которая присутствовала, когда субканал стал неактивным, могла быть заменена и, следовательно, программе следует проверить целостность этой среды.

3. Рекомендуется, чтобы программа обследовала содержимое субканала посредством последующей подачи STORE SUBCHANNEL, когда MODIFY SUBCHANNEL
20 устанавливает код 0 ситуации. Использование STORE SUBCHANNEL представляет собой способ для определения, был ли изменен указанный субканал. Неудача обследования субканала после установки кода 0 ситуации посредством MODIFY SUBCHANNEL может приводить к ситуациям, появления которых программа не ожидает.

25 START SUBCHANNEL

Канальной подсистеме передается сигнал асинхронно выполнить функцию запуска для связанного устройства, и параметры выполнения, которые содержатся в указанном ORB, располагаются на указанном субканале.

30 Регистр 1 общего назначения содержит слово идентификации подсистемы, которое указывает субканал, который необходимо запустить. Адрес второго операнда является логическим адресом ORB и должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации.

Параметры выполнения, содержащиеся в ORB, располагаются на субканале.

35 При выполнении START SUBCHANNEL субканал является ожидающим статус только со вторичным статусом, и бит (L) формата слова расширенного статуса равен нулю, ситуация ожидания статуса на субканале отбрасывается.

Субканал делается ожидающим запуск, и на субканале указывается функция запуска. Если второй операнд указывает ORB командного режима, субканал остается в командном режиме. Если второй операнд указывает ORB режима передачи, субканал
40 входит в режим передачи. Когда субканал входит в режим передачи, LPUM устанавливается в ноль, если не существует предыдущей специальной привязки; иначе LPUM не меняется.

Логически до установки кода 0 ситуации, ситуации нерабочего тракта на субканале, если таковые имеются, сбрасываются.

45 Канальной подсистеме подается сигнал асинхронно выполнять функцию запуска. Функция запуска кратко описывается ниже в разделе "Связанные функции", а подробно описывается впоследствии.

Устанавливается код 0 ситуации, чтобы указывать, что действия, описанные выше,

были предприняты.

Связанные функции

После выполнения START SUBCHANNEL канальная подсистема асинхронно выполняет функцию запуска.

5 Содержимое ORB, отличающееся от полей, которые должны содержать все нули, проверяется на достоверность. По некоторым моделям поля ORB, которые должны содержать нули, проверяются асинхронно, а не во время выполнения команды. Когда недостоверные поля обнаруживаются асинхронно, субканал становится ожидающим статус с первичным, вторичным и предупреждающим статусом и с указанными кодом
10 1 отложенной ситуации и программной проверкой. В этом случае операция ввода-вывода или цепочка операций ввода-вывода на устройстве не инициируется, а ситуация указывается битом ожидания запуска, сохраняемым как единица, когда SCSW сбрасывается выполнением TEST SUBCHANNEL.

По некоторым моделям доступность тракта проверяется асинхронно, а не во время
15 выполнения команды. Когда никакой канальный тракт недоступен для выбора, субканал становится ожидающим статус с первичным и вторичным статусами и с указанным кодом 3 отложенной ситуации. Операция ввода-вывода или цепочка операций ввода-вывода на устройстве не инициируется, и эта ситуация указывается битом ожидания запуска, сохраняемым как единица, когда SCSW сбрасывается выполнением TEST
20 SUBCHANNEL.

Если позволяет ситуация, выбирается канальный тракт, и инициируется выполнение канальной программы, которая указана в ORB.

Особые ситуации

Код 1 ситуации устанавливается, и никакое другое действие не производится, когда
25 субканал является ожидающим статус, когда выполняется START SUBCHANNEL. По некоторым моделям код 1 ситуации не устанавливается, когда субканал является ожидающим статус только со вторичным статусом; вместо этого ситуация ожидания статуса отклоняется.

Код 2 ситуации устанавливается и никакое другое действие не производится, когда
30 на субканале выполняется функция запуска, остановки или сброса.

Код 3 ситуации устанавливается, и никакое другое действие не предпринимается, когда субканал не является рабочим для START SUBCHANNEL. Субканал является нерабочим для START SUBCHANNEL, если субканал не предоставлен в канальной системе, не имеет связанного с ним достоверного номера устройства или не активирован.

35 Субканал также является нерабочим для START SUBCHANNEL, по некоторым моделям, когда для выбора недоступен никакой канальный тракт. По этим моделям отсутствие доступного канального тракта обнаруживается как часть выполнения START SUBCHANNEL. По другим моделям доступность канального тракта проверяется только как часть асинхронной функции запуска.

40 START SUBCHANNEL может сталкиваться с программными исключительными ситуациями, описанными или перечисленными ниже.

В слове 1 ORB командного режима биты 26-30 должны быть нулями, а в слове 2 ORB командного режима бит 0 должен быть нулем. Иначе, по некоторым моделям, распознается исключительная ситуация операнда. По другим моделям генерируется
45 ситуация прерывания ввода-вывода, обозначающая программную проверку, как часть асинхронной функции запуска.

START SUBCHANNEL также может сталкиваться с программными исключительными ситуациями, перечисленными ниже. Когда средство нескольких наборов субканалов

не установлено, биты 32-47 регистра 1 общего назначения должны содержать шестнадцатеричное значение 0001; иначе распознается исключительная ситуация операнда.

Когда средство нескольких наборов субканалов установлено, биты 32-44 регистра 1 общего назначения должны содержать нули, биты 45-46 должны содержать достоверное значение, а бит 47 должен содержать единицу; иначе распознается исключительная ситуация операнда.

Второй операнд должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации. Выполнение START SUBCHANNEL подавляется во всех исключительных ситуациях адресации и защиты.

Результирующий код ситуации:

0 Функция инициирована

1 Ожидание статуса

2 Занят

3 Нерабочий

Программные исключительные ситуации:

- Доступ (извлечение, операнд 2)

- Операнда

- Приоритетной операции

- Спецификации

Функция запуска и функция возобновления

Функции запуска и возобновления иницируют операции ввода-вывода, как описано ниже. Функция запуска применяется для субканалов, работающих или в командном режиме, или в режиме передачи. Функция возобновления применяется только для субканалов, работающих в командном режиме.

После выполнения START SUBCHANNEL и RESUME SUBCHANNEL канальная подсистема выполняет функции запуска и возобновления, соответственно, чтобы инициировать операцию ввода-вывода со связанным устройством. Исполнение функции запуска или возобновления состоит из: (1) выполнения операции управления трактами, (2) выполнения операции ввода-вывода или цепочки операций ввода-вывода со связанным устройством, и (3) перевода субканала в ожидание статуса, указывая на завершение функции запуска. Функция запуска иницирует выполнение канальной программы, которая указана в ORB, который, в свою очередь, указан в операнде START SUBCHANNEL, в отличие от функции возобновления, которая иницирует выполнение отложенной канальной программы, если имеется, начиная в CCW, которое вызвало приостановку; иначе функция возобновления выполняется так, как если бы она была функцией запуска.

Управление трактами в функции запуска и функции возобновления

Операция управления трактами выполняется канальной подсистемой во время исполнения функции запуска или возобновления, чтобы выбрать доступный канальный тракт, который может быть использован для выбора устройства, чтобы инициировать операцию ввода-вывода с этим устройством. Предпринимаемые действия таковы:

1. Если субканал в текущее время ожидает старта и устройство активно, функция запуска остается ожидающей на субканале, пока вторичный статус для предыдущей функции запуска не принимается от связанного устройства, и субканал делается единственно ожидающим запуск. Когда статус принимается и не описывает ситуации предупреждающего прерывания, субканал не делается ожидающим статус, и затем иницируется выполнение ожидающей функции запуска. Если статус описывает ситуацию

предупреждающего прерывания, субканал становится ожидающим статус со вторичным статусом и предупреждающим статусом, ожидающая функция запуска не инициируется, устанавливается код 1 отложенной ситуации, а бит ожидания запуска остается равным единице. Если субканал в текущее время является единственно ожидающим запуск,

инициируется выполнение функции запуска, как описано ниже.

2. Если на субканале для канального тракта существует выделенная привязка, канальная подсистема выбирает этот тракт для выбора устройства. Если при попытке выбрать устройство встречается занятая ситуация, и на субканале существует назначенная привязка, функция запуска остается в состоянии ожидания, пока для этого канального тракта не сбрасывается внутреннее указание занятости. Когда внутреннее указание занятости сбрасывается, выполнение ожидающей функции запуска инициируется на этом канальном тракте.

3. Если для выбора нет доступного канального тракта, и назначенной привязки в субканале для канального тракта не существует, канальный тракт не выбирается.

4. Если все канальные тракты, которые доступны для выбора, были испробованы, и один или несколько из них используются для активной связи с другими устройствами, или, в другом случае, если канальная подсистема на одном или нескольких из этих канальных трактов столкнулась с ситуацией занятости устройства управления или занятости устройства, или сочетанием этих ситуаций на одном или нескольких из этих канальных трактов, функция запуска остается ожидающей на субканале, пока канальный тракт, устройство управления или устройство, по необходимости, не станут доступными.

5. Если (1) функция запуска должна быть инициирована на канальном тракте с устройством, подключенным к устройству управления типа-1, и (2) никакое другое устройство не подключено к тому же устройству управления, субканал которого имеет или назначенную привязку к тому же канальному тракту, или работающую привязку к тому же канальному тракту, где первичный статус не был принят для этого субканала, то этот канальный тракт выбирается, если он доступен для выбора; иначе этот канальный тракт не выбирается. Однако если другой канальный тракт к устройству доступен для выбора, и не существует привязок, описанных выше, выбирается этот канальный тракт. Если никакой другой канальный тракт не доступен для выбора, функция запуска или возобновления, по необходимости, остается в состоянии ожидания, пока канальный тракт не станет доступным.

6. Если устройство подключено к устройству управления типа-3, и если по меньшей мере одно другое устройство подключено к тому же устройству управления, субканал которого имеет назначенную привязку к тому же канальному тракту, может быть выбран другой канальный тракт, доступный для выбора, или функция запуска остается в состоянии ожидания, пока не сбросится назначенная привязка для другого устройства.

7. Если канальный тракт был выбран, и получено указание занятости во время выбора устройства, чтобы инициировать выполнение первой команды ожидающей канальной программы CCW или чтобы передать ТССВ ожидающей канальной программы TCW, канальный тракт, по которому получено указание занятости, снова для этого устройства или устройства управления (в зависимости от полученного указания занятости устройства или устройства управления) не используется, пока внутреннее указание занятости не сбросится.

8. Если во время попытки выбора устройства, чтобы инициировать выполнение первой команды, определенной для функции запуска или подразумеваемой для функции возобновления для канальной программы CCW, или инициировать передачу ТССВ для функции запуска для канальной программы TCW (как описано в действии 7 выше),

канальная подсистема получает указание занятости, она выполняет одно из следующих действий:

а. Если устройство определено для работы в многотрактном режиме, и полученное указание занятости представляет занятость устройства, то функция запуска или возобновления остается ожидающей, пока внутреннее указание занятости не сбросится.

б. Если устройство определено для работы в многотрактном режиме, и полученное указание занятости представляет занятость устройства управления, или если устройство определено для работы в одноканальном режиме, канальная подсистема пробует выбор устройства путем выбора альтернативного канального тракта, который доступен для выбора, и продолжает операцию управления трактами, пока либо функция запуска или возобновления не иницируется, либо выбор устройства не будет испробован по всем канальным трактам, доступным для выбора. Если функция запуска или возобновления не была инициирована канальной подсистемой после того, как были перепробованы все канальные тракты, доступные для выбора, функция запуска или возобновления остается ожидающей, пока внутреннее указание занятости не сбросится.

с. Если субканал имеет назначенную привязку, то применяется действие 2, описанное на стр.15-20.

9. Когда, во время попытки выбора передать первую команду для канальной программы CCW или передать TCCB для канальной программы TCW, устройство оказывается нерабочим, а соответствующий канальный тракт является рабочим для субканала, распознается ситуация нерабочего тракта, и состояние канального тракта на субканале изменяется с рабочего для субканала на нерабочее для субканала. Ситуации нерабочего тракта на субканале, если таковые имеются, сохраняются, пока субканал затем не становится ожидающим сброс, ожидающим запуск или ожидающим возобновление (если канал был приостановлен), во время чего ситуации нерабочего тракта сбрасываются. Однако если соответствующий канальный тракт является нерабочим для субканала, ситуация нерабочего тракта не распознается. Когда устройство оказывается нерабочим во время попытки выбора передать первую команду или TCCB по канальному тракту, который доступен для выбора, происходит одно из следующих действий:

а. Если для этого канального тракта существует назначенная привязка, то это единственный канальный тракт, который доступен для выбора; следовательно, дальнейшие попытки инициировать функцию запуска или возобновления прекращаются, и распознается ситуация прерывания.

б. Если не существует назначенной привязки и имеются альтернативные канальные тракты, доступные для выбора, которые не были испробованы, один из этих канальных трактов выбирается, чтобы попробовать выбрать устройство и передать первую команду для канальной программы CCW или TCCB для канальной программы TCW.

с. Если не существует назначенной привязки, нет альтернативных канальных трактов, доступных для выбора, которые не были испробованы, и устройство оказалось рабочим на по меньшей мере одном из канальных трактов, которые были испробованы, функция запуска или возобновления остается ожидающей на субканале, пока канальный тракт, устройство управления или устройство, по необходимости, не станет доступным.

д. Если не существует назначенной привязки, нет альтернативных канальных трактов, доступных для выбора, которые не были испробованы, и устройство оказалось нерабочим на всех канальных трактах, которые были испробованы, дальнейшие попытки инициировать функцию запуска или возобновления прекращаются, и распознается ситуация прерывания.

10. Когда субканал активен и должна быть инициирована операция ввода-вывода с устройством, происходят все выборы устройства, в соответствии с указанием LPUM, если на субканале не определен многотрактный режим. Например, если для канальной программы CCW определено сцепление команд, канальная подсистема передает первую и все последующие команды, описывающие цепочку операций ввода-вывода, по одному и тому же канальному тракту.

Выполнение операций ввода-вывода

После того как канальный тракт выбран, канальная подсистема, если позволяет ситуация, инициирует выполнение операции ввода-вывода со связанным устройством. Выполнение дополнительных операций ввода-вывода может следовать за инициацией и выполнением первой операции ввода-вывода.

Для субканалов, работающих в командном режиме, канальная подсистема может выполнять семь типов команд: записать, читать, читать в обратном направлении, контролировать, опознать, опознать ID и передать в канале. Каждая команда, кроме передать в канале, инициирует соответствующую операцию ввода-вывода. За исключением периодов, когда выполнение канальной программы на субканале отложено, субканал активен от приема первой команды до распознавания на субканале первичной ситуации прерывания. Если первичная ситуация прерывания распознается до получения первой команды, субканал не становится активным. Как правило, первичная ситуация прерывания вызывается сигналом окончания работы канала или, в случае сцепления команд, - сигналом окончания работы канала для последнего CCW цепочки. Устройство является активным до распознавания на субканале вторичной ситуации прерывания. Как правило, вторичная ситуация прерывания вызывается сигналом окончания работы устройства, или, в случае сцепления команд, - сигналом окончания работы устройства для последнего CCW цепочки.

Для субканалов, работающих в режиме передачи, канальная подсистема может передавать шесть типов команд для выполнения: записать, читать, контролировать, опознать, опознать ID и опросить. Каждая команда инициирует соответствующую операцию устройства. Когда одна или несколько команд передаются на устройство ввода-вывода в ТССВ, субканал остается ожидающим старт пока не будет передан первичный статус.

Замечания для программирования:

В одноканальном режиме все передачи команд, данных и статусов для операции ввода-вывода или цепочки операций ввода-вывода происходят на канальном тракте, по которому на устройство была передана первая команда.

Когда устройство имеет установленное средство динамического отсоединения, операция ввода-вывода или цепочка операций ввода-вывода может выполняться в многотрактном режиме. Чтобы работать в многотрактном режиме, MODIFY SUBCHANNEL должна была быть предварительно выполнена для субканала с битом 13 слова 1 SCHIB, заданным в единицу. Кроме того, устройство должно быть установлено для многотрактного режима выполнением определенных зависящих от модели команд, соответствующих этому типу устройства. Общие процедуры для выполнения операций многотрактного режима следующие:

1. Установка

а. Команда типа установки многотрактного режима должна быть успешно выполнена устройством на каждом канальном тракте, которому предстоит быть элементом устанавливаемой многотрактной группы; иначе многотрактный режим работы может привести к непредсказуемым результатам на субканале. Если, по какой-либо причине,

один или несколько физически доступных канальных трактов к устройству не включаются в многотрактную группу, эти канальные тракты не должны быть доступны для выбора, пока субканал работает в многотрактном режиме. Канальный тракт может делаться недоступным для выбора путем установки соответствующего бита LPM в ноль или в SCHIB до выполнения MODIFY SUBCHANNEL, или в ORB до выполнения START SUBCHANNEL.

б. Когда команда типа установки многотрактного режима передается на устройство, только один канальный тракт должен быть логически доступен, чтобы избежать выбора альтернативного канального тракта для выполнения этой функции запуска; иначе ситуации занятости устройства могут быть обнаружены канальной подсистемой на более чем одном канальном тракте, что может привести к непредсказуемым результатам для последующих операций многотрактного режима. Этот тип процедуры установки следует использовать всякий раз, когда изменяется количество элементов многотрактной группы.

2. Выход из многотрактного режима

Чтобы выйти из многотрактного режима и продолжить обработку в одното́рактном режиме, может использоваться любая из следующих двух процедур:

а. Команда типа расформирования многотрактного режима может выполняться для любого канального тракта многотрактной группы. После этой команды должны следовать или (1) выполнение MODIFY SUBCHANNEL с 13 битом слова 1 SCHIB установленным в ноль, или (2) определение только одного канального тракта как логически доступного в LPM. Функция запуска не должна выполняться на субканале, работающем в многотрактном режиме с множеством канальных трактов, доступных для выбора, в то время как устройство работает в одното́рактном режиме; иначе на субканале для этой функции или последующих функций запуска могут возникнуть непредсказуемые результаты.

б. Команда типа расформирования многотрактного режима выполняется на каждом канальном тракте многотрактной группы (обращение установки). После этой команды должны следовать или (1) выполнение MODIFY SUBCHANNEL с 13 битом слова 1 SCHIB установленным в ноль, или (2) определение только одного канального тракта как логически доступного в LPM. Никакая функция запуска не может выполняться на субканале, работающем в многотрактном режиме с множеством канальных трактов, доступных для выбора, в то время как устройство работает в одното́рактном режиме; иначе на субканале для этой функции или последующих функций запуска могут возникнуть непредсказуемые результаты.

Блокирование данных

Данные, записанные устройством ввода-вывода, разделяются на блоки. Размер блока зависит от устройства; например, блок может быть карточкой, строкой печати или информацией, записанной между двумя последовательными промежутками на магнитной ленте.

Максимальный размер информации, которая может быть передана в одной операции ввода-вывода, составляет один блок. Операция ввода-вывода прерывается, когда связанная область основного запоминающего устройства исчерпывается или достигается конец блока, в зависимости от того, что происходит первым. Для некоторых операций, таких как запись на магнитопротяжном устройстве или на терминале запросов, блоки не определены, и количество передаваемой информации контролируется только программой.

Блок запроса операции

Блок (ORB) запроса операции является операндом START SUBCHANNEL. ORB определяет параметры, которые должны использоваться при управлении этой конкретной функцией запуска. Эти параметры включают параметр прерывания, ключ субканала, адрес первого CCW или TCW, биты управления операцией, числа управления приоритетом и спецификацию логической доступности канальных трактов к указанному устройству.

Содержимое ORB помещается на указанном субканале во время выполнения START SUBCHANNEL, до установки кода 0 ситуации. Если выполнение приводит к ненулевому коду ситуации, содержимое ORB не помещается на указанный субканал.

Два крайних справа бита адреса ORB должны быть нулями, располагая ORB на границе слова; иначе распознается исключительная ситуация спецификации. Когда установлено средство (FCX) расширений fibre channel, контроль (B) (слово 1, бит 13) типа канальной программы ORB определяет тип канальной программы, которая указана посредством ORB. Когда B равно нулю, ORB указывает канальную программу CCW. Когда B равно единице, ORB указывает канальную программу TCW. Только устройства ввода-вывода, которые поддерживают FCX, распознают канальные программы TCW.

Если содержимое ORB, который указывает канальную программу CCW, располагается на указанном субканале во время выполнения START SUBCHANNEL, субканал остается в командном режиме. Таким образом, такой ORB также известен как ORB командного режима. Если содержимое ORB, который указывает канальную программу TCW, располагается на указанном субканале во время выполнения START SUBCHANNEL, субканал входит в режим передачи. Таким образом, такой ORB также известен как ORB режима передачи.

TEST PENDING INTERRUPTION

Код прерывания ввода-вывода для ожидающего прерывания ввода-вывода на субканале сохраняется в местоположении, указанном адресом второго операнда, и ожидающее прерывание ввода-вывода сбрасывается.

Адрес второго операнда, когда не равен нулю, является логическим адресом местоположения, в котором должен храниться код прерывания ввода-вывода из двух слов, состоящий из слов 0 и 1. Адрес второго операнда должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации.

Если адрес второго операнда равен нулю, код прерывания ввода-вывода из трех слов, состоящий из слов 0-2, сохраняется в реальных местоположениях 184-195. В этом случае защита младшего адреса и защита с контролем по ключу не используются.

В режиме регистра доступа, когда адрес второго операнда равен нулю, невозможно предсказать, происходит ли трансляция регистра доступа для регистра B2 доступа. Если трансляция происходит, результирующий элемент управления адресного пространства не используется; то есть, код прерывания по-прежнему хранится в реальных местоположениях 184-195.

Ожидающие запросы прерывания ввода-вывода принимаются только для тех подклассов прерываний ввода-вывода, которые разрешены маской подкласса прерываний ввода-вывода в управляющем регистре 6 CPU, выполняющего команду. Если не существует запросов прерывания ввода-вывода, разрешенных управляющим регистром 6, код прерывания ввода-вывода не сохраняется, положение второго операнда не изменяется, и устанавливается код 0 ситуации.

Если ожидающий запрос прерывания ввода-вывода принимается, код прерывания ввода-вывода сохраняется, ожидающий запрос прерывания ввода-вывода сбрасывается,

и устанавливается код 1 ситуации. Код прерывания ввода-вывода, который сохраняется, является таким же, какой сохранялся бы, если бы произошло прерывание ввода-вывода. Однако слова PSW не подкачиваются, как при возникновении прерывания ввода-вывода. Выполнение команды определяется следующим образом:

5 Слово идентификации подсистемы (SID):

Биты 32-63 SID располагаются в слове 0.

Параметр прерывания: Слово 1 содержит четырехбайтный параметр, который был определен программой и передан на субканал в слове 0 ORB или PMCW. Когда устройство передает предупреждающий статус, а параметр прерывания не был ранее
10 передан на субканал путем выполнения START SUBCHANNEL или MODIFY SUBCHANNEL, это поле содержит нули.

Слово идентификации прерывания: Сохраненное слово 2 содержит слово идентификации прерывания, которое дополнительно идентифицирует источник прерывания ввода-вывода. Слово 2 сохраняется только когда адрес второго операнда
15 равен нулю.

Слово идентификации прерывания определяется следующим образом:

Бит (A): Бит 0 слова идентификации прерывания определяет тип ожидающего запроса прерывания ввода-вывода, которое было сброшено. Когда бит 0 равен нулю, запрос прерывания ввода-вывода был связан с субканалом.

20 Подкласс (ISC) прерывания ввода-вывода: Позиции битов 2-4 слова идентификации прерывания содержат беззнаковое двоичное целое число, в диапазоне 0-7, которое определяет подкласс прерывания ввода-вывода, связанный с субканалом, для которого был сброшен ожидающий запрос прерывания ввода-вывода. Остальные позиции битов зарезервированы и установлены в нули.

25 Особые ситуации

TEST PENDING INTERRUPTION может сталкиваться с программными исключительными ситуациями, описанными или перечисленными ниже.

Второй операнд должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации.

30 Выполнение TEST PENDING INTERRUPTION подавляется во всех исключительных ситуациях адресации и защиты.

Результирующий код ситуации:

0 Код прерывания не сохранен

1 Код прерывания сохранен

35 2 -

3 -

Программные исключительные ситуации:

- Доступа (сохранения, операнда 2, только ненулевого адреса второго операнда)

- Приоритетной операции

40 - Спецификации

Замечания для программирования:

1. TEST PENDING INTERRUPTION следует исполнять только с нулевым адресом второго операнда, когда прерывания ввода-вывода маскированы. Иначе код прерывания ввода-вывода, сохраненный командой, может быть потерян, если возникает прерывание
45 ввода-вывода. Код прерывания ввода-вывода, который идентифицирует источник прерывания ввода-вывода, принятого после TEST PENDING INTERRUPTION, также сохраняется в местоположениях 184-195, заменяя код прерывания ввода-вывода, который был сохранен командой.

2. В режиме регистра доступа, когда адрес второго операнда равен нулю, исключительная ситуация доступа распознается, если возникает трансляция регистра доступа и регистр доступа находится в состоянии ошибки. Это исключение можно предотвратить путем установки поля B2 в ноль или путем расположения шестнадцатеричного значения 00000000, 00000001 или любого другого достоверного содержимого в регистр доступа.

STORE SUBCHANNEL

Информация управления и статуса для указанного субканала сохраняется в указанном SCHIB.

Регистр 1 общего назначения содержит слово идентификации подсистемы, которое указывает субканал, для которого необходимо сохранить информацию. Адрес второго операнда является логическим адресом SCHIB и должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации.

Когда расширенное средство блока измерения ввода-вывода не установлено, информация, которая сохраняется в SCHIB, состоит из слова контроля управления трактами, SCSW и трех слов зависящей от модели информации. Когда расширенное средство блока измерения ввода-вывода установлено, информация, которая сохраняется в SCHIB, состоит из слова контроля управления трактами, SCSW, поля адреса блока измерения и одного слова зависящей от модели информации.

Выполнение STORE SUBCHANNEL не изменяет никакой информации на субканале.

Код 0 ситуации устанавливается, чтобы указывать, что информация управления и статуса для указанного субканала была сохранена в SCHIB. Когда выполнение STORE SUBCHANNEL приводит к установке кода 0 ситуации, информация в SCHIB указывает на согласованное состояние субканала.

Особые ситуации

Код 3 ситуации устанавливается, и никакое другое действие не предпринимается, когда указанный субканал не является рабочим для STORE SUBCHANNEL. Субканал является нерабочим для STORE SUBCHANNEL, если субканал не предоставлен в канальной подсистеме.

[0004] STORE SUBCHANNEL может встречать программные исключительные ситуации, описанные или перечисленные ниже.

Когда средство нескольких наборов субканалов не установлено, биты 32-47 регистра 1 общего назначения должны содержать шестнадцатеричное значение 0001; иначе распознается исключительная ситуация операнда.

Когда средство нескольких наборов субканалов установлено, биты 32-44 регистра 1 общего назначения должны содержать нули, биты 45-46 должны содержать достоверное значение, а бит 47 должен содержать единицу; иначе распознается исключительная ситуация операнда.

Второй операнд должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации.

Результирующий код ситуации:

0 SCHIB сохранен

1 -

2 -

3 Нерабочий

Программные исключительные ситуации:

- Доступа (сохранения, операнда 2)

- Операнда

- Приоритетной операции
- Спецификации

Замечания для программирования:

1. Статус устройства, которое сохраняется в SCSW, может включать указания на
5 занятость устройства, занятость устройства управления или окончания работы устройства управления.

2. Информация, которая сохраняется в SCHIB, получается из субканала. Команда STORE SUBCHANNEL не приводит к опросу канальной подсистемой адресуемого устройства.

10 3. STORE SUBCHANNEL может выполняться в любое время, чтобы замерить ситуации, существующие на субканале, не вызывая сброса никаких ожидающих статус ситуаций.

4. Повторяющегося выполнения STORE SUBCHANNEL без промежуточной задержки (например, чтобы определить, когда субканал меняет состояние) следует избегать, поскольку повторяющиеся обращения к субканалу со стороны ЦП могут задерживать
15 или препятствовать доступу к субканалу со стороны канальной системы с целью обновления субканала.

TEST SUBCHANNEL

Информация управления и статуса для субканала сохраняется в указанном IRB. Регистр 1 общего назначения содержит слово идентификации подсистемы, которое
20 указывает субканал, для которого необходимо сохранить информацию. Адрес второго операнда является логическим адресом IRB и должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации.

Информация, которая сохраняется в IRB, состоит из SCSW, слова расширенного статуса и слова расширенного управления.

25 Если субканал является ожидающим статус, бит ожидания статуса поля управления статусом устанавливается в единицу. То, является или нет субканал ожидающим статус, оказывает влияние на функции, которые выполняются при выполнении TEST SUBCHANNEL.

30 Когда субканал является ожидающим статус и выполняется TEST SUBCHANNEL, информация, как описано выше, сохраняется в IRB, после чего следует сброс определенных ситуаций и указаний, которые существуют на субканале. Если субканал находится в режиме передачи, сброс этих ситуаций, особенно функции запуска, помещает субканал в командный режим. Если запрос прерывания ввода-вывода является ожидающим для субканала, запрос сбрасывается. Код 0 ситуации устанавливается,
35 чтобы указать, что эти действия были предприняты.

Когда субканал не является ожидающим статус и выполняется TEST SUBCHANNEL, информация сохраняется в IRB, и никакие ситуации или указания не сбрасываются. Код 1 ситуации устанавливается, чтобы указать, что эти действия были предприняты.

Особые ситуации

40 Код 3 ситуации устанавливается, и никакое другое действие не предпринимается, когда субканал не является рабочим для TEST SUBCHANNEL. Субканал является нерабочим для TEST SUBCHANNEL, если субканал не предоставлен, не имеет достоверного номера устройства, связанного с ним, или не активирован.

45 TEST SUBCHANNEL может сталкиваться с программными исключительными ситуациями, описанными или перечисленными ниже.

Когда средство нескольких наборов субканалов не установлено, биты 32-47 регистра 1 общего назначения должны содержать шестнадцатеричное значение 0001; иначе распознается исключительная ситуация операнда.

Когда средство нескольких наборов субканалов установлено, биты 32-44 регистра 1 общего назначения должны содержать нули, биты 45-46 должны содержать достоверное значение, а бит 47 должен содержать единицу; иначе распознается исключительная ситуация операнда.

5 Второй операнд должен быть указан на границе слова; иначе распознается исключительная ситуация спецификации.

Когда выполнение TEST SUBCHANNEL прерывается в исключительных ситуациях адресации и защиты, состояние субканала не меняется. Результирующий код ситуации:

0 IRB сохранен; субканал ожидает статус

10 1 IRB сохранен; субканал не ожидает статус

2 -

3 Нерабочий

Программные исключительные ситуации:

- Доступа (сохранения, операнда 2)

15 - Операнда

- Приоритетной операции

- Спецификации

Замечания для программирования:

1. Статус устройства, которое сохраняется в SCSW, может включать указания на
20 занятость устройства, занятость устройства управления или окончания работы устройства управления.

2. Информация, которая сохраняется в IRB, получается из субканала. Команда TEST SUBCHANNEL не вынуждает канальную подсистему опрашивать адресуемое устройство.

3. Когда происходит прерывание ввода-вывода, это является результатом ситуации
25 ожидания статуса на субканале, и, как правило, TEST SUBCHANNEL выполняется, чтобы сбросить статус. TEST SUBCHANNEL также может выполняться в любое другое время, чтобы замерить ситуации, существующие на субканале.

4. Повторяющегося выполнения TEST SUBCHANNEL, чтобы определить, когда
была завершена функция запуска, следует избегать, поскольку существуют ситуации,
30 в которых завершение функции запуска может быть или не может быть инициировано. Например, если канальная подсистема удерживает ситуацию проверки управления интерфейса (IFCC) в ожидании (для какого-либо субканала), поскольку другой субканал уже является ожидающим статус, и если функция запуска, проверяемая посредством TEST SUBCHANNEL, имеет в качестве единственного доступного для выбора тракта
35 канальный тракт с ситуацией IFCC, то стартовая функция не может быть инициирована до тех пор, пока ситуация ожидания статуса в другом субканале не будет сброшена, позволяя ситуации IFCC быть указанной на субканале, к которому она применяется.

5. Повторяющегося выполнения TEST SUBCHANNEL без промежуточной задержки, например, чтобы определить, когда субканал меняет состояние, следует избегать,
40 поскольку повторяющиеся обращения к субканалу со стороны ЦП могут задерживать или препятствовать доступу к субканалу со стороны канальной системы. Выполнение TEST SUBCHANNEL множеством центральных процессоров для одного субканала примерно в одно время может иметь тот же эффект, и его также следует избегать.

6. Приоритет обработки прерываний ввода-вывода ЦП можно менять посредством
45 выполнения TEST SUBCHANNEL. Когда выполняется TEST SUBCHANNEL, и указанный субканал имеет ожидающий запрос прерывания ввода-вывода, этот запрос прерывания ввода-вывода сбрасывается, и SCSW сохраняется, в независимости от какого-либо ранее установленного приоритета. Относительный приоритет остальных запросов

прерываний ввода-вывода остается неизменным.

Как будет понятно специалисту в данной области техники, особенности настоящего изобретения могут быть реализованы системой, способом или компьютерным программным продуктом. Соответственно, особенности настоящего изобретения могут
 5 иметь форму полностью аппаратного варианта осуществления, полностью программного варианта осуществления (включая встроенное программное обеспечение, резидентное программное обеспечение, микрокод и т.п.) или варианта осуществления, сочетающего программные и аппаратные особенности, которые все в данном документе в целом могут называться "контуром", "модулем" или "системой". Кроме того,
 10 особенности настоящего изобретения могут иметь форму компьютерного программного продукта, реализованного в одной или нескольких машиночитаемой среде (-ах), с реализованным на них машиночитаемым программным кодом.

Может использоваться любое сочетание одной или нескольких машиночитаемых сред. Машиночитаемая среда может быть машиночитаемой запоминающей средой.
 15 Машиночитаемая запоминающая среда может быть, например, без ограничения, электронной, магнитной, оптической, электромагнитной, инфракрасной или полупроводниковой системой, аппаратом или устройством, или любым подходящим сочетанием вышеуказанного. Более конкретные примеры (не исчерпывающий список) машиночитаемых запоминающих сред включают следующее: электрическое соединение,
 20 имеющее один или несколько проводов, съемную дискету для компьютера, жесткий диск, оперативное запоминающее устройство (ОЗУ), постоянное запоминающее устройство (ПЗУ), стираемое программируемое постоянное запоминающее устройство (СПЗУ или флэш-память), оптоволоконно, постоянное запоминающее устройство на съемном диске (CD-ROM), оптическое запоминающее устройство, магнитное
 25 запоминающее устройство или любое подходящее сочетание вышеуказанного. В контексте этого документа машиночитаемая запоминающая среда может быть любой материальной средой, которая может содержать или хранить программу для использования посредством системы, аппарата или устройства выполнения команд или в связи с ними.

30 Как показано на фиг.10, в одном примере компьютерный программный продукт 1000 содержит, например, одну или несколько постоянных машиночитаемых запоминающих сред 1002 для хранения в них машиночитаемого программного кода или логики 1004 для обеспечения и улучшения одной или нескольких особенностей настоящего изобретения.

35 Программный код, воплощенный в машиночитаемой среде, может передаваться с использованием соответствующей среды, включая без ограничения беспроводную, проводную линию, оптоволоконный кабель, ВЧ-среду и т.д. или любое подходящее сочетание вышеупомянутого.

Компьютерный программный код для выполнения операций, обеспечивающих
 40 особенности настоящего изобретения, может быть написан на любом сочетании одного или нескольких языков программирования, включая объектно-ориентированный язык программирования, такой как Java, Smalltalk, C++ и т.п., и традиционные процедурные языки программирования, такие как язык программирования C, ассемблер или подобные языки программирования. Программный код может полностью выполняться на
 45 компьютере пользователя, частично на компьютере пользователя, в качестве автономного пакета программного обеспечения, частично на компьютере пользователя и частично на удаленном компьютере или полностью удаленном компьютере или сервере. В последнем случае удаленный компьютер может быть соединен с компьютером

пользователя посредством сети любого типа, включая локальную вычислительную сеть (LAN) или глобальную вычислительную сеть (WAN), или соединение может осуществляться с внешним компьютером (например, по сети Интернет с помощью Интернет-провайдера).

5 Особенности настоящего изобретения описаны в данном документе со ссылкой на структурные схемы и/или блок-схемы способов, устройств (систем) и компьютерных программных продуктов в соответствии с вариантами осуществления изобретения. Будет понятно, что каждый блок на структурных схемах и/или блок-схемах и сочетания
10 блоков на структурных схемах и/или блок-схемах могут быть реализованы посредством команд управления компьютерных программ. Эти команды управления компьютерной программой могут подаваться на процессор компьютера общего назначения, компьютера специального назначения или другого программируемого устройства для
15 обработки данных с целью создания машины, в которой команды, которые выполняются посредством процессора компьютера или другого программируемого устройства для обработки данных, создают средство для реализации функций/действий, определенных блоком или блоками на структурных схемах и/или блок-схемах.

Эти команды управления компьютерной программой также могут храниться в машиночитаемой среде, которая может предписывать компьютеру, другому программируемому устройству обработки данных или другим устройствам действовать
20 конкретным образом, так что команды, хранящиеся в машиночитаемой среде, образуют изделие обработки, содержащее команды, которые реализуют функцию/действие, определенные блоком или блоками на структурных схемах и/или блок-схемах.

Команды управления компьютерной программой также могут загружаться в компьютер, другое программируемое устройство обработки данных или другие
25 устройства, чтобы вызывать выполнение последовательности оперативных этапов компьютером, другим программируемым устройством или другими устройствами, чтобы создавать компьютерно-реализованный процесс, так что команды, которые выполняются на компьютере или другом программируемом устройстве, обеспечивают процессы для реализации функций/действий, заданных блоком или блоками на
30 структурных схемах и/или блок-схемах.

Приведенные на чертежах структурные схемы и блок-схемы иллюстрируют архитектуру, функциональные возможности и работу возможных осуществлений систем, способов и компьютерных программных продуктов в соответствии с различными вариантами осуществления настоящего изобретения. В связи с этим каждый блок на
35 структурных схемах или блок-схемах может представлять модуль, сегмент или часть кода, которая содержит одну или несколько выполняемых команд для реализации заданной логической функции(-й). Следует также отметить, что в некоторых альтернативных осуществлениях указанные в блоке функции могут происходить не в том порядке, в котором они представлены на графических материалах. Например, два
40 блока, показанные последовательно, на самом деле могут выполняться в значительной мере одновременно, или блоки могут иногда выполняться в обратном порядке, в зависимости от связанных функциональных возможностей.

Следует также отметить, что каждый блок на блок-схемах и/или структурных схемах и сочетания блоков на блок-схемах и/или структурных схемах могут быть реализованы
45 посредством специализированных аппаратных систем, которые выполняют заданные функции или действия, или сочетаний специализированного аппаратного обеспечения и компьютерных команд.

В придачу к вышесказанному, одна или несколько особенностей настоящего

изобретения может предоставляться, предлагаться, вводиться, управляться, обслуживаться и т.п. поставщиком услуг, который предлагает управление пользовательскими средами. Например, поставщик услуг может создавать, поддерживать, обслуживать и т.п. компьютерный код и/или вычислительную инфраструктуру, которая выполняет одну или несколько особенностей настоящего изобретения для одного или нескольких пользователей. В ответ поставщик услуг может получать оплату от пользователя, например, согласно подписке и/или соглашению. Дополнительно или в качестве альтернативы, поставщик услуг может получать плату за продажу рекламы одной или нескольким третьим сторонам.

Согласно одной из особенностей настоящего изобретения для выполнения одной или нескольких особенностей настоящего изобретения может быть развернуто приложение. В качестве одного из примеров, разворачивание приложения включает предоставление вычислительной инфраструктуры, способной выполнять одну или несколько особенностей настоящего изобретения.

В качестве еще одной особенности настоящего изобретения, разворачивание вычислительной инфраструктуры может включать интегрирование машиночитаемого кода в вычислительную систему, в которой код в сочетании с вычислительной системой способен выполнять одну или несколько особенностей настоящего изобретения.

В качестве еще одной особенности настоящего изобретения может быть представлен способ интегрирования вычислительной инфраструктуры, включающий интегрирование машиночитаемого кода в компьютерную систему. Компьютерная система содержит машиночитаемую среду, причем машиночитаемая среда содержит одну или несколько особенностей настоящего изобретения. Код в сочетании с компьютерной системой способен выполнять одну или несколько особенностей настоящего изобретения.

Хотя выше описаны различные варианты осуществления, они являются лишь примерами. Например, вычислительные среды других архитектур могут включать и использовать одну или несколько особенностей настоящего изобретения. Например, серверы, отличающиеся от серверов z196, могут включать, использовать и/или получать преимущества от одной или нескольких особенностей настоящего изобретения. Кроме того, могут использоваться другие команды управления и/или команды; и команды управления/команды могут включать дополнительную, меньшую и/или отличающуюся информацию, по сравнению с описанной в данном документе. Возможны многие варианты.

Выгоду от одной или нескольких особенностей настоящего изобретения могут получать и другие типы вычислительных сред. Например, может применяться система обработки данных, подходящая для хранения и/или выполнения программного кода, которая включает по меньшей мере два процессора, прямо или непрямо соединенных с элементами памяти через системную шину. Элементы памяти включают, например, локальную память, используемую во время фактического выполнения программного кода, устройство массовой памяти и кэш-память, которая обеспечивает временное хранение по меньшей мере некоторого программного кода с целью уменьшения количества извлечений кода из устройства массовой памяти во время выполнения.

Устройства ввода-вывода (включая без ограничения клавиатуры, экраны, устройства указания, DASD (устройства памяти с прямым доступом), ленту, CD диски, DVD диски, флэш-устройства и другие среды памяти, и т.п.) могут подключаться к системе или прямо, или посредством промежуточных контроллеров ввода-вывода. Сетевые адаптеры также могут подключаться к системе, чтобы позволять системе обработки данных соединяться с другими системами обработки данных или удаленным принтерам или

устройствам памяти посредством промежуточных частных или общих сетей. Модемы, кабельные модемы и сетевые карты Ethernet являются лишь некоторыми из доступных типов сетевых адаптеров.

Фиг.11 представляет собой изображение характерных компонентов хост-компьютерной системы 5000 для реализации одной или нескольких особенностей настоящего изобретения. Характерный хост-компьютер 5000 содержит один или несколько ЦП 5001, связанных с памятью (т.е. центральным запоминающим устройством) 5002 компьютера, а также интерфейсы ввода-вывода к запоминающим устройствам 5011 и сети 5010 для связи с другими компьютерами или SAN и т.п. ЦП 5001 совместим с архитектурой, имеющей структурированный набор команд и структурированные функциональные возможности. ЦП 5001 может иметь динамическую трансляцию 5003 адреса (DAT) для преобразования программных адресов (виртуальных адресов) в действительные адреса памяти. DAT обычно содержит буфер 5007 быстрого преобразования адреса (TLB) для кэширования трансляций, чтобы при последующих доступах к блоку памяти 5002 компьютера не требовалась задержка трансляции адреса. Обычно между памятью 5002 компьютера и процессором 5001 используется кэш 5009. Кэш 5009 может быть иерархическим, имеющим кэш большой емкости, доступный для более чем одного ЦП, и более быстродействующих кэшей (низкого уровня) меньшей емкости между кэшем большой емкости и каждым ЦП. В некоторых осуществлениях кэши низкого уровня разделяются, чтобы предоставлять отдельные кэши низкого уровня для выборки команд и доступа к данным. В одном варианте осуществления команда извлекается из памяти 5002 блоком 5004 выборки команд посредством кэша 5009. Команда декодируется в блоке 5006 декодирования команд и отправляется (с другими командами в некоторых вариантах осуществления) в блок или блоки 5008 выполнения команд. Обычно используется несколько блоков 5008 выполнения, например, блок выполнения арифметических команд, блок выполнения с плавающей точкой и блок выполнения команд перехода. Команда выполняется блоком выполнения, который осуществляет доступ к операндам из определяемых командами регистров или памяти по мере необходимости. Если доступ (загрузку или сохранение) к операнду необходимо осуществлять из памяти 5002, блок 5005 загрузки/сохранения обычно осуществляет доступ под управлением выполняемой команды. Команды могут выполняться в аппаратных схемах или во внутреннем микрокоде (аппаратно-программном обеспечении), или с использованием их сочетания.

Как указано, компьютерная система содержит информацию в локальном (или основном) запоминающем устройстве, а также адресные, защитные, контрольные и корректирующие записи. Некоторые особенности адресации включают формат адресов, концепцию адресных пространств, различные типы адресов и способ, которым адрес одного типа транслируется в адрес другого типа. Некоторые из основных запоминающих устройств содержат постоянно назначенные местоположения в запоминающем устройстве. Основное запоминающее устройство обеспечивает систему с запоминающим устройством с прямой адресацией и быстрым доступом к данным. В основное запоминающее устройство должны загружаться (из устройств ввода) и данные, и программы, до того как они могут быть обработаны.

Основное запоминающее устройство может содержать одно или несколько буферных запоминающих устройств меньшей емкости более быстрого доступа, иногда называемых кэшами. Кэш обычно физически связан с ЦП или процессором ввода-вывода. Физическая структура и использование различных запоминающих сред, как правило, не видимы программой, за исключением производительности.

Для команд и операндов, хранимых в памяти, могут быть предусмотрены отдельные кэши. Информация в кэше содержится в форме непрерывных байтов на целочисленной границе, называемой блоком кэша или строкой кэша (или, для краткости, - строкой). Одна модель может предоставлять команду EXTRACT CACHE ATTRIBUTE (команда
 5 извлечения атрибута кэша), которая возвращает размер строки кэша в байтах. Одна модель также может предоставлять команду PREFETCH DATA (предварительной выборки данных) и команду PREFETCH DATA RELATIVE LONG (предварительной выборки данных относительно большой длины) для предварительной выборки данных из запоминающего устройства в кэш данных или команд или для освобождения данных
 10 из кэша.

Запоминающее устройство рассматривается как длинная горизонтальная строка битов. Для большинства операций доступ к запоминающему устройству осуществляется последовательно слева направо. Строка битов подразделяется на блоки из восьми битов. Восьмибитный блок называется байтом, который является основным
 15 конструктивным блоком всех форматов представления информации. Местоположение каждого байта в запоминающем устройстве идентифицируется уникальным неотрицательным целым числом, которое является адресом местоположения этого байта или просто адресом байта. Соседние местоположения байтов имеют последовательные адреса, начинающиеся слева с 0 и последовательно следующие слева
 20 направо. Адреса представляют собой беззнаковые двоичные целые числа, содержащие 24, 31 или 64 бита.

Информация между запоминающим устройством и ЦП или канальной подсистемой передается по одному байту или группой байтов за один раз. Если не оговорено иное, например, в системе z/Architecture® хранящаяся группа байтов адресуется посредством
 25 крайнего левого байта группы. Число байтов в группе подразумевается или прямо определяется операцией, которую необходимо выполнить. Используемая в работе ЦП группа байтов называется полем. В каждой группе байтов, например, в z/Architecture®, биты последовательно нумеруются слева направо. В z/Architecture® крайние левые биты
 30 иногда именуются "старшими" битами, а крайние правые разряды - "младшими" битами. Однако номера битов не являются адресами ячеек запоминающего устройства. Обращаться по адресу можно только к байтам. Чтобы работать с отдельными битами хранящегося байта, осуществляется доступ ко всему байту. Биты в байте пронумерованы
 35 слева направо от 0 до 7 (например, в z/Architecture®). Биты в адресе могут быть пронумерованы от 8 до 31 или от 40 до 63 в случае 24-битных адресов или от 1 до 31 или от 33 до 63 в случае 31-битных адресов и от 0 до 63 в случае 64-битных адресов. В любом другом формате фиксированной длины из множества байтов биты, образующие формат, последовательно нумеруются, начиная с 0. В целях обнаружения ошибок, и
 40 предпочтительно их исправления, с каждым байтом или группой байтов может передаваться один или несколько контрольных битов. Такие контрольные биты генерируются автоматически машиной и не могут непосредственно управляться программой. Емкость запоминающего устройства выражается в числе байтов. Когда длина хранящегося поля операнда подразумевается кодом операций команды, говорят, что поле имеет фиксированную длину, которая может составлять 1, 2, 4, 8 или 16 байтов.
 45 Для некоторых команд могут подразумеваться более длинные поля. Когда длина хранящегося поля операнда не подразумевается, а прямо указывается, говорят, что поле имеет переменную длину. Операнды переменной длины могут различаться по длине с шагом в 1 байт (или для некоторых команд с шагом в 2 байта и другими шагами).

При сохранении информации в запоминающем устройстве замещается содержимое местоположений только тех байтов, которые включены в указанное поле, несмотря на то, что ширина физического пути к запоминающему устройству может быть больше длины сохраняемого поля.

5 Некоторые единицы информации должны находиться на целочисленной границе в запоминающем устройстве. Применительно к единице информации граница называется целочисленной, когда адрес ее ячейки запоминающего устройства кратен длине единицы информации в байтах. Полям длиной 2, 4, 8 и 16 байтов на целочисленной границе даются особые названия. Полуслово представляет собой группу из 2 идущих подряд байтов на двухбайтной границе и является основным структурным блоком команд. Слово представляет собой группу из 4 идущих подряд байтов на четырехбайтной границе. Двойное слово представляет собой группу из 8 идущих подряд байтов на восьмибайтной границе. Учетверенное слово представляет собой группу из 16 идущих подряд байтов на шестнадцатибайтной границе. Когда адреса запоминающего устройства обозначают полуслова, слова, двойные слова и учетверенные слова, двоичное представление адреса содержит один, два, три или четыре крайних правых нулевых бита, соответственно. Команды должны находиться на двухбайтных целочисленных границах. Хранящиеся операнды большинства команд не содержат требования выравнивания на границах.

20 В устройствах, в которых реализованы отдельные кэши для команд и операндов, может происходить значительная задержка, если программа сохраняется в строке кэша, из которой впоследствии осуществляется извлечение команд, независимо от того, изменяются ли накопителем при сохранении команды, извлечение которых осуществляется впоследствии.

25 В одном варианте осуществления изобретение может быть реализовано на практике программным обеспечением (иногда называемым лицензионным внутренним кодом, аппаратно-программным обеспечением, микрокодом, милликодом, пикокодом и т.п., любое из чего будет согласоваться с настоящим изобретением). Обращаясь к фиг.11, к программному коду программного обеспечения, которое воплощает настоящее изобретение, обычно получает доступ процессор 5001 хост-системы 5000 из 30 запоминающих устройств 5011 долговременной памяти, таких как накопитель на CD-диске, накопитель на магнитной ленте или накопитель на жестких дисках. Программный код программного обеспечения может быть воплощен в любой из множества известных сред для использования с системой обработки данных, такой как дискета, накопитель на жестких дисках или CD-диск. Код может распределяться в таких средах или может 35 распределяться пользователям из компьютерной памяти 5002 или запоминающего устройства одной компьютерной системы по сети 5010 другим компьютерным системам для использования пользователями таких других систем.

Программный код включает операционную систему, которая управляет функцией 40 и взаимодействием различных компонентов компьютера и одной или нескольких прикладных программ. Обычно программный код подкачивается по частям из запоминающего устройства 5011 в относительно более быстродействующее запоминающее устройство 5002 компьютера, в которой он доступен для обработки процессором 5001. Методы и способы для воплощения программного кода системы программного обеспечения в памяти, в физических средах и/или распределения программного кода через сети хорошо известны и более глубоко в данном документе обсуждаться не будут. Программный код, созданный и хранящийся в материальной среде (включая без ограничения модули электронной памяти (ОЗУ), флэш-память,

компакт-диски (CD), DVD-диски, магнитную ленту и т.п.) часто называют "компьютерным программным продуктом". Содержащая компьютерный программный продукт среда обычно может считываться устройством обработки данных предпочтительно в компьютерной системе для выполнения устройством обработки

данных.

Фиг.12 представляет собой изображение характерной рабочей станции или серверной аппаратной системы, в которой настоящее изобретение может быть реализовано на практике. Система 5020, представленная на фиг.12, содержит характерную базовую компьютерную систему 5021, такую как персональный компьютер, рабочая станция или сервер, включая необязательные периферийные устройства. Базовая компьютерная система 5021 имеет один или несколько процессоров 5026 и шину, используемую для соединения и связи процессора(-ов) 5026 и других компонентов системы 5021 и в соответствии с известными способами. Шина соединяет процессор 5026 с памятью 5025 и долговременным запоминающим устройством 5027, которое может включать накопитель на жестких дисках (например, включающий любое из магнитного накопителя, CD-диска, DVD-диска и флэш-памяти) или, например, накопитель на магнитной ленте. Система 5021 также может включать адаптер пользовательского интерфейса, который соединяет микропроцессор 5026 через шину с одним или несколькими интерфейсными устройствами, такими как клавиатура 5024, мышь 5023, принтер/сканнер 5030 и/или другие интерфейсные устройства, которые могут быть любыми устройствами пользовательского интерфейса, такими как сенсорный экран, клавиатура цифрового ввода и т.п. Шина также соединяет устройство 5022 отображения, такое как ЖК-экран или монитор, с микропроцессором 5026 посредством адаптера устройства отображения.

Система 5021 может поддерживать связь с другими компьютерами или компьютерными сетями посредством сетевого адаптера, способного поддерживать связь 5028 с сетью 5029. Примерами сетевых адаптеров являются каналы связи, кольцевая сеть с маркерным доступом, сеть Ethernet или модемы. Альтернативно система 5021 может поддерживать связь с помощью беспроводного интерфейса, такого как карта CDPD (сотовой системы передачи пакетов цифровых данных). Система 5021 может быть связана с другими такими компьютерами в локальной вычислительной сети (LAN) или глобальной вычислительной сети (WAN), или система 5021 может быть клиентом в клиент/серверной распределенной системе с другим компьютером, и т.п. Все эти конфигурации, а также соответствующее коммуникационное аппаратное и программное обеспечение известны в данной области техники.

Фиг.13 представляет собой изображение сети 5040 обработки данных, в которой настоящее изобретение может быть реализовано на практике. Сеть 5040 обработки данных может включать множество отдельных сетей, таких как беспроводная сеть и проводная сеть, каждая из которых может включать множество отдельных рабочих станций 5041, 5042, 5043, 5044. Кроме того, как известно специалистам в данной области техники, она может содержать одну или несколько сетей LAN, где LAN может включать множество интеллектуальных рабочих станций, соединенных с хост-процессором.

Согласно фиг.13 сети также могут содержать компьютерные мэйнфреймы или серверы, такие как шлюз (клиент-сервер 5046) или сервер приложений (удаленный сервер 5048, который может осуществлять доступ к хранилищу данных, а также может быть доступен непосредственно с рабочей станции 5045). Шлюз 5046 служит точкой входа в каждую отдельную сеть. Шлюз необходим при соединении одного сетевого протокола с другим. Шлюз 5046 предпочтительно может быть соединен линией связи

с другой сетью (например, сетью Интернет 5047). Шлюз 5046 также может быть прямо соединен с одной или несколькими рабочими станциями 5041, 5042, 5043, 5044 посредством линии связи. Шлюз может быть реализован с использованием сервера IBM eServer™ System z[®], поставляемым на рынок International Business Machines Corporation.

Одновременно на фиг.12 и фиг.13 представлен программный код программного обеспечения, который может реализовывать настоящее изобретение, может быть доступен процессору 5026 системы 5020 из долговременных запоминающих сред 5027, таких накопитель на CD-дисках или накопитель на жестких дисках. Программный код программного обеспечения может быть воплощен в любой из множества известных сред для использования с системой обработки данных, такой как дискета, жесткий диск или CD-диск. Код может распределяться в таких средах или может распределяться пользователям 5050, 5051 из памяти или запоминающего устройства одной компьютерной системы по сети другим компьютерным системам для использования пользователями таких других систем.

Альтернативно, программный код может быть осуществлен в памяти 5025, с доступом к нему процессором 5026 с применением процессорной шины. Такой программный код включает операционную систему, которая управляет функцией и взаимодействием различных компонентов компьютера и одной или нескольких прикладных программ 5032. Программный код обычно подкачивается по частям из запоминающих сред 5027 в быстродействующую память 5025, где он доступен для обработки процессором 5026. Методы и способы осуществления программного кода программного обеспечения в памяти, в физических средах и/или распределения кода программного обеспечения посредством сетей хорошо известны, и в данном документе подробнее описываться не будут. Программный код, созданный и хранящийся в материальной среде (включая без ограничения модули электронной памяти (ОЗУ), флэш-память, компакт-диски (CD), DVD-диски, магнитную ленту и т.п.) часто называют «компьютерным программным продуктом». Содержащая компьютерный программный продукт среда обычно может считываться устройством обработки данных предпочтительно в компьютерной системе для выполнения устройством обработки данных.

Кэш, который является наиболее легкодоступным для процессора (обычно более быстродействующий и менее объемный, чем другие кэши процессора), представляет собой низший (L1 или уровня 1) кэш, а основное запоминающее устройство (основная память) представляет собой кэш высшего уровня (L3, если имеется 3 уровня). Кэш низшего уровня часто делится на кэш команд (I-кэш), хранящий машинные команды, подлежащие выполнению, и кэш данных (D-кэш), содержащий операнды, хранимые в памяти.

Фиг.14 представляет собой изображение иллюстративного варианта осуществления процессора для процессора 5026. Обычно, чтобы буферизировать блоки памяти с целью повышения производительности процессора, используется один или несколько уровней кэша 5053. Кэш 5053 представляет собой высокоскоростной буфер, хранящий строки кэша данных памяти, которые предположительно будут использоваться. Типичные строки кэша содержат 64, 128 или 256 байтов данных памяти. Для кэширования команд и кэширования данных часто используются отдельные кэши. Согласованность кэшей (синхронизация копий строк в памяти и в кэшах) часто обеспечивается различными алгоритмами слежения, хорошо известными в данной области техники. Основное запоминающее устройство 5025 процессорной системы часто называют кэшем. В процессорной системе, имеющей 4 уровня кэша 5053, основное запоминающее

устройство 5025 иногда называют кэшем 5 уровня (L5), поскольку оно обычно является более быстродействующим и содержит лишь часть энергонезависимого запоминающего устройства (DASD, лента и т.д.), которое доступно для компьютерной системы. Основное запоминающее устройство 5025 «кэширует» страницы данных, которые постранично перемещаются в основное запоминающее устройство 5025 и из него операционной системой.

Программный счетчик (счетчик команд) 5061 отслеживает адрес текущей команды, подлежащей выполнению. Счетчик команд в процессоре z/Architecture[®] является 64-битным и может быть усечен до 31 или 24 битов с целью поддержки предыдущих ограничений адресации. Счетчик команд обычно воплощен в PSW (слове статуса программы) компьютера, так что он сохраняется при переключении контекста. Таким образом, работающая программа, имеющая значение счетчика команд, может прерываться, например, операционной системой (при переключении контекста из программной среды в среду операционной системы). PSW программы сохраняет значение счетчика команд, пока программа неактивна, и программный счетчик (в PSW) операционной системы используется, пока выполняется операционная система. Обычно счетчик команд увеличивается на величину, равную числу байтов текущей команды. RISC-команды (вычисления с сокращенным набором команд) обычно имеют фиксированную длину, тогда как CISC-команды (вычисления со сложным набором команд) обычно имеют переменную длину. Команды IBM z/Architecture[®] являются CISC-командами, имеющими длину 2, 4 или 6 байтов. Счетчик 5061 команд изменяется, например, или операцией переключения контекста, или операцией выбранного перехода команды перехода. В операции переключения контекста текущее значение счетчика команд сохраняется в слове статуса программы вместе с другой информацией о состоянии выполняемой программы (такой как коды состояний), и загружается новое значение счетчика команд, указывающее на команду нового программного модуля, который необходимо выполнить. Операция выбранного перехода выполняется, чтобы позволить программе принимать решения или выполнять цикл в программе путем загрузки результата команды перехода в счетчик 5061 команд.

Обычно для выборки команд от имени процессора 5026 используется блок 5055 выборки команд. Блок выборки или извлекает «следующие последовательные команды», целевые команды команд выбранного перехода или первые команды программы, следующей за переключением контекста. В современных блоках выборки команд часто применяют методы предварительной выборки, чтобы эмпирически осуществлять предварительную выборку команд на основании вероятности того, что предварительно выбранные команды могли бы быть использованы. Например, блок выборки может осуществлять выборку 16 байтов команды, которая содержит следующую последовательную команду, и дополнительных байтов других последовательных команд.

Затем извлеченные команды выполняются процессором 5026. В одном варианте осуществления извлеченная команда(-ы) передаются блоку 5056 диспетчеризации блока выборки. Блок диспетчеризации декодирует команду(-ы) и пересылает информацию о декодированной команде(-ах) соответствующим блокам 5057, 5058, 5060. Блок 5057 выполнения обычно принимает информацию о декодированных арифметических командах от блока 5055 выборки команд и выполняет арифметические операции с операндами в соответствии с кодом операции команды. Операнды подаются на блок 5057 выполнения предпочтительно или из памяти 5025, структурированных регистров 5059 или из ближайшего поля выполняемой команды. Сохраненные результаты

выполнения хранятся или в памяти 5025, регистрах 5059 или в другом машинном аппаратном обеспечении (таком как управляющие регистры, регистры PSW и т.п.).

Процессор 5026 обычно имеет один или несколько блоков 5057, 5058, 5060 для выполнения функции команды. Согласно фиг.15А блок 5057 выполнения может сообщаться со структурированными регистрами 5059 общего назначения, блоком 5056 декодирования/диспетчеризации, блоком 5060 загрузки/сохранения и другими процессорными блоками 5065 посредством интерфейсной логики 5071. Блок 5057 выполнения может применять несколько регистровых схем 5067, 5068, 5069, чтобы хранить информацию, с которой будет работать арифметическое логическое устройство (ALU) 5066. ALU выполняет арифметические операции, такие как сложение, вычитание, умножение и деление, а также логические функции, такие как И, ИЛИ и исключающее ИЛИ (XOR), циклический сдвиг и смещение. ALU предпочтительно поддерживает специализированные операции, зависящие от структуры. Другие схемы могут обеспечивать другие структурированные средства 5072, включающие, например, коды ситуаций и логику поддержки восстановления. Обычно результат операции ALU хранится в схеме 5070 выходного регистра, которая может направлять результат ряду других функций обработки. Хотя существует множество конструкций процессоров, настоящее описание предназначено лишь обеспечить характерное понимание одного варианта осуществления.

Например, команда ADD выполняется блоком 5057 выполнения, обладающим арифметическими и логическими функциональными возможностями, тогда как, например, команда с плавающей точкой выполняется блоком вычислений с плавающей точкой, обладающим специальными возможностями работы с плавающей точкой. Блок выполнения предпочтительно работает с указанными командой операндами путем выполнения функции, определенной кодом операции, на операндах. Например, команда ADD может выполняться блоком 5057 выполнения на операндах, находящихся в двух регистрах 5059, указанных в регистровых полях команды.

Блок 5057 выполнения выполняет арифметическое сложение двух операндов и сохраняет результат в третьем операнде, где третий операнд может быть третьим регистром или одним из двух исходных регистров. Блок выполнения предпочтительно использует арифметическое логическое устройство (ALU) 5066, способное выполнять ряд логических функций, таких как смещение, циклический сдвиг, И, ИЛИ и исключающее ИЛИ, а также ряд алгебраических функций, включая любую из сложения, вычитания, умножения, деления. Некоторые ALU 5066 разработаны для скалярных операций, а некоторые - для плавающей точки. В зависимости от архитектуры данные могут иметь обратный порядок следования байтов (когда наименьший значимый байт соответствует старшему байтовому адресу) или прямой порядок следования байтов (когда наименьший значимый байт соответствует младшему байтовому адресу). IBM z/Architecture[®] имеет обратный порядок следования байтов. В зависимости от архитектуры поля чисел со знаком могут быть представлены в виде прямого кода, дополнения до единицы или дополнения до двух. Число в форме дополнения до двух выгодно в том смысле, что ALU не нужно поддерживать процедуру вычитания, поскольку и отрицательное, и положительное значение в форме дополнения до двух в ALU требует только сложения. Числа обычно описаны в сокращенном виде, в котором 12-битное поле определяет адрес блока из 4096 байтов и обычно описывается, например, как 4-кбайтный (килобайтный) блок.

Согласно фиг.15Б информация команды перехода для выполнения команды перехода обычно передается блоку 5058 перехода, который часто применяет алгоритм

предсказания переходов, такой как таблица 5082 истории переходов, чтобы предсказывать исход перехода до завершения других условных операций. Цель текущей команды перехода извлекается и выполняется по предположению до завершения условных операций. Когда условные операции завершаются, выполненные по предположению команды перехода или завершаются, или отбрасываются на основании ситуаций условной операции и предположенного исхода. Типичная команда перехода может проверять коды ситуаций и переход к целевому адресу, если коды ситуаций отвечают требованию перехода команды перехода, причем целевой адрес может рассчитываться на основании нескольких чисел, включая, например, числа из полей регистра или непосредственного поля команды. Блок 5058 перехода может применять ALU 5074, имеющее множество схем 5075, 5076, 5077 входных регистров и схему 5080 выходного регистра. Блок 5058 перехода, может поддерживать связь, например, с регистрами 5059 общего назначения, блоком 5056 декодирования/диспетчеризации или другими схемами 5073.

Выполнение группы команд может прерываться по ряду причин, включая, например, переключение контекста, инициированное операционной системой, исключительную ситуацию или ошибку программы, приводящую к переключению контекста, сигнал прерывания ввода-вывода, приводящий к переключению контекста, или многопоточную деятельность множества программ (в многопоточной среде). Предпочтительно переключение контекста сохраняет информацию о состоянии выполняемой в данный момент программы, а затем загружает информацию о состоянии другой запускаемой программы. Информация о состоянии может сохраняться, например, в аппаратных регистрах или в памяти. Информация о состоянии предпочтительно содержит значение счетчика команд, указывающее на очередную команду, которую необходимо выполнить, коды ситуаций, информацию о трансляции памяти и содержимое структурированного регистра. Деятельность по переключению контекста может осуществляться аппаратными схемами, прикладными программами, программами операционной системы или аппаратно-программным кодом (микрокодом, пикокодом или лицензионным внутренним кодом (LIC) по отдельности или в сочетании).

Процессор осуществляет доступ к операндам в соответствии с определенными командами способами. Команда может предоставлять непосредственный операнд, использующий значение части команды, может предоставлять одно или несколько регистровых полей, прямо указывающих или на регистры общего назначения, или на регистры особого назначения (например, регистры с плавающей точкой). Команда может использовать подразумеваемые регистры, идентифицируемые полем кода операции как операнды. Команда может использовать для операндов местоположения в памяти. Местоположение в памяти операнда может предоставляться регистром, непосредственным полем или сочетанием регистров и непосредственного поля, примером чего является средство дальнего смещения z/Architecture[®], в котором команда определяет базовый регистр, индексный регистр и непосредственное поле (поле смещения), которые суммируются, чтобы предоставлять, например, адрес операнда в памяти. Под местоположением в данном документе понимается местоположение в основной памяти (основном запоминающем устройстве), если не указано иное.

Согласно фиг.16В процессор осуществляет доступ к запоминающему устройству, используя блок 5060 загрузки/сохранения. Блок 5060 загрузки/сохранения может выполнять операцию загрузки путем получения адреса целевого операнда в памяти 5053 и загрузки операнда в регистр 5059 или другое местоположение в памяти 5053, или может выполнять операцию сохранения путем получения адреса целевого операнда

в памяти 5053 и сохранения данных, полученных из регистра 5059 или другого местоположения памяти 5053, в местоположении целевого операнда в памяти 5053. Блок 5060 загрузки/сохранения может действовать по предположению и осуществлять доступ к памяти в последовательности, которая по порядку не соответствует последовательности команд, однако блок 5060 загрузки/сохранения должен поддерживать для программ видимость того, что команды выполнялись по порядку. Блок 5060 загрузки/сохранения может поддерживать связь с регистрами 5059 общего назначения, блоком 5056 декодирования/диспетчеризации, интерфейсом 5053 кэша/памяти или другими элементами 5083 и содержит различные регистровые схемы, устройства ALU 5085 и управляющую логику 5090, чтобы вычислять адреса запоминающего устройства и обеспечивать формирование последовательности конвейера для сохранения порядка операций. Некоторые операции могут не сохранять порядок, но блок загрузки/сохранения обеспечивает функциональные возможности для того, чтобы операции с нарушенным порядком казались программе выполненными по порядку, как хорошо известно в данной области техники.

Предпочтительно адреса, которые «видит» прикладная программа, часто называют виртуальными адресами. Виртуальные адреса иногда называют «логическими адресами» и «исполнительными адресами». Эти виртуальные адреса являются виртуальными в том смысле, что они перенаправляются в местоположение в физической памяти посредством одной из ряда технологий динамической трансляции адреса (DAT), включая без ограничения простую префиксацию виртуального адреса величиной смещения, трансляцию виртуального адреса посредством одной или нескольких таблиц трансляции, причем таблицы трансляции предпочтительно содержат по меньшей мере таблицу сегментов и таблицу страниц по отдельности или в сочетании, предпочтительно таблицу сегментов, содержащую запись, указывающую на таблицу страниц. В z/Architecture® предоставляется иерархия трансляции, включающая первую таблицу области, вторую таблицу области, третью таблицу области, таблицу сегментов и необязательную таблицу страниц. Эффективность трансляции адресов часто повышается за счет использования буфера (TLB) ассоциативной трансляции, который содержит записи, отображающие виртуальный адрес для соответствующего местоположения в физической памяти. Записи создаются, когда DAT транслирует виртуальный адрес с помощью таблиц трансляции. Последующее использование виртуального адреса может затем использовать запись быстросействующего TLB, а не медленный последовательный доступ к таблице трансляции. Содержимым TLB может управлять ряд алгоритмов замещения, включая алгоритм замещения наиболее давней по использованию страницы (LRU).

В том случае, когда процессор является процессором многопроцессорной системы, каждый процессор отвечает за сохранение совместно используемых ресурсов, таких как средства ввода-вывода, кэши, TLB и память, взаимно заблокированных для обеспечения непротиворечивости. Обычно для поддержания непротиворечивости кэшей используются технологии «слежения». Чтобы облегчать совместное использование, в среде слежения каждая строка кэша может отмечаться как находящаяся в любом состоянии из состояния совместного использования, состояния монопольного использования, измененного состояния, недостоверного состояния и т.п.

Блоки 5054 ввода-вывода (фиг. 14) обеспечивают процессор средствами подключения к периферийным устройствам, включая, например, накопители на магнитной ленте, диски, принтеры, устройства отображения и сети. Блоки ввода-вывода часто представляются компьютерной программе драйверами программного обеспечения. В мэйнфреймах, таких как System z® производства IBM®, каналные адаптеры и адаптеры

открытых систем являются блоками ввода-вывода мэйнфрейма, которые обеспечивают связь между операционной системой и периферийными устройствами.

Выгоду от одной или нескольких особенностей настоящего изобретения могут получать и другие типы вычислительных сред. Например, среда может содержать эмулятор (например, программные или другие механизмы эмуляции), в котором эмулируется конкретная архитектура (включая, например, выполнение команд, структурированные функции, такие как трансляция адреса, и структурированные регистры) или ее подмножество (например, в частной компьютерной системе, имеющей процессор и память). В такой среде одна или несколько функций эмуляции эмулятора могут осуществлять одну или несколько особенностей настоящего изобретения, даже если компьютер, на котором работает эмулятор, может иметь архитектуру, отличающуюся от эмулируемых мощностей. Например, в режиме эмуляции декодируется конкретная эмулируемая команда или операция, и для реализации отдельной команды или операции создается соответствующая функция эмуляции.

В эмулирующей среде хост-компьютер содержит, например, память для хранения команд и данных; блок выборки команд для извлечения команд из памяти и необязательно обеспечения локальной буферизации извлеченной команды; блок декодирования команд для приема извлеченных команд и определения типа команд, которые были извлечены; и блок выполнения команд для выполнения команд.

Выполнение может включать загрузку данных из памяти в регистр; сохранение данных обратно память из регистра; или выполнение арифметической или логической операции некоторого типа, как определяется блоком декодирования. В одном примере каждый блок реализован в программном обеспечении. Например, выполняемые блоками операции реализуются в виде одной или нескольких подпрограмм в программном обеспечении эмулятора.

Более конкретно, в мэйнфрейме структурированные машинные команды используются программистами, сегодня, как правило, программистами, работающими на языке C, часто посредством приложения-компилятора. Эти команды, хранящиеся в запоминающей среде, могут выполняться в исходном формате на сервере z/Architecture IBM или в другом случае в машинах, реализующих другие архитектуры. Они могут эмулироваться в существующих и будущих мэйнфрейм-серверах IBM® и на других машинах IBM® (например, серверах Power Systems и серверах System x®). Они могут выполняться в машинах, работающих на Linux, на широком ряде машин, использующих

аппаратное обеспечение производства IBM®, Intel®, AMD™ и других. Помимо выполнения на этом аппаратном обеспечении под z/Architecture®, может использоваться Linux, равно как и машины, которые используют эмуляцию посредством Hercules, UMX или FSI (Fundamental Software, Inc), когда выполнение обычно происходит в режиме эмуляции. В режиме эмуляции программное обеспечение эмуляции выполняется собственным процессором, чтобы эмулировать архитектуру эмулируемого процессора.

Собственный процессор, чтобы выполнять эмуляцию эмулируемого процессора, как правило, выполняет программное обеспечение эмуляции, содержащее или аппаратно-программное обеспечение, или собственную операционную систему. Программное обеспечение эмуляции отвечает за извлечение и выполнение команд архитектуры эмулируемого процессора. Программное обеспечение эмуляции поддерживает счетчик эмулируемых программ, чтобы отслеживать границы команд. Программное обеспечение эмуляции может извлекать одну или несколько эмулируемых машинных команд за раз и преобразовывать одну или несколько эмулируемых машинных команд в

соответствующую группу собственных машинных команд для выполнения собственным процессором. Эти преобразованные команды могут кэшироваться, так что можно добиться более быстрого преобразования. Тем не менее, программное обеспечение эмуляции должно поддерживать правила архитектуры эмулируемого процессора, чтобы

5 обеспечивать корректную работу операционных систем и приложений, написанных для эмулируемого процессора. Кроме того, программное обеспечение эмуляции должно обеспечивать ресурсы, определенные архитектурой эмулируемого процессора, включая без ограничения управляющие регистры, регистры общего назначения, регистры с плавающей точкой, функцию динамической трансляции адреса, включая, например,

10 таблицы сегментов и таблицы страниц, механизмы прерывания, механизмы переключения контекста, часы истинного времени (TOD) и структурированные интерфейсы к подсистемам ввода-вывода, чтобы операционная система или прикладная программа, созданная для работы на эмулируемом процессоре, могла запускаться на собственном процессоре, имеющем программное обеспечение эмуляции.

15 Конкретная эмулируемая команда декодируется, и для выполнения функции отдельной команды вызывается подпрограмма. Функция программного обеспечения эмуляции, эмулирующая функцию эмулируемого процессора, реализуется, например, в подпрограмме или драйвере на языке С или каким-либо другим способом предоставления драйвера для конкретного аппаратного обеспечения, который будет

20 доступен специалистам в данной области техники после понимания описания предпочтительного варианта осуществления. Множество патентов, относящихся к эмуляции программного и аппаратного обеспечения, включая без ограничения патент на изобретение США №5551013 под названием "Multiprocessor for Hardware Emulation", by Beausoleil et al.; и патент на изобретение США №6009261 под названием "Preprocessing

25 of Stored Target Routines for Emulating Incompatible Instructions on a Target Processor", by Scaizi et al.; и патент на изобретение США №5574873 под названием "Decoding Guest Instruction to Directly Access Emulation Routines that Emulate the Guest Instructions", by Davidian et al.; и патент на изобретение США №6308255 под названием "Symmetrical Multiprocessing Bus and Chipset Used for Coprocessor Support Allowing Non-Native Code to

30 Run in a System", by Gorishek et al.; и патент на изобретение США №6463582 под названием "Dynamic Optimizing Object Code Translator for Architecture Emulation and Dynamic Optimizing Object Code Translation Method", by Lethin et al.; и патент на изобретение США №5790825 под названием "Method for Emulating Guest Instructions on a Host Computer Through Dynamic Recompilation of Host Instructions", by Eric Traut, каждый из которых включается в данный

35 документ посредством ссылки во всей своей полноте; и многие другие, демонстрируют множество известных способов достижения эмуляции формата команд, структурированного для другой машины, на целевой машине, доступных специалистам в данной области техники.

Фиг.16 представляет собой изображение примера эмулируемой компьютерной хост-системы 5092, которая эмулирует компьютерную хост-систему 5000' хост-архитектуры.

40 В эмулируемой компьютерной хост-системе 5092 хост-процессор (ЦП) 5091 является эмулируемым хост-процессором (или виртуальным хост-процессором) и содержит процессор 5093 эмуляции, имеющий другую собственную архитектуру набора команд, чем у процессора 5091 хост-компьютера 5000'. Эмулируемая компьютерная хост-система

45 5092 содержит память 5094, доступную для процессора 5093 эмуляции. В иллюстративном варианте осуществления память 5094 разделена на часть памяти 5096 хост-компьютера и часть 5097 программ эмуляции. Память 5096 хост-компьютера доступна для программ эмулируемого хост-компьютера 5092 в соответствии с архитектурой хост-компьютера.

Процессор 5093 эмуляции выполняет собственные команды структурированного набора команд, имеющих структуру, отличную от структуры команд эмулируемого процессора 5091, собственные команды извлекаются из памяти 5097 подпрограмм эмуляции, и может осуществлять доступ к хост-команде для выполнения из программы в памяти 5096 хост-компьютера путем применения одной или нескольких команд, полученных в подпрограмме последовательности и доступа/декодирования, которая может декодировать хост-команду(-ы), к которой осуществляется доступ, чтобы определять подпрограмму выполнения собственных команд для эмуляции функции хост-команды, к которой осуществляется доступ. Другие средства, которые определены для архитектуры компьютерной хост-системы 5000', могут эмулироваться подпрограммами структурированных средств, включая такие средства, как, например, регистры общего назначения, управляющие регистры, поддержка подсистемы динамического преобразования адреса и ввода-вывода и кэш процессора. Подпрограммы эмуляции также могут получать преимущество от функций, доступных в процессоре 5093 эмуляции (таких как регистры общего назначения и динамическая трансляция виртуальных адресов), чтобы улучшать производительность подпрограмм эмуляции. Также может предоставляться специальное аппаратное обеспечение и механизмы разгрузки, чтобы помогать процессору 5093 эмулировать функции хост-компьютера 5000'.

Используемая в данном документе терминология имеет целью описание лишь конкретных вариантов осуществления и не предназначена ограничивать изобретение. Как используются в данном документе, формы единственного числа также включают формы множественного числа, если иное прямо не следует из контекста. Также будет понятно, что термины "содержит" и/или "содержащий", когда используются в этом описании, определяют наличие указанных признаков, чисел, этапов, операций, элементов и/или компонентов, но не исключают наличия или добавления одного или нескольких других признаков, чисел, шагов, операций, элементов, компонентов и/или их групп.

Подразумевается, что соответствующие структуры, материалы, действия и эквиваленты всех элементов "средство или этап плюс функция" в предоставленной ниже формуле изобретения, если таковые имеются, включают любую структуру, материал или действие для выполнения функции в сочетании с другими заявленными элементами, как заявлено конкретным образом. Описание настоящего изобретения было представлено с целью демонстрации и описания, и не подразумевается полным или ограниченным изобретением в раскрытой форме. Специалистам обычной квалификации в данной области техники будут очевидны многочисленные модификации и изменения, не выходящие за пределы объема изобретения. Вариант осуществления был выбран и описан, чтобы наилучшим образом пояснить принципы изобретения и его практическое применение и позволить специалистам в данной области техники понять изобретение для различных вариантов осуществления с различными изменениями, которые подходят для конкретного целевого применения.

Управление программно-аппаратным обеспечением твердотельного запоминающего устройства

Технологии предшествующие твердотельным энергонезависимым оперативным запоминающим устройствам (RAM), совместно именуемым твердотельными запоминающими устройствами (SCM), быстро устаревают. В результате SCM начинают появляться в мэйнфрейм-серверах корпоративного класса. Например, SCM может использоваться в качестве физической замены для вращающихся дисков с механической рукой, при этом интерфейс SCM подобен интерфейсу более традиционных запоминающих устройств прямого доступа (DASD) Однако, вследствие кардинально

различных рабочих характеристик SCM и DASD, новые применения могут использовать SCM скорее в качестве дополнительной памяти и при этом извлекать пользу из механизма, который скрывает DASD реализацию и подробности реализации. Также характеристики надежности, доступности и обслуживаемости (RAS) готовых к использованию SCM предложений могут требовать улучшений, с тем чтобы сделать их более подходящими для вычислительной среды корпоративного класса.

Может использоваться уровень абстракции программно-аппаратного обеспечения, который управляет твердотельным запоминающим устройством (SCM), например, в корпоративном сервере. Стандартные адаптеры ввода-вывода (I/O) SCM могут автоматически конфигурироваться и управляться таким образом, который исключает критическую точку поломки, обеспечивает выделение ресурсов множеству логических разделов и прозрачным образом балансирует рабочую нагрузку на SCM устройства.

Предоставлено вычислительное устройство, такое как корпоративная вычислительная система, при этом следует понимать, что SCM представляет собой главным образом флэш-память и обеспечивает характеристики сохранения данных при отсутствии энергоснабжения. В общем способе реализации SCM выполнено в виде твердотельного устройства - (SSD) диска, который может представлять собой один элемент, или может быть выполнен в виде нескольких SSD дисков соединенных последовательно с адаптерами ввода-вывода. Адаптеры ввода-вывода физически соединены с концентраторами ввода-вывода, которые в рабочем состоянии расположены между адаптерами ввода-вывода и блоком основной памяти вычислительного устройства, хотя следует понимать, что наличие концентраторов ввода-вывода не обязательно, и что адаптеры ввода-вывода могут быть напрямую соединены с блоком основной памяти. Адаптеры ввода-вывода могут быть взаимно соединены друг с другом, или могут быть независимыми друг от друга. Основная память соединена с одним или более центральным процессором (ЦП) и/или одним или более устройствами обработки данных с тем, чтобы определить, в каком процессоре постоянно находится программно-аппаратное обеспечение.

При такой конфигурации следует ожидать, что ресурсы SSD будут использоваться в высокой степени различными программами, выполняемыми в процессоре в данный момент. Как правило, управление таким использованием достигается посредством использования логических разделов, причем каждые выделенные им ресурсы (например, память, ЦП) оказываются независимыми и полностью присваиваются, когда они могут в действительности использоваться в качестве части физически совместного основного ресурса. Таким образом, SSD конфигурируются как физически совместные ресурсы с емкостью запоминающего устройства, распределенной среди одного или более логических разделов.

Для этого микрокод сохраняется в аппаратном обеспечении адаптеров ввода-вывода и определяет выполняемые команды для обеспечения определенной функциональности ввода-вывода, такой как подготовка и форматирование SSD для с целью выделения различным логическим разделам. Например, микрокод адаптера ввода-вывода определяет выполняемые команды для формирования первого набора томов объема памяти с защитой RAID 10 (чередование и зеркалирование) на пространстве, доступном на SSD, соединенных с адаптером ввода-вывода и на SSD, соединенных с адаптером ввода-вывода. Подобным образом микрокод адаптера ввода-вывода определяет выполняемые команды для формирования второго набора томов объема памяти с защитой RAID 10 (чередование и зеркалирование) на пространстве, доступном на SSD, соединенных с адаптером ввода-вывода и на SSD, соединенных с адаптером ввода-

вывода. Таким образом, следует отметить, что когда адаптеры ввода-вывода физически соединены посредством провода, физическое соединение необязательно и дополнительно следует отметить, что программно-аппаратное обеспечение может быть реализовано таким образом, чтобы выполнять подобные функции, хотя они выполняются на

5 устройствах обработки данных.

С описанным выше микрокодом, хранимом в адаптерах ввода-вывода, программно-аппаратное обеспечение, находящееся в процессоре, служит для определения уровня абстракции для управления SCM в добавок к уровню абстракции, предоставляемому микрокодом. Например, когда микрокод адаптера ввода-вывода определяет

10 выполняемые команды для формирования набора томов объема памяти с защитой RAID 10, программно-аппаратное обеспечение определяет условия для реализации данных функций и инструктирует адаптер ввода-вывода для формирования набора томов на SSD, соединенных с адаптером ввода-вывода и SSD, соединенных с другим адаптером ввода-вывода. Это будет описано ниже более подробно.

15 Способ управления и получения доступа к SCM в корпоративной вычислительной среде, предусматривающий внедрение уровня абстракции посредством программно-аппаратного обеспечения находящегося в процессоре, который автоматически выполняет функции, которые обычно требуют участия администратора или программ, написанных особым образом с учетом эксплуатационных характеристик SCM

20 технологии. Инициированное посредством функции специальной конфигурации, выданной системной консолью во время установки, изобретение относится к автоматическому конфигурированию двух или более адаптеров ввода-вывода, к каждому из которых подсоединено одно или более SSD, включая SCM, а также к распределению SSD для образования защищенных RAID массивов независимых от физических

25 соединений или отсоединений между адаптерами ввода-вывода. Таким образом, физическая реализация способа может быть скрыта от пользователя.

В соответствии с другими аспектами изобретения, распределение SCM достигается посредством разбивания на приращения фиксированного размера с авторизацией для доступа к данным приращениям, запрашиваемым пользовательскими разделами. Также

30 программно-аппаратное обеспечение, находящееся в процессоре, управляет распределением и доступом к приращениям, таким образом, обеспечивая доступ только авторизованным разделам к определенному приращению, и когда пользовательский раздел утрачивает авторизацию для доступа к определенному разделу, программно-аппаратное обеспечения сбрасывает приращение твердотельного запоминающего

35 устройства в ноль и помещает приращение обратно в пул для выделения другому разделу. При доступе к SCM для программного логического адреса может внедряться уровень абстракции с тем, чтобы таким образом скрыть физический адрес соответствующего адаптера ввода-вывода. Данный уровень абстракции, а также прозрачное распределение рабочей нагрузки обеспечиваются посредством чередования

40 адресного пространства линейного логического SCM между двумя или более адаптерами ввода-вывода, используемыми для доступа к SSD.

В ходе процедур установки SSD соединяются с адаптерами ввода-вывода, и адаптеры соединяются с основной памятью. Как отмечено выше, адаптеры ввода-вывода могут быть соединены друг с другом, или могут быть независимыми, или физически

45 отсоединенными друг от друга. При подаче энергии на вычислительную систему и последующем выполнении вычислительной системой последовательности инициализации, до этого момента неконфигурированные адаптеры ввода-вывода опознаются программно-аппаратным обеспечением во время прохождения по шине ввода-вывода.

Как только адаптеры ввода-вывода опознаны, программно-аппаратное обеспечение инициирует, или оно может быть направлено на инициирование операции "создание пары" для инициализирования адаптеров ввода-вывода. Операция инициализации контролируется программно-аппаратным обеспечением и управляет адаптерами ввода-вывода с тем, чтобы форматировать каждый из SSD, соединенных с адаптерами ввода-вывода, а также группировать SSD в два или более массива соединенные адаптерами ввода-вывода, и защищенные, например, защитой RAID 10 (чередование и зеркалирование) таким образом, что каждый массив может включать один набор томов (Vset). В качестве части операции инициализации, программно-аппаратное обеспечение дает команду адаптерам ввода-вывода на сброс всего адресного пространства набора томов Vsets.

При завершении команды создания пары, обеспечивается логическое распределение наборов томов. Когда адаптеры ввода-вывода соединяются проводом, адаптер ввода-вывода генерирует набор томов Vset A, занимающий адресные пространства A1-An на SSD, соединенных с ним, а также зеркалированные адресные пространства A1-An на SSD, соединенных с адаптером ввода-вывода, тогда как адаптер ввода-вывода генерирует наборы томов Vset B, занимающие адресные пространства A1-An на SSD соединенных с ним и зеркалированные адресные пространства A1-An на SSD 10, соединенных с адаптером ввода-вывода. И наоборот, программно-аппаратное обеспечение, выполняемое на процессоре, генерирует Vset C, занимающий адресные пространства C1-Cn на SSD, соединенных с ним, а также зеркалированные адресные пространства C1-Cn на SSD, соединенных с адаптером ввода-вывода, при этом программно-аппаратное обеспечение генерирует Vset D, занимающий адресные пространства D1-Dn на SSD, соединенных с ним, а также зеркалированные адресные пространства D1-Dn на SSD, соединенных с адаптером ввода-вывода. Таким образом, адаптеры ввода-вывода могут использоваться при виртуальном соединении, однако в ином случае они останутся физически независимыми и отсоединенными друг от друга.

Таким образом, плоские, видимые для пользователя адресные пространства SCM могут быть нанесены при физической реализации, которая является независимой от физической реализации (или ее отсутствия). Доступ к наборам томов Vset A и Vset B осуществляется посредством адаптеров ввода-вывода, соответственно, посредством которых выполняется микрокод, который обеспечивает чередование и зеркалирование данных. Доступ к наборам томов Vset C и Vset D осуществляется посредством уровня представления данных программно-аппаратного обеспечения, который обеспечивает чередование и зеркалирование данных.

Адресное пространство SCM чередуется с адресными пространствами наборов томов Vset на, например, мегабайтной границе, где доступ к четным мегабайтным адресам SCM обеспечивается посредством Vset A (адаптер ввода-вывода), при этом доступ к нечетным мегабайтным адресам SCM обеспечивается посредством Vset B (адаптер ввода-вывода). Когда установлено несколько пар адаптеров, соответствующие адресные пространства SCM могут чередоваться или не чередоваться с наборами пар адаптеров. Зеркалирование массивов между SSD посредством отдельных адаптеров ввода-вывода обеспечивает допуск на случай поломки адаптера и обеспечивает возможность неразрушающего восстановления адаптера. То есть, в то время когда адаптер недоступен вследствие поломки, адресное пространство SCM обслуживается другим адаптером (адаптерами) благодаря зеркалированию. Также деление SSD на два или более массива обеспечивает деление работы между двумя адаптерами ввода-вывода.

Нанесение адресного пространства SCM при физической реализации (или ее

отсутствии) очевидное для пользователя может упростить проектирование программно-аппаратного обеспечения посредством устранения выбора и может обеспечить решение, которое одинаково совместимо с определенными доступными на данный момент серверными конфигурациями.

- 5 Адресное пространство SCM, обеспеченное всеми установленными адаптерами ввода-вывода, может быть логически разделено, например, на 16 гигабайтные приращения или, в частности, на приращения заданного размера или размеров. Приращение представляет собой блок, на который между логическими разделами делится адресное пространство SCM. Когда приращение выделяется разделу,
- 10 программно-аппаратное обеспечение обеспечивает то, чтобы раздел обладал эксклюзивным использованием пространства, и чтобы приращение было недоступным для других разделов. Если приращение не выделяется определенному разделу, но доступно для выделения, приращение помещается в свободно-доступный пул посредством программно-аппаратного обеспечения. Перевыделение приращения из
- 15 раздела может быть выполнено пользователем, и перед помещением приращения в свободно-доступный пул программно-аппаратное обеспечение сбрасывает содержимое приращения в ноль. Когда приращение сброшено (длительная операция), приращение может учитываться в свободно-недоступном пуле.

20 Формула изобретения

1. Способ выполнения команды для выполнения команды конфигурирования твердотельного запоминающего устройства (Configure Storage Class Memory) в вычислительной системе, имеющей доступ к основному запоминающему устройству и твердотельному запоминающему устройству, причем твердотельное запоминающее
- 25 устройство (180) включает множество твердотельных устройств, включающий:
- получение подсистемой (160) ввода-вывода, осуществляемое на основании запроса управляющей программы, выполняющейся внутри раздела системы или виртуализированного гостя системы и не имеющей непосредственного доступа к твердотельному запоминающему устройству, информации о конфигурации
- 30 твердотельного запоминающего устройства;
- сообщение подсистемой ввода-вывода управляющей команде по меньшей мере части полученной информации о конфигурации твердотельного запоминающего устройства, причем по меньшей мере часть полученной информации содержит максимальное количество приращений твердотельного запоминающего устройства,
- 35 которое может быть сконфигурировано в запрашивающую конфигурацию, включающую раздел системы или виртуализированного гостя системы;
- получение подсистемой ввода-вывода, на основании вышеупомянутого сообщения управляющей программе, блока (700) запроса, содержащего командный код (704), указывающий команду конфигурирования твердотельного запоминающего устройства
- 40 и значение (708) общего размера для определения запрашиваемого количества приращений твердотельного запоминающего устройства, подлежащего конфигурированию в адресное пространство твердотельного запоминающего устройства из доступного для системы пула твердотельного запоминающего устройства, причем запрошенное количество приращений конфигурируют для запрашивающей
- 45 конфигурации, а адресное пространство твердотельного запоминающего устройства представляет собой одно адресное пространство, перекрывающее множество твердотельных устройств и заполняемое приращениями твердотельного запоминающего устройства для запрашивающей конфигурации, и скрывает подробности физической

реализации твердотельного запоминающего устройства;

на основании командного кода - инициирование процесса конфигурирования для конфигурирования твердотельного запоминающего устройства, при этом процесс сконфигурирован для выделения адресному пространству твердотельного
 5 запоминающего устройства запрашиваемого количества приращений твердотельного запоминающего устройства, определенного в общем значении размера, причем инициирование процесса конфигурирования включает выполнение одной или нескольких проверок достоверности;

продолжение осуществления процесса конфигурирования твердотельного
 10 запоминающего устройства в ответ на положительный результат одной или нескольких проверок достоверности; причем одна или несколько проверок достоверности включают определение того, что запрашиваемое общее значение размера не превышает количество приращений твердотельного запоминающего устройства в инициализированном состоянии, а продолжение осуществления процесса конфигурирования включает
 15 изменение внутреннего элемента управления для каждого конфигурируемого приращения, обеспечивающее возможность конфигурирования приращений таким образом, чтобы быть доступным для запросов перемещения ввода-вывода запрашивающей конфигурации; и

сохранение кода ответа в поле кода ответа блока ответа, указывающего на то, был
 20 ли инициирован процесс конфигурирования, при этом блок ответа содержит код длины, указывающий длину блока ответа и поле кода ответа.

2. Способ по п. 1, в котором одна или несколько проверок достоверности дополнительно включают определение того, превышает ли число уже
 сконфигурированных приращений твердотельного запоминающего устройства плюс
 25 запрошенный общий размер максимальный предел конфигурируемых приращений твердотельного запоминающего устройства, причем процесс конфигурирования продолжается в ответ на непревышение максимального количества приращений твердотельного запоминающего устройства.

3. Способ по п. 2, в котором одна или несколько проверок достоверности
 30 дополнительно включают проверку поля длины блока запроса; проверку того, является ли доступной для выполнения команда конфигурирования твердотельного запоминающего устройства; проверку того, что блок запроса имеет достоверный формат, и проверку того, что подсистема ввода-вывода способна осуществить команду конфигурирования твердотельного запоминающего устройства, причем процесс
 35 конфигурирования продолжается в ответ на положительный результат одной или нескольких проверок достоверности.

4. Способ по п. 1, в котором инициирование процесса конфигурирования дополнительно включает запуск асинхронного процесса для осуществления
 конфигурирования твердотельного запоминающего устройства в ответ на
 40 положительный результат одной или нескольких проверок достоверности, причем содержимое сконфигурированного твердотельного запоминающего устройства устанавливают в ноль.

5. Способ по п. 4, дополнительно включающий индикацию завершения асинхронного процесса в ответе на уведомление.

6. Способ по п. 5, в котором блок ответа на уведомление содержит код ответа, который показывает результаты попытки выполнения команды конфигурирования
 твердотельного запоминающего устройства.

7. Способ по п. 6, в котором блок ответа на уведомление получают посредством

выполнения команды сохранения информации о событиях (Store Event Information) для получения информации о событии.

8. Способ по п. 5, в котором блок запроса дополнительно сконфигурирован для включения в него коррелятора (710) асинхронного завершения, возвращаемого в блоке ответа на уведомление и используемого для возобновления обработки команды конфигурирования твердотельного запоминающего устройства.

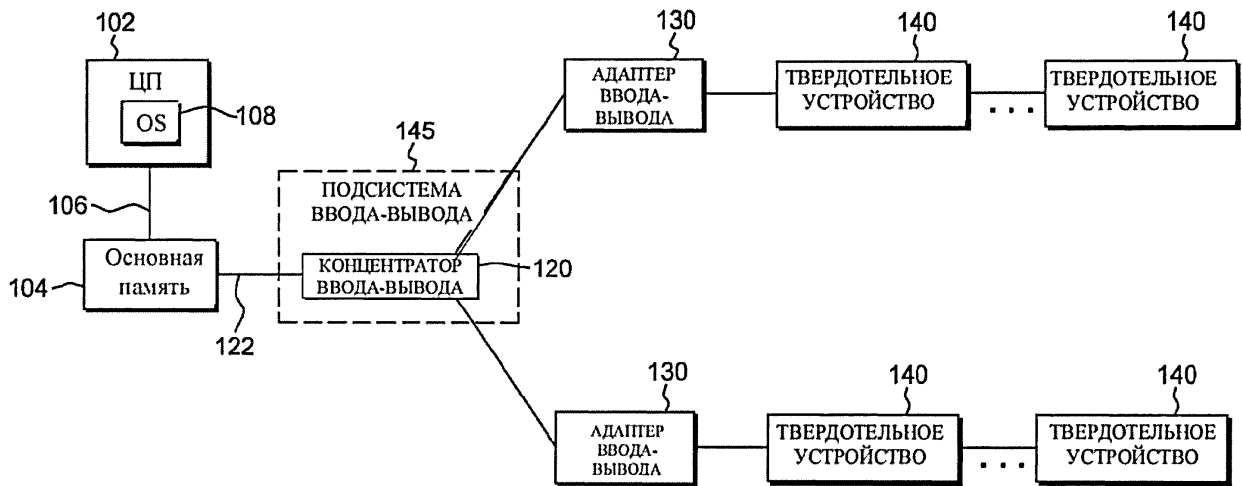
9. Способ по п. 1, в котором процесс конфигурирования включает: настройку одного или нескольких внутренних элементов управления для обеспечения доступности приращений твердотельного запоминающего устройства; и установку рабочего состояния приращения твердотельного запоминающего устройства.

10. Способ по п. 1, в котором блок запроса дополнительно сконфигурирован для включения в него коррелятора (710) асинхронного завершения, используемого для возобновления оригинального потока, инициировавшего команду конфигурирования твердотельного запоминающего устройства.

11. Компьютерная система для выполнения команды конфигурирования твердотельного запоминающего устройства (Configure Storage Class Memory) в вычислительной среде, содержащей основное запоминающее устройство и твердотельное запоминающее устройство (180), включающее множество твердотельных устройств, содержащая память и процессор, связанный с памятью, причем компьютерная система сконфигурирована для осуществления способа по одному из предыдущих пунктов.

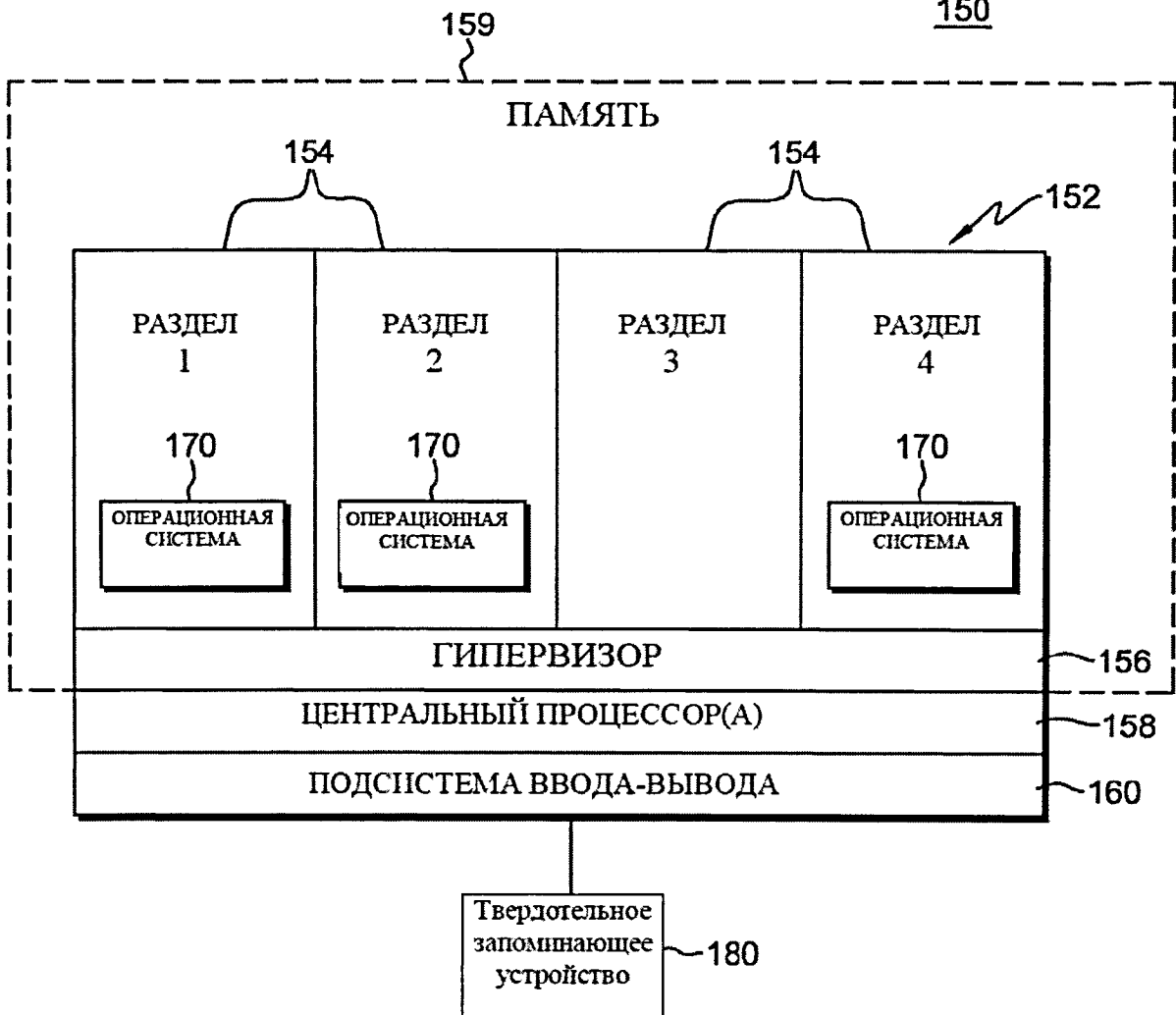
12. Машиночитаемый носитель (1002) данных, содержащий компьютерный программный продукт (1000) для выполнения команды конфигурирования твердотельного запоминающего устройства (Configure Storage Class Memory) в вычислительной среде, содержащей основное запоминающее устройство и твердотельное запоминающее устройство (180), включающее множество твердотельных устройств, при этом указанный компьютерный программный продукт содержит команды (1004) для выполнения устройством обработки данных с целью осуществления способа по одному из пп. 1-10.

100



ФИГ. 1А

150



ФИГ. 1Б

EADM ORB**200**

- 202 — ПАРАМЕТР ПРЕРЫВАНИЯ
- 204 — КЛЮЧ СУБКАНАЛА
- 205 — УПРАВЛЕНИЕ РАСШИРЕНИЕМ ORB
- 206 — АДРЕС (AOB) БЛОКА ОПЕРАЦИЙ EADM
- 208 — ПРИОРИТЕТ CSS
- 210 — ПРИОРИТЕТ SCM
- 212 — FMT

ФИГ. 2А**БЛОК ОПЕРАЦИЙ EADM (AOB)****220**

- 222 — БЛОК ЗАПРОСА EADM
- 224 — БЛОК ОТВЕТА EADM
- 226 — MSB0 – MSBx

ФИГ. 2Б**БЛОК ЗАПРОСА EADM****222**

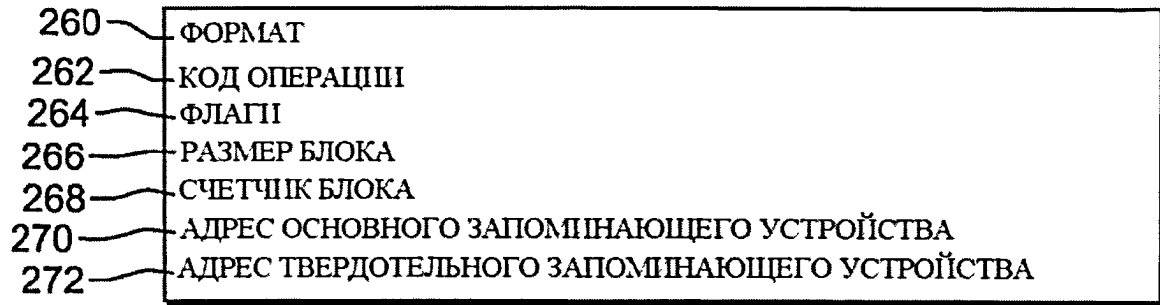
- 230 — ФОРМАТ
- 232 — КОД КОМАНДЫ
- 234 — СЧЕТЧИК MSB

ФИГ. 2В**БЛОК ОТВЕТА EADM****224**

- 240 — ФОРМАТ
- 242 — ФЛАГИ ПРЕРЫВАНИЙ
- 244 — ИДЕНТИФИКАТОРЫ БЛОКА УПРАВЛЕНИЯ ПРЕРЫВАНИЯМИ
- 246 — ФЛАГИ ДЕЙСТВИТЕЛЬНОСТИ ПОЛЕЙ
- 248 — КОД КВАЛИФИКАТОРА ИСКЛЮЧЕНИЙ
- 250 — АВАРИЙНЫЙ АДРЕС MSB
- 252 — АВАРИЙНЫЙ АДРЕС AIDAW
- 254 — АВАРИЙНЫЙ АДРЕС ОСНОВНОГО ЗАПОМИНАЮЩЕГО УСТРОЙСТВА
- 256 — АВАРИЙНЫЙ АДРЕС ТВЕРДОТЕЛЬНОГО ЗАПОМИНАЮЩЕГО УСТРОЙСТВА

ФИГ. 2Г

БЛОК СПЕЦИФИКАЦИИ ПЕРЕМЕЩЕНИЯ ЕАДМ 226



ФИГ. 2Д

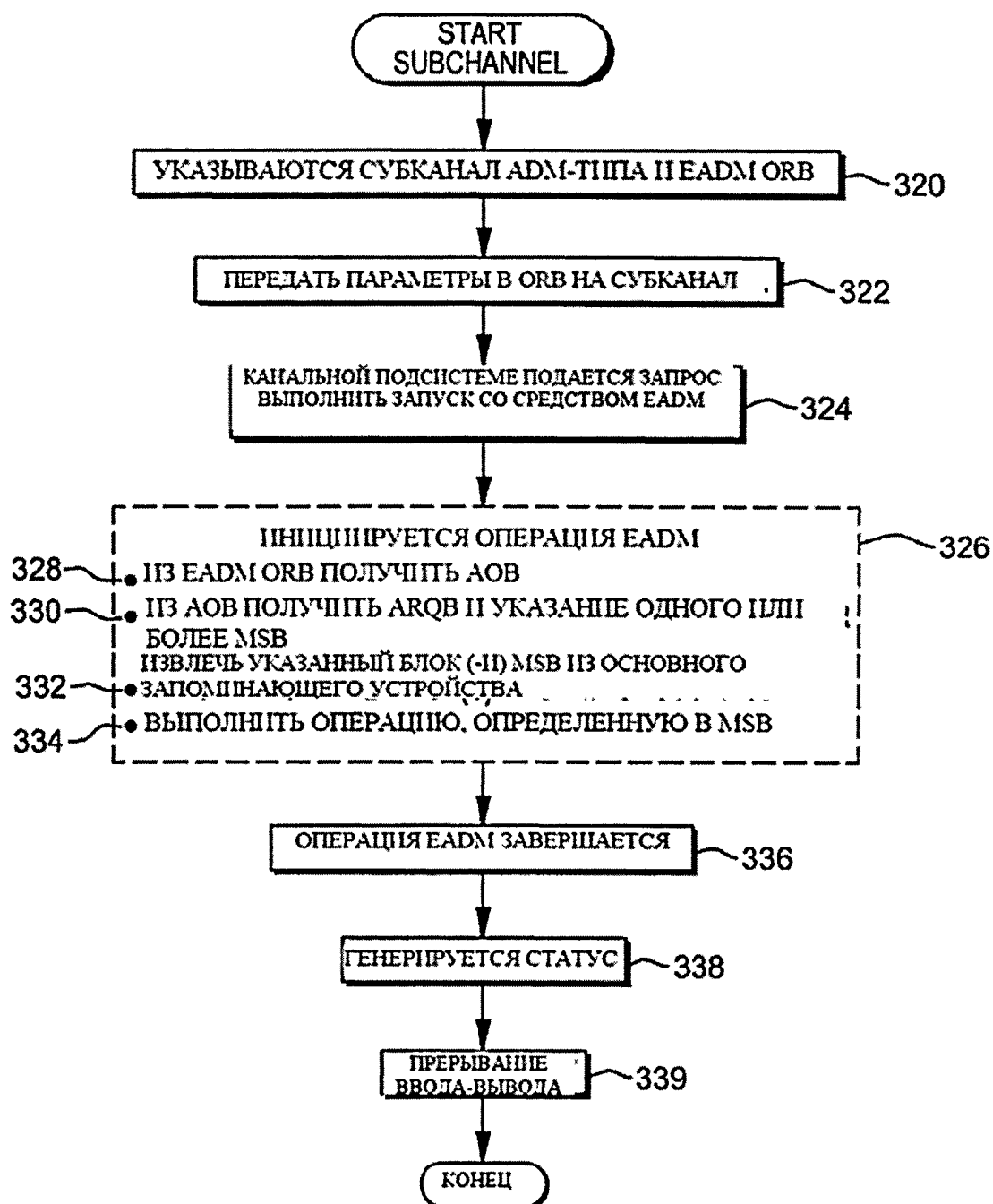
КОСВЕННОЕ АДРЕСНОЕ СЛОВО ДАННЫХ ЕАДМ 280



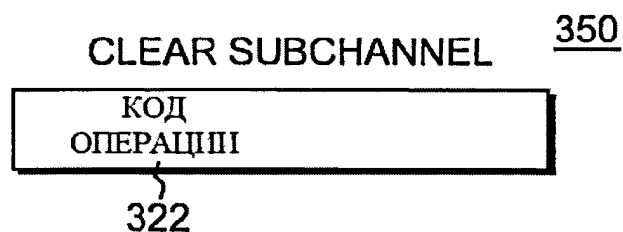
ФИГ. 2Е



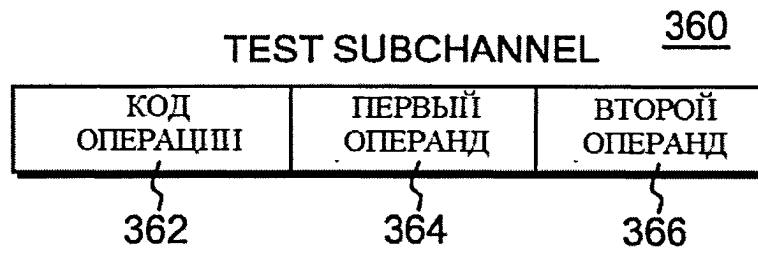
ФИГ. 3А



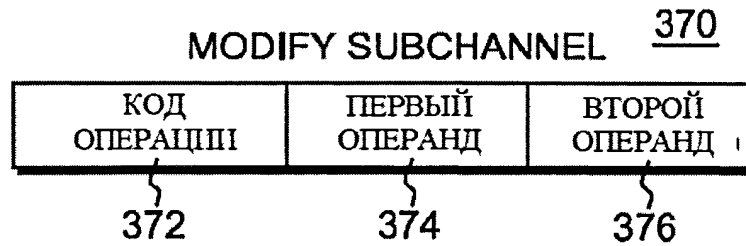
ФИГ. 3Б



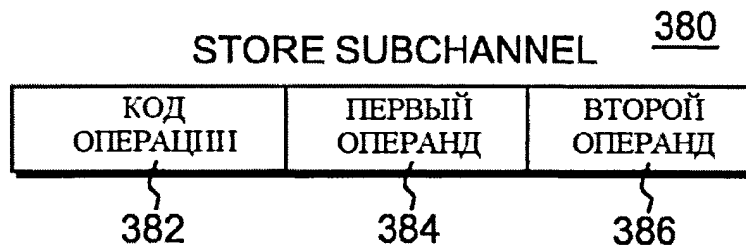
ФИГ. 3В



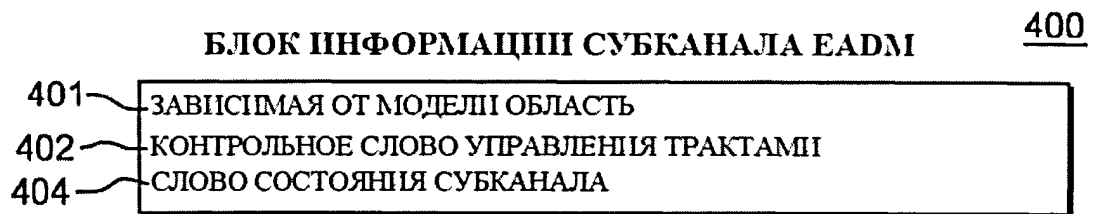
ФИГ. 3Г



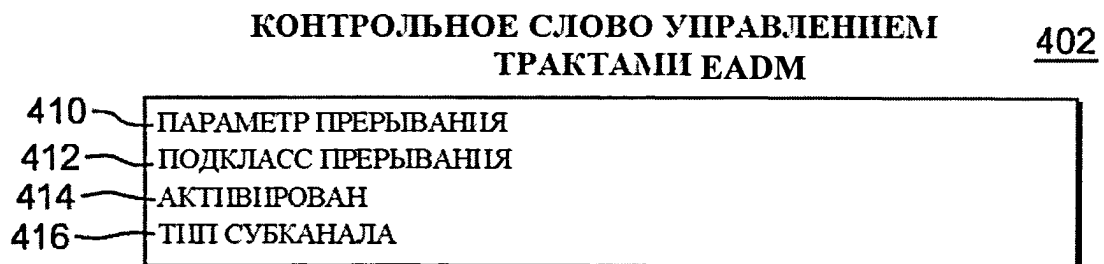
ФИГ. 3Д



ФИГ. 3Е



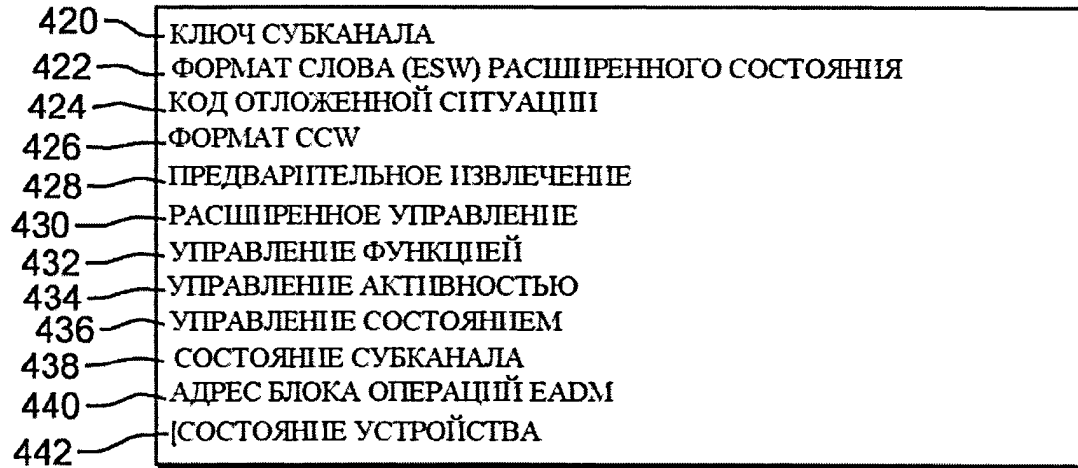
ФИГ. 4А



ФИГ. 4Б

СЛОВО СОСТОЯНИЯ СУБКАНАЛА

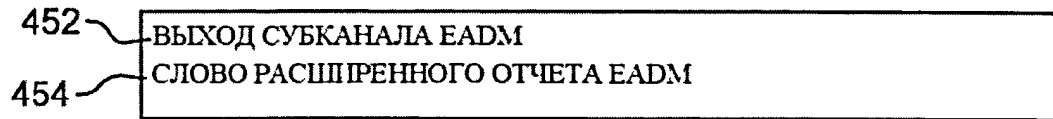
404



ФИГ. 4В

EADM ESW

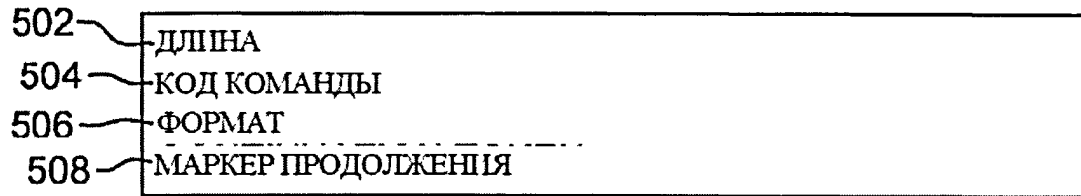
450



ФИГ. 4Г

**БЛОК ЗАПРОСА
СОХРАНЕНИЯ ИНФОРМАЦИИ SCM**

500



ФИГ. 5А

**БЛОК ОТВЕТА
СОХРАНЕНИЯ ИНФОРМАЦИИ SCM** 520

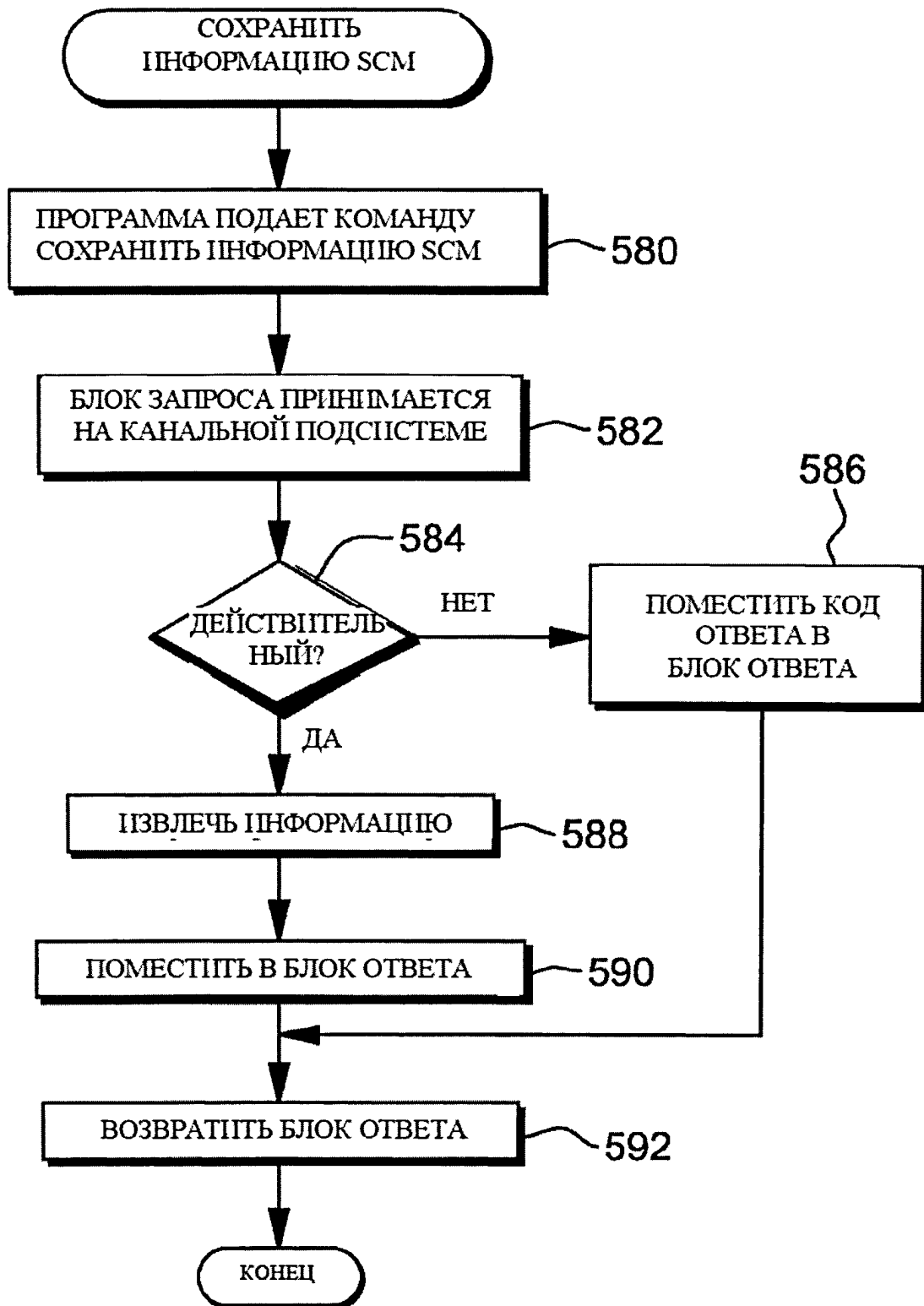
522	ДЛИНА
524	КОД ОТВЕТА
526	ФОРМАТ
528	RQ
530	МАКСИМАЛЬНОЕ ЧИСЛО БЛОКОВ ДЛЯ MSB
532	МАКСИМАЛЬНЫЙ АДРЕС SCM
534	РАЗМЕР ПРИРАЩЕНИЯ SCM
536	МАКСИМАЛЬНОЕ ЧИСЛО MSB
538	МАКСИМАЛЬНОЕ ЧИСЛО НАСТРАИВАЕМЫХ ПРИРАЩЕНИЙ SCM
540	ОБЩЕЕ ЧИСЛО ИНИЦИАЛИЗИРОВАННЫХ ПРИРАЩЕНИЙ SCM ДЛЯ СРС
542	ОБЩЕЕ ЧИСЛО НЕИНИЦИАЛИЗИРОВАННЫХ ПРИРАЩЕНИЙ SCM ДЛЯ СРС
544	РАЗМЕР БЛОКА ИЗМЕРЕНИЯ SCM
546	МАКСИМАЛЬНОЕ ЧИСЛО РЕСУРСНЫХ ЧАСТЕЙ SCM
548	РАЗМЕР ЭЛЕМЕНТА ДАННЫХ SCM
550	МАРКЕР ПРОДОЛЖЕНИЯ
552	СПИСОК АДРЕСОВ ТВЕРДОТЕЛЬНОГО ЗАПОМИНАЮЩЕГО УСТРОЙСТВА

ФИГ. 5Б

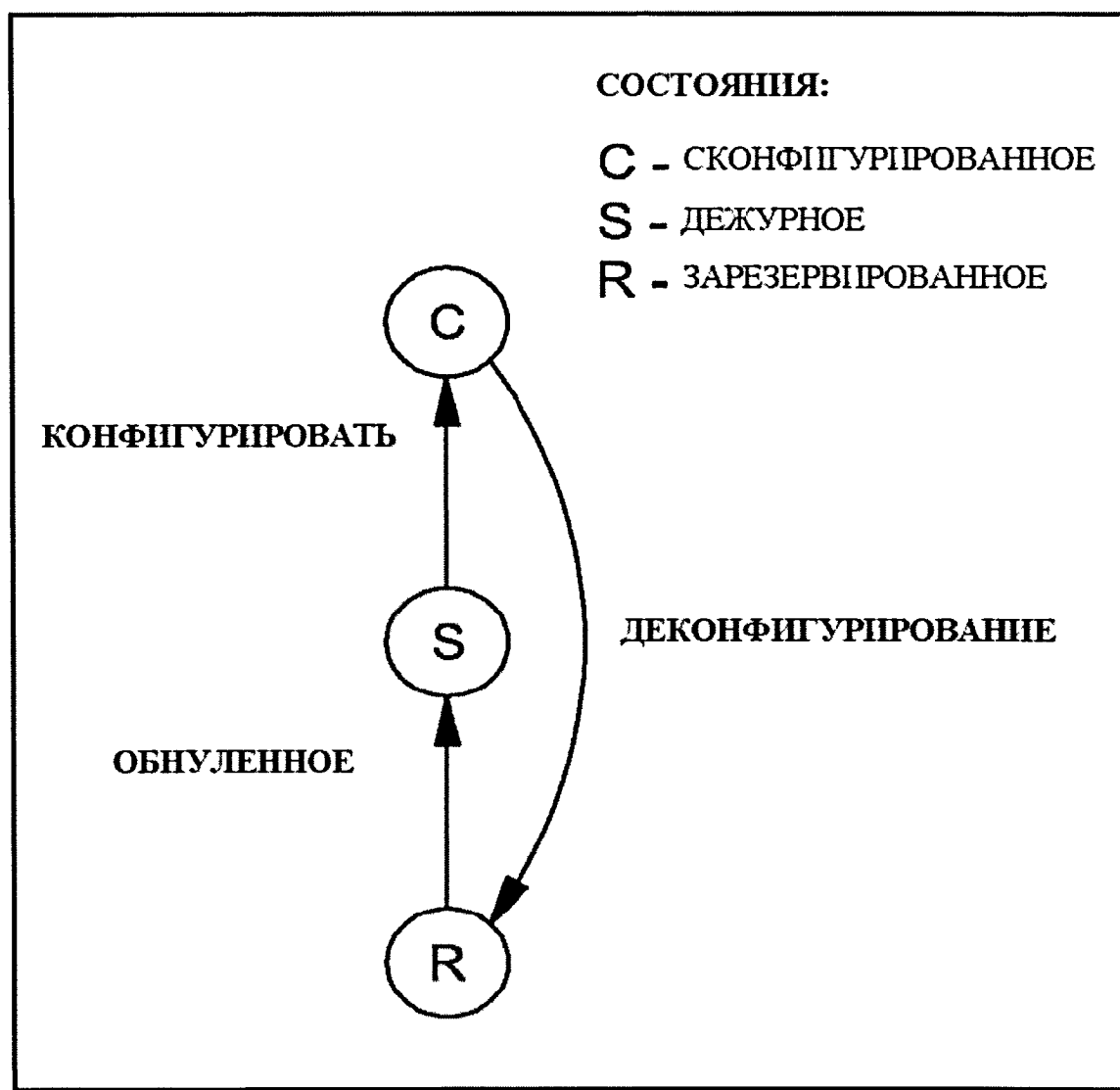
**ЗАПИСЬ СПИСКА АДРЕСОВ
ТВЕРДОТЕЛЬНОГО ЗАПОМИНАЮЩЕГО
УСТРОЙСТВА** 552

560	АДРЕС SCM
562	АТРИБУТ СОХРАНЯЕМОСТИ
564	СОСТОЯНИЕ ОПЕРАЦИИ
566	СОСТОЯНИЕ ДАННЫХ
568	РАНГ
570	R
572	ID РЕСУРСА

ФИГ. 5В



ФИГ. 5Г



ФИГ. 6А

СОСТОЯНИЯ ОПЕРАЦИИ:

ОР - РАБОЧЕЕ

ТЕ - ВРЕМЕННАЯ ОШИБКА

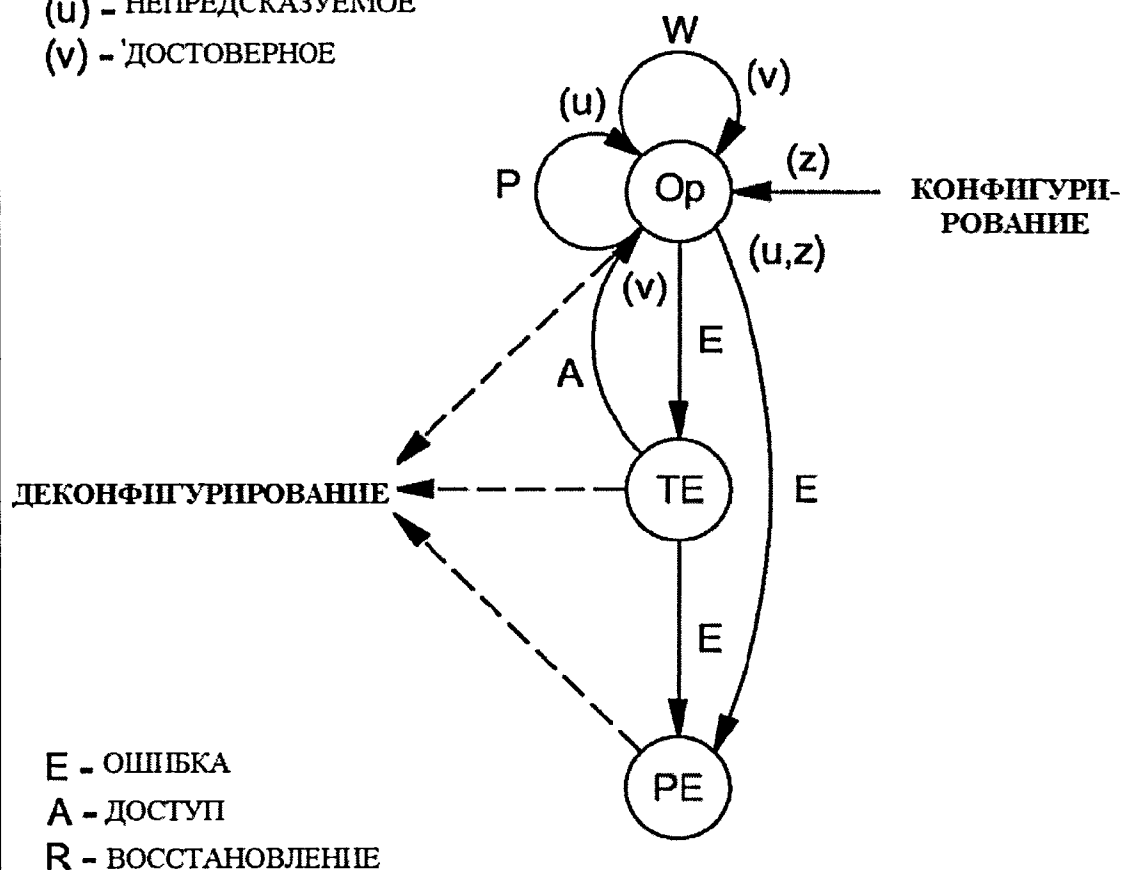
РЕ - ПОСТОЯННАЯ ОШИБКА

СОСТОЯНИЯ ДАННЫХ:

(z) - ОБНУЛЕННОЕ

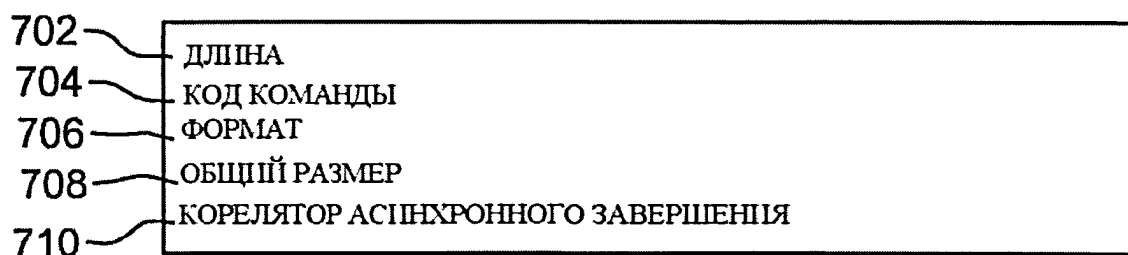
(u) - НЕПРЕДСКАЗУЕМОЕ

(v) - ДОСТОВЕРНОЕ

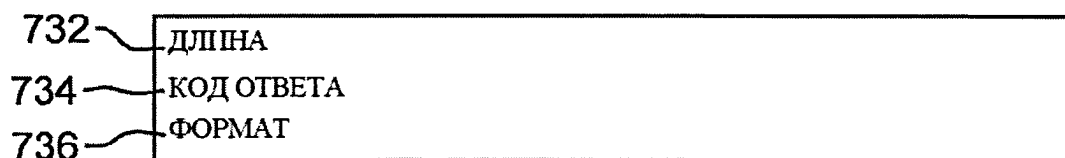
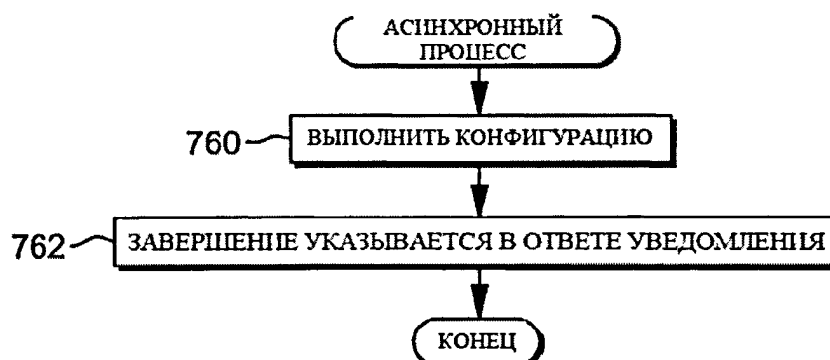
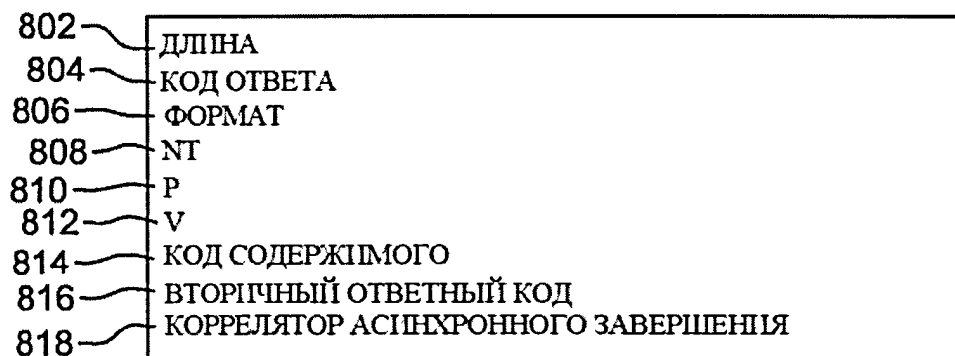


ФИГ. 6Б

**БЛОК ЗАПРОСА КОНФИГУРАЦИИ
ТВЕРДОТЕЛЬНОГО ЗАПОМИНАЮЩЕГО
УСТРОЙСТВА**

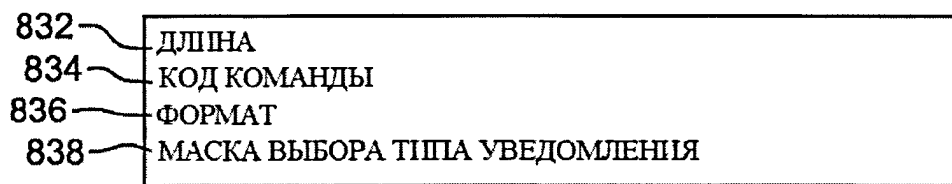
700**ФИГ. 7А**

**БЛОК ОТВЕТА КОНФИГУРАЦИИ
ТВЕРДОТЕЛЬНОГО ЗАПОМИНАЮЩЕГО
УСТРОЙСТВА**

730**ФИГ. 7Б****ФИГ. 7Г****ОТВЕТ УВЕДОМЛЕНИЯ**800**ФИГ. 8А**

**БЛОК ЗАПРОСА СОХРАНЕНИЯ
ИНФОРМАЦИИ СОБЫТИЯ**

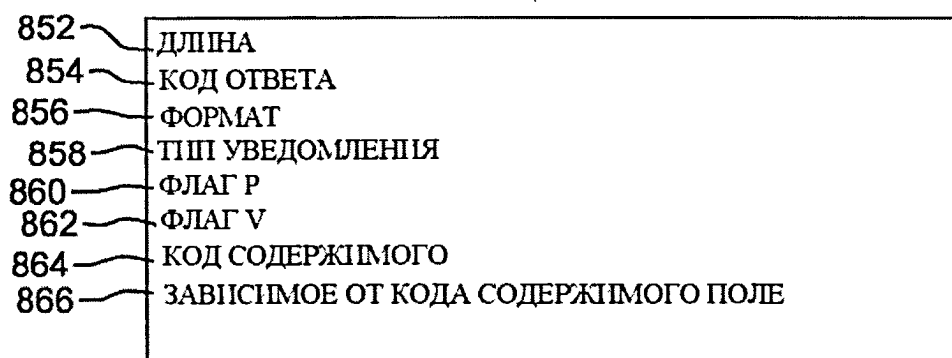
830



ФИГ. 8Б

**БЛОК ОТВЕТА
СОХРАНЕНИЯ
ИНФОРМАЦИИ СОБЫТИЯ**

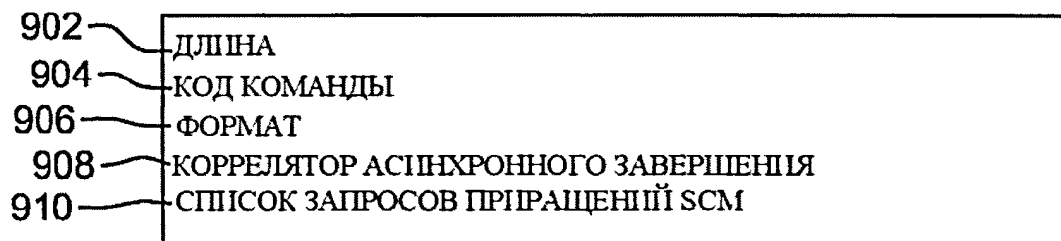
850



ФИГ. 8В

**БЛОК ЗАПРОСА ДЕКОНФИГУРАЦИИ
ТВЕРДОТЕЛЬНОГО ЗАПОМИНАЮЩЕГО
УСТРОЙСТВА**

900



ФИГ. 9А

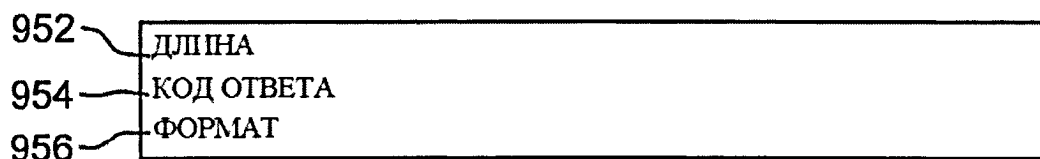
ЗАПИСЬ СПИСКА ЗАПРОСОВ ПРИРАЩЕНИЯ SCM

920

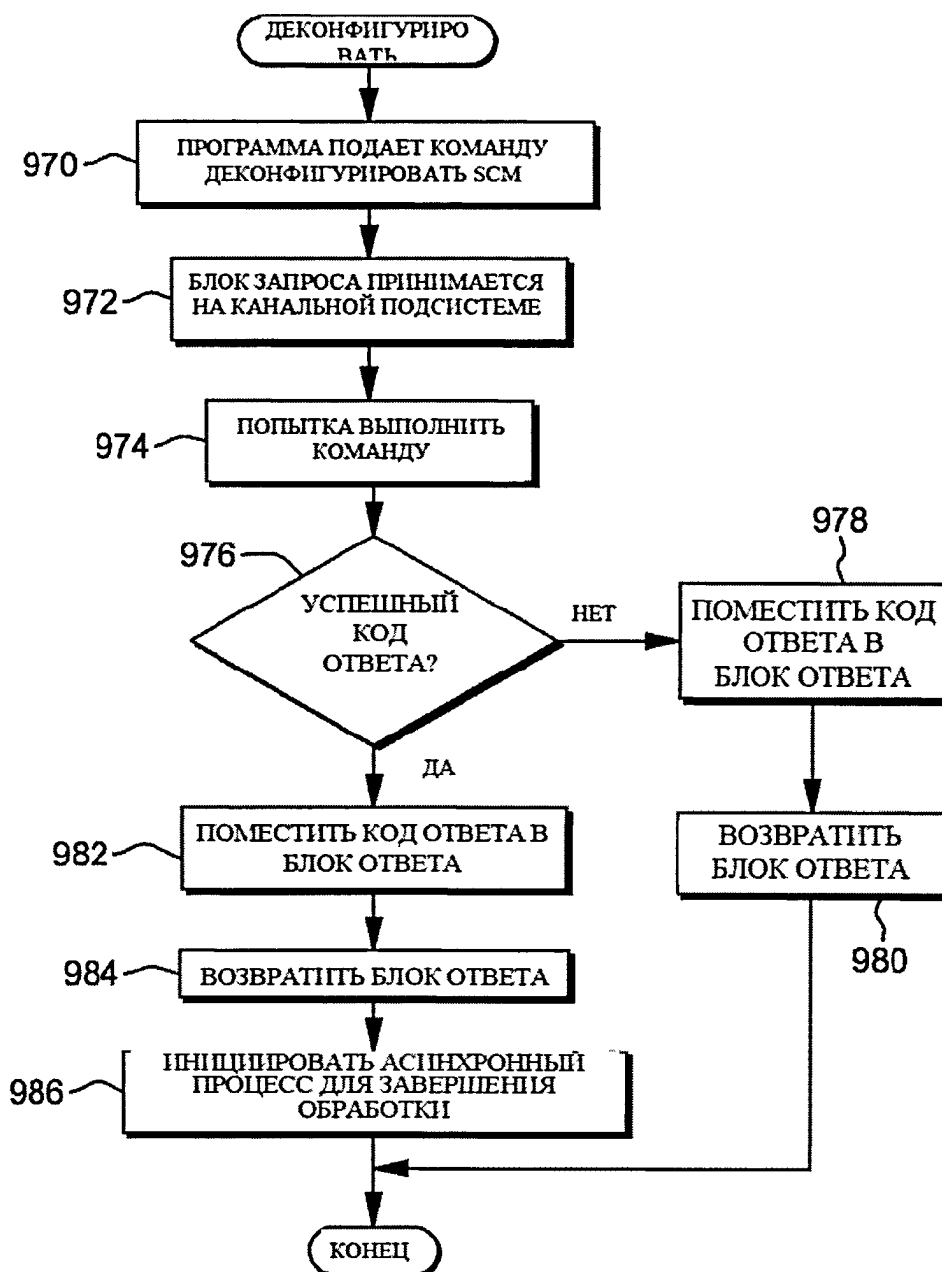


ФИГ. 9Б

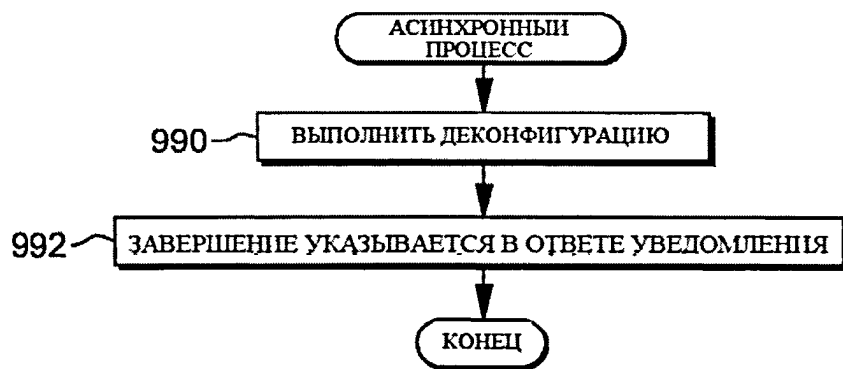
**БЛОК ОТВЕТА (КОМАНДЫ)
ДЕКОНФИГУРАЦИИ ТВЕРДОТЕЛЬНОГО
ЗАПОМНАЮЩЕГО УСТРОЙСТВА**

950

ФИГ. 9В



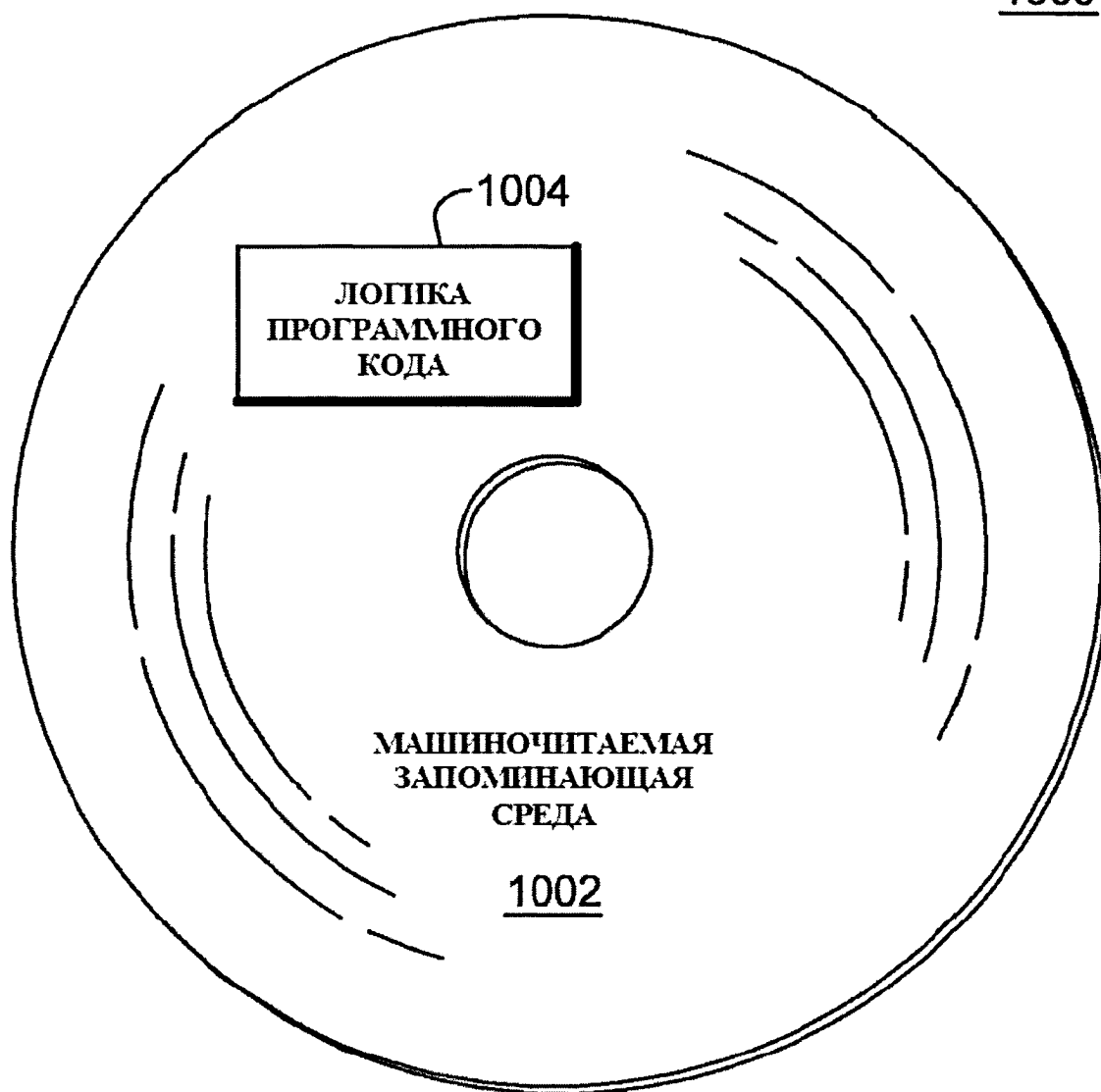
ФИГ. 9Г



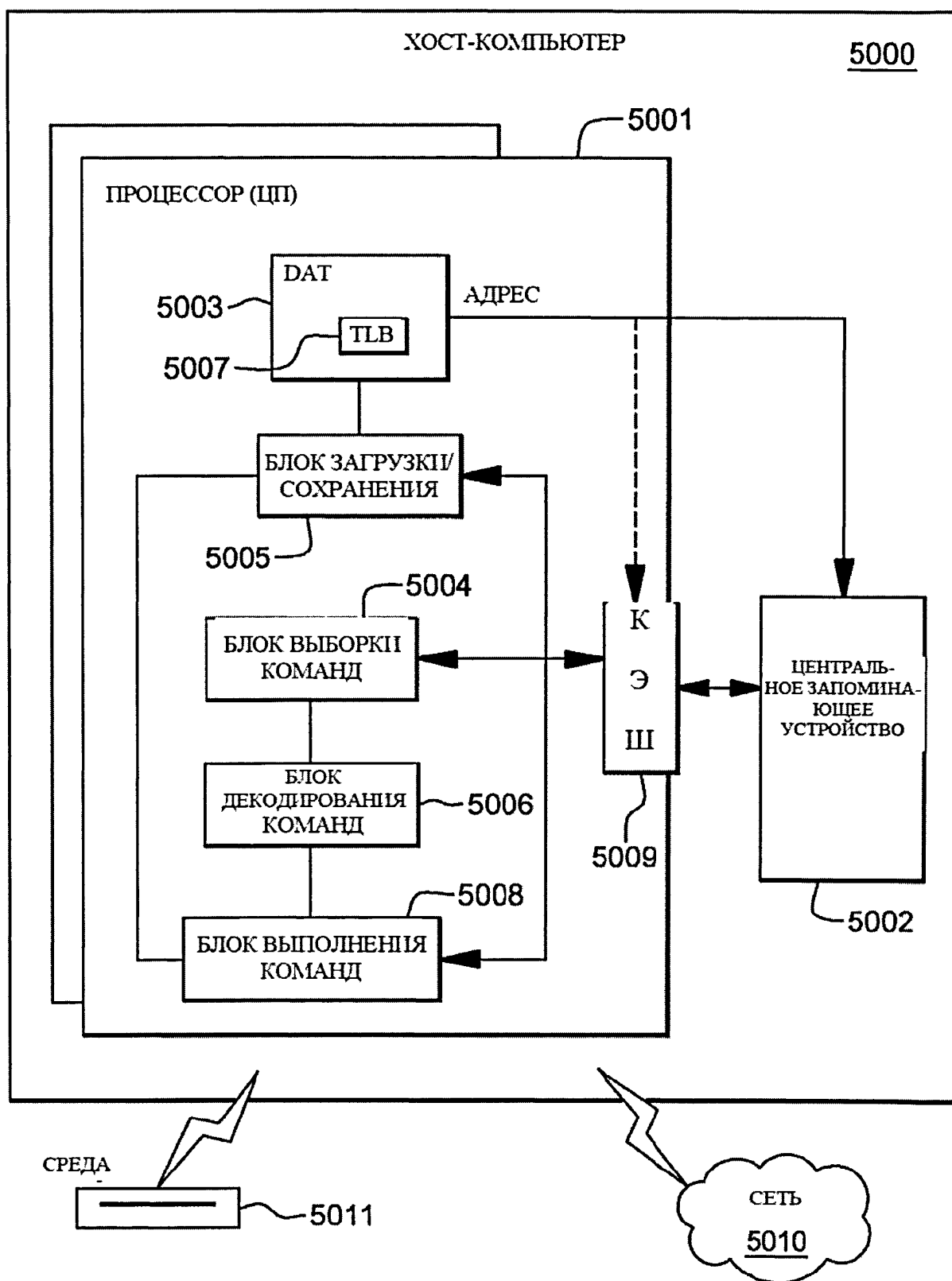
ФИГ. 9Д

КОМПЬЮТЕРНЫЙ
ПРОГРАММНЫЙ
ПРОДУКТ

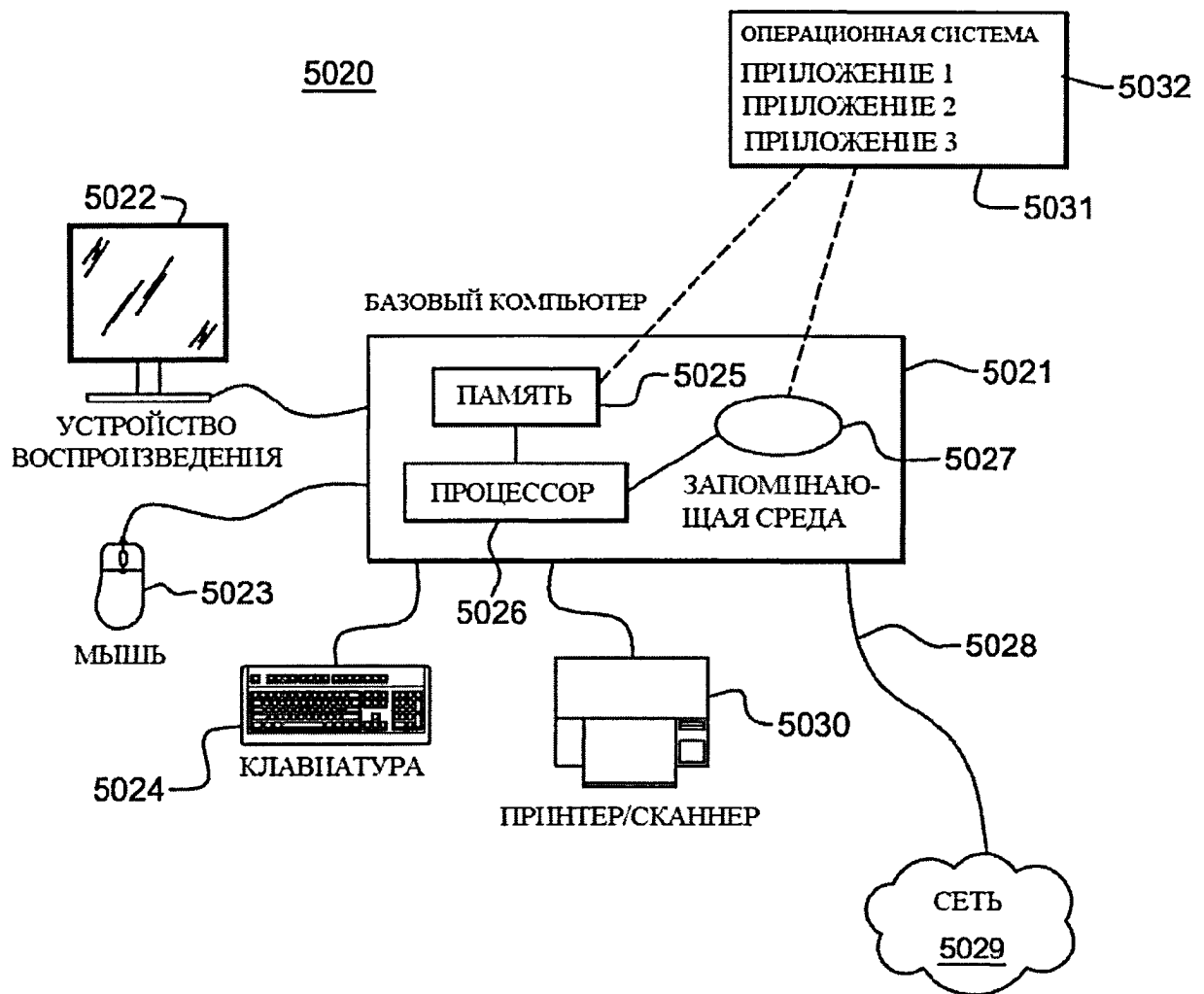
1000



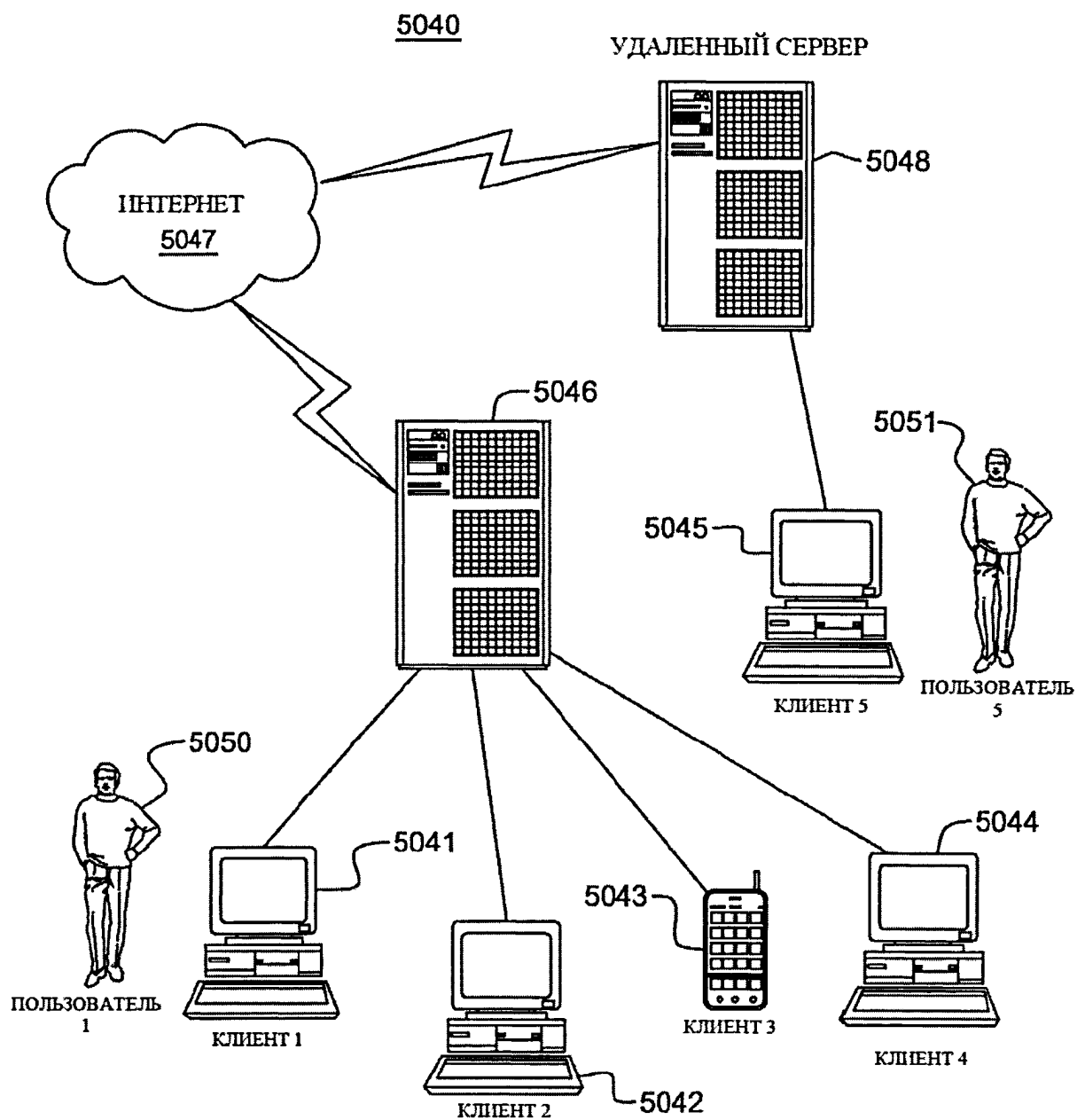
ФИГ. 10



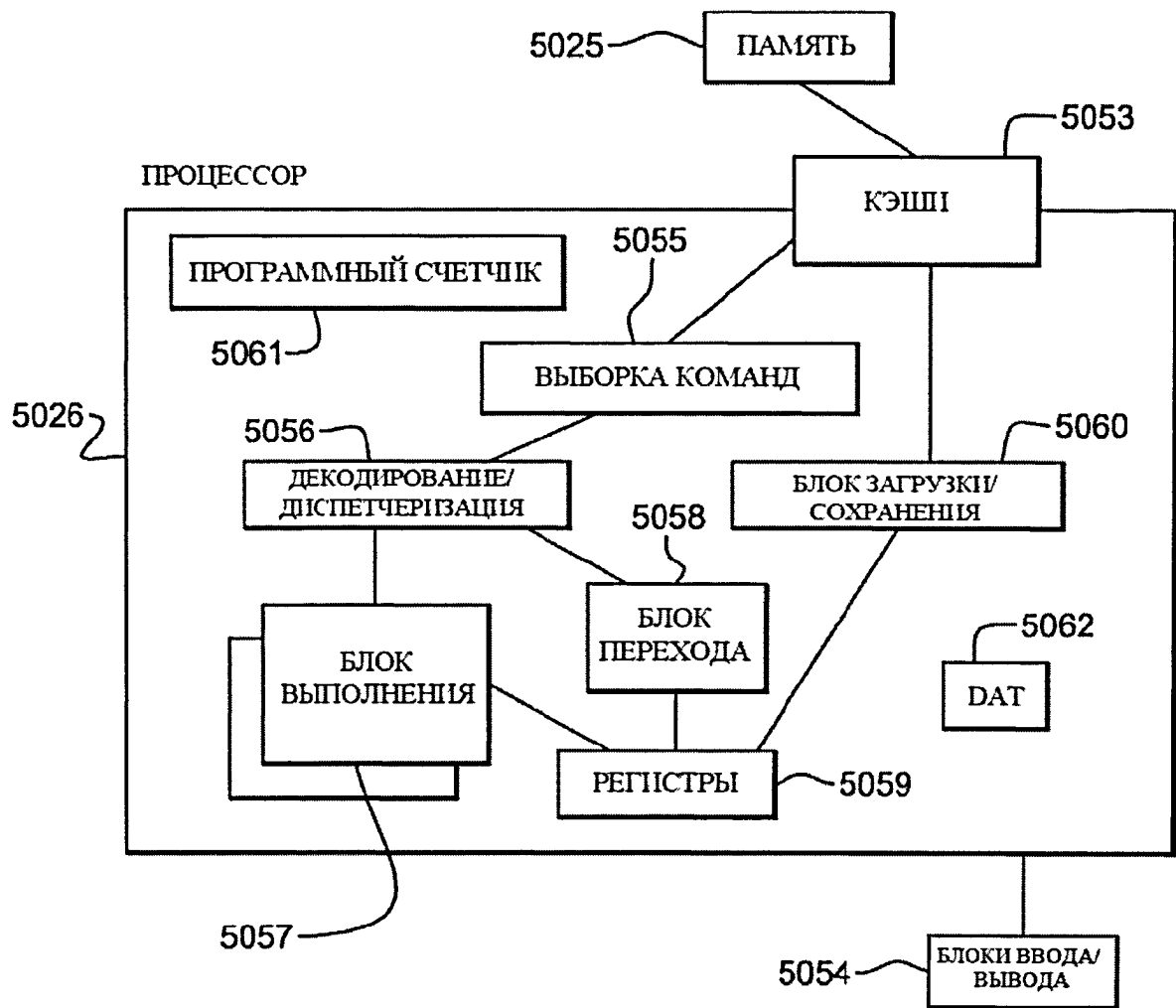
ФИГ. 11



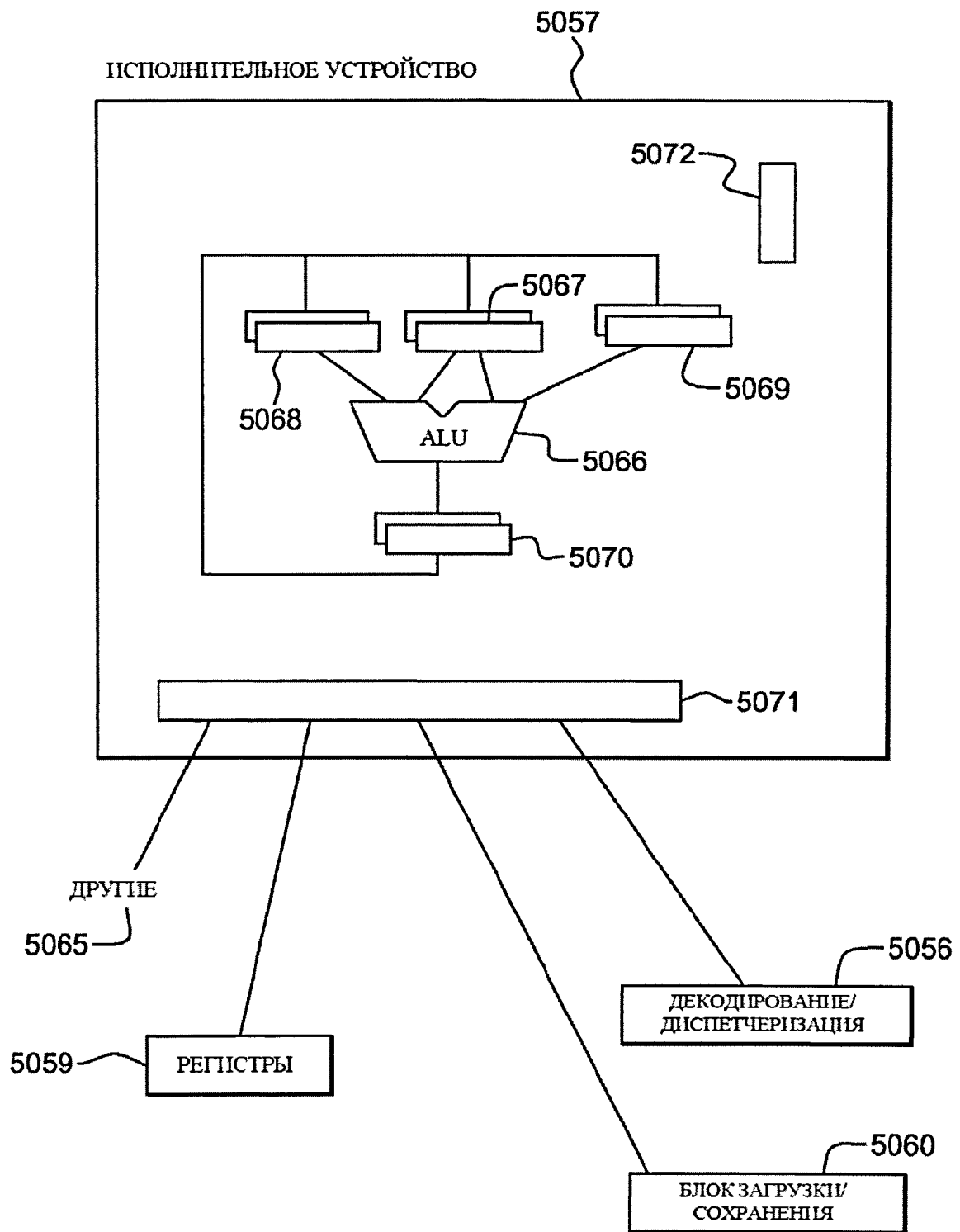
ФИГ. 12



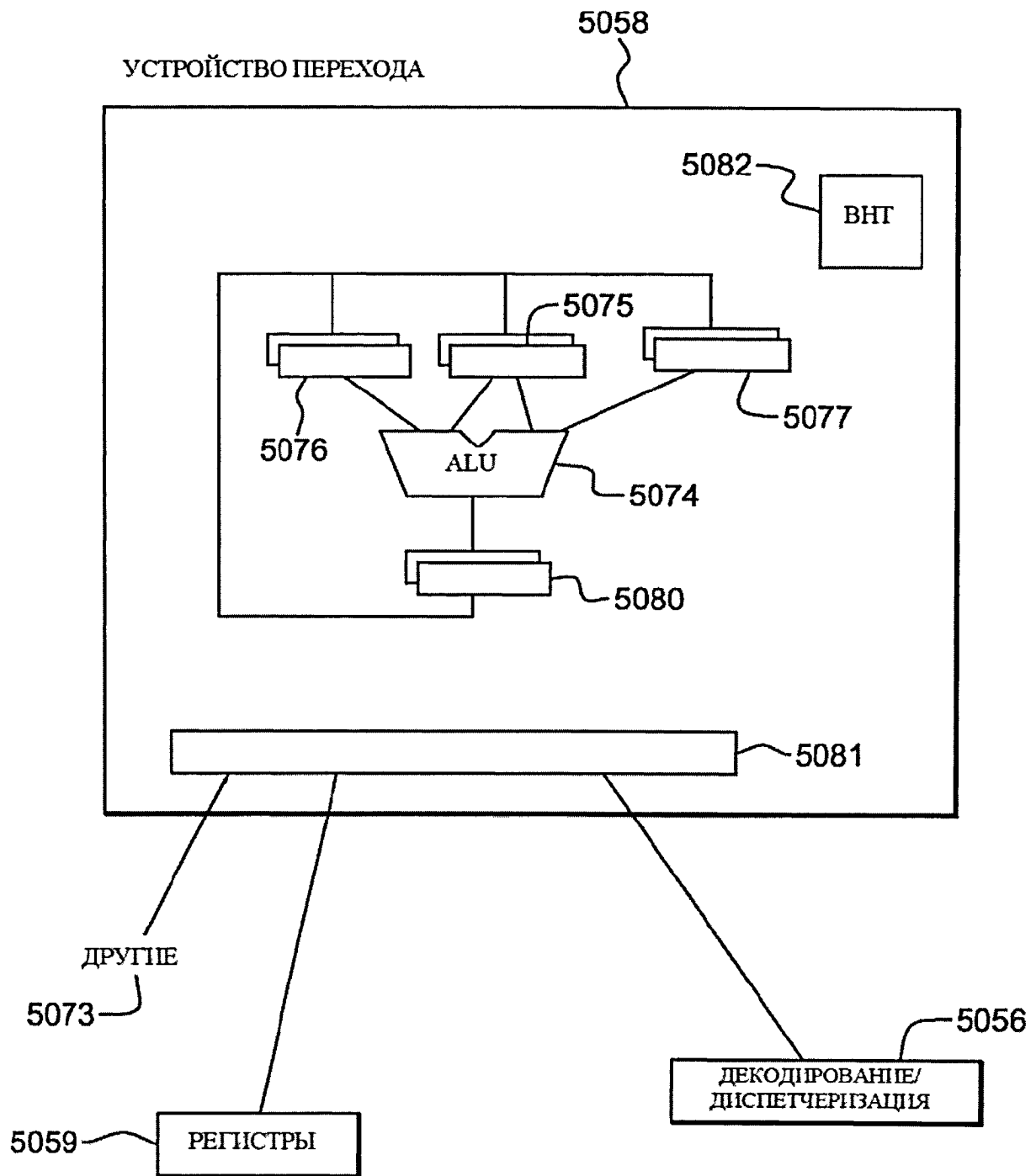
ФИГ. 13



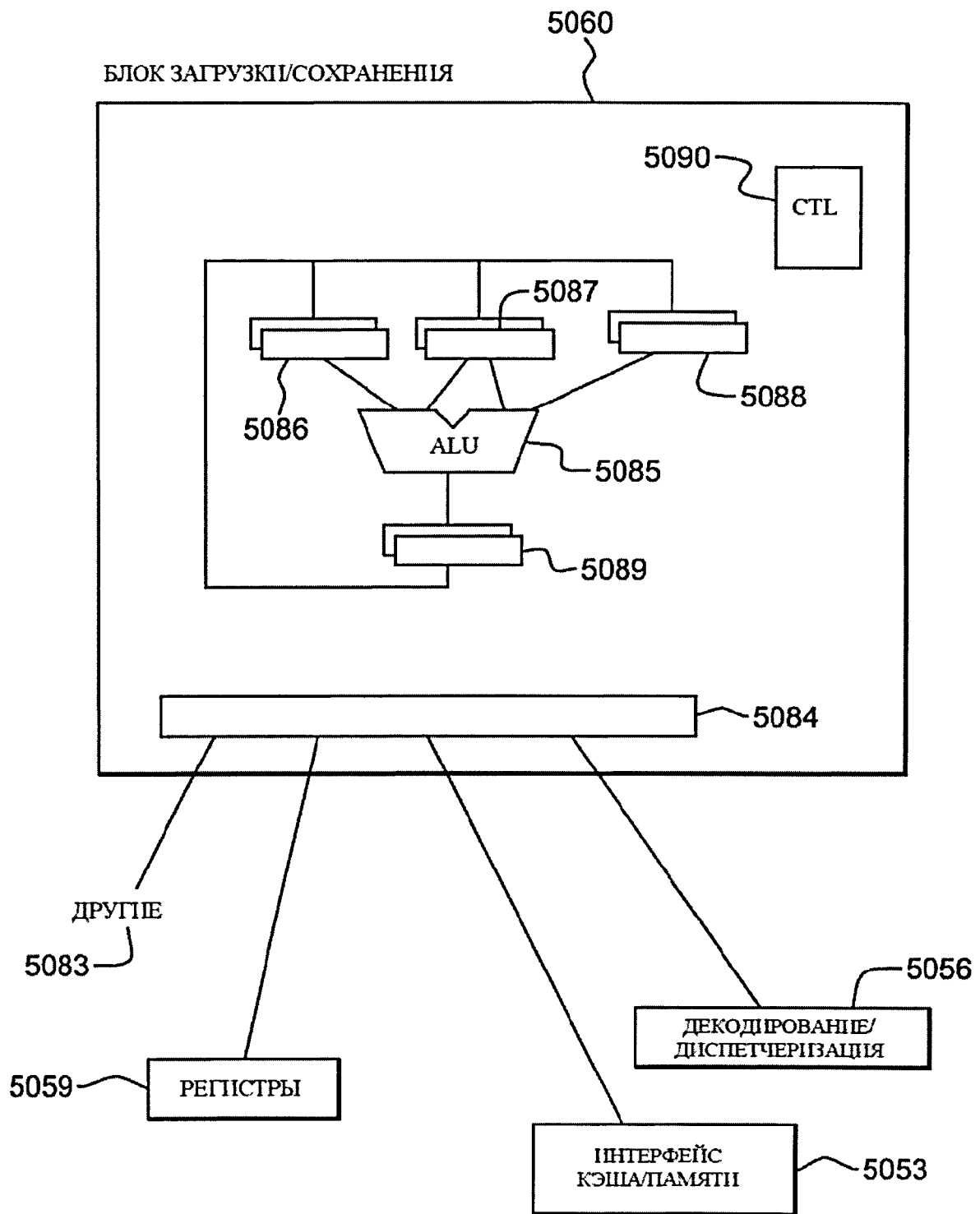
ФИГ. 14



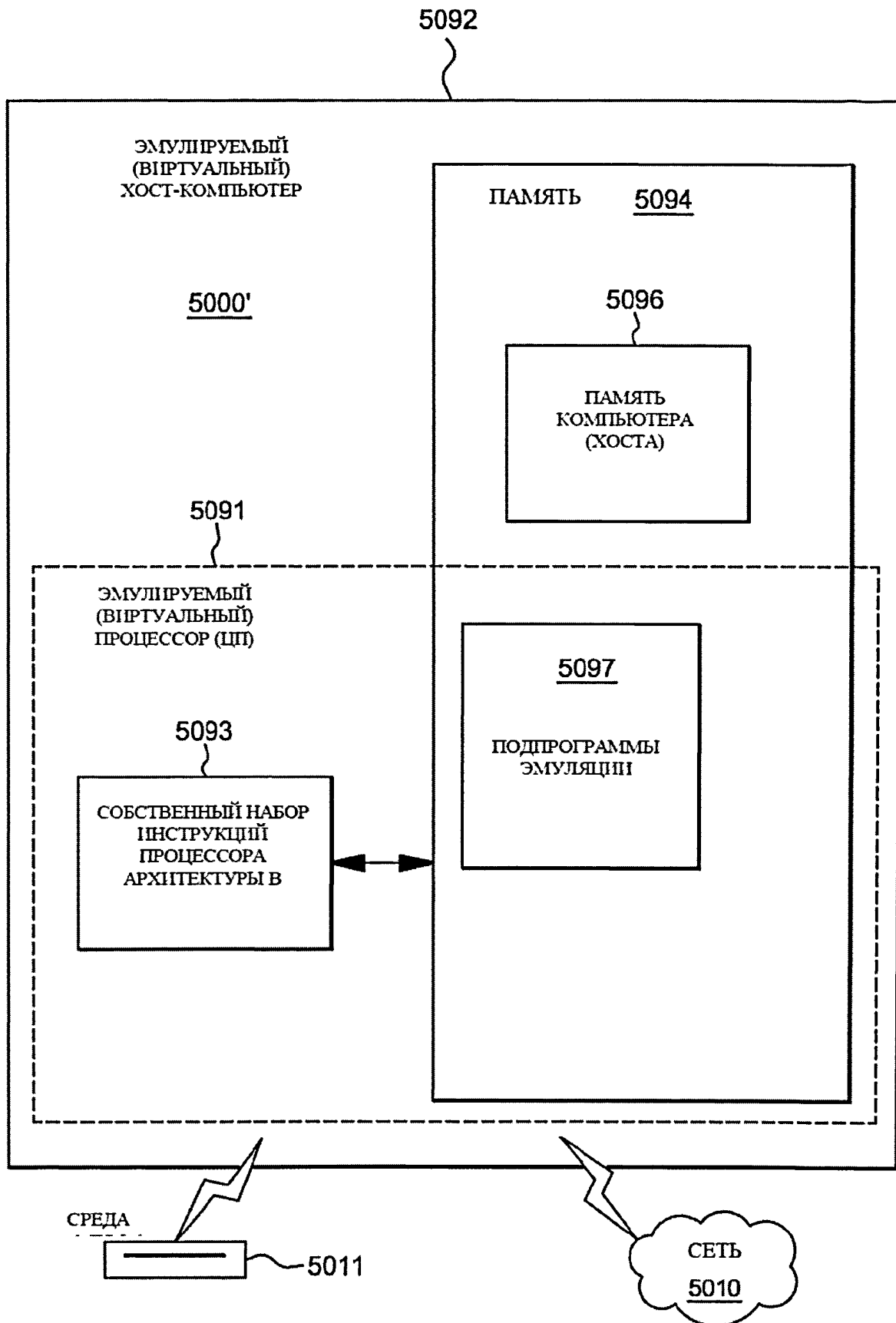
ФИГ. 15А



ФИГ. 15Б



ФИГ. 15В



ФИГ. 16