РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 799 750**(13) **C9**

(51) МПК
*G16B 50/00* (2019.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

# (12) СКОРРЕКТИРОВАННОЕ ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

Примечание: библиография отражает состояние при переиздании

(52) СПК
*G16B 50/00 (2023.02)*

(21)(22) Заявка: **2021118824, 07.06.2017**

(24) Дата начала отсчета срока действия патента:
**07.06.2017**

Приоритет(ы):
(30) Конвенционный приоритет:
**25.04.2017 US 15/497,149;**
**09.03.2017 US 62/469,442;**
**23.02.2017 US 62/462,869;**
**11.01.2017 US 15/404,146;**
**28.10.2016 US 62/414,637;**
**26.09.2016 US 62/399,582;**
**07.06.2016 US 62/347,080**
Номер и дата приоритета первоначальной заявки, из которой данная заявка выделена:
**2018140888 07.06.2016**

(43) Дата публикации заявки: **12.11.2021** Бюл. № **32**

(45) Опубликовано: **11.07.2023**

(15) Информация о коррекции:
**Версия коррекции №1 (W1 C2)**

(48) Коррекция опубликована:
**07.09.2023 Бюл. № 25**

Адрес для переписки:
**190900, Санкт-Петербург, ВОХ-1125, Нилова Мария Иннокентьевна**

(72) Автор(ы):
**ВАН РОЙН, Питер (US),**
**РЮЛЕ, Майкл (US),**
**МЕХЬО, Рами (US),**
**СТОУН, Гэвин (US),**
**ХАМ, Марк (US),**
**ОДЖАРД, Эрик (US),**
**ПТАШЕК, Амнон (US)**

(73) Патентообладатель(и):
**ИЛЛЮМИНА, ИНК. (US)**

(56) Список документов, цитированных в отчете о поиске: **RU 2282242 C2, 20.08.2006. AU 2014335877 A1, 05.05.2016. US 9322872 B2, 26.04.2016.**

## (54) БИОИНФОРМАЦИОННЫЕ СИСТЕМЫ, УСТРОЙСТВА И СПОСОБЫ ДЛЯ ВЫПОЛНЕНИЯ ВТОРИЧНОЙ И/ИЛИ ТРЕТИЧНОЙ ОБРАБОТКИ

(57) Реферат:

Изобретение относится к биоинформатике. Предложены система, способ и устройство для выполнения биоинформационного анализа на данных секвенирования генома. В частности, предложены способ и система для улучшения точности определения вариантов посредством совместной оценки ридов, которые картируют две или более областей референсной последовательности, которые являются гомологичными. Способ включает: обращение к скоплению ридов; определение набора вариантов-кандидатов из скопления; оценку каждого варианта; формирование файла определения вариантов. Система содержит: компьютер и устройство хранения с инструкциями для инициирования обращения к скоплению ридов,



RU 2799750 C9

RU 2799750 C9

определения вариантов-кандидатов и установления порядка их обработки, оценки каждого варианта и формирования файла определения вариантов. Также предложено машиночитаемое устройство хранения с инструкциями, инициирующими выполнение операций для улучшения точности определения вариантов. Изобретение расширяет арсенал средств для реализации биоинформационных протоколов. Технический результат, достигаемый заявленным изобретением, заключается в увеличении точности определения вариантов. 3 н. и 17 з.п. ф-лы, 51 ил.

RU 2799750 C9

RU 2799750 C9

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(19) **RU** (11) **2 799 750**(13) **C9**

(51) Int. Cl.
*G16B 50/00* (2019.01)

## (12) ABSTRACT OF INVENTION

Note: Bibliography reflects the latest situation

(52) CPC
*G16B 50/00 (2023.02)*

(21)(22) Application: **2021118824, 07.06.2017**

(24) Effective date for property rights:
**07.06.2017**

Priority:
(30) Convention priority:
**25.04.2017 US 15/497,149;**
**09.03.2017 US 62/469,442;**
**23.02.2017 US 62/462,869;**
**11.01.2017 US 15/404,146;**
**28.10.2016 US 62/414,637;**
**26.09.2016 US 62/399,582;**
**07.06.2016 US 62/347,080**
Number and date of priority of the initial application, from which the given application is allocated:
**2018140888 07.06.2016**

(43) Application published: **12.11.2021** Bull. № **32**

(45) Date of publication: **11.07.2023**

(15) Correction information:
**Corrected version no1 (W1 C2)**

(48) Corrigendum issued on:
**07.09.2023 Bull. № 25**

Mail address:
**190900, Sankt-Peterburg, BOX-1125, Nilova Mariya Innokentevna**

(72) Inventor(s):
**VAN ROJN, Piter (US),**
**RYULE, Majkl (US),**
**MEKHO, Rami (US),**
**STOUN, Gevin (US),**
**KHAM, Mark (US),**
**ODZHARD, Erik (US),**
**PTASHEK, Amnon (US)**

(73) Proprietor(s):
**ILLYUMINA, INK. (US)**

RU 2799750 C9

RU 2799750 C9

## (54) BIOINFORMATION SYSTEMS, DEVICES AND METHODS FOR SECONDARY AND/OR TERTIARY PROCESSING

(57) Abstract:

FIELD: bioinformatics.

SUBSTANCE: system, method and device for performing bioinformatics analysis on genome sequencing data. In particular, a method and system is provided for improving the accuracy of variant detection by co-scoring reads that map two or more regions of a reference sequence that are homologous. The method includes: accessing a cluster of reads; determining a set of candidate options from the cluster; evaluation of each option; formation of a file of definitions of options. The system contains: a computer and a storage device with instructions for initiating a call to a collection of reads, determining candidate variants and establishing the order of their processing, evaluating each variant and generating a variant definition file. Also provided is a computer-readable storage device with instructions

initiating the execution of operations to improve the accuracy of determining options. The invention expands the arsenal of tools for the implementation of bioinformatic protocols.

EFFECT: increased accuracy of determining variants.

20 cl, 51 dwg



RU 2799750 C9

Перекрестная ссылка на родственную заявку

[001] Настоящая заявка испрашивает приоритет в соответствии с §119(e) раздела 35 Свода законов США по предварительной заявке на патент США №62/347,080, поданной 7 июня 2016 г., озаглавленной «Bioinformatics Systems, Apparatuses, and Methods Executed on an Integrated Circuit Processing Platform»; по предварительной заявке на патент США №62/399,582, поданной 26 сентября 2016 г., озаглавленной «Bioinformatics Systems, Apparatuses, and Methods Executed on an Integrated Circuit Processing Platform»; предварительной заявке на патент США №62/414,637, поданной 28 октября 2016 г., озаглавленной «Bioinformatics Systems, Apparatuses, and Methods Executed on an Integrated Circuit Processing Platform»; по предварительной заявке на патент США №62/462,869, поданной 23 февраля 2017 г., озаглавленной «Bioinformatics Systems, Apparatuses, and Methods Executed on a Quantum Processing Platform»; по предварительной заявке на патент США №62/469,442, поданной 9 марта 2017 г., озаглавленной «Bioinformatics Systems, Apparatuses, and Methods Executed on an Integrated Circuit Processing Platform», ссылки на которые сделаны в соответствии с данным параграфом и которые полностью включены в настоящий документ посредством ссылки. Настоящая заявка является также частично продолжающейся заявкой и испрашивает приоритет в соответствии с §120 раздела 35 Свода законов США по заявке на патент США №15/404,146, поданной 11 января 2017 г., озаглавленной «Genomic Infrastructure for On-Site or Cloud-Based DNA and RNA Processing and Analysis»; и по заявке на патент США №15/497,149, поданной 25 апреля 2017 г., озаглавленной «Bioinformatics Systems, Apparatuses, and Methods Executed on a Quantum Processing Platform».

Область техники

[002] Объект изобретения, описанный в настоящем документе, относится к биоинформатике и, в частности, к системам, устройствам и способам реализации биоинформационных протоколов, таких как выполнение одной или более функций для анализа геномных данных на интегральной схеме, такой как платформа аппаратной обработки.

Уровень техники

[003] Как подробно описано в настоящем документе, некоторые основные вычислительные проблемы анализа секвенирования ДНК с высокой пропускной способностью заключаются в необходимости справляться со взрывным ростом доступных геномных данных, потребности в повышенной точности и чувствительности при сборе этих данных и потребности в быстрых, эффективных и точных вычислительных средствах при выполнении анализа на широком диапазоне наборов данных секвенирования, полученных из таких геномных данных.

[004] Для того чтобы идти в ногу с такой повышенной пропускной способностью секвенирования, обеспечиваемой секвенаторами нового поколения, обычно применяли многопоточные программные средства, которые исполнялись на все большем и большем количестве более быстрых процессоров в вычислительных кластерах с дорогостоящей высокодоступной памятью, которая требуют существенной энергии и значительных расходов на информационно-техническое обеспечение. Важно отметить, что будущие повышения пропускной способности секвенирования приведут к ускорению роста затрат в реальном денежном выражении на эти решения по вторичной обработке.

[005] Для решения по меньшей мере частично этих и других подобных проблем предложены устройства, системы и способы их использования, описанные в настоящем документе.

Раскрытие сущности изобретения

[006] Настоящее изобретение относится к устройствам, системам и способам их использования при выполнении одного или более протоколов геномики и/или биоинформатики на данных, формируемых посредством процедуры первичной обработки, например на данных генетической последовательности. Например, согласно различным аспектам в настоящем документе предложены устройства, системы и способы, выполненные с возможностью осуществления протоколов вторичного и/или третичного анализа генетических данных, таких как данные, сформированные путем секвенирования РНК и/или ДНК, например, с помощью секвенатора нового поколения (СНП). В конкретных примерах реализации предусмотрены один или более конвейеров вторичной обработки для обработки данных генетической последовательности. В других вариантах реализации предусмотрены один или более конвейеров третичной обработки для обработки данных генетической последовательности, например, где конвейеры и/или их отдельные элементы обеспечивают превосходную чувствительность и улучшенную точность в более широком диапазоне полученных из последовательности данных по сравнению с доступным в настоящее время в данной области техники.

[007] Например, в настоящем документе предложена система, такая как для осуществления одного или более конвейеров анализа последовательности и/или генома на данных генетической последовательности и/или других полученных из нее данных. В различных вариантах реализации система может включать в себя один или более электронных источников данных, которые обеспечивают цифровые сигналы, представляющие множество ридов генетических и/или геномных данных, например, где каждое из множества ридов геномных данных включает в себя последовательность нуклеотидов. Система может также включать в себя память, например, DRAM или кэш, такую как для хранения одного или более из последовательных ридов, одной или множества генетических референсных последовательностей и одного или более индексов одной или более генетических референсных последовательностей. Система может дополнительно включать в себя одну или более интегральных схем, таких как FPGA, ASIC или sASIC, и/или ЦПУ и/или ГПУ, причем интегральная схема, например, применительно к FPGA, ASIC или sASIC, может быть образована из набора жестко смонтированных цифровых логических схем, которые взаимосвязаны множеством физических электрических межсоединений. Система может дополнительно включать в себя квантовый вычислительный процессор для использования при реализации одного или более способов, описанных в настоящем документе.

[008] В различных вариантах реализации одно или более из множества электрических межсоединений может представлять собой вход в одну или более интегральных схем, которые могут быть соединены или выполнены с возможностью соединения, например, напрямую, через подходящее монтажное соединение или опосредованно, например, посредством беспроводного сетевого соединения (например, облака или гибридного облака), с электронным источником данных. Независимо от соединения с секвенатором интегральная схема по настоящему изобретению может быть выполнена с возможностью приема множества ридов геномных данных, например непосредственно из секвенатора или из связанной памяти. Риды могут быть представлены в цифровом закодированном виде в стандартном файловом формате FASTQ или BCL. Соответственно, система может включать в себя интегральную схему, имеющую одно или более электрических межсоединений, которые могут представлять собой физическое межсоединение, включающее в себя интерфейс памяти, чтобы обеспечивать интегральной схеме возможность доступа к памяти.

[009] В частности, жестко смонтированная цифровая логическая схема интегральной

схемы может быть выполнена в виде набора движков обработки, такого где каждый движок обработки может быть сформирован из подмножества жестко смонтированных цифровых логических схем для выполнения одного или более этапов в конвейере анализа последовательности, генома и/или третичного анализа, как описано ниже в настоящем документе, на множестве ридов генетических данных, а также на других данных, полученных из генетических данных. Например, каждое подмножество жестко смонтированных цифровых логических схем может быть в монтажной конфигурации для выполнения одного или более этапов в конвейере анализа. Кроме того, в том случае, когда интегральная схема представляет собой матрицу FPGA, такие этапы в процессе анализа последовательности или дальнейшего анализа могут включать в себя частичное изменение конфигурации матрицы FPGA во время процесса анализа.

[0010] В частности, набор движков обработки может включать в себя модуль картирования, например в монтажной конфигурации, чтобы в соответствии по меньшей мере с некоторыми из последовательности нуклеотидов в риде из множества ридов осуществлять доступ к индексу одной или более генетических референсных последовательностей из памяти через интерфейс памяти для картирования рида на один или более сегментов указанных одной или более генетических референсных последовательностей на основе индекса. Кроме того, набор движков обработки может включать в себя модуль выравнивания в монтажной конфигурации, чтобы получать доступ к одной или более генетическим референсным последовательностям из памяти через интерфейс памяти для выравнивания рида, например, картированного рида, на одну или более позиций в одном или более сегментах указанных одной или более генетических референсных последовательностей, например, полученных из модуля картирования и/или хранящихся в памяти.

[0011] Набор движков обработки также может включать в себя модуль сортировки, чтобы сортировать каждый выровненный рид в соответствии с одной или более позициями в одной или более генетических референсных последовательностей. Кроме того, набор движков обработки может включать в себя модуль определения вариантов, такой как для обработки картированных, выровненных и/или сортированных ридов, например, относительно референсного генома, для создания тем самым файла записи HMM, и/или определения вариантов для работы с ними, и/или детализации вариаций между секвенированными генетическими данными и данными референсного генома. В различных случаях одно или более из множества физических электрических соединений может включать в себя выход из интегральной схемы для обмена результирующими данными из модуля картирования и/или модулей выравнивания, и/или сортировки, и/или определения вариантов.

[0012] В частности, что касается модуля картирования, то в различных вариантах реализации предложена система для осуществления конвейера анализа картирования на множестве ридов генетических данных с помощью индекса генетических референсных данных. В различных случаях генетическая последовательность, например, рид и/или генетические референсные данные, могут быть представлены последовательностью нуклеотидов, которая может храниться в памяти системы. Модуль картирования может быть включен в интегральную схему и может быть сформирован из множества предварительно сконфигурированных и/или жестко смонтированных цифровых логических схем, которые соединены между собой множеством физических электрических межсоединений, причем физические электрические межсоединения могут включать в себя интерфейс памяти для обеспечения интегральной схемы возможностью доступа к памяти. В более конкретных вариантах реализации жестко смонтированные

цифровые логические схемы могут быть выполнены в виде набора движков обработки, такого где каждый движок обработки может быть сформирован подмножеством жестко смонтированных цифровых логических схем для выполнения одного или более этапов в конвейере анализа последовательностей на множестве ридов геномных данных.

[0013] Например, в одном варианте реализации набор движков обработки может включать в себя модуль картирования в жестко смонтированной конфигурации, где модуль картирования и/или один или более из его движков обработки выполнены с возможностью приема рида геномных данных, например посредством одного или более из множества физических электрических межсоединений, и выделения части рида таким образом, чтобы формировать из него затравку. В таком случае рид может быть представлен последовательностью нуклеотидом, а затравка может представлять подмножество последовательности нуклеотидов, представленной ридом. Модуль картирования может включать в себя память или быть выполнен с возможностью подключения к памяти, которая содержит одно или более ридов, одну или более затравок ридов, по меньшей мере часть одного или более референсных геномов и/или один или более индексов, причем такой индекс построен из одного или более референсных геномов. В определенных случаях движок обработки модуля картирования может использовать затравку и индекс для вычисления адреса в индексе на основе затравки.

[0014] После того, как адрес вычислен или получен и/или сохранен иным образом, например, во встроенной или внешней памяти, к этому адресу можно получать доступ в индексе в памяти, чтобы принимать запись из адреса, например, запись, представляющую информацию о позиции в генетической референсной последовательности. Затем эта информация о позиции может быть использована для определения одной или более совпадающих позиций из рида в генетической референсной последовательности на основе записи. Потом по меньшей мере одна из совпадающих позиций может быть выведена в память через интерфейс памяти.

[0015] В другом варианте реализации набор движков обработки может включать в себя модуль выравнивания, например в предварительно сконфигурированной и/или жестко смонтированной конфигурации. В этом случае один или более из движков обработки могут быть выполнены с возможностью приема одной или более картированных позиций для данных рида посредством одного или более из множества физических электрических межсоединений. После этого можно получать доступ к каждой картированной позиции в памяти (внутренней или внешней) для извлечения сегмента референсной последовательности/генома, соответствующего картированной позиции. На каждом извлеченном референсном сегменте можно вычислить выравнивание рида вместе с оценкой выравнивания. По завершении вычисления можно выбрать и вывести по меньшей мере одно выравнивание рида с лучшей оценкой выравнивания. В различных случаях модуль выравнивания также реализует динамический алгоритм программирования при вычислении выравнивания, например, один или более алгоритмов Смита-Ватермана, с линейной или аффинной оценкой гэпов, алгоритм выравнивания с гэпами и/или алгоритм выравнивания без гэпов. В конкретных случаях вычисление выравнивания может включать в себя сначала выполнение выравнивания без гэпов с каждым референсным сегментом и на основе результатов выравнивания без гэпов выбор референсных сегментов для дальнейшего выполнения с ними выравниваний с гэпами.

[0016] В различных вариантах реализации может быть предусмотрен модуль определения вариантов для выполнения улучшенных функций определения вариантов, которые, когда они реализованы в одной или обоих программной и/или аппаратной

конфигурациях, обеспечивают превосходную скорость обработки, более хорошую точность результата обработки и улучшенную общую эффективность, чем способы, устройства и системы, известные в настоящее время в данной области техники. А именно, в соответствии с одним аспектом предложены улучшенные способы выполнения операций определения вариантов в программном обеспечении и/или аппаратном обеспечении, например для выполнения одной или более операций HMM на данных генетической последовательности. Согласно другому аспекту предложены новые устройства, включающие в себя интегральную схему для выполнения таких улучшенных операций определения вариантов, причем по меньшей мере часть операции определения вариантов реализована в аппаратном обеспечении.

[0017] Соответственно, в различных случаях способы, описанные в настоящем документе, могут включать в себя картирование, с помощью первого подмножества жестко смонтированных и/или квантовых цифровых логических схем, множества ридов на один или более сегментов указанных одной или более генетических референсных последовательностей. Кроме того, способы могут включать в себя получение доступа интегральными и/или квантовыми схемами, например, посредством одного или более из множества физических электрических межсоединений, к одному или более картированным ридам и/или одной или более генетическим референсным последовательностям из связанной с ними памяти или кэша; и выравнивание с помощью второго подмножества жестко смонтированных и/или квантовых цифровых логических схем множества картированных ридов на одном или более сегментах указанных одной или более генетических референсных последовательностей.

[0018] В различных вариантах реализации способ может дополнительно включать в себя получение доступа интегральной /или квантовой схемой, например, с помощью одного или более из множества физических электрических межсоединений, к выровненному множеству ридов из связанной с ними памяти или кэша. В таком случае способ может включать в себя сортировку с помощью третьего подмножества жестко смонтированных и/или квантовых цифровых логических схем выровненного множества ридов в соответствии с их позициями в одной или более генетических референсных последовательностей. В определенных случаях способ может также включать в себя вывод, например, с помощью одного или более из множества физических электрических межсоединений интегральной и/или квантовой схемы, результирующих данных картирования, и/или выравнивания, и/или сортировки, например, где результирующие данные содержат позиции картированного, и/или выровненного, и/или сортированного множества ридов.

[0019] В некоторых случаях способ может дополнительно включать в себя использование полученных результирующих данных, например, с помощью еще одного подмножества жестко смонтированных и/или квантовых цифровых логических схем, в целях определения того, как картированные, выровненные и/или сортированные данные, полученные из секвенированного генетического образца субъекта, отличается от референсной последовательности, чтобы создать файл определения вариантов, описывающий генетические различия между двумя образцами. Соответственно, в различных вариантах реализации способ может также включать в себя получение доступа интегральной /или квантовой схемой, например, с помощью одного или более из множества физических электрических межсоединений, к картированному, и/или выровненному, и/или сортированному множеству ридов из связанной с ними памяти или кэша. В таком случае способ может включать в себя выполнение функции определения вариантов, например, операции HMM или операции парной HMM, на

ридах, к которым получен доступ, с помощью третьего или четвертого подмножества жестко смонтированных и/или квантовых цифровых логических цепей для создания файла определения вариантов, подробно описывающего, как картированные, выровненные и/или сортированные риды отличаются от одной или более референсных последовательностей, например гаплотипа.

[0020] Соответственно, согласно определенным аспектам изобретения в настоящем документе предложено компактное аппаратное обеспечение, например, на основе микросхемы, или квантовая ускоренная платформа для выполнения вторичного и/или третичного анализа на генетических данных и/или данных геномного секвенирования. В частности, предложены платформа или конвейер из жестко смонтированных и/или квантовых цифровых логических схем, которые специально выполнены с возможностью выполнения вторичного и/или третичного генетического анализа, например на секвенированных генетических данных или полученных из них геномных данных. В частности, может быть предусмотрен набор жестко смонтированных и/или квантовых логических схем, которые могут быть выполнены в виде набора движков обработки, такого где движки обработки могут присутствовать в предварительно сконфигурированной, и/или жестко смонтированной, и/или квантовой конфигурации на платформе обработки по настоящему изобретению, и могут быть специально выполнены с возможностью осуществления вторичных операций картирования, и/или выравнивания, и/или определения вариантов, относящихся к генетическому анализу на данных ДНК и/или РНК, и/или могут быть специально выполнены с возможностью осуществления другой третичной обработки на данных результатов.

[0021] В конкретных случаях настоящие устройства, системы и способы их использования оптимизированы таким образом, чтобы при выполнении одного или более протоколов геномики и/или биоинформатики вторичной и/или третичной обработки обеспечивать улучшение скорости обработки, которая на порядки величины быстрее стандартных конвейеров вторичной обработки, реализованных в программном обеспечении. Кроме того, конвейеры и/или их компоненты, которые приведены в настоящем документе, обеспечивают повышенную чувствительность и точность на широком диапазоне наборов данных, полученных из последовательности данных, в целях геномной и биоинформационной обработки. В различных вариантах реализации одна или более из этих операций могут быть выполнены интегральной схемой, которая является частью или выполнена в виде центрального процессорного устройства общего назначения, и/или графического процессорного устройства, и/или квантового процессорного устройства.

[0022] Например, геномика и биоинформатика являются областями, связанными с применением информационной технологии и компьютерной науки к сфере генетики и/или молекулярной биологии. В частности, методы биоинформатики могут быть применены к обработке и анализу различных генетических и/или геномных данных, например, от индивида, для определения качественной и количественной информации об этих данных, которая может быть использована различными практикующими медицинскими специалистами при разработке профилактических, терапевтических и/или диагностических способов предотвращения, лечения, уменьшения интенсивности и/или по меньшей мере выявления болезненных состояний и/или их возможности и, таким образом, улучшения безопасности, качества и эффективности здравоохранения на индивидуальном уровне. Следовательно, поскольку области геномики и биоинформатики сконцентрированы на развитии персонализированного здравоохранения, они стимулируют индивидуализированное здравоохранение, которое

является упреждающим, а не реагирующим, и это дает нуждающемуся в лечении индивиду возможность большего вовлечения в поддержание собственного здоровья. Преимущество использования технологий генетики, геномики и/или биотехнологии, описанных в настоящем документе, состоит в том, что качественный и/или количественный анализ молекулярно-биологических (например, генетических) данных может быть выполнен на широком диапазоне наборов образцов при значительно более высоком показателе скорости и зачастую более точно, ускоряя тем самым появление системы персонализированного здравоохранения. В частности, в различных вариантах реализации относящиеся к геномике и/или биоинформатике задачи могут формировать геномный конвейер, который включает в себя один или более из конвейера микроматричного анализа, конвейера анализа генома, например, полногеномного анализа, конвейера анализа генотипирования, конвейера анализа экзома, конвейера анализа микробиома, конвейера анализа генотипирования, включая совместное генотипирование, конвейеров анализа вариантов, включая структурные варианты, соматические варианты, и GATK, а также конвейер анализа секвенирования РНК и конвейеры других генетических анализов.

[0023] Соответственно, для использования этих преимуществ существуют улучшенные и более точные программные реализации для осуществления одного или ряда таких основанных на биоинформатике аналитических методов, например для развертывания с помощью ЦПУ общего назначения и/или ГПУ, и/или они могут быть реализованы в одной или более квантовых схем квантовой платформы обработки. Однако, для способов и систем на основе программного обеспечения традиционной конфигурации, как правило, характерно, что они трудоемкие, требуют много времени для выполнения на таких процессорах общего назначения и подвержены ошибкам. Поэтому реализуемые в данном документе системы биоинформатики, которые могут выполнять эти алгоритмы, например, реализованные в программном обеспечении, с помощью ЦПУ и/или ГПУ квантового процессорного устройства с меньшими затратами труда и/или интенсивностью обработки при более высоком проценте точности, будут полезны.

[0024] Такие реализации разработаны и представлены в настоящем документе, например, где геномные и/или биоинформационные анализы осуществляются оптимизированным программным обеспечением, исполняемым на ЦПУ, и/или ГПУ, и/или квантовом компьютере в системе, которая использует данные генетической последовательности, полученные с помощью процессоров и/или интегральных схем по данному изобретению. Кроме того, необходимо отметить, что стоимость анализа, хранения и совместного использования этих необработанных цифровых данных намного превышает стоимость их создания. Соответственно, в настоящем документе предложены также способы хранения и/или извлечения «точно в срок», которые оптимизируют хранение подобных данных таким образом, что вместо того, чтобы тратиться на коллективное хранение таких данных, применяют быстрое повторное формирование данных. Следовательно, способы формирования, анализа и хранения данных «точно в срок» или «JIT», представленные в настоящем документе, устраняют основное узкое место, которое является давно назревшим, но не устраненным препятствием между постоянно растущим формированием и хранением данных и реальной возможностью проникновения в суть медицинских проблем на их основе.

[0025] Поэтому в настоящем документе представлены системы, устройства и способы для реализации протоколов геномики и/или биоинформатики или их части, таких как для выполнения одной или более функций анализа геномных данных, например, в одном или обоих из интегральной схемы, такой как аппаратная платформа обработки,

и процессора общего назначения, такого как для выполнения одной или более биоаналитических операций в программном обеспечении и/или прошивке. Например, как указано далее в настоящем документе, в различных реализациях предложена интегральная схема и/или квантовая схема для ускорения одного или более процессов на платформе первичной, вторичной и/или третичной обработки. В различных случаях интегральная схема может быть использована при выполнении относящихся к генетической аналитике задач, таких как картирование, определение вариантов, сжатие, распаковка и т.п., ускоренным образом, и поэтому интегральная схема может включать в себя аппаратно ускоренную конфигурацию. Кроме того, в различных случаях может быть предусмотрена интегральная и/или квантовая схема, такая как схема, являющаяся частью процессора, который выполнен с возможностью осуществления одного или более протоколов геномики и/или биоинформатики на сформированных картированных и/или выровненных данных и/или данных с определенными вариантами.

[0026] В частности, в первом варианте реализации первая интегральная схема может быть образована из FPGA, ASIC и/или sASIC, которые соединены с материнской платой или иным образом прикреплены к ней и сконфигурированы или, в случае FPGA, могут быть запрограммированы с помощью прошивки, которую нужно сконфигурировать как набор жестко смонтированных цифровых логических схем, которые выполнены с возможностью осуществления по меньшей мере первого набора функций анализа последовательности в конвейере геномного анализа, например, где интегральная схема выполнена, как описано в настоящем документе выше, с возможностью включения в себя одной или более цифровых логических схем, которые устроены в виде набора движков обработки, выполненных с возможностью осуществления одного или более этапов в операции картирования, выравнивания и/или определения вариантов на генетических данных для создания данных результатов анализа последовательности. Первая интегральная схема может также включать в себя выход, например, сформированный из множества физических электрических межсоединений, такой как для передачи результирующих данных из процедур картирования, и/или выравнивания, и/или других процедур в память.

[0027] Кроме того, может быть включена вторая интегральная и/или квантовая схема, соединенная с материнской платой или иным образом прикрепленная к ней и обменивающаяся данными с памятью посредством интерфейса связи. Вторая интегральная и/или квантовая схема может быть образована как центральное процессорное устройство (ЦПУ) или графическое процессорное устройство (ГПУ), или квантовое процессорное устройство (КПУ), которое выполнено с возможностью приема результирующих данных картированной, и/или выровненной, и/или подвергнутой определению вариантов последовательности, и может быть выполнена с возможностью реагирования на один или более программных алгоритмов, которые выполнены с возможностью выдачи ЦПУ или ГПУ инструкции на выполнение одной или более геномных и/или биоинформационных функций конвейера геномного анализа на результирующих данных анализа картированной, выровненной или подвергнутой определению вариантов последовательности. А именно, относящиеся к геномике и/или биоинформатике задачи могут формировать конвейер геномного анализа, который включает в себя один или более из конвейера микроматричного анализа, конвейера анализа генома, например, полногеномного анализа, конвейера анализа генотипирования, конвейера анализа экзома, конвейера анализа микробиома, конвейеров анализов генотипирования, включая совместное генотипирование, конвейеров анализа вариантов, включая структурные варианты, соматические варианты, и GATK, а также

конвейер анализа секвенирования РНК и конвейеры других генетических анализов.

[0028] Например, в одном варианте реализации ЦПУ, и/или ГПУ, и/или КПУ второй интегральной схемы может содержать программное обеспечение, которое выполнено с возможностью организации конвейера анализа генома для осуществления конвейера полногеномного анализа, такого как конвейер полногеномного анализа, который включает в себя один или более из анализа вариации по всему геному, анализа ДНК по всему экзому, анализа РНК по всему транскриптому, функционального анализа генов, функционального анализа белков, анализа связывания белков, количественного генного анализа и/или анализа сборки белков. В определенных случаях конвейер полногеномного анализа может быть выполнен в целях одного или более из анализа родства, анализа личного анамнеза, диагностики заболеваний, поиска новых лекарственных средств и/или профилирования белков. В конкретном случае конвейер полногеномного анализа осуществляют в целях анализа онкологии. В различных случаях результаты этих данных могут быть сделаны доступными, например, глобально, по всей системе.

[0029] В различных случаях ЦПУ, и/или ГПУ, и/или квантовое процессорное устройство (КПУ) второй интегральной и/или квантовой схемы может содержать программное обеспечение, которое выполнено с возможностью организации конвейера анализа генома для осуществления анализа генотипирования, такого как анализ генотипирования, включающий в себя совместное генотипирование. Например, анализ совместного генотипирования может быть выполнен с помощью вычисления баесовской вероятности, например, вычисления байесовской вероятности, которое дает в результате абсолютную вероятность того, что данный определенный генотип является истинным генотипом. В других случаях программное обеспечение может быть выполнено с возможностью осуществления анализа метагенома для создания результирующих данных метагенома, которые могут быть, в свою очередь, использованы при выполнении анализа микробиома.

[0030] В определенных случаях первая и/или вторая интегральные схемы и/или память могут быть заключены в плату расширения, такую как плата межсоединения периферийных компонентов (PCI). Например, в различных вариантах реализации одна или более из интегральных схем могут быть одной или более микросхемами, соединенными с платой PCIe или иным образом связанными с материнской платой. В различных случаях интегральные и/или квантовые схемы и/или микросхемы могут быть компонентом в секвенаторе, или компьютере, или сервере, таком как часть фермы серверов. В конкретных вариантах реализации интегральные и/или квантовые схемы, и/или платы расширения, и/или компьютеры, и/или серверы могут быть выполнены с возможностью доступа через Интернет, например облако.

[0031] Кроме того, в некоторых случаях память может быть энергозависимой оперативной памятью (ОЗУ), например памятью с прямым доступом (DRAM). В частности, в различных вариантах реализации память может включать в себя по меньшей мере две памяти, такие как первая память, которая представляет собой HMEM, например для хранения данных референсной последовательности гаплотипа, и вторая память, которая представляет собой RMEM, например для хранения рида данных геномной последовательности. В конкретных случаях каждая из двух памятей может содержать порт записи и/или порт считывания, например, где каждый из порта записи и порта считывания имеет доступ к отдельному тактовому генератору. Кроме того, каждая из двух памятей может содержать триггерную конфигурацию для хранения множества данных генетической последовательности и/или результатов обработки.

[0032] Соответственно, согласно другому аспекту система может быть выполнена с возможностью совместного использования ресурсов памяти среди ее составных частей, например, при выполнении некоторых вычислительных задач посредством программного обеспечения, такого как выполняемое с помощью ЦПУ, и/или ГПУ, и/или квантовой платформы обработки, и/или выполнения других вычислительных задач посредством прошивки, например посредством аппаратного обеспечения связанной интегральной схемы, такой как FPGA, ASIC и/или sASIC. Этого можно достичь различными путями, такими как прямое слабое или жесткое связывание между ЦПУ/ГПУ/КПУ и матрицей FPGA, например микросхемой или платой PCIe. Такие конфигурации могут быть особенно полезны при распределении операций, относящихся к обработке больших структур данных, связанных с геномными и/или биоинформационными анализами и предназначенных для использования и доступа, как ЦПУ/ГПУ/КПУ, так и связанной интегральной схемой. В частности, в различных вариантах реализации при обработке данных посредством геномного конвейера, как описано в настоящем документе, например, для ускорения общей функции обработки, синхронизации и эффективности, на данных могут выполняться ряд различных операций, причем эти операции могут вовлекать как программные, так и аппаратные компоненты обработки.

[0033] Следовательно, может потребоваться совместное использование данных или иной обмен ими между программными компонентами, выполняющимися на ЦПУ, и/или ГПУ, и/или КПУ, и/или аппаратным компонентом, встроенным в микросхему, например матрицу FPGA. Соответственно, один или более из различных этапов в конвейере геномной и/или биоинформационной обработки или его части, могут быть выполнены одним устройством, например ЦПУ/ГПУ/КПУ, а один или более из различных этапов может быть выполнен жестко смонтированным устройством, например матрицей FPGA. В таком случае ЦПУ/ГПУ/КПУ и/или матрица FPGA могут быть связаны с возможностью обмена данными таким образом, чтобы обеспечивать возможность эффективной передачи таких данных, причем связывание может включать совместное использование ресурсов памяти. Чтобы достичь такого распределения задач и совместного использования информации для выполнения подобных задач различные ЦПУ/ГПУ/КПУ могут быть слабо или жестко связаны друг с другом и/или аппаратными устройствами, например FPGA или другим набором микросхем, например посредством межсоединения быстрого доступа.

[0034] В частности, в различных вариантах реализации предложена платформа геномного анализа. Например, платформа может включать в себя материнскую плату, память, множество интегральных и/или квантовых схем, таких как формирующие один или более из ЦПУ/ГПУ/КПУ, модуль картирования, модуль выравнивания, модуль сортировки и/или модуль определения вариантов. А именно, в конкретных вариантах реализации платформа может включать в себя первую интегральную и/или квантовую схему, такую как интегральная схема, формирующая центральное процессорное устройство (ЦПУ) или графическое процессорное устройство (ГПУ), или квантовая схема, формирующая квантовый процессор, которая реагирует на одну или более программ или другие алгоритмы, которые выполнены с возможностью выдачи ЦПУ/ГПУ/КПУ инструкции на выполнение одного или более наборов функций геномного анализа, как описано в настоящем документе, например, где ЦПУ/ГПУ/КПУ включает в себя первый набор физических электронных межсоединений для соединения с материнской платой. В различных случаях память может быть тоже присоединена к материнской плате и может быть также электронно соединена с ЦПУ/ГПУ/КПУ,

например, посредством по меньшей мере части первого набора физических электронных межсоединений. В таких случаях память может быть выполнена с возможностью хранения множества ридов геномных данных, и/или по меньшей мере одной или более генетических референсных последовательностей, и/или индекса одной или более генетических референсных последовательностей.

[0035] Кроме того, платформа может включать в себя одну или более других интегральных схем, например, где каждая из других интегральных схем формирует программируемую пользователем вентильную матрицу (FPGA), имеющую второй набор физических электронных межсоединений для соединения с ЦПУ/ГПУ/КПУ и памятью, например посредством протокола двухточечного соединения. В таком случае, например, когда интегральная схема представляет собой матрицу FPGA, матрица FPGA может быть выполнена с возможностью программирования с помощью прошивки для конфигурирования набора жестко смонтированных цифровых логических схем, которые взаимно соединены множеством физических межсоединений, для выполнения второго набора функций геномного анализа, например, картирования, выравнивания, определения вариантов и т.д. В частности, жестко смонтированные цифровые логические схемы матрицы FPGA могут быть выполнены в виде набора движков обработки для осуществления одного или более предварительно сконфигурированных этапов в конвейере анализа последовательностей геномного анализа, например, где наборы движков обработки включают в себя один или более из модулей картирования, и/или выравнивания, и/или определения вариантов, причем модули могут быть сформированы из отдельных или одних и тех же подмножеств движков обработки.

[0036] Как было указано, система может быть выполнена с возможностью включения в себя одного или более движков обработки, и в различных вариантах реализации включенный движок обработки может сам может быть выполнен с возможностью определения одной или более вероятностей перехода для последовательности нуклеотидов рида геномной последовательности, переходящей из одного состояния в другое, например, из состояния совпадения в состояние индел, или из состояния совпадения в состояние делеции и/или обратно, например из состояния инсерции или делеции обратно в состояние совпадения. Кроме того, в различных случаях интегральная схема может иметь конвейерную конфигурацию и/или может включать в себя второе, и/или третье, и/или четвертое подмножество жестко смонтированных цифровых логических схем, например включающих второй набор движков обработки, где второй набор движков обработки содержит модуль картирования, выполненный с возможностью картирования рида геномной последовательности на референсную последовательность гаплотипа для создания картированного рида. Также может быть включено третье подмножество жестко смонтированных цифровых логических схем, например, где третий набор движков обработки содержит модуль выравнивания, выполненный с возможностью выравнивания картированного рида на одну или более позиций в референсной последовательности гаплотипа. Также может быть дополнительно включено четвертое подмножество жестко смонтированных цифровых логических схем, например где четвертый набор движков обработки содержит модуль сортировки, выполненный с возможностью сортировки картированного и/или выровненного рида по его относительным положениям в хромосоме. Как и выше, в различных подобных случаях модуль картирования, и/или модуль выравнивания, и/или модуль сортировки, например, вместе с модулем определения вариантов, может быть физически встроен в плату расширения. И в определенных вариантах реализации плата расширения может быть физически объединена с генетическим секвенатором,

таким как секвенатор нового поколения и т.п.

[0037] Соответственно, согласно одному аспекту предложено устройство для выполнения одного или более этапов конвейера анализа последовательности, например на генетических данных, где генетические данные содержат одну или более генетических референсных последовательностей, таких как последовательность гаплотипа или гипотетического гаплотипа, индекс одной или более генетических референсных последовательностей и/или множество ридов, например генетических и/или геномных данных, причем данные могут храниться в одном или более совместно используемых запоминающих устройств, и/или могут быть обработанными с помощью ресурса распределенной обработки, такого как ЦПУ/ГПУ/КПУ и/или FGPA, которые связаны, например, жестко или слабо, вместе. Таким образом, в различных случаях устройство может содержать интегральную схему, где интегральная схема может содержать одну или более (например, набор) жестко смонтированных цифровых логических схем, причем набор жестко смонтированных цифровых логических схем может быть взаимно соединен, например посредством одного или множества физических электрических межсоединений.

[0038] Соответственно, система может быть выполнена с возможностью включения в себя интегральной схемы, сформированной из одной или более цифровых логических схем, которые взаимно соединены множеством физических электрических межсоединений, причем одно или более из множества физических электрических межсоединений имеет один или более интерфейсов памяти и/или кэша для доступа интегральной схемы к памяти и/или хранящимся в ней данным и для извлечения их, например, с обеспечением когерентности кэша между ЦПУ/ГПУ/КПУ и связанной микросхемой, например матрицей FPGA. В различных случаях цифровые логические схемы могут включать в себя первое подмножество цифровых логических схема, например, где первое подмножество цифровых логических схем может быть выполнено в виде первого набора движков обработки, причем движок обработки может быть выполнен с возможностью доступа к данным, хранящимся в кэше и/или непосредственно или опосредованно соединенной памяти. Например, первый набор движков обработки может быть выполнен с возможностью осуществления одного или более этапов анализа картирования, и/или выравнивания, и/или сортировки, как описано выше, и/или анализа HMM на риде данных геномной последовательности и данных последовательности гаплотипа.

[0039] Более конкретно, первый набор движков обработки может содержать модуль HMM, например, в первой конфигурации подмножества цифровых логических схем, который выполнен с возможностью доступа в памяти, например, через интерфейс памяти, по меньшей мере к некоторым из последовательности нуклеотидов в риде данных геномной последовательности и данных последовательности гаплотипа, и может быть также выполнен с возможностью выполнения анализа HMM на по меньшей мере некоторых из последовательности нуклеотидов в данных последовательности гаплотипа для создания результирующих данных HMM. Кроме того, одно или более из множества физических электрических межсоединений может включать в себя выход из интегральной цепи, например для передачи результирующих данных HMM из модуля HMM, например, в ЦПУ/ГПУ/КПУ или на сервер или кластер серверов.

[0040] Соответственно, согласно одному аспекту предложен способ осуществления конвейера анализа последовательностей, например на данных генетической последовательности. Генетические данные могут содержать одну или более генетических референсных последовательностей или последовательностей гаплотипа, один или более

индексов одной или более генетических последовательностей и/или последовательностей гаплотипа, и/или множество ридов геномных данных. Способ может включать в себя одно или более из приема, получения доступа, картирования, выравнивания, сортировки различных итераций данных генетической последовательности и/или использования их результатов в способе создания одного или более файлов определения вариантов. Например, в определенных вариантах реализации способ может включать в себя прием на вход в интегральную схему из электронного источника данных одного или более из множества ридов геномных данных, где каждый рид геномных данных может содержать последовательность нуклеотидов.

[0041] В различных случаях интегральная схема может быть образована из множества жестко смонтированных цифровых логических схем, которые могут быть выполнены в виде одного или более движков обработки. В таком случае движок обработки может быть сформирован из подмножества жестко смонтированных цифровых логических схем, которые могут быть в монтажной конфигурации. В таком случае движок обработки может быть выполнен с возможностью осуществления одного или более предварительно сконфигурированных этапов, например для реализации одного или более из приема, получения доступа, картирования, выравнивания, сортировки различных итераций данных генетической последовательности и/или использования их результатов в способе создания одного или более файлов определения вариантов. В некоторых вариантах реализации предложенные цифровые логические схемы могут быть взаимно связаны, например с помощью множества физических электрических межсоединений, которые могут включат в себя вход.

[0042] Способ может также включать в себя получение доступа интегральной схемой по одному или более из множества физических электрических соединений из памяти к данным для выполнения одной из операций, подробно описанных в настоящем документе. В различных случаях интегральная схема может быть частью набора микросхем, например, встроенной или иным образом входящей как часть в матрицу FPGA, ASIC или структурированную ASIC, а память может быть напрямую или опосредованно соединена с одной или обеими микросхемами и/или связанными с ними ЦПУ/ГПУ/КПУ. Например, память может быть множеством памятей, одна из которых соединена с микросхемой и ЦПУ/ГПУ/КПУ, который сам связан с микросхемой, например слабо.

[0043] В других случаях память может быть одинарной памятью, которая может быть соединена с ЦПУ/ГПУ/КПУ, который сам жестко связан с матрицей FPGA посредством сильного межсоединения обработки или межсоединения быстрого доступа, например, QPI, и тем самым доступна для матрицы FPGA, например, с обеспечением когерентности кэша. Соответственно, интегральная схема может быть напрямую или опосредованно соединена с памятью, чтобы получать доступ к данным, имеющим отношение к выполнению функций, представленных в настоящем документе, например для получения доступа к одному или более из множества ридов, одной или более генетических референсных или теоретических референсных последовательностей и/или индексу одной или более генетических референсных последовательностей, например при выполнении операции картирования.

[0044] Поэтому в различных случаях реализации различных аспектов изобретения могут включать в себя, без ограничений: устройства, системы и способы, включающие один или более признаков, которые подробно описаны в настоящем документе, а также изделия, которые содержат материально воплощенный машиночитаемый носитель информации, выполненный с возможностью инициирования осуществления одной или

более машинами (например, компьютерами и т.д.) операций, описанных в настоящем документе. Аналогичным образом также описаны компьютерные системы, которые могут содержать один или более процессоров и/или одну или более памятей, соединенных с одним или более процессорами. Соответственно, компьютеризованные способы, соответствующие одной или более реализациям текущего объекта изобретения, могут быть осуществлены одним или более процессорами данных, находящимися в одной вычислительной системе или множестве вычислительных системах, содержащих множество компьютеров, например в вычислительном или супервычислительном банке.

[0045] Такие множественные вычислительные системы могут быть соединены и могут обмениваться данными и/или командами либо другими инструкциями и т.п. посредством одного или более соединений, включая, без ограничений, соединение по сети (например, Интернет, беспроводная глобальная сеть, локальная сеть, глобальная сеть, проводная сеть, физическое электрическое межсоединение и т.п.), через прямое соединение между одной или более множественными вычислительными сетями и т.д. Память, которая может быть машиночитаемым носителем информации, может содержать, кодировать, хранить и т.п. одну или более программ, которые вызывают выполнение одним или более процессорами одной или более операций, связанных с одним или более алгоритмами, описанными в настоящем документе.

[0046] Подробные сведения об одном или более вариантах объекта изобретения, описанного в настоящем документе, изложены на прилагаемых чертежах и в описании, приведенном ниже. Другие признаки и преимущества объекта изобретения, описанного в настоящем документе, будут очевидны из описания и чертежей и формулы изобретения. Хотя определенные признаки раскрытого в настоящее время объекта изобретения описаны в целях иллюстрации в связи с программной системой ресурсов предприятия или иным коммерческим программным решением или архитектурой, совершенно ясно, что такие признаки не предназначены для ограничения. Формула изобретения, которая следует за данным описанием, предназначена для определения объема защищенного объекта изобретения.

## КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

[0047] Прилагаемые чертежи, которые включены в данное описание изобретения и являются его частью, показывают определенные аспекты объекта изобретения, раскрытого в настоящем документе, и, вместе с описанием, помогают объяснять некоторые из принципов, связанных с описанными реализациями.

[0048] На ФИГ. 1A изображена платформа секвенирования с множеством генетических образцов на ней, а также изображено множество примеров плиток, как и трехмерное представление секвенированных ридов.

[0049] На ФИГ. 1B изображено представление проточной кюветы с представленными различными полосами.

[0050] На ФИГ. 1C изображен нижний угол платформы проточной кюветы, приведенной на ФИГ. 1B, показывающий группу секвенированных ридов.

[0051] На ФИГ. 1D изображен виртуальный массив результатов секвенирования, выполненного на ридах, приведенных на ФИГ. 1 и 2, где риды указаны в выходном столбце в порядке столбцов.

[0052] На ФИГ. 1E изображен способ, с помощью которого можно осуществлять транспозицию итоговых ридов из столбцов в порядке столбцов в ряды в порядке рядов.

[0053] На ФИГ. 1F изображена транспозиция итоговых ридов из столбцов в порядке столбцов в ряд в порядке рядов.

[0054] На ФИГ. 1G изображены компоненты системы для выполнения транспозиции.

[0055] На ФИГ. 1H изображен порядок транспозиции.

[0056] На ФИГ. 1I изображена архитектура для электронной транспозиции секвенированных данных.

[0057] На ФИГ. 2 изображена основанная на 3 состояниях модель HMM, иллюстрирующая вероятности транспозиции перехода из одного состояния в другое.

[0058] На ФИГ. 3A изображено высокоуровневое представление интегральной схемы по изобретению, включающей в себя структуру интерфейса HMM.

[0059] На ФИГ. 3B изображена интегральная схема, приведенная на ФИГ. 3A, более подробно показывающая особенности кластера HMM.

[0060] На ФИГ. 4 изображен обзор потока относящихся к HMM данных через систему, включая программные и аппаратные соединения.

[0061] На ФИГ. 5 изображен пример соединений манжеты кластера HMM.

[0062] На ФИГ. 6 изображено высокоуровневое представление основных функциональных блоков в пределах примера аппаратного ускорителя HMM.

[0063] На ФИГ. 7 изображен пример структуры матрицы HMM и потока аппаратной обработки.

[0064] На ФИГ. 8 изображен увеличенный вид части ФИГ. 2, показывающий поток данных и зависимости между соседними ячейками при вычислениях состояния M, I и D HMM в пределах матрицы.

[0065] На ФИГ. 9 изображены примеры вычислений, полезных для обновлений состояний M, I, D.

[0066] На ФИГ. 10 изображены схемы обновления состояний M, I и D, в том числе влияние упрощенных допущений, показанных на ФИГ. 9, которые относятся к вероятностям перехода, и влияние совместного использования некоторых ресурсов сумматора M, I, D с заключительными операциями суммирования.

[0067] На ФИГ. 11 изображены подробности вычисления состояний M, I, D логарифмической области.

[0068] На ФИГ. 12 изображена диаграмма переходов состояний HMM, показывающая взаимосвязь между GOP, GCP и вероятностями перехода.

[0069] На ФИГ. 13 изображена схема формирования вероятностей перехода HMM и значений Prior для поддержки общей диаграммы переходов состояний, приведенной на ФИГ. 12.

[0070] На ФИГ. 14 изображена упрощенная диаграмма переходов состояний HMM, показывающая взаимосвязь между GOP, GCP и вероятностями перехода.

[0071] На ФИГ. 15 изображена схема формирования вероятностей перехода HMM и значений Prior для поддержки упрощенной диаграммы перехода состояний.

[0072] На ФИГ. 16 изображен пример теоретической матрицы HMM и показано, как можно пройти такую матриц HMM.

[0073] На ФИГ. 17A представлен способ выполнения процедуры предварительной обработки совместного обнаружения в множестве областей.

[0074] На ФИГ. 17B представлен пример способа вычисления матрицы связности, такой как в процедуре предварительной обработки, показанной на ФИГ. 17A.

[0075] На ФИГ. 18A изображен пример события между двумя гомологичными секвенированными областями в скоплении ридов.

[0076] На ФИГ. 18B изображены построенные риды, приведенные на ФИГ. 18A, обозначающие разницу между двумя последовательностями.

[0077] На ФИГ. 18C изображены различные пузыри графа де Брейна, которые могут быть использованы при выполнении ускоренной операции определения вариантов.

[0078] На ФИГ. 18D изображено представление функции обрезания дерева, как описано в настоящем документе.

[0079] На ФИГ. 18E изображен один из пузырей, приведенных на ФИГ. 18C.

[0080] На ФИГ. 19 приведено графическое представление примера скопления, относящегося к матрице, показанной на ФИГ. 17.

[0081] На ФИГ. 20 приведена матрица обработки для выполнения процедуры предварительной обработки, представленной на ФИГ. 17A и B.

[0082] На ФИГ. 21 приведен пример образования пузыря в графе де Брейна в соответствии со способами, показанными на ФИГ. 20.

[0083] На ФИГ. 22 приведен пример пути варианта через иллюстративный графа де Брейна.

[0084] На ФИГ. 23 приведено графическое представление примера функции сортировки.

[0085] На ФИГ. 24 приведен другой пример матрицы обработки для обрезанной процедуры совместного обнаружения в множестве областей.

[0086] На ФИГ. 25 показано совместное скопление парных ридов для двух областей.

[0087] На ФИГ. 26 приведена таблица вероятности в соответствии с описанием в настоящем документе.

[0088] На ФИГ. 27 приведен другой пример матрицы обработки для процедуры совместного обнаружения в множестве областей.

[0089] На ФИГ. 28 представлен выбор решений-кандидатов для совместного скопления, показанной на ФИГ. 25.

[0090] На ФИГ. 29 представлен другой выбор решений-кандидатов для скопления, показанного на ФИГ. 28, после того, как выполнена функция обрезания.

[0091] На ФИГ. 30 представленные окончательные решения-кандидаты, показаны на ФИГ. 28, и их соответствующие вероятности после выполнения функции MRJD.

[0092] На ФИГ. 31 показаны кривые РХП для MRJD и обычного детектора.

[0093] На ФИГ. 32 показаны те же самые результаты, что и на ФИГ. 3, отображаемые в виде функции от подобия последовательности референсов.

[0094] На ФИГ. 33A изображен пример архитектуры, иллюстрирующий слабое связывание между ЦПУ и матрицей FPGA по данному изобретению.

[0095] На ФИГ. 33B изображен пример архитектуры, иллюстрирующий жесткое связывание между ЦПУ и матрицей FPGA по данному изобретению.

[0096] На ФИГ. 34A изображено слабое связывание между ЦПУ и FPGA по данному изобретению.

[0097] На ФИГ. 34B изображен альтернативный вариант реализации прямого связывания между ЦПУ и матрицей FPGA, приведенного на ФИГ. 34A.

[0098] На ФИГ. 35 изображен альтернативный вариант реализации корпуса объединенных ЦПУ и матрицы FPFA, где эти два устройства совместно используют общую память и/или кэш.

[0099] На ФИГ. 36 показано ядро ЦПУ, совместно использующих одну или более памятей и/или кэшей, причем ЦПУ выполнены с возможностью обмена данными с одной или более матрицами FPGA, которые могут также включать в себя совместно используемую или общую память или кэши.

[00100] На ФИГ. 37 показан пример способа передачи данных по всей системе.

[00101] На ФИГ. 38 более подробно изображен вариант реализации, приведенный на ФИГ. 36.

[00102] На ФИГ. 39 изображен пример способа обработки одного или более заданий

системы по настоящему изобретению.

[00103] На ФИГ. 40A изображена блок-схема для геномной инфраструктуры местной или облачной геномной обработки и анализа.

[00104] На ФИГ. 40B изображена блок-схема облачной платформы геномной обработки для выполнения BioIT-анализа, описанного в настоящем документе.

[00105] На ФИГ. 40C изображена блок-схема примера конвейера геномной обработки и анализа.

[00106] На ФИГ. 40D изображена блок-схема примера конвейера геномной обработки и анализа.

[00107] На ФИГ. 41A изображена блок-схема местной и/или облачной вычислительной функции, приведенной на ФИГ. 40A, для геномной инфраструктуры локальной или облачной геномной обработки и анализа.

[00108] На ФИГ. 41B более подробно изображена блок-схема, приведенная на ФИГ. 41A, в части, касающейся вычислительной функции для геномной инфраструктуры локальной или облачной геномной обработки и анализа.

[00109] На ФИГ. 41C более подробно изображена блок-схема, приведенная на ФИГ. 40, в части, касающейся сторонней функции аналитики для геномной инфраструктуры локальной или облачной геномной обработки и анализа.

[00110] На ФИГ. 42A изображена блок-схема, иллюстрирующая конфигурацию гибридного облака.

[00111] На ФИГ. 42B более подробно изображена блок-схема, приведенная на ФИГ. 42A, которая иллюстрирует конфигурацию гибридного облака.

[00112] На ФИГ. 42C более подробно изображена блок-схема, приведенная на ФИГ. 42A, которая иллюстрирует конфигурацию гибридного облака.

[00113] На ФИГ. 43A изображена блок-схема, иллюстрирующая конвейер первичной, вторичной и/или третичной обработки, который представлен в настоящем документе.

[00114] На ФИГ. 43B приведен пример анализа эпигенетики третичной обработки для выполнения с помощью способов и устройств системы, описанной в настоящем документе.

[00115] На ФИГ. 43C приведен пример анализа метилирования третичной обработки для выполнения с помощью способов и устройств системы, описанной в настоящем документе.

[00116] На ФИГ. 43D приведен пример анализа структурных вариантов третичной обработки для выполнения с помощью способов и устройств системы, описанной в настоящем документе.

[00117] На ФИГ. 43E приведен пример анализа третичной когортной обработки для выполнения с помощью способов и устройств системы, описанной в настоящем документе.

[00118] На ФИГ. 43F приведен пример анализа третичной обработки совместного генотипирования для выполнения с помощью способов и устройств системы, описанной в настоящем документе.

[00119] На ФИГ. 44 изображена блок-схема конвейера анализа по изобретению.

[00120] На ФИГ. 45 приведена блок-схема аппаратной архитектуры процессора в соответствии с реализацией изобретения.

[00121] На ФИГ. 46 приведена блок-схема аппаратной архитектуры процессора в соответствии с другой реализацией.

[00122] На ФИГ. 47 приведена блок-схема аппаратной архитектуры процессора в соответствии с еще одной реализацией.

[00123] На ФИГ. 48 показан конвейер анализа генетических последовательностей.

[00124] На ФИГ. 49 показаны этапы обработки с использованием аппаратной платформы анализа генетической последовательности.

[00125] На ФИГ. 50A показано устройство в соответствии с реализацией изобретения.

[00126] На ФИГ. 50B показано другое устройство в соответствии с реализацией изобретения.

[00127] На ФИГ. 51 показана система геномной обработки в соответствии с реализацией.

## ОСУЩЕСТВЛЕНИЕ ИЗОБРЕТЕНИЯ

[00128] Как в краткой форме изложено выше, настоящее изобретение относится к устройствам, системам и способам их использования при выполнении одного или более протоколов геномики и/или биоинформатики, таких как протокол картирования, выравнивания, сортировки и/или определения вариантов, на данных, формируемых посредством процедуры первичной обработки, например на данных генетической последовательности. Например, согласно различным аспектам устройства, системы и способы, предложенные в настоящем документе, выполнены с возможностью осуществления протоколов вторичного анализа генетических данных, таких как данные, сформированные секвенированием РНК и/или ДНК, например, с помощью секвенатора нового поколения (СНП). В конкретных вариантах реализации предусмотрены один или более конвейеров вторичной обработки для обработки данных генетической последовательности, например, где конвейеры и/или их отдельные элементы могут быть реализованы в программном обеспечении, аппаратном обеспечении или их сочетании с распределением и/или оптимизацией, чтобы обеспечивать превосходную чувствительность и улучшенную точность в более широком диапазоне данных, полученных из последовательности, по сравнению с доступным в настоящее время в данной области техники. Кроме того, как кратко изложено выше, настоящее изобретение относится к устройствам, системам и способам их использования при выполнении одного или более третичных протоколов геномики и/или биоинформатики, таких как протокол микроматричного анализа, протокол анализа генома, например, полногеномного анализа, протокол анализа генотипирования, протокол анализа экзома, протокол анализа эпигенома, протокол анализа метагенома, протокол анализа микробиома, протокол анализа генотипирования, включая совместное генотипирование, протоколы анализа вариантов, включая структурные варианты, соматические варианты, и GATK, а также протоколы секвенирования РНК и другие протоколы генетических анализов, например, на картированных, выровненных и/или других данных генетической последовательности, например использование одного или более файлов определения вариантов.

[00129] Соответственно, в настоящем документе предложены технологии анализа с использованием программно и/или аппаратно, например, на основе микросхемы, ускоренной платформы для выполнения вторичного и/или третичного анализа данных секвенирования ДНК/РНК. Более конкретно, платформа, или конвейер, движков обработки, например в программно реализованной и/или жестко смонтированной конфигурации, которая специально выполнена с возможностью осуществления вторичного генетического анализа, например, картирования, выравнивания, сортировки и/или определения вариантов; и/или может быть специально выполнена с возможностью осуществления третичного генетического анализа, такого как микроматричный анализ, анализ генома, например, полногеномный анализ, анализ генотипирования, анализ экзома, анализ эпигенома, анализ метагенома, анализ микробиома, анализ

генотипирования, включая анализ совместного генотипирования, анализ вариантов, включая анализ структурных вариантов и анализ GATK, а также анализ секвенирования РНК и другие генетические анализы, например, применительно к данным генетического секвенирования, которые могли быть сформированы в оптимизированном формате, обеспечивающем улучшение скорости обработки, которая на порядки величины быстрее стандартных конвейеров, реализованных исключительно в известном программном обеспечении. Кроме того, представленные в настоящем документе конвейеры обеспечивают более хорошую чувствительность и точность на широком диапазоне наборов данных, полученных из последовательности, такой как последовательности, полученные из нуклеиновых кислот или белков.

[00130] Как указано выше, в различных случаях цель обработки в биоинформатике состоит в определении отдельных геномов и/или белковых последовательностей людей, причем эти определения могут быть использованы в протоколах исследования генов, а также для профилактических и/или терапевтических режимов для улучшения жизнедеятельности каждого отдельного человека и человечества в целом. Кроме того, знание генома и/или комбинации белков индивида может быть использовано, например, в поиске новых лекарственных средств и/или испытаниях FDA для лучшего прогнозирования с учетом специфики, какие лекарственные средства, если они существуют, вероятно, будут воздействовать на индивида, и/или какие лекарственные средствам, вероятно, будут иметь вредные побочные эффекты, например путем анализа генома и/или полученного из него белкового профиля индивида и сравнения их с прогнозируемой биологической реакцией на введение такого лекарственного средства.

[00131] Такая обработка в биоинформатике обычно предполагает три четко определенные, но, как правило, отдельные фазы обработки информации. Первая фаза, называемая первичной обработкой, включает в себя секвенирование ДНК/РНК, где получают ДНК и/или РНК субъекта и подвергают различным обработкам, с помощью которых генетический код субъекта преобразуют в машиночитаемый цифровой код, например в файл FASTQ. Вторая фаза, называемая вторичной обработкой, включает в себя использование сформированного цифрового генетического кода субъекта для определения генетического строения субъекта, например определения геномной нуклеотидной последовательности субъекта. И третья фаза, называемая третичной обработкой, включает в себя выполнение одного или более анализов генетического строения субъекта для определения из него информации, полезной в терапевтических целях.

[00132] Соответственно, после того, как генетический код субъекта секвенирован, например, с помощью секвенатора нового поколения, чтобы получить машиночитаемое цифровое представление генетического кода субъекта, например, в формате файла FASTQ и/или BCL, возможно, будет полезна дальнейшая обработка кодированных в цифровом виде данных генетической последовательности, полученной из секвенатора и/или протокола секвенирования, например, путем применения вторичной обработки к представленным в цифровом виде данным. Эта вторичная обработка, например, может быть использована для картирования, и/или выравнивания, и/или сборки иным образом полногеномного и/или белкового профиля индивида, например, когда определяют полное генетическое строение, где последовательно определяют все до единого нуклеотиды всех без исключения хромосом так, чтобы идентифицировать состав всего генома индивида. При такой обработке геном индивида может быть собран, например, путем сравнения с референсным геномом, таким как референсный стандарт, например, с одним или более геномами, полученными из проекта генома человека и

т.п., для определения того, как генетическое строение индивида отличается от генетического состава контрольных индивидов. Этот процесс обычно называют определением вариантов. Поскольку отличия между ДНК одного человека с другим встречаются 1 раз на 1000 пар оснований, такой процесс определения вариантов может быть весьма трудоемким и времязатратным, требующим множества этапов, которые, возможно, потребуется выполнять один за другим и/или одновременно, например, в конвейерном режиме, чтобы проанализировать геномные данные субъекта и определить, как эта генетическая последовательность отличается от данного референса.

[00133] При выполнении конвейера вторичного анализа, такого как для формирования файла определения вариантов для данной исследуемой последовательности отдельного субъекта, от субъекта может быть получен генетический образец, например образец ДНК, РНК, белка. Затем ДНК/РНК субъекта может быть секвенрирована, например, с помощью секвенатора нового поколения (СНП) и/или технологии «секвенатор на микросхеме», например, на этапе первичной обработки, чтобы получить множество сегментов последовательности считывания («ридов»), охватывающее полностью или частично геном индивида, например с избыточностью. Конечный продукт, сформированный с помощью устройства для секвенирования, может представлять собой коллекцию коротких последовательностей, например ридов, которые представляют небольшие сегменты генома субъекта, например, короткие генетические последовательности, представляющие полный геном индивида. Как было указано, информация, представленная этими ридами, может быть файлом изображения или файлом в цифровом формате, таком как FASTQ, BCL или другой аналогичный файловый формат.

[00134] В частности, в типичном протоколе вторичной обработки генетическое строение субъекта собирают путем сравнения с референсным геномом. Это сравнение включает в себя реконструкцию генома индивида из миллионов и миллионов коротких последовательностей рида и/или сравнение всего ДНК индивида с примером модели последовательности ДНК, В типичном протоколе вторичной обработки из секвенатора принимают изображение, файл FASTQ и/или BCL, содержащие необработанные секвенированные данные рида. Чтобы сравнить геном субъекта со стандартным референсным геномом, необходимо определить, где каждое из этих ридов картируется на референсный геном, например, как каждый из них выравнивается относительно другого, и/или как каждый рид может быть также отсортирован по порядку хромосом, чтобы определить, в какой позиции находится каждый рид, и какой хромосоме он принадлежит. Одна или более из этих функций могут предшествовать выполнению функция определения вариантов на полноразмерной последовательности, например после сборки. А именно, после того, как определено, какой части генома принадлежит каждый рид, можно определить генетическую полноразмерную последовательность, а затем можно оценить различия между генетическим кодом субъекта и генетическим референсным кодом.

[00135] Например, основанная на референсе сборка в типичном протоколе сборки вторичной обработки включает в себя сравнение секвенированной геномной ДНК/РНК субъекта с секвенированной геномной ДНК/РНК одного или более стандартов, например, известных референсных последовательностей. В качестве помощи для ускорения этих процессов разработаны различные алгоритмы картирования, выравнивания, сортировки и/или определения вариантов. Поэтому данные алгоритмы могут включать в себя некоторый вариант одного или более из: картирования, выравнивания и/или сортировки миллионов ридов, полученных из изображения (файла

FASTQ и/или BCL), которые переданы секвенатором, для определения местоположения каждого конкретного рида на каждой хромосоме. Следует отметить, что эти процессы могут быть реализованы в программном обеспечении или аппаратном обеспечении, например, с помощью способов и/или устройств, описанных в патентах США №№ 9,014,989 и 9,235,680, права на которые принадлежат компании Edico Genome Corporation, и которые полностью включены в настоящий документ путем ссылки. Часто общей особенностью функционирования этих различных алгоритмов и/или аппаратных реализаций является использование ими индекса и/или массива для ускорения их функции обработки.

[00136] Например, что касается картирования, большое количество секвенированных ридов (например, все) могут быть обработаны для определения возможных местоположений в референсном геноме, на который могли бы быть выровнены эти риды. Один из методов, который может быть использован в этих целях, заключается в прямом сравнении рида с референсным геномом, чтобы найти все позиции совпадения. Другой метод состоит в использовании массива префиксов или суффиксов или построении дерева префиксов или суффиксов с целью картирования ридов на различные позиции в референсном геноме. Типичным алгоритмом, полезным при выполнении такой функции, является преобразование Барроуза-Уилера, которое используют для картирования ридов на референс с помощью формулы сжатия, которая сжимает повторяющиеся последовательности данных.

[00137] Еще один метод заключается в использовании хэш-таблицы, например, когда выбранное подмножество ридов, k-мер выбранной длины «k», например, затравку, помещают в хэш-таблицу в качестве ключей, а референсную последовательность разбивают на части, равные по длине k-меру, и эти части и их местоположения вставляют с помощью алгоритма в хэш-таблицу в те места таблицы, на которые они отображаются в соответствии с функцией хэширования. Типичным алгоритмом для выполнения этой функции является «BLAST», Basic Local Alignment Search Tool. Такие программы на основе хэш-таблицы сравнивают исследуемые нуклеотидные или белковые последовательности с одной или более баз данных стандартных референсных последовательностей и вычисляют статистическую значимость совпадений. Подобным образом можно определить вероятное местоположение любого данного рида относительно референсного генома. Эти алгоритмы полезны, поскольку они требуют меньше памяти, преобразований, таблиц перекодировки (LUT) и, следовательно, требуют меньше вычислительных ресурсов и времени при выполнении своих функций, чем было бы в ином случае, например, если бы геном субъекта собирали путем прямого сравнения, например без использования этих алгоритмов.

[00138] Кроме того, может быть выполнена функция выравнивания для определения всех возможных местоположений картирования данного рида на геном, например в тех случаях, когда рид можно картировать на множество позиций в геноме, которые в действительности являются местоположением, из которого он был фактически получен, например путем секвенирования с этого места с помощью исходного протокола секвенирования. Эту функцию можно выполнить на ряде ридов (например, картированных ридов) генома и можно получить строку упорядоченных нуклеотидных оснований, представляющую частично или полностью геномную последовательность ДНК/РНК субъекта. Наряду с упорядоченной генетической последовательностью каждому нуклеотиду в данной позиции можно присвоить оценку, представляющую для любой данной нуклеотидной позиции вероятность того, что нуклеотид, например, «A», «C», «G», «T» (или «U»), предполагаемый в этой позиции, действительно является

нуклеотидом, который принадлежит этой назначенной позиции. В число типичных алгоритмов для выполнения функция выравнивания входят алгоритмы Нидлмана-Вунша и Смита-Ватермана. В любом случае эти алгоритмы выполняют выравнивания последовательностей между строкой исследуемой геномной последовательности субъекта и строкой референсной геномной последовательности, тем самым вместо сравнения полногеномных последовательностей друг с другом сравнивают выбранные сегменты возможных длин.

[00139] После того, как ридам назначены позиции, например, относительно референсного генома, что может включать в себя определение принадлежности рида конкретной хромосоме и/или его смещения от начала этой хромосомы, риды можно отсортировать по позиции. Это может позволить в последующих анализах использовать преимущества процедур с избыточной выборкой, описанных в настоящем документе. Все риды, которые перекрывают данную позицию в геноме, будут рядом друг с другом после сортировки и могут быть организованы в скопление (pileup) и без труда исследованы, чтобы определить, согласуются ли большинство из них с референсным значением или нет. Если нет, вариант можно отметить флагом.

[00140] Например, в различных вариантах реализации способы по настоящему изобретению могут включать в себя формирование файла определения вариантов (VCF), идентифицирующего один или более (например, все) генетические варианты у индивида, ДНК/РНК которого секвенировали, например, в соответствии с одним или более референсных геномов. Например, после того, как фактический геном образца известен и сравнен с референсным геномом, между этим двумя геномами можно определить вариации и составить список всех вариаций/отклонений между референсными геномами и геномом образца, например можно создать файл определения вариантов. В частности, согласно одному аспекту можно сформировать файл определения вариантов, содержащий все вариации генетической последовательности субъекта относительно референсных последовательностей.

[00141] Как указано выше, такие вариации между двумя генетическими последовательностями могут быть обусловлены рядом причин. Следовательно, чтобы сформировать такой файл, геном субъекта необходимо секвенировать и снова построить, прежде чем определять его варианты. Однако существуют несколько проблем, которые могут возникнуть при попытке формирования такой сборки. Например, возможны проблемы с химией, секвенатором и/или человеческими ошибками, которые происходят в процессе секвенирования. Кроме того, возможны генетические артефакты, которые делают такую реконструкцию проблематичной. Например, типичной проблемой при выполнении таких сборок является то, что иногда имеются огромные части генома, которые повторяют сами себя, такие как длинные секции генома, которые включают в себя одни и те же строки нуклеотидов. Следовательно, так как любая генетическая последовательность уникальна не везде, возможны трудности с определением того, где в геноме картируется и выравнивается идентифицированный рид. Кроме того, возможен однонуклеотидный полиморфизм (ОНП), например там, где одно основание в генетической последовательности субъекта было заменено на другое; возможны более обширные замены множества нуклеотидов; возможны инсерция или делеция, например когда одно или множество оснований добавлены в генетическую последовательность субъекта или удалены из нее; и/или возможен структурный вариант, например такой, который вызван скрещиванием двух ножек хромосом, и/или возможно просто смещение, приводящее к сдвигу в последовательности.

[00142] Соответственно, для вариации существую две возможности. Во-первых,

существует действительная вариация в данном конкретном месте, например, когда геном человека в конкретном месте действительно отличается от референса, например, имеется естественная вариация, обусловленная ОНП (заменой одного основания), инсерция или делеция (длиной в один или более нуклеотидов), и/или имеется структурный вариант, например, когда материал ДНК из одной хромосомы перекрещивает другую хромосому или ножку, или когда определенная область дважды встречается в ДНК. Или же вариация может быть вызвана наличием проблемы в данных рида, из-за химии или машины, секвенатора или выравнивателя, или иной человеческой ошибки. Способы, описанные в настоящем документе, могут быть использованы таким образом, чтобы компенсировать эти типы ошибок и, в частности, чтобы отличать ошибки в вариации, обусловленные химией, машинной или человеком, от реальных вариаций в секвенированном геноме. Точнее говоря, системы, устройства и способы их использования, описанные в настоящем документе, разработаны таким образом, чтобы четко различать эти два различных типа вариаций и, следовательно, лучше обеспечивать точность любых сформированных файлов вариантов, чтобы правильно выявлять истинные варианты.

[00143] Поэтому в конкретных вариантах реализации предложена платформа технологий для выполнения генетических анализов, где платформа может включать в себя выполнение одной или более из функций: картирования, выравнивания, сортировки, локального повторного выравнивания, маркировки дубликатов, перекалибровки оценки качества основания, определения вариантов, сжатия и/или распаковки. Например, в соответствии с различными аспектами может быть предусмотрен конвейер, который включает в себя выполнение одной или более аналитических функций, как описано в настоящем документе, на геномной последовательности одного или более индивидов, например, на данных, полученных в файле изображения и/или в цифровом файле формата FASTQ или BCL из автоматизированного секвенатора. Типичный конвейер, подлежащий выполнению, может включать в себя один или более секвернирований генетического материала, например, часть или весь геном, одного или более субъектов, причем генетический материал может содержать ДНК, оцДНК, РНК, рРНК, тРНК и т.п. и/или, в некоторых случаях, генетический материал может представлять кодируемые или некодируемые области, такие как экзомы и/или эписомы ДНК. Конвейер может включать в себя одно или более из выполнения процедуры обработки изображения, операции определения оснований и/или исправления ошибки, например, в оцифрованных генетических данных, и/или может включать в себя одно или более из выполнения функции картирования, выравнивания и/или сортировки на генетических данных. В определенных случаях конвейер может включать в себя выполнение одного или более из повторного выравнивания, удаления дубликатов, перекалибровки оценки качества основания, редукции и/или сжатия и/или распаковки на оцифрованных генетических данных. В определенных случаях конвейер может включать в себя выполнение операции определения вариантов, такой как скрытая марковская модель, на генетических данных.

[00144] Соответственно, в определенных случаях реализация одной или более из этих функций платформы предназначена для выполнения одного или более из определения и/или реконструкции консенсусной геномной последовательности субъекта, сравнения геномной последовательности субъекта с референсной последовательностью, например, референсной или модельной генетической последовательностью, определения того, каким образом геномная ДНК или РНК субъекта отличается от референсной, например, определения вариантов, и/или для выполнения третичного анализа на геномной последовательности субъекта, например, для анализа вариации по всему геному,

функционального анализа генов, функционального анализа белков, например, анализа связывания белков, а также для различных анализов диагностической, и/или профилактической, и/или терапевтической оценки.

[00145] Как указано выше, в соответствии с одним аспектом одна или более из этих функций платформы, например, функций картирования, выравнивания, сортировки, повторного выравнивания, маркировки дубликатов, перекалибровки оценки качества основания, определения вариантов, сжатия и/или распаковки, выполнены с возможностью реализации в программном обеспечении. В соответствии с некоторыми аспектами одна или более из этих функций платформы, например, функций картирования, выравнивания, сортировки, локального повторного выравнивания, маркировки дубликатов, перекалибровки оценки качества основания, распаковки, определения вариантов, сжатия и/или распаковки, выполнены с возможностью реализации в аппаратном обеспечении, например прошивке. В соответствии с определенными аспектами эти технологии генетического анализа могут использовать улучшенные алгоритмы, которые могут быть реализованы программным обеспечением, которое выполняется с менее интенсивной обработкой, и/или с меньшими временными затратами, и/или более высоким процентом точности, например, аппаратно реализованные функциональные возможности более быстрые, требуют менее интенсивной обработки и более точные.

[00146] Например, в определенных вариантах реализации предусмотрены улучшенные алгоритмы для выполнения такой первичной, вторичной и/или третичной обработки, как описано в настоящем документе. Улучшенные алгоритмы направлены на более эффективное и/или более точное выполнение одной или более из функций картирования, выравнивания, сортировки и/или определения вариантов, например, на файле изображения и/или цифровом представлении данных последовательности ДНК/РНК, полученном с платформы секвенирования, например, в формате файла FASTQ или BCL, полученном из автоматизированного секвенатора, такого как один из описанных выше. В конкретных вариантах реализации улучшенные алгоритмы могут быть направлены на более эффективное и/или более точное выполнение одной или более из функций локального повторного выравнивания, маркировки дубликатов, перекалибровки оценки качества оснований, определения вариантов, сжатия и/или распаковки. Кроме того, как более подробно описано ниже в настоящем документе, в определенных вариантах реализации эти технологии генетического анализа могут использовать один или более алгоритмов, таких как улучшенные алгоритмы, которые могут быть реализованы с помощью одного или более из программного обеспечения и/или аппаратного обеспечения, которые выполняются с менее интенсивной обработкой, и/или с меньшими временными затратами, и/или более высоким процентом точности, чем различные традиционные программные реализации для выполнения того же самого. В различных случаях предусмотрены улучшенные алгоритмы для реализации на квантовой платформе обработки.

[00147] Поэтому в соответствии с различными аспектами в настоящем документе предложены системы, устройства и способы для реализации протоколов биоинформатики, таких как для выполнения одной или более функций анализа генетических данных, таких как геномные данные, например, посредством одного или более оптимизированных алгоритмов и/или одной или более интегральных и/или квантовых схем, например на одной или более аппаратных платформах обработки. В одном случае предложены системы и способы для реализации одного или более алгоритмов, например, в программном обеспечении, и/или прошивке, и/или с помощью

квантовой схемы обработки, для выполнения одного или более этапов анализа геномных данных в протоколах биоинформатики, например, когда этапы могут включать в себя выполнение одного или более из: картирования, выравнивания, сортировки, локального повторного выравнивания, маркировки дубликатов, перекалибровки оценки качества основания, определения вариантов, сжатия и/или распаковки; и могут также включать в себя один или более этапов на платформе третичной обработки. Соответственно, в определенных случаях в настоящем документе предложены способы, включающие в себя алгоритмы программной, программно-аппаратной, аппаратной и/или квантовой обработки для выполнения способов, где способы включают в себя выполнение алгоритма, такого как алгоритм для реализации одной или более функций генетического анализа, таких как картирование, выравнивание, сортировка, повторное выравнивание, маркировка дубликатов, перекалибровка оценки качества основания, определение вариантов, сжатие, распаковка, и/или одного или более протоколов третичной обработки, где алгоритм, например, включающий в себя прошивку, оптимизирован в соответствии со способом, которым он должен быть реализован.

[00148] В частности, когда алгоритм должен быть реализован в программном решении, алгоритм и/или обслуживающие его процессы, оптимизированы таким образом, чтобы они работали быстрее и/или с более высокой точностью при выполнении этой средой. Аналогичным образом, когда функции алгоритма должны быть реализованы в аппаратном решении, например прошивке, аппаратное обеспечение разработано для выполнения этих функций и/или обслуживающих их процессов оптимальным образом, чтобы работать быстрее и/или с более высокой точностью при выполнении этой средой. Кроме того, когда алгоритм должен быть реализован в решении квантовой обработки, алгоритм и/или обслуживающие его процессы, оптимизированы таким образом, чтобы они работали быстрее и/или с более высокой точностью при выполнении этой средой. Эти способы, например, могут быть использованы, например, в итеративных картировании, выравнивании, сортировке, определении вариантов и/или процедуре третичной обработки. В другом случае предложены системы и способы для реализации функций одного или более алгоритмов для выполнения одного или более этапов анализа геномных данных в протоколе биоинформатики, как указано в настоящем документе, причем функции реализуются на аппаратном и/или квантовом ускорителе, который может быть связна или не соединен с одним или более процессорами общего назначения, и/или суперкомпьютерами, и/или квантовыми компьютерами.

[00149] Точнее говоря, в некоторых случаях предложены способы и/или оборудование для реализации этих способов с целью выполнения вторичной аналитики на данных, имеющих отношение к генетическому составу субъекта. В одном случае аналитика, подлежащая выполнению, может включать в себя основанную на референсе реконструкцию генома субъекта. Например, основанное на референсе картирование включает в себя использование референсного генома, которым может быть сформирован в результате секвенирования генома одного или множества индивидов, или он может быть объединением принадлежащих различным людям ДНК/РНК, которые объединены таким образом, чтобы создать прототипный стандартный референсный геном, с которым можно сравнить генетический материал, например ДНК/РНК, любого индивида, например, для определения и реконструкции генетической последовательности индивида и/или для определения разницы между их генетическим строением и этим стандартным референсом, например, для определения вариантов.

[00150] В частности, причина выполнения вторичного анализа на секвенированной ДНК/РНК субъекта состоит в том, чтобы определить, как ДНК/РНК субъекта

отличается от ДНК/РНК эталона, чтобы определить одно, множество или все отличия нуклеотидной последовательности субъекта от нуклеотидной последовательности референса. Например, отличия между генетическими последовательностями любых двух случайно выбранных людей встречаются 1 раз на примерно 1000 пар оснований, что с учетом свыше 3 миллиардов пар оснований в полном геноме составляет вариацию из до 3000000 отличающихся пар оснований на человека. Определение этих отличий может быть полезным, например, в протоколе третичного анализа, например, для прогнозирования возможности возникновения болезненного состояния, например, вследствие генетического нарушения, и/или вероятности успеха профилактического или терапевтического воздействия, например, на основе того, каким ожидается взаимодействие профилактики или терапии с ДНК субъекта или формируемыми при этом белками. В различных случаях может оказаться полезным выполнение реконструкции генома субъекта как de novo, так и на основе референса, чтобы подкрепить результаты одной результатами другой, и чтобы улучшить точность протокола определения вариантов, если требуется.

[00151] Соответственно, согласно одному аспекту в различных вариантах реализации после того, как реконструирован геном субъекта и/или сформирован файл VCF, такие данные могут быть затем подвергнуты третичной обработке с целью их интерпретации, например, для определения того, что эти данные означают с точки зрения выявления болезней, которым может подвергнуться или не подвергнуться этот человек, и/или для определения терапий или изменений стиля жизни, которыми, возможно, пожелает воспользоваться данный субъект, чтобы устранить и/или предотвратить болезненное состояние. Например, генетическая последовательность субъекта и/или его файл определения вариантов могут быть проанализированы для определения клинически уместных генетических маркеров, которые указывают на наличие или возможность болезненного состояния и/или эффективность, с которой может воздействовать на субъекта рекомендуемый терапевтический или профилактически режим. Затем эти данные могут быть использованы для обеспечения субъекту одного или более терапевтических или профилактических режимов с тем, чтобы улучшить качество жизни субъекта, например, вылечить и/или предотвратить болезненное состояние.

[00152] В частности, после того, как определены одна или более генетических вариаций индивида, такая информация файла определения вариантов может быть использована для подготовки полезной с медицинской точки зрения информации, которая, в свою очередь, может быть использована для определения, например, с помощью известных моделей статистического анализа, относящихся к здоровью данных и/или полезной с медицинской точки зрения информации, например, в диагностических целях, например, для диагностирования болезни или ее возможности, клинической интерпретации (например, поиска маркеров, которые представляют вариант болезни), определения того, следует ли включить субъекта в различные клинических испытания или исключить из них, и других таких целей. Более конкретно, в различных случаях сформированные данные результатов обработки методами геномики и/или биоинформатики могут быть использованы при выполнении одного или более третичных протоколов геномики и/или биоинформатики, таких как протокол микроматричного анализа, протокол анализа генома, например, полногеномного анализа, протокол анализа генотипирования, протокол анализа экзома, протокол анализа эпигенома, протокол анализа метагенома, протокол анализа микробиома, протокол анализа генотипирования, включая совместное генотипирование, протоколы анализов вариантов, включая структурные варианты, соматические варианты, и GATK, а также

протоколы секвенирования РНК и другие протоколы генетических анализов.

[00153] Поскольку существует конечное число болезненных состояний, которые вызываются генетическими нарушениями, при третичной обработке варианты определенного типа, например, известные тем, что они связаны с возникновением болезненных состояний, могут быть уточнены, например, путем определения того, включены ли один или более генетических маркеров болезни в файл определения вариантов субъекта. Поэтому в различных случаях способы, описанные в настоящем документе, могут включать в себя анализ, например, сканирование, VCF и/или сформированной последовательности на предмет известных связанных с болезнями вариантов последовательности, например, присутствующих по этой причине в базе данных геномных маркеров, чтобы выявить наличие генетического маркера в VCF и/или сформированной последовательности, и при наличии такового проверять присутствие или возможность генетически обусловленного болезненного состояния. Так как существуют огромное количество известных генетических вариаций и огромное количество индивидов, страдающих от болезней, вызываемых такими вариациями, в некоторых вариантах реализации способы, описанные в настоящем документе, могут охватывать формирование одной или более баз данных, связывающих секвенированные данные полного генома и/или связанного с ними файла определения вариантов, например, от одного или множества индивидов, с болезненным состоянием, и/или поиск в сформированных базах данных с целью определения того, имеет ли конкретный субъект генетический состав, который предрасполагает его к наличию такого болезненного состояния. Такой поиск может включать в себя сравнение одного полного генома с одним или более другими, или фрагмента генома, такого как фрагмент, содержащий только вариации, с одним или более фрагментами одного или более других геномов, например, в базе данных референсных геномов или их фрагментов.

[00154] Поэтому в различных случаях конвейер по данному изобретению может содержать один или более модулей, где модули выполнены с возможностью осуществления одной или более функций, таких как обработка изображения или определение оснований, и/или операция исправления ошибок, и/или картирование, и/или выравнивание, например, выравнивание с гэпами и без гэпов, и/или функция сортировки генетических данных, например, секвенированных генетических данных. И в различных случаях конвейер может содержать один или более модулей, где модули выполнены с возможностью осуществления одного или более из локального повторного выравнивания, удаления дубликатов, перекалибровки оценки качества основания, определения вариантов, например, HMM, редукции и/или распаковки на генетических данных. Кроме того, конвейер может содержать один или более модулей, где модули выполнены с возможностью осуществления протокола третичного анализа, такого как протоколы микроматричного анализа, протоколы анализа генома, например, полногеномного анализа, протоколы анализа генотипирования, протоколы анализ экзома, протоколы анализа эпигенома, протоколы анализа метагенома, протоколы анализа микробиома, протоколы анализа генотипирования, включая протоколы анализа совместного генотипирования, протоколы анализа вариантов, включая совместное генотипирование, протоколы анализов вариантов, включая структурные варианты, соматические варианты, и протоколы GATK, а также протоколы секвенирования РНК и другие протоколы генетических анализов.

[00155] Многие из этих модулей могут выполняться либо программным обеспечением, либо аппаратным обеспечением, локально или удаленно, например, посредством программного обеспечения или аппаратного обеспечения, скажем, на облаке, например,

на удаленном сервере и/или банке серверов, таком как квантовый вычислительный кластер. Кроме того, многие из этих модулей и/или этапов конвейера являются необязательными и/или могут быть расположены в любом логическом порядке и/или полностью опущены. Например, программное обеспечение и/или аппаратное обеспечение, описанные в настоящем документе, могут включать в себя или не включать обработку изображения и/или определение оснований, или алгоритм исправления последовательности, например, когда могут быть опасения, что такие функции могут привести к статистической систематической ошибке. Следовательно, система может включать себя или не включать функцию определения оснований и/или исправления последовательности, соответственно, в зависимости от требуемого уровня точности и/или эффективности. И, как указано выше, одна или более функций конвейера могут быть использованы при формировании геномной последовательности субъекта, например, посредством реконструкции генома на основе референса. Кроме того, в определенных случаях выходными данными из конвейера вторичной обработки может быть файл определения вариантов (VCF, gVCF), указывающий частично или полностью варианты в геноме или его части.

[00156] В частности, после того как ридам назначены позиции относительно референсного генома, что может включать в себя определение того, какой хромосоме принадлежит рид, и/или его смещения от начала этой хромосомы, из них можно удалить дубликаты и/или отсортировать их, например по позиции. Это позволяет в последующих анализах использовать преимущества различных протоколов с избыточной выборкой, описанных в настоящем документе. Все риды, которые перекрывают данную позицию в геноме, могут быть расположены рядом друг с другом после сортировки, и они могут быть накоплены, например чтобы образовать скопление, и без труда исследованы, чтобы определить, согласуются ли большинство из них с референсным значением или нет. Если нет, как указано выше, вариант можно пометить.

[00157] Соответственно, как указано выше в отношении картирования, файл изображения, файл BCL и/или файл FASTQ, полученные из секвенатора, состоят из множества, например, от миллионов до миллиардов или более, ридов, состоящих из коротких строк данных последовательности нуклеотидов, представляющих часть и или весь геном индивида. Например, первый этап в конвейерах вторичного анализа, описанных в настоящем документе, представляет собой прием геномных и/или биоинформационных данных, например, из устройства формирования геномных данных, такого как секвенатор. Как правило, данные, создаваемые секвенатором, например, секвенатором нового поколения, могут быть в формате файла BCL, который в некоторых случаях может быть преобразован в формат файла FASTQ, до или после передачи, например, на платформу вторичной обработки, описанную в настоящем документе. В частности, при секвенировании человеческого генома необходимо идентифицировать ДНК и/или РНК субъекта, основание за основанием, причем результатом такого секвенирования является файл BCL. Файл BCL - это двоичный файл, который содержит определения оснований и оценки качества, сделанные для каждого основания каждой последовательности коллекции последовательностей, которая составляет по меньшей мере часть или полный геном субъекта.

[00158] По традиции сформированный секвенатором файл BCL преобразуют в файл FASTQ, который затем может быть передан на платформу вторичной обработки, такую как описанная в настоящем документе, для дальнейшей обработки, например, для определения из него геномной вариации. Файл FASTQ - это файл в текстовом формате для передачи и хранения как биологической последовательности (например,

последовательности нуклеотидов), так и ее соответствующих оценок качества, где каждая буква последовательности, например A, C, G, T и/или U, и оценка качества может быть закодирована одним символов ASCII для краткости. Соответственно, в данной и других системах в целях дальнейшей обработки используют именно файл FASTQ. Хотя использование файла FASTQ для геномной обработки полезно, преобразование сформированного файла BCL в файл FASTQ, которое реализовано в секвенаторе, занимает много времени и неэффективно. Поэтому, в соответствии с одним аспектом предложены устройства и способы для прямого преобразования файла BCL в файл FASTQ и/или непосредственного ввода таких данных в конвейеры представленной платформы, как описано в настоящем документе.

[00159] Например, в различных вариантах реализации секвенатор нового поколения или изготовленный по технологии «секвенатор на микросхеме» может быть выполнен с возможностью осуществления операции секвенирования на принимаемых генетических данных. Например, как показано на ФИГ. 1A, генетические данные 6a могут быть связаны с платформой 6, предназначенной для вставки в секвенатор нового поколения для итеративного секвенирования, таким образом, что каждая последовательность будет наращиваться путем пошагового добавления одного нуклеотида за другим. А именно, платформа 6 секвенирования может содержать ряд шаблонных последовательностей 6a нуклеотидов субъекта, которые расположены в виде сетки с образованием плиток 6b на платформе 6, шаблонные последовательности 6a которой подлежат секвенированию. Платформа 6 может быть добавлена к проточной кювете 6c секвенатора, который выполнен с возможностью выполнения реакций секвенирования.

[00160] По мере того, как происходят реакции секвенирования, на каждом этапе к платформе 6 проточной кюветы 6c добавляется нуклеотид, имеющий флуоресцентную метку 6d. Если происходит реакция гибридизации, наблюдается флуоресценция и делается снимок, затем изображение обрабатывается, и выполняется соответствующее определение основания. Это повторяется для одного основания за другим до тех пор, пока все шаблонные последовательности, например, полный геном, не будут секвенированы и преобразованы в риды с созданием тем самым данных рида системы. Таким образом, по завершении секвенирования сформированные данные, например риды, необходимо передать с платформы секвенирования в систему вторичной обработки. Например, эти данные изображения, как правило, преобразуют в файл BCL и/или FASTQ, который затем может быть перенесен в систему.

[00161] Однако в различных случаях этот процесс преобразования и/или переноса может быть сделан более эффективным. А именно, в настоящем документе представлены способы и архитектуры для ускоренного преобразования файла BCL в файлы, которые могут быть быстро обработаны в системе вторичной обработки. Точнее говоря, в конкретных случаях вместо передачи необработанных файлов BCL или FASTQ созданные изображения, представляющие каждую плитку операции секвенирования, могут быть перенесены непосредственно в систему и подготовлены для картирования, выравнивания и т.д. Например, плитки могут передаваться в потоковом режиме через соответствующим образом сконфигурированную карту PCIe в ASIC, FPGA или КПУ, где из них могут быть непосредственно выделены данные рида, а риды продвинуты в движки картирования и выравнивания и/или другие движки обработки.

[00162] В частности, что касается переноса данных с плиток, полученных секвенатором, в FPGA/ЦПУ/ГПУ/КПУ, как показано на ФИГ. 1A, платформа 6 секвенирования может быть изображена в виде 3-мерного куба 6c, внутри которого

могут быть сформированы растущие последовательности 6a. По существу, как показано на ФИГ.1B, платформа 6 секвенирования может состоять из 16 полос, 8 в передней части и 8 в задней части, которые могут быть выполнены с возможностью формирования около 96 плиток 6b. Внутри каждой плитки 6b находятся множество шаблонных последовательностей 6a, подлежащих секвенированию с формированием тем самым ридов, каждый из которых представляет последовательность нуклеотидов для данной области генома субъекта, каждый столбец представляет один файл, а в цифровой кодировке представляет 1 байт для каждого файла, по 8 битов на файл, например, где 2 бита представляют найденное основание, а оставшиеся 6 битов представляют оценку качества.

[00163] Более конкретно, что касается секвенирования нового поколения, его обычно выполняют на стеклянных пластинах 6, формирующие проточные кюветы 6c, которые вводят в автоматизированный секвенатор для секвенирования. Как показано на ФИГ. 1B, проточная кювета 6c представляет собой платформу 6, состоящую из 8 вертикальных столбцов и 8 горизонтальных рядов (передних и задних), вместе образующих 16 полос, где каждой полосы достаточно для секвенирования полного генома. ДНК и/или РНК 6a субъекта, подлежащую секвенированию, связывают в отведенных позициях между непроницаемыми для текучей среды пересечениями столбцов и рядов платформы 6 с образованием плиток 6b, где каждая плитка содержит шаблонный генетический материал 6a, подлежащий секвенированию. Следовательно, платформа 6 секвенирования содержит множество шаблонных нуклеотидных последовательностей субъекта, причем последовательности расположены в виде сетки плиток на платформе (см. ФИГ. 1B). Затем генетические данные 6 секвенируют итеративным образом, и при этом каждую последовательность наращивают пошаговым введением одного нуклеотида за другим в проточную кювету, причем каждый этап итеративного наращивания представляет цикл секвенирования.

[00164] Как было указано, после каждого этапа получают изображение, а растущая последовательность, например, изображений, образует основу, с помощью которой формируют файл BCL. Как показано на ФИГ. 1C, риды из процедуры секвенирования могут образовывать кластеры, и именно эти кластеры образуют теоретический 3-мерный куб 6c. Соответственно, внутри этого теоретического 3-мерного куба каждое основание каждой растущей нуклеотидной нити, которую секвенируют, будет иметь измерение x и измерение y. Данные изображения, или плитки 6b, из этого 3-мерного куба 6c могут быть выделены и собраны в двумерную карту, из которой можно сформировать матрицу, как показано на ФИГ. 1D. Эту матрицу формируют из циклов секвенирования, которые представляют горизонтальную ось, и идентификаторов ридов, которые представляют вертикальную ось. Соответственно, как показано на ФИГ. 1C, секвенированные риды образуют кластеры в проточной кювете 6c, причем кластеры могут быть определены с помощью вертикальной и горизонтальной оси, цикл за циклом и основание за основанием, и данные из каждого цикла для каждого рида могут быть вставлены в матрицу, приведенную на ФИГ. 1D, например, в потоковом и/или конвейерном режиме.

[00165] А именно, каждый цикл представляет потенциальный прирост каждого рида в проточной кювете путем добавления одного нуклеотида, который при секвенировании одного или нескольких человеческих геномов может представлять прирост около 1 миллиарда или более ридов на полосу. Прирост каждого рида, например, за счет добавления нуклеотидного основания, идентифицируют путем итеративного получения изображение плиток 6b проточной кюветы 6c между этапами приращения. С помощью этих изображений выполняют определение оснований, определяют оценки качества и

формируют виртуальную матрицу, приведенную на ФИГ. 1D. Соответственно, в матрицу будут введены как определение основания, так и оценка качества, причем каждая плитка из каждого цикла представляет отдельный файл. Необходимо отметить, что при выполнении секвенирования на интегральной схеме считываемые электронные данные могут заменить данные изображения.

[00166] Например, как показано на ФИГ. 1D, сама матрица будет итеративно прирастать по мере получения и обработки изображений, определения оснований и определения оценок для каждого рида, цикл за циклом. Это повторяют для каждого основания в риде для каждой плитки проточной кюветы. Например, кластеры ридов, показанные на ФИГ. 1C, могут быть пронумерованы и введены в матрицу как вертикальная ось. Аналогичным образом номер цикла может быть введен как горизонтальная ось, а затем можно ввести определение основания и оценку качества, чтобы заполнить матрицу по столбцу, ряд за рядом. Соответственно, каждый рид будет представлен некоторым количеством оснований, например, примерно от 100 или 150 до 1000 или более оснований на рид в зависимости от секвенатора, и на каждую плитку может приходиться до 10 миллионов или более ридов. Поэтому, если имеется около 100 плиток, в каждой из которых 10 миллионов ридов, матрица будет содержать около 1 миллиарда ридов, которые нужно организовывать и передавать в потоковом режиме в устройство вторичной обработки.

[00167] Соответственно, такая организация имеет основополагающее значение для быстрой и эффективной обработки данных. Поэтому, в соответствии с одним аспектом в настоящем документе предложены способы для транспортировки данных, представленных виртуальной матрицей секвенирования, таким образом, чтобы данные могли эффективно передаваться в потоковом режиме в конвейеры системы, описанной в настоящем документы. Например, формирование данных секвенирования, которые представлены звездообразным кластером, изображенным на ФИГ. 1C, в значительной степени неорганизованные, что создает трудности с точки зрения обработки данных. В частности, по мере формирования данных с помощью операции секвенирования они организуются в виде одного файла на цикл, а это означает, что по завершении операции секвенирования формируются миллионы и миллионы файлов, которые представлены на ФИГ. 1E данными в столбцах, разграниченными сплошными линиями. Однако в целях вторичной и/или третичной обработки, как описано в настоящем документе, данные файла должны быть реорганизованы в данные рида, разграниченные пунктирными линиями на ФИГ. 1E.

[00168] Более конкретно, чтобы эффективнее передавать в потоковом режиме формируемые секвенатором данные на вторичную обработку данных, следует переставлять представленные виртуальной матрицей данные, например, путем реорганизации данных файла из построения столбец за столбцом плиток в каждом цикле в построение ряд за рядом, идентифицируя основания каждого рида. А именно, структура данных формируемых файлов, образующих матрицу, по мере их создания секвенатором, организуется цикл за циклом, столбец за столбцом. С помощью процессов, описанных в настоящем документе, эти данные могут быть переставлены, например, по существу одновременно, чтобы они были представлены, как показано в виртуальной матрице, рид за ридом, ряд за рядом, где каждый ряд представляет отдельный рид, а каждый рид представлен порядковым номером определений оснований и оценками качества, идентифицирующими таким образом как последовательность для каждого рида, так и его достоверность. Таким образом, при операции перестановки, как описано в настоящем документе, данные в памяти могут быть реорганизованы, например, в

виртуальной матрице, из построения столбец за столбцом, представляющего порядок ввода данных, в построение ряд за рядом, представляющее порядок вывода данных, меняя тем самым порядок данных из вертикальной организации на горизонтальную организацию. Кроме того, хотя этот процесс может быть эффективно реализован в программном обеспечении, он может быть выполнен даже еще эффективнее и быстрее путем реализации в аппаратном обеспечении и/или с помощью квантового процессора.

[00169] Например, в различных случаях этот процесс перестановки может быть ускорен за счет реализации в аппаратном обеспечении. Например, в одной реализации на первом этапе программное обеспечение главного устройства, например, секвенатора, может записывать входные данные в память, связанную с матрицей FPGA, столбец за столбцом, например в порядке ввода. А именно, по мере формирования данных и сохранения их в связанной памяти эти данные могут быть организованы в файлы, цикл за циклом, где данные сохраняются как самостоятельные отдельные файлы. Эти данные могут быть представлены 3-кубом, изображенным на ФИГ. 1A. Эти формируемые данные, которые организованы в столбцы, могут быть затем построены в очередь и/или переданы в потоковом режиме, например, в режиме реального времени, в аппаратное обеспечение, где специализированные движки обработки выстроят в очередь организованные в столбцы данные и переставят эти данные из конфигурации столбец за столбцом в порядке циклов в конфигурацию ряд за рядом в порядке ридов таким образом, как описано выше в настоящем документе, например, путем преобразования 3-мерных данных плиток в 2-мерную матрицу, тем самым данные столбцов могут быть реорганизованы в данные рядов, например рид за ридом. Эти переставленные данные могут быть затем сохранены в памяти в более стратегически важном порядке.

[00170] Например, программное обеспечение главного устройства может быть выполнено с возможностью записи входных данных в память, связанную с микросхемой, например матрицей FPGA, например, по столбцам в порядке ввода, и подобно аппаратному обеспечению может быть выполнено с возможностью построения в очередь данных таким образом, чтобы они считывались в память стратегически важным образом, например, как показано на ФИГ. 1F. А именно, аппаратное обеспечение может включать в себя массив регистров 8a, в которые могут быть распределены файлы циклов и реорганизованы в данные отдельных ридов, например, путем записи одного основания из столбца в регистры, которые организованы в ряды. Точнее говоря, как показано на ФИГ. 1G, аппаратное устройство 1, содержащее движок 8 обработки перестановки, может включать в себя порт 8a DRAM, который может выстраивать в очередь данные, подлежащие перестановке, причем порт выполнен с возможностью функционального соединения с интерфейсом 8b памяти, который связан с множеством регистров и/или внешней памятью 8c, и выполнен с возможностью обработки повышенного количества транзакций за цикл, где выстроенные в очередь данные передаются пакетами.

[00171] В частности, эта перестановка может происходить по одному сегменту данных за раз, например, когда очередь для доступа к памяти организована таким образом, чтобы максимально использовать преимущество скорости передачи DDR. Например, применительно к DRAM это означает, что минимальная длина пакета DDR может, например, составлять 64 байта. Соответственно, доступ к упорядоченным в столбцы данным, хранящимся в главной памяти, может выполняться таким образом, чтобы при каждом обращении к памяти получать столбец общим размером, соответствующим, например, 64 байтам данных. Следовательно, за одно обращение к памяти можно получить доступ к части плитки, например, представляющей соответствующие «64»

цикла или файлов, столбец за столбцом.

[00172] Однако, как показано на ФИГ. 1F, хотя данные в главной памяти доступны в виде данных столбцов, при передаче в аппаратное обеспечение они могут быть загружены в связанные памяти меньшего объема, например, регистры, в другом порядке, и тем самым данные могут быть преобразованы в байты, например, 64 байта, данных рида ряд за рядом, например, в соответствии с минимальной скоростью передачи пакета DDR, чтобы формировать соответствующие «64» единиц или блоков памяти за каждое обращение к памяти. Это показано на примере виртуальной матрицы, изображенной на ФИГ. 1D, где доступ к множеству ридов, например 64, получают поблочно и считывают их в память в виде сегментов, как представлено на ФИГ. 1E, например, когда на каждый регистр или триггер приходится конкретный рид, например, 64 цикла × 64 рида × 8 бит на рид = 32K триггеров. А именно, этого можно достичь всевозможными разными способами в аппаратном обеспечении, например, когда запись входных данных организована в порядке столбцов, а запись выходных данных организована в порядке рядов. Следовательно, в этой конфигурации аппаратное обеспечение может быть выполнено таким образом, чтобы считывать из и/или записывать в «64» различных адреса за цикл.

[00173] Более конкретно, аппаратное обеспечение может быть связано с массивом регистров так, что каждое основание рида направляется и записывается в один регистр (или множество регистров, расположенных в ряд), например, когда каждый блок завершен, заново упорядоченные данные ряда могут быть переданы в память в качестве выходных данных, например, данных FASTQ, строка за строкой. После этого к данным FASTQ могут обращаться один или более дальнейших движков обработки системы вторичной обработки для дальнейшей обработки, например, с помощью движка картирования, выравнивания и/или определения вариантов, как описано в настоящем документе. Необходимо отметить, что, как описано в настоящем документе, перестановку выполняют в небольших блоках, однако, в зависимости от обстоятельств, система может быть также выполнена с возможностью обработки более крупных блоков.

[00174] Как было указано, после того, как файл BCL преобразован в файл FASTQ, как было описано выше, и/или файл BCL или FASTQ иным образом принят платформой вторичной обработки, на принятых данных может быть выполнена операция картирования. Картирование, как правило, включает в себя нанесение ридов на все местоположения в референсном геноме, где имеется совпадение. Например, в зависимости от размера рида, могут быть одно или множество местоположений, где этот рид по существу совпадает с соответствующей последовательностью в референсном геноме. Следовательно, картирование и/или другие функции, описанные в настоящем документе, могут быть выполнены с возможностью определения того, какое из всех возможных местоположений, где одно или более ридов могут совпадать с референсным геномом, действительно является истинным местом, на которые они картируются.

[00175] Например, в различных случаях можно сформировать или иным образом обеспечить индекс референсного генома, чтобы можно было искать риды или части ридов, например, в таблице подстановки (LUT), по ссылке на индекс, тем самым извлекая указатели местоположений в референсе для картирования ридов на референс. Такой индекс референса можно построить в различных формах и обращаться к нему различным образом. В некоторых способах индекс может содержат дерево префиксов и/или суффиксов. В конкретных способах индекс может быть получен из референса с помощью преобразования Барроуза-Уилера. Таким образом, в качестве альтернативы или в

дополнение к использованию дерева префиксов или суффиксов на данных можно выполнить преобразование Барроуза-Уилера. Например, преобразование Барроуза-Уилера можно использовать для сохранения древовидной структуры данных, абстрактно эквивалентной дереву префиксов и/или суффиксов, в компактном формате, например, в пространстве, выделенном для хранения референсного генома. В различных случаях данные хранятся не в древовидной структуре, а, скорее, данные референсной последовательности представлены в линейном списке, который можно скремблировать в другой порядок, чтобы преобразовать его совершенно особенным образом так, чтобы сопутствующий алгоритм позволял осуществлять поиск референса по ссылке на риды образца, чтобы по существу перемещаться по «дереву».

[00176] Кроме того, в различных случаях индекс может содержать одну или более хэш-таблиц, а способы, описанные в настоящем документе, могут включать в себя хэш-функцию, которую можно выполнять на одной или более частях ридов с целью картирования ридов на референс, например, на индекс референса. Например, в качестве альтернативы или в дополнение к использованию одного или обоих из дерева префиксов/суффиксов и/или преобразования Барроуза-Уилера на референсном геноме и данных последовательности субъекта с целью поиска мест, где они картируются друг на друга, другой такой способ включает в себя создание индекса хэш-таблицы и/или выполнение хэш-функции. Индекс хэш-таблицы может быть крупной ссылочной структурой, которую строят из последовательностей референсного генома, и которую потом можно сравнивать с одной или более частями рида для определения того, где они могут совпасть друг с другом. Аналогичным образом индекс хэш-таблицы может быть построен из частей ридов, который можно затем сравнивать с одной или более последовательностями референсного геном и использования тем самым для определения того, где они могут совпасть друг с другом.

[00177] Реализация хэш-таблицы - это быстрый способ выполнения сопоставления с шаблоном. Выполнение каждого поиска занимает почти постоянное количество времени. Такой метод можно противопоставить методу Барруоза-Уилера, который может потребовать множество проб (количество может меняться в зависимости от того, сколько битов требуется для нахождения уникального образца) на исследуемую последовательность, чтобы найти совпадение, или методу двоичного поиска, который требует $\log_2(N)$ проб, где N - количество затравочных образцов в таблице. Кроме того, даже если хэш-функция может разбить референсный геном на сегменты затравок любой данной длины, например, 28 пар оснований, и может затем преобразовать затравки в цифровое, например, двоичное, представление из 56 битов, доступ ко всем 56 битам одновременно или одинаковым образом не требуется. Например, хэш-функция может быть реализована таким образом, чтобы адрес для каждой затравки обозначался числом менее 56 битов, например, от около 18 до около 44 или 46 битов, например, от около 20 до около 40 битов, например, от около 24 до около 36 битов, в том числе от около 28 до около 32 или 30 битов могут быть использованы в качестве начального ключа или адреса для получения доступа к хэш-таблице. Например, в определенных случаях от около 26 до около 29 битов могут использоваться в качестве первичного ключа доступа для хэш-таблицы, и остаются от около 27 до около 30, которые могут использоваться в качестве средства для двойной проверки первого ключа, например, если первый и второй ключи поступают одновременно в одну и ту же ячейку в хэш-таблице, после чего вполне очевидно, что указанное местоположение является тем местом, которому они принадлежат.

[00178] Например, первая часть представленной в цифровом виде затравки, например,

от около 26 до около 32, скажем, около 29 битов, может образовывать первичный ключ доступа и быть хэширована, и ее поиск может быть выполнен на первом этапе. А на втором этапе оставшиеся от около 27 до около 30 битов, например, вторичный ключ доступа, могут быть вставлены в хэш-таблицу, например, в цепочку хэширования, в качестве средства для подтверждения первого прохода. Соответственно, для любой затравки ее первоначальные биты адреса могут быть хэшированы на первом этапе, а биты вторичного адреса могут быть использованы на втором этапе, этапе подтверждения. В таком случае первая часть затравок может быть вставлена в первичное местоположение записи, а вторая часть может быть вставлена в таблицу в местоположении второй цепочки записи. И, как указано выше, в различных случаях эти два разных местоположения записи могут быть позиционно разделены, например записью формата цепочки.

[00179] В конкретных случаях для сравнения референса с ридом или его частями может быть использовано линейное сканирование методом перебора. Однако использование линейного поиска методом перебора для сканирования референсного генома на предмет местоположений, где затравка совпадает, возможно, придется проверить свыше 3 миллиардов местоположений. Такой поиск может быть выполнен в соответствии со способами, описанными в настоящем документе, в программном или аппаратном обеспечении. Тем не менее, при использовании подхода на основе хэширования, который изложен в настоящем документе, каждый поиск затравки может занимать приблизительно постоянное количество времени. Часто местоположение может быть выявлено за несколько, например, одно, обращений к памяти. Однако в случаях, где множество затравок картируют на одно и то же местоположение в таблице, например, они не вполне уникальны, для нахождения текущей искомой затравки могут быть выполнены несколько дополнительных обращений к памяти. Следовательно, даже если может оказаться 30M или более возможных местоположений, в которых рид длиной в 100 нуклеотидов соответствует референсному геному, хэш-таблица и хэш-функция могут быстро определить, где в референсном геноме может появиться этот рид. Поэтому при использовании индекса хэш-таблицы для определения мест, где рид картируется и выравнивается, выполнять поиск полного референсного генома, например, перебором, не требуется.

[00180] Ввиду вышеизложенного, в этих целях можно использовать любую подходящую хэш-функцию, однако в различных случаях хэш-функция, используемая для определения адреса таблицы для каждой затравки, может быть проверкой с использованием циклического избыточного кода (CRC), которая может быть основана на примитивном многочлене 2k-бит, как указано выше. В альтернативном варианте реализации может быть использован сопоставитель тривиальной хэш-функции, например, путем простого отбрасывания некоторых из 2k битов. Однако в различных случаях CRC может быть более сильной хэш-функцией, которая может лучше отделять похожие затравки, избегая при этом переполнения таблицы. Это может оказаться особенно полезным в отсутствии штрафов на скорость при вычислении CRC, как в случае со специализированным программным обеспечением, описанным в настоящем документе. В таких случаях хэшированная запись, заполненная для каждой затравки, может включать в себя позицию референса, где встречается эта затравка, и флаг, указывающий, была ли она обратно комплементирована перед хэшированием.

[00181] Результат, возвращаемый после выполнения функции картирования, может представлять собой список возможных вариантов мест, где одно или более, например, каждый рид картируется на один или более референсных геномов. Например, выходными

данными каждого картированного рида может быть список возможных местоположений, где рид может быть картирован на совпадающую последовательность в референсном геноме. В различных вариантах реализации может выполняться поиск точного совпадения с референсом для по меньшей мере фрагмента, например затравки рида, если не всего рида. Соответственно, в различных случаях не требуется точного совпадения всех частей всех ридов со всеми частями референсного генома.

[00182] Как описано в настоящем документе, все эти операции могут быть выполнены с помощью программного обеспечения или могут быть реализованы аппаратно, например, в интегральной схеме, такой как микросхема, например, как часть печатной платы. Например, функции одного или более алгоритмов могут быть встроены в микросхему, такую как матрица FPGA (программируемая пользователем вентильная матрица) или схема ASIC (специализированная интегральная схема), и могут быть оптимизированы для более эффективной работы за счет реализации их в таком аппаратном обеспечении. Кроме того, одна или более, например, две или все три, из этих функций картирования, могут образовывать модуль, такой как модуль картирования, который может формировать часть системы (например, конвейер), используемую в процессе определения фактической полной геномной последовательности индивида или ее части.

[00183] Преимущество реализации хэш-модуля в аппаратном обеспечении состоит в том, что процессы могут быть ускорены и, следовательно, выполняться значительно быстрее. Например, когда программное обеспечение может включать в себя различные инструкции для выполнения одной или более из этих различных функций, реализация таких инструкций часто требует сохранения, и/или вызова, и/или считывания, и/или интерпретации данных и инструкций, например перед исполнением. Однако, как указано выше и подробно описано в настоящем документе, можно жестко смонтировать микросхему для выполнения этих функций без необходимости вызова, интерпретации и/или выполнения одной или более из последовательностей инструкций. Скорее, микросхему можно подключить для непосредственного выполнения таких функций. Соответственно, согласно различным аспектам изобретение относится к изготовляемой на заказ жестко смонтированной машине, которая может быть сконфигурирована таким образом, чтобы описанный выше модуль картирования, например, хэширования, частично или полностью был реализован с помощью одной или более сетевых схем, жестко смонтированных на микросхеме, такой как матрица FPGA или схема ASIC.

[00184] Например, в различных случаях построение индекса хэш-таблицы и выполнение хэш-функции могут осуществляться на микросхеме, а в других случаях индекс хэш-таблицы может формироваться вне микросхемы, например, с помощью программного обеспечения, выполняемого главным ЦПУ, но после формирования индекс загружается в аппаратное обеспечение или иным образом становится доступны для него и используется микросхемой, например, во время работы хэш-модуля. В частности, в различных случаях микросхема, такая как матрица FPGA, может быть выполнена с возможностью жесткого связывания с главным ЦПУ, например, посредством межсоединения с низкой задержкой, такого как межсоединение QPI. Более конкретно, микросхема и ЦПУ могут быть выполнены с возможностью жесткого связывания соединения вместе таким образом, чтобы совместно использовать один или более ресурсов памяти, например DRAM, в конфигурации с обеспечением когерентности кэша. В таком случае главная память может строить и/или включать в себя индекс референса, например, хэш-таблицу, которая может храниться в главной памяти, но быть легко доступной для матрицы FPGA, например, для использования ее

при выполнении хэширования или другой функции картирования. В конкретных вариантах реализации одно или более из ЦПУ и матрицы FPGA могут содержать один или более кэшей или регистров, которые могут быть соединены вместе для образования когерентной конфигурации, такой что данные, хранящиеся в одном кэше, могли быть по существу дублированы другим кэшем.

[00185] Соответственно, ввиду вышеизложенного, во время выполнения одна или более предварительно построенных хэш-таблиц, например, содержащих индекс референсного генома, или создаваемая или подлежащая созданию хэш-таблица, могут быть загружены во встроенную память или могут по меньшей мере быть сделаны доступными для главного приложения, как более подробно описано ниже в настоящем документе. В таком случае риды, например, хранящиеся в формате файла FASTQ, могут быть отправлены главным приложением во встроенные движки обработки, например, память или кэш или другой регистр, связанный с ними, например, для использования движков картирования, и/или выравнивания, и/или сортировки, например, когда их результаты могут быть отправлены в функцию определения вариантов и использоваться ею. В этой связи, как указано выше, в различных случаях может быть сформировано скопление перекрывающихся затравок, например, с помощью функции формирования затравок, и выделена из секвенированных ридов, или пары ридов, и после того как затравки сформированы, их можно хэшировать, например, относительно индекса, и искать в хэш-таблице, чтобы определять позиции-кандидаты картирования ридов в референсе.

[00186] Более конкретно, в различных случаях может быть предусмотрен модуль картирования, например, когда модуль картирования выполнен с возможностью осуществления одной или более функций картирования, например, в жестко смонтированной конфигурации. А именно, жестко смонтированный модуль картирования может быть выполнен с возможностью осуществления одной или более функций, обычно совершаемых одним или более алгоритмами, выполняемыми на ЦПУ, например, функций, которые, как правило, реализуют в основанном на программном обеспечении алгоритме, который создает дерево префиксов и/или суффиксов, выполняет преобразование Барроуза-Уилера и/или выполняет хэш-функцию, например, хэш-функцию, которая использует или иным образом опирается на индексирование хэш-таблицы, скажем, референса, например референсной геномной последовательности. В таких случаях хэш-функция может быть структурирована таким образом, чтобы реализовывать стратегию, такую как оптимизированная стратегия картирования, которая может быть осуществлена с возможностью сведения к минимуму количества выполняемых обращений к памяти, например, прямых доступов к памяти большого объема, чтобы таким образом максимально повысить использование пропускной способности встроенной или иным образом связанной памяти, которая может быть существенно ограничена, например пространством внутри архитектуры микросхемы.

[00187] Кроме того, в определенных случаях, чтобы сделать систему более эффективной, главное ЦПУ/ГПУ/КПУ может быть жестко связано со связанным аппаратным обеспечением, например матрицей FPGA, например, посредством интерфейса с низкой задержкой, такого как Quick Path Interconnect («QPI»), чтобы обеспечивать движкам обработки интегральной схемы возможность беспрепятственного доступа к главной памяти. В конкретных случаях межсоединение между главным ЦПУ и соединенной микросхемой и ее соответствующими связанными памятями, например одним или более устройствами DRAM, может быть выполнено с возможностью поддержания когерентности кэша. Таким образом, в различных вариантах реализации

может быть предусмотрена интегральная схема, которая предварительно сконфигурирована, например, предварительно смонтирована, таким образом, чтобы включать в себя одну или более цифровых логических схем, которые могут быть в монтажной конфигурации и могут быть взаимно соединены, например, с помощью множества физических электрических межсоединений, и в различных вариантах реализации жестко смонтированные цифровые логические схемы могут организованы в один или более движков обработки с образованием одного или более модулей, таких как модуль картирования.

[00188] Соответственно, в различных случаях может быть предусмотрен модуль картирования, например, в первой предварительно сконфигурированной монтажной, например, жестко смонтированной, конфигурации, где модуль картирования выполнен с возможностью осуществления различных функций картирования. Например, модуль картирования может быть выполнен с возможностью доступа по меньшей мере к некоторым из последовательности нуклеотидов в риде из множества ридов, полученных из секвенированного генетического образца субъекта, и/или генетической референсной последовательности, и/или индекса одной или более генетических референсных последовательностей в памяти или связанном с ней кэше, например, посредством интерфейса памяти, такого как межсоединение процесса, например Quick Path Interconnect и т.д. Модуль картирования может также быть выполнен с возможностью картирования рида на один или более сегментов указанных одной или более генетических референсных последовательностей, например на основе индекса. Например, в различных конкретных вариантах реализации алгоритм и/или модуль картирования, представленные в настоящем документе, могут быть использованы для построения или создания иным образом хэш-таблицы, с помощью которой можно сравнивать рид, или его часть, секвенированного генетического материала субъекта с одним или более сегментов референсного генома для получения картированных ридов. В таком случае по завершении выполнения картирования может быть выполнено выравнивание.

[00189] Например, после того, как определено, где находятся все возможные совпадения затравок с референсным геномом, необходимо определить, какое из всех этих возможных местоположений, где возможно совпадение данного рида, действительно является правильной позицией, с которой он выровнен. Таким образом, после картирования может быть множество позиций, где одно или более ридов, по-видимому, совпадают с референсным геномом. Следовательно, могут существовать множество затравок, которые, как представляется, указывают в точности одно и то же, например, они могут совпадать в точности с одной и той же позицией на референсе, если учитывать позицию затравки в риде. Поэтому для каждого данного рида необходимо определить подлинное выравнивание. Это определение можно осуществить несколькими различными способами.

[00190] В одном случае можно оценить все риды, чтобы определить их правильное выравнивание относительно референсного генома на основе позиций, указанных каждой затравкой из рида, которое вернуло информацию о позиции во время процесса картирования, например, хэшированного поиска. Однако в различных случаях перед выполнением выравнивания можно выполнить функцию фильтрации затравочной цепочки на одной или более затравок. Например, в определенных случаях затравки, связанные с данным ридом, которые, по-видимому, картируются на одно и то же общее место в референсном геноме, могут быть агрегированы в одну цепочку, которая ссылается на ту же общую область. Все затравки, связанные с одним ридом, могут быть сгруппированы в одну или более затравочных цепочек, чтобы каждая затравка входила

только в одну цепочку. Именно такие цепочки затем приводят к выравниваю рида с каждой указанной позицией в референсном геноме.

[00191] В частности, в различных случаях все затравки, которые имеют одни и те же подтверждающие данные, указывающие на то, что они принадлежат одним и тем же общим местоположениям в референсе, могут быть собраны вместе для формирования одной или более цепочек. Поэтому затравки, которые группируются вместе или по меньшей создают впечатление, что они окажутся рядом друг с другом в референсном геноме, например, в пределах определенной полосы, будут сгруппированы в цепочку затравок, а те, что находятся за пределами этой полосы, будут превращены в другую цепочку затравок. После того, как эти различные затравки агрегированы в одну или более различных затравочных цепочек, можно определить, какая из цепочек действительно представляет правильную цепочку, подлежащую выравниванию. Этом можно сделать, по меньшей мере частично, с помощью алгоритма фильтрации, который представляет собой эвристический алгоритм, выполненный с возможностью устранения слабых затравочных цепочек, с большой вероятностью не являющихся верными.

[00192] Результатом выполнения одной из этих функций картирования, фильтрации и/или редактирования является список ридов, который для каждого рида содержит список всех возможных местоположений, в которых рид может совпасть с референсным геномом. Следовательно, функцию картирования можно выполнить так, чтобы быстро определить, где риды из файла изображения, файла BCL и/или файла FASTQ, полученного из секвенатора, картируются на референсный геном, например, куда в полном геноме картируются различные риды. Однако при наличии ошибки в любом из ридов или генетической вариации можно не получить точного совпадения с референсом и/или могут быть несколько мест, с которыми, по-видимому, совпадают одно или более ридов. Поэтому необходимо определить, где различные риды действительно выровнены относительно генома в целом.

[00193] Соответственно, после картирования, и/или фильтрации, и/или редактирования определены позиции местоположений для большого количества ридов, причем для некоторых из отдельных ридов определены множество позиций местоположений, и теперь нужно установить, какие из всех возможных местоположений в действительности являются истинными или наиболее вероятными местоположениями, с которым выравниваются различные риды. Такое выравнивание может быть выполнено с помощью одного или более алгоритмов, таких как алгоритм динамического программирования, который сопоставляет картированные риды с референсным геномом и выполняет функцию выравнивания на нем. В качестве примера функция выравнивания сравнивает одно или более, например все риды с референсом, скажем, путем наложения одного на другое в графическом режиме, например, в таблице, такой как виртуальный массив или матрица, где последовательность одного из референсных геномов или картированных ридов помещают на одно измерение или ось, например, горизонтальную ось, а другую помещают на противоположные измерения или ось, такую как вертикальная ось. Затем поверх массива пропускают воображаемый фронт волны оценки, чтобы определить выравнивание ридов относительно референсного генома, например, путем вычисления оценок выравнивания для каждой ячейки в матрице.

[00194] Фронт волны оценки представляет одну или более, например, все, ячейки матрицы или части тех ячеек, которые могут быть оценены независимо друг от друга и/или одновременно в соответствии с правилами динамического программирования, применимыми в алгоритме выравнивания, например Смита-Ватермана или Нидлмана-Вунша, или родственных алгоритмах. Оценки выравнивания можно сравнить

последовательно или в других порядках, например, путем вычисления всех оценок в верхнем ряду слева направо, затем всех оценок в следующем ряду слева направо и т.д. При таком подходе развертывающийся по диагонали диагональный фронт волны представляет оптимальную последовательность пакетов оценок, вычисляемых одновременно или параллельно в серии этапов фронта волны.

[00195] Например, в одном варианте реализации окно референсного генома, содержащее сегмент, на который был картирован рид, может быть помещено на горизонтальную ось, а рид может быть расположен на вертикальной оси. Подобным образом формируют массив или матрицу, например, виртуальную матрицу, в результате чего нуклеотид в каждой позиции в риде можно сравнить с нуклеотидом в каждой позиции в окне референса. По мере прохождения фронта волны по массиву рассматриваются все потенциальные пути выравнивания рида с окном референса, в том числе нужно ли внести какие-либо изменения в одну последовательностью, чтобы рид совпал с референсной последовательностью, например, путем замены одного или более нуклеотидов рида на другие нуклеотиды, или вставки одного или более нуклеотидов в одну последовательность, или удаления одного или более нуклеотидов из одной последовательности.

[00196] Формируют оценку выравнивания, представляющую степень изменений, которые потребовалось бы внести для достижения точного выравнивания, причем эта оценка и/или другие связанные данные могут быть сохранены в данных ячейках массива. Каждая ячейка массива соответствует вероятности того, что нуклеотид в ее позиции на оси ридов выровнен с нуклеотидом в ее позиции на оси референса, а оценка, сформированная для каждой ячейки, представляет частичное выравнивание, заканчивающееся позициями ячейки в риде и окне референса. Наивысшая оценка, сформированная в любой ячейке, представляет лучшее общее выравнивание рида относительно окна референса. В различных случаях выравнивание может быть глобальным, где весь рид должен быть выровнен относительно некоторой части окна референса, например, с помощью алгоритма Нидлмана-Вунша или подобного алгоритма; или в других случаях выравнивание может быть локальным, где только часть рида может быть выровнено относительно части окна референса, например, с помощью алгоритма Смита-Ватермана или подобного алгоритма.

[00197] Соответственно, в различных случаях функция выравнивания может быть выполнена, например, на данных, полученных из модуля картирования. Таким образом, в различных случаях функция выравнивания может образовывать модуль, такой как модуль выравнивания, который может формировать часть системы (например, конвейер), которую используют, например, в дополнение к модулю картирования, в процессе определения фактической полной геномной последовательности индивида или ее части. Например, результат, возвращаемый после выполнения функции картирования, например, из модуля картирования, такой как список возможных вариантов мест, где одно или более или все риды картируются на одну или более позиций в одном или более референсных геномов, может быть использован функций выравнивания для определения фактического выравнивания последовательности секвенированной ДНК субъекта.

[00198] Такая функция выравнивания всегда может пригодиться, поскольку, как описано выше, часто по ряду различных причин секвенированные риды не всегда совпадают в точности с референсным геномом. Например, в одном или более ридов может быть ОНП (однонуклеотидный полиморфизм), например, замена одного нуклеотида другим в одной позиции; может быть «индел» (indel), инсерция или делеция

одного или более оснований в одной или более последовательностях рида, причем эта инсерция или делеция не присутствует в референсном геноме; и/или может быть ошибка секвенирования (например, ошибки в приготовлении образца, и/или риде секвенатора, и/или выходных данных секвенатора и т.д.), вызывающая одну или более из этих очевидных вариаций. Соответственно, когда рид отличается от референса, например вследствие ОНП или индела, причина может быть в том, что референс отличается от истинной последовательности ДНК, взятой в качестве образца, или в том, что рид отличается от истинной последовательности ДНК, взятой в качестве образца. Проблема состоит в том, чтобы выяснить, как правильно выровнять риды на референсный геномом с учетом того факта, что, по всей вероятности, между этими двумя последовательностями будет множество различных отличий.

[00199] В различных случаях входными данными функции выравнивания, например, из функции картирования, такой как дерево префиксов/суффиксов, или преобразование Барроуза-Уилера, или хэш-таблица и/или хэш-функция, может быть список возможных вариантов мест, где одно или более ридов могут быть картированы на одну или более позиций одного или более референсных последовательностей. Например, любой данный рид может совпадать с любым количеством позиций в референсном геноме, например, в 1 местоположении, или 16, или 32, или 64, или 100, или 500, или 1000 или более местоположениях, где данный рид картируется на геном. Однако любой отдельный рид был получено, например, секвенирован, только из одной определенной части генома. Следовательно, чтобы найти, откуда был получен данный конкретный рид, можно выполнить функцию выравнивания, например, выравнивание Смита-Ватермана с гэпами и без гэпов, выравнивание Нидлмана-Вунша и т.д., чтобы определить, где в геноме в действительности были получены одно или более ридов, например, путем сравнения всех возможных местоположений, где имеет место совпадение, и определения того, какой из всех возможных вариантов является наиболее вероятным местоположением, из которого был секвенирован рид, исходя из наивысшей оценки выравнивания местоположений.

[00200] Как было указано, для выполнения такой функции выравнивания обычно используют алгоритм. Например, для выравнивания двух или более последовательностей друг с другом можно использовать алгоритм выравнивания Смита-Ватермана и/или Нидлмана-Вунша. В этом случае они могут быть использованы таким образом, чтобы для любой данной позиции, где рид картируется на референсный геном, определить вероятности того, что картирование действительно выполнено в позиции, откуда происходит рид. Как правило, эти алгоритмы выполнены с возможностью осуществления программным обеспечением, однако в различных случаях, таких как представленные в настоящем документе, один или более из этих алгоритмов может быть выполнен с возможностью осуществления в аппаратном обеспечении, как более подробно описано ниже в настоящем документе.

[00201] В частности, функцию выравнивания используют по меньшей мере частично для выравнивания одного или более, например, всех, ридов с референсным геномом, несмотря на наличие одной или более позиций несовпадений, например, ОНП, инсерций, делеций, структурных артефактов и т.д., чтобы определить, где эти риды, по всей видимости, правильно впишутся в геномом. Например, одно или более ридов сравнивают с референсным геномом и определяют наилучшее возможное совпадение рида с геномом, учитывая при этом замены, и/или инделы, и/или структурные варианты. Однако, чтобы лучше определять, какая из модифицированных версий рида лучше всего вписывается в референсный геном, необходимо учитывать предполагаемые изменения, и поэтому

можно также выполнить функцию оценки.

[00202] Например, можно выполнить функцию оценки, например, как часть общей функции выравнивания, в рамках которой модуль выравнивания выполняет свою функцию и вводит одно или более изменений в последовательность, сравниваемую с другой последовательностью, например, чтобы достичь более хорошего или наилучшего соответствия между ними, при этом для каждого изменения, вносимого для достижения более хорошего выравнивания, из начальной оценки вычитают некоторое число, например, либо из идеальной оценки, либо из нулевой начальной оценки, таким образом, чтобы при выполнении выравнивания определять также оценку этого выравнивания, например, когда обнаруживают совпадения, оценку увеличивают, а при каждом внесенном изменении накладывают штраф, и таким образом можно определять лучшее возможно соответствие для возможных выравниваний путем выявления из всех возможных модифицированных ридов того рида, соответствие которого геному имеет наивысшую оценку. Соответственно, в различных случаях функция выравнивания может быть выполнена с возможностью определения лучшей комбинации изменений, которые нужно внести в рид (-ы) для достижения выравнивания с наивысшей оценкой, и тогда это выравнивание может быть определено как правильное или наиболее вероятное выравнивание.

[00203] Поэтому, ввиду вышеизложенного, существуют по меньшей мере две цели, которые могут быть достигнуты за счет выполнения функции выравнивания. Одна из них - это отчет о наилучшем выравнивании, включающий в себя позицию в референсном геноме и описание изменений, которые необходимы для того, чтобы рид совпал с референсным сегментом в этой позиции, а другая - оценка качества выравнивания. Например, в различных случаях выходными данными из модуля выравнивания может быть Compact Idiosyncratic Gapped Alignment Report, например, строка CIGAR, где выходная строка CIGAR представляет собой отчет, подробно описывающий все изменения, которые вносили в риды, чтобы достичь для них наиболее соответствующего выравнивания, например, подробные инструкции по выравниванию, указывающие, каким образом исследуемая последовательность действительно выравнивает с референсом. Вывод такой строки CIGAR может быть полезным на последующих стадиях обработки для более хорошего определения того, что для данной геномной нуклеотидной последовательности индивида прогнозированные вариации в сравнении с референсным геномом действительно являются истинными вариациями, а не просто обусловлены ошибкой машины, программного обеспечения или человека.

[00204] Как было указано выше, в различных вариантах реализации выравнивание, как правило, выполняют последовательно, причем алгоритм и/или прошивка принимают данные последовательности рида (например, из модуля картирования), принадлежащие риду, и одно или более возможных местоположений, где этот рид потенциально может быть картирован на один или более референсных геномов, а также принимают данные геномной последовательности (например из одной или более памятей, такой как связанные DRAM), относящиеся к одной или более позиций в одном или более референсных геномах, на которые может быть картирован рид. В частности, в различных вариантах реализации модуль картирования обрабатывает риды, например, из файла FASTQ, и картирует каждое из них на одну или более позиций в референсном геноме, на которую они, возможно, выровнены. Затем выравниватель берет эти прогнозированные позиции и использует их для выравнивания ридов на референсный геном, например, путем построения виртуального массива, с помощью которого риды можно сравнивать с референсным геномом.

[00205] При выполнении этой функции выравниватель оценивает каждую картированную позицию для каждого отдельного рида и, в частности, оценивает те риды, которые картированы на множество возможных местоположений в референсном геноме, и для каждой позиции оценивает возможность того, что она является правильной. Затем он сравнивает лучшие оценки, например, две лучшие оценки, и принимает решение о том, где действительно выравнивается конкретный рид. Например, при сравнении первой и второй лучших оценок выравнивания выравниватель проверяет разницу между оценками, и если разница между ними большая, то оценка достоверности того, что позиция с большей оценкой является правильной, будет высокой. Однако, если разница между ними маленькая, например, нулевая, то оценка достоверности выбора из двух позиций одной из них в качестве правильной позиции, из которой получен рид, низкая и, возможно, будет полезна дополнительная обработка, чтобы четко определить истинное местоположение в референсном геноме, из которого получен рид.

[00206] Поэтому выравниватель, в частности, ищет наибольшую разницу между первой и второй лучшими оценками достоверности для принятия решения о том, что данный рид картируется на данное местоположение в референсном геноме. В идеале оценка лучшего возможного варианта выравнивания значительно выше оценки второго лучшего выравнивания для данной последовательности. Существуют множество различных способов реализации метода оценки выравнивания, например, можно оценивать каждую ячейку массива или подмножество ячеек, например, в соответствии со способами, описанными в настоящем документе. В различных случаях параметры оценки совпадений нуклеотидов, несовпадений нуклеотидов, инсерций и делеций могут иметь любые различные положительные, отрицательные или нулевые значения. В различных случаях эти параметры оценки могут быть изменены на основе имеющейся информации. Например, точные выравнивания могут быть достигнуты путем изменения параметров оценки, в том числе любого или всех из оценок совпадения нуклеотидов, оценок несовпадения нуклеотидов, штрафов на гэп (инсерция и/или делеция), штрафов на открытие гэпа и/или штрафов на продление гэпов, в соответствии с оценкой качества основания, связанной с текущим нуклеотидом или позицией рида. Например, бонусы и/или штрафы оценки могут быть уменьшены, когда оценка качества основания указывает на высокую вероятность наличия ошибок секвенирования или других ошибок. Чувствительную к качеству основания оценку можно реализовать, например, с помощью фиксированной или выполненной с возможность конфигурирования таблицы подстановки, доступной с помощью оценки качества основания, которая возвращает соответствующие параметры оценки.

[00207] В случае аппаратной реализации в интегральной схеме, такой как матрица FPGA или схема ASIC, фронт волны оценки может быть реализован в виде линейного массива ячеек оценки, например, 16 ячеек, или 32 ячейки, или 64 ячейки, или 128 ячеек и т.п. Каждая из ячеек оценки может быть построена из цифровых логических элементов в монтажной конфигурации для вычисления оценок выравнивания. Таким образом, для каждого этапа фронта волны, например, каждого тактового цикла или некоторых других фиксированных или переменных единиц времени, каждая из ячеек оценки, или часть ячеек, вычисляет оценку или оценки, требуемые для новой ячейки в виртуальной матрице выравнивания. Теоретически различные ячейки оценки считаются находящимися в различных позициях матрицы выравнивания, соответствующих фронту волны оценки, как отмечалось в настоящем документе, например, вдоль прямой линии, проходящей из нижней левой части в верхнюю правую часть матрицы. Как известно из области разработки цифровых логических устройств, физические ячейки оценки и составляющая

их цифровая логика не должны быть физически расположены подобным образом на интегрированной схеме.

[00208] Соответственно, по мере того, как фронт волны шаг за шагом прокатывается по виртуальной матрице выравнивания, воображаемые позиции ячеек оценки соответствующим образом обновляют каждую ячейку, например, умозрительно «перемещаются» на шаг вправо или, например, на шаг вниз в матрице выравнивания. Все ячейки оценки совершают одинаковое относительное воображаемое перемещение, сохраняя порядок диагонального фронта волны. Всякий раз, когда фронт волны перемещается в новое положение, например, за счет шага в вертикальном направлении вниз или шага в горизонтальном направлении вправо в матрице, ячейки оценки прибывают в новые воображаемые позиции и вычисляют оценки выравнивания для ячеек виртуальной матрицы выравнивания, в которые они вошли. В такой реализации соседние ячейки оценки в линейном массиве соединены для обмена исследуемыми (принадлежащими риду) нуклеотидами, референсными нуклеотидами и ранее вычисленными оценками выравнивания. Нуклеотиды окна референса могут последовательно подаваться на один конец фронта волны, например, в верхнюю правую ячейку оценки в линейном массиве, и могут последовательно сдвигаться оттуда вниз вдоль фронта волны, чтобы в любой данный момент времени сегмент референсных нуклеотидов, равный по длине количеству ячеек оценки, присутствовал в этих ячейках, по одному следующему один за другим нуклеотидов в каждой следующей одна за другой ячейке.

[00209] Например, всякий раз, когда фронт волны перемещается на шаг в горизонтальном направлении, следующий референсный нуклеотид подают в верхнюю правую ячейку, а другие референсные нуклеотиды сдвигаются вниз влево по фронту волны. Этот сдвиг референсных нуклеотидов может быть реальным отражением воображаемого перемещения фронта волны ячеек оценки вправо в матрице выравнивания. Таким образом, нуклеотиды рида могут последовательно подаваться на противоположный конец фронта волны, например, в нижнюю левую ячейку оценки в линейном массиве, и могут последовательно сдвигаться оттуда вверх вдоль фронта волны, чтобы в любой данный момент времени сегмент исследуемых нуклеотидов, равный по длине количеству ячеек оценки, присутствовал в этих ячейках, по одному следующему один за другим нуклеотидов в каждой следующей одна за другой ячейке. Аналогичным образом всякий раз, когда фронт волны перемещается на шаг в вертикальном направлении, следующий исследуемый нуклеотид подают в нижнюю левую ячейку, а другие исследуемые нуклеотиды сдвигаются вверх вправо по фронту волны. Этот сдвиг исследуемых нуклеотидов может быть реальным отражением воображаемого перемещения фронта волны ячеек оценки вниз в матрице выравнивания. Соответственно, подавая команду на сдвиг референсных нуклеотидов, можно перемещать фронт волны на шаг в горизонтальном направлении, а подавая команду на сдвиг исследуемых нуклеотидов, можно перемещать фронт волны на шаг в вертикальном направлении. Таким образом, для получения в целом диагонального перемещения фронта волны, например, чтобы следовать типичному выравниванию исследуемой и референсной последовательностей без инсерций или делеций, можно попеременно подавать команды на перемещение фронта волны на шаг в вертикальном и горизонтальном направлениях.

[00210] Соответственно, соседние ячейки оценки в линейном массиве могут быть соединены для обмена ранее вычисленными оценками выравнивания. В различных алгоритмах оценки выравнивания, таких как алгоритм Смита-Ватермана или Нидлмана-

Вунша и т.п., оценки в каждой ячейке виртуальной матрицы выравнивания могут быть вычислены с помощью раннее вычисленных оценок в других ячейках матрицы, например, в трех ячейках, расположенных непосредственно слева от текущей ячейки, выше текущей ячейки и вверх влево по диагонали от текущей ячейки. Когда ячейка оценки вычисляет новые оценки для другой позиции матрицы, в которую она входит, она должна извлечь такие ранее вычисленные оценки, соответствующие таким другим позициям матрицы. Эти ранее вычисленные оценки могут быть получены из хранилища ранее вычисленных оценок внутри этой же ячейки и/или из хранилища ранее вычисленных оценок в одной или двух соседних ячейках оценки в линейном массиве. Дело в том, что три вносящие вклад в оценку позиции в виртуальной матрице выравнивания (непосредственно слева, сверху и сверху слева по диагонали) могли быть оценены либо текущей ячейкой оценки, либо одной из ее соседних ячеек оценки в линейном массиве.

[00211] Например, ячейка непосредственно слева в матрице могла быть оценена текущей ячейкой оценки, если самый последний шаг фронта волны был в горизонтальном направлении (вправо), или могла быть оценена соседней ячейкой снизу слева в линейной матрице, если самый последний шаг фронта волны был в вертикальном направлении (вниз), Аналогичным образом ячейка непосредственно сверху в матрице могла быть оценена текущей ячейкой оценки, если самый последний шаг фронта волны был в вертикальном направлении (вниз), или могла быть оценена соседней ячейкой сверху справа в линейной матрице, если самый последний шаг фронта волны был в горизонтальном направлении (вправо), В частности, ячейка сверху слева по диагонали в матрице могла быть оценена текущей ячейкой оценки, если два самых последних шага фронта волны были в разных направлениях, например, вниз затем направо, или вправо затем вниз, или могла быть оценена соседней ячейкой сверху справа в линейном массиве, если два самых последних шага фронта волны были оба в горизонтальном направлении (вправо), или могла быть оценена соседней ячейкой снизу слева в линейном массиве, если два самых последних шага фронта волны были оба в вертикальном направлении (вниз).

[00212] Соответственно, учитывая информацию о направлениях последних одного или двух шагов фронта волны, ячейка оценки может выбрать надлежащие ранее вычисленные оценки, получив доступ к ним внутри себя и/или в соседних ячейках оценки, используя соединение между соседними оценками. В качестве варианта на наружных входах для оценок ячеек оценки на двух концах фронта волны могут быть жестко смонтированы недопустимые нулевые, или имеющие минимальное значение оценки, чтобы они не повлияли на новые вычисления оценок в этих крайних ячейках. Благодаря реализуемому таким образом фронту волны в линейном массиве ячеек оценки при таком соединении для сдвига референсных и исследуемых нуклеотидов по массиву в противоположных направлениях, чтобы умозрительно перемещать фронт волны пошагово в вертикальном и горизонтальном направлении, например, по диагонали, и соединении для доступа к оценкам, ранее вычисленным соседними ячейками, чтобы вычислять оценки выравнивания в новых позициях ячеек виртуальной матрицы, в которых входит фронт волны, можно, соответственно, оценивать в виртуальной матрице полосу ячеек шириной с фронт волны, например, путем подачи команд на последовательное пошаговое перемещение фронта волны, чтобы он прокатился по матрице.

[00213] Следовательно, чтобы выровнять новый рид и окно референса, фронт волны может начинаться изнутри матрицы оценки, или, преимущественно, может постепенно входить в матрицу оценки снаружи, начиная, например, слева, или сверху, или по

диагонали слева и сверху с верхнего левого угла матрицы. Например, фронт волны может начинаться с его верхней левой ячейки оценки, расположенной сразу слева от верхней левой ячейки виртуальной матрицы, а затем фронт волны может вкатываться вправо в матрицу с помощью серии горизонтальных шагов, оценивая горизонтальную полосу ячеек в верхней левой области матрицы. Когда фронт волны достигает прогнозированного соотношения выравнивания между референсной и исследуемой последовательностью, или когда обнаруживается совпадение на основе возрастания оценок выравнивания, фронт волны может начать прокатываться по диагонали вниз вправо за счет попеременных вертикальных и горизонтальных шагов, оценивая диагональную полосу ячеек посередине. Когда нижняя левая ячейка оценки фронта волны достигает низа матрицы выравнивания, фронт волны может начать снова прокатываться вправо за счет последовательных горизонтальных шагов, оценивая горизонтальную полосу ячеек в нижней правой области матрицы, до тех пор, пока некоторые или все ячейки фронта волны не выйдут за границы матрицы выравнивания.

[00214] Одна или более таких процедур выравнивания могут быть выполнены с помощью любого подходящего алгоритма выравнивания, такого как алгоритм выравнивания Нидлмана-Вунша и/или алгоритма выравнивания Смита-Ватермана, которые могут быть изменены для приведения в соответствие с функциональными возможностями, описанными в настоящем документе. Вообще оба эти и подобные им алгоритмы по сути работают, в некоторых случаях, аналогичным образом. Например, как указано выше, эти алгоритмы выравнивания, как правило, строят виртуальный массив похожим образом так, что в различных случаях, горизонтальная верхняя граница может быть выполнена с возможностью представления геномной референсной последовательности, которая может быть выложена по всему верхнему ряду массива в соответствии с ее составов пар оснований. Аналогичным образом вертикальная граница может быть выполнена таким образом, чтобы представлять секвенированные и картированные исследуемые последовательности, которые были расположены в порядке вниз вдоль первого столбца так, что порядок их нуклеотидной последовательности в основном совпадает с нуклеотидной последовательностью референса, на который они картированы. Тогда промежуточные ячейки могут быть заполнены оценками вероятности того, что соответствующее основание исследуемой последовательности в данной позиции расположено в этом местоположении относительно референса. При выполнении этой функции полоса захвата может перемещаться по диагонали по всей матрице, заполняя оценки внутри промежуточных ячеек, и, начиная с указанной позиции, можно определить вероятность для каждого основания исследуемой последовательности.

[00215] Что касается функции выравнивания Нидлмана-Вунша, которая создает оптимальные глобальные (или полуглобальные) выравнивания, выравнивающие все последовательности рида на некоторый сегмент референсного генома, управление направлением движения фронта волны может быть сконфигурировано так, чтобы он, как правило, прокатывался от самого верхнего края матрицы выравнивания, до самого нижнего края. По завершении проката фронта волны выбирают максимальную оценку на нижней границе матрицы выравнивания (соответствующей концу рида), и выравнивание отслеживают в обратном направлении к ячейке на верхнем крае матрицы (соответствующем началу рида). В различных случаях, описанных в настоящем документе, риды могут быть любой длины, могут быть любого размера, и для описания выполнения выравнивания не требуется обширных параметров, например, в различных случаях длина рида может быть такой же, как у хромосомы. Однако в таком случае

размер памяти и длина хромосомы могут быть ограничивающим фактором.

[00216] Что качается алгоритма Смита-Ватермана, который формирует оптимальные локальные выравнивания, выравнивая всею последовательность рида или часть последовательности рида на некоторый сегмент референсного генома, этот алгоритм может быть выполнен с возможностью осуществления поиска лучшей возможной оценки на основе полного или частичного выравнивания рида. Поэтому в различных случаях оцениваемая фронтом волны полоса может не доходить до верхнего и/или нижнего краев матрицы выравнивания, например, если очень длинный рид имеет лишь затравки при картировании его середины на референсный геном, но обычно фронт волны все же может выполнять оценку от верха до низа матрицы. Локальное выравнивание обычно достигают двумя регулировками. Во-первых, запрещено падение оценок выравнивания ниже нуля (или некоторого другого нижнего порога), и если, в противном случае, вычисленная оценка ячейки будет отрицательной, ее заменяют нулевой оценкой, представляющей начало нового выравнивания. Во-вторых, в качестве завершающей точки выравнивания используют максимальную оценку выравнивания, полученную в любой ячейке матрицы, необязательно вдоль нижнего края. Выравнивание отслеживают в обратном порядке от этой максимальной оценки вверх и влево по всей матрице до нулевой оценки, которую используют в качестве начальной позиции локального выравнивания, даже если она не находится в верхнем ряду матрицы.

[00217] Ввиду вышеизложенного существуют несколько возможных путей через виртуальный массив. В различных вариантах реализации фронт волны начинается с верхнего левого угла виртуального массива и движется вниз к идентификаторам максимальной оценки. Например, результаты всех возможных выравниваний можно собрать, обработать, коррелировать и оценить, чтобы определить максимальную оценку. Когда достигнут конец границы или конец массива и/или определено вычисление, приведшее к наивысшей оценке для всех обработанных ячеек (например, выявлена общая наивысшая оценка), можно выполнить обратное отслеживание, чтобы найти путь, который привел к достижению этой наивысшей оценки. Например, можно найти путь, который ведет к прогнозированной максимальной оценке, и после этот можно выполнить аудит, чтобы определить, каким образом была получена эта максимальная оценка, например, путем перемещения в обратном направлении, следуя стрелкам выравнивания с лучшей оценкой, указывающими обратное прохождение пути, который привел к достижению выявленной максимальной оценки, например, вычисленной с помощью ячеек оценки фронта волны.

[00218] Эта обратная реконструкция или обратное отслеживание включает в себя начало движения с определенной максимальной оценки и возвращение в начало через предыдущие ячейки с прокладыванием пути через ячейки, имеющие оценки, которые привели к достижению максимальной оценки, от самого верха таблицы и обратно до начальной границы, такой как начало массива или нулевая оценка в случае локального выравнивания. Во время обратного отслеживания после достижения конкретной ячейки в матрице выравнивания следующий шаг обратного отслеживания совершают в соседнюю ячейку непосредственно слева, или сверху, или по диагонали вверх влево, которая внесла вклад в лучшую оценку, выбранную для построения оценки в текущей ячейке. Таким образом можно отследить эволюцию максимальной оценки, тем самым выяснив, каким образом была достигнута максимальная оценка. Обратное отслеживание может завершиться в углу, или на крае, или на границе, или может закончиться на нулевой оценке, например, в левом верхнем углу массива. Соответственно, именно таким обратным отслеживанием определяют правильное выравнивание и тем самым

получают выходную строку CIGAR, которая показывает, как геномная последовательность, или ее часть, из образца, взятого у индивида, совпадает с геномной последовательностью референсной ДНК или иным образом выравнивается на нее.

[00219] После того, как определено, куда картируется каждый рид, и также определено, где выравнивается каждый рид, например, каждому соответствующему риду даны позиция и оценка качества, отражающая вероятность того, что эта позиция является правильным выравниванием, так что нуклеотидная последовательность для ДНК субъекта известна, можно проверить порядок различных ридов и/или геномную последовательность нуклеиновых кислот субъекта, например, путем определения идентичности каждой нуклеиновой кислоты в ее правильном порядке в геномной последовательности образца. Поэтом в соответствии с некоторыми аспектами настоящее изобретение относится к функции обратного отслеживания, например, являющейся частью модуля выравнивания, который выполняет как функцию выравнивания, так и функцию обратного отслеживания, такого как модуль, который может быть частью конвейера модулей, например, конвейера, который предназначен для приема необработанных данных рида последовательности, например, в виде геномного образца индивида, и картирования и/или выравнивания этих данных, которые могут быть затем сохранены.

[00220] Для облегчения операции обратного отслеживания полезно сохранять вектор оценки для каждой оцененной ячейки в матрице выравнивания, кодирующий решение относительно выбора оценки. В случае классических интерпретаций оценки Смита-Ватермана и/или Нидлмана-Вунша с линейными штрафами на гэп вектор оценки может кодировать четыре возможности, которые могут быть, необязательно, сохранены в виде 2-битового целого числа от 0 до 3, например: 0 = новое выравнивание (выбрана нулевая оценка); 1 = вертикальное выравнивание (выбрана оценка из ячейки сверху, изменена за счет штрафа на гэп); 2 = горизонтальное выравнивание (выбрана оценка из ячейки слева, изменена за счет штрафа на гэп); 3 = диагональное выравнивание (выбрана оценка из ячейки сверху и слева, изменена за счет оценки совпадения или не совпадения нуклеотида). Необязательно, можно также сохранять вычисленные оценки для каждой оцененной ячейки матрицы (в дополнение к максимальной достигнутой оценке выравнивания, которую обычно сохраняют), но для обратного отслеживания в этом нет необходимости, а данная информация может занимать огромные объемы памяти. После этого выполнение обратного отслеживания сводится к следующим векторам оценки; когда обратное отслеживание достигло данной ячейки в матрице, следующий шаг обратного отслеживания определяется сохраненным вектором оценки для этой ячейки, например: 0 = завершить обратное отслеживание; 1 = обратное отслеживание вверх; 2 = обратное отслеживание влево; 3 = обратное отслеживание вверх влево.

[00221] Такие векторы оценки могут храниться в двумерной таблице, скомпонованной в соответствии с размерами матрицы выравнивания, которая может заполняться только записями, соответствующими ячейкам, оцениваемым с помощью фронта волны. В альтернативном варианте реализации для экономии памяти, упрощения записи векторов оценки по мере их формирования и более простого приспособления матриц выравнивания различных размеров, векторы оценки можно сохранять в таблице, размер каждой строки которой определяется в зависимости от сохраняемых векторов оценки из одного фронта волны ячеек оценки, например, 128 битов для хранения 64 2-битовых векторов оценки фронта волны, состоящего из 64 ячеек, а количество строк равно максимальному количеству шагов фронта волны в операции выравнивания. Кроме

того, для этого варианта можно вести запись направлений различных шагов фронта волны, например, сохраняя дополнительный, например, 129-й, бит в каждой строке таблицы, кодируя, например, с помощью 0 вертикальный шаг фронта волны, предшествующий данному положению фронта волны, а с помощью 1 горизонтальный шаг фронта волны, предшествующий данному положению фронта волны. Этот дополнительный бит можно использовать во время обратного отслеживания для контроля за тем, каким позициям виртуальной матрицы оценки соответствуют векторы оценки в каждой строке таблицы, чтобы после каждого последующего шага можно было извлекать правильный вектор оценки. Если шаг обратного отслеживания вертикальный или горизонтальный, следующий вектор оценки следует извлекать из предыдущей строки таблицы, но если шаг обратного отслеживания диагональный, следующий вектор оценки следует извлекать двумя строками выше, так как фронт волны должен сделать два шага, чтобы перейти от оценки любой одной ячейки к оценке ячейки справа внизу от нее.

[00222] В случае аффинной оценки гэпов информация вектора оценки может быть расширена, например, до 4 битов на оцениваемую ячейку. В дополнение к, например, 2-битовому индикатору направления выбора оценки, могут быть добавлены два 1-битовых флага - флаг вертикального продления и флаг горизонтального продления. В соответствии с методами расширений аффинной оценки гэпов для алгоритмов Смита-Ватермана, Нидлмана-Вунша или подобных алгоритмов, для каждой ячейки в дополнение к первичной оценке выравнивания, представляющей выравнивания с лучшей оценкой, прекращающееся в этой ячейке, следует формировать «вертикальную оценку», соответствующую максимальной оценке выравнивания, достигаемой этой ячейкой с помощью завершающего вертикального шага, и «горизонтальную оценку», соответствующую максимальной оценке выравнивания, достигаемой этой ячейкой с помощью завершающего горизонтального шага; и при вычислении любой из трех оценок вертикальный шаг в ячейку может быть вычислен либо с помощью первичной оценки из ячейки сверху минус штраф на открытие гэпа, либо с помощью вертикальной оценки из ячейку сверху минус штраф на продление гэпа в зависимости от того, что больше; а горизонтальный шаг в ячейку может быть вычислен либо с помощью первичной оценки из ячейки слева минус штраф на открытие гэпа, либо с помощью горизонтальной оценки из ячейки слева минус штраф на продление гэпа в зависимости от того, что больше. В случае выбора вертикальной оценки минус штраф на продление гэпа в векторе оценки должен быть установлен флаг вертикального продления, например, «1», а в противном случае он должен быть снят, например, «0».

[00223] В случае выбора горизонтальной оценки минус штраф на продление гэпа в векторе оценки должен быть установлен флаг горизонтального продления, например, «1», а в противном случае он должен быть снят, например, «0». Во время обратного отслеживания в случае аффинной оценки гэпов всякий раз, когда при обратном отслеживании совершают вертикальный шаг вверх из данной ячейки и флаг вертикального продления в векторе оценки этой ячейки установлен, следующий шаг обратного отслеживания тоже должен быть вертикальным независимо от вектора оценки для ячейки сверху. Аналогичным образом всякий раз, когда при обратном отслеживании совершают горизонтальный шаг влево из данной ячейки и флаг горизонтального продления в векторе оценки этой ячейки установлен, следующий шаг обратного отслеживания тоже должен быть горизонтальным независимо от вектора оценки для ячейки слева. Соответственно, такой таблицы векторов оценки, имеющей, например, 129 битов на строку для 64 ячеек при использовании линейной оценки гэпов,

или 257 битов на строку для 64 ячеек при использовании аффинной оценки гэпов, и некоторое количество, NR, строк, достаточно для поддержки обратного отслеживания после завершения оценки выравнивания, где фронт волны оценки выполняет NR шагов или менее.

[00224] Например, при выравнивании ридов из 300 нуклеотидов количество шагов фронта волны может быть всегда меньше 1024, поэтому таблица может занимать 257×1024 битов, или приблизительно 32 килобайта, что во многих случаях может быть разумной локальной памятью внутри интегральной схемы. Но если требуется выравнивать очень длинные риды, например, из 100000 нуклеотидов, требования к памяти для векторов оценки могут быть довольно большими, например, 8 мегабайтов, что может быть слишком дорогим для включения ее в качестве локальной памяти внутрь интегральной схемы. Для такой поддержки информацию векторов оценки можно записывать в память большой емкости вне интегральной схемы, например, DRAM, но тогда слишком дорогими могут стать требования к полосе пропускания, например, 257 битов на тактовый цикл для каждого модуля выравнивания, что может привести к затору и резко уменьшить производительность выравнивателя. Соответственно, желательно иметь способ для размещения векторов оценки до завершения выравнивания, чтобы сохранять ограниченными требования, предъявляемые ими к памяти, например, для выполнения инкрементальных обратных отслеживаний, формирующих инкрементальные частичные строки CIGAR, например, из начальных частей из истории векторов оценки выравнивания, чтобы такие начальные части векторов оценки могли быть отброшены. Проблема состоит в том, что, как предполагается, обратное отслеживание начинается с завершающей точки выравнивания, ячейки с максимальной оценкой, которая неизвестна до тех пор, пока не завершится оценка выравнивания, поэтому любое обратное отслеживание, начинающееся до завершения выравнивания, может начаться из неверной ячейки, не по возможному окончательному оптимальному пути выравнивания.

[00225] Поэтому предложен способ для выполнения инкрементального обратного отслеживания на основе частичной информации о выравнивании, например, содержащей частичные сведения о векторах оценки для ячеек матрицы выравнивания, оцененных на текущий момент. Исходя из границы выполненного на данный момент выравнивания, например, конкретного положения фронта волны, обратное отслеживание начинают из всех позиций ячеек на границе. Такое обратное отслеживание из всех граничных ячеек может быть выполнено последовательно или, преимущественно, особенно в случае аппаратной реализации, все обратные отслеживания могут быть выполнены вместе. Выделять символические записи выравнивания, например, строки CIGAR, из этих многочисленных обратных отслеживаний не требуется; нужно только определять, какие позиции матрицы выравнивания они проходят во время обратного отслеживания. При реализации одновременного обратного отслеживания с границы можно использовать инициализированные, например, установленные на «1», 1-битовые регистры в количестве, соответствующем количеству ячеек выравнивания, которые показывают, проходит ли какое-либо из обратных отслеживаний через соответствующую позицию. Для каждого шага одновременного обратного отслеживания можно проверять векторы оценок, соответствующие всем текущим «1» в этих регистрах, например, из одной строки таблицы векторов оценки, чтобы определять следующий шаг обратного отслеживания, соответствующий каждой «1» в регистрах, ведущий к следующей позиции для каждой «1» в регистрах, для следующего шага одновременного обратного отслеживания.

[00226] Важно отметить, что весьма велика вероятность объединения множества «1» в регистрах в общие позиции, соответствующие множеству одновременных обратных отслеживаний, сливающихся в общие маршруты обратного отслеживания. После того, как два или более одновременных отслеживаний сливаются, далее они все время остаются объединенными, так как с этого момента они будут использовать информацию вектора оценки из одной и той же ячейки. Было замечено, эмпирически и по теоретическим соображениям, что с высокой вероятностью все одновременные отслеживания сольются в сингулярный маршрут обратного отслеживания за относительно малое количество шагов обратного отслеживания, количество которых, например, может быть кратно, количеству ячеек оценки во фронте волны с небольшим коэффициентом, например, 8-кратным. Например, в случае фронта волны из 64 ячеек с высокой вероятностью все обратные отслеживания с данной границы фронта волны сольются в один маршрут обратного отслеживания в пределах 512 шагов обратного отслеживания. Или же, также возможно, и не редко, что все обратные отслеживания завершатся в пределах этого числа, например, 512 шагов обратного отслеживания.

[00227] Соответственно, множество одновременных обратных отслеживаний могут быть выполнены с границы оценки, например, оцененной позиции фронта волны, достаточно далеко, чтобы они все либо завершились, либо слились в единственный маршрут обратного отслеживания, например, за 512 шагов или меньше. Если они все сольются в сингулярный маршрут обратного отслеживания, то начиная с места в матрице, где они слились, или на любом расстоянии дальше по сингулярному маршруту обратного отслеживания возможно инкрементальное обратное отслеживание на основе частичной информации о выравнивании. Далее начинают обратное отслеживание из точки слияния или любого расстояния дальше назад с помощью обычных способов сингулярного обратного отслеживания, включающих в себя запись соответствующей символической записи, например, частичной строки CIGAR. Это инкрементальное обратное отслеживание и, например, частичная строка CIGAR, должны быть частью любого возможного окончательного обратного отслеживания, и, например, полной строки CIGAR, которая получится по завершении выравнивания, если такое окончательное обратное отслеживание не завершится раньше достижения границы оценки, где начались одновременные обратные отслеживания, так как если оно достигнет границы оценки, оно должно следовать одним из маршрутов одновременного отслеживания и слиться с сингулярным маршрутом обратного отслеживания, выделенным теперь инкрементально.

[00228] Следовательно, все векторы оценки для областей матрицы, соответствующих инкрементально выделенному обратному отслеживанию, например, во всех строках таблицы для позиций фронта волны, предшествующих началу выделенного сингулярного обратного отслеживания, могут быть безопасно отброшены. Если при выполнении завершающего обратного отслеживание из ячейки с максимальной оценкой оно завершается раньше достижения границы оценки (или же, если оно завершается раньше достижения начала выделенного сингулярного обратного отслеживания), инкрементальную символическую запись, например, частичную строку CIGAR, можно отбросить. Если завершающее обратное отслеживание продолжается до начала выделенного сингулярного обратного отслеживания, тогда символическая запись его выравнивания может быть наращена на инкрементальную символическую запись выравнивания, например, частичную строку CIGAR. Кроме того, при очень длинном выравнивании процесс выполнения одновременного обратного отслеживания с границы оценки, например, оцененной позиции фронта волны, до тех пор, пока все обратные

отслеживания не завершатся или не сольются, с последующим сингулярным обратным отслеживанием с выделением символической записи выравнивания можно повторять многократно с различных последовательных границ оценки. Инкрементальная символическая запись выравнивания, например, частичная строка CIGAR, из каждого последовательного инкрементального обратного отслеживания может быть затем наращена на накопленные предыдущие символические записи выравнивания, если новое одновременное обратное наращивание или сингулярное обратное наращивание не завершится раньше, и в этом случае накопленные предыдущие символические записи выравнивания можно отбросить. Аналогичным образом возможное завершающее обратное отслеживание наращивает свою символическую запись на самые последние накопленные символические записи выравнивания для полного описания обратного отслеживания, например, строки CIGAR.

[00229] Соответственно, при таком способе память для хранения векторов оценки может оставаться ограниченной в предположении, что одновременные обратные отслеживания всегда сливаются за ограниченное количество шагов, например 512 шагов. В редких случаях, когда одновременным обратным отслеживаниям не удается слиться или завершиться за ограниченное количество шагов, могут быть предприняты различные исключительные действия, в том числе прекращение текущего выравнивания вследствие сбоя или повтор его с более высоким ограничением или без ограничения, возможно, другим или традиционным способом, таким как сохранение всех векторов оценки для полного выравнивания, например, во внешнем DRAM. В качестве варианта, возможно, будет целесообразно прекратить такое выравнивание вследствие сбоя, поскольку это случается крайне редко, и даже еще реже такое неудавшееся выравнивание будет иметь выравнивание с лучшей оценкой, которое будет включено в отчет о выравнивании.

[00230] В необязательном варианте хранилище векторов оценки может быть разделено, физически или логически, на ряд различных блоков, например, по 512 строк каждый, и конечную строку в каждом блоке можно будет использовать в качестве границы оценки для начала одновременного обратного отслеживания. Необязательно, может потребоваться, чтобы одновременное обратное отслеживание завершилось или слилось в пределах одного блока, например, за 512 шагов. Необязательно, если одновременные обратные отслеживания сливаются за меньшее количество шагов, совместное обратное отслеживание, тем не менее, может быть продолжено по всему блоку, прежде чем приступать к выделению сингулярного обратного отслеживания в предыдущем блоке. Соответственно, после того, как векторы оценки полностью записаны в блок N и начинается запись в блок N+1, можно начать одновременное обратное отслеживание в блоке N, а затем сингулярное обратное отслеживание и выделение символической записи выравнивания в блоке N-1. Если скорость одновременного обратного отслеживания, сингулярного обратного отслеживания и оценки выравнивания аналогичны или идентичны, и эти операции могут выполняться одновременно, например, в параллельном аппаратном обеспечении в интегральной схеме, то сингулярное обратное отслеживание в блоке N-1 может происходить одновременно с заполнением векторами оценки блока N+2, а когда наступит пора заполнять блок N+3, блок N-1 может быть освобожден и использован повторно.

[00231] Следовательно, при такой реализации можно использовать минимум 4 блока векторов оценки, сменяющих друг друга по циклу. Поэтому общее хранилище векторов оценки для модуля выравнивателя может быть, например, из 4 блоков по 257×512 битов в каждом, или приблизительно 64 килобайта. В качестве варианта, если текущая

максимальная оценка выравнивания соответствует более раннему блоку, чем текущая позиция фронта волны, этот блок и предыдущий блок могут быть придержаны, а не использованы повторно, чтобы завершающее обратное отслеживание можно было начать из этой позиции, если она останется позицией с максимальной оценкой; придерживание дополнительных 2 блоков таким образом даст, например, как минимум 6 блоков.

[00232] В другом варианте для поддержки перекрывающихся выравниваний, где фронт волны оценки постепенно перемещается из одной матрицы выравнивания в следующую, как описано выше, могут быть использованы дополнительные блоки, например, 1 или 2 дополнительных блока, например, всего 8 блоков, например, приблизительно 128 килобайтов. Соответственно, при циклическом использовании такого ограниченного количества блоков, например, 4 блока или 8 блоков, возможно выравнивание и обратное отслеживание длинных ридов, например, 100000 нуклеотидов, или всей хромосомы, без использования внешней памяти для векторов оценки. Необходимо понимать, что касается вышеизложенного, то хотя в некоторых случаях функция картирования упоминалась в описаниях как, например, сопоставитель, и/или в некоторых случаях функция выравнивания могла упоминаться как, например, выравниватель, эти различные функции могут выполняться одновременно одной и той же архитектурой, которую в данной области техники обычно называют выравнивателем. Соответственно, в различных случаях функция картирования и функция выравнивания, как описано в настоящем документе, могут выполняться общей архитектурой, которую можно понимать как выравниватель, особенно в тех случаях, когда для выполнения функции выравнивания сначала нужно выполнить функцию картирования.

[00233] В различных случаях устройства, системы и способы их использования по настоящему изобретению могут быть выполнены с возможностью осуществления одного или более из выравниваний полного рида с гэпами и/или без гэпов, которые затем могут быть оценены для определения надлежащего выравнивания для ридов в наборе данных. Например, в различных случаях на данных, подлежащих обработке, может быть выполнена процедура выравнивания без гэпов, причем за этой процедурой выравнивания без гэпов могут следовать одно или более выравниваний с гэпами и/или процедура выборочного выравнивания Смита-Ватермана. Например, на первом этапе может быть сформирована цепочка выравнивания без гэпов. Как описано в настоящем документе, такие функции выравнивания без гэпов могут выполняться быстро без необходимости учета гэпов, причем после первого этапа выполнения выравнивания без гэпов может следовать выполнение выравнивания с гэпами.

[00234] Например, можно выполнить функцию выравнивания, чтобы определить, как любая данная нуклеотидная последовательность, например рид, выравнивается на референсную последовательность, без необходимости вставки гэпов в одном или более ридах и/или референсе. Важная часть выполнения такой функции выравнивания состоит в том, чтобы определить, где и какие несовпадения имеются в исследуемой последовательности по сравнению с последовательностью референсного генома. Однако вследствие колоссальной гомологии в геноме человека любая данная нуклеотидная последовательность, теоретически, имеет склонность к совпадению в значительной степени с репрезентативной референсной последовательностью. Имеющиеся несовпадения, скорее всего, будут обусловлены однонуклеотидным полиморфизмом, который относительно легко обнаружить, или они будут вызваны инсерцией или делецией в исследуемой последовательности, которые намного труднее обнаружить.

[00235] Следовательно, при выполнении функции выравнивания в большинстве

случаев исследуемая последовательность имеет тенденцию к совпадению с референсной последовательностью, а если имеется несовпадение вследствие ОНП, это будет легко определено. Следовательно, для выполнения такого анализа не требуется относительно большого объема вычислительной мощности. Однако при наличии инсерций или делеций в исследуемой последовательности по сравнению с референсной последовательностью возникают трудности, поскольку такие инсерции и делеции приводят к гэпам при выравнивании. Такие гэпы требуют более обширной и сложной платформы обработки для правильного определения выравнивания. Тем не менее, поскольку процент инделов будет невелик, нужно будет выполнить относительно небольшой процент протоколов выравнивания с гэпами по сравнению с выполняемыми миллионами выравниваний без гэпов. Поэтому лишь небольшой процент всех функций выравнивания без гэпов приведет к необходимости дальнейшей обработки вследствие наличия индела в последовательности и, следовательно, потребует выравнивания с гэпами.

[00236] Когда в процедуре выравнивания без гэпов указан индел, только эти последовательности пропускаются в движок выравнивания для дальнейшей обработки, например, в движок обработки, выполненный с возможностью осуществления продвинутой функции выравнивания, такой как выравнивание Смита-Ватермана (SWA). Поэтому, поскольку нужно выполнять выравнивание либо без гэпов, либо с гэпами, устройства и системы, описанные в настоящем документе, являются намного более эффективным использованием ресурсов. Более конкретно, в определенных вариантах реализации на данной подборке последовательностей выравнивание может быть выполнено как без гэпов, так и с гэпами, например, одно за другим, затем для каждой последовательности сравнивают результаты и лучший результат выбирают. Такая конфигурация может быть реализована, например, когда желательно повышение точности и имеется повышенное количество времени и ресурсов для выполнения требуемой обработки.

[00237] В частности, в различных случаях первый этап выравнивания может быть выполнен без привлечения функции Смита-Ватермана, требующей интенсивной обработки. Следовательно, множество выравниваний без гэпов могут быть выполнены с меньшими требованиями к ресурсам и меньшими временными затратами, а поскольку требуется меньше ресурсов, на микросхеме нужно выделить меньше пространства для такой обработки. Таким образом, можно выполнить больше обработки с использованием меньшего числа обрабатывающих элементов, требующих меньше времени, поэтому можно выполнить больше выравниваний и достичь более высокой точности. Более конкретно, для реализации выполнения выравниваний Смита-Ватермана нужно выделять меньше ресурсов микросхемы с использованием меньшей площади микросхемы, так как для обрабатывающих элементов, необходимых для выполнения выравниваний без гэпов, не требуется такая большая площадь микросхемы, как для выполнения выравнивания с гэпами. Поскольку требования к ресурсам микросхемы снижаются, можно выполнить больший объем обработки за более короткий период времени, а за счет большего объема обработки, который можно выполнить, можно достичь более высокой точности.

[00238] Соответственно, в таких случаях можно использовать протокол выравнивания без гэпов, например, который должен выполняться соответствующим образом сконфигурированными ресурсами для выравнивания без гэпов. Например, как описано в настоящем документе, в различных вариантах реализации предусмотрен движок обработки выравнивания, например, когда движок обработки выполнен с возможностью приема цифровых сигналов, например, представляющих одно или более ридов геномных

данных, таких как цифровые данные, выражающие одну или более нуклеотидных последовательностей, из электронного источника данных, и картирования и/или выравнивания этих данных на референсную последовательность, например, путем выполнения сначала функции выравнивания без гэпов на этих данных, причем при необходимости за функцией выравнивания без гэпов может последовать функция выравнивания с гэпами, например, путем выполнения протокола выравнивания Смита-Ватермана.

[00239] Поэтому в различных случаях выполняют функцию выравнивания без гэпов на сплошной части рида, например, с помощью выравнивателя без гэпов, и если выравнивание без гэпов проходит от начала до конца, например рид является полным, выравнивание с гэпами не выполняют. Однако, если результаты выравнивания без гэпов указывают на наличие индела, например рид обрезан или является неполным по иным причинам, можно выполнить выравнивание с гэпами. Таким образом, результаты выравнивания без гэпов могут быть использованы для определения необходимости выравнивания с гэпами, например, когда выравнивание без гэпов достигло области гэпа, но не распространилось на всю длину рида, например, когда рид, возможно, обрезан, например, слабо обрезан, и в том месте, где он обрезано, затем может быть выполнено выравнивание с гэпами.

[00240] Поэтому в различных вариантах реализации, основанных на полноте и оценках выравнивания, выравнивание с гэпами применяют только в том случае, если выравнивание без гэпов завершается, будучи обрезанным, например, не проходит от начала до конца. Более конкретно, в различных вариантах реализации лучшая оценка выравнивания без гэпов и/или с гэпами, которую можно определить, может быть оценена и использована в качестве линии отсечки для принятия решения о том, является ли оценка достаточно хорошей для оправдания дальнейшего анализа, например путем выполнения выравнивания с гэпами. Таким образом, полнота выравнивания и его оценка могут быть использованы так, чтобы высокая оценка указывала на то, что выравнивание завершено и, следовательно, не имеет гэпов, а низкая оценка указывает на то, что выравнивание не завершено, и нужно выполнить выравнивание с гэпами. Следовательно, когда достигнута высокая оценка, выравнивание с гэпами не выполняют, а выравнивание с гэпами выполняют только тогда, когда оценка достаточно низкая. Конечно, в различных случаях может быть применен подход с выравниванием перебором, например, когда в архитектуре микросхемы развернут ряд выравнивателей с гэпами и/или без гэпов, чтобы обеспечить возможность выполнения большего числа выравниваний, и таким образом может быть проверено большее количество данных.

[00241] Более конкретно, в различных вариантах реализации каждый движок картирования и/или выравнивания может включать в себя один или более, например, два, модулей выравнивателя Смита-Ватермана. В определенных случаях эти модули могут быть выполнены с возможностью поддержки глобального (от начала до конца) выравнивания без гэпов и/или локального (обрезанного) выравнивания с гэпами, выполнения аффинной оценки гэпов, и могут быть выполнены с возможностью формирования бонусов оценки за отсутствие обрезания на каждом конце. Возможна также поддержка чувствительной к качеству оснований оценки совпадения и несовпадения. Когда в состав входят два модуля выравнивания, например, как часть интегральной схемы, например, каждый выравниватель Смита-Ватермана может быть построен в виде антидиагонального фронта волны ячеек оценки, причем этот фронт волны «перемещается» по виртуальному прямоугольнику выравнивания, оценивая ячейки, через которые он прокатывается.

[00242] Однако в случае длинных ридов фронт волны Смита-Ватермана может быть также выполнен с возможностью поддержки автоматического управления направлением движения фронта волны, чтобы отслеживать лучшее выравнивание через накопленные инделы для обеспечения того, чтобы фронт волны выравнивания и оцениваемые ячейки не вышли из полосы оценки. Логические движки могут быть выполнены с возможностью проверки текущих оценок фронта волны, нахождения максимумом, пометки подмножеств ячеек на пороговом расстоянии ниже максимума и назначения в качестве мишени средней точки между двумя флагами экстремумов в фоновом режиме. В таком случае автоматическое управление направлением движения может быть выполнено с возможностью прохождения по диагонали, когда мишень находится в центре фронта волны, но может быть выполнено с возможностью прохождения прямо по горизонтали или вертикали при необходимости возврата мишени в центр, если она смещается, например вследствие наличия инделов.

[00243] Выходными данными из модуля выравнивания является файл SAM (текст) или BAM (например, двоичная версия файла SAM) вместе с оценкой качества картирования (MAPA), которая отражает достоверность того, что прогнозированное и выровненное местоположение рида относительно референса действительно то самое, откуда получен рид. Соответственно, после того, как определено, где каждый рид картирован, а также определено, где каждый рид выровнен, например, каждому соответствующему риду даны позиция и оценка качества, отражающая вероятность того, что эта позиция является правильным выравниванием, так что нуклеотидная последовательность для ДНК субъекта известна, как и то, как ДНК субъекта отличается от референса (например, определена строка CIGAR), различные риды, представляющие геномную последовательность нуклеиновых кислот субъекта могут быть отсортированы по местоположению в хромосоме, чтобы можно было определить точное местоположение рида на хромосомах. Поэтому в соответствии с некоторыми аспектами настоящее изобретение относится к функции сортировки, например, которая может быть выполнена модулем сортировки, который может быть частью конвейера модулей, например, конвейера, предназначенного для приема необработанных данных рида последовательности, например, в виде геномного образца индивида, и картирования и/или выравнивания этих данных, которые могут быть затем сохранены.

[00244] Более конкретно, после того, как ридам назначены позиции, например, относительно референсного генома, что может включать в себя определение того, какой хромосоме принадлежит рид, и/или его смещения от начала этой хромосомы, риды можно отсортировать по позиции. Сортировка может быть полезна, например, в последующих анализах, так как при помощи ее все риды, которые перекрывают данную позицию в геноме, могут быть сформированы в скопление, чтобы находится друг возле друга, например, после обработки модулем сортировки, в результате чего можно легко определить, согласуются ли большинство ридов с референсным значением или нет. Таким образом, если большинство ридов не согласуются с референсным значением, определение варианта может пометить. Следовательно, сортировка может включать в себя одну или более сортировок ридов, которые выровнены относительно одной и той же позиции, например, одной и той же позиции хромосомы, для создания скопления, чтобы все риды, которые покрывают одно и то же местоположение, были физически сгруппированы вместе; и может также включать в себя анализ ридов в скоплении для определения того, где риды могут указывать фактический вариант в геноме по сравнению с референсным геномом, причем этот вариант можно отличить, например, с помощью согласования скопления, от ошибки, такой как ошибка

считывания машиной или ошибка в методах секвенирования, которая может проявляться малой частью ридов.

[00245] После того, как данные получены, имеются один или более других модулей, с помощью которых можно очистить эти данные. Например, один модуль, который может входить, например, в конвейер анализа последовательностей, такой как для определения геномной последовательности индивида, может быть модулем локального повторного выравнивания. Например, часто трудно определить инсерции и делеции, которые возникают в конце рида. Причина в том, что алгоритм Смита-Ватермана или аналогичный процесс выравнивания испытывает недостаток контекста касательно индела, чтобы можно было выполнить оценку для обнаружения его присутствия. Поэтому фактический индел может быть указан в отчете как один или более ОНП. В таком случае точность прогнозированного местоположения для любого данного рида может быть улучшена за счет выполнения локального повторного выравнивания на картированных, и/или выровненных, и/или сортированных данных рида.

[00246] В таких случаях могут быть использованы конвейеры для помощи в выяснении истинного выравнивания, например, когда рассматриваемая позиция находится в конце любого данного рида, эта же позиция, вероятно, будет посредине некоторого другого рида в данном скоплении. Соответственно, при выполнении локального повторного выравнивания могут быть проанализированы различные риды в скоплении, чтобы определить, указывают ли некоторые риды в скоплении на наличие инсерции или делеции в данной позиции, где другой рид не содержит индела или, скорее, имеет замену в этой позиции, тогда можно вставить индел, например, в референс, где он не присутствует, и можно повторно выровнять риды в локальном скоплении, которое перекрывает эту область, чтобы посмотреть, будет ли при этом достигнута более хорошая совокупная оценка, чем в том случае, когда там не было инсерции и/или делеции. Если улучшение имеется, весь набор ридов в скоплении может быть пересмотрен, и если оценка всего набора улучшилась, то становится понятно, что в этой позиции действительно был индел. Подобным образом можно компенсировать отсутствие достаточного контекста для более точного выравнивания рида в конце хромосомы для любого отдельного рида. Поэтому при выполнении локального повторного выравнивания исследуют одно или более скоплений, где могут находиться один или более инделов, и определяют, можно ли улучшить общую оценку выравнивания путем добавления индела в любую данную позицию.

[00247]Другой модуль, который может входить, например, в конвейер анализа последовательностей, таком как для определения геномной последовательности индивида, может быть модулем маркировки дубликатов. Например, функция маркировки дубликатов может быть выполнена с возможностью компенсации ошибок химии, которые могут возникать во время фазы секвенирования. Например, как описано выше, во время некоторых процедур секвенирования последовательности нуклеиновых кислот прикрепляют к бусинам и строят на этой основе с помощью меченных нуклеотидных оснований. В идеале получится по одному риду на бусину. Однако иногда к одной бусине прикрепляется множество ридов, что приводит к чрезмерному количеству копий прикрепленных ридов. Это явление известно как дупликация ридов.

[00248] После выполнения выравнивания и получения результатов и/или выполнения функции сортировки, локальной повторной сортировки и/или удаления дубликатов на полученных данных можно использовать функцию определения вариантов. Например, типичная функция определения вариантов или ее часть может быть выполнена с возможностью реализации в программной и/или аппаратной конфигурации, например

в интегральной схеме. В частности, определение вариантов представляет собой процесс, который включает в себя позиционирование всех ридов, выровненных относительно данного местоположения на референсе, в группировки таким образом, что все перекрывающиеся области из всех различных выровненных ридов образуют «скопление». Затем скопление ридов, покрывающее данную область референсного генома, анализируют для определения того, какой наиболее вероятный фактический контент пробы ДНК/РНК индивида находится в этой области. Затем это повторяют поэтапно для каждой области в геноме. Из полученного контента формирует список различий, называемых «вариациями» или «вариантами» референсного генома, с указанием для каждого из них соответствующего уровня доверия и других метаданных.

[00249] Наиболее распространенными вариантами являются однонуклеотидные полиморфизмы (ОНП), при которых одно основание отличается от референса. ОНП встречаются примерно 1 раз на 1000 позиций в геноме человека. Следующими по распространенности являются инсерции (вставки в референс) и делеции (пропуски в референсе). Они чаще всего встречаются при малых длинах, но могут быть любой длины. Однако возникают дополнительные трудности, так как вследствие того, что выбор секвенированных сегментов («ридов») происходит случайным образом, некоторые области будут иметь более глубокое покрытие, чем другие. Существуют также множество сложных вариантов, которые включают в себя замены множества оснований и комбинации инделов и замен, которые могут быть восприняты как замены, изменяющие длину. Стандартные определители вариантов на базе программного обеспечения испытывают трудности при выявлении всего этого и имеют различные ограничения на длины вариантов. Более специализированные определители вариантов в программной и/или аппаратной реализации необходимы для выявления более длинных вариаций и многих вариаций экзотических «структурных вариантов», включающих в себя крупные изменения хромосом.

[00250] Однако определение вариантов является сложной процедурой для реализации в программном обеспечении и на порядки величины более сложной для развертывания в аппаратном обеспечении. Чтобы учесть и/или обнаружить эти типы ошибок, стандартные определители вариантов могут выполнить одну или более из следующих задач. Например, они могут начать с набора гипотетических генотипов (содержимого одной или двух хромосом в локусе) и с помощью байесовских расчетов оценить апостериорную вероятность того, что каждый генотип является истинным в свете наблюдаемых подтверждающих данных, и сообщить наиболее вероятный генотип вместе с уровнем доверия. В силу этого определители вариантов могут быть простыми или сложными. Простые определители вариантов рассматривают только столбцы оснований в скоплении выровненных ридов точно в той позиции, где осуществляется определение. Более совершенные варианты определителей представляют собой «определители на основе гаплотипа», которые могут быть выполнены с возможностью учета контекста, например, в окне вокруг выполняемого определения.

[00251] «Гаплотип» - это конкретное содержимое ДНК (нуклеотидная последовательность, список вариантов и т.д.) в одной общей «нити», например, одна или две диплоидные нити в области, и определитель гаплотипов учитывает байесовские импликации того, какие различия связаны появлением в одном и том же риде. Соответственно, протокол определения вариантов, который представлен в настоящем документе, может реализовывать одну или более функций, например, выполняемых определителем гаплотипов Genome Analysis Tool Kit (GATK), и/или с помощью средства Hidden Markov Model (HMM), и/или функции графа де Брейна, например, когда одна

или более из этих функций, обычно выполняемая определителем гаплотипов GATK, и/или средством HMM, и/или функцией графа де Брейна, может быть реализована в программном и/или аппаратном обеспечении.

[00252] Более конкретно, в соответствии с реализациями в данном документе всевозможные разные операции определения вариантов могут быть выполнены с возможностью осуществления в программном или аппаратном обеспечении, и могут включать в себя один или более из следующих этапов. Например, функция определения вариантов может включать в себя выявление активной области, например, для выявления мест, где множество ридов не согласуются с референсом, и для формирования окна вокруг выявленной активной области, чтобы только эта область могла быть выбрана для дальнейшей обработки. Кроме того, возможна локализованная сборка гаплотипа, например, когда для каждой данной активной области все перекрывающиеся риды могут быть собраны в матрицу «графа де Брейна» (De Bruijn graph, DBG) Из этого DBG можно выделять различные маршруты через матрицу, где каждый маршрут образует гаплотип-кандидат, например, гипотезы о том, какая истинная последовательность ДНК может быть на по меньшей мере одной нити. Кроме того, возможно выравнивание гаплотипа, например, когда каждый выделенный гаплотип-кандидат может быть выровнен, например, с помощью алгоритма Смита-Ватермана, обратно на референсный геном, чтобы определить, какие вариации референса он означает. Кроме того, может быть выполнено вычисление правдоподобия рида, например, когда рид может быть проверен относительно каждого гаплотипа или гипотезы, чтобы оценить вероятность наблюдения рида в предположении, что гаплотип был получен из истинной исходной ДНК.

[00253] Что касается этих процессов, вычисление правдоподобия рида обычно будет наиболее ресурсоемкой и времязатратной операцией, часто требующей оценки парной HMM. Кроме того, построение графов де Брейна для каждого скопления ридов вместе со связанными операциями локального или глобального выявления уникальных K-меров, как описано ниже, тоже может быть ресурсоемким и/или времязатратным. Соответственно, в различных вариантах реализации одно или более из различных вычислений, относящихся к выполнению одного или более из этих этапов, могут быть выполнены с возможностью реализации оптимальным образом в программном или аппаратном обеспечении, например, с возможностью осуществления ускоренным образом интегрированной схемой, как описано в настоящем документе.

[00254] Как указано выше, в различных вариантах реализации определитель гаплотипов по данному изобретению, реализованный в программном и/или аппаратном обеспечении или их сочетании, может быть выполнен с возможностью включения в себя одной или более из следующих операций: выявление активной области, локализованная сборка гаплотипа, выравнивание гаплотипа, вычисление правдоподобия рида и/или генотипирование. Например, устройства, системы и/или способы по данному изобретению могут быть выполнены с возможностью осуществления одной или более из операций картирования, выравнивания и/или сортировки на данных, полученных из секвенированной ДНК/РНК субъекта, для формирования картированных, выровненных и/или сортированных данных результатов. Затем эти данные результатов могут быть очищены, например, путем выполнения на них операции удаления дубликатов, и/или эти данные могут быть переданы в один или более специализированных движков обработки определителя гаплотипов для выполнения операции определения вариантов, в том числе одного или более вышеупомянутых этапов, на этих данных результатов, чтобы тем самым сформировать файл определения

вариантов. Поэтому все риды, которые были секвенированы, и/или картированы, и/или выровнены на конкретные позиции в референсном геноме, могут быть подвергнуты дальнейшей обработке, чтобы определить, как определенная последовательность отличается от референсной последовательности в любой данной точке в референсном геноме.

[00255] Соответственно, в различных вариантах реализации устройство, система и/или способ их использования, которые описаны в настоящем документе, могут включать в себя систему определителя варианта или гаплотипа, которая реализована в программной и/или аппаратной конфигурации для выполнения операции выявления активной области на полученных данных результатов. Выявление активной области включает в себя выявление и определение мест, где множество ридов, например в скоплении ридов, не согласуются с референсом, а также включает в себя формирование одного или более окон вокруг отличий («активных областей») таким образом, чтобы область внутри окна могли быть выбрана для дальнейшей обработки. Например, во время этапа картирования и/или выравнивания выявленные риды картируют и/или выравнивают на области в референсном геноме, откуда они, как ожидается, взяты из генетической последовательности субъекта.

[00256] Однако, поскольку секвенирование выполняют таким образом, чтобы создать избыточную выборку секвенированных ридов для любой данной области генома, в любой данной позиции в референсной последовательности можно увидеть скопление из любых и/или всех секвенированных ридов, которые привязываются и выравниваются относительно этой области. Все эти риды, которые выравниваются и/или перекрываются в данной позиции области или скопления, могут быть введены в систему определителя вариантов. Таким образом, любое данный анализируемый рид можно сравнить с референсом в его предполагаемой области перекрытия, и этот рид можно сравнить с референсом, чтобы определить, имеет ли ее последовательность какие-либо отличия от известной последовательности референса. Если рид укладывается на референс без всяких инсерций или делеций и все основания одинаковые, то выравнивание определяют как хорошее.

[00257] Следовательно, любой данный картированный и/или выровненный рид может иметь основания, которые отличаются от референса, например, рид может содержать один или более ОНП, образующих позицию, где основание не совпадает; и/или рид может иметь одну или более инсерций и/или делеций, например, создающих гэп в выравнивании. Соответственно, в любом из этих случаев будут одно или более несовпадений, которые нужно учитывать в дальнейшей обработке. Тем не менее, чтобы сэкономить время и повысить эффективность, такие дальнейшие обработки следует ограничить случаями, где выявленное несовпадение является нетривиальным, например, не вызванным помехой, отличием. При определении значимости несовпадения места, где множество ридов в скоплении не согласуются с референсом, могут быть идентифицированы как активная область, затем вокруг этой активной области можно использовать окно, чтобы выбрать локус расхождения, который после этого может быть подвергнут дальнейшей обработке. Однако отличие должно быть нетривиальным. Это можно определить множеством способов, например, для каждого рассматриваемого локуса можно вычислить вероятно его непринадлежности референсу, например, путем анализа оценки качества на основе совпадений и несовпадений оснований таким образом, что превышение данного порога считается достаточно значимым показателем того, что эти риды не согласуются с референсом в значительной мере.

[00258] Например, если 30 из картированных и/или выровненных ридов укладываются

и/или накладываются с образование скопления в данной позиции в референсе, например в активной области, и лишь 1 или 2 из этих 30 ридов не согласуются с референсом, то можно считать, что минимальный порог для дальнейшей обработки не удовлетворен, а несогласующиеся риды можно игнорировать, принимая во внимание тот факт, что 28 или 29 ридов согласуются. Однако, если не согласуются 3, или 4, или 5, или 10, или более ридов в скоплении, то это несогласование можно считать достаточно статистически значимым для оправдания дальнейшей обработки, и вокруг выявленный областей отличия можно определить активную область. В таком случае можно рассмотреть окно активной области, определяющее основания, окружающие это отличие, чтобы улучшить контекст области, окружающей отличие, и можно предпринять дополнительные этапы обработки, такие как применение гауссовского распределения и суммирование вероятностей непринадлежности референсу, распределенных по всем соседним частям, чтобы дополнительно изучить и обработать эту область для выявления того, нужно ли объявить активную область и, если да, то какие вариации относительно референса действительно присутствуют в этой области, если таковые имеются. Поэтому определение активной области выявляет те области, где может понадобиться дополнительная обработка, чтобы выяснить, действительно ли имеет место вариация, или произошла ошибка.

[00259] В частности, поскольку во многих случаях нежелательно подвергать дальнейшей обработке каждую область в скоплении последовательностей, можно определить активную область, в которой находятся только те области, где может потребоваться дальнейшая обработка, чтобы четко определить, действительно ли это вариация, или произошла ошибка, которая может быть определена, как требующая дальнейшей обработки. И, как указано выше, размер окна активной области может определяться размером предполагаемой вариации. Например, в различных случаях границы активного окна могут меняться от 1 или 2 или от около 10 до 20, или даже от около 25 до около 50, и ли от около 200 до около 300, или от около 500 до около 1000 оснований или более, причем дальнейшая обработка происходит только внутри границ активного окна. Конечно, размер активного окна может быть любой подходящей длины, такой, чтобы она обеспечивала контекст для определения статистической важности отличия.

[00260] Следовательно, если имеются только одно или два изолированных отличия, возможно, потребуется, чтобы активное окно охватывало одно или более из нескольких десятков оснований в активной области, чтобы иметь достаточно контекста для статистического определения наличия действительного варианта. Однако, если имеется кластер или группа отличий, или если имеются инделы, для которых требуется больше контекста, окно можно сделать большего размера. В любом случае может потребоваться проанализировать все без исключения отличия, которые могут быть в кластерах, чтобы проанализировать их все в одной или более активных областях, поскольку это может обеспечить вспомогательную информацию о каждом отдельном отличии и сэкономит время обработки за счет уменьшения количества затрагиваемых активных окон. В различных случаях границы активной области могут быть определены с помощью активных вероятностей, которые удовлетворяют данному порогу, такому как от около 0,00001, или около 0,00001, или около 0,0001 или меньше до около 0,002, или около 0,02, или около 0,2 или больше. И если активная область длиннее данного порога, например, около 300-500 оснований или 1000 оснований или более, то эту область можно разбить на подобласти, например, на подобласти, определяемые локусом с наинизшей оценкой активной вероятности.

[00261] В различных случаях после выявления активной области может быть выполнена процедура локализованной сборки гаплотипа. Например, в каждой активной области все собранные в скопление и/или перекрывающиеся риды могут быть собраны в «граф де Брейна» (DBG). DBG может быть ориентированным графом, основанным на всех ридах, которые перекрыли выбранную активную область, где активная область может быть длиной около 200, или около 300, или около 400, или около 500 оснований или более, и внутри этой активной области нужно определить наличие и/или идентичность вариантов. В различных случаях, как указано выше, активная область может быть удлинена, например, путем включения еще около 100 или около 200 или более оснований в каждом направлении исследуемого локуса, чтобы сформировать удлиненную активную область, например, когда может потребоваться дополнительный окружающий контекст. Соответственно, именно в окне активной области, удлиненном или нет, собирают в скопление все риды, которые имеют части, перекрывающие активную область, например, чтобы создать скопление, выявляют перекрывающиеся части и последовательности ридов направляют в систему определителя гаплотипа и тем самым собирают вместе в виде графа де Брейна, что во многом напоминает складывание мозаики.

[00262] Соответственно, в любом данном активном окне будут риды, которые образуют скопление таким образом, что скопление в целом содержит путь последовательности, с помощью которого перекрывающиеся области различных перекрывающихся ридов в скоплении покрывают всю последовательность внутри активного окна. Таким образом, в любом данном локусе в активной области будет множество ридов, перекрывающих этот локус, хотя и любой данный рид может не простираться на всю активную область. В результате этого различные области различных ридов внутри скопления используют в DBG для определения того, действительно ли имеется вариант для любого данного локуса в последовательности внутри активной области или нет. Поскольку это определение осуществляют именно внутри активного окна, учитываются именно части любого данного рида в пределах границ активного окна, а части за пределами активного окна можно отбросить.

[00263] Как было указано, именно те секции ридов, которые перекрывают референс в пределах активной области, подают в систему DBG. Затем система DBG собирает риды, подобно мозаике, в граф, после чего для каждой позиции в последовательности на основании коллекции перекрывающихся ридов для этой позиции определяют, имеется ли совпадение или несовпадение для любого данного рида, а если несовпадение существует, то какова вероятность этого несовпадения. Например, когда существуют прерывистые места, где сегменты ридов в скоплении перекрывают друг друга, они могут быть выровнены друг с другом на основе их областей совпадения, а объединяя в строку или сшивая вместе совпадающие ридов на основе их точек совпадения, для каждой позиции в пределах этого сегмента можно определить, совпадают ли или не совпадают, и в какой степени, риды в любой данной позиции. Следовательно, если два или более ридов накладываются и тождественно совпадают друг с другом на какое-то время, получается граф с одной строкой; однако, когда как только два или более ридов попадают в точку, где они отличаются, в графе образуется разветвление и получаются две или более расходящихся строк, пока эти два или более ридов снова не совпадут.

[00264] Следовательно, пути через граф часто бывают не прямой линией. Например, когда k-меры рида отличаются от k-меров референса и/или k-меров одного или более перекрывающихся ридов, например в скоплении, в графе образуется «пузырь» в точке отличия, приводящей к двум расходящимся строкам, которые продолжатся вдоль двух

разных путей до тех пор, пока эти две последовательности снова не совпадут. Каждой вершине можно присвоить взвешенную оценку, показывающую, сколько раз соответствующие k-меры перекрываются во всех ридах в скоплении. В частности, каждому пути, проходящему через сформированный граф с одной стороны на другую, может быть назначен счетчик. И когда одни и те же k-меры формируются из множества ридов, например, когда каждый k-мер имеет одну и тужу структуры последовательности, они могут быть учтены в графе путем увеличения счетчика для этого пути, где k-мер перекрывает уже существующий путь k-мера. Таким образом, когда один и тот же k-мер формируется из множества перекрывающихся ридов, имеющих одну и ту же последовательность, структура пути по графу будет повторяться снова и снова, и счетчик для прохождения этого пути через граф будет инкрементально увеличиваться соответствующим образом. В таком случае структуру записывают только для первого экземпляра k-мера и счетчик инкрементально увеличивают для каждого k-мера, который повторяет эту структуру. В этом режиме можно получить различные риды в скоплении, чтобы определить, какие вариации имеют место и где.

[00265] Подобным образом можно сформировать матрицу графа, взяв все возможные k-меры из N оснований, например k-меры из 10 оснований, которые могут быть сформированы из каждого данного рида путем последовательного прохождения по всей длине рида сегментами по десять оснований, где начало каждых новых десяти оснований смещено на одно основание от последнего сформированного сегмента из 10 оснований. Затем эту процедуру можно повторить, сделав то же самое для каждого другого рида в скоплении в пределах активного окна. После этого сформированные k-меры можно выровнять друг с другом таким образом, чтобы области идентичного совпадения между сформированными k-мерами совпадали с областями, где они перекрываются, для построения таким образом структуры данных, например, графа, которую можно затем просканировать и определить процент совпадения и несовпадения. В частности, референс и любые ранее обработанные k-меры, выровненные на него, можно отсканировать на предмет следующего формируемого k-мера, чтобы определить, совпадает и/или перекрывается ли текущий формируемый k-мер с какой-либо частью ранее сформированного k-мера, и там, где обнаруживается совпадение, текущий формируемый k-мер может быть вставлен в граф в соответствующей позиции.

[00266] После того, как граф построен, его можно отсканировать и на основе этого сопоставления можно определить, могут ли любые данные ОНП и/или инделы относительно референса быть действительной вариацией в генетическом коде субъекта, или они являются результатом ошибки обработки или иной ошибки. Например, если все или значительная часть k-меров всех или значительной части ридов в данной области содержат одно и то же несовпадение в виде ОНП и/или индела, но отличаются от референса одинаково, то можно определить, что в геноме субъекта действительно существует вариация в виде ОНП и/или индела по сравнению с референсным геномом. Однако, если только ограниченное количество k-меров из ограниченного количества ридов проявляют артефакт, это, вероятно, вызвано ошибкой машины, и/или обработки, и/или другой ошибкой и не свидетельствует о наличии истинной вариации в этой исследуемой позиции,

[00267] Как было указано, там, где есть подозрение на вариацию, в графе будет сформирован пузырь. А именно, там, где все k-меры во всей данной области ридов соответствуют референсу, они выстроятся таким образом, что образуют линейный граф. Однако там, где имеется отличие между основаниями в данном локусе, в этом локусе с отличием граф разветвится. Этот разветвление может быть в любой позиции

в пределах k-мера, и, следовательно, в этой точке отличия k-мер из 10 оснований, включающий в себя это отличие, отклонится от остальных k-меров в графе. В таком случае будет сформирован новый узел, формирующий другой путь через граф.

[00268] Следовательно, там, где все может быть согласовано, например, последовательность в данном новом k-мере наносимая на граф, совпадает с последовательностью, на которую он выравнивается в графе вплоть до точки отличия, путь для этого k-мера будет соответствовать пути для графа в целом и будет линейным, но после точки несходства появится новый путь через граф для приведения в соответствие с отличием, представленным в последовательности k-мера, вновь нанесенного на граф. Это расхождение будет представлено новым узлом в графе. В таком случае любые новые k-меры, подлежащие добавлению к графу, который совпадает с вновь расходящимся путем, увеличат счетчик в этом узле. Следовательно, для каждого рида, которое поддерживает ребро, счетчик будет увеличен инкрементально.

[00269] В различных таких случаях k-мер и/или рид, который он представляет, когда-нибудь начнет снова совпадать, например, после точки расхождения, так что теперь появится точка схождения, где k-мер начинает совпадать с главным путем через граф, представленный k-мерами референсной последовательности. Например, естественно, через некоторое время риды, которые поддерживают разветвленный узел, должны вновь присоединиться к графу со временем. Таким образом, со временем k-меры для данного рида снова присоединятся к главному пути. Более конкретно, в случае ОНП в данном локусе в пределах рида k-мер, начинающийся в этом ОНП, отклонится от главного пути и будет оставаться отделенным примерно в течение 10 узлов, так как существуют 10 оснований в k-мере, которые перекрывают этот локус несовпадения между ридом и референсом. Таким образом, в случае ОНП в 11-й позиции k-меры, покрывающие этот локус в пределах рида, вновь присоединятся к главному пути, так как точное совпадение возобновится. Следовательно, для k-меров рида, имеющего ОНП в данном локусе, потребуется десять сдвигов, чтобы вновь присоединиться к главному графу, представленному референсной последовательностью.

[00270] Как указано выше, обычно существует один главный путь, или линия, или остов, т.е., референсный путь, и когда имеется расхождение, на узле, где существует отличие между ридом и остовным графом, образуется пузырь. Таким образом, существуют некоторые риды, которые отклоняются от остова и образуют пузырь, причем этот расхождение может указывать на наличие варианта. По мере обработки графа вдоль референсного остова могут формироваться пузыри в пузырях внутри пузырей, так что они образуют стопку, и можно создать множество путей через граф. В таком случае могут существовать главный путь, представленный референсным остовом, один путь первого расхождения и еще один путь второго расхождения в пределах первого расхождения, все в пределах данного окна, причем каждый путь через граф может представлять действительную вариацию или может быть артефактом, например, вызванным ошибкой секвенирования, и/или ошибкой ПЦР, и/или ошибкой обработки и т.п.

[00271] После того, как такой граф построен, необходимо определить, какие пути через граф представляют действительные вариации в пределах генома образца, а какие являются всего лишь артефактами. Хотя и ожидается, что риды, содержащие ошибки обработки или машины, не будут поддерживаться большинством ридов в скоплении образца, тем не менее, это не всегда так. Например, ошибки в обработке ПЦР могут, как правило, быть результатом ошибки клонирования, которое происходит во время приготовления пробы ДНК, причем такие ошибки обычно приводят к добавлению

инсерции и/или делеции в клонированную последовательность. Такие ошибки-инделы могут быть более согласованными между ридами и могут накручиваться при формировании множества ридов, которые имеют одну и ту же ошибку вследствие данной ошибки при ПЦР-клонировании. Следовательно, такие ошибки могут привести к более высокой линии подсчета для такой точки.

[00272] Таким образом, после того как сформирована матрица графа с множеством путей через граф, следующая стадия заключается в прохождении, и тем самым выделении, всех путей через граф, например, слева направо. Один путь будет референсным остовом, но будут другие пути, которые следуют различным пузырям вдоль пути. Необходимо пройти все пути и занести их счетчики в таблицу. Например, если граф содержит путь с двухуровневым пузырем в одном месте и трехуровневый пузырь в другом месте, через этот граф будет $(2\times3)^6$ путей. Поэтому каждый из путей нужно выделить по отдельности, а выделенные пути называют гаплотипами-кандидатами. Такие гаплотипы-кандидаты представляют гипотезы в отношении того, что могло бы реально представлять действительную ДНК субъекта, которая была секвенирована, и для проверки этих гипотез могут быть использованы последующие этапы обработки, включая одно или более из выравнивания гаплотипа, вычисления правдоподобия рида и/или генотипирования, чтобы найти вероятности того, что какая-либо и/или каждая из этих гипотез верная. Следовательно, реализация реконструкции графа де Брейна представляет способ надежного выделения хорошего набора гипотез для проверки.

[00273] Например, при выполнении функции определения вариантов, как описано в настоящем документе, может быть реализована операция выявления активной области, например, для выявления мест, где множество ридов в скоплении в пределах области не согласуются с референсом, и для формирования окна вокруг выявленной активной области, чтобы только эта область моглы быть выбрана для дальнейшей обработки. Кроме того, локализованная сборка гаплотипа может иметь место, например, когда для каждой данной активной области все перекрывающиеся риды в скоплении могут быть собраны в матрицу «графа де Брейна» (DBG) Из этого DBG можно выделять различные маршруты через матрицу, где каждый маршрут образует гаплотип-кандидат, например, гипотезы того, для истинная последовательность ДНК может быть на по меньшей мере одной нити.

[00274] Кроме того, возможно выравнивание гаплотипа, например, когда каждый выделенный гаплотип-кандидат может быть выровнен, например, с помощью алгоритма Смита-Ватермана, обратно относительно референсного генома, чтобы определить, какие вариации референса он означает. Кроме того, может быть выполнено вычисление правдоподобия рида, например, когда рид может быть проверен относительно каждого гаплотипа, чтобы оценить вероятность наблюдения рида, предполагающего, что гаплотип был получен из истинной исходной ДНК. Наконец, может быть реализована операция генотипирования и создан файл нахождения вариантов. Как указано выше, любая или все из этих операций могут быть выполнены с возможностью реализации оптимизированным образом в программном и/или аппаратном обеспечении, и в различных случаях в виду ресурсоемкого и времязатратного характера построения матрицы DNG и выделение из нее гаплотипов-кандидатов и/или в виду ресурсоемкого и времязатратного характера выполнения выравнивания гаплотипа и/или вычисление правдоподобия рида, которое может включать в себя использование оценки скрытой марковской модели HMM), эти операции (например, локализованная сборка гаплотипа, и/или выравнивание гаплотипа, и/или вычисление правдоподобия рида) или их части

могут быть выполнены с возможностью реализации одной или более функций этих операций в аппаратном виде, например, выполнения их ускоренным образом интегральной схемой, как описано в настоящем документе. В различных случаях эти задачи могут быть выполнены с возможностью реализации одной или более квантовыми схемами, такими как квантовое вычислительное устройство.

[00275] Соответственно, в различных случаях устройства, системы и способы их использования могут быть выполнены с возможностью осуществления выравнивания гаплотипа и/или вычисления правдоподобия рида. Например, как было указано, каждый выделенный гаплотип может быть выровнен, например, с помощью алгоритма Смита-Ватермана, обратно относительно референсного генома, чтобы определить, какие вариации референса он означает. В различных примерах случаев может выполняться оценка, например, в соответствии со следующими примерами параметров оценки: соответствие = 20,0; несоответствие = -15,0; открытие гэпа = -26,0; и продление гэпа = -1,1; могут быть использованы другие параметры оценки. Соответственно, таким образом можно сформировать строку CIGAR и связать с гаплотипом, чтобы создать собранный гаплотип, причем собранный гаплотип может быть в конечном счете использован для выявления вариантов. Соответственно, подобным образом можно вычислить правдоподобие данного рида, связываемого с данным гаплотипом, для всех комбинаций рид/гаплотип. В таких случаях правдоподобие можно вычислить с помощью скрытой марковской модели (HMM).

[00276] Например, различные собранные гаплотипы могут быть выровнены в соответствии с моделью динамического программирования, аналогично выравниванию Смита-Ватермана. В таком случае можно сформировать виртуальную матрицу, например, где гаплотип-кандидат, например, сформированный с помощью DBG, может быть расположен на одной оси виртуального массива, а рид может быть расположен на другой оси. Затем можно заполнить матрицу оценками, формируемыми путем прохождения выделенных путей через граф и вычисления вероятностей того, что данный путь является истинным путем. Следовательно, в таком случае отличие данного протокола выравнивания от типичного протокола выравнивания Смита-Ватермана состоит в том, что для отыскания наиболее вероятного пути через массив используют вычисление максимального правдоподобия, такое как вычисление, выполняемое с помощью модели HMM, которая выполнена таким образом, чтобы обеспечивать полную вероятность для выравнивания ридов с гаплотипом. Поэтому выполнять реальное выравнивание строки CIGAR в этом случае не требуется. Вместо этого рассматривают все возможные выравнивания и суммируют их вероятности. Оценка парной HMM является ресурсоемкой и времязатратной, и поэтому реализация ее операций в аппаратной конфигурации в интегральной схеме или с помощью квантовых схем на квантовой вычислительной платформе имеет большое преимущество.

[00277] Например, можно проверить каждый рид с каждым потенциальным гаплотипом, чтобы оценить вероятность наблюдения рида в предположении, что гаплотип является истинным представлением исходного образца ДНК. В различных случаях это вычисление может быть выполнено путем оценки «парной скрытой марковской модели» (HMM), которая может быть выполнена с возможностью моделирования различных возможных способов, которыми мог быть изменен гаплотип-кандидат, таких как ошибки ПЦР или секвенирования и т п., и вариация, внесенная в наблюдаемый рид. В таких случаях оценка HMM может быть выполнена с использованием метода динамического программирования для вычисления полной вероятности любой последовательности переходов между марковскими состояниями,

достигающих наблюдаемого рида в виде возможности того, что любое расхождение в риде может быть результатом модели ошибок. Соответственно, такие вычисления HMM могут быть выполнены с возможностью анализа всех возможных ОНП и инделов, которые могли быть введены в одно или более ридов, например, за счет артефактов усиления и/или секвенирования.

[00278] В частности, парная HMM учитывает в виртуальной матрице все возможные выравнивания рида с референсными гаплотипами-кандидатами наряду с вероятностью, связанной с каждым из них, где все вероятности суммируют. Чтобы получить одну общую вероятность для каждого рида, суммируют все вероятности всех вариантов вдоль данного пути. Затем этот процесс выполняют для каждой пары гаплотипа и рида. Например, если имеется кластер из шести скоплений, перекрывающих данную область, например, область из шести гаплотипов-кандидатов, и если скопление содержит около ста ридов, потребуется выполнить 600 операций HMM. Более конкретно, если имеются 6 гаплотипов, то на протяжении пути должны быть 6 ветвей, и необходимо вычислить вероятность того, что каждая из них является правильным путем, который соответствует действительному генетическому коду субъекта для данной области. Следовательно, необходимо учитывать каждый путь для всех ридов, и нужно вычислить вероятность для каждого рида, что вы прибудете на этот данный гаплотип.

[00279] Парная скрытая марковская модель является приблизительной моделью того, как истинный гаплотип в образце ДНК может превратиться в возможный отличающийся обнаруживаемый рид. Было замечено, что эти типы преобразований являются комбинацией ОНП и инделов, которые были введены в набор генетического образца процессом ПЦР, одним или более других этапов приготовления образца и/или ошибкой, вызванной процессом секвенирования и т.п. Как показано на ФИГ. 2, чтобы учесть эти типы ошибок, можно использовать основополагающую базовую модель из 3 состояний, где: (M = совпадение выравнивания, I = инсерция, D = делеция), и где возможен любой переход, кроме I <-> D.

[00280] Как показано на ФИГ. 2, содержащая переходы базовой модели из 3 состояний совершаются не во временной последовательности, а, скорее, в последовательности продвижения по последовательностям гаплотипа -кандидата и рида, начиная с позиции 0 в каждой последовательности, где первое основание находится в позиции 1. Переход к M означает позицию +1 в обеих последовательностях; переход к I означает позицию +1 только последовательности рида; а переход к D означает позицию +1 только в последовательности гаплотипа. Такую же модель из 3 состояний можно также сконфигурировать в качестве основы для алгоритмов Смита-Ватермана и/или Нидлмана-Вунша, которые описаны в настоящем документе. Соответственно, такая модель из 3 состояний, которая изложена в настоящем документе, может быть использована в процессе Смита-Ватермана и/или Нидлмана-Вунша для обеспечения тем самым аффинной оценки гэпа (индела), в которой предполагается, что открытие гэпа (введение I или D) будет менее вероятно, что продление гэпа (продолжение пребывания в состоянии I или D). Таким образом, в данном случае парную HMM можно рассматривать как выравнивание, и можно создать строку CIGAR для кодирования последовательности различных переходов между состояниями.

[00281] В различных случаях базовая модель из 3 состояний может быть усложнена путем разрешения изменения вероятностей в зависимости от позиции. Например, вероятности всех M-переходов могут быть умножены не априорные вероятности наблюдения следующего основания рида, определяемые его оценкой качества основания и соответствующим следующим основанием гаплотипа. В таком случае оценки качества

оснований могут быть преобразованы в вероятность ошибки ОНП при секвенировании. Когда два основания совпадают, в качестве априорной вероятности берут единицу минус вероятность этой ошибки, а когда они не совпадают, берут вероятность ошибки, деленную на 3, так как существуют 3 возможных результата ОНП.

[00282] Выше рассмотрена абстрактная «марковская» модель. В различных случаях может быть также определена последовательность переходов с максимальным правдоподобием, которая в настоящем документе называется выравниванием, и может быть выполнена с помощью алгоритма Нидлмана-Вунша или другого алгоритма динамического программирования. Но, в различных вариантах, при выполнении функции определения вариантов, как описано в настоящем документе, выравнивание методом максимального правдоподобия или любое конкретное выравнивание не должно иметь первостепенного значения. Скорее, можно вычислить полную вероятность, например, путем расчета полной вероятности наблюдения рида данного гаплотипа, которая является суммой вероятностей всех возможных путей перехода через граф от нулевой позиции рида в любой позиции гаплотипа до конечной позиции рида в любой позиции гаплотипа, где вероятность каждой составляющей пути является просто произведением вероятностей различных составляющих переходов.

[00283] Нахождение суммы вероятностей пути может быть также выполнено с помощью виртуального массива и алгоритма динамического программирования, как описано выше, например, в каждой ячейке матрицы (0 … N)×(0 … M) существуют три вычисленных значения вероятности, соответствующие переходам в состояния M, D и I (или, что то же, имеются 3 матрицы). Верхняя строка (нулевая позиция рида) матрицы может быть инициализирована вероятностью 1,0 в состояниях D и вероятностью 0,0 в состояниях I и M; а остальная часть левого столбца (нулевая позиция гаплотипа) может быть инициализирована всеми нулями. (В программном обеспечении начальные вероятности D могут быть установлены близкими к максимальному значению двойной точности, например, 2^1020, чтобы избежать обращения в машинный нуль, но этот фактор может быть нормализован позже.)

[00284] Эта зависимость вычисления «3 к 1» ограничивает порядок, в котором могут быть вычислены ячейки. Они могут быть вычислены слева направо в каждой строке с переходом по строкам сверху вниз или сверху вниз в каждом столбце с переходом вправо. Кроме того, они могут быть вычислены в антидиагональных фронтах волны, где следующий шаг заключается в вычислении всех ячеек (n, m), а n+m равно приращенному номеру шага. Преимущество этого порядка фронта волны состоит в том, что все ячейки на антидиагонали могут быть вычислены независимо друг от друга. Тогда нижняя строка матрицы в конечной позиции рида может быть выполнена с возможностью представления вычисленных выравниваний. В таком случае определитель гаплотипа будет действовать путем суммирования вероятностей I и M всех ячеек нижней строки. В различных вариантах реализации система может быть настроена так, чтобы в нижней строке были запрещены переходы D, или там можно использовать вероятность 0,0 перехода D, чтобы избежать двойного подсчета.

[00285] Как описано в настоящем документе, в различных случаях каждая оценка HMM может действовать на паре последовательностей, например, на паре из гаплотипа-кандидата и рида. Например, в пределах данной активной области каждый набор гаплотипов может быть оценен с помощью HMM относительно набора ридов. В таком случае полоса пропускания на входе программного или аппаратного обеспечения может быть уменьшена и/или сведена к минимуму за счет однократной передачи набора ридов и набора гаплотипов и предоставления программному и/или аппаратному

обеспечению возможности формировать N×M пар операций. В определенных случаях средство оценки Смита-Ватермана может быть выполнено с возможностью выстраивания в очередь отдельных операций HMM, каждая со своей собственной копией данных рида и гаплотипа. Модуль выравнивания Смита-Ватермана (SW) может быть выполнен с возможностью выполнения вычисления парной HMM в линейном пространстве или может действовать в логарифмическом вероятностном пространстве. Это полезно для сохранения точности по всему огромному диапазону значений вероятности, представленных числами с фиксированной запятой. Однако в других случаях могут быть использованы операции с плавающей запятой.

[00286] Имеются три параллельных умножений (например, суммирований в логарифмическом пространстве), затем два последовательных суммирования (конвейеры ~5-6-ступенчатой аппроксимации), затем еще одно умножение. В таком случае полный конвейер может быть длиной около L = 12-16 циклов. Вычисления I & D могут быть примерно половинной длины. В конвейер может подаваться множество входных вероятностей, например, 2, или 3, или 5, или 7, или более входных вероятностей в каждом цикле, например, из одной или более уже вычисленных соседних ячеек (M и/или D слева, M и/или I сверху, и/или M и/или I и/или D сверху слева). Он может также содержать в каждом цикле одно или более оснований гаплотипа и/или одно или более оснований ридов, например, вместе со связанными параметрами, такими как предварительно обработанные параметры, Он выводит итоговый набор M & I & D для одной ячейки за каждый цикл после сквозной задержки.

[00287] Как указано выше, при выполнении функции определения вариантов, как описано в настоящем документе, может быть составлен граф де Брейна, и когда все риды в скоплении идентичны, DBG будет линейным. Однако, когда имеются отличия, в графе будут формироваться «пузыри», которые указывают на области отличий, приводящие к множеству путей, расходящихся от совпадения при выравнивании с референсом, и затем позже снова соединяющихся в совпадении при выравнивании. Из этого DBG можно выделить различные пути, которые образуют гаплотипы-кандидаты, например, гипотезы относительно того, какой может быть истинная последовательность ДНК на по меньшей мере одной нити, причем эти гипотезы могут быть проверены путем выполнения операции HMM, или модифицированной HMM, на этих данных. Более того, можно использовать функцию генотипирования, например, когда могут быть сформированы диплоидные комбинации гаплотипов-кандидатов, и для каждой из них может быть вычислена условная вероятность наблюдения всего скопления ридов. Затем эти результаты можно ввести в модуль байесовской формулы для вычисления безусловной вероятности того, что каждый генотип является истинным, при условии, что наблюдается все скопление ридов.

[00288] Следовательно, в соответствии с устройствами, системами и способами их использования, описанными в настоящем документе, в различных случаях может быть выполнена операция генотипирования, причем эта операция генотипирования может быть выполнена с возможностью реализации оптимизированным образом в программном обеспечении, и/или аппаратном обеспечении, и/или квантовым процессорным устройством. Например, можно сформировать возможные диплоидные комбинации гаплотипов-кандидатов и для каждой комбинации вычислить условную вероятность наблюдения всего скопления ридов, например, с помощью составных вероятностей наблюдения каждого данного рида каждого гаплотипа на основе оценки парной HMM. Результаты этих вычислений подают в модуль байесовской формулы для вычисления безусловной вероятности того, что каждый генотип является истинным,

при условии, что наблюдается все скопление ридов.

[00289] Соответственно, согласно различным аспектам настоящее изобретение относится к системе для выполнения операции определения гаплотипов или вариантов на сформированных и/или предоставленных данных с целью создания тем самым файла определения вариантов. А именно, как описано выше, в отдельных случаях файл определения вариантов может быть цифровым или иным подобным файлом, который кодирует отличия между одной и другой последовательностью, например, разницу между последовательностью образца и референсной последовательностью. В частности, в различных случаях файл определения вариантов может быть текстовым файлом, который формулирует или иным образом детально излагает генетические и/или структурные вариации в генетическом строении индивида по сравнению с одним или более референсными геномами.

[00290] Например, гаплотип представляет собой набор генетических, например, ДНК и/или РНК, вариаций, таких как полиморфизм, который присутствует в хромосомах человека и в силу этого может быть передан потомку и тем самым унаследован. В частности, гаплотип может означать комбинацию аллелей, например, одну из множества альтернативных форм гена таких, которые могут возникать в результате мутации, причем аллельные вариации обычно обнаруживают в одном и том же месте хромосомы. Следовательно, при определении идентичности генома человека важно знать, какую форму из множества различных возможных аллелей кодирует генетическая последовательность определенного человека. В конкретных случаях гаплотип может означать один или более, например, набор, нуклеотидных полиморфизмов (например, ОНП), которые могут быть найдены в одной и той же позиции на одной и той же хромосоме.

[00291] Как правило, в различных вариантах реализации для определения генотипа например, аллельных гаплотипов, для субъекта, как описано выше в настоящем документе, можно прибегнуть к основанному на программном обеспечении алгоритму, такому как алгоритм, использующий программу определения гаплотипов, например, GATK, для одновременного определения ОНП, и/или инсерций, и/или делеций, т.е. инделов, в генетической последовательности индивида. В частности, алгоритм может включать в себя один или более протоколов сборки гаплотипа, например, для локальной сборки гаплотипа с самого начала в одной или более активных областях обрабатываемой генетической последовательности. Такая обработка обычно подразумевает развертывание функции обработки, называемой скрытой марковской моделью (HMM), которая представляет собой стохастическую и/или статистическую модель, используемую для воплощения случайно изменяющихся систем, например, в предположении, что будущие состояния в системе зависят только от настоящего состояния, а не от ряда предшествующих ему событий.

[00292] В таких случаях моделируемая система обладает характеристиками или иным образом предполагается марковским процессом с ненаблюдаемыми (скрытыми) состояниями. В конкретных случаях модель может содержать простую динамическую байесовскую сеть. В частности, что касается определения генетической вариации в ее простейшей форме, существует одна из четырех возможностей идентификации любого данного основания в обрабатываемой последовательности, например, при сравнении сегмента референсной последовательности, например, гипотетического гаплотипа, и сегмента последовательности ДНК или РНК субъекта, например рида, полученного из секвенатора. Однако, чтобы определить такую вариацию, в первую очередь нужно секвенировать ДНК/РНК субъекта, например, с помощью секвенатора нового поколения

(СНП), чтобы создать считывание или «риды», которые определяют генетический код субъекта. Далее, после того, как геном субъекта секвенирован для получения одного или более ридов, различные представления, представляющие ДНК и/или РНК субъекта, необходимо картировать и/или выровнять, как более подробно описано выше в настоящем документе. Затем следующий этап обработки заключается в определении того, как гены субъекта, которые только что определены, например, картированы и/или выровнены, отличаются от генов прототипной референсной последовательности. Поэтому при выполнении такого анализа предполагается, что рид, потенциально представляющий данный ген субъекта, является представлением прототипного гаплотипа, хотя и с различными ОНП и/или инделами, которые были определены к данному времени.

[00293] А именно, в соответствии с конкретными аспектами предложены устройства, системы и/или способы их практического применения, например, для выполнения функции определения гаплотипов и/или вариантов, например, развертывания функции HMM, например, в ускоренном определителе гаплотипов. В различных случаях для преодоления этих и других подобных проблем, известных в данной области техники, представленный в настоящем документе ускоритель HMM может быть выполнен с возможностью выполнения операций таким образом, чтобы его можно было реализовать в программном обеспечении, реализовать в аппаратном обеспечении, или чтобы его реализация и/или управление иным образом осуществлялись комбинированно, частично программным обеспечением и/или частично аппаратным обеспечением, и/или с возможностью использования реализаций квантовых вычислений. Например, в соответствии с конкретным аспектом, изобретение относится к способу, с помощью которого можно определять данные, относящиеся к идентичности последовательности ДНК и/или РНК субъекта, и/или как генетическая информация субъекта может отличаться от референсного генома.

[00294] В таком случае способ может быть осуществлен путем реализации функции определения гаплотипов или вариантов, например, использующей протокол HMM. В частности, функция HMM может быть выполнена в программном обеспечении, аппаратном обеспечении или посредством одной или более квантовых схем, например, на ускоренном устройстве, в соответствии со способом, описанным в настоящем документе. В таком случае ускоритель HMM может быть выполнен с возможностью приема и обработки секвенированных, картированных и/или выровненных данных для обработки их, например, с целью создания файла определения вариантов, а также для передачи обработанных данных обратно по всей системе. Соответственно, способ может включать в себя развертывание системы, в которой данные могут быть отправлены из процессора, такого как управляемый программным обеспечением ЦПУ или ГПУ или даже КПУ, в определитель гаплотипов, реализующий ускоренную HMM, причем этот определитель гаплотипов может быть развернут на микропроцессорной микросхеме, такой как FPGA, ASIC или структурированная ASIC, или может быть реализован с помощью одной или более квантовых схем. Способ может также включать в себя этапы обработки данных для получения результирующих данных HMM, причем эти результаты могут быть потом возвращены обратно в ЦПУ, и/или ГПУ, и/или КПУ.

[00295] В частности, в одном варианте реализации, как показано на ФИГ. 3А, предложена конвейерная система для биоинформатики, содержащая ускоритель HMM. Например, в одном случае конвейерная система для биоинформатики может быть выполнена в виде системы 1 определения вариантов. Система на фигуре изображена в аппаратной реализации, но может быть также реализована посредством одной или

более квантовых схем, таких как квантовая вычислительная платформа. А именно, на ФИГ. 3А приведено высокоуровневое представление структуры интерфейса HMM. В конкретных вариантах реализации система 1 определения вариантов выполнена с возможностью ускорения по меньшей мере части операции определения вариантов, такой как операция HMM. Поэтому в настоящем документе в различных случаях система определения вариантов может упоминаться как система 1 HMM. Система 1 содержит сервер, имеющий одно или более процессорных устройств 1000 (ЦПУ/ГПУ/КПУ), выполненных с возможностью осуществления одной или более стандартных программ, относящихся к секвенированию и/или обработке генетической информации, например, для сравнения секвенированной генетической последовательности с одной или более референсных последовательностей.

[00296] Кроме того, система 1 содержит периферийное устройство 2, такое как плата расширения, которое содержит микросхему 7, например FPGA, ASIC или sASIC. В некоторых случаях могут быть предусмотрены одна или более квантовых схем, выполненных с возможностью осуществления различных операций, указанных в настоящем документе. Необходимо также отметить, что термин ASIC может в равной степени означать структурированную ASIC (sASIC), когда это уместно. Периферийное устройство 2 содержит межсоединение 3 и интерфейс 4 шины, например, параллельной или последовательной шины, который соединяет ЦПУ/ГПУ/КПУ 1000 с микросхемой 7. Например, устройство 2 может содержать межсоединение периферийных компонентов, такое как PCI, PCI-X, PCIe или QPI (межсоединение быстрого доступа), и может содержать интерфейс 4 шины, который выполнен с возможностью функционального и/или пригодного для обмена данными соединения ЦПУ/ГПУ/КПУ 1000 с устройством 2, например для высокоскоростной передачи данных с низкой задержкой. Соответственно, в конкретных случаях интерфейс может представлять собой интерфейс 4 межсоединения периферийных компонентов типа экспресс (PCIe), который связан с микросхемой 7, причем эта микросхема содержит ускоритель 8 HMM. Например, в конкретных случаях ускоритель 8 HMM выполнен с возможностью осуществления функции HMM, например, когда функция HMM в определенных вариантах реализации может быть по меньшей мере частично реализована в аппаратном обеспечении FPGA, AISC или sASIC или посредством одной или более соответствующим образом сконфигурированных квантовых схем.

[00297] В частности, на ФИГ. 3А представлено высокоуровневое изображение ускорителя 8 HMM с приведенной в качестве примера организацией одного или более движков 13, таких как множество движков 13a-$13_{m+1}$, для осуществления одного или более процессов функции определения вариантов, например, включая задачу HMM. Соответственно, ускоритель 8 HMM, который может состоять из распределителя 9 данных, например, CentCom, и одного или множества кластеров 11-$11_{n+1}$ обработки, которое могут быть организованы как, или иным образом включать в себя, один или более экземпляров 13, например, когда каждый экземпляр может быть выполнен в виде движков обработки, такого как малогабаритный движок 13a-$13_{m+1}$. Например, распределитель 9 может быть выполнен с возможностью приема данных, например, из ЦПУ/ГПУ/КПУ 1000, и распределения или передачи иным образом этих данных в один или более из множества кластеров 11 обработки HMM.

[00298] В частности, в определенных вариантах реализации распределитель 9 может быть расположен логически между встроенным интерфейсом 4 PCIe и модулем 8 ускорителя HMM, например, когда интерфейс 4 обменивается данными с

распределителем 9, например, по межсоединению или иной соответствующим образом сконфигурированной шине 5, например, по шине PCIe. Модуль распределителя 9 может быть выполнен с возможностью обмена данными с одним или более кластерами 11 ускорителя НММ, например по одной или более кластерным шинам 10. Например модуль 8 ускорителя НММ может быть выполнен в виде, или иным образом включать в себя, массив кластеров 11a-$11_{n+1}$, например, когда каждый кластер 11 НММ может быть выполнен в виде или иным образом включать в себя концентратор 11 кластеров и/или может включать в себя один или более экземпляров 13, причем экземпляр может представлять собой движок 13 обработки, который выполнен с возможностью осуществления одной или более операций на данных, принимаемых им. Соответственно, в различных вариантах реализации каждый кластер 11 может быть сформирован как, или иным образом включать в себя, концентратор 11a-$11_{n+1}$, где каждый из концентраторов может быть выполнен с возможностью функциональной связи с множеством экземпляров 13a-$13_{m+1}$ движка ускорителя НММ, например когда каждый концентратор 11 кластеров может быть выполнен с возможностью направления данных во множество движков 13a-$13_{m+1}$ обработки в пределах кластера 11.

[00299] В различных случаях ускоритель 8 НММ выполнен с возможностью сравнения каждого основания секвенированного генетического кода субъекта, например, в формате рида, с различными известными или сформированными гаплотипами-кандидатами референсной последовательности и определения вероятности того, что любое данное основание в изучаемой позиции либо совпадает, либо не совпадает с соответствующим гаплотипом, например, рид содержит ОНП, инсерцию или делецию, что приводит к вариации основания в исследуемой позиции. В частности, в различных вариантах реализации ускоритель 8 НММ выполнен с возможностью присвоения вероятностей перехода для последовательности оснований рида, осуществляемых между каждым из этих состояний, совпадение («M»), инсерция («I») или делеция («D»), как более подробно описано ниже в настоящем документе.

[00300] Более конкретно, в зависимости от конфигурации, функция ускорения НММ может быть реализована в программном обеспечении, например, с помощью ЦПУ/ГПУ/КПУ 1000 и/или микросхемы 7, и/или может быть реализована в аппаратном обеспечении и может быть представлена в микросхеме 7, например, расположенной на карте или плате 2 расширения периферии. В различных вариантах реализации эта функциональная возможность может быть реализована частично в виде программного обеспечения, например, выполняемого ЦПУ/ГПУ/КПУ 1000, и частично в виде аппаратного обеспечения, реализованного на микросхеме 7 или посредством одной или более схем квантовой обработки. Соответственно, в различных вариантах реализации микросхема 7 может быть представлена на материнской плате ЦПУ/ГПУ/КПУ 1000 или она может быть частью периферийного устройства 2, или и то, и другое. Соответственно, модуль 8 ускорителя НММ может содержать или иным образом быть связанным с различными интерфейсами, например, 3, 5, 10 и/или 12, чтобы обеспечивать эффективную передачу данных в движки 13 обработки и из них.

[00301] Соответственно, как показано на ФИГ. 2 и 3, в различных вариантах реализации предусмотрена микросхема 7, выполненная с возможностью осуществления функции определения вариантов, например гаплотипов. Микросхема 7 может быть связана с ЦПУ/ГПУ/КПУ 1000, например, непосредственно сопряжена с ними, например, установлена на материнскую плату компьютера, или опосредованно соединена с ними, например, являться частью периферийного устройства 2, которое выполнено с

возможностью функционального соединения с ЦПУ/ГПУ/КПУ 1000, например, посредством одного или более межсоединений, например 3, 4, 5, 10 и/или 12. В данном случае микросхема 7 присутствует на периферийном устройстве 2. Необходимо понимать, что хотя ускоритель выполнен в виде микросхемы, он может быть также выполнен в виде одной или более квантовых схем квантового процессорного устройства, где квантовые схемы выполнены в виде одного или более движков обработки для выполнения одной или более функций, описанных в настоящем документе.

[00302] Следовательно, периферийное устройство 2 может содержать параллельную или последовательную шину 4 расширения, например, для соединения периферийного устройства 2 с центральным процессорным устройством (ЦПУ/ГПУ/КПУ) 1000 компьютера и/или сервера, например, посредством интерфейса 3, например DMA. В конкретных случаях периферийное устройство 2 и/или последовательная шина 4 расширения может представлять собой интерфейс межсоединения периферийных компонентов типа экспресс (PCIe), который выполнен с возможностью обмена данными с микросхемой 7, например через соединение 5. Как описана в настоящем документе, микросхема 7 может, по меньшей мере частично, быть выполнена как, или включать в себя иным образом, ускоритель 8 HMM. Ускоритель 8 HMM может быть выполнен как часть микросхемы 7, например, в аппаратной реализации и/или в виде кода, предназначенного для исполнения вместе с ней, и выполнен с возможностью осуществления функции определения вариантов, например, для выполнения одной или более операций скрытой марковской модели, на данных, подаваемых в микросхему 7 с помощью ЦПУ/ГПУ/КПУ 1000, например, через интерфейс 4 PCIe. Аналогичным образом после того, как одна или более функций определения вариантов выполнены, например, одна или более операций HMM выполнены, их результаты могут быть переданы из ускорителя 8 HMM схемы 7 по шине 4 в ЦПУ/ГПУ/КПУ 1000, например посредством соединения 3.

[00303] Например, в конкретных случаях предусмотрен ЦПУ/ГПУ/КПУ 1000 для обработки и/или передачи информации и/или исполнения инструкций наряду с микросхемой 7, которая лишь частично выполнена в виде ускорителя 8 HMM. ЦПУ/ГПУ/КПУ 1000 обменивается данными с микросхемой 7 посредством интерфейса 5, который выполнен с возможностью содействия обмену данными между ЦПУ/ГПУ/КПУ 1000 и ускорителем 8 HMM микросхемы 7 и, следовательно, может соединять с возможностью обмена данными ЦПУ/ГПУ/КПУ 1000 с ускорителем 8 HMM, который является частью микросхемы 7. Для содействия этим функция микросхема 7 содержит модуль 9 распределителя, который может представлять собой модуль CentCom, выполненный с возможностью передачи данных на множество движков 13 HMM, например посредством одного или более кластеров 11, где каждый движок 13 выполнен с возможностью приема и обработки данных, например, посредством выполнения на них протокола HMM, вычисления конечных значений, вывода его результатов и повторения этих действий. В различных случаях выполнение протокола HMM может включать в себя определение одной или более вероятностей перехода, как описано ниже в настоящем документе. В частности, каждый движок 13 HMM может быть выполнен с возможностью осуществления заданий, в том числе таких, как одно или более формирований и/или оценок виртуальной матрицы HMM, для создания тем самым и вывода конечного суммарного значения, причем конечное суммарное значение выражает вероятное правдоподобие того, что определенное основание совпадает или отличается от соответствующего основания в последовательности гипотетического гаплотипа, как описано ниже в настоящем документе.

[00304] На ФИГ. 3B приведено подробное изображение кластера 11 HMM, показанного на ФИГ. 3A. В различных вариантах реализации каждый кластер 11 HMM содержит один или более экземпляров 13 HMM. Могут быть предусмотрены один или множество кластеров, например, в соответствии объемом предоставляемых ресурсов, например, на микросхеме или квантовом вычислительном процессоре. В частности, может быть предусмотрен кластер HMM, который выполнен в виде концентратора 11 кластеров. Концентратор 11 кластеров получает данные, относящиеся к одному или более заданий 20, из распределителя 9 и дополнительно соединен с возможностью обмена данными с одним или более, например, множеством, экземпляров 13 HMM, например, посредством одной или более шин 12 экземпляров HMM, на которые концентратор 11 кластеров передает данные заданий 20.

[00305] Полоса пропускания для передачи данных по всей системе может быть процессом с относительно низкой полосой пропускания, и после того, как задание 20 принято, система 1 может быть выполнена с возможностью выполнения этого задания, например, без необходимости обращения из микросхемы 7 к памяти. В различных вариантах реализации в любой данный момент времени одно задание 20a отправляют на один движок 13a обработки, а несколько заданий $20_{a\text{-}n}$ могут быть распределены концентратором 11 кластеров на несколько движков 13a-$13_{m+1}$ обработки, например, когда каждый из движков обработки 13 будет работать над одним заданием 20, например, один сравнением между одним или более ридами и одной или более последовательностями гаплотипа, параллельно и высокими скоростями. Как описано ниже, выполнение такого задания 20 может, как правило, включать в себя формирование виртуальной матрицы, посредством которой последовательности «рида» субъекта могут быть сравнены с одной или более, например, двумя, последовательностями гипотетического гаплотипа, чтобы определить различия между ними. В таких случаях одно задание 20 может включать в себя обработку одной или более матриц, имеющих множество ячеек, которые нужно обработать для каждого выполняемого сравнения, например, основание за основанием. Поскольку геном человека насчитывает около 3 миллиардов пар оснований, при анализе генома человека с 30-кратным избыточным покрытием (что эквивалентно примерно 20 триллионам ячеек в матрицах всех связанных заданий по HMM) нужно будет выполнить порядка от 1 до 2 миллиардов различных заданий.

[00306] Соответственно, как описано в настоящем документе, каждый экземпляр 13 HMM может быть выполнен с возможностью осуществления протокола HMM, например, формирования и обработки матрицы HMM, на данных последовательности, таких как данные, принятые из ЦПУ/ГПУ/КПУ 1000. Например, как объяснялось выше, при секвенировании генетического материала субъекта, такого как ДНК или РНК, ДНК/РНК разбивают на сегменты, например, длиной до около 100 оснований. Затем проверяют идентичность этих сегментов из 100 оснований, например, с помощью автоматизированного секвенатора, и «считывают» в текстовый файл FASTQ или другой формат, который хранит идентичность каждого основания рида вместе с оценкой качества Phred (например, это обычно число от 0 до 63 на логарифмической шкале, где оценка 0 означает минимальную величину достоверности того, что определенное основание является верным, а оценки от 20 до 45 обычно считают приемлемыми как относительно точные).

[00307] В частности, как указано выше, оценка качества Phred является индикатором качества, который измеряет качество идентификации идентичностей нуклеотидов, сформированных процессором секвенирования, например, автоматизированным

секвенатором ДНК/РНК. Следовательно, каждое основание рида содержит свою собственную оценку качества, например, Phred, основанную на том, как секвенатор оценил качество данной конкретной идентификации. Phred представляет достоверность, с которой секвенатор оценивает, что он правильно определил идентичность основания. Затем эта оценка Phred используется модулем 8 HMM, как подробно описано ниже, для дальнейшего определения точности каждого определенного основания в риде по сравнению с гаплотипом, на который оно было картировано и/или выровнено, например, определения ее вероятностей перехода в состояния «совпадение», «инсерция» и/или «делеция», например в состояние «совпадение» и из него. Необходимо отметить, что в различных вариантах реализации система 1 может модифицировать или иным образом корректировать первоначальную оценку Phred перед выполнением протокола HMM с ее использованием, например, с учетом соседних оснований/оценок и/или фрагментов соседней ДНК и в предположении влияния таких факторов на оценку Phred основания, например, исследуемую ячейку.

[00308] В таких случаях, как показано на ФИГ. 4, система 1, например, компьютер/квантовое программное обеспечение, может определить и идентифицировать различные активные области $500_n$ в пределах секвенированного генома, которые могут быть использованы и/или иным образом подвергнуты дальнейшей обработке, как описано в настоящем документе, и которая может быть разбита на задания $20_n$, которые могут быть распараллелены среди различных ядер и доступных потоков 1007 по всей системе 1. Например, такие активные области 500 могут быть выявлены в качестве источников вариации между секвенированным и референсным геномом. В частности, ЦПУ/ГПУ/КПУ 1000 может иметь множество выполняющихся потоков 1007, идентифицируя активные области 500a, 500b и 500c, компилируя и агрегируя всевозможные разные задания $20_n$, которые нужно выполнить, например, посредством надлежащим образом сконфигурированного агрегатора 1008, на основе активных областей 500a-c, исследуемых в данный момент. Для того чтобы система 1 могла работать эффективно, может быть использовано любое подходящее количество потоков 1007, например, чем больше потоков, тем меньше потрачено активного времени на ожидание.

[00309] По завершении идентификации, компиляции и/или агрегирования потоки 1007/1008 передадут активные задания 20 в распределитель 9 данных, например, CentCom, модуля 8 HMM, например, через интерфейс 4 PCIe, например, в режиме «самонаведения», и затем перейдут к другим процессам в ожидании, когда модуль 8 HMM отправит выходные данные обратно, чтобы согласовать их обратно с соответствующей активной областью 500, на которую они картируются и/или выравниваются. После этого распределитель 9 данных распределяет задания 20 всевозможных разным кластерам 11 HMM, например, задание за заданием. Если все работает эффективно, это может быть в формате «первым пришел, первым обслужен», но это необязательно. Например, в различных вариантах реализации необработанные данные задания и обработанные результаты задания могут быть отправлены по всей системе по мере их доступности.

[00310] В частности, на ФИГ. 2, 3 и 4 различные данные задания 20 могут быть агрегированы в страницы по 4K байтов, которые могут быть отправлены посредством PCIe 4 в и через CentCom 9 и далее в движки 13 обработки, например, посредством кластеров 11. Количество отправляемых данных может быть больше или меньше 4K байтов, но, как правило, они будут содержать 100 заданий HMM на страницу данных объемом 4K (например, 1024). В частности, эти данные потом перевариваются распределителем 9 данных и подаются в каждый кластер 11, например, одну страницу

объемом 4K посылают на один кластер 11. Однако это необязательно, так как каждое отдельное задание 20 может быть послан в любой данный кластер 11 в зависимости от того, какой кластер становится доступным и когда.

[00311] Соответственно, подход на основе кластера 11, который представлен здесь, эффективно и с высокой скоростью распределяет поступающие данные движкам 13 обработки. В частности, по мере поступления данных на интерфейс 4 PCIe из ЦПУ/ГПУ/КПУ 1000, например, по соединению 3 DMA, принимаемые данные затем отправляются по шине 5 PCIe в распределитель 9 CentCom микросхемы 7 определителя вариантов. После чего распределитель 9 отправляет данные в один или более кластеров 11 обработки HMM, например, по одной или более специализированным шинам 10 кластеров, причем кластер 11 может затем передать эти данные в один или более экземпляров 13 обработки, например, по одной или более шин 12 экземпляров, например, для обработки. В этом случае интерфейс 4 PCIe выполнен с возможностью подачи данных через периферийную шину 5 расширения, распределитель 9 и/или шины 10 и/или 12 кластера и/или экземпляра с быстрой скоростью, например, с такой, которая может обеспечивать занятость одного или более, например, всех экземпляров $13_{a-(m+1)}$ ускорителя HMM в пределах одного или более, например, всех, кластеров $11_{a-(n+1)}$ HMM, например, в течение длительного периода времени, например, полного времени, причем на протяжении периода, в течение которого система 1 работает, задания 20 обрабатываются, и в то же время система успевает выводить данные HMM, которые должны быть отправлены обратно в один или более ЦПУ 1000, через интерфейс 4 PCIe.

[00312] Например, любая неэффективность интерфейсов 3, 5, 10 и/или 12, которая приводит к периоду простоя одного или более экземпляров 13 ускорителя HMM, может увеличивать общее время обработки системы 1. В частности, при анализе генома человека могут быть порядка двух или более миллиардов различных заданий 20, которые должны быть распределены различным кластерам 11 HMM и обработаны в течение некоторого периода времени, например, менее 1 часа, менее 45 минут, менее 30 минут, менее 20 минут, включая 15 минут, 10 минут, 5 минут или менее.

[00313] Соответственно, на ФИГ. 4 дан обзор примера потока данных по всему программному и/или аппаратному обеспечению системы 1, как описано в целом выше. Как показано на ФИГ. 4, система 1 может быть выполнена, частично, с возможностью передачи данных, например, между интерфейсом 4 PCIe и распределителем 9, таким как, CentCom, например, по шине 5 PCIe. Кроме того, система 1 может быть также выполнена, частично, с возможностью передачи принимаемых данных, например, между распределителем 9 и одним или более кластерами 11 HMM, например, посредством одной или более шин 10 кластеров. Таким образом, в различных вариантах реализации ускоритель 8 HMM может содержать один или более кластеров 11, например, один или более кластеров 11, выполненных с возможностью осуществления одного или более процессов функции HMM. В таком случае имеется интерфейс, такой как кластерная шина 10, которая соединяет CentCom 9 с кластером 11 HMM.

[00314] Например, на ФИГ. 5 приведена высокоуровневая схема, изображающая интерфейс входа в модуль 8 HMM и выхода из него, такой как вход и выход модуля кластера. Как показано на ФИГ. 6, каждый кластер 11 HMM может быть выполнен с возможностью обмена данными, например, приема данных из и/или отправки результирующих данных, например, суммарных данных, с распределителем 9 данных CentCom через кластерную шину 10. В частности, может быть предусмотрен любой подходящий интерфейс 5, если он позволяет интерфейсу 4 PCIe обмениваться данными с распределителем 9 данных. Более конкретно, шина 5 может представлять собой

межсоединение, содержащее логику интерпретации, полезную для сообщения распределителю 9 данных, какая логика интерпретации может быть выполнена с возможностью приспособления к любому протоколу, используемому для обеспечения этой функциональной возможности. А именно, в различных случаях межсоединение может выполнено в виде шины 5 PCIe.

[00315] Кроме того, кластер 11 может быть выполнен с возможностью использования в нем одной или более областей тактовой частоты и, следовательно, в пределах кластера 11 могут быть представлены один или более тактовых генераторов. В конкретных случаях могут быть предусмотрено множество областей тактовой частоты. Например, может быть предусмотрен более медленный тактовый генератор, например, для передач данных, скажем, в кластер 11 и из него. Кроме того, может быть предусмотрен более быстрый, например, высокоскоростной, тактовый генератор, который может использоваться экземплярами 13 HMM при выполнении различных вычислений состояний, описанных в настоящем документе.

[00316] В частности, в различных вариантах реализации, как показано на ФИГ. 6, система 1 может быть установлена таким образом, чтобы в первом случае, когда распределитель 9 данных использует существующий CentCom IP, может быть предусмотрена манжета, такая как уплотнительное кольцо, причем уплотнительное кольцо выполнено с возможностью трансляции сигнала в интерфейс 5 CentCom из интерфейса или шины 10 кластера HMM и обратно. Например, кластерная шина 10 HMM может соединять с возможностью обмена данными и/или функционально ЦПУ/ГПУ 1000 с различными кластерами 11 модуля 8 ускорителя HMM. Таким образом, как показано на ФИГ. 6, структурированные записанные и/или считанные данные для каждого гаплотипа и/или каждого рида могут быть отправлены по всей системе 1.

[00317] Вслед за заданием 20, вводимым в движок HMM, движок 13 HMM может, как правило, приступать к работе либо: a) немедленно, если она находится в состоянии простоя (IDLE), либо b) по завершении текущей назначенной задачи. Необходимо отметить, что каждый движок 13 ускорителя HMM может обрабатывать входные сигналы с попеременным переключением (например, может работать с одним набором данных во время загрузки другого), тем самым сводя к минимуму непроизводственную потерю времени между заданиями. Соответственно, манжета 11 кластера HMM может быть выполнена с возможностью автоматического приема входного задания 20, отправленного распределителем 9 данных и назначения его одному или более экземпляров 13 движка HMM в кластере 11, который может принять новое задание. На стороне программного обеспечения не требуется управления, которое может выбирать определенный экземпляр 13 движка HMM для определенного задания 20. Однако в различных случаях программное обеспечение может быть выполнено с возможностью управления такими экземплярами.

[00318] Соответственно, ввиду вышеизложенного, система 1 может быть рационализирована при передаче данных результатов обратно в ЦПУ/ГПУ/КПУ, и благодаря этой эффективности имеется не много данных, которые нужно вернуть в ЦПУ/ГПУ/КПУ, чтобы добиться пользы от результатов. Это позволяет системе достигать времени выполнения операции определения вариантов около 30 минут или меньше, например, около 25, или около 20 минут или меньше, например, около 18 или около 15 минут или меньше, включая около 10 или около 7 минут или меньше, даже около 5 или около 3 минут или меньше в зависимости от конфигурации системы.

[00319] На ФИГ. 6 приведено высокоуровневое представление различных функциональных блоков примера движка 13 HMM в пределах аппаратного ускорителя

8 на FPGA или ASIC 7. В частности, в пределах аппаратного ускорителя 8 HMM имеются множество кластеров 11, и в пределах каждого кластера 11 имеются множество движков 13. На ФИГ. 6 приведен пример движка 13 HMM. Как показано на ФИГ. 6, движок 13 может содержать интерфейс 12 шины экземпляра, множество памятей, например, HMEM 16 и RMEM 18, различные другие компоненты 17, логику 15 управления HMM, а также интерфейс 19 вывода результатов. В частности, на стороне движка шина 12 экземпляра HMM выполнена с возможность функционального соединения с памятями, HMEM 16 и RMEM 18, и может содержать логику интерфейса, которая обменивается данными с концентратором 11 кластеров, причем концентратор кластеров обменивается данными с распределителем 9, который, в свою очередь, обменивается данными с интерфейсом 4 PCIe, который обменивается данными с программным обеспечением определения вариантов, выполняемым ЦПУ/ГПУ и/или сервером 1000. Таким образом, шина 12 экземпляра HMM принимает данные из ЦПУ 1000 и загружает их в одну или более памятей, например HMEM и RMEM. Эта конфигурация может быть также реализована в одной или более квантовых схем и адаптирована соответствующим образом.

[00320] В этих случаях должен быть выделен достаточный объем памяти, чтобы можно было загружать по меньшей мере два или более гаплотипов, например, два гаплотипа, в HMEM 16, для данной последовательности рида, которую загружают, например, в RMEM 18, что в случае загрузки множества гаплотипов снижает нагрузку на полосу пропускания шины 5 PCIe. В конкретных случаях два гаплотипа или две последовательности рида могут быть загружены в их соответствующие памяти, что позволит обрабатывать четыре последовательности вместе во всех соответствующих комбинациях. В других случаях могут быть загружены, например, четыре, или восемь или шестнадцать, пар последовательностей и аналогичным образом обработаны в комбинации, например, чтобы еще более облегчить нагрузку на полосу пропускания, если требуется.

[00321] Кроме того, может быть зарезервировано достаточно памяти, чтобы в ней можно было реализовать структуру с попеременным переключением, так чтобы после того, как памяти загружены новым заданием 20a, например, на одной стороне памяти, подавался сигнал о новом задании, и логика управления 15 могла начать обработку нового задания 20a, например, путем формирования матрицы и выполнения необходимых вычислений, как описано ниже в настоящем документе. Соответственно, благодаря этому другая сторона памяти остается доступной, чтобы в нее можно было загрузить другое задание 20b, которое может быть загружена туда, пока первое задание 20a обрабатывается, а по завершении задания 20a можно было сразу же начинать обработку задания 20b с помощью логики 15 управления.

[00322] В таком случае матрица для задания 20b может быть обработана фактически без непроизводственной потери времени, например, одного или двух тактовых циклов, между завершением обработки первого задания 20a и началом обработки второго задания 20b. Следовательно, при использовании обеих сторон структуры памяти с попеременным переключением HMEM 16 может, как правило, хранить 4 последовательности гаплотипа, например, два фрагмента, а RMEM 18 может хранить, как правило, 2 последовательности рида. Эта конфигурация с попеременным переключением полезна, так как требует лишь немного дополнительного объема памяти, но позволяет удвоить пропускную способность движка 13.

[00323] Во время и/или после обработки памяти 16, 18 снабжают данными блок 17a калькулятора вероятностей перехода и таблицы подстановки (LUT), который выполнен с возможностью вычисления различной информации, относящейся к значениям «Prior»,

как объясняется ниже, и который, в свою очередь, подает данные результатов Prior в блок 17b калькулятора состояний M, I и D для использования при вычислении вероятностей перехода. Кроме того, могут быть включены одна или более сверхоперативных ОЗУ 17c, например, для хранения состояний M, I и D на границе полосы захвата, например, значений нижней строки полосы захвата обработки, которые, как было указано, в различных случаях могут быть любым подходящим количеством ячеек, например, около 10 ячеек, в длину, чтобы быть соразмерными с длиной полосы захвата 35.

[00324] Кроме того, может быть включен отдельный блок 19 интерфейса вывода результатов, чтобы по завершении суммирований они, например, 4 32-битовых слова, могли быть немедленно переданы обратно в программное обеспечение определения вариантов ЦПУ/ГПУ/КПУ 1000. Необходимо отметить, что эта конфигурация может быть сконфигурирована таким образом, чтобы система 1, а именно, калькулятор 17b M, I и D, не дожидались, пока интерфейс 19 вывода очистится, например, до тех пор, пока он не очистит результаты после того, как выполнит задание 20. Поэтому в данной конфигурации могут быть три ступени конвейере, функционирующие в унисон, чтобы создать общесистемный конвейер, такой как загрузка памяти, выполнение вычислений MID и вывод результатов. Кроме того, следует отметить, что любой данный движок 13 HMM является одним из многих со своими собственными интерфейсами 19 вывода, однако они могут совместно использовать общий интерфейс 10 для возврата в распределитель 9 данных. Следовательно, концентратор 11 кластеров будет включать в себя возможности управления для управления передачей («xfer») информации через ускоритель 8 HMM во избежание конфликтов.

[00325] Соответственно, далее подробно описаны процессы, выполняемые в каждом модуле движков 13 HMM, когда он принимает данные гаплотипа и рида, обрабатывает их и выводит полученные в результаты данные, относящиеся к ним, как большей часть описано в общих чертах выше. А именно, вычисления с широкой полосой пропускания в движке 13 HMM в пределах кластера 11 HMM направлены на вычисление и/или обновление значений состояний вставки (M), инсерции (I) и делеции (D), которые используют при определении того, совпадает ли конкретный исследуемый рид с референсом гаплотипа, а также в какой степени совпадает, как описано выше. В частности, рид вместе с оценкой Phred и значением штрафа на открытие гэпа (GOP) для каждого основания в риде передают в кластере 11 из распределителя 9 и тем самым назначают конкретному движку 13 обработки для обработки. Затем эти данные используются калькулятором 17 M, I и D движка 13 обработки для определения того, является ли определенное основание в риде более или менее правдоподобным, чтобы быть правильным и/или совпадать с соответствующим ему основанием в гаплотипе, или быть результатом вариации, например, инсерцией или делецией; и/или если имеется вариация, то вызвана ли эта вариация, по всей видимости, действительной изменчивостью в гаплотипе, или, скорее, артефактом вследствие ошибки в системах формирования последовательности, и/или картирования, и/или выравнивания.

[00326] Как указано выше, часть такого анализа включает в себя калькулятор 17 MID, определяющий вероятности перехода от одного основания к другому в риде, переходящим из одного состояния M, I или D в другое, в сравнении с референсом, например, из состояния совпадения в другое состояние совпадения, или из состояния совпадения либо в состояния инсерции, либо в состояния делеции. При выполнении таких определений каждую из связанных вероятностей перехода определяют и учитывают, когда оценивают, является ли наблюдаемый переход между ридом и

референсом истинной вариацией, а не просто какой-то ошибкой машины или обработки. В этих целях полезно использовать оценку Phred для каждого рассматриваемого основания при определении вероятностей переходов в состояние совпадения и из него, например, из состояния совпадения в состояние инсерции или делеции, например, с гэпом, при сравнении. Подобным образом определяют также вероятности перехода продления гэпа или выход из состояния с гэпом, например, состояния инсерции или делеции, обратно в состояние совпадения. В конкретных случаях вероятности входа в состояние делеции или инсерции и выхода из него, например, выхода из состояния продления гэпа, могут быть фиксированный величины, и могут упоминаться в настоящем документе как вероятность продления гэпа или штраф на продление гэпа. Тем не менее, в различных случаях такие штрафы на продление гэпа могут быть плавающими и, следовательно, подверженными изменению в зависимости от требований к точности конфигурации системы.

[00327] Соответственно, как показано на ФИГ. 7 и 8, для каждой возможной пары рида и гаплотипа вычисляют значение каждого из состояний M, I и D. В таком случае можно сформировать виртуальную матрицу 30 из ячеек, содержащих оцениваемый рид на одной оси матрицы и связанную последовательность гаплотипа на другой оси матрицы, таким образом, что каждая ячейка в матрице представляет позицию основания в риде и референсе гаплотипа. Следовательно, если каждая из последовательности рида и гаплотипа имеет длину в 100 оснований, матрица 30 будет содержать 100 на 100 ячеек, данную часть которой нужно будет обработать, чтобы определить правдоподобие и/или степень, в которой каждый конкретный рид совпадает с данных конкретным референсом. Поэтому после того как виртуальная матрица 30 сформирована, она может быть использована для определения различных переходов состояний, которые имеют место при перемещении от одного основания в последовательности рида к другому, и сравнения с такими же переходами последовательности гаплотипа, например, как изображено на ФИГ. 7 и 8. А именно, движок 13 обработки выполнен с возможностью обработки множества ячеек параллельно и/или последовательно при прохождении матрицы с помощью логики 15. Например, как изображено на ФИГ. 7, виртуальная полоса 35 захвата обработки распространяется и перемещается поперек и в низ матрицы 30, например, слева направо, обрабатывая отдельные ячейки матрицы 30 по диагонали вниз справа налево.

[00328] Точнее говоря, как показано на ФИГ. 7, каждая отдельная виртуальная ячейка в пределах матрицы 30 содержит значение состояния M, I и D, которое нужно вычислить, чтобы оценить характер идентичности найденного основания и, как изображено на ФИГ. 7, можно четко увидеть зависимости между этими данными для каждой ячейки в этом процессе. Следовательно, для определения данного состояния M текущей обрабатываемой ячейки, в текущую ячейку необходимо передвинуть состояния совпадения, инсерции и делеции ячейки, расположенной сверху по диагонали от нее, и использовать при вычислении состояния M ячейки, вычисляемого в данный момент (например, таким образом, продвижение диагонали вниз вперед по матрице указывает на совпадение),

[00329] Однако для определения состояния I в текущую обрабатываемую ячейку нужно передвинуть только состояния совпадения и инсерции из ячейки непосредственно выше текущей обрабатываемой ячейки (таким образом, при переходе в состояние инсерции получаем продвижение вертикально вниз «с гэпом»). Аналогичным образом для определения состояния D в текущую обрабатываемую ячейку нужно передвинуть только состояния совпадения и делеции из ячейки непосредственно слева от текущей

ячейки (таким образом, при переходе в состояние делеции получаем продвижение горизонтально поперек «с гэпом»). Как показано на ФИГ, 7, после того, как начинается вычисление ячейки 1 (затененная ячейка в крайней сверху строке), можно также начать обработку ячейки 2 (затененная ячейка во второй строке), не ожидая никаких результатов из ячейки 1, поскольку между этой ячейкой в строке 2 и ячейкой в строке 1, где начинается обработка, нет зависимости по данным. В результате образуется обратная диагональ 35, где продвижение продолжается вниз и влево, как показано красной стрелкой. Этот подход с продвижение обратной диагонали 35 повышает эффективность обработки и пропускную способность всей системы. Аналогичным образом данные, формируемые в ячейке 1, могут быть немедленно продвинуты в ячейку внизу и прямо справа от крайней сверху ячейки 1, тем самым продвигая полосу 35 захвата.

[00330] Например, на ФИГ. 7 изображен пример структуры 35 матрицы HMM, показывающей поток аппаратной обработки. Матрица 30 включает в себя индекс оснований гаплотипа, например, содержащий 36 оснований, расположенных слева направо вдоль верхнего края горизонтальной оси, а также включает в себя индекс оснований рида, например, содержащий 10 оснований, расположенных сверху вниз вдоль бокового края вертикальной оси таким образом, что формировать структуру ячеек, где выбранная ячейка может быть заполнена состояниями вероятности M, I и D и вероятностями перехода из текущего состояния в соседнее состояние. В таком случае, как более подробно описано выше, перемещение из состояния совпадения в состояние совпадения приводит к продвижению вперед по диагонали по матрице 30, тогда как перемещение из состояния совпадения в состояние инсерции приводит к продвижению вертикально вниз с образованием гэпа, а перемещение из состояния совпадения в состояние делеции приводит к продвижению по горизонтали с образованием гэпа. Следовательно, как показано на ФИГ. 8, для данной ячейки при определении состояний совпадения, инсерции и делеции для каждой ячейки используют вероятности совпадения, инсерции и делеции ее трех примыкающих ячеек.

[00331] Указывающая вниз стрелка на ФИГ. 7 представляет параллельный и последовательный характер движков обработки, которые выполнены с возможностью создания полосы захвата 35 или волны, перемещающейся поступательно вдоль виртуальной матрицы в соответствии с зависимостями данных (см. ФИГ. 7 и 8) для определения состояний M, I и D для каждой конкретной ячейки в структуре 30. Соответственно, в определенных случаях может потребоваться вычислить идентичности каждой ячейки в направлении вниз и по диагонали, как объяснено выше, а не просто рассчитать каждую ячейку исключительно вдоль вертикальной или горизонтальной оси, хотя это можно сделать, если требуется. Это обусловлено повышенным временем ожидания, например, задержкой, которое потребуется при обработке виртуальных ячеек матрицы 30 по отдельности и последовательно только вдоль вертикальной или горизонтальной оси, например, посредством аппаратной конфигурации.

[00332] Например, в таком случае при движении линейно и последовательно по виртуальной матрице 30, например, строка за строкой или столбец за столбцом, для обработки каждой новой ячейки вычисления каждой предыдущей ячейки должны быть завершены, что повышает время ожидания в целом. Однако при распространении вероятностей M, I, D каждой новой ячейки в направлении вниз и по диагонали системе 1 не нужно дожидаться, пока обработка ее предшествующей ячейки, например, строки один, завершится, прежде чем начинать обработку примыкающей ячейки в строке два матрицы. Это позволяет параллельно и последовательно обрабатывать ячейки по

диагонали, а также позволяет скрывать различные задержки вычислений конвейера, связанные с расчетами состояний M, I и D. Соответственно, по мере перемещения полосы 35 захвата по матрице 30 слева направо вычислительная обработка перемещается по диагонали вниз, например, влево (как показано стрелкой на ФИГ. 7). Эта конфигурация может быть особенно полезна для аппаратных и/или основанных на квантовых схемах реализаций, например, когда задержка памяти и/или межтактовая задержка имеют первостепенное значение.

[00333] В таких конфигурациях действительно ценным результатом каждого вызова движка 13 HMM, например, по завершении вычисления всей матрицы 30, может быть нижняя строка (например, строка 35 на ФИГ. 21), содержащая состояния M, и D, где состояния M и I могут быть суммированы (состояния D можно игнорировать на этом этапе, так как они уже выполнили свою функцию при обработке вычислений вверху), чтобы получить конечное суммарное значение, которое может быть единственной вероятностью, оценивающей для каждого индекса рида и гаплотипа вероятность наблюдения рида, например, в предположении, что гаплотип был действительно взят из исходной ДНК.

[00334] В частности, результатом обработки матрицы 30, например, изображенной на ФИГ. 7, может быть одно значение, представляющее вероятность того, что рид является фактическим представлением данного гаплотипа. Эта вероятность представляет собой значение между 0 и 1 и получается путем суммирования всех состояний M и I с нижней строки ячеек в матрице 30 HMM. По существу оценивается именно возможность того, что произошла какая-то ошибка в секвенаторе или соответствующих методах приготовления ДНК перед секвенированием, приведшая к неправильному возникновению несовпадения, инсерции или делеции в риде, в действительности отсутствующих в генетической последовательности субъекта. В таком случае рид не является истинным отражением фактической ДНК субъекта.

[00335] Следовательно, учитывая такие производственные ошибки, можно определять, что в действительности представляет любой данный рид относительно гаплотипа, что позволяет системе лучше определять, как генетическая последовательность субъекта, например, в целом, может отличаться от референсной последовательности. Например, можно сравнивать множество гаплотипов с множеством последовательностей считывания, формируя оценки для всех них и определяя на основе того, какие совпадения имеют лучшие оценки, какова действительная идентичности геномной последовательности индивида, и/или как она в действительности отличается от референсного генома.

[00336] Более конкретно, на ФИГ. 8 изображена увеличенная часть матрицы 30 состояний HMM, показанной на ФИГ. 7. Как показано на ФИГ. 8, при данном внутреннем составе каждой ячейки в матрице 30, как и структуре матрицы в целом, вероятность состояний M, I и D для любой данной «новой» вычисляемой ячейки зависит от состояний M, I и D нескольких окружающих ее соседей, которые уже вычислены. В частности, как более подробно показано на ФИГ. 1 и 16, в примере конфигурации вероятность переход из состояния совпадения в другое состояние совпадения может составлять приблизительно 0,9998, а вероятность перехода из состояния совпадения в состояние инсерции или делеции, например, с гэпом, (штраф на открытие гэпа) составляет только 0,0001. Кроме того, при пребывании в состоянии гэпа вследствие инсерции или в состоянии гэпа вследствие делеции вероятность остаться в состоянии гэпа (штраф на продление или продолжение) может составлять лишь 0,1, тогда как вероятность возврата в состояние совпадения составляет 0,9. Необходимо отметить, что в соответствии с

этой моделью все вероятности входа в данное состояние или выхода из него в сумме должны давать единицу. В частности, обработка матрицы 30 вращается вокруг вычисления вероятностей перехода с учетом различных штрафом на открытие гэпа или продолжение гэпа, в конце вычисляют сумму.

[00337] Следовательно, эти вычисленные вероятности перехода в состояния получают главным образом их непосредственно примыкающих ячеек в матрице 30, например, из ячеек, которые находятся непосредственно слева, сверху и слева по диагонали от данной вычисляемой ячейки, как показано на ФИГ. 16. Кроме того, вероятности перехода в состояния могут частично получать из оценки качества «Phred», которая прилагается к каждому основанию рида. Поэтому данные вероятности перехода полезны при вычислении значений состояний M, I и D для конкретной ячейки и, аналогичным образом, для любой связанной новой вычисляемой ячейки. Необходимо отметить, что, как описано в настоящем документе, штрафы на открытие гэпа и продолжение гэпа могут быть фиксированными значениями, однако в различных случаях штрафы на открытие гэпа и продолжение гэпа могут быть переменными и, следовательно, могут быть запрограммированы в системе, хотя и с использованием дополнительных аппаратных ресурсов, специально предназначенных для определения таких расчетов переменной вероятности перехода. Такие случаи могут быть полезны, если требуется повышенная точность. Тем не менее, когда такие значения предполагаются постоянными, можно уменьшить использование ресурсов и/или размер микросхемы, что приводит к повышению скорости обработки, как объяснено ниже.

[00338] Соответственно, при получении каждого нового значения состояния M, I и D используют множество вычислений и/или других математических операций, таких как умножения и/или сложения. В таком случае, чтобы максимально увеличить пропускную способность вычисления, примитивные математические операции, используемые при каждом вычислении состояния перехода M, I и D, можно организовать в виде конвейера. Такая конвейерная организация может быть выполнена таким образом, чтобы соответствующие тактовые частоты были высокими, но при этом глубина конвейера могла быть нетривиальной. Кроме того, такой конвейер можно выполнить таким образом, чтобы он имел конечную глубину, и в таких случаях для выполнения операций может потребоваться более одного тактового цикла.

[00339] Например, эти вычисления могут выполняться при высоких скоростях внутри процессора 7, например, около 300 МГц. Этого можно достичь путем тщательной конвейерной организации FPGA или ASIC с помощью регистров, чтобы между каждым триггерным переключением производилось мало математических операций. Эта структура конвейера приводит к множеству циклов задержки при переходе от ввода состояния совпадения до вывода, но при данной обратной диагональной структуре вычисления, показанной на ФИГ. 7 выше, эти задержки могут быть скрыты по всей матрице 30 HMM, например, когда каждая ячейка представляет один тактовый цикл.

[00340] Следовательно, количество вычислений состояний M, I и D может быть ограничено. В таком случае движок 13 обработки может быть выполнен таким образом, чтобы группировка, например, полоса 35 захвата, ячеек во множестве строк матрицы 30 могла быть обработана как группа (например, по диагонали вниз влево, как показано на ФИГ. 7), прежде чем переходить к обработке второй полосы захваты ниже, например, когда вторая полоса захвата содержит такое же количество подлежащих обработке ячеек в строках, что и первая. Подобным образом аппаратная реализация ускорителя 8, описанная в настоящем документе, может быть выполнена с возможностью повышения общей эффективности системы, как описано выше.

[00341] В частности, на ФИГ. 9 показан пример вычислительной структуры для выполнения различных вычислений состояний, описанных в настоящем документе. Более конкретно, на ФИГ. 9 показаны три специализированных логических блока 17 движка 13 обработки для производства вычислений состояний, используемых при формировании каждого значения состояния M, I и D для каждой конкретной ячейки или группировки ячеек, обрабатываемой в матрице 30 HMM. Эти логические блоки могут быть реализованы в аппаратном обеспечении, но в некоторых случаях могут быть реализованы в программном обеспечении, например, выполняемом одной или более квантовыми схемами. Как показано на ФИГ. 9, вычисление 15a состояния совпадения задействует больше операций, чем вычисления 15b инсерции или 15c делеции, поскольку при вычислении 15a состояния совпадения текущей обрабатываемой ячейки текущее вычисление совпадения включает в себя все предыдущие состояния совпадения, инсерции и делеции примыкающих ячеек наряду с различными данными «Prior» (см. ФИГ. 9 и 10), тогда как вычисления состояний инсерции и делеции включают в себя только состояния совпадения и либо инсерции, либо делеции, соответственно. Поэтому, как показано на ФИГ. 9, при вычислении состояния совпадения используют три умножителя состояний, а также два сумматора и, наконец, конечный умножитель, который учитывает значение Prior, например, Phred. Однако при вычислении состояния I или D задействуют только два умножителя и один сумматор. Следует отметить, что в аппаратном обеспечении умножители более ресурсоемкие, чем сумматоры.

[00342] Соответственно, в различной степени значения состояний M, I и D для обработки каждой новой ячейки в матрице 30 HMM используют знание или предварительное вычисление следующих значений, таких как «предыдущие» значения состояний M, I и D слева, сверху и/или по диагонали слева и выше текущей вычисляемой ячейки в матрице HMM. Кроме того, такие значения, представляющие предыдущую информацию или значения «Prior», могут быть, по меньшей мере частично, основаны на оценке качества «Phred» и зависеть от того, совпадают ли основание рида и основание референса в данной ячейке в матрице 30, или они разные. Такая информация особенно полезна при определении состояния совпадения. А именно, как показано на ФИГ. 9, в таких случаях в основном существуют семь «вероятностей перехода» (из M в M, из I в M, из D в M, из I в I, из M в I, из D в D и из M в D), которые указывают и/или оценивают вероятность наблюдения открытия гэпа, например, наблюдения перехода из состояния соответствия в состояние инсерции или делеции; наблюдения закрытия гэпа, например, перехода из состояния инсерции или делеции обратно в состояние совпадения; и наблюдения следующего состояния, остающегося в прежнем состоянии, например, из соответствия в соответствие, из инсерции в инсерцию, из делеции в делецию.

[00343] Значения состояния (например, в любой ячейке, подлежащей обработке в матрице 30 HMM), значения Prior и вероятности перехода являются значениями в диапазоне

[0,1]. Кроме того, известны также начальные условия для ячеек, которые находятся слева и правя по краям матрицы 30 HMM. Как видно из логики 15a на ФИГ. 9, существуют четыре операции умножения и две операции сложения, которые могут быть использованы при вычислении конкретного состояния M, определяемого для любой данной обрабатываемой ячейки. Аналогичным образом, как видно из логик 15b и 15c, при вычислении каждого состояния I и каждого состояния D, соответственно, используют два умножения и одно сложение. В совокупности вместе с умножителем предварительных данных это дает в сумме восемь операций умножения и четыре операции сложения для вычислений состояний M, I и D каждой одной ячейки матрицы

30 HMM, подлежащей обработке.

[00344] Результат заключительного суммирования, например, строки 34 на ФИГ. 16, вычисления матрицы 30, например, для одного задания 20 сравнения одного рида с одним или двумя гаплотипами, представляет собой сумму окончательных состояний M и I по всей нижней строке 34 матрицы 30, которая является суммарным окончательным значением, выводимым из ускорителя 8 HMM и подаваемым в ЦПУ/ГПУ/КПУ 1000. Данное окончательное суммарное значение представляет, насколько хорошо рид совпадает с гаплотипом (-ами). Это значение является вероятностью, например, меньше единицы, для отдельного задания 20a, которая затем может быть сравнена с выходным результатом другого задания 20b, например, относящегося к той же самой активной области 500. Следует отметить, что существуют порядка 20 триллионом ячеек HMM, которые нужно оценить в «типичном» геноме человека с 30-кратным покрытием, причем эти 20 триллионов ячеек HMM распределены по от 1 до 2 миллиардов матриц 30 HMM всего соответствующего задания 20 HMM.

[00345] После этого результаты таких вычислений могут быть сравнены друг с другом, чтобы более точно определить, например, путем сравнения оснований одного за другим, как генетическая последовательность субъекта отличается от последовательности одного или более референсных геномов. Для заключительного вычисления суммы можно повторно использовать сумматор, который уже применялся для вычисления состояний M, I и/или D отдельных ячеек, чтобы вычислить окончательное суммарное значение, например, путем включения мультиплексора в выбор повторно используемых сумматоров, тем самым включая одну последнюю дополнительную строку, например, к времени вычисления, в матрицу, чтобы она вычисляла эту окончательную сумму, что в случае рида длиной в 100 оснований приводит к непроизводственным затратам ресурсов около 1%. В альтернативных вариантах реализации для выполнения таких вычислений можно использовать специально предназначенные аппаратные ресурсы. В различных случаях логика сумматоров для вычислений состояний M и D может быть использована для вычисления окончательной суммы, причем сумматор состояния D может быть эффективно применен таким образом, чтобы он не использовался иным образом в заключительной обработке, приводящей к суммарным значениями.

[00346] В определенных случаях эти вычисления и соответствующие процессы могут быть выполнены с возможностью соответствия выходу данной платформы секвенирования, например, содержащей множество секвенаторов, который в совокупности могут быть в состоянии вывода (в среднем) нового генома человека с 30-кратным покрытием каждые 28 минут (хотя они выходят из множества секвенаторов группами примерно по 150 геномов раз в три дня), В таких случаях, когда текущие операции картирования, выравнивания и определения вариантов выполнены с возможностью вписывания в такую платформу секвенирования технологий обработки, часть из 28 минут (например, около 10 минут), которую занимает секвенирование генома кластером секвенирования, может быть использована соответствующим образом сконфигурированным средством картирования и/или выравнивателем, как описано в настоящем документе, для получения результаты файла изображения /BCL/FASTQ из секвенатора и выполнения этапов картирования и/или выравнивания генома, например, последующей обработки после секвенатора. В результате остается около 18 минут периода времени секвенирования для выполнения этап определения вариантов, основной вычислительной частью которого является операция HMM, например, до того, как секвенатор нуклеотидов секвенирует следующий геном, например, в течение следующих

28 минут. Соответственно, в таких случаях 18 минут могут быть выделены на вычисление 20 триллионов ячейке HMM, которые нужно обработать в соответствии с обработкой генома, например, когда обработка ячеек HMM включают в себя около двенадцати операций (например, восемь операций умножения и/или четыре операции сложения). Такая пропускная способность делает возможной следующую вычислительную динамику (20 триллионов ячеек HMM) × (12 математических операций на ячейку)/(18 минут × 60 секунд/минуту), что дает пропускную способность около 222 миллиардов операций в секунду при непрерывной работе.

[00347] На ФИГ. 10 показаны логические блок 17 движка обработки, приведенного на ФИГ. 9, в том числе схемы обновления состояний M, I и D, которые представляют упрощение схемы, приведенной на ФИГ. 9. Система может быть выполнена с возможностью неограничения памяти, чтобы один экземпляр 13 движка HMM (например, который вычисляет все одиночные ячейки в матрице 30 HMM со скоростью одна ячейка за тактовый цикл, в среднем, плюс непроизводственные затраты ресурсов) мог быть дублирован множество раз (по меньшей мере 65~70 раз, чтобы добиться эффективной пропускной возможности, как описано выше). Тем не менее, чтобы свести к минимуму размер аппаратного обеспечения, например размер микросхемы 2 и/или использование связанных с ней ресурсов и/или предпринять дальнейшую попытку включить на микросхему 2 столько экземпляров 13 движка HMM, сколько требуется и/или возможно, можно внести упрощения в логические блоки 15a'-c' экземпляра обработки 13 для вычисления одной или более вероятностей перехода, которые нужно вычислить.

[00348] В частности, можно предположить, что штраф на открытие гэпа (GOP) и штраф на продолжение гэпа (GCP), как описано выше, например, для инсерций и делеций, одинаковые и известны заранее конфигурации микросхемы. Данное упрощение означает, что вероятности перехода из I в M и из D в M идентичны. В таком случае один или более умножителей, например, показанных на ФИГ. 9, могут быть устранены, например, предварительным сложением состояний I и D перед умножением на общую вероятность перехода из индела в M. Например, в различных вариантах, если вычисления состояний I и D предполагаются одинаковыми, то вычисления состояний для ячейки можно упростить, как показано на ФИГ. 10. В частности, если значения состояний I и D одинаковые, то состояние I и состояние D можно сложить, а затем сумму умножить на одно значение, тем самым сэкономив умножение. Это можно сделать, поскольку, как показано на ФИГ. 10, штраф на продолжение и/или закрытие гэпа для состояний I и D одинаковые. Однако, как указано выше, система может быть выполнена с возможностью вычисления разных значений вероятностей переходов для обоих состояний I и D, и в таком случае данное упрощение не будет использоваться.

[00349] Кроме того, в дальнейшем описании вместо того, чтобы специально предназначенная схема или другие вычислительные ресурсы, выполненные определенным образом с возможностью выполнения заключительной операции суммирования внизу матрицы HMM, настоящий ускоритель 8 HMM может быть выполнен с возможностью эффективного присоединения одной или более дополнительных строк к матрице 30 HMM с точки зрения вычислительного времени, например, непроизводственных затрат ресурсов, которое он тратит на выполнение вычисления, и может также быть выполнен с возможностью «одалживать» один или более сумматоров из логики вычисления 15a M-состояния и 15c D-состояния, например, путем мультиплексирования в окончательных суммарных значений в существующие сумматоры по мере надобности для выполнения фактического окончательного

вычисления суммирования. В таком случае заключительная логика, включающая блоки логики 15a M-состояния, логики 15b I-состояния и логики 15c D-состояния, которые вместе образуют часть экземпляра 17 MID HMM, может содержать 7 умножителей и 4 сумматора наряду с различным задействуемым мультиплексированием.

[00350] Соответственно, на ФИГ. 10 показаны схемы 15a', 15b' и 15c' обновления состояний M, I и D с учетом упрощающих допущений в отношении вероятностей перехода, а также с учетом совместного использования различных ресурсов M, I и/или D, например, ресурсов сумматоров, для заключительных операций суммирования. Кроме того, к блоку вычисления M-состояния может быть добавлен блок задержки в путь M-состояния, как показано на ФИГ. 10. Эта задержка может быть добавлена для компенсации задержек в фактической аппаратной реализации операций умножения и сложения и/или для упрощения логики управления, например, 15.

[00351] Как показано на ФИГ. 9 и 10, эти соответствующие умножители и/или сумматоры могут быть умножителями и сумматорами с плавающей запятой. Однако в различных случаях, как показано на ФИГ. 11, может быть реализована конфигурация логарифмической области, где в такой конфигурации все умножители превращаются в сумматоры. На ФИГ. 12 показано, как выглядели вычисления в логарифмической области, если все умножители превратились бы в сумматоры, например, 15a", 15b" и 15c", как это происходит при использовании вычислительной конфигурации логарифмической области. В частности, вся логика умножителей превращается в сумматор, но сам сумматор превращается или иным образом включает в себя функцию, такую как: $f(a,b) = \max(a,b) - \log_2(1+2^{(-[a-b])})$, например, когда логарифмическая часть уравнения может поддерживаться в LUT, глубина и физический размер которой определяются требуемой точностью.

[00352] При типичных длинах последовательностей рида и гаплотипа и значениях, обычно наблюдаемых для оценок качества (Phred) и соответствующих вероятностей перехода, требования к динамическому диапазону значений состояний HMM могут быть довольно серьезными. Например, при реализации модуля HMM в программном обеспечении различные задания 20 HMM могут приводить к неполному выполнению, например, при реализации на значениях состояний одинарной (32-битовой) точности с плавающей запятой. Это означает динамический диапазон, который больше 80 степеней 10, и поэтому требует повышения программного обеспечения определения вариантов до работы со значениями состояний двойной (64-битовой) точности с плавающей запятой. Однако полное 64-битовое представление двойной точности с плавающей запятой во многих случаях имеет некоторые отрицательные последствия, например, если должны быть реализован высокоскоростное аппаратное обеспечение, нужно будет повысить требования к памяти и ресурсам вычислительного конвейера, тем самым занимая больше места на микросхеме и/или замедляя согласование по времени. В таких случаях может быть реализовано представление чисел в линейной области только с фиксированной запятой. Тем не менее, требования по динамическому диапазону к значениям состояний в данном варианте реализации делают битовые ширины в определенных обстоятельствах менее желательными. Соответственно, в таких случаях может быть реализовано представление чисел в логарифмической области только с фиксированной запятой, как описано в настоящем документе.

[00353] В такой схеме, как показано на ФИГ. 11, вместо представления в памяти и вычисления фактического значения состояния может быть представлен -log по основанию 2. Это может дать несколько преимуществ, в том числе использование операций умножения в линейном пространстве, которое переводит в операции сложения

в логарифмическое пространстве; и/или данное представление чисел в логарифмической области по своей природе поддерживает более широкий динамический диапазон лишь при небольшом увеличении количества целочисленных битов. Эти вычисления обновлений состояний M, I, D в логической области показаны на ФИГ. 11 и 12.

[00354] Как можно заметить при сравнении конфигурации логики 17 на ФИГ. 11 и ФИГ. 9, операции умножения исчезают в логарифмической области. Вернее, они заменяются операциями сложения, а операции сложения преобразуются в функцию, которая может быть выражена как операция максимума с последующим добавлением поправочного коэффициента, например, посредством LUT, где поправочный коэффициент является функцией от разницы между двумя значениями, суммируемыми в логарифмической области. Такой поправочный коэффициент может быть либо вычислен, либо сформирован из таблицы подстановки. Что эффективнее использовать, вычисление поправочного коэффициента или реализация его с помощью таблицы подстановки, зависит от требуемой точности (битовой ширины) разницы между значениями суммы. Поэтому в конкретных случаях количество битов для представления состояния в логарифмической области может быть примерно от 8 до 12 целочисленных битов плюс от 6 до 24 битов для дробных битов в зависимости от требуемого уровня качества для любой данной реализации. Это означает где-то от 14 до 36 битов всего для представления значения состояния в логарифмической области. Кроме того, было определено, что существуют представления с фиксированной запятой в логарифмической области, которые могут обеспечивать приемлемое качество и приемлемые размер и скорость аппаратного обеспечения.

[00355] В различных случая для каждого задания 20 HMM обычно обрабатывают одну последовательность рида, что, как было указано, может включать в себя сравнение с двумя последовательностями гаплотипа. Аналогично вышесказанному в отношении памяти гаплотипа, в памяти 18 последовательности рида тоже может быть использована структура с попеременным переключением, чтобы обеспечить различным программно реализованным функциям возможность записи информации нового задания 20b HMM в то время, когда текущее задание 20b все еще выполняется экземпляром 13 движка HMM. Следовательно, в качестве памяти для хранения последовательности рида может потребоваться одна память 1024×32 с двумя портами (например, один порт для записи, один порт для чтения, и/или отдельные тактовые генераторы для портов записи и чтения).

[00356] В частности, как описано выше, в различных случаях архитектура, используемая системой 1 выполнена с возможностью формирования виртуальной матрицы 30 при определении того, совпадает ли данное основание в секвенированном геноме образца с соответствующим основанием в одном или более референсных геномах, где референсный геном теоретически расположен по горизонтальной оси, тогда как секвенированные последовательности, представляющие геном образца, теоретически расположены в порядке убывания вдоль вертикальной оси. Следовательно, при выполнении вычисления HMM движка 13 обработки HMM, который описан в настоящем документе, выполнен с возможностью прохождения этой виртуальной матрицы 30 HMM. Такая обработка может быть изображена как на ФИГ. 7, когда полоса 35 захвата движется по диагонали виз и поперек виртуального массива, выполняя различные вычисления HMM для каждой ячейки виртуального массива, как показано на ФИГ. 8.

[00357] Более конкретно, данный теоретический проход подразумевает обработку первой группировки строк ячейки 35a из матрицы 30 во всей ее полноте, например, для всех оснований гаплотипа и рида в пределах группировки, прежде чем переходить вниз

к следующей группировке строк 35b (например, к следующей группе оснований рида). В таком случае значения состояний M, I и D для первой группировки сохраняют на нижнем крае этой первоначальной группировки строк, чтобы эти значения состояний M, I и D могли быть затем использованы для подачи верхней строки следующей группировки (полосы захвата) вниз в матрице 30. В различных случаях система 1 может быть выполнена с возможностью обеспечения подачи в ускоритель 8 HMM гаплотипов и/или ридов длиной до 1008, и так как числовое представление использует W-битов для каждого состояния, это вытекает в память размером 1008 слов × W битов для хранения состояний M, I и D.

[00358] Соответственно, как было указано, такая память может быть памятью с одним портом или с двумя портами. Кроме того, может быть также предусмотрена сверхоперативная память уровня кластера, например, для хранения результатов границы полосы захвата. Например, в соответствии с вышеизложенным описанием обсуждаемые памяти уже сконфигурированы для каждого экземпляра 13 движка. В конкретных реализациях HMM множество экземпляров $13a\text{-}_{(n+1)}$ движков могут быть сгруппированы в кластер 11, который обслуживается одним соединением, например, шиной 5 PCIe, с интерфейсом 4 PCIe и 3 DMA посредством 9 CentCom. Чтобы более эффективно использовать полосу пропускания PCIe с помощью существующих функциональных возможностей CentCom 9, можно создать множество экземпляров кластеров $11a\text{-}_{(n+1)}$.

[00359] Следовательно, в типичной конфигурации в пределах кластера $11_n$ создают где-то от 16 до 64 экземпляров движков $13_m$, а в типичной реализации модуля 8 HMM на FPGA/ASIC могут быть созданы от одного до четырех кластеров (например, в зависимости от того, является ли это специально предназначенной для HMM FPGA для обработки изображений, или должна ли HMM делить полезную площадь FPGA с секвенатором/сопоставителем/выравнивателем и/или другими модулями, как описано в настоящем документе). В конкретных случаях на уровне кластера 11 в аппаратном обеспечении HMM может быть небольшой объем используемой памяти. Эта память может быть использована в качестве эластичной памяти «первым пришел, первым обслужен» («FIFO») для сбора выходных данных из экземпляра 13 движка HMM в кластере и передачи их на CentCom 9 для дальнейшей передачи обратно в программное обеспечение на ЦПУ 1000 посредством интерфейса 3 DMA и 4 PCIe. Теоретически эта FIFO может быть очень маленькой (порядка двух 32-битовых слов), поскольку, как правило, после поступления FIFO данные почти сразу же передаются в 9 CentCom. Однако для поглощения потенциальных разрывов на пути выходных данных размер этой памяти FIFO может быть параметризуемым. В конкретных случаях FIFO может использоваться с глубиной в 512 слов. Таким образом, требованием к памяти уровня кластера может быть одна память 512×32 с двумя портами (отдельные порты чтения и записи, одна и та же тактовая область).

[00360] На ФИГ. 12 приведены различные переходы 17b состояний HMM, изображающие взаимосвязь между штрафами на открытие гэпа (GOP), штрафами на продление гэпа (GCP) и вероятностями перехода, участвующими в определении того, совпадает ли данная последовательность рида с конкретной последовательностью гаплотипа, и насколько. При выполнении такого анализа движка 13 HMM содержит по меньшей мере три логических блока 17b, таких как логический блок 15a для определения состояния совпадения, логический блок 15b для определения состояния инсерции и логический блок 15c для определения состояния делеции. Эти логики 17 для вычисления состояний M, I и D при надлежащей конфигурации функционируют

эффективно во избежание узких мест с широкой полосой пропускания, например, с потоком вычислений HMM. Однако после того, как базовая архитектура вычисления M, I, D определена, можно также сконфигурировать и реализовать другие улучшения системы, чтобы избежать образования других узких мест в системе.

[00361] В частности, система 1 может быть выполнена с возможностью максимального повышения эффективности подачи информации из ядра 1000 компьютера в модуль 2 определителя вариантов и обратно, чтобы не создавать узких мест, которые могли бы ограничить общую пропускную способность. Одним из таких блоков, который снабжает логику 17 вычисления состояний M, I, D ядра HMM, является блок вычисления вероятностей перехода и предварительных данных. Например, как показано на ФИГ. 9, каждый тактовый цикл использует представление семи вероятностей перехода и одно значение Prior на входе в блок 15a вычисления состояний M, I, D. Однако после упрощений, которые приводят к архитектуре, приведенной на ФИГ. 10, для каждого цикла на входе блока вычисления состояний M, I, D используются только четыре уникальные вероятности перехода и одно значение Prior. Соответственно, в различных случаях эти расчеты могут быть упрощены и могут быть сформированы результирующие значения. Таким образом достигаются повышение пропускной способности, эффективности и снижение возможности формирования узких мест на этой стадии процесса.

[00362] Кроме того, как описано выше, значения Prior являются значениями, формируемыми с использованием качества рида, например, оценки Phred, конкретного основания для текущей оцениваемой ячейки в виртуальной матрице 30 HMM. Эту взаимосвязь можно описать с помощью следующих уравнений: Во-первых, исследуемого рида Phred можно выразить как вероятность = 10^(-(Phred рида/10)). Затем можно вычислить Prior в зависимости от того, совпадает ли основание рида с основанием гипотетического гаплотипа: Если основание рида и основание гипотетического гаплотипа совпадают: Prior = 1 - Phred рида, выраженную как вероятность. В противном случае: Prior = (Phred рида, выраженная как вероятность)/3. Операция деления на три в этом последнем уравнении отражает тот факт, что существуют только четыре возможных основания (A, C, G, T). Следовательно, если основание рида и основание гаплотипа не совпали, значит совпасть должно одно из трех оставшихся оснований, и каждую из трех вероятностей моделируют как одинаково правдоподобную.

[00363] Оценки Phred для каждого основания подают в аппаратный ускоритель 8 HMM в виде 6-битовых значений. Поэтому уравнения для значений Prior имеют 64 возможных результата для случая «совпадение» и 64 возможных результата для случая «несовпадение». Это можно эффективно реализовать в аппаратном обеспечении в виде таблицы подставки размером в 128 слов, где адрес в таблице подстановки представляет собой 7-битовую величину, образованную путем конкатенации значения Phred с одним битом, который указывает, совпадает ли основание рида с основание гипотетического гаплотипа или нет.

[00364] Кроме того, что касается определения вероятностей перехода совпадения в инсерцию и/или совпадения в делецию, в различных вариантах реализации архитектуры для аппаратного ускорителя 8 HMM можно определить отдельные штрафы на открытие гэпа (GOP) для перехода состояния из совпадения в инсерцию и перехода из совпадения в делецию, как указано выше. Это равносильно тому, что значениям M2I и M2D на диаграмме переходов состояний, изображенной на ФИГ. 12, различаются. Поскольку значения GOP получают из аппаратного ускорителя 8 HMM в виде 6-битовых значений типа Phred, вероятности перехода в состояние открытия гэпа могут быть вычислены в

соответствии со следующими уравнениями: Вероятность перехода M2I = 10^(-(GOP(I) рида/10)) и вероятность перехода M2D = 10^(-(GOP(D) рида/10)). Аналогично получению значений Prior в аппаратном обеспечении для получения значений M2I и M2D можно использовать простую таблицу подставки размером в 64 слова. Если GOP(I) и GOP(D) вводят в аппаратный ускоритель 8 HMM как потенциально разные значения, то можно использовать две такие таблицы подстановки (или одну таблицу подстановки с совместным использованием ресурсов, потенциально работающую с удвоенной тактовой частотой по сравнению с остальной схемы).

[00365] Кроме того, что касается определения вероятностей перехода совпадения в совпадение, в различных случаях вероятность перехода из совпадения в совпадение можно вычислить следующим образом: Вероятность перехода M2M = 1 - (вероятность перехода M2I + вероятность перехода M2D). Если вероятности перехода M2I и M2D могут быть сконфигурированы так, чтобы быть меньшими или равными значению ½, то в различных вариантах реализации приведенное выше уравнение может быть реализовано в аппаратном оборудовании таким образом, чтобы повысить общую эффективность и пропускную способность, например за счет переработки уравнения: Вероятность перехода M2M = (0,5 - вероятность перехода M2I) + (0,5 - вероятность перехода M2D). Эта перезапись уравнения позволяет получать M2M с использованием двух 64-элементных таблиц подстановки с последующим сумматором, где таблицы подстановки хранят результаты.

[00366] Более того, что касается определения вероятностей перехода инсерции в инсерцию и/или делеции в делецию, вероятности перехода I2I и D2D являются функциями от значений вероятности продолжения гэпа (GCP), вводимых в аппаратный ускоритель 8 HMM. В различных случаях эти значения GCP могут быть 6-битовыми значениями типа Phred, присваеваемыми каждому основанию рида. Значения I2I и D2D могут быть получены следующим образом: Вероятность перехода I2I = 10^(-(GCP(I) рида/10)) и вероятность перехода D2D = 10^(-(GCP(D) рида/10)). Аналогично некоторым другим вероятностям перехода, рассмотренным выше, значения I2I и D2D могут быть эффективно реализованы в аппаратном обеспечении и могут включать две таблицы подстановки (или одну таблицу подстановки с совместно используемыми ресурсами), например, в той же форме и с таким же контекстом, что и таблицы подставки перехода из совпадения к инделу, рассмотренные выше. То есть, каждая таблица постановки имеет 64 слова.

[00367] Кроме того, что касается определения вероятностей перехода инсерции и/или делеции в совпадение, вероятности перехода I2M и D2M являются функциями от значений вероятности продолжения гэпа (GCP), которые можно вычислить следующим образом: Вероятность перехода I2M = 1 - вероятность перехода I2I и вероятность перехода D2M = 1 - вероятность перехода D2D, где вероятности перехода I2I и D2D могут быть получены, как обсуждалось выше. Простая операция вычитания для реализации вышеуказанных уравнений может быть более дорогой по аппаратным ресурсам, чем просто реализация другой состоящей из 64 слов таблицы подстановки и использование двух ее экземпляров для реализации получения I2M и D2M. В таких случаях каждая таблица постановки имеет 64 слова. Конечно, во всех соответствующих вариантах реализации простые и сложные операции могут формироваться с помощью программного обеспечения, сконфигурированного подходящим образом.

[00368] На ФИГ. 13 приведена электрическая схема 17a для упрощенного вычисления вероятностей перехода HMM и значений Prior, как описано выше, которая поддерживает общую диаграмму переходов состояний, изображенную на ФИГ. 12. Как показано на

ФИГ. 13, в различных случаях представлена простая архитектура 17a программного ускорителя, причем ускоритель может быть выполнен с возможностью включения отдельных значений GOP для переходов инсерции и делеции и/или в ней могут быть отдельные значения GCP для переходов инсерции и делеции. В таком случае стоимость формирования семи уникальных вероятностей перехода и одного значения Prior за каждый тактовый цикл может быть сконфигурирована таким образом, как указано ниже: восемь таблиц подстановки по 64 слова, одна таблица подстановки на 128 слов и один сумматор.

[00369] Кроме того, аппаратное обеспечение 2, которое представлено в настоящем документе, может быть выполнена с возможностью вмещения стольких экземпляров 13 движка HMM, сколько возможно поместить на целевую микросхему (такую как FPGA, sASIC или ASIC). В таком случае стоимость реализации логики 17a формирования вероятностей перехода и значений Prior может быть существенно снижена относительно стоимостей, которые обеспечиваются приведенными ниже конфигурациями. Во-первых, вместо поддержки более общей версии переходов состояний, таких как показаны на ФИГ. 13, например, где могут быть отдельные значения для GOP(I) и GOP(D), в различных случаях можно предполагать, что значения GOP для инсерции и делеции одинаковые для данного основания. Это приводит к нескольким упрощениям аппаратного обеспечения, как указано выше.

[00370] В таких случаях можно использовать только одну таблицу подстановки объемом 64 слова для формирования значения M2Индел, заменяющего оба значения вероятности M2I и M2D, тогда как в более общем случае, как правило, используют две таблицы. Аналогичным образом для формирования значения вероятности перехода M2M можно использовать только одну таблицу подстановки объемом 64 слова, тогда как в общем случае могут использоваться две таблицы и сумматор, поскольку M2M может быть теперь вычислено как 1-2×M2Индел.

[00371] Во-вторых, можно сделать допущение, что зависимое от секвенатора значение GCP как для инсерции, так и для делеции, одинаковое, И что это значение не изменяется в течение задания 20 HMM. Это означает, что: вместо отдельных значений I2I и D2D можно вычислять одну вероятность перехода из индела в индел с использованием таблицы подстановки объемом 64 слова вместо двух таблиц; и вместо отдельных значений I2M и D2M можно вычислять одну вероятность перехода из индела в совпадение с использованием одной таблицы подстановки объемом 64 слова вместо двух таблиц.

[00372] Кроме того, можно сделать еще одно упрощающее допущение, которое предполагает, что значения из инсерции в инсерцию и из делеции в делецию (I2I и D2D) и от инсерции к совпадению и от делеции к совпадению (I2M и D2M) не только идентичны между переходами инсерции и делеции, но могут быть статическим для конкретного задания 20 HMM. Таким образом, всего в более общей архитектуре с вероятностями перехода I2I, D2D, I2M и D2M можно удалить четыре таблицы подстановки. В различных подобных случаях можно сделать так, чтобы статические вероятности из индела в индел и из индела в совпадение вводились посредством программного обеспечения или через параметр RTL (и чтобы их можно было программировать в двухпотоковом режиме в FPGA). В определенных случаях эти значения могут быть выполнены с возможностью двухпотокового программирования, и в определенных случаях может быть реализован режим тренировки, использующий тренировочную последовательность для дальнейшего улучшения точности вероятности перехода для данного прогона секвенатора или геномного анализа.

[00373] На ФИГ. 14 показано, как может выглядеть диаграмма новой логики 17b перехода состояния при реализации этих различных упрощающих допущений. А именно, на ФИГ. 14 показана упрощенная диаграмма переходов состояния HMM, изображающая взаимосвязи между GOP, GCP и вероятностями перехода с упрощениями, изложенными выше.

[00374] Аналогичным образом на ФИГ. 15 показана электрическая схема 17a,b для формирования вероятностей перехода и величин Prior HMM, которая поддерживает упрощенную диаграмму переходов состояний, изображенную на ФИГ. 14. На ФИГ 15 приведена реализация схемы данной диаграммы переходов. Таким образом, в различных случаях для аппаратного ускорителя 8 HMM стоимость формирования вероятностей переходов и Prior за каждый тактовый цикл сокращена на: две таблицы подстановки объемом 64 слова, одну таблицу подстановки объемом 128 слов.

[00375] Как указано выше, логика 15 управления движка выполнена с возможностью формирования виртуальной матрицы и/или прохода матрицы таким образом, чтобы достигать края полосы захвата, например, посредством высокоуровневых конечных автоматов, где результирующие данные могут быть окончательно суммированы, например, посредством логики 19 управления заключительным суммированием, и сохранены, например посредством логики put/get.

[00376] Соответственно, как показано на ФИГ. 16, в различных вариантах реализации предложен способ создания и/или прохождения матрицы 30 ячеек HMM. А именно, на ФИГ. 16 показан пример того, как логика 15 управления укорителем HMM проходит по виртуальным ячейкам в матрице HMM туда и обратно. Например, предположим для примера, что каждая операция умножения и сложения имеют задержку в 5 тактовых циклов, тогда худшая задержка вычислений обновления состояний M, I, D составит 20 тактовых циклов и будет иметь место при вычислении обновления M. В вычислениях обновления состояния I и D на половину меньше операций, то есть задержка для этих операций составляет 10 тактовых циклов.

[00377] Последствия задержки для операций вычисления M, I и D можно понять с помощь ФИГ. 16, на котором показаны различные примеры зависимостей данных между ячейками. В таких случаях информация о состоянии I, M и D данной ячейки подается в вычисления состояния D ячейки в матрице HMM, которая находится непосредственно справа (т.е. имеет то же самое основание рида, что и данная ячейка, но имеет следующее основание гаплотипа). Аналогичным образом информация о состоянии I, M и D данной ячейки подается в вычисления состояния I ячейки в матрице HMM, которая находится непосредственно ниже (т.е. имеет то же самое основание гаплотипа, что и данная ячейка, но имеет следующее основание рида). Поэтому в конкретных случаях состояния M, I и D текущей ячейки подаются в вычисления состояний D и I ячеек на следующей диагонали матрицы ячеек HMM.

[00378] Аналогичным образом состояния M, I и D данной ячейки подаются в вычисление состояния M ячейки, которая находится справа и снизу (т.е. имеет следующее основание гаплотипа И следующее основание рида). Эта ячейка фактически отдалена на две диагонали от ячейки, которую она снабжает (ввиду того, что вычисления состояний I и D опираются на состояния из ячейки, которая отдалена на одну диагональ). Это качество, когда вычисления состояний I и D, опирающихся на ячейки, отдаленные на одну диагональ, тогда как вычисления состояния M опираются на ячейки, отдаленные на две диагонали, благотворно влияет на разработку аппаратного обеспечения.

[00379] В частности, при таких конфигурациях вычисления состояний I и D могут быть выполнены с возможностью убыстрения вдвое (например, 10 циклов) по сравнению

с вычислениями состояния M (например, 20 циклов). Следовательно, если вычисления состояния M начинаются за 10 циклов до вычислений состояний I и D для той же ячейки, то вычисления состояний M, I и D для ячейки в матрице 30 HMM все завершатся одновременно. Кроме того, если проходить матрицу 30 по диагонали так, что полоса 35 захвата содержит примерно 10 ячеек (например, с охватом десяти оснований рида), то: Состояния M и D, создаваемые данной ячейкой в координатах (i, j), где i относится к гаплотипу, а j относится к риду, могут быть использованы в вычислениях состояния D ячейки (i+1, j), как только они пройдут весь вычислительный конвейер ячейки (i, j).

[00380] Состояния M и I, создаваемые данной ячейкой в координатах (i, j), где i относится к гаплотипу, а j относится к риду, могут быть использованы в вычислениях состояния I ячейки (i, j+1) через один тактовый цикл после того, как только они пройдут весь вычислительный конвейер ячейки (i, j). Аналогичным образом состояния M, I и D, создаваемые данной ячейкой в координатах (i, j), где i относится к гаплотипу, а j относится к риду, могут быть использованы в вычислениях состояния M ячейки (i+1, j+1) через один тактовый цикл после того, как только они пройдут весь вычислительный конвейер ячейки (i, j). В своей совокупности вышеизложенное означает, что для состояний M, I и D вдоль диагонали полосы захвата, которая простирается на длину полосы захвата, например, десять оснований, требуется очень маленькая специализированная память. В таком случае требуются лишь регистры для задержки на один тактовый цикл значений M, I и D ячейки (i, j), чтобы использовать их в вычислениях M в ячейке (i+1, j+1) M и в вычислениях I в ячейке (i, j+1) за один тактовый цикл. Более того, все это происходит неким волшебным образом, так как вычисления состояния M для данной ячейки начинаются за 10 тактовых циклов до вычислений состояний I и D для этой же ячейки, что естественным образом приводит к одновременному выводу новых состояний M, I и D для любой данной ячейки.

[00381] Ввиду вышеизложенного, и как показано на ФИГ. 16, логика 15 управления ускорителем HMM может быть выполнена с возможностью обработки данных в каждой ячейке виртуальной матрицы 30 с прохождением матрицы. В частности, в различных вариантах реализации операции начинаются в ячейке (0,0), причем вычисления состояния M начинаются за 10 тактовых циклов до начала вычислений состояний I и D. Следующей ячейкой должна быть ячейка (1,0). Однако результаты вычислений состояний I и D из ячейки (0,0) будут доступны с задержкой в десять циклов после их начала. Поэтому аппаратное обеспечение вставляет девять «мертвых» циклов в вычислительный конвейер. Это показано в виде ячеек с индексом гаплотипа ниже нуля на ФИГ. 16.

[00382] По завершении мертвого цикла, который имеет эффективную позицию ячейки в матрице (-9,-9), значения состояний M, I и D для ячейки (0,0) доступны. После этого они (например, выводы состояний M и D ячейки (0,0)) могут быть сразу же использованы для вычислений состояния D ячейки (0,1). Спустя один тактовый цикл значения состояний M, I и D из ячейки (0,0) могут быть использованы для начала вычислений состояния I ячейки (0,1) и вычислений состояния M ячейки (1,1).

[00383] Следующей ячейкой для прохождения может быть ячейка (2,0). Однако результаты вычислений состояний I и D из ячейки (1,0) будут доступны с задержкой в десять циклов после их начала. Поэтому аппаратное обеспечение вставляет восемь «мертвых» циклов в вычислительный конвейер. Это показано в виде ячеек с индексом гаплотипа ниже нуля, как на ФИГ. 16 вдоль той же диагонали, где находятся ячейки (1,0) и (0,1). По завершении мертвого цикла, который имеет эффективную позицию ячейки в матрице (-8,-9), значения состояний M, I и D для ячейки (1,0) доступны. После этого они (например, выводы состояний M и D ячейки (1,0)) сразу же используются для

вычислений состояния D ячейки (2,0).

[00384] Спустя один тактовый цикл значения состояний M, I и D из ячейки (1,0) могут быть использованы для начала вычислений состояния I ячейки (1,1) и вычислений состояния M ячейки (2,1). К тому же значения состояний M и D из ячейки (0,1) могут быть одновременно использованы для вычислений состояния D ячейки (1,1). Спустя один тактовый цикл значения состояний M, I и D из ячейки (0,1) используются для начала вычислений состояния I ячейки (0,2) и вычислений состояния M ячейки (1,2).

[00385] Теперь следующей ячейкой для прохождения может быть ячейка (3,0). Однако результаты вычислений состояний I и D из ячейки (2,0) будут доступны с задержкой в десять циклов после их начала. Поэтому аппаратное обеспечение вставляет семь «мертвых» циклов в вычислительный конвейер. Это опять же показано в виде ячеек с индексом гаплотипа ниже нуля на ФИГ. 16 вдоль той же диагонали, где находятся ячейки (2,0), (1,1) и (0,2). По завершении мертвого цикла, который имеет эффективную позицию ячейки в матрице (-7,-9), значения состояний M, I и D для ячейки (2,0) доступны. После этого они (например, выводы состояний M и D ячейки (2,0)) сразу же используются для вычислений состояния D ячейки (3,0). И, таким образом, начинается вычисление других десяти ячеек на диагонали.

[00386] Такая обработка может продолжаться до конца последней полной диагонали в полосе 35a захвата, что, в данном примере (с длиной рида 35 и длиной гаплотипа 14), произойдет после диагонали, которая начинается в ячейке с координатами гаплотипа и рида (13,0). После прохождения ячейки (4,9) на ФИГ. 16 следующей ячейкой для прохождения должна быть ячейка (13,1). Однако результаты вычислений состояний I и D из ячейки (12,1) будут доступны с задержкой в десять циклов после их начала.

[00387] Поэтому аппаратное обеспечение может быть выполнено с возможностью начала операций, связанных с первой ячейкой в следующей полосе 35 захвата, например, в ячейке с координатами (0, 10). После обработки ячейки (0, 10) может быть пройдена ячейка (13, 1). Затем проходят всю диагональ ячеек, начиная с ячейки (13, 1), пока не будет достигнута ячейка (5, 9). Аналогичным образом после прохождения ячейки (5, 9) следующей ячейкой для прохождения должна быть ячейка (13, 2). Однако, как и прежде, результаты вычислений состояний I и D из ячейки (12, 2) могут быть доступны с задержкой в десять циклов после их начала. Поэтому аппаратное обеспечение может быть выполнено с возможностью начала операций, связанных с первой ячейкой на второй диагонали следующей полосы 35b захвата, например, в ячейке с координатами (1, 10), за которой следует ячейка (0, 11).

[00388] После обработки ячейки (0, 11) может быть пройдена ячейка (13, 2) в соответствии со способами, описанными выше. Затем проходят всю диагональ 35 ячеек, начиная с ячейки (13, 2), пока не будет достигнута ячейка (6, 9). Кроме того, после прохождения ячейки (6, 9) следующей ячейкой для прохождения должна быть ячейка (13, 3). Однако и здесь результаты вычислений состояний I и D из ячейки (12, 3) могут быть доступны с периодом задержки в десять циклов после их начала. Поэтому аппаратное обеспечение может быть выполнено с возможностью начала операций, связанных с первой ячейкой на третьей диагонали следующей полосы 35c захвата, например, в ячейке с координатами (2, 10), за которой следуют ячейки (1, 11) и (0, 12) и т.п.

[00389] Это продолжается, как указано выше, до тех пор, пока не будет пройдена последняя ячейка в первой полосе 35a захвата (ячейка с координатами гаплотипа и рида (13, 9)), и в этот момент логика может полностью переключиться на прохождение второй диагонали во второй полосе 35b захвата, начиная с ячейки (9, 10). Схема, кратко

описанная выше, повторяется столько раз, сколько полос захвата из 10 оснований потребуется, пока не будет достигнута нижняя полоса 35c захвата (в данном примере это ячейки, которые связаны с основаниями рида, имеющими индекс 30 или больше).

[00390] Внизу полосы 35 захвата может быть вставлено больше мертвых ячеек, как показано на ФИГ. 16, в качестве ячеек с индексами рида больше 35 и индексами гаплотипа больше 13. Кроме того, в заключительной полосе 35c захвата может быть фактически добавлена дополнительная строка ячеек. Эти ячейки указаны на линии 35 на ФИГ. 16 и относятся к специальному тактовому циклу, в котором происходят заключительные операции суммирования в каждой диагонали заключительной полосы захвата. В этих циклах складываются состояния M и I ячеек непосредственно вверху, и этот результат сам суммируется с текущей заключительной суммой (которая инициализируется нулем на левом крае матрицы 30 HMM).

[00391] В контексте вышесказанного и с учетом ФИГ. 16 можно отметить, что в данном примере с ридом длиной 35 и гаплотипом длиной 14 имеются 102 мертвых цикла, 14 циклов, связанных с заключительными операциями суммирования, и 20 циклов задержки конвейера, что в итоге дает 102+14+20 = 146 циклов непроизводственных затрат ресурсов. Также можно отметить, что любого задания 20 HMM с длиной рида более 10 мертвые циклы в верхнем левом углу ФИГ. 16 не зависят от длины рида. Кроме того, можно отметить, что мертвые циклы в нижней и нижней правой части ФИГ. 16 зависят от длины рида, причем минимальное количество мертвых циклов для ридов равно mod(длина рида, 10) = 9, а минимальное количество мертвых циклов равно mod (длина рида, 10) = 0. Также можно отметить, что процент циклов непроизводственных затрат ресурсов от общего количества циклов оценки матрицы 30 HMM уменьшается по мере увеличения длин гаплотипов (матрица больше при частично фиксированном количестве циклов непроизводственных затрат ресурсов) или по мере увеличения длин ридов (примечание: это касается процента непроизводственных затрат ресурсов, связанных с заключительным суммированием в матрице, снижаемым по мере увеличения разности между длиной рида и количеством строк). С помощью таких данных гистограммы из анализов репрезентативных полных геномов человека было установлено, что прохождение матрицы HMM описанным выше образом приводите к менее чем 10% непроизводственных затрат ресурсов для обработки полного генома.

[00392] Для сокращения количества циклов непроизводственных затрат ресурсов можно также использовать дополнительные способы, в том числе приведенные ниже. Наличие специализированной логики для заключительных операций суммирования вместо использования сумматоров совместно с логикой вычисления состояний M и D. Этим устраняется одна строка матрицы 30 HMM. Использование мертвых циклов для начала операций матрицы HMM для следующего задания HMM в очереди.

[00393] Каждая группировка из десяти строк матрицы 30 HMM составляет «полосу 35 захвата» в функции ускорителя HMM. Следует отметить, что полоса захвата может быть увеличена или уменьшена для удовлетворения требований к эффективности и/или пропускной способности системы. Следовательно, длина полосы захвата может иметь длину от около пяти строк до менее чем около пятидесяти строк, скажем, от около десяти строк до около сорока пяти строк, например, от около пятнадцати или около двадцати строк до около сорока или около тридцати пяти строк, в том числе от около двадцати пяти строк до около тридцати строк.

[00394] С учетом исключений, отмеченных в разделе выше, относящиеся к циклам получения данных, которые иначе были бы мертвыми циклами на правом крае матрицы на ФИГ. 16, матрицу HMM можно обрабатывать по одной полосе захвата за раз. Как

показано на ФИГ. 16, состояния ячеек в нижней строке каждой полосы 35a захвата подают в логику вычисления состояний в верхней строке следующей полосы 35b захвата. Следовательно, возможно, потребуется сохранять (put) и извлекать (get) информацию о состоянии для этих ячеек в нижней строке, или на крае, каждой полосы захвата.

[00395] Логика для выполнения этого, может включать в себя одно или более из следующего: по завершении вычислений состояний M, I и D для ячейки в матрице 30 HMM с mod(индекс рида, 10) = 9 результат сохраняют в память для хранения состояний M, I, D. При начале вычислений состояний M и I (например, когда вычисления состояния D не требуют информации из ячеек сверху в матрице) для ячейки в матрице 30 HMM с ячейки с mod(индекс рида, 10) = 0 извлекают ранее сохраненную информацию о состояниях M, I и D из соответствующего места в памяти для хранения состояний M, I, D. Следует отметить, что в этих случаях при подаче значений состояний M, I и D в строку 0 (верхнюю строку) вместо вычисления состояний M и I в матрице 30 HMM использую просто заданное постоянное значение и его не нужно вызвать из памяти, что справедливо и в отношении значений состояний M и D, которые подают в столбец 0 (левый столбец) вычислений состояния D.

[00396] Как отмечено выше, ускоритель HMM может содержать или не содержать специализированные ресурсы для суммирования в аппаратном ускорителе HMM, который существует просто в целях заключительных операций суммирования. Однако, в конкретных случаях, как описано в настоящем документе, к нижней части матрицы 30 HMM может быть добавлена дополнительная строка, а тактовые циклы, связанные с этой дополнительной строкой, могут быть использованы для заключительных операций суммирования. Например, собственно суммирования можно достичь путем заимствования (например, как показано на ФИГ. 13) сумматора из логики вычисления состояния M для выполнения операции M+I, а также заимствования сумматора из логики вычисления состояния D для сложения вновь полученной суммы M+I с текущим накопленным значением заключительного суммирования. В таком случае логика управления, чтобы активировать операцию заключительного суммирования, может вклиниваться всякий раз, когда индекс рида, который направляет операцию прохождения HMM, равен длине последовательности рида, вводимой для задания. Эти операции можно наблюдать на линии 34 в направлении в низ матрицы 30 HMM, изображенной на ФИГ. 16.

[00397] Следовательно, как можно было заметить выше, в одной реализации определитель вариантов может использовать движки сопоставителя и/или выравнивателя для определения правдоподобия мест происхождения различных ридов, например, в отношении данного местоположения, такого как местоположение на хромосоме. В таких случаях определитель вариантов может быть выполнен с возможностью обнаружения лежащей в основе последовательности в этом местоположении, например, независимо от других областей, не примыкающих непосредственно к нему. Это, в частности, полезно и хорошо работает, когда интересующая область не похожа ни на никакую другую область генома на протяжении одного рида (ли пары ридов для секвенирования спаренных концов). Однако значительная часть генома человека не удовлетворяет этому критерию, что может создать проблемы с выполнением определения вариантов, например, с процессом реконструкции генома субъекта из ридов, полученных с помощью СНП.

[00398] В частности, хотя в секвенировании ДНК произошли громадные улучшения, определение вариантов остается трудной проблемой, по большей части вследствие избыточной структуры генома. Тем не менее, как описано в настоящем документе,

сложности, представляемые избыточностью генома, можно преодолеть, по меньшей мере частично, с использованием подхода на основе данных коротких ридов. Более конкретно, устройства, системы и способы их применения, описанные в настоящем документе, могут быть выполнены с возможностью концентрации на гомологичных или подобных областях, которые в противном случае могли бы отличаться низкой точностью определения вариантов. В определенных случаях такая низкая точность определения вариантов может объясняться трудностями, наблюдаемыми в картировании и выравниваниях ридов на гомологичные области, которые, как правило, могут приводить к риду с очень низким качеством рида (MAPQ). Соответственно, в настоящем документе предложены стратегические реализации, которые точно определяют варианты (ОНП, инделы и т.п.) в гомологичных областях, например, путем совместного рассмотрения информации, представленной в этих гомологичных областях.

[00399] Например, многие области генома являются гомологичными, например, они имеют почти идентичные копии, находящиеся повсюду в геноме, например во множестве местоположений, и в результате истинное исходное местоположение может отличаться высокой неопределенностью. А именно, если группа ридов картирована с низкой достоверностью, например, вследствие явной гомологии, типичный определитель вариантов может игнорировать и не обрабатывать эти риды, даже если они могут содержать полезную информацию. В других случаях, если рид ошибочно картирован (например, первичное выравнивание не является истинным источником рида), это может привести к ошибкам обнаружения. Точнее говоря, ранее реализованные технологии секвенирования с короткими ридами были подвержены этим проблемам, и традиционные способы обнаружения часто сохраняют полную неясность в отношении больших областей.

[00400] В некоторых случаях для смягчения данных проблем можно использовать секвенирование длинных ридов, однако это, как правило, стоит намного дороже и/или чаще приводит к ошибкам, занимает больше времени и/или отличается другими недостатками. Поэтому в различных случаях, возможно, будет полезно выполнить операцию совместного обнаружения во множестве областей, которая описана в настоящем документе. Например, вместо рассмотрения каждой области по отдельности и/или выполнения и анализа секвенирования длинных ридов можно использовать метод совместного обнаружения во множестве областей (MRJD), например, когда протокол MRJD рассматривает множество, например все, местоположений, из которых могла быть получена каждая группа ридов, и пытается обнаружить лежащие в основе последовательности вместе, например, совместно, используя всю имеющуюся информацию, что можно сделать вне зависимости от низких или ненормальных оценок достоверности и/или определенности.

[00401] Например, в случае диплоидного организма со статистически равномерным покрытием в анализе поиска вариантом можно выполнить байесовское вычисление методом перебора. Однако при вычислении MLRD методом перебора сложность вычисления быстро растет с количеством областей N и количеством K гаплотипов-кандидатов, которые нужно рассмотреть. В частности, чтобы рассмотреть все комбинации гаплотипов-кандидатов, количество решений для гаплотипов-кандидатов, для которых нужно вычислить вероятности, часто может увеличиваться экспоненциально. Например, как более подробно описано ниже, при реализации перебора ряд гаплотипов-кандидатов включают в себя множество активных позиций, причем в случае использования метода сборки на основе графа для формирования списка гаплотипов-кандидатов в операции поиска вариантов, например путем

построения графа де Брейна, как описано в настоящем документе, количество активных позиций является количеством независимых «пузырей» в графе. Следовательно, реализация такого вычисления методом перебора может быть непозволительно дорогой, и сами по себе байесовские вычисления методом перебора могут быть непозволительно сложными.

[00402] Соответственно, согласно одному аспекту, как показано на ФИГ. 17A, предложен способ снижения сложности таких вычислений методом перебора. Например, как описано выше, хотя скорость и точность секвенирования ДНК/РНК резко улучшились, особенно в отношении способов, описанных в настоящем документе, определение вариантов, например, процесс реконструкции генома субъекта из ридов, которые создает секвенатор, остается сложной проблемой, по большей части вследствие избыточной структуры генома. Устройства, системы и способы, описанные в настоящем документе, выполнены с возможностью снижения сложностей, предъявляемых избыточностью генома, с помощью подхода, основанного на данных коротких ридов, в отличие от секвенирования длинных ридов. В частности, в настоящем документе предложены способы выполнения обнаружения очень длинного рида с учетом гомологичных и/или подобных областей генома, которые обычно отличаются низкой точностью определения вариантов, без необходимости выполнения секвенирования длинного рида.

[00403] Например, в одном варианте реализации предложены система и способ для выполнения совместного обнаружения во множестве областей. А именно, в первом случае может быть выполнена обычная операция определения вариантов, например, с использованием способов, описанных в настоящем документе. В частности, обычный определитель вариантов может использовать референсную геномную последовательность, которая представляет все основания в модельном геноме. Эти референсы образуют остов анализа, с помощью которого геном субъекта сравнивают с референсным геномом. Например, как отмечалось выше, с помощью секвенатора нового поколения геном субъекта может быть разбит на подпоследовательности, например риды, обычно около 100-1000 оснований в каждом, причем эти риды могут быть картированы и выровнены на референс, что во многом напоминает сборку мозаики из фрагментов.

[00404] После того, как геном субъекта картирована и/или выровнен с использованием данного референсного генома для сравнения с фактическим геномом субъекта, можно определить, в какой степени и как геном субъекта отличается от референсного генома, например, последовательно для каждого основания. В частности, при сравнении генома субъекта с одним или более референсных геномов, например, одного основания за другим, анализ выполняют, итеративно перемещаясь вдоль последовательности и сравнивая одну последовательность с другими, чтобы определить, согласуются ли они или нет. Соответственно, каждое основание в пределах последовательности представляет позицию, которую нужно определить, например, как представлено позицией A на ФИГ. 18A.

[00405] А именно, для каждой позиции A референса, которую нужно определить относительно генома субъекта, скопление последовательностей, или ридов, будут картированы или выровнены таким образом, чтобы все риды из более крупного набора образов могли покрывать друг друга в любой данной позиции A. В частности, эта избыточная выборка может содержать ряд ридов, например, от одного до сотни или более, где каждый рид в скоплении имеет нуклеотиды, перекрывающие определяемую область. Следовательно, определение этих ридов от одного основания к другому

включает в себя формирование окна обработки, которое скользит вдоль последовательности, выполняя определения, причем длина окна, например, количество оснований, исследуемых в любой данный момент времени, образует активную область определения. Следовательно, окно представляет активную область оснований в определяемом образце, где определение включает в себя сравнение каждого основания в данной позиции, например, A, во всех ридах скопления в пределах активной области, при этом идентичность основания в данной позиции в ряде скоплений ридов обеспечивает доказательство истинной идентичности основания в данной определяемой позиции.

[00406] Для этой цели на основании соответствующей оценки достоверности MAPQ, получаемой для каждого сегмента рида, можно в целом определить, в рамках определенной оценки достоверности, что картирование или выравнивание было выполнено точно. Однако по-прежнему остается вопрос, каким бы незначительным он ни был, по поводу того, является ли картирование или выравнивание ридов точным или нет, и действительно ли одно или более ридов принадлежат еще какому-нибудь месту. Соответственно, согласно одному аспекту в настоящем документе предложены устройства и способы для улучшения достоверности при выполнении определения вариантов.

[00407] В частности, в различных случаях определитель вариантов может быть выполнен с возможностью осуществления одной или более операций совместного обнаружения во множестве областей, как описано в настоящем документе, что можно использовать для придания большей достоверности достижимых результатов. Например, в таком случае определитель вариантов может быть выполнен с возможностью анализа различных областей в геном для определения конкретных областей, которые представляются похожими. Например, как показано на ФИГ. 18A, могут существовать референсная область A и референсная область B, где используемые в качества референса области очень схожи друг с другом, например за исключением нескольких областей с несходством пар оснований, таких как где пример Ref A имеет «A», и пример Ref B имеет «T», но за пределами этих несходств в любом другом месте исследуемой области могут совпадать. Благодаря степени схожести эти две области, например, Ref A и Ref B, как правило, будут считаться гомологичными, или паралогичными, областями.

[00408] Как показано на фигуре, референсные области A и B похожи на 99%. Могут быть другие области, например, Ref C и Ref D, которые относительно похожи, например, похожи примерно на 93%, но по сравнению с похожестью на 99% между референсными областями A и B, референсные области C и D не будут считаться гомологичными, или по меньшей мере будет иметь меньше шансов реально быть гомологичными. В таком случае процедуры определения вариантов могут быть выполнены с возможностью адекватного определения различий между референсными областями C и D, но могут, в определенных случаях, иметь трудности с определением различий между высокогомологичными референсными областями A и B, например вследствие их высокой гомологии. В частности, ввиду степени несхожести референсных последовательностей A и B с референсными последовательностями C и D не следует ожидать, что картирование и выравнивание на референсную последовательность A или B, будет ошибочно принято за картирование на референсную последовательность C или D. Однако можно ожидать, что риды, которые картируются и выравниваются на референсную последовательность A, могут быть ошибочно картированы на референсную последовательность B.

[00409] Учитывая степень гомологии, ошибочное картирование между областями

A и B может быть вполне вероятным. Соответственно, для повышения точности, было бы целесообразно, чтобы была в состоянии различать и/или учитывать разницу между гомологичными областями, например, при выполнении процедуры картирования, выравнивания и/или определения вариантов. А именно, при формировании скопления ридов, которые картируют или выравнивают на область в пределах Ref A, и при формировании скопления ридов, которые картируют и выравнивают на область в пределах Ref B, любое из ридов может быть в действительности ошибочно картировано на неверное место и, поэтому, чтобы добиться более высокой точности, при выполнении операций определения вариантов, описанных в настоящем документе, эти гомологичные области и риды, сопоставляемые и выравнивание не них, должны рассматриваться вместе, например, в протоколе совместного обнаружения, таком как протокол совместного обнаружения во множестве областей, который описан в настоящем документе.

[00410] Соответственно, в настоящем документе представлены устройства, системы, а также способы их использования, которые относятся к совместному обнаружению во множестве областей (MRJD), например когда множество, например все, ридов из различных скоплений различных выявленных гомологичных областей рассматривают вместе, например, когда вместо выполнения одно определения вариантов для каждого местоположения осуществляют совместное обнаружение для всех местоположений, которые представляются гомологичными. Выполнение таких совместных определений обладает преимуществом, так как прежде чем пытаться проделать определение для каждого референса по отдельности, сначала нужно было бы определить, на какую область какого референса различные исследуемые риды действительно картируются и выравниваются, а это по своей сути точно не известно, и именно эту проблему решает предлагаемое совместное обнаружение. Следовательно, поскольку области двух референсов настолько похожи, очень трудно определить, какие риды на какие области картируются. Однако, если определение этих областей выполняется совместно, нет необходимости принимать предварительное решение о том, какие гомологичные риды на какую референсную область картируются, Поэтому при выполнении совместного определения можно предположить, что любые риды в скоплении области на одной референсе, например, A, которые гомологичны другой области на втором референсе, например, B, могли бы принадлежать либо Ref. A, либо Ref. B.

[00411] Следовательно, если требуется, в дополнение к алгоритму определения вариантов, реализованному в устройствах, системах и способах, изложенных в настоящем документе, может быть реализован протокол MRJD. Например, за одну итерацию алгоритм определения вариантов принимает во внимание подтверждающие данные, представленные в картированных и/или выровненных ридах для данной области в геноме образца и референсном геноме, на основе сравнения с референсным геномом анализирует возможность фактического наличия в геноме образца того, что могло бы в нем присутствовать, и с учетом данного доказательства принимает решение по поводу того, как образец действительно отличается от референса, например, с учетом данного доказательства алгоритм определения вариантов определяет наиболее вероятный ответ, в чем разница между ридом и референсом. Однако MRJD является также алгоритмом, который может быть реализован наряду с алгоритмом определения вариантов, причем MRJD выполнен с возможностью оказания помощи определителю вариантов в более точном определении, является ли наблюдаемое отличие, например, в риде субъекта, действительно истинным отклонением от референса.

[00412] Соответственно, первый этап в анализе MRJD включает в себя выявление

гомологических областей на основе процента соответствия между последовательностью в множестве областей одного или более референсов, например, Ref. A и Ref. B, и последовательностями скоплений в одной или более областей ридов субъекта. В частности, Ref. A и Ref. B могут быть в действительности диплоидными формами одного и того же генетического материала, например, когда существуют два копии данной области хромосомы. Следовательно, при анализе диплоидных референсов в различных позициях Ref A может иметь один конкретный нуклеотид, а в той же самой позиции в Ref. B может присутствовать другой нуклеотид. В данном примере Ref. A и Ref. B являются гомозиготными в позиции A для «A». Однако, как показано на ФИГ. 18A, ДНК субъекта является гетерозиготной в этой позиции A, например, когда по отношению к ридам скопления области Ref. A один аллель хромосомы субъекта имеет «A», а другой аллель имеет «C», а что касается Ref. B, другая копия хромосомы субъекта имеет «A» в обоих аллелях в позиции A. Это тоже приобретает более сложный характер, когда анализируемая проба содержит мутацию, например, в одной из тех естественным образом возникающих переменных позициях, таких как гетерозиготный ОНП в позиции A (не показан).

[00413] Как показано на Ref. A, изображенном на ФИГ. 18B, в позиции A образец субъекта может содержать риды, которые указывают на наличие гетерозиготности в позиции A, например, когда некоторые из ридов содержат «C» в этой позиции, а некоторые из ридов указывают «A» в этой позиции (например, гаплотип$_{a1}$ = «A», $H_{a2}$ = «C»); что касается Ref. B, риды в позиции A указывают гомозиготность, например, когда все риды в скоплении имеют «A» в этой позиции (например, $H_{b1}$ = «A», $H_{b2}$ = «A»). Однако MJRD преодолевает эти трудности за счет одновременного выполнения совместного определения путем анализа всех ридов, которые картируются на обе области референса, с учетом при этом возможности того, что риды могут быть в неверном местоположении. После выявления различных гомологичных областей следующим этапом является определение соответствия между гомологичными референсными областями и затем, в соответствии с MRJD, определение с помощью сопоставителя и/или выравнивателей того, где различные подходящие риды «предположительно картируются» между двумя гомологичными областями, может быть отброшено, а, вернее, все риды в любом из скоплений в этих гомологичных областях могут быть рассмотрены совместно с учетом знания того, что любое из этих ридов может принадлежать любой из сравниваемых гомологичных областей. Следовательно, вычисления для определения этих совместных определений, как подробно изложено ниже, учитывают возможность того, что любое из этих ридов поступает из любой гомологических референсных областей и, где применимо, из любого из гаплотипов любой из референсных областей.

[00414] Необходимо отметить, что хотя вышесказанное касалось множества областей гомологии в пределах референса, тот же самый анализ может быть также применен к обнаружению одной области. Например, как показано на ФИГ. 18B, даже если область одна, в любой данной области могут присутствовать два отдельных гаплотипа, например, $H_1$ и $H_2$, которые генетический образец субъекта может иметь для конкретной области, а поскольку они являются гаплотипами, они, скорее всего, будут очень похожи друг на друга. Следовательно, если бы эти позиции анализировались отдельно друг от друга, было бы трудно определить, имеются ли здесь учитываемые истинные вариации. Таким образом, вычисления, выполняемые в отношении гомологичных областей, также полезны для негомологичных областей, так как любая конкретная область, вероятно,

будет диплоидом, например, имеющим как первый гаплотип ($H_1$), так и второй гаплотип ($H_2$), и поэтому совместный анализ этих областей улучшит точность системы. Аналогичным образом в случае области с двумя референсами, например, гомологичной области, как описано выше, определяют именно $H_{A1}$ и $H_{A2}$ для первой области и $H_{A1}$ и $H_{A2}$ для второй области (что равносильно двум нитям для каждой хромосомы и двум областям для каждой нити = 4 диплоидным типам, как правило).

[00415] Соответственно, MRJD можно использовать для определения первоначального ответа по поводу одной или более, например, всех, гомологичных областей, а затем можно применить обнаружение одной области обратно к одной или более, например, всем, одинарным или негомологичным областям, например, с использованием того же самого базового анализа, и, таким образом, можно достичь большей точности. Следовательно, можно также выполнить несовместное определение одной области. Например, что касается определения одной области, для гаплотипов-кандидатов $H_{A1}$ в текущих итерациях референсный геном может быть длиной около 300-500 пар оснований, и сверху референса строят граф, например, де Брейна, как показано на ФИГ. 18C, например, из K-меров из ридов, где любое местоположение, которое отличается от референса, образует отклоняющийся путь или «пузырь» на графе, из которого выделяют гаплотипы, где каждый выделенный гаплотип, например, отклоняющийся путь, образует потенциальную гипотезу о том, что могло быть на одной из двух нитей хромосомы в конкретном месте исследуемой активной области.

[00416] Однако если имеется много отклоняющихся путей, например, по всему графу образуется много пузырей, как показано на ФИГ. 18C, и выделено большое количество гаплотипов, то можно ввести отсечку по максимуму для поддержания управляемости вычислений. Отсечка может быть на любом статистически значимом количестве, таком как 35, 50, 100, 125-128, 150, 175, 200 или более и т.п. Тем не менее, в определенных случаях может быть рассмотрено большее количество, например, все, гаплотипы.

[00417] В таких случаях вместо выделения полных гаплотипов от истока до стока с начала до конца, например, от начала последовательности до ее конца, выделять нужно только последовательности, связанные с отдельными пузырями, например, выравнивать на референс нужно только пузыри. Соответственно, из DBG выделяют пузыри, последовательности выравнивают на референс и из этих выравниваний можно определить конкретные ОНП, инсерции, делеции и т.п. с точки зрения того, почему последовательности различных пузырей отличаются от референса. Следовательно, в этом смысле все разные гипотетические гаплотипы для анализа можно получить путем смешивания и сопоставления последовательностей, относящихся ко всем различным пузырям в различных комбинациях. При таком подходе не требуется перечислять все гаплотипы, подлежащие выделению. Эти способы для выполнения совместного обнаружения во множестве областей описаны более подробно ниже в настоящем документе.

[00418] Кроме того, гипотетически, даже если все из этих гаплотипов-кандидатов могут быть проверены, алгоритм выращивания дерева может быть выполнен там, где создаваемый граф начинает выглядеть как растущее дерево. Например, ветвящийся древовидный граф совместных гаплотипов/диплотипов может быть построен таким образом, что по мере роста дерева лежащий в основе алгоритм функционирует как для выращивания, так и для обрезания дерева одновременно по мере того, как производится все больше и больше вычислений и становится очевидно, что всевозможные разные гипотезы-кандидаты просто слишком неправдоподобные. Следовательно, по мере

выращивания и обрезания дерева не все гипотетические гаплотипы нужно вычислять.

[00419] А именно, что касается функции выравнивания дерева, когда существует несогласованность между двумя референсами или между референсами и ридами по поводу того, какое основание присутствует в данных решаемых позициях, необходимо определить, какое основание действительно принадлежит какой позиции, и ввиду таких несогласованностей необходимо определить, какие различия могут быть вызван ОНП, инделами и т.п., а какие являются ошибками машины. Соответственно, при выращивании дерева, например, путем выделения пузырей из графа де Брейна, например, с помощью выравнивания Смита-Ватермана или Нидлмана-Вунша, и позиционирования их в пределах появляющегося древовидного графа, каждый пузырь, подлежащий выделению, становится в древовидном графе событием, которое представляет возможные ОНП, инделы и/или другие отличия от референса (см. ФИГ. 18C).

[00420] В частности, на DBG пузыри представляют несовпадения с референсом, например, представляющие инделы (когда основания были добавлены или удалены), ОНП (когда основания отличаются) и т.п. Поэтому по мере выравнивания пузырей на референс (-ы) различные отличия между ними распределяют по категориям как события и формируют список различных событий, например пузырей. В силу этого определение тогда превращается в следующее: какая комбинация возможных событий, например, возможных ОНП и инделов, привела к фактическим вариациям в генетической последовательности субъекта, например, является истинной в каждом из фактических различных гаплотипах, например, 4, исходя из вероятности. Более конкретно, любой один кандидат, например, совместный диплотип-кандидат, формирующий корень $G_0$ (представляющий события для данного сегмента) может иметь 4 гаплотипа, и какой из четырех гаплотипов будет формировать идентифицированное подмножество событий.

[00421] Однако, как показано на ФИГ. 18D, при выполнении функции выращивания и/или обрезания дерева полный список всего подмножества комбинаций событий может быть, а может и не быть, определен весь сразу. Вместо этого определение начинают в одной позиции $G_0$, например, с одного события, и из него выращивают дерево по одному событию за раз, что в результате функция обрезания может оставить различные маловероятные события нерешенными. Таким образом, что касается функции выращивания дерева, то, как показано на ФИГ. 18D, вычисление начинается с определения гаплотипов, например, $H_{A1}$, $H_{A2}$, $H_{B1}$, $H_{B2}$ (для диплоидного организма), причем начальные гаплотипы считаются все нерешенными по отношению к их соответствующим референсам, например, Ref. A и Ref. B, по существу ни с одним из присутствующих событий.

[00422] Соответственно, первоначальной отправной точкой является корень дерева, представляющий собой $G_0$, и совместный диплотип, у которого все события нерешенные. Затем конкретное событие, например начальный пузырь, выбирают в качестве начала для определения, тем самым начальное событие должно быть решено для всех гаплотипов, где это событие может быть первой точкой расхождения с референсом, например, по отношению к потенциальному присутствию ОНП или индела в позиции один. Как объяснено на ФИГ. 18E, в позиции один имеется событие или пузырь, например, ОНП, где «C» заменено на «A», так что референс имеет «A» в позиции один, а исследуемый рид имеет «C». В таком случае, так как для этой позиции в скоплении имеются 4 гаплотипа, и каждый может иметь либо «A», как референс, или событие «C», потенциально существуют $2^4 = 16$ возможностей для решения этой позиции. Следовательно, вычисление переходит сразу с корня на 16 ветвей, представляющих

потенциальные разрешения для каждого события в позиции один.

[00423] Поэтому, как показано на ФИГ. 18D, можно определить все потенциальные последовательности для всех четырех гаплотипов, например, $H_{A1}$, $H_{A2}$, $H_{B1}$, $H_{B2}$, где в позиции один имеется либо «A», что соответствует референсу, либо событие «C», указывающее на наличие ОНП, для этого одного события, где событие «C» определяется путем изучения различных путей пузыря через граф. Итак, для каждой ветви или дочернего узла каждая ветвь может отличаться на основе правдоподобия основания в позиции один, соответствующего или вытекающего из референса, тогда как остальные события остаются нерешенными. Затем этот процесс будет повторен для каждого узла ветвления и для каждого основания в пределах пузырей вариации, чтобы решить все события для всех гаплотипов. Следовательно, можно пересчитать вероятности наблюдения любого конкретного рида при условии различных потенциальны гаплотипов.

[00424] В частности, для каждого узла могут быть четыре гаплотипа, и каждый гаплотип можно сравнить с каждым ридом в скоплении. Например, в одном варианте реализации движка Смита-Ватермана, Нидлмана-Вунша и/или HMM анализирует каждый узел и рассматривает каждый из четырех гаплотипов для каждого узла. Следовательно, формирование каждого узла активирует движок Смита-Ватермана и/или HMM для анализа этого узла путем рассмотрения всех гаплотипов, например, 4, для этого узла в сравнении с каждым ридом, где движок Смита-Ватермана и/или HMM рассматривает по одному гаплотипу на один рид для каждого из гаплотипов и каждого из ридов для всех жизнеспособных узлов.

[00425] Таким образом, если в данном примере предположить, что имеется гетерозиготный ОНП «C» для одной области одного гаплотипа, например, одна нить хромосомы имеет «C», а все остальные нити имеют в этой позиции другие основания, например, они все совпадают с референсом «A», то следует ожидать, что все риды в скоплении поддерживают эти данные, например, большинство из них имеют «A» в позиции один, а меньшинство, например, около ¼, ридов имеют «C» в позиции один в случае истинного узла. Поэтому, если любые ранее наблюдаемые события в другом узле демонстрирую множество «C» в позиции один, то этот узел вряд ли будет истинным узлом, например, будет иметь низкую вероятность, поскольку в скоплении не будет достаточно ридов с «C» в этой позиции, чтобы сделать его появление вероятным. А именно, более вероятно будет то, что существование «C» в этой позиции в исследуемых ридах свидетельствует об ошибке секвенирования или другой ошибке исследования, а не о наличии истинного гаплотипа-кандидата. Следовательно, если определенные узлы обрываются, имея низкие вероятности по сравнению с истинным узлом, причина в том, что они не поддерживаются большинством ридов, например в скоплении, и поэтому эти узлы можно обрезать, тем самым отбросив узлы низкой вероятности, но при этом сохранив истинные узлы.

[00426] Соответственно, после того, как позиция один события определена, можно определить следующую позицию события, и описанный здесь процесс может быть потом повторен для этой новой позиции по отношению к любому из выживших узлов, которые до сих пор не обрезаны. В частности, событие два может быть выбрано из существующих доступных узлов, и это событие может служить в качестве корня $G_1$ для определения вероятной идентичности основания в позиции два, например, опять путем определения новых гаплотипов, например, 4, а также их различных ветвей, например, 16, объясняющих возможные вариации в отношении позиции 2. Таким образом, повторяя этот же самый процесс теперь можно решить событие 2. Следовательно, как показано

на ФИГ. 18D, после того, как позиция 1 определена, можно выбрать новый узел для позиции 2 и рассмотреть его 16 гаплотипов-кандидатов. В таком случае можно определить кандидатов для каждого $H_{A1}$, $H_{A2}$, $H_{B1}$, $H_{B2}$, но теперь, поскольку позиция 1 уже решена в отношении определения идентичности нуклеотида для каждого гаплотипа в позиции 1, именно позиция 2 будет теперь решена для каждого гаплотипа в позиции 2, как указано на ФИГ. 18D, показывающей позицию 2.

[00427] По завершении этого процесса после того, как все события обработаны и решены, например, включая дочерние узлы и их дочерние узлы, которые не были обрезаны, можно изучить необрезанные узлы дерева и на основе оценок вероятности определить, какое дерево представляет совместный диплотип, например, какая последовательность имеет наивысшую вероятность быть деревом. Следовательно, таким образом, ввиду функции обрезания, нет необходимости в построении полного дерева, например, большая часть дерева оборвется, будучи обрезанной по мере продолжения анализа, поэтому общий объем вычислений сильно сокращен по сравнению с функциями без обрезания, хотя он и существенно больше, чем при выполнении определения несовместного диплотипа, например определения одной области. Соответственно, представленные модули аналитики могут с высокой степенью точности определять и решать две или более областей высокой гомологии, например, используя анализ совместного диплотипа, где традиционные способы просто не в состоянии вообще решать такие области, например, вследствие ложно положительных вариантов и неразрешимости.

[00428] В частности, различные реализации определителя вариантов могут быть выполнены с возможностью просто невыполнения анализа на областях высокой гомологии. Представленные итерации преодолевают эти и другие подобные проблемы в данной области техники. Более конкретно, представленные устройства, системы и способы их использования могут быть выполнены с возможностью рассмотрения большей части, например, всех, гаплотипов, несмотря на наличие областей высокой гомологии. Конечно, скорость этих вычислений может быть дополнительно увеличена за счет невыполнения вычислений, когда можно определять, что результаты таких вычислений имеют низкую вероятность в плане истинности, например за счет реализации функции обрезания, как описано в настоящем документе.

[00429] Преимущество этих конфигураций, например, решение и обрезание совместного диплотипа, состоит в том, что размер окна активной области, например, анализируемых оснований, может быть увеличен от около нескольких сотен обрабатываемых оснований до нескольких тысяч или даже десятков тысяч оснований, которые могут быть обработаны вместе, например в одной непрерывной активной области. Такое увеличение размера активного окна анализа позволяет рассматривать больше подтверждающих данных при определении идентичности любого конкретного нуклеотида в любой данной позиции, тем самым обеспечивая больше контекста, в рамках которого можно выполнить более точное определение идентичности нуклеотида. Аналогичным образом более широкий контекст позволяет лучше связывать вместе подтверждающие данные при сравнении одного или более ридов, покрывающих одну или более областей, которые имеют одно или более отклонений от референса. Следовательно, таким образом одно событие может быть соединено с другим, которое само может быть соединено с еще одним событием и т.д., и на основе этих связей можно выполнять более точное определение по отношению к данному конкретному рассматриваемому событию, тем самым обеспечивая информативность подтверждающих данных из еще более отдаленных, например, от сотен до тысяч или более отдаленных

оснований, при выполнении настоящего определения вариантов (несмотря на тот факт, что любой данный рид, как правило, имеет длину в сотни оснований), и благодаря этому еще больше повышая точность процессов, описанных в настоящем документе.

[00430] В частности, подобным образом можно добиться дальнейшего включения в активную область от тысяч до десятков тысяч, даже сотен тысяч или более оснований и, следовательно, можно избежать способа формирования графа де Брейна путем выделения всех гаплотипов, поскольку нужно исследовать лишь ограниченное число гаплотипов, тех, что имеют пузыри, которые могут быть жизнеспособными, причем даже те из них, которые жизнеспособны, могут быть обрезаны после того, как становятся нежизнеспособными, а для остающихся жизнеспособными можно использовать связывание в цепочку, чтобы улучшить точность осуществляемого в конечном счете определения вариантов. Все это стало возможным благодаря квантовому и/или аппаратному вычислению. Это также может быть выполнено, но медленнее, в программном обеспечении с помощью ЦПУ или ГПУ,

[00431] Касательно приведенных выше примеров необходимо отметить, что вероятности входных данных, например, ридов, определяются с учетом гипотетических гаплотипов, созданных с помощью граф де Брейна. Однако, возможно, также будет полезно использовать теорему Байеса, например, для определения вероятности ридов с учетом совместного диплотипа вплоть до противоположной вероятности определения из теории совместного диплотипа наилучшего соответствия с учетом оцениваемых ридов и подтверждающих данных. Соответственно, как показано на ФИГ. 18С, на основе сформированного графа де Брейна после того, как выполнено совместное обнаружение во множестве областей и/или обрезание, будет получен набор потенциальных гаплотипов, и затем эти гаплотипы будут проверены на соответствие фактическим ридами субъекта. А именно, каждое горизонтальное поперечное сечение представляет гаплотип, например, B1, который может быть затем подвергнут обработке с помощью другого протокола HMM для проверки на соответствие ридам, чтобы определить вероятность конкретного рида при условии гаплотипа B1.

[00432] Однако, в определенных случаях гаплотип, например, B1, может быть еще не полностью определен, однако выполнение HMM все же может принести пользу, и в таком случае можно выполнить модифицированное вычисление HMM, например, операцию частично определенной HMM, (PD)-HMM, обсуждаемую ниже, когда в гаплотипе допускается наличие неопределенных вариантов, например, ОНП и/или инделов, которые еще предстоит определить, и по сути это вычисление подобно вычислению наилучшей возможной вероятности достижимого ответа с учетом любой комбинации вариантов в нерешенных позициях. Таким образом, это еще больше облегчает функцию итеративного выращивания дерева, где фактическое выращивание дерева, например, выполнение операций PD-HMM, не должно ограничиваться только теми вычислениями, где все возможные варианты известны. Следовательно, таким образом можно выполнить ряд вычислений PD-HMM итеративным путем, чтобы вырастить дерево узлов несмотря на тот факт, что в конкретных гаплотипах-кандидатах все еще имеются неопределенные области неизвестных возможных событий, и так, где появляется возможность обрезать дерево, ресурсы PD-HMM могут быть плавно переключены с вычисления обрезанных узлов на обработку только тех возможностей, которые имеют наивысшую вероятность успешного охарактеризования истинного генотипа.

[00433] Соответственно, при определении вероятности конкретного основания действительно присутствующего в любой одной позиции, идентичность основания в

этой позиции может быть определена на основе идентичности в этой позиции в каждой области хромосомы, например, в каждом гаплотипе, который представляет жизнеспособный кандидат. Следовательно, любой кандидат, определение которого выполняется, является идентичностью данного основания в рассматриваемой позиции в каждом из четырех гаплотипов одновременно. В частности, определяется именно вероятность наблюдения ридов каждом из скоплении при условии определенного правдоподобия. А именно, каждый кандидат представляет совместный диплотип, и в силу этого каждый кандидат содержит около четырех гаплотипов, которые могут быть представлены в следующем уравнении как G = генотип, где G = четырем гаплотипам одной диплоидной области хромосомы генома, например совместного диплотипа. В таком случае вычислять нужно именно вероятность фактического наблюдения каждого из выявленных оснований-кандидатов рида последовательностей в скоплениях в предположении, что они действительно истинные. Это начальное определение может быть выполнено с помощью вычисления гаплотипа HMM, как указано ниже в настоящем документе.

[00434] Например, «СовместныйДиплотип» = 4 гаплотипа: (область A: $H_{A1}H_{A2}$, и область B:·$H_{B1}H_{B2}$) = G → P(R/G) как определено с помощью HMM (модель ошибок) = ПР(r/G) =

$$\frac{P(r/\text{HA1}) \ + \ \ldots \ + \text{P(r/Hn)}}{n}$$

[00435] Следовательно, если предполагается, что конкретный гаплотип $H_{a1}$ является истинной последовательностью в этой области, и рид происходит оттуда, то каковы шансы, что данный $H_{a1}$ этой последовательности рида действительно наблюдался. Соответственно, вычислитель HMM работает над тем, чтобы определить, в предположении, что гаплотип $H_{a1}$ истинным, каково правдоподобие реального наблюдения данной исследуемой последовательности рида.

[00436] А именно, если рид действительно совпадает с гаплотипом, вероятность, конечно, будет очень высокой. Однако, если конкретный исследуемый рид не совпадает с гаплотипом, то любое отклонение от него должно объясняться ошибкой исследования, например, ошибкой секвенирования или оборудования для секвенирования, а не действительной вариацией. Следовательно, вычисление HMM является функцией от моделей ошибок. А именно, возникает вопрос, какова вероятность необходимой комбинации ошибок, которые должны были бы произойти, чтобы наблюдались эти конкретные анализируемые риды. Следовательно, в данной модели рассматривается не только одна область, но и множество позиций в множестве областей на множестве нитей одновременно (например, вместо рассмотрения, скорее всего, двух гаплотипов в одной области, теперь одновременно рассматривается вероятность четырех гаплотипов для любой данной позиции, одновременно, с использованием всех данных ридов из всех исследуемых областей). Теперь рассмотрим более подробно эти процессы, например, обрезание дерева, совместное обнаружение во множестве областей и PD-HMM.

[00437] А именно, как показано на ФИГ. 17 и 18, предложена цепочка обработки высокого уровня, например, когда цепочка обработки может включать в себя один или более из следующих этапов: выявление и ввод гомологичных областей, выполнение предварительной обработки входных гомологичных областей, выполнение обнаружения очень длинного рида (VLRD) или совместного обнаружения во множестве областей (MJRD) и вывод файла определения вариантов. В частности, что касается выявления гомологичных областей, в качестве первичных входных данных в движок обработки

совместного обнаружения во множестве областей, реализующий алгоритм MRJD, может быть введен файл картированных, выровненных и/или сортированных данных SAM и/или BAM, например, CRAM. Движок обработки MRJD может быть частью интегральной схемы, например, ЦПУ, и/или ГПУ, и/или квантовой вычислительной платформы, исполняющей программное обеспечение, например, квантовый алгоритм, или может быть реализован в FPGA, ASIC и т.п. Например, описанный выше сопоставитель и/или выравниватель может быть использован для формирования файла CRAM и настроен для вывода N вторичных выравниваний для каждого рида вместе с первичными выравниваниями. Затем эти первичные и вторичные выравнивания могут быть использованы для определения списка гомологичных областей, причем гомологические области могут вычислены на основе определяемого пользователем порога подобия между N областями референсного генома. Этот список выявленных гомологичных областей может быть затем подан на стадию предварительной обработки соответствующим образом сконфигурированного модуля MRJD.

[00438] Соответственно, на стадии предварительной обработки для каждого набора гомологичных областей сначала может быть сформировано совместной скопление, например, с помощью первичных выравниваний из одной или более, например каждой из, областей в наборе. См., например, ФИГ. 19. Затем с помощью этого совместного скопления может быть сформирован список активных/кандидатов позиций вариантов (ОНП/инделов), с помощью которого каждый из этих вариантов-кандидатов может быть обработан и оценен с посредством движка (-ов) предварительной обработки MRJD. Чтобы уменьшить сложность вычислений, можно вычислить матрицу связности, с помощью которой можно определить порядок обработки вариантов-кандидатов.

[00439] В таких вариантах реализации алгоритм совместного обнаружения во множестве областей оценивает каждый выявленный вариант-кандидат на основе порядка обработки, определенного в сформированной матрице связности. Во-первых, можно сформировать и выдать один или более совместных диплотипов-кандидатов ($G_i$) для варианта-кандидата. Затем можно вычислить апостериорные вероятности каждого совместного диплотипа ($P(G_i|R)$). На основе этих апостериорных вероятностей можно вычислить матрицу генотипа. Затем N диплотипов с самыми низкими апостериорными вероятностями можно обрезать, чтобы сократить вычислительную сложность расчетов. После этого можно включить следующий вариант-кандидат, который обеспечивает подтверждающие данные для текущего оцениваемого варианта-кандидата, и повторить вышеописанный процесс. Включив информацию, например, из одного или более, например, всех, вариантов-кандидатов из одной или более, например, всех, областей в наборе гомологических областей для текущего варианта, можно выполнить определение вариантов на основе заключительной матрицы генотипирования. Таким образом, каждую из активных позиций можно оценить, как описано выше, и тем самым получить окончательный файл VCF.

[00440] В частности, как показано на ФИГ. 17B, этап обработки MRJD может быть реализован, например, включением одного или более следующих этапов или блоков. Загружают выявленное и собранное совместное скопление, затем из собранного совместного скопления создают список вариантов-кандидатов и вычисляют матрицу связности. В частности, в различных вариантах может быть осуществлен метод предварительной обработки, например, для выполнения одной или более операций определения вариантов, таких как операция совместного обнаружения множества ридов. Такие операции могут включать в себя один или более блоков предварительной обработки, в том числе: этапы, относящиеся к загрузке совместных скоплений,

формированию из совместного скопления списка вариантов-кандидатов и вычисление матрицы связности. Теперь рассмотрим подробнее каждый из блоков и связанных с ними потенциальных этапов.

[00441] А именно, в процедуру анализа может быть включен первый блок предварительной обработки совместного скопления. Например, для идентифицируемого промежутка могут быть выделены различные референсные области, например, из картированных и/или выровненных ридов. В частности, с помощью списка гомологичных областей можно сформировать совместное скопление для каждого набора гомологичных областей. После этого с помощью задаваемого пользователем промежутка можно выделить N референсных областей, соответствующих N гомологичным областям в наборе. Затем можно выровнять одну или более, например, все референсные области, например с помощью выравнивания Смита-Ватермана, что можно использовать для формирования системы универсальных координат всех оснований в N референсных областях. Далее, затем можно выделить из входного файла SAM или BAM все первичные риды, соответствующие каждой области, и картировать на универсальные координаты. Это картирование можно выполнить, как описано в настоящем документе, например, с помощью информации о выравнивании (CIGAR), представленной в файле CRAM для каждого картирования. В условиях, когда некоторые риды не были ранее картированы, эти риды можно картировать и/или выровнять, например, с помощью алгоритма Смита-Ватермана, на их соответствующую референсную область.

[00442] Более конкретно, после того, как совместное скопления сформировано и загружено (см., например, ФИГ. 19), можно создать список вариантов-кандидатов, например из совместного скопления. Например, чтобы выделить различные варианты-кандидаты (ОНП/инделы), которые могут быть выявлены из совместного скопления, можно создать граф де Брейна (DBG) или другой граф сборки. После создания DBG на нем можно найти различные пузыри, чтобы получить список вариантов-кандидатов.

[00443] В частности, при наличии всех ридов можно построить граф, используя каждую референсную область в качестве остова. Затем все позиции выявленного варианта-кандидата можно выровнять на универсальные координаты. Затем можно вычислить матрицу связности, которая определяет порядок обработки активных позиций, который может быть функций от длин ридов и/или размера инсерции. На ФИГ. 19 показан упоминаемый в настоящем документе пример совместного скопления из двух гомологичных областей в хромосоме 1. Хотя это скопление построено на основании двух гомологичных областей хромосомы 1, это сделано лишь для примера, так как данный процесс создания скопления можно использовать для любых и всех гомологичных областей вне зависимости от хромосомы.

[00444] Как показано на ФИГ. 20, список вариантов-кандидатов можно создать следующим образом. Сначала можно сформировать совместное скопление и построить граф де Брейна (DBG) или другой граф сборки в соответствии со способами, описанными в настоящем документе. Затем с помощью DBG можно выделить варианты-кандидаты из совместных скоплений. DBG строят таким образом, чтобы формировать пузыри, которые указывают вариации, представляющие альтернативные пути через граф, причем каждый альтернативный путь является гаплотипом-кандидатом. См., например, ФИГ. 20 и 21.

[00445] Соответственно, различные пузыри на графе представляют список позиций гаплотипов варианта-кандидата. Поэтому при наличии всех ридов можно построить DBG, используя каждую референсную область в качестве остова. Затем все позиции

варианта-кандидата можно выровнять на универсальные координаты. В частности, на ФИГ. 20 приведена блок-схема, показывающая процесс формирования DBG и использования его для создания гаплотипов-кандидатов. Точнее говоря, граф де Брейна можно использовать для создания списка вариантов-кандидатов ОНП и инделов. С учетом наличия N областей, которые совместно обрабатываются с помощью MRJD, можно построить N графов де Брейна. В таком случае в каждом графе может использовать одну референсную область в качестве остова и все риды, соответствующие N областям.

[00446] Например, в одном методе реализации после построения графа DBG из него можно выделить гаплотипы-кандидаты на основе событий-кандидатов. Однако при использовании протокола предварительной обработки MRJD, как описано в настоящем документе, можно совместно обрабатывать N областей, например, когда длина областей может составлять несколько тысяч или более оснований и количество гаплотипов, которые нужно выделить, может расти очень быстро по экспоненциальному закону. Соответственно, чтобы уменьшить вычислительную сложность, вместо выделения всех гаплотипов из графов нужно выделить только пузыри, которые представляют варианты-кандидаты.

[00447] Пример структур пузырей, образуемых на графе де Брейна, показан на ФИГ. 21. Выделен ряд областей, подлежащих совместной обработке. Это определяет один из двух путей обработки, которым можно следовать. Если выявлены все совместные области, для формирования DBG можно использовать все риды. Можно выделить пузыри, показывающие возможные варианты, чтобы идентифицировать различные гаплотипы-кандидаты. А именно, для каждого пузыря можно выполнить выравнивание Смита-Ватермана на альтернативные пути к референсному остову. Отсюда можно выделить варианты-кандидаты и сохранить события из каждого графа.

[00448] Однако, в других случаях после того, как выполнен первый процесс для формирования одного или более DBG и/или i теперь равно 0, можно сформировать объединение всех событий-кандидатов из всех DBG и удалить оттуда любые дубликаты. В таком случае можно все варианты-кандидаты можно картировать, например, на систему универсальных координат, чтобы создать список вариантов-кандидатов, и этот список вариантов-кандидатов можно отправить в качестве входных данных в модуль обрезания, такой как модуль MRJD. Пример выполнения выделения пузыря вместо выделения всех гаплотипов показан на ФИГ. 22. В данном случае выделяют и обрабатывают именно только область пузыря, показывающую возможные варианты, как описано в настоящем документе.

[00449] А именно, после того, как репрезентативные пузыри выделены, можно выполнить глобальное выравнивание, например, выравнивание Смита-Ватермана, путей пузыря и соответствующего референсного остова, чтобы получить варианты-кандидаты и их позиции в референсе. Это можно сделать для всех выделенных пузырей на всех графах де Брейна. Далее, из N графов можно получить объединение всех выделенных вариантов-кандидатов, удалить дубликаты потенциально возможных вариантов, при наличии таковых, а позиции уникальных вариантов-кандидатов можно картировать на систему универсальных координат, полученную из совместного скопления. Это дает окончательный список позиций вариантов-кандидатов для N областей, которые могут действовать в качестве входных данных «обрезанного» алгоритма MRJD.

[00450] В частности, с помощью блоков обработки, которые описаны выше в настоящем документе, можно построить матрицу связности. Например, матрицу

связности можно использовать для определения порядка обработки активных, например, кандидатов, позиций, например, как функцию от длины рида и размера инсерции. Например, для дальнейшего снижения вычислительной сложности можно вычислить матрицу связности, чтобы определить порядок обработки выявленных вариантов-кандидатов, которые получены из графа де Брейна. Эту матрицу можно построить и использовать в качестве функции сортировки или вместе с ней для определения того, какие варианты-кандидаты нужно обрабатывать в первую очередь. Поэтому данная матрица связности может быть функцией от средней длины ридов и размера инсерции ридов со спаренными концами. Соответственно, для данного варианта-кандидата другие позиции потенциально возможных вариантов, которые являются целочисленными кратными размера инсерции или находятся в пределах длины рида, имеют более высокие веса по сравнению с вариантами-кандидатами в других позициях. Причина в том, что эти варианты-кандидаты с большей вероятностью обеспечивают подтверждающие данные для текущего оцениваемого варианта. На ФИГ. 23 показан пример функции сортировки, которая реализована в настоящем документе, для средней длины ридов 101 и размера инсерции 300.

[00451] Что касается функции обрезания MRJD, примеры этапов обрезанного алгоритма MRJD, упоминаемого в настоящем документе, показан на ФИГ. 24. Например, входными данными платформы и алгоритма MRJD является совместное скопление из N областей, например все варианты-кандидаты (ОНП/инделы), априорные вероятности, основанные на модели мутации, и матрица связности. Соответственно, входными данными платформы обработки обрезанного MRJD могут быть совместное скопление, выявленные активные позиции, сформированная матрица связности и модель апостериорной вероятности и/или ее результаты.

[00452] Далее, можно обработать каждый вариант-кандидат в списке, а остальные варианты можно последовательно добавлять в качестве подтверждающих данных для текущего обрабатываемого варианта-кандидата с помощью матрицы связности. Соответственно, при наличии текущего варианта-кандидата и каких-либо поддерживающих вариантов-кандидатов можно сформировать совместные диплотипы-кандидаты. Например, совместный диплотип представляет собой набор из 2N гаплотипов, где N - количество совместно обрабатываемых областей. Количество совместных диплотипов-кандидатов M является функцией от количества совместно обрабатываемых областей, количества рассматриваемых активных/кандидатов вариантов и количества фаз. Пример формирования совместных диплотипов показан ниже.

Для: P = 1, количество позиций рассматриваемых активных/кандидатов вариантов;

N = 2, количество совместно обрабатываемых областей;

$M = 2^{2 \cdot N \cdot P} = 2^4 = 16$ совместных диплотипов-кандидатов

[00453] Следовательно, пусть имеется одна активная позиция-кандидат и даны все риды и обе референсные области, и пусть этими двумя гаплотипами будут «A» и «G».

Уникальные гаплотипы = «A» и «G».

Гаплотипы-кандидаты = «AA», «AG», «GA» и «GG»', (4 потенциально возможных гаплотипа на 1 область).

Совместные диплотипы-кандидаты =

```
'AAAA', 'AAAG', 'AAGA','AAGG'
'AGAA', 'AGAG', 'AGGA', 'AGGG'
'GAAA', 'GAAG', 'GAGA','GAGG'
  'GGAA', 'GGAG', 'GGGA','GGGG'
```

[00454] Соответственно, используя совместные диплотипы-кандидаты, можно вычислить правдоподобия ридов с учетом гаплотипа для каждого гаплотипа в каждом наборе совместных диплотипов-кандидатов. Это можно сделать с помощью алгоритма HMM, как описано в настоящем документе. Однако, при этом алгоритм HMM может быть модифицирован по сравнению с его стандартным применением, чтобы учитывать варианты-кандидаты (ОНП/инделы) в гаплотипе, которые еще не обработаны. Соответственно, правдоподобия ридов можно вычислить при наличии совместного диплотипа ($P(r_i|G_m)$) с помощью результатов из модифицированной HMM. Это можно сделать с помощью следующей формулы.

[00455] В случае совместного определения в 2 областях:

$G_m$ =

$[\vartheta_{11,m}, \vartheta_{12,m}, \vartheta_{21,m}, \vartheta_{22,m}]$, причем в $\vartheta_{ij,m}$, i - область, а j - фаза, $P(r_i|G_m)$ =

$$\frac{P(ri|\vartheta 11,m)+P(ri|\vartheta 12,m)+P(ri|\vartheta 21,m)+P(ri|\vartheta 22,m)}{4}$$

$P(R|G_m) = \prod_i P(ri|Gm)$. При наличии $P(r_i|G_m)$ можно легко вычислить $P(R|G_m)$ для всех ридов. Далее, с помощью формулы Байеса можно вычислить априорную вероятность ($P(G_i|R)$) из $P(R|G_i)$ и априорные вероятности ($P(G_i)$).

$P(G_i|R) = P(R|G_i)\ P(G_i)/\sum_k P(R|Gk)\ P(Gk)$.

[00456] Кроме того, можно вычислить промежуточную матрицу генотипа для каждой области с учетом апостериорных вероятностей для всех совместных диплотипов-кандидатов. Для каждой комбинации событий в матрице генотипа можно суммировать все апостериорные вероятности всех совместных диплотипов, поддерживающих данное события. На этом этапе матрицу генотипа можно рассматривать как «промежуточную», так как включены не все варианты-кандидаты, поддерживающие текущий кандидат. Однако, как было замечено ранее, количество совместных диплотипов-кандидатов растет экспоненциально в зависимости от количества позиций вариантов-кандидатов и количества областей. Это, в свою очередь, экспоненциально увеличивает вычисления, требуемые для расчета апостериорных вероятностей. Следовательно, чтобы снизить вычислительную сложность на этой стадии, ряд совместных диплотипов можно обрезать на основе апостериорных вероятностей, для поддержания количества совместных диплотипов, которое может быть определено пользователем и запрограммировано. Наконец, можно обновить окончательную матрицу генотипов на основе определяемой пользователем метрики вариантов, которые вычисляют с помощью промежуточной матрицы генотипа. Различные этапы этих процессов показаны на блок-схеме процесса, изображенной на ФИГ. 24.

[00457] Вышеуказанный процесс можно повторять до тех пор, пока все варианты-кандидаты не будут включены в качестве подтверждающих данных для текущих вариантов-кандидатов, обрабатываемых с помощью матрицы связности. После того, как все варианты-кандидаты включены, выполняется обработка текущего варианта-кандидата. Для обработки вариантов-кандидатов возможны также другие критерии остановки. Например, процесс может быть остановлен, когда достоверность перестала расти при добавления очередных вариантов-кандидатов. Этот анализ, как показано в качестве примера на ФИГ. 24, может быть перезапущен и повторен аналогичным

образом для всех других вариантов-кандидатов в списке, что приведет к окончательному файлу определения вариантов на выхода MRJD. Соответственно, вместо рассмотрения каждой области по отдельности можно использовать протокол совместного обнаружения во множестве областей, который описан в настоящем документе, чтобы рассматривать все местоположения, из которых, возможно, произошла группа ридов, поскольку он пытается обнаружить лежащие в основе последовательности совместно, используя всю имеющуюся информацию.

[00458] Соответственно, в случае совместного обнаружения во множестве областей в примере протокола MRJD могут быть использованы одно или более из следующих уравнений в соответствии со способами, описанными в настоящем документе. А именно, вместо рассмотрения каждой области по отдельности, MRJD рассматривает множество местоположений, из которых, возможно, произошла группа ридов, и пытается совместно обнаружить лежащие в их основе последовательности, например, используя столько доступной информации, например, всю, сколько будет полезно. Например, в одном примере варианта осуществления:

[00459] Пусть N будет количеством областей, подлежащих совместной обработке. И пусть $H_k$ будет гаплотипом-кандидатом, k = 1…K, каждый из которых содержит различные ОНП, инсерции и/или делеции по сравнению с референсной последовательностью. Каждый гаплотип $H_k$ представляет одну область вдоль одной нити (или «фазу», например, материнскую или отцовскую), и они необязательно непрерывные (например, могут содержат гэпы или «безразличные» последовательности).

[00460] Пусть $G_m$ будет решением-кандидатом для обеих фаз Ф = 1,2 (для диплоидного организма) и всех областей n = 1…N:

$$G_m = \begin{bmatrix} G_{m,1,1} \ldots & G_{m,1,N} \\ G_{m,2,1} \ldots & G_{m,2,N} \end{bmatrix}$$

где каждый элемент $G_{m,\Phi,n}$ является гаплотипом, выбранным из набора гаплотипов-кандидатов $\{H_1 \ldots H_k\}$.

[00461] Во-первых, можно вычислить вероятность каждого рида для каждого гаплотипа $P(r_i|H_k)$, например, с помощью скрытой марковской модели (HMM). В случае наборов данных с парными ридами $r_i$ указывает пару $\{r_{i,1}, r_{i,2}\}$, а $P(r_i|H_k) = P(r_{i,1}|H_k)$ $P(r_{i,2}|H_k)$. В случае наборов данных со связанными ридами (например, риды в штрихкодах), $r_i$ указывает группу ридов $\{r_{i,1} \ldots r_{i,NL}\}$, которые образуются из одной и той же молекулы, а $P(r_i|H_k) = \prod_{n=1}^{NL} P(ri, n|Hk)$.

[00462] Далее, для каждого решения-кандидата $G_m$, m=1…M вычисляют условную вероятность каждого рида $P(r_i|G_m) = \frac{1}{2N}\sum_{n=1}^{N} \sum_{\Phi=1}^{2} P(ri|Gm, \Phi, n)$ и условную вероятность каждого полного скопления $R = \{r_1 \ldots r_{NR}\}$: $P(R|G_m) = \prod_{i=1}^{NR} P(ri|Gm)$.

[00463] Далее, вычисляют апостериорную вероятность каждого решения-кандидата при условии наблюдаемого скопления: $P(G_m|R) = P(R|Gm)P(Gm)/\sum_{i=1}^{M} P(R|Gi)P(Gi)$, геде

$P(G_m)$ указывает апостериорную вероятность решения-кандидата, которое подробно описано ниже в настоящем документе.

[00464] Наконец, вычисляют относительную вероятность каждого варианта-кандидата $V_j \frac{P(Vj|R)}{P(ref \mid R)} = \sum_{n|Gm=>vj} P(Gm|R) / \sum_{m|Gm=>ref} P(Gm|R)$, например, где $G_m \rightarrow V_j$ указывает, что $G_m$ поддерживает вариант $V_j$, а $G_m \rightarrow$ ref указывает, что $G_m$ поддерживает референс. В файле VCF это может быть указано как оценка качества по шкале phred: $QUAL(V_j) = -10\log_{10}\frac{P(Vj|R)}{P(ref \mid R)}$.

[00465] Пример процесса выполнения различных операций определения вариантов описан в настоящем документе со ссылкой на ФИГ. 25, где сравниваются традиционный процесс обнаружения и MRJD. А именно, на ФИГ. 25 показано совместное скопление парных ридов для двух областей, референсные последовательности которых отличаются только 3 основаниями во всем диапазоне, представляющем интерес. Известно, что все риды происходят либо из области №1, либо из области №2, но не известно с определенностью, из какой области происходит любой отдельный рид. Отметим, как описано выше, основания показаны только для позиций, где два референса отличаются, например, области пузырей, или где риды отличаются от референса. Эти области называют активными позициями. Все другие позиции можно игнорировать, так как они не влияют на вычисление.

[00466] Соответственно, как показано на ФИГ. 25, в традиционном детекторе пары ридов 1-16 будут картированы на область №2, и только они будут использованы для определения вариантов в области №2. Все эти риды совпадают с референсом в области №2, поэтому варианты не будут найдены. Аналогичным образом пары ридов 17-23 будут картированы на область №1, и только они будут использованы для определения вариантов в области №1. Как можно заметить, все эти риды совпадают с референсом в области №1, поэтому варианты не будут найдены. Однако, пары ридов 24-32 в равной мере пригодны области №1 и области №2 (каждая отличается одним основанием от референса №1 и референса №2), поэтому картирование неопределенное, и типичный определитель вариантов просто проигнорирует эти риды. Поэтому традиционный определитель вариантов не выполнит определение вариантов ни для одной из двух областей, как показано на ФИГ. 25.

[00467] Однако, в случае MRJD как показано на ФИГ. 25, результат полностью отличается от результата, полученного путем применения традиционных способов. Соответствующие вычисления приведены ниже. В данном случае N = 2 областям. Кроме того, имеются три позиции, каждая с 2 основаниями-кандидатами (основания, число которых достаточно низкое, можно безопасно игнорировать, и в данном примере число равно нулю для всех, кроме 2 оснований, в каждой позиции). Если рассматривать все комбинации, это даст $K = 2^3 = 8$ гаплотипов-кандидатов: $H_1$ = CAT, $H_2$ = CAA, $H_3$ = CCT, $H_4$ = CCA, $H_5$ = GAT, $H_6$ = GAA, $H_7$ = GCT, $H_8$ = GCA.

[00468] При вычислении перебором, когда учитывают все комбинации из всех гаплотипов-кандидатов, количество потенциально возможных решений будет $M = K^{2N} = 8^{2\cdot2} = 4096$, и для каждого решения-кандидата $G_m$ можно вычислить $P(G_m/R)$. Это вычисление для двух решений-кандидатов проиллюстрировано ниже.

$$G_{m1} = \begin{bmatrix} CAT & GCA \\ CAT & GCA \end{bmatrix}, G_{m2} = \begin{bmatrix} CAT & GCA \\ CCT & GCA \end{bmatrix}$$

Где $G_{m1}$ не имеет вариантов (это решение, найденное традиционным детектором), а $G_{m2}$ имеет единственный гетерозиготный ОНП A→C в позиции №2 области №1.

[00469] Вероятность $P(r_i|H_k)$ зависит от различных факторов, в том числе от качества основания и других параметров HMM. Можно предположить, что присутствуют только ошибки определения основания и все ошибки определения основания одинаково вероятны, поэтому $P(r_i|H_k) = (1 - p_e)^{Np(i)-Ne(i)}(p_e/3)^{Ne(i)}$, где $p_e$ - вероятность ошибки определения основания, $N_p(i)$ - количество активных позиций основания (-ий), перекрываемых ридом i, а $N_e(i)$ - количество ошибок для рида i в предположении гаплотипа $H_k$. Соответственно, можно предположить, что $p_e = 0{,}01$, что соответствует качеству основания 20 по шкале phred. В таблице, приведенной на ФИГ. 26, показаны $P(r_i|H_k)$ для всех пар ридов и всех гаплотипов-кандидатов. В двух крайних справа столбцах показаны $P(r_i|G_{m1})$ и $P(r_i|G_{m2})$ с указанием произведения внизу. На ФИГ. 26 показано, что $P(R|G_{m1}) = 3{,}5^{-30}$ и $P(R|G_{m2}) = 2{,}2^{-15}$, то есть разница в 15 порядков величины в пользу $G_{m2}$.

[00470] Апостериорные вероятности $P(G_m|R)$ зависят от априорных вероятностей $P(G_m)$. В завершение этого примера можно предположить простую модель независимо и одинаково распределенных (IID) случайных величин таким образом, что априорная вероятность решения-кандидата с Nv вариантами составляет $(1 - p_v)^{N \cdot Np - Nv}(p_v/9)^{Nv}$, где $N_p$ - количество активных позиций (3 в данном случае) , а Pv - вероятность варианта, в данном примере предполагаемая равной 0,01. Это дает $P(G_m) = 7{,}22e\text{-}13$, и $P(G_{m2}) = 0{,}500$. Следует отметить, что $G_{m2}$ является гетерозиготным по все области № 1, и все гетерозиготные пары гаплотипов имеют зеркально отображенное представление с одинаковой вероятностью (полученное простым обменом местами фаз). В данном случае сумма вероятностей для $G_{m2}$ и его зеркального отражения составляет 1,000. Вычислив вероятности отдельных вариантов, можно увидеть гетерозиготный ОНП A→C в позиции 2 области № 1 с оценкой качества 50,4 по шкале phred.

[00471] Соответственно, как можно заметить, что вычислительная сложность операции определения вариантов перебором колоссальная, причем эту сложность можно снизить путем выполнения совместного обнаружения во множестве областей, как описано в настоящем документе. Например, сложность вычисления вышеприведенных расчетов быстро растет с количеством областей N и количеством K гаплотипов-кандидатов. Чтобы рассмотреть все комбинации гаплотипов-кандидатов, количество решений-кандидатов, для которых нужно вычислить вероятности, составляет $M = K^{2N}$. В реализации перебора количество гаплотипов-кандидатов составляет $K = 2^{Np}$, где $N_p$ - количество активных позиций (например, как объяснено выше, если для формирования списка гаплотипов-кандидатов используют методы сборки графа, то Np - число независимых пузырей на графе). Следовательно, вычисление простым перебором может оказаться непозволительно дорогим для реализации. Например, если $N = 3$ и $N_p = 10$, количество решений-кандидатов составляет $M = 2^{3 \cdot 2 \cdot 10} = 2^{60} = 10^{18}$.

Поэтому на практике не приняты значения $N_p$ намного выше этого.

[00472] Следовательно, поскольку байесовское вычисление перебором может быть непозволительно сложным, далее описаны дальнейшие способы снижения сложности таких вычислений. Например, на первом этапе другого варианта реализации, начинающегося с небольшого количества позиций $N_p^j$ (или даже одной позиции $N_p^j = 1$), на этих позициях может быть выполнено байесовское вычисление. В конце вычисления варианты-кандидаты, чья вероятность попадает ниже заданного порога, могут быть удалены, например, с помощью функции обрезания дерева, как описано выше. В таком случае порог может быть адаптивным.

[00473] Далее, на втором этапе количество позиция $N_p^j$ может быть увеличено на небольшое число $\Delta N_p$ (например, так: $N_p^{j+1} = N_p^j + \Delta N_p$), и выжившие кандидаты могут быть объединены с одним или более, например, всеми, возможными кандидатами новых позициях, например, в функции выращивания дерева. Затем этапы (1) выполнения байесовского вычисления, (2) обрезания дерева и (3) выращивания дерева можно повторять, например, последовательно, до тех пор, пока не будут удовлетворены критерии остановки. Затем история порога может быть использована для определения достоверности результата (например, вероятность того, что истинное решение было найдено или не найдено). Этот процесс проиллюстрирован блок-схемой на ФИГ. 27.

[00474] Необходимо понимать, что у этого подхода существует множество возможных вариантов. Например, как было указано, порог обрезания может быть адаптивным, например, основанным на количестве выживших кандидатов. Например, в простой реализации порог может устанавливаться для поддержания количества кандидатов ниже фиксированного числа, тогда как в более сложной реализации порог может устанавливаться на основе анализа затрат и выгод включения дополнительных кандидатов. Кроме того, критерии остановки могут состоять в том, что результат найден с достаточным уровнем достоверности, или что достоверность в начальной позиции перестала расти при добавлении новых позиций. Более того, в более сложной реализации может выполняться анализ затрат и выгод продолжения добавления еще позиций. Кроме того, как показано на ФИГ. 27, порядок добавления новых позиций может зависеть от нескольких критериев, таких как расстояние до начальных позиций, или насколько высоко соединены эти позиции с уже включенными позициями (например, величина перекрытия с парными ридами).

[00475] Полезным признаком данного алгоритма является то, что вероятность того, что истинное решение не было найдено, может быть определена количественно. Например, полезную оценку получают путем простого суммирования вероятностей всех обрезанных ветвей на каждом этапе: $P_{pruned} = P_{pruned} + \sum_{m \in pruned\ set} P(G_m^j|R)$. Такая оценка полезна для вычисления достоверности получающихся в результате определений вариантов: $\frac{P(vj \mid R)}{P(ref \mid R)} = \sum_{m|Gm=>vj} P(Gm|R) + Ppruned / \sum_{m|Gm=>ref} P(Gm|R) + Ppruned$.

Хорошие оценки достоверности существенны для создания хороших кривых рабочей

характеристики приемника (РХП). Это главное преимущество данного способа обрезания над другими методами сокращениями сложности, подходящими к данному случаю.

[00476] Вернемся к примеру скопления на ФИГ. 25 и, начиная с крайней левой позиции (позиция №1), будем продвигаться вправо по одной позиции основания за раз, используя в качестве порога обрезания оценку 60 по шкале phred на каждой итерации. Пусть $\{G_m^j, m=1\ldots M_j\}$ представляет решения-кандидаты на j-й итерации. На ФИГ. 28 показаны решения-кандидаты на первой итерации, представляющие все комбинации оснований C и G, перечисленные в порядке убывания вероятности. Для любого решения с эквивалентными зеркально-отраженными представлениями (полученными путем обмена местами фаз) здесь показано только одно представление. Для всех решений-кандидатов можно вычислить вероятности, и те вероятности, которые ниже порога обрезания (указанного сплошной линией на ФИГ. 28), можно отбросить. Как показано на ФИГ. 28, в результате способов обрезания, описанных в настоящем документе, выживут шесть кандидатов.

[00477] Далее, как показано на ФИГ. 29, дерево можно выращивать путем нахождения всех комбинаций выживших на итерации №1 кандидатов и оснований-кандидатов (C и A) в позиции №2. Частичный список новых кандидатов показан на ФИГ. 29 опять в порядке убывания вероятности. Снова можно вычислить вероятности и сравнить с порогом обрезания, и в данном случае выживут 5 кандидатов.

[00478] Наконец, можно определить все комбинации выживших на итерации №2 кандидатов и оснований кандидатов в позиции №3(A и T). Окончательные кандидаты и их связанные вероятности показаны на ФИГ. 30. Соответственно, при вычислении вероятностей отдельных вариантов получится гетерозиготный ОНП A→C в позиции №2 области №1с оценкой качества 50,4 по шкале phred, что совпадает с результатом, полученным с помощью вычисления перебором. В данном примере обрезание не оказало значительного влияния на конечный результат, но в целом обрезание может влиять на вычисление, часто приводя к более достоверной оценке.

[00479] Существуют множество возможных вариантов реализации этого подхода, которые могут влиять на рабочие характеристики и сложность системы, и разные варианты могут подходить для разных сценариев. Например, возможны отличия в принятии решения о том, какие области нужно включать. Например, определитель вариантов может быть выполнен таким образом, чтобы перед выполнением совместного обнаружения во множестве областей определять, нужно ли обрабатывать данную активную область отдельно или совместно с другими областями, и если совместно, то может затем определять, какие области нужно включать. В других случаях некоторые реализации могут опираться на список вторичных выравниваний, предоставляемый сопоставителем для информации или принятия решения иным образом. В других реализациях может использоваться база данных гомологичных областей, вычисленных в автономном режиме, например, на основе поиска референсного генома.

[00480] Соответственно, полезным этапом в таких операциях является принятие решения о том, какие позиции нужно включать. Например, необходимо отметить, что различные области, представляющие интерес, могут быть не самодостаточными и/или изолированными от примыкающих областей. Следовательно, информация в скоплении может влиять на вероятность разделенных оснований значительно сильнее, чем общая длина рида (например, длина парного рида или длина длинной молекулы). Поэтому

необходимо принимать решение, какие позиции включать в вычисление MRJD, и количество позиций не ограничено (даже при обрезании). Например, в некоторых реализациях могут обрабатываться перекрывающиеся блоки позиций и обновляться результаты для подмножества позиций на основе уровня достоверности в этих позициях или полноты подтверждающих данных в этих позициях (например, позиции возле середины блока, как правило, имеют более полные подтверждающие данные, чем те, что расположены возле края).

[00481] Другим определяющим фактором может быть порядок, в котором могут добавляться новые позиции. Например, в случае обрезанного MRJD порядок добавления новых позиций может влиять на рабочие характеристики. Например, в некоторых реализациях новые позиции могут добавляться на основе расстояния до уже включенных позиций или степени связности с этими позициями (например, количества ридов, перекрывающих обе позиции). Кроме того, существуют также множество вариантов того, как может выполняться обрезание. В примере, приведенном выше, обрезание основывалось на фиксированном пороге вероятности, но обычно порог обрезания может быть адаптивным или основанным на количестве выживших кандидатов. Например, в простой реализации порог может устанавливаться для поддержания количества кандидатов ниже фиксированного числа, тогда как в более сложной реализации порог может устанавливаться на основе анализа затрат и выгод включения дополнительных кандидатов.

[00482] В различных реализациях обрезание может выполняться на основе вероятностей $P(R|G_m)$ вместо априорных вероятностей $P(G_m|R)$. Преимуществом здесь является возможность устранения эквивалентных зеркально отраженных представлений по всем областям (в дополнение к фазам). Данное преимущество, по меньшей мере частично, компенсируется недостатком, заключающимся в том, то кандидаты с очень низкими априорными вариантами не обрезаются, что в различных случаях может быть полезным. Поэтому полезное решение может зависеть от сценария. Если обрезание выполняется, например, на основе $P(R|G_m)$, то после заключительной итерации будет выполняться байесовское вычисление.

[00483] Кроме того, в приведенном выше примере процесс останавливался после обработки все позиций оснований в показанном скоплении, но возможны и другие критерии остановки. Например, если поиск решения осуществляется только для подмножества позиций оснований (например, при обработке перекрывающихся блоков), процесс может быть остановлен, когда результат для этого подмножества найден с достаточным уровнем достоверности, или когда достоверность перестала расти по мере добавления еще позиций. Однако в более сложны реализациях может выполняться какого-либо рода анализ затрат и выгод с присвоением весом стоимости вычисления в зависимости потенциальной ценности добавления дополнительных позиций.

[00484] Возможно, априорные вероятности тоже будут полезны. Например, в приведенных выше примерах использовалась простая модель IID, но могут быть использованы и другие модели. Например, необходимо отметить, что кластеры вариантов более распространены, чем можно было предсказать с помощью модели IID. Также необходимо отметить, что варианты с большей степенью вероятности встречаются в позициях, где референсы отличаются друг от друга. Следовательно, учет таких знаний в априорных вероятностях $P(G_m)$ может улучшить характеристики обнаружения и дать более хорошие кривые РХП. В частности, необходимо отметить, что среди специалистов в области геномики нет четкого понимания в отношении

априорных вероятностей для гомологических областей. Поэтому в некоторых реализациях возможно обновление априорных моделей по мере появления более качественной информации. Это может происходить автоматически по мере получения дополнительных результатов. Такие обновления могут основываться на других биологических образцах или других областях генома того же образца, изучение которых может быть применено к способам, изложенным в настоящем документе, для дальнейшего продвижения более быстрого и точного анализа.

[00485] Соответственно, в некоторых случаях может быть реализован итеративный процесс MRJD. А именно, методика, описанная в настоящем документе, может быть расширена, чтобы обеспечить возможность передачи сообщений между связанными областями с целью дальнейшего снижения сложности и/или повышения характеристик обнаружения системы. Например, результаты вычисления в одном месте могут быть использованы в качестве входной априорной вероятности для вычисления в соседнем месте. Кроме того, в некоторых реализациях может использоваться сочетание обрезания и выполнения итераций для достижения требуемого компромисса между рабочими характеристиками и сложностью.

[00486] Кроме того, возможна реализация приготовления образца для оптимизации процесса MRJD. Например, в случае секвенирования спаренных концов, возможно, будет полезно иметь жесткое распределение по размеру инсерции при использовании традиционного обнаружения. Однако в различных случаях введение вариации в размер инсерции могло бы значительно улучшить рабочие характеристики MRJD. Например, образец может быть приготовлен для умышленного введения бимодального распределения, многомодального распределения или распределения с колоколообразной кривой с более высокой вариацией, чем, как правило, реализуют для традиционного обнаружения.

[00487] На ФИГ. 31 показаны кривые РХП для MRJD и традиционного детектора для человеческого образца NA12878 в выбранных областях генома с одной гомологичной копией, так что N = 2, с меняющимися степенями подобия референсной последовательности. Для этого набора данных использовано секвенирование спаренных концов с длиной рида 101 и средним размер инсерции приблизительно 400. Как показано на ФИГ. 31, MRJD обеспечивает резко улучшенную чувствительность и специфичность в этих областях по сравнению с традиционными способами обнаружения. На ФИГ. 32 показаны те же самые результаты как функция от подобия последовательности, измеряемой в окне из 1000 оснований (например, если референсы отличаются 10 основаниями из 1000, то подобие составляет 99,0 процента). Можно заметить, что в случае этого набора данных традиционное обнаружение начинает плохо работать при подобии последовательности ~0,98, тогда как MRJD работает достаточно хорошо вплоть до 0,995 и даже выше.

[00488] Кроме того, в различных случаях данная методика может быть расширена, чтобы обеспечить возможность передачи сообщений между связанными областями с целью дальнейшего снижения сложности и/или повышения характеристик обнаружения. Например, результаты вычисления в одном месте могут быть использованы в качестве входной априорной вероятности для вычисления в соседнем месте, и в некоторых реализациях может использоваться сочетание обрезания и выполнения итераций для достижения требуемого компромисса между рабочими характеристиками и сложностью. В конкретных случаях, как указано выше, перед выполнением совместного обнаружения во множестве областей определитель вариантов может определять, следует ли обрабатывать данную активную область отдельно или совместно с другими областями.

Кроме того, как указано выше, некоторые реализации могут опираться на список вторичных выравниваний, предоставляемый сопоставителем, для принятия такого решения. В других реализациях может использоваться база данных гомологичных областей, вычисленных в автономном режиме на основе поиска референсного генома.

[00489] Ввиду вышеизложенного, можно реализовать частично определенную скрытую марковскую модель (PD-HMM) таким образом, чтобы использовать преимущества MRJD. Например, MRJD может отдельно оценивать вероятность наблюдения части или всех ридов при данном возможном совместном диплотипе, который содержит по одному гаплотипу на каждую плоидию в каждой гомологичной референсной области, например, в случае двух гомологичных областей в диплоидных хромосомах каждый совместный диплотип будет содержать четыре гаплотипа. В таких случаях можно рассматривать все или часть возможных гаплотипов, например, путем конструирования, например, за счет изменения каждой референсной области с помощью каждого возможного подмножества из всех вариантов, для которых имеются нетривиальные подтверждающие данные. Однако, в случае длинных гомологичных референсных областей количество возможных вариантов большое, поэтому количество гаплотипов (комбинаций вариантов) вырастает экспоненциально, и количество совместных диплотипов (комбинаций гаплотипов) может стать астрономическим.

[00490] Следовательно, для сохранения управляемости вычислениями MRJD, возможно, нецелесообразно проверять все возможные совместные диплотипы. Скорее, в некоторых случаях систему можно выполнить с возможностью проверки только небольших подмножеств «наиболее вероятных» совместных диплотипов. Эти «наиболее вероятные» совместные диплотипы могут быть определены путем инкрементального построения дерева частично определенных совместных диплотипов. В таких случаях каждый узел дерева может быть частично определенным диплотипом, который содержит частичной определенный гаплотип на каждую плоидию каждой гомологичной референсной области. В этом случае частично определенный гаплотип может содержать референсную область, модифицированную частично определенным подмножеством возможных вариантов. Соответственно, частично определенное подмножество возможных вариантов может для каждого возможного варианта содержать индикацию одного из трех состояний: вариант определен и присутствует, или вариант определен и отсутствует, или вариант еще не определен, например, он может присутствовать или отсутствовать. В корне дерева все варианты не определены во всех гаплотипах; узлы дерева, разветвляющиеся последовательно при отдалении от корня имеют последовательно больше вариантов, определенных как присутствующие или отсутствующие в каждом гаплотипе каждого совместного диплотипа узла.

[00491] Кроме того, в контексте данного дерева совместных диплотипов, как описано выше, объем вычислений MRJD сохраняется ограниченным и управляемым за счет обрезания ветвей дерева, в которых все узлы совместных диплотипов маловероятны, например, с вероятностью от умеренной до крайней, по сравнению с другими более вероятными ветвями или узлами. Соответственно, такое обрезание можно выполнять на ветвях или узлах, которые до сих пор лишь частично определены, например несколько или много вариантов еще не определены как присутствующие или отсутствующие в гаплотипах совместного диплотипа обрезанного узла. Следовательно, в таком случае полезно иметь возможность оценки или связывания правдоподобия каждого наблюдаемого рида в предположении истинности частично определенного гаплотипа. Вычисление модифицированной парной скрытой марковской модели (pHMM), обозначаемой «PD-HMM» в случае «частично определенной парной скрытой марковской

модели», полезно для оценки вероятности P(R|H) наблюдения рида R в предположении, что истинный гаплотип H* согласуется с частично определенным гаплотипом H. В данном контексте «согласуется» означает, что некоторый конкретный истинный гаплотип H* согласуется с частично определенным гаплотипом H с точки зрения всех вариантов, присутствие или отсутствие которых определено в H, но для вариантов, не определенных в H, H* может согласоваться с референсной последовательностью, как модифицированной, так и немодифицированной каждым неопределенным вариантом.

[00492] Отметим, что простого выполнения обычного вычисления pHMM, как правило, недостаточно для охвата некоторым выбранным подгаплотипом H только определенных позиций вариантов. Вообще важно построить дерево совместных диплотипов с неопределенными вариантами, решаемыми в эффективном порядке, который обычно отличается от их геометрического порядка, поэтому частично определенный гаплотип H будет, как правило, иметь много неопределенных позиций вариантов, перемежающихся с определенными. Для правильного рассмотрения связанных с инделами ошибок ПЦР полезно использовать напоминающее pHMM вычисление, охватывающее все определенные варианты и значительный радиус вокруг них, что может быть не совместимо с попытками избежать неопределенных позиций вариантов.

[00493] Соответственно, входные данные PD-HMM могут включать в себя подвергнутую определению вариантов нуклеотидную последовательность рида R, оценки качества основания (например, по шкале phred) определенных нуклеотидов рида R, исходный гаплотип H0 и список неопределенных вариантов (редакций) из H0. В число неопределенных вариантов могут входить замены одного основания (ОНП), замены множества оснований (МНП), инсерции и делеции. Преимуществом является то, что этого может быть достаточно для поддержки неопределенных ОНП и делеций. Неопределенные МНП могут быть неполностью, но в достаточной степени представлены как множество независимых ОНП, Неопределенная инсерция может быть представлена первой редакцией инсерции в исходном гаплотипе с последующим указанием неопределенной делеции, которая отменяет данную инсерцию.

[00494] На неопределенные делеции можно наложить ограничения, чтобы облегчить реализацию жестко смонтированного движка с ограниченной памятью и логикой, например, запретить перекрытие двух неопределенных делеций (удаление одних и тех же оснований исходного гаплотипа). При необходимости проверки частично определенного гаплотипа с помощью неопределенных вариантов, нарушающих такие ограничения, это можно решить путем преобразования одного или более неопределенных вариантов в определенные варианты за счет большего числа операций PD-HMM, охватывающих случаи с присутствием и отсутствием этих вариантов. Например, если две неопределенные делеции A и B нарушают ограничение вследствие перекрытия друг с другом в исходном гаплотипе H0, то делеция B может быть отредактирована в H0 с получением H0B, и могут быть выполнены две операции PD-HMM, использующие только неопределенную делецию A, одну для исходного гаплотипа H0, а другую для исходного гаплотипа H0B, и полученная в результате этих двух операций PD-HMM максимальная вероятность может быть сохранена.

[00495] Результатом операции PD-HMM может быть оценка максимума P(R|H*) среди всех гаплотипов H*, которые могут быть сформированы путем редактирования H0 с использованием только любого подмножества неопределенных вариантов. Максимизация может выполняться локально и вносить вклад в напоминающее pHMM динамическое программирование в данной ячейке, как если бы примыкающий

неопределенный вариант присутствовал или отсутствовал в гаплотипе в зависимости от того, какая оценка выше, например, вносить вклад в более высокую частную вероятность. Такая максимизация во время динамического программирования может привести к более высоким оценках максимума P(R|H*), чем истинная максимизация она отдельных чистых гаплотипах H*, но разница обычно несущественная.

[00496] Неопределенные ОНП могут быть включены в PD-HMM путем разрешения задания одного или более значений совпадающих нуклеотидов для каждой позиции гаплотипа. Например, если основание 30 гаплотипа H0 представляет собой «C», а неопределенный ОНП заменяет это основание «C» на «T», то гаплотип в операции PD-HMM может указывать позицию 30 как оба основания, «C» и «T». При обычном динамическом программировании pHMM любой переход в состояние «M» приводит к умножению вероятности пути на вероятность правильного определения основания (если позиция гаплотипа совпадает с позицией рида) или на вероятность определенной ошибки определения основания (если позиция гаплотипа не совпадает с позицией рида); в случае PD-HMM это изменено путем использования вероятности правильного определения, если позиция рида совпадает с любой из возможных позиций гаплотипа (например, «C» или «T»), и вероятности ошибки определения основания в противном случае.

[00497] Неопределенные делеции гаплотипа могут быть включены в PD-HMM путем пометки флагом необязательно удаляемых позиций гаплотипа и изменения динамического программирования pHMM с тем, чтобы позволить путям выравнивания проходить, минуя по горизонтали сегменты гаплотипа с неопределенной делецией без потери вероятности. Это можно сделать различными способами, но с соблюдением общего правила, заключающегося в том, что значения вероятности состояний M, I и/или D могут переходить по горизонтали (вдоль оси гаплотипа) по всему промежутку неопределенной делеции без уменьшения обычных вероятностей открытия гэпа и продления гэпа.

[00498] В одном конкретном варианте реализации позиции, где начинаются неопределенные делеции, помечают флагом «F1», а позиции, где неопределенные делеции заканчиваются, помечают флагом «F2». В дополнение к «состояниям» M, I и D (представлениям частной вероятности) для каждой ячейки матрицы HMM (гаплотип по горизонтали/рид по вертикали) каждая ячейка PD-HMM может дополнительно содержать состояния «обхода» BM, BI и BD. В помеченных флагом F1 столбцах гаплотипа состояния BM, BI и BD принимают значения, копируемые из состояний M, I и D ячейки слева, соответственно. В не помеченных флагом F2 столбцах гаплотипа, в частности, в столбцах, начинающихся со столбца, помеченного флагом F1, и далее внутрь неопределенной делеции, состояния BM, BI и BD передают свои значения состояниям BM, BI и BD ячейки справа, соответственно. В помеченных флагом F2 столбцах гаплотипа вместо состояний M, I и D, используемых для вычисления состояний примыкающих ячеек, используют максимум M и BM, максимум I и BI и максимум D и BD, соответственно. Это показано в качестве примера в столбце F2 как мультиплексированный выбор сигналов из регистров M и BM, I и BI, D и BD.

[00499] Отметим, что хотя регистры состояния BM, BI и DB могут быть представлены в столбцах с F1 по F2, и максимизирующие мультиплексоры M/BM, I/BI и D/BD могут быть показаны в столбце F2, эти компоненты могут присутствовать для вычислений всех ячеек, позволяя обрабатывать неопределенные делеции в любой позиции и обеспечивая возможность множества неопределенных делеций с соответствующими флагами F1 и F2 по всему гаплотипу. Отметим также, что флаги F1 и F2 могут быть в

одном и том же столбце, когда неопределенная делеция состоит из одного основания. Так же необходимо отметить, что матрицу PD-HMM ячеек можно изобразить в виде схематического представления логических вычислений состояний M, I, D, BM, BI и BD, но в аппаратной реализации может присутствовать меньшее количество ячеек, вычисляющих логические элементы, причем организованных в конвейер соответствующим образом для вычисления значений состояний M, D, I, BM, BI и BD с высокими тактовыми частотами, а ячейки матрицы могут вычисляться с различной степенью распараллеливания аппаратного обеспечения в различных порядках в соответствии с внутренне присущими логическими зависимостями вычисления PD-HMM.

[00500] Таким образом, в данном варианте реализации значения состояний pHMM в одном столбце могут находиться непосредственно слева от делеции, причем они могут быть захвачены и переданы вправо, без изменения, в крайний справа столбец этой промежуточной делеции, где они будут подставлены в вычисления pHMM всякий раз, когда превзойдут оценки нормального пути. В случае выбора этих максимальных значений значения состояний «обхода» BM, BI и BD представляют результаты локального динамического программирования в предположении наличия делеции, тогда как значения «нормальных» состояний M, и D представляют результаты локального динамического программирования в предположении отсутствия неопределенной делеции.

[00501] В другом варианте реализации может быть использовано одно состояние обхода, например, состояние BM, принимающее значение из состояния M столбца, помеченного флагом F1, или принимающее сумму состояний M, D и/или I. В другом варианте реализации вместо использования состояний «обхода» удаляют штрафы на открытие гэпа/продление гэпа в столбцах неопределенных делеций. В другом варианте реализации состояния обхода вносят аддитивный вклад в динамическое программирование вправо от неопределенных делеций, а не используются для локальной максимизации. В еще одном варианте реализации используют в большем или меньшем количестве, или по-другому определенные, или по-другому располагаемые флаги позиции гаплотипа для инициирования обхода или подобных действий, например, один флаг, указывающий на принадлежность к неопределенной делеции. В дополнительном варианте реализации могут участвовать две или более перекрывающиеся неопределенные делеции, например, с использованием дополнительных флагом и/или состояний обхода. Кроме того, поддерживаются неопределенные инсерции в гаплотипе вместо или в дополнение к неопределенным делециям. Аналогичным образом поддерживаются неопределенные инсерции и/или делеции на оси рида вместо или в дополнение к неопределенным делециям и/или инсерциям на оси гаплотипа. В другом варианте реализации поддерживаются неопределенные замены множества оснований в качестве неделимых вариантов (все присутствуют или все отсутствуют). В еще одно варианте реализации поддерживаются неопределенные замены переменной длины в качестве неделимых вариантов. В другом варианте реализации неопределенные варианты штрафуют с использованием фиксированной или конфигурируемой вероятности или коррекций оценки.

[00502] Данное вычисление PD-HMM может быть реализовано в виде аппаратного движка, например, в технологии FPGA или ASIC, путем расширения архитектуры аппаратного движка для «обычного» вычисления pHMM, или может быть реализовано с помощью одной или более квантовых схем на квантовой вычислительной платформе. В дополнение к конвейерной логике движка для вычисления, передачи и хранения значений состояний M, I и D для различных или последовательных ячеек может быть

создана параллельная логика для вычисления, передачи и хранения значений состояний BM, BI и BD, как описано выше в настоящем документе. Ресурсы памяти и порты для хранения и извлечения значений состояний M, I и D могут быть дополнены аналогичными или более широкими или глубокими ресурсами памяти и портами для хранения и извлечения значений состояний BM, BI и BD. Флаги, такие как F1 и F2, могут храниться в памятях наряду со связанными основаниями гаплотипов.

[00503] Множество совпадающих нуклеотидов для, например, позиций неопределенного ОНП гаплотипа, могут быть закодированы таким образом, чтобы использовать вектор, содержащий по одному биту на каждое возможное значение нуклеотида. Зависимости вычисления ячеек в матрице pHMM остаются неизменными в PD-HMM, поэтому порядок и организация конвейера вычислений множества ячеек может оставаться тем же самым для PD-HMM. Однако, задержка по времени и/или тактовым циклам для полного вычисления ячейки несколько увеличивается в случае PD-HMM, поскольку требуется сравнивать значения «обычных» и «обходных» состояний и выбирать из них те, что больше. Соответственно, возможно, что преимуществом будет включение одной или более дополнительных стадий конвейера для вычисления ячеек PD-HMM, что приведет к дополнительной задержке в тактовых цикла. Кроме того, другим преимуществом может быть расширение каждой «полосы захвата» ячеек, вычисляемых с помощью одной или более строк, чтобы дольше поддерживать конвейер в заполненном состоянии без проблем с зависимостями.

[00504] Данное вычисление PD-HMM в два раза больше значений состояний (BM, BI, и BD в дополнение к M, I и D), чем обычное вычисление pHMMB, и может потребовать примерно в два раза больше аппаратных ресурсов для реализации движка с эквивалентной пропускной способностью. Однако движок PD-HMM обладает преимуществами экспоненциальной скорости и эффективности с точки зрения увеличения неопределенных вариантов по сравнению с обычным движком pHMM, совершающего один прогон для каждого гаплотипа, представляющего отличающуюся комбинацию присутствующих или отсутствующих неопределенных вариантов. Например, если частично определенный гаплотип имеет 30 неопределенных вариантов, каждый из которых может независимо присутствовать или отсутствовать, то существуют 2^30, или более 1 миллиарда, различных специфических гаплотипов, которые пришлось бы обрабатывать с помощью pHMM в противном случае.

[00505] Соответственно, эти и другие подобные операции, описанные в настоящем документе, могут быть выполнены таким образом, чтобы лучше понимать и точнее прогнозировать, что случилось с геномом субъекта, что риды изменились относительно референса. Например, даже если мутации, возможно, встречаются случайным образом, существуют случаи, когда правдоподобие их появления представляется потенциально предсказуемой до некоторой степени. В частности, в некоторых случаях, когда встречаются мутации, они могут возникать в определенных известных местах и в определенных формах. Более конкретно, мутации, если они происходят, будут возникать на одном или другом аллеле, или на том и другом, и обычно чаще возникают в одних местах, чем в других, например, на концах хромосом. Следовательно, эта и другая связанная информация может быть использована для разработки моделей мутации, которые могут быть сформированы и использованы для лучшей оценки вероятного присутствия мутации в одной или более областях генома. Например, учитывая различные априорные знания, например, одну или более моделей мутации, при выполнении анализов геномных вариаций, можно добиться более хороших и более точных результата геномного анализа, например, с более точными разграничениями генетических мутаций.

[00506] Такие модели мутации могут позволить учитывать частоту и/или местоположение различных известных мутаций и/или мутаций, которые, как представляется, происходят в сочетании друг с другом или иным неслучайным образом. Например, установлено, что вариации встречаются преимущественно ближе к концам данной хромосомы. Таким образом, известные модели мутаций могут быть сформированы, сохранены в базе данных, описанной в настоящем документе и использоваться системой для более хорошего прогнозирования наличия одной или более вариаций в анализируемых геномных данных. Кроме того, можно также реализовать процесс машинного обучения, как описано более подробно ниже в настоящем документе, чтобы различные данные результатов, получаемые с помощью анализов, выполняемых здесь, могли быть проанализированы и использованы для более хорошего информирования системы о том, когда нужно предпринимать конкретное определение вариантов, например, в соответствии с принципами машинного обучения, описанными в настоящем документе. А именно, машинное обучение может быть реализовано на совокупных наборах данных, особенно в отношении определенных вариаций, и это обучение может быть использовано для более хорошего формирования более всеобъемлющих моделей мутации, которые, в свою очередь, могут быть использованы для выполнения более точных определений вариаций.

[00507] Следовательно, система может быть выполнена с возможностью рассмотрения различных данных вариаций, проверять данные на различные корреляции и, в случае обнаружения корреляции, такая информация может быть использована для более хорошего взвешивания и, следовательно, более точного определения наличия других вариаций в других образцах генома, например на регулярной основе. Соответственно, подобным образом система, в особенности движок определения вариантов, может постоянно обновляться изученными данными о корреляции варианта, чтобы достигать прогресса в более качественном определении вариантов для получения более хороших и более точных данных результатов.

[00508] А именно, можно использовать телеметрию для обновления растущей модели мутации, чтобы достигать улучшенного анализа в системе. Это может быть особенно полезно при анализе образцов, которые некоторым образом связаны друг с другом, например, принадлежат одной и той же географической популяции, и/или может быть использовано для определения того, какой референсный геном из множества референсных геномов может быть лучшим референсным геномом для анализа с его помощью конкретного образца. Кроме того, в различных случаях модель мутации и/или телеметрия могут быть использованы, чтобы лучше выбирать референсный геном для использования в процессах системы и тем самым улучшать точность и эффективность результатов системы. В частности, когда в одном или более описанных в настоящем документе анализов могут быть использованы множество референсных геномов, при выборе для использования преимущество может быть отдано конкретному референсному геному, например, благодаря применению модели мутации при выборе наиболее подходящего референсного генома для применения.

[00509] Необходимо отметить, что при выполнении вторичного анализа фундаментальная структура для каждой картируемоей или выравниваемой области генома может содержать один или более основополагающих генов. Соответственно, в различных случаях это понимание основополагающих генов/или функций белков, которые они кодируют, может обеспечить полезную информацию при выполнении вторичного анализа. В частности, третичные показатели и/или результаты могут быть полезны в протоколах вторичного анализа, выполняемых представленной системой,

например, в процессе биологической контекстно-зависимой модели мутации. Более конкретно, поскольку ДНК кодирует гены, а гены кодируют белки, информация о таких белках, которые приводят к мутациям и/или одиозным функциям, может быть использована для информирования моделей мутации, используемых при выполнении вторичного и/или третичного анализа на геноме субъекта.

[00510] Например, третичный анализ, например, набор образцов генов, кодирующих мутированные белки, может быть информативным при выполнении вторичного анализа геномных областей, о которых известно, что они кодируют такие мутации. Следовательно, как указано выше, различные результат третичной обработки могут быть использованы для информирования и/или обновления моделей мутации, описанных в настоящем документе, для достижения более высокой точности и эффективности при выполнении различных операций вторичного анализа, описанных в настоящем документе. А именно, информация о мутированных белках, например, контекстуальный третичный анализ, может быть использована для обновления модели мутации прим выполнении вторичного анализа тех областей, о которых известно, что они кодируют белки и/или потенциально содержат такие мутации.

[00511] Соответственно, ввиду вышеизложенного, в случае вариантов реализации, включающих в себя ускоренные с помощью FPGA приложения картирования, выравнивания, сортировки и/или определения вариантов, одна или более из этих функций могут быть реализованы в одном или обоих из программных или аппаратных (АО) компонентах обработки, например, в программном обеспечении, исполняемом на традиционных ЦПУ, ГПУ, КПУ, и/или в прошивке, которая может быть внедрена в FPGA, ASIC, sASIC и т.п. В таких случаях ЦПУ и FPGA должны быть выполнены с возможностью обмена данными, чтобы передавать результаты с одного этапа в одном устройстве, например, ЦПУ или FPGA, для обработки на следующем этапе в другом устройстве. Например, при выполнении функции картирования построение больших структур данных, таких как индекс референса, может быть реализовано с помощью ЦПУ, а выполнение хэш-функции применительно к ним может быть реализовано с помощью FPGA. В таком случае ЦПУ может строить структуры данных, сохранять их в связанной памяти, такой как DRAM, причем память затем может быть доступна для движков обработки, выполняемых на FPGA.

[00512] Например, в некоторых вариантах реализации обмены данными между ЦПУ и FPGA могут быть реализованы с помощью любого подходящего межсоединения, такого как периферийная шина, например, шина PCIe, USB или сетевой интерфейс, такой как Ethernet. Однако шина PCIe может обеспечивать сравнительно слабую интеграцию между ЦПУ и FPGA, за счет чего задержки передачи между ними могут быть довольно высокими. Соответственно, хотя одно устройство (например, ЦПУ или FPGA) может получать доступ к памяти, прикрепленной к другому устройству (например, посредством передачи DMA), вызываемые области памяти выполнены без возможности кэширования, поскольку не имеют средств для поддержания когерентности кэша между двумя устройствами. В результате передачи между ЦПУ и FPGA ограничены выполнением между большими этапами обработки высокого уровня, и большое количество заданий на ввод и вывод должны быть организованы в очередь между устройствами, чтобы не замедлять друг друга в ожидании операций с высокой задержкой. Это замедляет различные операции обработки, описанные в настоящем документе. Кроме того, когда FPGA получает доступ к выполненной без возможности кэширования памяти ЦПУ, вся нагрузка такого обращения к памяти ложится на внешние интерфейсы памяти ЦПУ, которые ограничены по полосе пропускания по сравнению

с их внутренними интерфейсами кэша.

[00513] Соответственно, вследствие таких слабых интеграций ЦПУ/FPGA обычно требуется иметь «централизованное» программное управление интерфейсом FPGA. В таких случаях различные программные потоки могут обрабатывать различные единицы данных, но когда эти потоки формируют работу для выполнения движком FPGA, эта работа должна быть агрегирована в «центральных» буферах, например, с помощью программного потока с одним агрегатором или с помощью доступа с агрегированием множества потоков посредством семафоров, причем управление передачей агрегированной работы с помощью пакетов DMA осуществляется центральным программным модулем, таким как драйвер пространства ядра. Следовательно, по мере создания результатов аппаратными движками происходит обратный процесс, причем программный драйвер принимает пакеты DMA из аппаратного обеспечения, а деагрегатор потоков распределяет результаты различным ожидающим программным рабочим потокам. Однако, это централизованное программное управление обменом данными с аппаратной логикой FPGA является громоздким и ресурсоемким, снижает эффективность программной поточно обработки и обмена данными между аппаратным и программным обеспечением, ограничивает практическую полосу пропускания обмена данными между аппаратным и программным обеспечением и резко повышает ее задержку.

[00514] Кроме того, как показано на ФИГ. 33A, слабая интеграция между ЦПУ 1000 и FPGA 7 может потребовать наличия у каждого устройства своей собственной специализированной внешней памяти, такой как DRAM 1014, 14. Как показано на ФИГ. 33A, ЦПУ 1000 имеет свою собственную DRAM 1014 на системной материнской плате, например, модули DIMM DDR3 или DDR4, тогда как FPGA 7 имеет свою собственную DRAM 14, например, 8 ГБ SODIMM, которая может быть напрямую соединена с FPGA 7 посредством одной или более шин 6 для DDR3, таких как шина PCIe с большой задержкой. Аналогичным образом ЦПУ 1000 может быть соединен с возможностью обмена данными со своим собственным DRAM 1014, например с помощью соответствующим образом сконфигурированной шины 1006. Как указано выше, FPGA 7 может быть выполнена с возможностью содержания одного или более движков 13 обработки, причем эти движки обработки могут быть выполнены с возможностью осуществления одной или более функций в биоинформационном конвейере, как описано в настоящем документе, например, когда FPGA 7 содержит движок 13a картирования, движок 13b выравнивания и движок 13c определения вариантов. Также могут быть включены другие движки, описанные в настоящем документе. В различных вариантах реализации один или оба из ЦПУ и FPGA могут быть выполнены с возможностью содержания кэш-памяти 1014a, 14a, соответственно, которая выполнена с возможностью хранения данных, например, результирующих данных, которые передаются в нее одним или более различными компонентами системы, такими как одна или более памятей и/или один или более движков обработки.

[00515] Многие операции, описанные в настоящем документе, которые подлежат выполнению с помощью FPGA 7 для геномной обработки, требуют доступа к большой памяти для выполнения основополагающих операций. А именно, ввиду использования больших единиц данных, например, референсных геномов из более 3 миллиардов нуклеотидов, свыше 100 миллиардов нуклеотидов необработанных данных секвенатора и т.д., FPGA 7 может потребоваться многократный доступ к главной памяти 1014, например, для доступа к индексу, такому как хэш-таблица объемом 30 ГБ, или другому индексу референсного генома, например, в целях картирования затравок из исследуемой

секвенированной ДНК/РНК на референсный геном из 3 миллиардов пар оснований и/или выборки сегментов-кандидатов, например, из референсного генома, для выравнивания на них.

[00516] Соответственно, в различных реализациях системы, описанной в настоящем документе, может понадобиться множество быстрых доступов к оперативной памяти одним или более жестко смонтированных движков 13, например, при выполнении операции картирования, выравнивания и/или определения вариантов. Однако, совершение FPGA 7 такого большого количества небольших произвольных доступов к памяти посредством периферийной шины 3 или другой сетевой линии связи с памятью 1014, присоединенной к главному ЦПУ 1000, может оказаться непозволительно непрактичным. Например, в таких случаях задержки возвращаемых данных могут быть очень большими, эффективность шины может быть очень низкой, например, для таких небольших произвольных доступов, и нагрузка на интерфейс 1006 внешней памяти ЦПУ может быть непозволительно большой.

[00517] Кроме того, в результате каждое устройство нуждается в своей собственной внешней памяти, причем полный форм-фактор полной платформы ЦПУ 1000 + FPGA 7 вынужден быть больше, чем было бы желательно, например, для некоторых областей применения. В таких случаях в дополнение к стандартной системной материнской плате для одного или более ЦПУ 1000 и поддерживающих микросхем 7 и памятей 1014 и/или 14, требуется пространство на плате для большого корпуса FPGA (которое может оказаться даже еще больше вследствие необходимости достаточного количества контактов для нескольких шин внешней памяти) и нескольких модулей памяти 1014, 14. Однако стандартные материнские платы не содержат этих компонентов, да и найти на них свободного пространства трудно, поэтому практический вариант реализации может быть выполнен с возможностью использования платы 2 расширения, содержащей FPGA 7, ее память 14 и другие поддерживающие компоненты, такие как источник питания, которая, например, соединена с гнездом расширения PCIe на материнской плате ЦПУ. Чтобы иметь пространство для платы расширения 2, система может быть выполнена в достаточно большом корпусе, например, в виде сервера 1U или 2U или более крупного сервера, монтируемого в стойке.

[00518] Ввиду вышеизложенного, в различных случаях, как показано на ФИГ. 33B, для преодоления этих факторов, возможно, потребуется конфигурировать ЦПУ 1000 в компоновке с жестким связыванием с FPGA 7. В частности, в различных случаях FPGA 7 может быть жестко связана с ЦПУ 1000, например, с помощью межсоединения 3 с малой задержкой, такого как межсоединение быстрого доступа (QPI). А именно, чтобы организовать более жесткую интеграцию ЦПУ + FPGA, эти два устройства могут быть соединены с помощью любого подходящего интерфейса с низкой задержкой, такого как «межпроцессорное соединение» и т.п, например INTELS® Quick Path Interconnect (QPI) или HyperTransport (HT).

[00519] Соответственно, как показано на ФИГ. 33B, система 1, предложенная в настоящем документе, содержит ЦПУ 1000 и процессор, такой как FPGA 7, причем оба устройства связаны с одним или более модулями памяти. Например, ЦПУ 1000 может быть соединено, например, с помощью соответствующим образом сконфигурированной шины 1006, с DRAM 1014, и, аналогичным образом, FPGA 7 соединена с возможностью обмена данными со связанной память 14 посредством шины 6 DDR3. Однако, в данном случае вместо того, чтобы соединяться друг с другом, например, посредством типичного межсоединения с низкой задержкой, например, интерфейса PCIe, ЦПУ 1000 соединено с FPGA 7 с помощью межсоединения 3 HyperTransport, такого как QPI. В таком случае

благодаря низкой задержке, присущей таким межсоединениям, связанные памяти 1014, 14 ЦПУ 1000 и FPGA 7 легко доступны друг для друга. Кроме того, в различных случаях ввиду этой жестко связанной конфигурации один или более кэшей 1114a/14a, связанных с устройствами, могут быть выполнены с возможностью поддержания когерентности друг с другом.

[00520] В число некоторых основных свойств такого жестко связанного межсоединения ЦПУ/FPGA входят широкая полоса пропускания, например, 12,8 ГБ/с; низкая задержка, например, 100-300 нс; адаптированный протокол, выполненный с возможностью обеспечения эффективных удаленных доступов к памяти и эффективных небольших передач в память, например, порядке 64 байтов или менее; и интеграция поддерживаемого протокола и ЦПУ для доступа к кэшу и когерентности кэша. В таких случаях естественным межсоединением для использования такой жесткой интеграции с данным ЦПУ 1000 может быть его собственное межпроцессорное соединение 1003, которое может быть использовано здесь для обеспечения возможности параллельной работы множества ядер и множества ЦПУ в пространстве совместно используемой памяти 1014, тем самым обеспечивая возможность доступа к стекам кэша друг друга и внешней памяти с поддержание когерентности кэша.

[00521] Соответственно, как показано на ФИГ. 34A и 34B, может быть предусмотрена плата 2, например, когда плата может быть выполнена с возможностью приема одного или более ЦПУ 1000, например, посредством множества межсоединений 1003, таких как собственные межпроцессорные соединения 1003a и 1003b. Однако в данном случае, как показано на ФИГ. 34A, ЦПУ 1000 выполнено с возможностью соединения с межсоединением 1003a, но вместо соединения с ним другого ЦПУ посредством межсоединения 1003b, с возможностью соединения с ним выполнена FPGA 7 по настоящему изобретению. Кроме того, система 1 выполнена таким образом, что ЦПУ 1000 может быть соединен со связанной FPGA 7, например, посредством межсоединения 3 жесткого связывания с малой задержкой. В таких случаях каждая память 1014, 14, связанная с соответствующим устройством 1000, 7, может быть выполнена с возможностью получения доступа друг к другу, например, с широкой полосой пропускания и поддержанием когерентности кэша.

[00522] Аналогичным образом, как показано на ФИГ. 34B, система также может быть выполнена с возможностью приема корпусов 1002a и/или 1002b, например, когда корпусы содержат один или более ЦПУ 1000a, 1000b, которые сильно связаны, например, посредством межсоединений 3a и 3b с низкой задержкой, с одной или более FPGA 7a, 7b, например, когда при данной архитектура системы каждый корпус 2a и 2b может быть соединен один с другим, например, посредством межсоединения 3 с жестким связыванием. Кроме того, как показано на ФИГ. 35, в различных случаях может быть предусмотрен корпус 1002a, причем корпус 1002a содержит ЦПУ 1000, выполненный с возможностью сильного связывания с интегральной схемой, такой как FPGA 7. В таком случае за счет сильного связывания ЦПУ 1000 и FPGA 7 система может быть построена с возможностью непосредственного совместного использования кэша 1014a с обеспечением согласованности, когерентности и легкодоступности для каждого из двух устройств, например, в отношении данных, хранящихся там.

[00523] Следовательно, в таких случаях FPGA 7 и один из корпусов 2a/2b может, в сущности, выдавать себя за другой ЦПУ и тем самым работать в среде поддерживающей когерентность кэша совместной используемой памяти с одним или более ЦПУ точно так же, как это делали бы множество ЦПУ на многогнездовой материнской плате 1002 или множество ядер ЦПУ в многоядерном ЦПУ. При таком межсоединении FPGA/

ЦПУ FPGA 7 может эффективно совместно использовать память 1014 ЦПУ, а не иметь свою собственную специализированную внешнюю память 14, которая может или не может быть включена или доступа. Следовательно, в такой конфигурации быстрые, краткие, произвольные доступы эффективно поддерживаются межсоединением 3, например, с малой задержкой. Это делает ее практичной и эффективной для доступа различных движков 13 обработки в FPGA 7 к большим структурам данных в памяти ЦПУ 1000.

[00524] Например, как показано на ФИГ. 37, предложена система для осуществления одного или более способов, описанных в настоящем документе, например, когда способ включает в себя один или более этапов для выполнения функций по настоящему изобретению, таких как одна или более из функций картирования, и/или выравнивания, и/или определения вариантов, которые описаны в настоящем документе, стандартным образом. В частности, на этапе (1) может быть сформирована или иным образом обеспечена структура данных, например, с помощью СНП и/или ЦПУ 1000, после чего эта структура данных может быть сохранена (2) в связанной памяти, такой как DRAM 1014. Структура данных может быть любой структурой данных, например, в том, что касается тех, что описаны в настоящем документе, но в этом случае может быть множеством ридов секвенированных данных, и/или референсным геномом, и/или индексом референсного генома, например, для выполнения функций картирования, и/или выравнивания, и/или определения вариантов.

[00525] На втором этапе (2), например в том, что касается функций картирования и/или выравнивания и т.п, FPGA 7, связанная с ЦПУ 1000, например, посредством интерфейса 3 с тесным связыванием, может получать доступ к связанной памяти 1014 ЦПУ для выполнения одного или более действий в отношении хранящихся секвенированных ридов, референсных геномов и/или их индексов. В частности, на этапе (3), например, в операции картирования, FPGA 7 может получать доступ к структуре данных, например, к секвенированным ридами и/или референсным последовательностям, для создания из них одной или более затравок, например, когда структура данных содержит одно или более ридов и/или последовательностей генома. В таком случае затравки, например, последовательности референса и/или рида, могут быть использованы в целях выполнения с ними хэш-функции, например, для создания одного или более ридов, которое картировано на одну или более позиций относительно референсного генома.

[00526] На дальнейшем этапе (3) полученные картированные результирующие данные могут быть сохранены, например, либо в главной памяти 1014, либо в связанном DRAM 14. Кроме того, после того, как данные картированы, FPGA 7, или ее движок 13 обработки, может быть реконфигурирована, например, частично реконфигурирована, в качестве движка выравнивания, который может затем получить доступ к сохраненной структуре картированных данных для выполнения на ней функции выравнивания с целью создания одного или более ридов, выровненных на референсный геном. На дополнительном этапе (4) главное ЦПУ может затем получить доступ к картированным и/или выровненным данным для выполнения на них одной или более функций, например, для создания графа де Брейна («DBG»), который затем может быть сохранен в его связанной памяти. Аналогичным образом на одном или более дополнительных этапов FPGA 7 может снова обратиться к памяти 1014 главного ЦПУ для получения доступа к DBG и выполнения на нем анализа HMM с целью создания одного или более файлов определения вариантов.

[00527] В конкретных случаях ЦПУ 1000 и FPGA 7 могут иметь одну или более кэш-

памятей, которые благодаря жесткому связыванию интерфейса между этими двумя устройствами обеспечат когерентность отдельных кэшей, например, в отношении промежуточных данных, например, данных результатов, хранящихся в них, например, в результате выполнения в них одной или более функций. Подобным образом данные могут совместно использоваться по существу беспрепятственно жестко связанными устройствами, тем самым позволяя функциям конвейера переплетаться, например, в биоинформационном конвейере. Поэтому в таком случае FPGA 7 больше не нужно иметь свою собственную присоединенную специализированную внешнюю память 14, и, следовательно, благодаря такой жестко связанной конфигурации сохраненные риды, референсный геном и/или индекс референсного генома, которые описаны в настоящем документе, могут интенсивно совместно использоваться, например, с поддержанием когерентности кэша, например, для картирования и выравнивания ридов и других операций обработки геномных данных.

[00528] Кроме того, как показано на ФИГ. 38, жестко связанные и поддерживающие когерентность кэша конфигурации, как и другие конфигурации компонентов, рассмотренные в настоящем документе, позволяют выполнять более мелкие операции низкого уровня в одном устройстве (например, ЦПУ или FPGA), прежде чем возвращать структуру данных или поток 20 обработки на другое устройство, например, для дальнейшей обработки. Например, в одном случае поток 20a ЦПУ может быть выполнен с возможностью организации больших количеств работ в очередь в аппаратной логику 13 FPGA для выполнения, и этот же или другой поток 20b может быть выполнен с возможностью последующей обработки большой очереди сформированных таким образом результатов, например, существенно позже. Однако, в различных случаях, возможно, будет более эффективно, если поток 20 ЦПУ, будет, как описано в настоящем документе, блокировать «вызов функции» в связанный аппаратный движок 13 FPGA, причем ЦПУ может быть установлено на возобновление исполнения программного обеспечения сразу по завершении аппаратной функции FPGA. Следовательно, вместо того, чтобы упаковывать структуры данных для потоковой передачи с помощью DMA 14 в FPGA 7 и распаковывать результаты по их возвращении, программный поток 20 мог бы просто предоставлять указатель памяти в движок 13 FPGA, который мог бы получать доступ к совместно используемой памяти 1014/14 и вносить изменения на месте с поддержанием когерентности кэша.

[00529] В частности, при такой взаимосвязи между структурами, предложенной в настоящем документе, глубина детализации взаимодействия программного обеспечения/аппаратного обеспечения может быть более мелкой, чтобы назначать более мелкие операции низкого уровня для выполнения различными аппаратными движками 13, например, путем вызова функций из различных выделенных программных потоков 20. Например, на слабо связанной платформе ЦПУ/FPGA для эффективного ускорения картирования, выравнивания и/или определения вариантов рида ДНК/РНК, может быть создан полный конвейер картирования/выравнивания/определения вариантов в виде одного или более программных и/или реализованных в FPGA движков, причем некартированные и невыровненные риды передаются в потоковом режиме из программного обеспечения в аппаратное обеспечение, где процесс может быть повторен, например, для определения вариантов. Что касается конфигураций, описанных в настоящем документе, они могут быть очень быстрыми. Однако в различных случаях такая система может страдать от ограничений гибкости, сложности и/или программируемости, например, вследствие того, что весь конвейер картирования/выравнивания и/или определения вариантов реализован в аппаратной схеме, которая,

хотя и выполнена с возможностью реконфигурирования в FPGA, как правило, значительно менее гибкая и программируемая, чем программное обеспечение, и может быть поэтому ограничена меньшей алгоритмической сложностью.

[00530] В отличие от этого за счет использования жесткого межсоединения ЦПУ/FPGA, такого как QPI или другое межсоединение, в конфигурациях, описанных в настоящем документе, несколько ресурсоемких дискретных операций, таких как формирование затравки и/или картирование, восстановительное сканирование, выравнивание без гэпов, выравнивание с гэпами, например, выравнивание Смита-Ватермана и т.д., могут быть реализованы в виде различных доступных по отдельности аппаратных движков 13 (см., например, ФИГ. 38), и общие алгоритмы картирования/выравнивания и/или определения вариантов могут быть реализованы в программном обеспечении, причем ускорение низкого уровня осуществляется за счет обращения к FPGA за специальными дорогостоящими этапами обработки. Такая инфраструктура обеспечивает полную возможность программирования программного обеспечения за рамками определенных вызовов ускорения, и делает возможной более высокую алгоритмическую сложность и гибкость, чем в случае стандартных жестко смонтированных операций.

[00531] Кроме того, в такой инфраструктуре исполнения программного обеспечения, ускоренного путем вызовов низкоуровневого аппаратного ускорения FPGA, функции аппаратного ускорения могут более легко совместно использоваться для множества целей. Например, когда аппаратные движки 13 формируют большие монолитные конвейеры, отдельные подкомпоненты конвейера могут быть, как правило, специально предназначены для их среды и взаимно соединены между собой только в пределах одного конвейера, который, если он не тесно связан, может, вообще говоря, не быть доступным для любой цели. Но многие операции обработки геномных данных, такие как выравнивание Смита-Ватермана, построение графа де Брейна или графа сборки и другие подобные операции, могут быть использованы в различных родительских алгоритмах верхнего уровня. Например, как описано в настоящем документе, выравнивание Смита-Ватермана может быть использовано в картировании и выравнивании рида ДНК/РНК, например, относительно референсного генома, но может быть также выполнено с возможностью применения определителями вариантов на основе гаплотипа, для выравнивания гаплотипов-кандидатов на референсный геном или друг на друга, или для секвенированных последовательностей, например, в анализе HMM и/или функции определения вариантов. Следовательно, привлечение различных дискретных низкоуровневых функций аппаратного ускорения посредством вызовов функций общим программным обеспечением может позволить максимально использовать одну и ту же логику ускорения, например, 13, по всему приложению обработки геномных данных, например, при выполнении как выравнивания, так и определения вариантов, например операций HMM.

[00532] В случае тесного межсоединения ЦПУ/FPGA, с практической точки зрения целесообразно также иметь распределенное, а не централизованное, управление с помощью программного обеспечения ЦПУ 1000 по каналам связи с различными аппаратными движками 13 FPGA, описанными в настоящем документе. В широко распространенных на практике многопотоковых, многоядерных и многопроцессорных архитектурах программного обеспечения множество программных потоков и процессов обмениваются данными и взаимодействуют беспрепятственно без каких-либо центральных программных модулей, драйверов или потоков для управления внутренним обменом данными. В таком формате это практично ввиду совместно используемой

памяти с поддержанием когерентности кэша, которая видна для потоков во всех ядрах всех ЦПУ, хотя физически когерентное совместное использование памяти ядрами и ЦПУ происходит за счет внутреннего обмена данными посредством процессорного межсоединения, например QPI или HT.

[00533] Подобным образом, как показано на ФИГ. 36-38, системы, предложенные в настоящем документе, могут иметь ряд ЦПУ и/или FPGA, которые могут быть в конфигурации тесно связанного межсоединения ЦПУ/FPGA, включающей в себя множество потоков, например, 20a, b, c, и множество процессов, выполняющихся в одном или множестве ядер и/или ЦПУ, например 1000a, 100b и 1000c. Поэтому компоненты системы выполнены с возможностью обмена данными и взаимодействия с распределением между друг другом, например, между всевозможными разными ЦПУ и/или аппаратными движками ускорения FPGA, например, за счет совместного использования памяти с поддержание когерентности кэша различными ЦПУ и FPGA. Например, как показано на ФИГ. 36, множество ядер ЦПУ 1000a, 1000b и 1000c могут быть связаны вместе так, чтобы совместно использовать одну или более памятей, например, DRAM 1014, и/или один или более кэшей, имеющих один или более слоев, например, L1, L2, L3 и т.д, или уровней, связанных с ними. Аналогичным образом, как показано на ФИГ, 38, в другом варианте реализации одно ЦПУ 1000 может быть выполнено с возможностью содержания множества ядер 1000a, 1000b и 1000c, которые могут быть связаны вместе таким образом, чтобы совместно использовать одну или более памятей, например, DRAM 1014, и/или один или более кэшей 1014a, имеющих один или более слоев или уровней, связанных с ними.

[00534] Поэтому в другом варианте реализации подлежащие обработке данные из одного или более программных потоков 20 из одного или более ядер 1000 ЦПУ в аппаратный движок 13, например, из FPGA, или наоборот может непрерывно и/или беспрепятственно обновляться в совместно используемой памяти 1014 или кэше и/или его слое, который виден каждому устройству. Кроме того, запросы на обработку данных в совместно используемой памяти 1014 или уведомления о результатах, обновляемых там, могут передаваться посредством сигнализации между программным и/или аппаратным обеспечением, например, по соответствующим образом сконфигурированной шине, например, шине DDR4, например, в очередях, которые могут быть реализованы внутри самой совместно используемой памяти. Для координации программного/аппаратного обеспечения могут быть также реализованы стандартные программные движки управления, передачи и защиты данных, такие как семафоры, взаимоисключающие блокировки и неразложимые целые числа.

[00535] Следовательно, в некоторых вариантах реализации, пример которых приведен на ФИГ. 36, и которые не требуют наличия своей собственной специализированной памяти 14 или иных внешних ресурсов у FPGA 7 ввиду когерентного совместного использования памяти посредством тесной межсоединения ЦПУ/FPGA, становится намного практичнее упаковывать FPGA 7 более компактным и присущим исходной системе образом в пределах традиционных материнских плат ЦПУ 1000 без использования плат расширения. См., например, ФИГ. 34A и 34B и ФИГ. 35. Существуют несколько вариантов упаковки. А именно, FPGA 7 может быть установлена на многопроцессорную материнскую плату в гнездо ЦПУ, как показано на ФИГ. 34A и 34B, например, с помощью надлежащей переходной платы, такой как небольшая плата 2 PC или альтернативная связанная проводами упаковка кристалла FPGA в пределах корпуса 2a микросхемы ЦПУ, где контакты гнезда ЦПУ соответствующим образом соединены с контактами FPGA, включая соединения питания и заземления, процессорное

межсоединение 3 (QPI, HT и т.д.) и другие системные соединения. Соответственно, кристалл FPGA и кристалл ЦПУ могут быть включены в один и тот же многокристальный корпус (MCP) с необходимыми соединениями, включая питание, заземление и межсоединение ЦПУ/FPGA, созданные внутри корпуса 2a. Межкристальные соединения могут быть выполнены методом межкристального проволочного монтажа, или путем соединения с общей подложкой или переходной платой, или с помощью связанных контактных площадок или сквозных отверстий через кремний между многоуровневыми кристаллами.

[00536] Кроме того, в различных реализациях FPGA и ядра ЦПУ могут быть изготовлены на одном кристалле (см. ФИГ. 35) методом «система на микросхеме» (SOC). В любом из этих случаев заказная логика, например, 17, может быть реализована внутри FPGA 7, как для обмена данными по межсоединению 3 ЦПУ/FPGA, например, с помощью надлежащим образом созданных протоколов, так и обслуживания, преобразования и/или маршрутизации запросов на доступ к памяти из внутренних движков 13 FPGA через межсоединение 3 ЦПУ/FPGA посредством надлежащих протоколов в совместно используемую память 1014a. Кроме того, вся эта логика или ее некоторая часть могут быть реализованы в заказном кремниевом кристалле во избежание использования пространства логики FPGA в этих целях, например, когда отвержденная логика может находиться на кристалле ЦПУ и/или кристалле FPGA, или на отдельном кристалле. Кроме того, в любом из этих случаев требования к подаче электропитания и теплоотдаче могут быть достигнуты соответствующим образом, например, в одном корпусе (MCP или SOC). Кроме того, размер FPGA и количество ядер ЦПУ можно выбрать так, чтобы оставаться в пределах безопасной мощности огибающей, и/или можно использовать динамические методы (управление тактовой частотой, тактовое стробирование, отключение ядра, силовые острова и т.д.) для регулирования потребления энергии в соответствии с изменением потребности в вычислениях ЦПУ и/или FPGA.

[00537] Все эти варианты упаковки имеют ряд общих преимуществ. Жестко интегрированная платформа ЦПУ/FPGA становится совместимой со стандартными материнскими платами и/или корпусами системы различных размеров. В случае установки FPGA через переходную плату в гнездо ЦПУ (см. ФИГ. 34A и 34B) можно использовать по меньшей мере двухгнездовую материнскую плату 1002. В других случаях можно использовать четырехгнездовую материнскую плату, чтобы обеспечить возможность реализации конфигураций 3 ЦПУ + 1 FPGA, 2 ЦПУ + 2 FPGA или 1 CPU + 3 FPGA и т.д. Если FPGA находится в одном корпусе с ЦПУ CPU (либо MCP, либо SOC), то можно использовать одногнездовую материнскую плату, потенциально в очень маленьком корпусе системы (хотя изображена двухгнездовая материнская плата); это также очень хорошо подходить для масштабирования в сторону увеличения, например, 4 FPGA и 4 многоядерных ЦПУ на 4-гнездовой серверной материнской плате, которые, тем не менее, могут работать в компактном корпусе, например в виде сервера 1U, монтируемого в стойке.

[00538] Соответственно, в различных случаях поэтому может существовать потребность в установке платы расширения, чтобы интегрировать ЦПУ и ускорение FPGA, так как FPGA 7 может быть интегрирована в гнездо 1003 ЦПУ. Такая реализация избавляет от дополнительных требований к пространству и электропитанию для платы расширения, и исключает вероятность появления различных дополнительных отказов, которые иногда возникают с картами расширения вследствие компонентов с относительно низкой надежностью. Кроме того, к FPGA или ЦПУ/FPGA в корпусах

или в гнездах ЦПУ можно применять стандартные решения для охлаждения ЦПУ (теплоотводы, тепловые трубки и/или вентиляторы), которые эффективны даже при их низкой стоимости, поскольку изготовляются в больших объемах, тогда как охлаждение плат расширения может быть дорогостоящим и неэффективным.

[00539] Аналогичным образом FPGA/переходная плата и/или ЦПУ/FPGA в одном корпусе могут полностью использовать питание из гнезда ЦПУ, например, 150 Вт, тогда как плата расширения может быть ограничена по питанию, например, 25 Вт или 75 Вт из шины PCIe. В различных случаях для приложений обработки геномных данных все эти варианты упаковки могут облегчить установку тесно связанной платформы вычисления ЦПУ + FPGA, например внутри секвенатора ДНК. Например, типичные современные секвенаторы ДНК «нового поколения» содержат оборудование для секвенирования (хранилище для образцов и реагентов, трубки и средства регулирования для текучей среды, матрицы датчиков, средства первичной обработки изображения и/или сигнала) в корпусе, который также содержит стандартную или заказную серверную материнскую плату, соединенную проводами с устройством секвенирования для управления секвенированием и получения данных секвенирования. Жестко интегрированная платформа ЦПУ + FPGA, описанная в настоящем документе, может быть получена в таком секвенаторе, например, путем простой установки в гнезда ЦПУ имеющейся у него материнской платы одного или более комплекса FPGA/материнская плата и/или FPGA/ЦПУ в одном корпусе, или, в альтернативном варианте реализации, путем установки новой материнской платы с ЦПУ и FPGA, например, тесно связанных, как описано в настоящем документе. Кроме того, все эти варианты упаковки могут быть выполнены с возможностью способствования упрощению разработки тесно интегрированной платформы ЦПУ + FPGA, например, в доступной из облака и/или находящей в центре данных серверной стойке, в которую входят компактные/плотные серверы с очень высокой надежностью/доступностью.

[00540] Следовательно, в соответствии с идеями, изложенными в настоящем документе, существует множество этапов обработки для картирования и выравнивания, сортировки и/или удаления дубликатов, для определения вариантов данных секвенирования ДНК (или РНК), которые могут меняться в зависимости от используемых технологий первичной, и/или вторичной, и/или третичной обработки и их применений. В число таких этапов обработки могут входить одно или более из следующего: обработка сигнала на электрических измерениях из секвенатора, обработка изображения на оптических измерениях из секвенатора, определение оснований с помощью обработанных данных сигнала или изображения для определения наиболее вероятной нуклеотидной последовательности и оценок достоверности, фильтрация секвенированных ридов с низким качеством или поликлональными кластерами, обнаружение и обрезание адаптеров, ключевых последовательностей, штрихкодов и концов ридов низкого качества, а также de novo сборка последовательности, формирование и/или использование графов де Брейна и/или графов последовательности, например, построение графа де Брейна и графа последовательности, редактирование, обрезание, очистка, окрашивание, аннотирование, сравнение, преобразование, расщепление, анализ, выбора подграфа, прохождение, обратное прохождение, поиск, фильтрация, импорт, экспорт, в том числе картирование ридов на референсный геном, выравнивание ридов на возможные местоположения картирования в референсном геноме, локальная сборка ридов, картированных на референсную область, сортировка ридов по выровненным позициям, маркировка и/или удаление перекрывающихся ридов для соответствия инделов, перекалибровка оценки качества оснований, определение

вариантов (одного образца или совместно), анализ структурных вариантов, анализ количество копий вариантов, определение соматических вариантов (например, только образца опухоли, совпадений опухоль/нормальная или опухоль/несовпавшая нормальная и т.д.), обнаружение границы сплайсинга РНК, анализ альтернативного сплайсинга РНК, сборка транскрипта РНК, анализ экспрессии транскрипта РНК, анализ дифференциальной экспрессии РНК, определение вариантов РНК, анализ отличия ДНК/РНК, анализ и определение метилирования ДНК, перекалибровка оценки качества вариантов, фильтрация вариантов, аннотирование вариантов с помощью баз данных известных вариантов, обнаружение и оценка загрязнения образца, прогнозирование фенотипа, тестирование на заболевание, прогнозирование реакции на терапию, разработка индивидуальной терапии, анализ родословной и истории мутации, анализ ДНК популяции, выявление генетических маркеров, кодирование геномных данных в стандартные форматы и/или файлы сжатия (например, FASTA, FASTQ, SAM, BAM, VCF, BCF), декодирование геномных данных из стандартных форматов, запрос, выбор или фильтрация подмножеств геномных данных, общие сжатие или распаковка геномных файлов (сжатие gzip, BAM), специализированные сжатие и распаковка геномных данных (CRAM), шифрование и расшифрование геномных данных, вычисление статистики, сравнение и представление геномных данных, сравнение результирующих геномных данных, анализ точности и составление отчета, сохранение, архивирование, извлечение, резервное копирование, восстановление и передача геномного файла, а также построение геномной базы данных, выполнение запросов, управление доступом, выделение данных и т.п.

[00541] Все эти операции могут быть довольно медленными и дорогостоящими при реализации на традиционных вычислительных платформах. Медлительность таких операций, реализованных исключительно программным способом, может быть, отчасти, вызвана сложностью алгоритмов, но, как правило, обусловлена очень вводом и выводом очень больших наборов данных, что приводит к большой задержке по сравнению с движением данных. Устройства или системы, описанные в настоящем документе, преодолевают эти проблемы, в частности, за счет конфигурации различных аппаратных движков обработки, ускорения с помощью различных аппаратных реализаций и/или, частично, за счет тесного связывания ЦПУ/FPGA. Соответственно, как показано на ФИГ. 39, одна или более, например, все, из этих операций могут быть ускорены за счет взаимодействия ЦПУ 1000 и FPGA 7, например, в модели распределенной обработки, как описано в настоящем документе. Например, в некоторых случаях (шифрование, общее сжатие, картирование и/или выравнивание ридов) вся операционная функция может быть по существу или полностью реализована в заказной логике FPGA (например, с помощью методики разработки аппаратного обеспечения, например, RTL), например, когда программное обеспечение ЦПУ в основном исполняет функцию компиляции пакетов больших данных для обработки рабочими потоками 20, например, путем агрегирования данных в различные задания, подлежащие обработке одним или более жестко смонтированными движками обработки, и подачи различных входных данных, например, в формате «первым пришел, первым обслужен», в один или более движков 13 FPGA и/или принимает результаты из них.

[00542] Например, как показано на ФИГ. 39, в различных вариантах реализации рабочий поток формирует различные пакеты данных задания, которые могут быть скомпилированы и/или переданы в потоковом режиме в более крупные пакеты заданий, которые могут быть поставлены в очередь и/или дополнительно агрегированы в качестве подготовки для передачи, например, посредством DDR3 в 7, например, посредством

протокола широкополосной двухточечной связи с малой задержкой, например QPI 3. В конкретных случаях данные могут быть буферизованы в соответствии с конкретными наборами данных, передаваемыми в FPGA. После того, как объединенные в пакет данные приняты FPGA 7, например, с поддержкой когерентности кэша, они могут обработаны и отправлены в один или более специализированных кластеров 11, откуда они могут быть направлены далее в один или более наборов движков обработки для обработки их там в соответствии с одной или более операций конвейера, описанных в настоящем документе.

[00543] После обработки данные результатов могут быть отправлены обратно в кластер и поставлены в очередь на отправку обратно по двухточечному межсоединению тесного связывания в ЦПУ для последующей обработки. В определенных вариантах реализации данные могут быть отправлены в поток деагрегатора для последующей обработки. По завершении последующей обработки данные могут быть отправлен обратно в первоначальный рабочий поток 20, который может ожидать эти данные. Такая распределенная обработка особенно полезна в случае функций, описанных выше в настоящем документе. В частности, эти функции отличаются тем, что их алгоритмическая сложность (хотя и требующая очень больших затрат сетевых вычислительных ресурсов) довольно ограничена и каждая из них может быть выполнена с возможностью обладания довольно равномерной вычислительной стоимостью всех своих различных подопераций.

[00544] Однако в различных случаях вместо обработки данных в больших пакетах могут выполняться более мелкие подпрограммы или протоколы отдельно взятых функций или элементов, например, относящиеся к одной или более функциям конвейера, а не выполняющие функции полной обработки для этого конвейера на этих данных. Следовательно, полезная стратегия может заключаться в выявлении одной или более критических функций для ресурсоемких вычислений в любой данной операции и затем реализовать эту подфункцию в заказной логике FPGA (аппаратное ускорение), например, для ресурсоемких подфункций, а остальную часть операции, и в идеале большую или значительную часть алгоритмической сложности, реализовать в программном обеспечении для выполнения в ЦПУ/ГПУ/КПУ, как описано в настоящем документе, например, в соответствии с ФИГ. 39.

[00545] Как правило, многие операции обработки геномных данных отличаются именно тем, что на небольшой процент алгоритмической сложности приходится большой процент общей вычислительной нагрузки. Вот типичный пример: на 20% алгоритмической сложности для выполнения данной функции могут приходиться 90% вычислительной нагрузки, тогда как на остальные 80% алгоритмической сложности может приходиться только 10% вычислительной нагрузки. Следовательно, в различных случаях компоненты системы, описанной в настоящем документе, могут быть выполнены с возможностью реализации большой, например, 20% или более, части сложности для осуществления с высокой эффективностью в заказной логике FPGA, которая может быть выполнена с возможностью отслеживания и управления в аппаратной конструкции и, таким образом, может быть выполнена с возможностью осуществления этого в FPGA; что, в свою очередь, может снизить вычислительную нагрузку на ЦПУ на 90%, тем самым обеспечив 10-кратное общее ускорение. Другие типичные примеры могут быть даже еще более экстремальными, например, когда на 10% логарифмической сложности может приходиться 98% вычислительной нагрузки, и в таком случае применения ускорения FPGA, как описано в настоящем документе, к составляющей 10% части сложности может быть даже еще проще, но может также

обеспечить до 50-кратного чистого ускорения. В различных случаях, где требуется предельно ускоренная обработки, одна или более функций могут выполняться квантовым вычислительным устройством.

[00546] Однако такие подходы к ускорению на основе «раздробленной» или распределенной обработки могут более практичными при реализации на тесно интегрированной платформе ЦПУ/ГПУ + FPGA, а не на слабо интегрированной платформе ЦПУ/ГПУ + FPGA. В частности, на слабо интегрированной платформе часть, например, функции, подлежащие реализации в логике FPGA, могут быть выбраны таким образом, чтобы свести к минимуму ввод данных в движки FPGA и свести к минимуму вывода данных из движков FPGA, например, для каждой единицы обработанных данных, и дополнительно можно было выполнить их с возможностью поддержания границы между программным/аппаратным обеспечением, способной выдерживать большие задержки. В таких случаях граница между аппаратной и программной частями может усиливаться, например, на слабо интегрированной платформе, для протаскивания через определенные точки стыка узкой полосы пропускания/широкой полосы пропускания, причем эти разделения могут быть в противном случае нежелательны в других отношениях при оптимизации разбиения на части алгоритмической сложности и вычислительной нагрузки. Этом может зачастую приводить к удлинению границ аппаратной части, охватывающих нежелательно большую часть алгоритмической сложности в жестко смонтированном формате, или к сжатию границ аппаратной части с нежелательным исключением частей с плотной вычислительной нагрузкой.

[00547] В отличие от этого на жестко связанной платформе ЦПУ/ГПУ + FPGA благодаря совестно используемой памяти с поддержанием когерентности кэша и широкополосному межсоединению с малой задержкой ЦПУ/ГПУ/FPGA части операции обработки геномных данных с низкой сложностью/высокой вычислительной нагрузкой могут быть выбраны очень точно для реализации в заказной логике FPGA (например, с помощью аппаратных движков, описанных в настоящем документе), при оптимизированных границах между программным/аппаратным обеспечением. В таком случае, даже если единица данных на желательной границе между программным/аппаратным обеспечением большая, она все равно может быть передана на обслуживание в аппаратный движок FPGA для обработки, просто за счет передачи указателя в конкретную единицу данных. В частности, в случае, который показан на ФИГ. 33B, аппаратный движок 13 FPGA 7 может обойтись без доступа к каждому элементу единицы данных, хранящихся в DRAM 1014; вместо этого он может получать доступ к необходимым элементам, например, в кэше 1014a, с помощью небольших эффективных доступов по межсоединению 3' с малой задержкой, обслуживающему кэш ЦПУ/ГПУ, те самым потребляя меньше совокупной полосы пропускания, чем в случае, если бы нужно было получить доступ ко всей единице данных и/или передать ее на FPGA 7, например, с помощью DMA памяти DRAM 1014 по слабому межсоединению 3, как показано на. 33A.

[00548] В таких случаях аппаратный движок 13 может аннотировать результаты обработки единице данных на месте в памяти 1014 ЦПУ/ГПУ, не передавая в потоковом режиме полную копию единицы данных с помощью DMA в память ЦПУ/ГПУ. Даже если требуемая граница между программным/аппаратным обеспечением не подходит для того, чтобы программный поток 20 выполнял с большой задержкой неблокирующую передачу на обслуживание в порядке очереди в аппаратный движок 13, она потенциально может создать блокирующий вызов функции в аппаратный движок 13, переходя в режим

сна на время короткой задержки, пока аппаратные движки не завершат работу, причем эта задержка резко уменьшается с помощью совместной памяти с поддержанием когерентности кэша, высокоскоростного межсоединения с малой задержкой и распределенной модели координации между программным/аппаратным обеспечением, как показано на ФИГ. 33В.

[00549] В конкретных случаях ввиду того, специфические алгоритмы и требования к обработке сигнала/изображения и определению оснований меняются от одной технологии секвенатора к другой, и поскольку количество необработанных данных с датчика секвенатора обычно колоссальное (оно сокращается до огромного после обработки сигнала/изображения и до умеренно большого после определения оснований), такие обработка сигнала/изображения и определение оснований могут эффективно выполняться в самом секвенаторе или на расположенном по соседству вычислительном сервере, соединенном посредством широкополосного канала передачи с секвенатором. Однако пропускная способность секвенаторов ДНК постоянно увеличивалась с темпами роста, превышающими закон Мура, так что существующих основанных на центральном процессорном устройстве(«ЦПУ») и/или графическом процессорном устройстве («ГПУ») обработок сигнала/изображения и определения оснований, реализуемых по отдельности и каждый сам по себе, стало все больше и больше не хватать для выполнения этой задачи. Тем не менее, поскольку жестко интегрированные платформы ЦПУ + FPGA и/или ЦПУ + FPGA и/или ГПУ/ЦПУ + FPGA могут быть выполнены компактными и легко реализуемыми в таком секвенаторе, например, в виде микросхемы ЦПУ, и/или ГПУ, и/или FPGA, помещенной на материнскую плату секвенатора, или легко устанавливаемое в сервер возле секвенатора, или облачную серверную систему с дистанционным доступом из секвенатора, такой секвенатор может быть идеальной платформой для обеспечения ускорения громоздких вычислений, оказываемого аппаратными движками FPGA/ASIC, описанными в настоящем документе.

[00550] Например, система, предложенная в настоящем документе, может быть выполнена с возможностью осуществления первичной, вторичной и/или третичной обработки, или ее части таким образом, чтобы реализовывать ее с помощью ускоренной платформы ЦПУ, ГПУ и/или FPGA; ЦПУ + FPGA; ГПУ + FPGA; ГПУ/ЦПУ + FPGA; КПУ; ЦПУ/КПУ; ГПУ/КПУ; ЦПУ и/или, ГПУ, и/или КПУ + FPGA. Кроме того, такие ускоренные платформы, например, содержащие один или более аппаратных движков FPGA и/или КПУ, полезны для реализации в облачных системах, как описано в настоящем документе. Например, обработка сигнала/изображения, алгоритмы определения оснований, картирования, выравнивания, сортировки, удаления дубликатов и/или определения вариантов, или их части, обычно требуют большого количества математических операций с плавающей запятой и/или фиксированной запятой, в особенности сложений и умножений. Эти функции могут быть также выполнены с возможностью осуществления одной или более схем квантовой обработки, например, реализованы на квантовой платформе обработки.

[00551] В частности, большие современные FPGA/квантовые схемы содержат тысячи высокоскоростных ресурсов умножения и сложения. Более конкретно, эти схемы могут содержать заказные движки, которые могут быть реализованы в них или с их помощью, причем заказные движки могут быть выполнены с возможностью осуществления параллельных арифметических операций со скоростями далеко превышающими возможности простых ЦПУ общего назначения. Аналогичным образом простые ГПУ имеют более сравнимые ресурсы для параллельных арифметических операций. Тем не менее, ГПУ часто имеют неудобные ограничения в отношении архитектуры и

программирования, которые могут сделать невозможным использование их в полной мере. Соответственно, эти арифметические ресурсы FPGA, и/или квантовой обработки, и/или ГПУ могут быть собраны в схему или иным образом сконфигурированную конструкцию для работы в точности предусмотренным образом с эффективностью почти 100%, например, для осуществления вычислений, необходимых функций, описанных в настоящем документе. Соответственно, можно добавить плату ГПУ в гнезда расширения на материнской плате с жестко интегрированными ЦПУ и/или FPGA, тем самым обеспечив возможность взаимодействия всех трех типов процессоров, хотя ГПУ может все же взаимодействовать со всеми своими собственным ограничениями и ограничениями слабой интеграции.

[00552] Более конкретно, что касается графических процессорных устройств (ГПУ), в различных случаях ГПУ может быть выполнено с возможностью реализации одной или более функций, как описано в настоящем документе, для ускорения скорости обработки основополагающих вычислений, необходимых для выполнения данной функции, полностью или частично. Более конкретно, ГПУ может быть выполнено с возможностью осуществления одной или более задач в протоколе картирования, выравнивания, сортировки, удаления дубликатов и/или определения вариантов, например, для ускорения одного или более вычислений, например, большого количества математических операций с плавающей запятой и/или фиксированной запятой, например, связанных с ними сложений и умножений, для совместной работы с ЦПУ и/или FPGA сервера с целью ускорения выполнения приложения или обработки и сокращения циклов вычислений, требуемых для осуществления таких функций. Облачные серверы, описанные в настоящем документе, с платами ГПУ/ЦПУ/FPGA могут быть выполнены с возможностью решения без труда ресурсоемких вычислительных задачам и обеспечения менее проблемного взаимодействия с пользователем при использовании для визуализации. Такие задачи, требующие ресурсоемких вычислений, могут быть также сброшены на облако, например, для выполнения квантовым вычислительным устройством.

[00553] Соответственно, если жестко интегрированные платформы ЦПУ + FPGA или ГПУ + FPGA и/или ЦПУ/ГПУ/FPGA с совместно используемой памятью применяются в секвенаторе или на прикрепленном или облачном сервере, например, для обработки сигнала/изображения, функций определения оснований, картирования, выравнивания, сортировки. удаления дубликатов и/или определения вариантов, то можно добиться выигрыша, например, в процессе инкрементальной разработки. Например, первоначально ограниченную часть вычислительной нагрузки, такую как программирование функции для определения оснований, картирования, выравнивания, сортировки, удаления дубликатов и/или определения вариантов, можно реализовать в одном или более движках FPGA, тогда как остальная работа может выполняться в платах расширения ЦПУ и/или ГПУ, Однако, модель с жесткой интеграцией ЦПУ/ГПУ/FPGA совместно используемой памятью, представленная в настоящем документе, может быть позже дополнительно сконфигурирована, чтобы облегчить инкрементальный выбор дополнительных требующих ресурсоемких вычислений функций для ускорения с помощью ГПУ, FPGA и/или квантового ускорения, которые могут быть затем реализованы в виде движков обработки, и различные их функции могут быть сброшены для выполнения в FPGA и/или в некоторых случаях могут быть сброшены на облако, например, для выполнения с помощью КПУ, тем самым ускоряя обработку сигнала/изображения/определения оснований/картирования/выравнивания/определения вариантов. Такие инкрементальные продвижения могут быть реализованы

по мере необходимости, чтобы не отставать от растущей пропускной способности различных технологий первичной, и/или вторичной, и/или третичной обработки.

[00554] Следовательно, картирование и выравнивание ридов, например, одного или более ридов, на референсный геном, как и сортировка, удаление дубликатов и/или определение вариантов, могут выиграть от такого ускорения с помощью ГПУ и/или FPGA, или КПУ. А именно, картирование и выравнивание и/или определение вариантов, или их части, могут быть реализованы частично или полностью виде заказной логики FPGA, например, с помощью потоковой передачи «подлежащих картированию, и/или выравниванию, и/или определению вариантов» ридов из памяти ЦПУ/ГПУ в движки картирования/выравнивания/определения вариантов FPGA и обратной потоковой передачи записей картированных, и/или выровненных, и/или подвергнутых определению вариантов ридов, которые могут быть затем отправлены в потоковом режиме на плату, например, при выполнении сортировки и/или определения вариантов. Ускорение FPGA работает только на слабо связанной платформе ЦПУ/ГПУ + FPGA, и в конфигурациях, описанных в настоящем документе, может быть чрезвычайно быстрым. Тем не менее, существуют некоторые дополнительные преимущества, которые могут быть достигнуты за счет перехода на жестко интегрированную платформу ЦПУ/ГПУ/КПУ + FPGA.

[00555] Соответственно, что касается картирования, выравнивания и определения вариантов, в некоторых вариантах реализации общее преимущество жестко интегрированных ЦПУ/ГПУ + FPGA и/или квантовой платформы обработки, как описано в настоящем документе, состоит в том, что ускорение картирования/выравнивания/определения вариантов, например, аппаратное ускорение, может быть эффективно разбито на несколько дискретных требующих ресурсоемких вычислений операций, таких как формирование и/или картирование затравки, формирование цепочки затравки, восстановительное сканирование спаренных концов, выравнивание без гэпов и выравнивание с гэпами (Смита-Ватермана или Нидлмана-Вунша), формирование графа де Брейна, выполнение вычисления HMM и т.п., например, когда программное обеспечение ЦПУ, и/или ГПУ, и/или квантового вычисления выполняет более легкие (но необязательно менее сложные) задачи и может совершать вызовы ускорения из дискретных аппаратных и/или других движков квантового вычисления по мере надобности. Такая модель может быть менее эффективной на типичной слабо интегрированной платформе ЦПУ/ГПУ + FPGA, например, вследствие больших объемов данных, подлежащих передаче туда и обратно между этапами, и больших задержек, но могут быть более эффективными на жестко интегрированной платформе ЦПУ + FPGA, ГПУ + FPGA и/или квантового вычисления с совместно используемой памятью с поддержание когерентности кэша, широкополосным межсоединением с малой задержкой и распределенной моделью координации программного/аппаратного обеспечения. Кроме того, например, в том, что касается определения вариантов, алгоритмы скрытой марковской модели (HMM) и/или динамического программирования (ДП), включая алгоритмы Витерби и алгоритм прямого хода, могут быть реализованы совместно с операцией определения оснований/картирования/выравнивания/сортировки/удаления дубликатов, например, для сравнения наиболее вероятной исходной последовательности, объясняющей наблюдаемые измерения датчика, в конфигурации, например, хорошо подходящей для параллельной ячеистых схем FPGA и/или квантовых схем, описанных в настоящем документе.

[00556] А именно, эффективное использование аппаратных и/или программных ресурсов в распределенной конфигурации обработки может быть результатом сокращения ускорения с помощью аппаратного обеспечения и/или квантового

вычисления для дискретных требующих ресурсоемких вычислений функций. В таких случаях несколько функций, описанных в настоящем документе, могут выполняться в монолитном строго аппаратном движке, чтобы требовать меньше ресурсоемких вычислений, но могут, несмотря на это, оставаться алгоритмически сложными и, следовательно, могут потреблять значительные количества физических ресурсов FPGA (таблицы подстановки, триггеры, блочные ОЗУ и т.д.). В таких случаях переход части или всех различных дискретных функций на программное обеспечение взял бы на себя имеющиеся циклы ЦПУ взамен на освобождение площади FPGA существенного размера. В определенных подобных случаях освобожденная площадь FPGA может быть использована для организации большей параллельности требующих ресурсоемких вычислений подфункций картирования/выравнивания/определения вариантов и таким образом усиления ускорения, или для других геномных функций ускорения. Таких преимуществ можно также достичь за счет реализации требующих ресурсоемких вычислений функций в одной или более специализированных квантовых схем для реализации квантовой вычислительной платформы.

[00557] Следовательно, в различных вариантах реализации алгоритмическая сложность одной или более функций, описанных в настоящем документе, может быть несколько снижена путем выполнения их строго в аппаратном обеспечении или строго в реализации квантовых вычислений. Однако, некоторые операции, такие как сравнение пар возможных выравниваний для ридов со спаренными концами и/или выполнения едва различимых оценок качества картирования (MAPQ), представляют очень низкие вычислительные нагрузки, и поэтому могли бы выиграть от более сложной и точной обработки в программном обеспечении ЦПУ/ГПУ и/или квантового вычисления. Поэтому, как правило, сокращение аппаратной обработки до специфических требующих ресурсоемких вычислений операций позволит использовать более сложные и точные алгоритмы в частях ЦПУ/ГПУ,

[00558] Кроме того, в различных вариантах реализации все или часть операций картирования/выравнивания/сортировки/удаления дубликатов/определения вариантов, описанные в настоящем документе, могут быть выполнены таким образом, чтобы более сложные алгоритмически вычисления могли выполняться на высоких уровнях в аппаратном обеспечении и/или в одной или более квантовых схем, например, когда вызываемые требующие ресурсоемких вычислений аппаратные и/или квантовые функции выполнены с возможностью осуществления в динамическом или итеративном порядке. В частности, монолитная конструкция строго аппаратной/квантовой обработки может быть реализована с возможностью более эффективного функционирования в линейном конвейере. Например, если во время обработки одно отображаемое выравнивание Смита-Ватермана свидетельствует о том, что истинный путь выравнивания выходит за пределы полосы оценки, например, полосы захвата, как описано выше, для исправления этого нужно вызвать еще одно выравнивание Смита-Ватермана. Следовательно, эти конфигурации могут по существу свести ускорение с помощью аппаратного обеспечения FPGA/квантовое ускорение к дискретным функциям, таким как формирование процедурных абстракций, которые сделают возможным более легкое создание сложности более высокого уровня поверх них.

[00559] Кроме того, в различных случаях гибкость в пределах алгоритмов картирования/выравнивания/определения вариантов и их функций может быть улучшена путем сведения программного и/или квантового ускорения к дискретным требующим ресурсоемких вычислений функциям и выполнения системы с возможностью осуществления других, например, менее ресурсоемких, частей, в программном

обеспечении ЦПУ и/или ГПУ, Например, хотя аппаратные алгоритмы могут быть модифицированы и реконфигурированы в FPGA, как правило, такие изменения в аппаратных конструкциях, например посредством прошивки, могут потребовать в несколько раз больших усилий, чем подобные изменения в программном обеспечении. В таких случаях требующие ресурсоемких вычислений части картирования, и выравнивания, и сортировки, и удаления дубликатов и/или определения вариантов, такие как картирования затравки, формирование цепочки затравки, восстановительное сканирование спаренных концов, выравнивание без гэпов, выравнивание с гэпам и HMM, которые относительно хорошо определены, являются поэтому стабильными функциями и не требуют частых алгоритмических изменений. Следовательно, эти функции могут быть соответствующим образом оптимизированы в аппаратном оборудовании, тогда как другие функции, которые могут быть выполнены программным обеспечением ЦПУ/ГПУ, больше подходят для инкрементального улучшения алгоритмов, что значительно проще в программном обеспечении. Однако после полной отработки они могут быть реализованы в аппаратном обеспечении. Одна или более из этих функций могут быть также выполнены с возможностью реализации в одной или более квантовых схем машины квантовой обработки.

[00560] Соответственно, в различных случаях определение вариантов (по отношению к ДНК или РНК, одного образца или совместной, генеративной или соматической и т.д.) тоже может выиграть от ускорения с помощью FPGA и/или квантового ускорения, например, в отношении его различных требующих ресурсоемких вычислений функций. Например, определители на основе гаплотипов, которые определяют основания на основе подтверждающих данных, полученных из контекста, обеспечиваемого в пределах окна вокруг потенциального варианта, как описано в выше, частот являются наиболее требовательными к ресурсоемким вычислениям. Эти операции, включая сравнение гаплотипа-кандидата (например, однонитевой нуклеотидной последовательности, представляющей гипотезу истинности последовательности по меньшей мере одной из нитей образца в исследуемом локусе генома) с каждым ридом секвенатора, например, для оценки условной вероятности наблюдения рида при условии истинности данного гаплотипа.

[00561] Такую операцию можно выполнить с помощью одного или более вычислений MRJD, парной скрытой марковской модели (парная-HMM) и/или частично определенной марковской модели (PD-HMM), которое суммирует вероятности возможных комбинаций ошибок при секвенировании или приготовлении образца (ПЦР и т.д.) с помощью алгоритма динамического программирования. Следовательно, в подобных случаях система может быть выполнена с возможностью ускорения вычисления парной-HMM или PD-HMM с помощью одного или более, например, параллельных, аппаратных движков FPGA или движков квантовой обработки, где программное обеспечение ЦПУ/ГПУ/КПУ может быть выполнено с возможностью исполнения остальной части родительского алгоритма определения вариантов на основе гаплотипов, либо на слабо интегрированной платформе, либо на жестко интегрированной платформе ЦПУ + FPGA, или ГПУ + FPGA, или ЦПУ и/или ГПУ + FPGA, и/или КПУ. Например, при слабой интеграции программные потоки могут строить и подготавливать граф де Брейна и/или граф сборки из ридов, перекрывающих выбранную активную область (окно или непрерывное подмножество референсного генома), выделять гаплотипы-кандидаты из графа и выстраивать в очередь пары гаплотип-рид для передачи с помощью DMA в аппаратные движки FPGA, например, для сравнения парной-HMM или PD-HMM. Те же самые или другие программные потоки могут затем принимать

результаты парной-HMM, поставленные в очередь и переданные с помощью DMA обратно из FPGA в память ЦПУ/ГПУ и выполнять генотипирование и байесовское вычисление вероятностей для осуществления окончательного определения вариантов. Конечно, одна или более из этих функций могут быть выполнены с возможностью выполнения на одной или более квантовых вычислительных платформ.

[00562] Например, как показано на ФИГ. 38, ЦПУ/ГПУ 1000 может включать в себя один или более, например, множество, потоков 20a, 20b и 20c, каждый из которых может иметь доступ к связанному DRAM 1014, причем DRAM имеет рабочие пространства 1014a, 1014b и 1014c, в пределах которых каждый поток 20a, 20bи 20c может иметь доступ, соответственно, для выполнения одной или более операций на одной или более структур данных, таких как большие структуры данных. Эти части памяти и их структуры данных могут быть доступны, например, через соответствующие части 1014a' кэша, например, для одного или более движков 13a, 13b, 13c обработки FPGA 7, причем движки обработки могут получать доступ к структурам референсных данных, например, при выполнении одной или более операций, описанных в настоящем документе, таких как картирование, выравнивание, сортировка и/или поиск вариантов. Благодаря широкополосному тесно связывающему межсоединению 3 данные, относящиеся к структурам данных и/или связанные с результатами обработки, могут по существу беспрепятственно совместно использоваться ЦПУ, и/или ГПУ, и/или КПУ, и/или связанной FPGA, например, с поддержанием когерентности кэша, для оптимизации эффективности обработки.

[00563] Соответственно, согласно одному аспекту может быть предусмотрена система, которая может быть выполнена с возможностью совместного использования ресурсов памяти среди ее составных частей, например, при выполнении некоторых вычислительных задач или подфункций посредством программного обеспечения, такого как выполняемое с помощью ЦПУ, и/или ГПУ, и/или КПУ, и/или выполнения других вычислительных задач или подпрограмм посредством прошивки, например посредством аппаратного обеспечения связанной интегральной схемы, такой как FPGA, ASIC и/или структурированная ASIC. Этого можно достичь различными путями, например посредством прямого слабого или жесткого связывания между ЦПУ/ГПУ/КПУ и микросхемой, например FPGA. Такие конфигурации могут быть особенно полезны при распределении операций, относящихся к обработке больших структур данных, как описано в настоящем документе, которые используют трудоемкие функции или подфункции, предназначенные для использования и доступа, как ЦПУ, и/или ГПУ и/или КПУ, так и интегральной схемой. В частности, в различных вариантах реализации при обработке данных посредством геномного конвейера, как описано в настоящем документе, например, для ускорения общей функции обработки, синхронизации и эффективности, на данных могут выполняться ряд различных операций, причем эти операции могут вовлекать как программные, так и аппаратные компоненты обработки.

[00564] Следовательно, может потребоваться совместное использование данных или иной обмен ими между программным компонентом, выполняющимися на ЦПУ, и/или ГПУ, и/или КПУ и аппаратным компонентом, встроенным в микросхему, например FPGA или ASIC. Соответственно, один или более из различных этапов в конвейере обработки или его части, могут быть выполнены одним устройством, например, ЦПУ/ГПУ/КПУ, а один или более из различных этапов могут быть выполнены другим устройством, например FPGA или ASIC. В таком случае ЦПУ и FPGA должны быть соединены с возможностью обмена данными, например, с помощью двухточечного межсоединения, таким образом, чтобы обеспечивать возможность эффективной передачи

таких данных, причем сопряжение может включать совместное использование ресурсов памяти. Чтобы добиться такого распределения задач и совместного использования информации для выполнения таких задач, ЦПУ, и/или ГПУ, и/или КПУ могут быть слабо или жестко связаны друг с другом и/или FPGA или другим набором микросхем, и может быть включена система управления рабочими потоками для эффективного распределения рабочей нагрузки.

[00565] Поэтому в конкретных вариантах реализации предложена платформа геномного анализа. Например, платформа может включать в себя материнскую плату, память, множество интегральных схем, например формирующих один или более из ЦПУ/ГПУ/КПУ, модуль картирования, модуль выравнивания, модуль сортировки и/или модуль определения вариантов. А именно, в конкретных вариантах реализации платформа может включать в себя первую интегральную схему, такую как интегральная схема, формирующая центральное процессорное устройство (ЦПУ) или графическое процессорное устройство (ГПУ), которое реагирует на один или более программных алгоритмов, которые выполнены с возможностью подачи ЦПУ/ГПУ инструкции на выполнение одного или более наборов функций геномного анализа, как описано в настоящем документе, например, где ЦПУ/ГПУ включает в себя первый набор физических электронных межсоединений для соединения с материнской платой. В других вариантах реализации предложено квантовое процессорное устройство, причем КПУ содержит одну или более квантовых схем, которые выполнены с возможностью осуществления одной или более функций, описанных в настоящем документе. В различных случаях предусмотрена память, которая может быть также присоединена к материнской плате и может быть также электронно соединена с ЦПУ и/или ГПУ, и/или КПУ, например посредством по меньшей мере части первого набора физических электронных межсоединений. В таких случаях память может быть выполнена с возможностью хранения множества ридов геномных данных, и/или по меньшей мере одной или более генетических референсных последовательностей, и/или индекса, например, в хэш-таблице, одной или более генетических референсных последовательностей.

[00566] Кроме того, платформа может включать в себя одну или более вторых интегральных схем, например, где каждая из вторых интегральных схем формирует программируемую пользователем вентильную матрицу (FPGA), или ASIC, или структурированную ASIC, имеющую второй набор физических электронных межсоединений для соединения с ЦПУ и памятью, например посредством протокола двухточечного соединения. В таком случае FPGA (или структурированная ASIC) может быть выполнена с возможностью программирования с помощью прошивки для конфигурирования набора жестко смонтированных цифровых логических схем, которые взаимно соединены множеством физических межсоединений для выполнения второго набора функций геномного анализа, например, картирования, выравнивания, сортировки, удаления дубликатов, определения вариантов и т.д., функции HMM и т.д. В частности, жестко смонтированные цифровые логические схемы FPGA могут быть выполнены в виде набора движков обработки для осуществления одного или более предварительно сконфигурированных этапов в конвейере анализа последовательностей платформы геномного анализа, например, где наборы движков обработки включают в себя один или более из модулей картирования, и/или выравнивания, и/или сортировки, и/или удаления дубликатов, и/или определения вариантов, причем модули могут быть сформированы из отдельных или одних и тех же подмножеств движков обработки.

[00567] Например, что касается определения вариантов, вычисление парной-HMM

или PD-HMM является одним из самых требующих ресурсоемких вычислений этапов протокола определения вариантов на основе гаплотипов. Следовательно, скорость определения вариантов может быть сильно улучшена путем ускорения данного этапа в одном или более движков FPGA или движков квантовой обработки. Однако можно получить дополнительную выгоду путем ускорения других требующих ресурсоемких вычислений этапов в дополнительных движках FPGA и/или КП QP, чтобы достичь более сильного ускорения определения вариантов или его части, или сокращения нагрузки на ЦПУ/ГПУ и количества необходимых ядер СПУ/ГПУ, или и того, и другого, как показано на ФИГ. 38.

[00568] В число дополнительных требующих ресурсоемких вычислений функций, имеющих отношение к определению вариантов, которые могут быть реализованы в движках FPGA и/или квантовой обработки, входят: обнаружение области, пригодной для определения, где выбирают для обработки области референсного генома, покрываемые выровненными ридами достаточной глубины и/или качества; обнаружение активной области, где выявляют локусы референсного генома с нетривиальными подтверждающими данными возможных вариантов и окна с достаточным контекстом вокруг этих локусов выбирают в качестве активных областей для дальнейшего анализа; построение графа де Брейна или другого графа сборки, где риды, перекрывающие активную область и/или K-меры из этих ридов собирают в граф; подготовка графа сборки, например, обрезание путей с низким покрытием или низким качеством, восстановление висящих начальных и конечных участков путей путем соединения их на референсном остове в графе, преобразование представления графа из K-меров в последовательность, объединение подобных ветвей и иное упрощение графа; выделение гаплотипов-кандидатов из собранного графа; а также выравнивание гаплотипов-кандидатов на референсный гном, например, с помощью выравнивания Смита-Ватермана, например, для определения вариантов (ОНП и/или инделов) в референсе, представляемом гаплотипом, и синхронизация их нуклеотидных позиций с референсом.

[00569] Все эти функции могут быть реализованы в виде высокопроизводительных аппаратных движков в FPGA и/или с помощью одной или более квантовых схем квантовой вычислительной платформы. Однако вызов такого разнообразия аппаратных функций ускорения из множества точек интеграции в программном обеспечении определения вариантов может стать неэффективным на слабо связанной платформе ЦПУ/ГПУ/КПУ + FPGA, и поэтому, возможно, целесообразной будет жестко интегрированная платформа ЦПУ/ГПУ/КПУ + FPGA. Например, различные способы пошаговой обработки, такие как построение, приготовление графа де Брейна или другого графа сборки и выделение из него гаплотипов может сильно выиграть от жестко связанной платформы ЦПУ/ГПУ/КПУ + FPGA. Кроме того, графы сборки представляют собой большие и сложные структуры данных, и многократная передача их между ЦПУ и/или ГПУ и FPGA может стать ресурсоемкой и препятствовать значительному ускорению.

[00570] Следовательно, идеальная модель для такой обработки графа с использованием жестко интегрированной платформы ЦПУ/ГПУ/КПУ и FPGA представляет собой хранение таких графов в совместно используемой памяти с поддержанием когерентности кэша для попеременной обработки с помощью ЦПУ, и/или ГПУ, и/или КПУ и аппаратных функций FPGA. В таком случае программный поток, обрабатывающий данных граф, может итеративно подавать команды на выполнение различных требующих ресурсоемких вычислений этапов обработки графа аппаратным движком, а затем программное обеспечение может проверять результаты и определять

следующие этапы между вызовами аппаратного обеспечения, например, как в приведенном в качестве примера процессе, изображенном на ФИГ. 39. Управление этой моделью обработки может осуществляться соответствующим образом сконфигурированной системой управления рабочими потоками и/или может быть выполнена с возможностью соответствия парадигмам программирования, таким как API структуры данных или объектно-ориентированный интерфейс способа, но с ускорением требующих ресурсоемких вычислений функций с помощью заказных аппаратных движков и/или движков квантовой обработки, которые на практике осуществляются за счет реализации на жестко связанной платформе ЦПУ, и/или ГПУ, и/или КПУ + FPGA с совместно используемой памятью, поддерживающей когерентность кэша и широкополосными межсоединениями ЦПУ/ГПУ/КПУ/FPGA с малой задержкой.

[00571] Соответственно, в дополнение к картированию и выравниванию ридов на референсный геном рида могут быть «de novo» собраны, например, без референсного генома, например, путем обнаружения явных перекрытий между ридами, например в скоплении, где они полностью или в основном согласуются, и объединения их в более длинные последовательности, контиги, каркасы или графы. Эту сборку можно также выполнять локально, например, с помощью всех ридов, для которых определено, что они картируются на данную хромосому или ее часть. При сборке таким образом возможно включение референсного генома или его сегмент в собираемую структуру.

[00572] В таком случае ввиду сложности соединения вместе последовательностей рида, которые неполностью согласуются, можно использовать структуру графа, например, когда перекрывающиеся риды могут согласовываться на одной последовательности в одном сегменте, но разветвляться на множество последовательностей в примыкающем сегменте, как объяснено выше. Поэтому такой граф сборки может быть графом последовательности, где каждые ребро или узел представляют один нуклеотид или последовательность нуклеотидов, которые, как считается, примыкают без зазора к последовательностям в соединенных ребрах или узлах. В конкретных случаях такой граф сборки может быть графом k-меров, где каждый узел представляет k-мер или нуклеотидную последовательность (как правило) фиксированной длины k, и где считается, что соединенные узлы перекрывают друг друга в более длинных наблюдаемых последовательностях, обычно перекрывающихся k - 1 нуклеотидами. В различных способах возможны одно или более преобразований между одним или более графами последовательности и графами k-меров.

[00573] Хотя графы сборки используют при определении вариантов на основе гаплотипов и некоторые используемые способы обработки графов похожи, существуют важные отличия. Графы de novo сборки обычно намного больше и используют более длинные k-меры. Тогда как графы сборки для определения вариантов ограничены довольно структурированными и относительно простыми графами, например, не имеющими циклов и проходящих от источника до стока вдоль остова референсной последовательности, графы de novo сборки обычно менее структурированные и более сложные, с циклами, висящими путями и другими аномалиями, не только разрешенными, но и подвергаемыми специальному анализу. Иногда используют окрашивание графа de novo сборки, назначая узлам и ребрам «цвета», означающие, например, из какого биологического образца они взяты, или совпадающую референсную последовательность. Следовательно, для графов de novo сборки требуется использовать более широкий выбор функций анализа и обработки графов, часто итеративно или рекурсивно, и, в частности, ввиду размера и сложности графов de novo сборки функции обработки, как правило, чрезвычайно требовательные к ресурсоемким вычислениям.

[00574] Следовательно, как указано выше, идеальная модель для такой обработки графа на жестко интегрированной платформе ЦПУ/ГПУ/КПУ + FPGA представляет собой хранение таких графов в совместно используемой памяти с поддержанием когерентности кэша для попеременной обработки между ЦПУ, и/или ГПУ, и/или КПУ и аппаратными функциями FPGA. В таком случае программный поток, обрабатывающий данных граф, может итеративно подавать команды на выполнение различных требующих ресурсоемких вычислений этапов обработки графа аппаратным движком, а затем проверяет результаты, чтобы тем самым определить следующие этапы, которые должны быть выполнены аппаратным обеспечением, например, маркируя соответствующие вызовы аппаратного оборудования. Подобно тому, как было отмечено выше, данная модель обработки извлекает огромную пользу из реализации на жестко связанной платформе ЦПУ + FPGA с совместной используемой памятью с поддержанием когерентности кэша и широкополосным межсоединением ЦПУ/FPGA с малой задержкой.

[00575] Кроме того, как описано ниже в настоящем документе, третичный анализ включает в себя геномную обработку, которая может следовать за сборкой графа и/или определением вариантов, что в клинических применениях может включать в себя аннотирование вариантов, прогнозирование фенотипа, тестирование на заболевание и/или прогнозирование реакции на терапию, как описано в настоящем документе. Причины полезности выполнения третичного анализа на такой жестко интегрированной платформе ЦПУ/ГПУ/КПУ + FPGA состоят в том, что такая конфигурация платформы обеспечивает эффективное ускорение первичной и/или вторичной обработки, которые требуют весьма ресурсоемких вычислений, и идеально подходит для продолжения третичного анализа на той же платформе в целях удобства и сокращения оборотного времени и для сведения к минимуму передачи и копирования больших файлов геномных данных. Поэтому слабо или жестко интегрированная платформа ЦПУ/ГПУ/КПУ + FPGA является хорошим выбором, но жестко связанная платформа может иметь дополнительные преимущества, поскольку этапы и способы третичного анализа весьма различаются в зависимости от области применения, и в любом случае, когда требующие ресурсоемких вычислений этапы замедляют третичный анализ, можно оптимизированным образом реализовать ускорение этих этапов с помощью заказной FPGA.

[00576] Например, третичный анализ на жестко интегрированной платформе ЦПУ/ГПУ/КПУ и/или FPGA особенно выгоден ввиду возможности повторного анализа геномным данных итерационным образом с использованием ускорения ЦПУ/ГПУ/КПУ и/или FPGA вторичной обработки в ответ на частичные или промежуточные третичные результаты, что может позволить извлечь дополнительную выгоду из конфигурации с жесткой интеграцией. Например, после того, как третичный анализ обнаруживает возможный фенотип или болезнь, но с ограниченной достоверностью истинности или ложности этого обнаружения, можно повторно выполнить целенаправленный вторичный анализ, уделяя особое внимание конкретным ридам и областям референса, влияющим на эту находку, тем самым улучшая точность и достоверность соответствующих определений вариантов, и, в свою очередь, улучшая достоверность определения обнаружения. Кроме того, если третичный анализ определяет информацию о генотипах родословного или структурного варианта, можно повторить вторичный анализ с использованием другого или модифицированного генома, который более подходит для конкретного индивида, тем самым повышая точность определений вариантов и улучшая точность дальнейших этапов третичного анализа.

[00577] Однако, если третичный анализ выполняется на только на платформе ЦПУ

после первичной и вторичной обработки (возможно, ускоренных на отдельной платформе), повторный анализ с помощью средств вторичного анализа, вероятно, будет слишком медленным, чтобы быть полезным на платформе третичного анализа самой по себе, и альтернатива заключается в передаче на более быструю платформу, которая тоже непозволительно медленная. Таким образом, в отсутствии аппаратного или квантового ускорения в какой-либо форме на платформе третичного анализа, первичную и вторичную обработку необходимо, как правило, завершать до начала третичного анализа без возможности простого повторного анализа или итеративного вторичного анализа и/или конвейерной организации аналитических функций. Но на платформе, ускоренный с помощью FPGA и/или квантовой обработки и особенно на платформе ЦПУ, и/или ГПУ, и/или КПУ, и/или FPGA, где вторичная обработка максимально эффективна, итеративный анализ становится практичным и полезным.

[00578] Соответственно, как указано выше, описанные в настоящем документе модули могут быть реализованы в аппаратном обеспечении микросхемы, например, могут быть жестко вмонтированы в нее, и в таких случаях их реализация может быть такова, что они смогут функционировать с более высокой скоростью и с большей точностью по сравнению с реализацией в программном обеспечении, например, когда имеются минимальные инструкции, которые нужно извлекать, считывать и/или исполнять. Кроме того, в различных случаях функции, подлежащие выполнению одним или более из этих модулей, могут быть распределены так, чтобы различные функции могли быть выполнены с возможностью реализации программным обеспечением главного ЦПУ, и/или ГПУ, и/или КПУ, тогда как в других случаях различные другие функции могли выполняться аппаратным обеспечением связанной FPGA, например, когда два или более устройств выполняются свои соответствующие функции друг с другом, например слаженным образом. Для таких целей можно жестко связать ЦПУ, ГПУ, КПУ и/или FPGA, или ASIC, или структурированную ASIC, например, посредством широкополосного соединения с малой задержкой, такого как QPI, CCVI, CAPI и т.п. Соответственно, в некоторых случаях функции с высокой вычислительной интенсивностью, подлежащие выполнению одним или более из этих модулей, могут быть выполнены квантовым процессором, реализованным одной или более квантовыми схемами.

[00579] Следовательно, при условии уникальной реализации аппаратной и/или квантовой обработки модули по настоящему изобретению могут функционировать непосредственно в соответствии со своими рабочими параметрами, например, без необходимости выборки, считывания и/или исполнения инструкций, как когда они реализованы исключительно в программном обеспечении ЦПУ. Кроме того, можно также снизить требования к памяти и времени обработки, например, когда обмены данными внутри микросхемы осуществляются посредством файлов, например, хранящихся локально в кэше FPGA/ЦПУ/ГПУ/КПУ, например, с поддержанием когерентности кэша, а не посредством широкомасштабного доступа к внешней памяти. Конечно, в некоторых случаях микросхема и/или плата может быть сделана такого размера, чтобы включать в себя больше памяти, например, столько, сколько на плате памяти, чтобы усилить возможности параллельной обработки, что приведет к еще более высоким скоростям обработки. Например, в определенных вариантах реализации микросхема по настоящему изобретению может содержать встроенное устройство DRAM, чтобы микросхеме не приходилось опираться на внешнюю память, что, таким образом, приведет к дальнейшему увеличению скорости обработки, например, когда можно использовать алгоритм Барроуза-Уилера или граф де Брейна вместо хэш-таблицы

и хэш-функции, которые могут в различных случаях опираться на внешнюю, например главную память. В таких случаях можно достичь выполнения части или всего конвейера за 6, или 10, или 12, или 15, или 20 минут или меньше, например от начала до конца.

[00580] Как указано выше, существуют всевозможные разные места, в которые может быть помещен модуль в аппаратном обеспечении, или может находиться на удалении от него, например на сервере, доступном на облаке. Когда данный модуль помещен на микросхеме, например, жестко вмонтирован в микросхему, его функция может выполняться аппаратным обеспечением, однако при необходимости модуль может быть помещен удаленно от микросхемы, и тогда платформа может содержать необходимые средства для отправки соответствующих данных в удаленное место, такое как сервер, например, квантовый сервер, доступный посредством облака, чтобы определенные функциональные возможности модуля могли быть задействованы для дальнейшей обработки данных в соответствии с выбираемыми пользователем требуемыми протоколами. Соответственно, часть платформы может содержать веб-интерфейс для выполнения одной или более задач в соответствии с функционированием одного или более модулей, описанных в настоящем документе. Например, когда картирование, выравнивание и/или сортировка - это все модули, которые могут иметь место на микросхеме, в различных случаях одно или более из локального повторного выравнивания, маркировки дубликатов, перекалибровки оценки качества оснований и/или поиска вариантов может происходить на облаке.

[00581] В частности, после того, как генетические данные сформированы и/или обработаны, например, в одном или более протоколах первичной и/или вторичной обработки, например, картированы, выровнены и/или отсортированы, например, для создания одного или более файлов определения вариантов, например, для определения того, как данные генетической последовательности субъекта отличаются от одной или более референсных последовательностей, в соответствии с дальнейшим аспектом настоящее изобретение может относиться к выполнению одной или более других аналитических функций на сформированных и/или обработанных генетических данных, например, для дальнейшей обработки, такой как третичная обработка, как показано на ФИГ. 40. Например, система может быть выполнена с возможностью дальнейшей обработки сформированных и/или подвергнутых вторичной обработке данных, например, путем пропускания их через один или более конвейеров 700 третичной обработки, таких как один или более из конвейера микроматричного анализа, конвейера анализа генома, например, полногеномного анализа, конвейера анализа генотипирования, конвейера анализа экзома, конвейера анализа микробиома, конвейера анализа генотипирования, включая совместное генотипирование, конвейера анализа вариантов, включая конвейеры структурных вариантов, конвейеры соматических вариантов, и конвейеры GATK и/или MuTect2, а также конвейеры секвенирования РНК и конвейеры других генетических анализов.

[00582] Кроме того, в различных случаях может быть предусмотрен дополнительный уровень обработки 800, например, для диагностики болезней, терапевтического воздействия и/или профилактического предупреждения, например, включая НИПТ, ОРИТН, рак, LDT, аграрно-биологические и другие такие данные диагностики болезней, профилактики и/или терапий, используя эти данные сформированные одним или более представленными первичными, и/или вторичными, и/или третичными конвейерами. Например, в число конкретных биоаналитических конвейеров входят конвейеры генома, конвейеры эпигенома, конвейера метагенома, конвейеры вариантов, например GATK/MuTect2, и другие такие конвейеры. Следовательно, устройства и способы, описанные

в настоящем документе, могут быть использованы для формирования данных генетических последовательностей, которые затем могут быть использованы для формирования одного или более файлов определения вариантов и/или другой связанной информации, которая может быть в дальнейшем подвергнута обработке другими конвейерами третичной обработки в соответствии с устройствами и способами, описанными в настоящем документе, например, для диагностики конкретных и/или общих заболеваний, а также для профилактических и/или терапевтических мер и/или методов воздействия на развитие. См., например, ФИГ. 41 B, C и 43.

[00583] Как описано выше, способы и/или системы, представленные в настоящем документе, могут включать в себя формирование и/или получение иным образом данных генетической последовательности. Такие данные могут быть сформированы или иным образом получены из любого подходящего источника, например с помощью СНП или «секвенатора, основанного на технологии микросхем». Способы и системы, описанные в настоящем документе, могут включать в себя выполнение на этих сформированных и/или полученных данных дальнейшей обработки, например, с использованием одного или более протоколов вторичной 600 обработки. Протоколы вторичной обработки могут включать в себя одно или более из картирования, выравнивания и сортировки сформированных данных генетической последовательности, например, для создания одного или более файлов определения вариантов, например, для определения того, как данные генетической последовательности субъекта отличаются от одной или более референсных последовательностей или геномов. Согласно другому аспекту изобретение может относиться к выполнению одной или более аналитических функций на сформированных и/или обработанных генетических данных, например, результирующих данных вторичной обработки, например, для дополнительной обработки, например, третичной обработки 700/800, которая может быть выполнена на или вместе с той же микросхемой или набором микросхем, на которой обеспечена вышеупомянутая технология секвенатора.

[00584] Соответственно, в первом случае, например, что касается формирования, получения и/или передачи данных генетической последовательности, как показано на ФИГ. 37-41, такие данные могут быть созданы либо локально, либо удаленно, и/или их результаты могут быть затем обработаны непосредственно, например, локальным вычислительным ресурсом 100, или могут быть переданы в удаленное место, например, на удаленный вычислительный ресурс 300, для дальнейшей обработки, например, для вторичной и/или третичной обработки, см. ФИГ. 42. Например, сформированные данные генетической последовательности могут быть обработаны локально и непосредственно, например, когда функциональные возможности секвенирования и вторичной обработки находятся на одном и том же наборе микросхем и/или в пределах одного и того же устройства в месте эксплуатации 10. Аналогичным образом сформированные данные генетической последовательности могут быть обработаны локально и опосредованно, например, когда функциональные возможности секвенирования и вторичной обработки выполняются по отдельности разными устройствами, которые совместно используют одно и то же оборудование или место, но могут быть разнесены в пространстве, хотя и соединены с возможностью обмена данными, например, по локальной сети 10. В следующем случае данные генетической последовательности могут быть произведены дистанционно, например, с помощью удаленного СНП, и полученные в результате данные могут быть переданы по облачной сети 30/50 в автономное удаленное место 300, например, отделенное географически от секвенатора.

[00585] А именно, как показано на ФИГ. 40А, в различных вариантах реализации устройство формирования данных, например, нуклеотидный секвенатор 110, может быть предусмотрен в месте эксплуатации, например, когда секвенатор представляет собой «секвенатор на микросхеме» или СНП, причем секвенатор связан с локальным вычислительным ресурсом 100 либо напрямую, либо опосредованно, например, с помощью соединения 10/30 локальной сети. Локальный вычислительный ресурс 100 может включать в себя или быть иным образом связаны с одним или более механизмами 110 формирования данных и/или механизмами 120 получения данных. Такие механизмы могут быть механизмами, выполненными с возможностью формирования и/или получения иным образом данных, таких как аналоговые, цифровые и/или электромагнитные данные, относящиеся к одной или более генетическим последовательностям субъекта или группы субъектов, например, когда данные генетической последовательности представлены в формате файла BCL или FASTQ.

[00586] Например, такой механизм 110 формирования данных может быть первичным процессором, таким как секвенатор, например, СНП, секвенатора на микросхеме или другой подобный механизм для формирования информации о генетической последовательности. Кроме того, такие механизмы 120 получения данных могут быть любым механизмом, выполненным с возможностью приема данных, например, сформированной информацией о генетической последовательности; и/или совместно с генератором 110 данных и/или вычислительным ресурсом 100 применения к этой информации одного или более протоколов вторичной обработки, например, конвейерных устройств вторичной обработки, выполненных с возможностью выполнения протоколов сопоставителя, выравнивателя, сортировщика и/или определителя вариантов на сформированных и/или полученных данных последовательности, как описано в настоящем документе. В различных случаях устройство 110 формирования данных и/или устройство 120 получения данных могут быть связаны сетью друг с другом, например, локальной сетью 10, такой как для локального хранилища 200; или могут быть связаны сетью, например, локальной и/или облачной сетью 30, такой как для передачи и/или приема данных, таких как цифровые данные, относящиеся к первичной и/или вторичной обработке информации о генетической последовательности, например, на удаленное место или с него, например, для удаленной обработки 300 и/или хранения 400. В различных вариантах реализации один или более их этих компонентов могут быть соединены с возможностью обмена данными с помощью гибридной сети, как описано в настоящем документе.

[00587] Локальный вычислительный ресурс 100 может также включать в себя или быть иным образом связан с компилятором 130 и/или процессором 140, таким как компилятор 130, выполненный с возможностью компиляции сформированных и/или полученных данных и/или связанных с ними данных, и процессор 140, выполненный с возможностью обработки сформированных и/или полученных данных, и/или скомпилированных данных, и/или управляющих данных системы 1 и ее компонентов, как описано в настоящем документе, например, для выполнения первичной, вторичной и/или третичной обработки. Например, может быть использован любой подходящий компилятор, однако в определенных случаях можно достичь дополнительной эффективности не только за счет реализации жестко связанной конфигурации, такой как рассмотрена выше, для эффективной и когерентной передачи данных между компонентами системы, но ее можно также достичь путем реализации конфигурации «точно в срок» (JIT) компилятора компьютерного языка. Кроме того, в определенных случаях процессор 140 может включать в себя систему управления рабочими потоками

для управления функционированием различных компонентов системы применительно к сформированным, принятым и/или подлежащим обработке данным посредством различных ступеней конвейеров платформы.

[00588] А именно, используемый в настоящем документе термин «точно в срок» (JIT) относится к устройству, системе и/или способу преобразования полученных и/или сформированных файлов из одного формата в другой. В структуре широкого использования система JIT, описанная в настоящем документе, может включать в себя компилятор 130 или иную вычислительную архитектуру, например, программу обработки, которая может быть реализована таким образом, чтобы преобразовывать различные коды из одной формы в другую. Например, в одной реализации компилятор JIT может быть выполнен с возможностью преобразования байткода, или другого программного кода, содержащего инструкции, который необходимо интерпретировать в инструкции, пригодные для отправки непосредственно в связанный процессор 140 для почти немедленного исполнения, например, без необходимости интерпретации инструкций с помощью конкретного машинного языка. В частности, после того, как программа кодирования, например, программы на языке Java, написана, операторы на исходном языке могут быть компилированы компилятором, например, компилятором с языка Java, в байткод вместо того, чтобы компилировать в код, который содержит инструкции, соответствующие любому данному конкретному языку обработки аппаратной платформы. Поэтому этот байткод, компилирующий действие, является независимым от платформ кодом, который может быть отправлен на любую платформу и выполнен на этой платформе независимо от процессора, лежащего в ее основе. Следовательно, подходящий компилятор может быть компилятором, который выполнен с возможностью компиляции байткода в специфичный для платформы исполнимый код, который затем может быть исполнен немедленно. В этом случае компилятор JIT может быть выполнен с возможностью немедленного преобразования одного формата файла в другой, например «на лету».

[00589] Следовательно, соответствующим образом сконфигурированный компилятор, как описано в настоящем документе, в состоянии преодолевать различные недостатки в данной области техники. А именно, после компиляции программ, которые были написаны на определенном языке, должны быть перекомпилированы и/или переписаны в зависимости от каждой конкретной компьютерной платформы, на которой они должны были быть реализованы. В современных системах компиляции компилятор может быть выполнен с возможностью записи и компиляции программы только один раз, и после записи в конкретной форме она может быть преобразована в одну или более других форм почти немедленно. Точнее говоря, компилятор 130 может представлять собой JIT или другой подобный формат компилятора динамической трансляции, который выполнен с возможностью написания инструкций на независимом от платформы языке, который не нужно перекомпилировать и/или переписывать в зависимости от конкретной компьютерной платформы, на которой он реализован. Например, в конкретной модели использования компилятор может быть выполнен с возможностью интерпретации скомпилированного байткода и/или других кодированных инструкций в инструкции, которые понятны данному конкретному процессору, по преобразованию одного формата файла в другой независимо от вычислительной платформы. По существу система JIT, описанная в настоящем документе, выполнена с возможностью приема одного генетического файла, например, представляющего генетический код, такого как файл BCL или FASTQ, сформированный генетическим секвенатором, и быстрого преобразования в другую форму, например, в файл SAM,

BAM и/или CRAM file, например, с помощью способов, описанных в настоящем документе.

[00590] В частности, в различных случаях система, описанная в настоящем документе, может включать в себя первый и/или второй компилятор 130a и 130b, такой как виртуальная компиляционная машина, которая выполняет преобразование одной или множества байткодовых инструкций за один раз. Например, использование компилятора «точно в срок» типа Java или другой соответствующим образом сконфигурированный второй компилятор в пределах представленной платформы системы позволит компилировать инструкции в байткод, который может быть затем преобразован в конкретный системный код, например, как будто программа была первоначально скомпилирована на данной платформе. Соответственно, после того, как код компилирован и/или перекомпилирован, например, с помощью компилятора (-ов) 130 JIT, он будет быстрее выполняться в процессоре 140 компьютера. Поэтому в различных вариантах реализации компиляция «точно в срок» (JIT) может быть выполнена с возможностью осуществления во время исполнения данной программы, например, во время выполнения, а не до исполнения. В таком случае сюда могут входить этапы трансляции в машинный код или трансляции в другой формат, который может быть затем исполнен непосредственно, тем самым обеспечивая возможность одного или более из компиляции перед исполнением (AOT) и/или интерпретации.

[00591] Более конкретно, в соответствии с реализацией в настоящей системе типичный поток данных обычно создает данные в одном или более форматах, производных от одной или более вычислительных платформ, таких как файловые форматы BCL, FASTQ, SAM, BAM, CRAM и/или VCF или их эквиваленты. Например, типичный секвенатор 110 ДНК, например, СНП, создает необработанные сигналы, представляющие определенные основания, которые называются в данном документе ридами, например, в виде файла BCL и/или FASTQ, который может быть, необязательно, подвергнут дальнейшей обработке, например, улучшенной обработке изображения, и/или сжат 150. Аналогичным образом риды сформированных файлов BCL/FASTQ могут быть затем подвергнуты дальнейшей обработке в системе, как описано в настоящем документе, для создания картированных и/или выровненных данных, и эти полученные данные, например, картированные и выровненные риды, могут быть в формате файла SAM или BAM, или, в качестве альтернативы, в формате файла CRAM. Далее, файл SAM или BAM может быть затем обработан, например, с помощью процедуры определения вариантов, для получения файла определения вариантов, такого как файл VCF или файл gVCF. Соответственно, после создания все эти полученные файлы BCL, FASTQ, SAM, BAM, CRAM и/или VCF являются (чрезвычайно) большими файлами, которые нужно сохранить, например, в архитектуре системной памяти, локально 200 или дистанционно 400. Хранение любого из одного этих файлов обходится дорого. Хранение всех файлов этих форматах обходится крайне дорого.

[00592] Как было указано, компиляция «точно в срок» (JIT) или другая двойная компиляция или другой анализ компиляции динамической трансляции могут быть выполнены и развернуты в данной системе с возможностью сокращения столь высоких расходов на хранение. Например, схема анализа JIT может быть реализована в данной системе таким образом, чтобы данные хранились только в одном формате (например, в формате сжатого файла FASTQ или BAM и т.д.), с одновременным обеспечением доступа к одному или более файловых форматов (например, BCL, FASTQ, SAM, BAM, CRAM и/или VCF и т.д.). Этот быстрый процесс преобразования файла может быть совершен с помощью быстрой обработки геномным данных с использованием

описанных в настоящем документе соответствующих аппаратных и/или квантовых платформ ускорения, например, таких как для картирования, выравнивания, сортировки и/или определения вариантов (или составляющих их функций, таких как удаление дубликатов, HMM и алгоритм Смита-Ватермана, сжатие и распаковка и т.п.) в аппаратных движках на интегральных схемах, таких как FPGA, или с помощью квантового процессора. Следовательно, благодаря реализации JIT или подобного анализа наряду с таким ускорением геномные данные могут обрабатываться с формированием требуемых форматов файлов на лету при скоростях, сравнимых с обычным доступом к файлам. Таким образом, значительная экономия на хранении может быть реализована за счет обработки типа JIT с небольшой потерей в скорости доступа или вообще без потери.

[00593] В частности, существуют два варианта, которые полезны для базового хранилища геномных данных, создаваемого, как описано в настоящем документе, таким образом, чтобы быть доступным для обработки типа JIT, а именно: хранение невыровненных ридов (например, сюда можно отнести сжатый файл FASTQ или невыровненные сжатые файлы SAM, BAM или CRAM), и хранение выровненных ридов (например, сюда можно отнести сжатые файлы BAM или CRAM). Однако, так как ускоренная обработка, описанная в настоящем документе, позволяет быстро получать любой из упомянутых файловых форматов, например, на лету, базовый формат файла для хранения может быть выбран таким образом, чтобы достигать наименьшего размера сжатого файла, тем самым снижая стоимость хранения. Следовательно, если учитывать сравнительно меньший размер файла для необработанных, например, исходных невыровненных, данных ридов, то в хранении невыровненных ридов так, чтобы поля данных были сведены к минимуму, есть преимущество. Аналогичным образом существует преимущество в хранении обработанных и сжатых данных, например в формате файла CRAM.

[00594] Более конкретно, учитывая большие скорости обработки, которые могут быть достигнуты с помощью устройств систем и способов их использования, описанных в настоящем документе, во многих случаях может не потребоваться хранить картированные и/или выровненные данные для всех до одного рида, так как эту информацию можно легко получить по мере надобности, например, на лету. Кроме того, хотя для хранения данных генетической последовательности обычно используют сжатый формат файла FASTQ (например, FASTQ.gz), такие данные невыровненных ридов могут также храниться в более совершенных сжатых форматах, например, после картирования и/или выравнивания в файлах SAM, BAM или CRAM, которые могут еще больше уменьшить размер файла, например, за счет использования компактного двоичного представления и/или более целенаправленных способов сжатия. Следовательно, эти файловые форматы могут быть сжаты перед хранением, распакованы после хранения и быстро обработаны, например, на лету, для преобразования одного файлового формата в другой.

[00595] Преимущество хранения выровненных ридов состоит в том, что содержимое последовательности большей части или всех до одного ридов может быть опущено. А именно, систему можно эффективно улучшить, а пространство для хранения сэкономить всего лишь за счет хранения разницы между последовательностями ридов и выбранным референсным геномом, например, которая указана позициями выравнивания варианта рида. Точнее говоря, поскольку отличия от референса обычно нечастые, выровненная позиция и список отличий часто могут храниться более компактно, чем исходная последовательность рида. Поэтому в различных случаях хранение в формате

выровненного рида, например, при хранении данных, относящихся к отличиям выровненных ридов, может быть предпочтительнее хранения данных невыровненного рида. В таком случае, если формат выровненного рида и/или определения вариантов используется в качестве базового формата хранения, например, в процедуре JIT, другие форматы, такие как SAM, BAM и/или CRAM, сжатых форматов файлов, тоже могут использоваться.

[00596] Наряду с данными файла выровненного и/или невыровненного рида, которые нужно хранить, можно также хранить самые разные другие данные, такие как метаданные, полученные из различных вычислений, определенных в настоящем документе. В число таких вычисленных данных могут входить картированные, выровненные и/или подвергнутые дальнейшей обработке данные, такие как, оценки выравнивания, достоверность картирования, редакционное расстояние от референса и т.д. В определенных случаях такие метаданные и/или другую дополнительную информацию не требуется держать в базовом хранилище для анализа JIT, например, как в тех случаях, когда они могут быть воспроизведены на лету, например, с помощью ускоренной обработки данных, описанной в настоящем документе.

[00597] Что касается метаданных, то эти данные могут представлять собой небольшой файл, который предписывает системе порядок выполнения перехода назад или вперед от одного формата файла к преобразованию в другой формат файла. Следовательно, файл метаданных позволят системе создавать двоично совместимую версию любого другого типа файла. Например, при продвижении вперед от исходного файла данных системе нужно только получать доступ к инструкциям в метаданных и реализовать их. Наряду с быстрым преобразованием формата файла JIT также обеспечивает быстрые сжатие и/или распаковку и/или сохранение, например, в кэше облачной памяти генома.

[00598] Как более подробно обсуждено ниже, после того, как данные последовательности сформированы 110, они могут быть сохранены локально 200 и/или сделаны доступными для дистанционного хранения, например, в доступной из облака кэш-памяти 400 типа облачного хранилища «Dropbox». Например, оказавшись в геномном хранилище Dropbox, данные могут представляться как доступные на облаке 50, и поэтому могут повергнуты дальнейшей обработке, например, по существу немедленно. Это особенно полезно при наличии множества систем 100/300 картирования/выравнивания/сортировки/определения вариантов, например, на одной из двух сторон интерфейса облака 50, способствующего автоматической загрузке и обработке данных, которые могут быть подвергнуты дальнейшей обработке, например, с помощью технологии JIT, описанной в настоящем документе.

[00599] Например, базовый формат хранения для компиляции JIT и/или обработки может содержать минимальные поля данных, такие как имя рида, оценки качества оснований, позиция выравнивания и/или ориентация в референсе и список отличий от референса, например, когда каждое поле может быть сжато оптимальным образом для его типа данных. Различные другие метаданные могут быть включены или иным образом связаны с файлом хранения. В таком случае базовое хранилище для анализа JIT может находиться в локальной файловой системе 200, например, на накопителях на жестких дисках и твердотельных накопителях, или в сетевом ресурсе хранения, таком как система 400 хранения типа хранилища объектов NAS или хранилища Dropbox. В частности, при хранении файлов различных форматов, таких как BCL, FASTQ, SAM, BAM, CRAM, VCF и т.д., созданных для геномного набора данных, который может быть подвергнут обработке JIT и/или сохранен, JIT или другая подобная система компиляции и/или анализа может быть выполнена с возможностью преобразования

данных в один базовый формат хранения для хранения. С файлом могут быть связаны и сохранены дополнительные данные, такие как метаданные и/или другие сведения (которые могут быть небольшого объема), необходимые для воспроизведения всех остальных требуемых форматов с помощью ускоренной обработки геномных данных. Такая дополнительная информация может включать в себя одно или более из следующего: список форматов, подлежащих воспроизведению, команды обработки данных для воспроизведения каждого формата, уникальный идентификатор (например, URL или хэш-функция MD5/SHA) референсного генома, настройки различных параметров, например, для картирования, выравнивания, сортировки, определения вариантов и/или любой другой обработки, которые описаны в настоящем документе, затравки рандомизации для этапов обработки, например, использование псевдорандомизации для детерминистического воспроизведения тех же самых результатов, пользовательский интерфейс и т.п.

[00600] В различных случаях данные, подлежащие сохранению и/или извлечению в JIT или аналогичной системы динамической обработки и/или анализа трансляции, могут быть представлены пользователю или другим приложениям разнообразными способами. Например, один вариант состоит в том, чтобы иметь хранилище анализа JIT в стандартном или настраиваемом формате файла «объект JIT», например, для хранения и/или извлечения в виде файла формата SAM, BAM, CRAM или другого файла настраиваемого формата, и обеспечения пользовательских средств для быстрого преобразования объекта JIT в требуемый формат (например, в локальном временном хранилище 200) с помощью ускоренной обработки, описанной в настоящем документе. Другой вариант заключается в представлении появления множества форматов файлов, таких как BCL, FASTQ, SAM, BAM, CRAM, VCF и т.д, пользователю и пользовательским приложениям таким образом, чтобы при доступе файловой системы к различным форматам файлов использовалась процедура JIT, и таким образом нужно было сохранять только один типа файла, а из этого файла можно было формировать все остальные файлы на лету. Еще одним вариантом является создание пользовательское средство, иначе принимающее специальные форматы файлов (BCL, FASTQ, SAM, BAM, CRAM, VCF, и т.д.), которые вместо этого могут быть представлены как объект JIT, и может автоматически вызывать анализ JIT, чтобы при вызове получать данные в требуемом формате данных, например, BCL, FASTQ, SAM, BAM, CRAM, VCF и т.д. автоматически.

[00601] Соответственно, процедуры JIT полезны для обеспечения доступа ко множеству форматов файлов, например, BCL, FASTQ, SAM, BAM, CRAM, VCF и т. п., из одного формата файла путем быстрой обработки базового хранящегося сжатого формата файла. Кроме того, технология JIT остается полезной, даже если доступ нужно получать только к одному формату файла, так как сжатие все равно достигается непосредственно в отношении хранения формата, к которому осуществляется доступ. В таком случае базовый формат хранения файла может отличаться от формата файла, к которому осуществляется доступ, и/или может содержать меньше метаданных, и/или может быть сжат более эффективно, чем формат, к которому осуществляется доступ. Кроме того, в таком случае, как обсуждалось выше, файл сжимают перед сохранением и распаковывают при извлечении, например автоматически.

[00602] В различных случаях способы анализа JIT, представленные в настоящем документе, могут быть также использованы для передачи геномных данных по Интернету или другой сети, чтобы сводить к минимуму время передачи и меньше потреблять полосу пропускания сети. В частности, в одном приложении хранения может

храниться один сжатый базовый формат файла и/или доступ к одному или более форматам может осуществляться посредством распаковки и/или ускоренной обработки геномных данных. Аналогичным образом в приложении передачи нужно передавать только один сжатый базовый формат файла, например, с исходного сетевого узла на сетевой узел назначения, например, когда базовый формат может быть выбран в первую очередь для получения наименьшего размера при сжатии, и/или когда все требуемые форматы файлов могут быть сформированы на узле назначения посредством или для обработки геномных данных, например на лету. Благодаря этому для хранения и/или передачи нужно будет использовать только один формат данных, из которого могут быть получены другие различные форматы файлов.

[00603] Соответственно, как показано на ФИГ. 40A, обработка геномных данных с аппаратным и/или квантовым ускорением, как описано в настоящем документе, может быть использована как на исходном узле сети для формирования и/или сжатия базового формата для передачи, так и на узле назначения для распаковки и/или формирования других требуемых форматов файлов с помощью ускоренной обработки геномных данных. Тем не менее, JIT или иной динамический анализ трансляции продолжает оставаться полезным в приложении передачи, даже если только один из исходного узла или узла назначения использует обработку геномных данных с аппаратным и/или квантовым ускорением. Например, сервер данных, который отправляет большие объемы геномных данных, может использовать обработку геномных данных с аппаратным и/или квантовым ускорением для формирования сжатого базового формата для передачи в различные места назначения. В таких случаях в каждом месте назначения могут использовать более медленную программную обработку геномных данных для формирования других требуемых форматов файлов. Следовательно, хотя скоростные преимущества анализа JIT ослабевают на узле назначения, время передачи и использование сети по-прежнему эффективно снижаются, и исходный узел в состоянии эффективно обслуживать множество таких передач за счет своего соответствующего устройства обработки геномных данных с аппаратным и/или квантовым ускорением.

[00604] Кроме того, в другом примере сервер данных, который принимает загрузки больших объемов геномных данных, например, из различных источников, может использовать обработку и/или сохранение геномных данных с аппаратным и/или квантовым ускорением, тогда как различные исходные узлы могут использовать более медленное программное обеспечение, исполняемое на ЦПУ/ГПУ, для формирования сжатого базового формата файла для передачи. В альтернативном варианте реализации обработка геномных данных с аппаратным и/или квантовым ускорением может быть использована одним или более промежуточными сетевыми узлами, такими как шлюзовый сервер, между исходным узлами и узлами назначения, для передачи и/или приема геномных данных в сжатом базовом формате в соответствии с JIT или другими способами динамического анализа трансляции, таким образом используя преимущества в виде уменьшения времени передачи и использования сети без перегрузки этих промежуточных сетевых узлов за счет чрезмерной программной обработки.

[00605] Следовательно, как показано на ФИГ. 40A, в определенных случаях локальный вычислительный ресурс 100 может содержать компилятор 130, такой как компилятор JIT, и может также содержать блок 150 уплотнителя, который выполнен с возможностью сжатия данных, например, формируемых и/или получаемых данных первичной и/или вторичной обработки (или третичные данные), которые могут быть сжаты, например, перед передачей по локальной сети 10, и/или облачной сети 30, и/или гибридной сети 50, например в процедуре анализа JIT, и которые могут быть

распакованы после передачи и/или перед использованием.

[00606] Как описано выше, в различных случаях система может содержать первую интегральную и/или квантовую схему 100, например, для выполнения операции картирования, выравнивания, сортировки и/или определения вариантов с целью формирования одних или более картированных, выровненных, сортированных, не содержащих дубликатов и/или подвергнутых определению вариантов данных результатов. Кроме того, система может содержать другую интегральную и/или квантовую схему 300, например, для использования данных результатов при выполнении одного или более анализов с помощью геномных и/или биоинформационных конвейеров, таких как третичная обработка. Например результирующие данные, сформированные первой интегральной и/или квантовой схемой 100, могут быть использованы, например, первой или второй интегральной и/или квантовой схемой 300 при выполнении процедуры дальнейшей конвейерной геномной и/или биоинформационной обработки. А именно, вторичная обработка геномных данных может быть выполнена первым процессором 100 с аппаратным и/или квантовым ускорением для создания данных результатов, а третичная обработка может быть выполнена на этих данных результатов, например, когда дальнейшую обработку выполняют с помощью ЦПУ, и/или ГПУ, и/или КПУ 300, которое оперативно соединено с первой интегральной схемой. В таком случае вторая схема 300 может быть выполнена с возможностью выполнения третичной обработки данных геномной вариации, полученных первой схемой 100. Соответственно, данные результатов, полученные из первой интегральной схемы, действуют в качестве движка анализа, приводящего в действие этапы дальнейшей обработки, описанные в настоящем документе в отношении третичной обработки, например, с помощью второй интегральной и/или квантовой схемы 300 обработки.

[00607] Однако данные, формируемые на каждом из этих этапов первичной, и/или вторичной, и/или третичной обработки, могут быть огромными, требующими очень больших затрат на ресурсы и/или память, например, для хранения, либо локально 200, либо удаленно 400. Например, на первом этапе первичной обработки сформированные данные 110 последовательности нуклеиновых кислот, например в формате файла BCL и/или FASTQ, могут быть приняты 120, например из СНП 110. Независимо от формата файла этих данных последовательности они могут быть использованы в протоколе вторичной обработки, как описано в настоящем документе. Возможность приема и обработки первичных данных последовательности непосредственно из СНП, например, в формате файла BCL и/или FASTQ, очень полезна. В частности, вместо преобразования файла данных последовательности из СНП, например, в формате файла BCL, в файл FASTQ, файл может быть принят непосредственно из СНП, например, как файл BCL, и может быть обработан, например, будучи принятым и обработанным системой JIT, например, на лету, в файл FASTQ и может быть затем обработан, как описано в настоящем документе, для создания картированных, выровненных, сортированных, не имеющих дубликатов и/или подвергнутых определению вариантов данных результатов, которые могут быть затем сжаты, например, в файл SAM, BAM и/или CRAM, и/или могут быть подвергнуты дальнейшей обработке, например, с использованием одного или более описанных конвейеров геномной третичной обработки.

[00608] Соответственно, после создания таких данных их нужно сохранить каким-либо образом. Однако такое хранение не только ресурсоемкое, но и дорогостоящее. А именно, в типичном геномном протоколе полученные секвенированные данные сохраняют в виде большого файла FASTQ. Затем, после обработки, например, путем

применения протокола картирования и/или выравнивания, создают файл BAM, который тоже, как правило, сохраняют, увеличивая стоимость хранения геномных данных, например, вследствие необходимости хранения обоих файлов, FASTQ и BAM. Кроме того, после обработки файла BAM, например, путем применения протокола определения вариантов, создают файл VCF, который тоже, как правило, нужно сохранить. В таком случае, чтобы адекватно обеспечить и использовать сформированные генетические данные, возможно, потребуется хранить все три файла, FASTQ, BAM и VCF, локально 200 или удаленно 400. Кроме того, исходный файл BCL тоже может храниться. Такое хранение неэффективно, а также требует интенсивного использования ресурсов памяти и стоит дорого.

[00609] Однако вычислительная мощность аппаратной и/или квантовой архитектуры обработки, реализованная в настоящем изобретении, наряду с компиляцией, сжатием и хранением JIT, сильно смягчают неэффективность, стоимость ресурсов и издержки. Например, с учетом реализуемых способов и скоростей обработки, достигаемых с помощью предложенных ускоренных интегральных схем, например, для преобразования файла BCL в файл FASTQ с последующим преобразованием файла FASTQ в файл SAM или файл BAM, а затем преобразования файла BAM в файл CRAM и/или файл VCF и обратно, предложенная система сильно сокращает количество вычислительных ресурсов и/или размеры файлов, необходимых для эффективной обработки и/или хранения таких данных. Преимущества этих систем и способов еще больше усиливаются за счет того, что нужно хранить только один формат файла, например, нужно хранить BCL, FASTQ, SAM, BAM, CRAM и/или VCF, из которого можно получить все другие форматы файлов и обработать. В частности, нужно сохранить только один формат файла, а из такого файла можно быстро, например, на лету, сформировать любой из других форматов файлов, в соответствии со способами, описанными в настоящем документе, например, в формате компиляции «точно в срок», или JIT.

[00610] Например, в соответствии с типичными способами известного уровня техники для обработки и хранения файлов FASTQ, формируемых секвенатором СНП, требуется большой объем вычислительных ресурсов, например, фермы серверов и большие банки памяти. В частности, в типичном случае после создания с помощью СНП большого файла FASTQ нужно будет использовать ферму серверов для приема и преобразования файла FASTQ в файл BAM и/или файл CRAM, причем обработка может занять до суток или более. Однако сам созданный файл BAM тоже нужно сохранить, что требует дополнительно времени и ресурсов. Аналогичным образом файл BAM или CRAM может быть обработан для формирования файла VCF, что также может занять еще сутки или более, и этот файл тоже нужно будет сохранить, тем самым затратив дополнительно ресурсы и понеся дальнейшие издержки. Более конкретно, в типичном случае на файл FASTQ для генома человека расходуется около 90 ГБ памяти на файл. Аналогичным образом типичный файл генома человека BAM может занимать около 160 ГБ. Возможно, потребуется также сохранить файл VCF, хотя такие файлы и довольно маленькие по сравнению с файлами FASTQ и/или BAM. В результате вторичной обработки могут быть также сформированы файлы SAM и CRAM, и их тоже может понадобиться сохранить.

[00611] До технологий, предложенных в настоящем документе, требовалось интенсивное использование вычислительных ресурсов для перехода с одного этапа на другой, например, с одного формата файла на другой, и поэтому, как правило, приходилось хранить все данные для этих форматов файлов. Частично это объясняется тем, что если пользователь захочет когда-либо вернуться и восстановить один или

более из этих файлов, потребуется огромное количество вычислительных ресурсов и времени, чтобы отмотать назад процессы, участвующие в восстановлении различных файлов, тем самым понеся высокие денежные расходы. Кроме того, эти файлы сжимают перед сохранением, а такое сжатие может занимать от около 2 до около 5 и около 10 часов или более, причем примерно такое же количество времени потребуется для распаковки перед повторным использованием. Вследствие этих высоких расходов типичные пользователи не будут сжимать такие файлы перед сохранением, а будут также, как правило, сохранять все два, три или более форматов файлов, например, BCL, FASTQ, BAM, VCF, неся повышенные издержки в течение более длительного периода времени.

[00612] Соответственно, протоколы JIT, применяемые в настоящем изобретении, могут использовать повышенные скорости обработки, достигаемые предложенными аппаратными и/или квантовыми ускорителями, для воплощения улучшенной эффективности при сокращении времени и затрат, как на обработку, так и на хранение. Вместо хранения 2, 3 или более экземпляров одних и тех же общих данных в различных форматах файлов, нужно хранить только один формат файла, а любой из других типов файлов может быть восстановлен на лету, например с помощью платформ ускоренной обработки, рассмотренных в настоящем документе. В частности, предложенные устройства и системы могут легко возвращаться от хранения файла FASTQ к файлу BCL, или продвигаться к файлу BAM и затем дальше к файлу VCF, например, менее чем за 30 минут, например в течение 20 минут или около 15 или 10 минут или меньше.

[00613] Следовательно, благодаря использованию конвейеров и скорости обработки, обеспечиваемой аппаратными/квантовыми движками обработки, описанными в настоящем документе, нужно хранить один формат файла, а остальные форматы файлов могут быть легко и быстро сформированы из него. Поэтому вместо необходимости хранения всех трех форматов файлов, нужно хранить один формат файла, из которого можно формировать остальные форматы файлов, например, на лету, точно в срок для дальнейших этапов обработки, требуемых для пользователя. Следовательно, система может быть выполнена с возможностью использования ее без труда таким образом, чтобы пользователь просто взаимодействовал с графическим пользовательским интерфейсом, например, представленным на связанном дисплее устройства, например, чтобы пользователь мог получать требуемое представление формата нажатием кнопки FASTQ, BAM, VCF и т.д. в ГПИ, причем один или более движков ускоренной обработки могли бы в фоновом режиме выполнять этапы ускоренной обработки, необходимые для восстановления запрошенного файла в затребованном формате файла из сохраненного файла.

[00614] Как правило, будут сохранять одну или более сжатых версий файла BCL, FASTQ, SAM, BAM, CRAM и/или VCF вместе с небольшим метафайлом, который содержит все конфигурации, использованные системой для создания сжатого и/или сохраненного файла. Такие данные метафайла содержат подробности того, как формировался конкретный формат файла, например, FASTQ и/или BAM, и/или какие шаги понадобятся для возврата назад или продвижения вперед для формирования любого из других форматов файлов. Этот процесс описан более подробно ниже в настоящем документе. Подобным образом процесс может быть продвинут вперед или возращен назад с помощью конфигурации, сохраненной в метафайле. Это может сократить на 80% или более потребности в хранении и экономические издержки, если вычислительная функция связана с функциями сохранения.

[00615] Соответственно, ввиду вышеизложенного, и как показано на ФИГ. 40A,

предложена облачная серверная система для аналитики и хранения данных. Например, с помощью доступной из облака северной системы, которая описана в настоящем документе, пользователь может соединяться с устройством хранения, например, для хранения входных данных. Например, удаленный пользователь может получать доступ к системе, чтобы ввести геномные и/или биоинформационные данные в систему, например для хранения и/или обработки их. В частности, удаленный пользователь системы, например, с помощью локального вычислительного ресурса 100, может получать доступ к системе 1 для выгрузки геномных данных, таких как один или более секвенированных геномов одного или более индивидов. Как подробно описано ниже, система может содержать пользовательский интерфейс, например, имеющий доступ к соответствующим образом сконфигурированному API, который позволит пользователю получить доступ к BioIT-платформе для выгрузки подлежащих обработке данных, управления параметрами обработки и/или загрузке выходных данных, например, данных результатов, с платформы.

[00616] А именно, система может содержать API, например, объекта S3 или «типа S3», который позволяет получать доступ к одной или более памятей системы для хранения 400 и/или приема сохраняемых файлов. Например, может присутствовать доступный из облака объект API, например, когда API выполнен с возможностью сохранения файлов данных в облаке 50, например, в одно или более ведер 500 памяти, например в ведро S3. Соответственно, система может быть выполнена с возможностью обеспечения пользователю доступа к удаленно хранящимся файлам, например, посредством API S3 или типа S3, например, путем доступа к API через облачный интерфейс на персональном вычислительном устройстве.

[00617] Такой API может быть поэтому выполнен с возможностью обеспечения доступа к облаку 50 для соединения тем самым пользователя с одним или более облачных серверов 300, описанных в настоящем документе, например, для выгрузки и/или загрузки данного хранящего файла, например, чтобы сделать его доступным между облачным сервером 300 и локальным накопителем 100 на жестких дисках. Это может быть полезно, например, для обеспечения удаленному пользователю возможности предоставления, получения доступа и/или загрузки данных на или с сервера 300 и последующего выполнения одного или более приложений и/или вычислений на этих данных, либо локально 100, либо на сервере 300, а затем вызова API для отправки преобразованных данных обратно на облако 50 или с него, например, для хранения 200 и/или дальнейшей обработки. Это особенно полезно для извлечения, анализа и хранения геномных данных.

[00618] Однако типичное облачное хранилище данных, например, хранилище «S3» стоит дорого. Эта стоимость возрастает при хранении больших количеств геномных и/или биоинформационных данных, где такие затраты становятся непомерно высокими. Кроме того, время, требуемое на запись, выгрузку и/или загрузку данных для использования, например, либо локально 100, либо удаленно 300, и/или расходы на хранение 400 также делают такие дорогие системы облачных хранилищ менее привлекательными. Предложенные решения, которые описаны в настоящем документе, преодолевают эти и другие недостатки.

[00619] В частности, вместо того, чтобы переходить на объект типичного хранилища «S3» или другой типичный облачный объект, предложенный в настоящем документе API представляет собой альтернативный совместимый с S3 API, который может быть реализован с возможностью снижения скорости передачи и/или стоимости хранения данных. В таком случае, когда пользователь желает сохранить файл, вместо того, чтобы

переходить на типичный облачный, например, S3, API, альтернативная система API услуг, например, например запатентованный S3-совместимый API, описанный в настоящем документе, запустит экземпляр вычисления, например, экземпляр ЦПУ и/или FPGA системы, который сожмет файл, сформирует индекс метаданных, чтобы указать, что представляют собой данные и/или как был сформирован файл и т.д., и затем сохранит сжатый файл с помощью S3-совместимого хранилища типа ведра 400. Соответственно, в настоящем документе представлена облачная 50 служба, использующая экземпляр 300 вычисления, который может быть запущен альтернативным API для сжатия данных перед сохранением 400. В таком случае то, что сохраняют, является поэтому не фактическим файлом, а, скорее, сжатой версией исходного файла.

[00620] А именно, в таком случае начальный файл может быть в первом формате, который может быть загружен в систему посредством патентованного S3-совместимого API, который принимает файл, например, файл F1, и может затем выполнить функцию вычисления на этом файле и/или сжать этот файл, например, посредством соответствующим образом сконфигурированного движка 300 обработки ЦПУ/ГПУ/КПУ/FPGA, который затем подготавливает сжатый файл для хранения в виде сжатого, например, сжатого файла F1. Однако, когда сжатый и хранящийся файл требуется извлечь, он может быть распакован и затем распакованный файл может быть отправлен пользователю. Преимуществом это ускоренной системы сжатия и распаковки является то, что хранение 400 сжатого файла означает невероятную экономию расходов на хранение, причем это преимущество становится возможным за счет функциональных возможностей вычисления и/или сжатия, достигаемых системами, описанными в настоящем документе.

[00621] Следовательно, благодаря быстрым и эффективным функциональным возможностям вычисления и/или сжатия, достигаемым представленными системами, пользователю даже не нужно знать, что файл сжимают перед сохранением и впоследствии распаковывают после хранения и представляют в интерфейсе пользователя. В частности, система функционирует настолько быстро и эффективно, что пользователю не нужно знать множестве этапов сжатия, вычисления и/или распаковки, которые имеют место при сохранении и/или извлечении запрошенных данных пользователю, все это происходит гармонично и своевременно. Однако тот факт, что представленная система хранения будет менее дорогой и более эффективной, чем предыдущие системы хранения, будет очевиден.

[00622] Соответственно, ввиду вышеизложенного, объектно-ориентированные услуги хранения, где услуги хранения могут быть предоставлены за более низкую стоимость за счет объединения экземпляра вычисления и/или сжатия вместе с функциональными возможностями хранения. В таком случае типичная стоимость хранения может быть заменена стоимостью вычисления, которая предлагается на значительно более низком уровне, поскольку, как указано в настоящем документе, стоимость вычисления может быть уменьшена благодаря реализации в ускоренном виде, например, с помощью матрицы FPGA и/или квантовой вычислительной платформы 300, как описано в настоящем документе. Следовательно, ускоренные платформы, описанные в настоящем документе, могут быть выполнены в виде быстрой и эффективной системы хранения и извлечения, позволяющей быстро сохранять сжатые данные, которые могут быть как сжаты, таки и сохранены, а также быстро распакованы и извлечены со значительно меньшими расходами и большей эффективностью и скоростью. Это особенно полезно, когда дело касается хранения 400 геномных данных, и совместимо с функциональными

возможностями обработки «точно в срок», описанными в настоящем документе выше. Поэтому в соответствии с устройствами, системами и способами, описанными в настоящем документе, может быть предусмотрена услуга хранения объектов, где услуга хранения реализует функциональные возможности быстрого сжатия, например, специфичного для геномики сжатия, для хранения данных результатов геномной обработки.

[00623] Более конкретно, как показано на ФИГ. 40A, в одном примере реализации предложенные в настоящем документе BioIT-системы могут быть выполнены таким образом, чтобы конвейерная серверная система 300, например, ее часть, принимала запрос в API, например, S3-совместимом API, который выполнен с возможностью функционального соединения с базой 400 данных, которая выполнена с возможностью связывания начального файла (F1) со сжатой версией файла (CF1), например на основе связанных метаданных. Аналогичным образом после того, как исходные файлы CF1 распакованы и обработаны, файл получающихся данных результатов (F2) может быть затем сжат и сохранен как файл CF2. Соответственно, когда файл требуется извлечь из базы 400 данных, сервер 300 имеет API, который уже связал исходный файл со сжатым файлом посредством соответствующим образом сконфигурированных метаданных, поэтому, когда запрашивается извлечение, контроллер управления рабочими потоками (WMS) системы запустит экземпляр 300 вычисления, который запустит надлежащий экземпляр вычисления, чтобы выполнить любые необходимые вычисления и/или распаковать файл для дальнейшей обработки, передачи и/или представления запрашивающему пользователю 100.

[00624] Следовательно, в различных вариантах реализации пример способа может включать в себя один или более их следующих этапов в любом логическом порядке: 1) Запрос поступает через API, например, S3-совместимый API, 2) API обменивается данными с WMS, 3) WMS заполняет базу данных и инициирует экземпляр (-ы) вычисления, 4) экземпляр (-ы) вычисления выполняет (-ют) необходимое сжатие она файле F1 и формируют характеристические метаданные и/или другие соответствующие ассоциации файла (X), например, для создания файла CF1 X1, 4) тем самым подготавливая данные для хранения 400. Затем этот процесс может быть повторен для файлов F2, F3, …, Fn, например, другой обработанной информации, чтобы WMS знал, как сжатый файл был сформирован, а также где и как он был сохранен. Необходимо отметить, что уникальным признаком данной системы является то, что доступ к сохраненным данным 400 может быть предоставлен нескольким другим пользователям 100 по существу одновременно. Например, системы и способы сжатия, описанные в настоящем документе, полезны в совокупности с BioIT-платформами, описанными в настоящем документе, благодаря чему в любое время в ходе процесса обработки данные результатов могут быть сжаты и сохранены в соответствии со способами, описанными в настоящем документе, и сделаны доступными для других пользователей с надлежащими полномочиями.

[00625] Что касается выполнения геномного анализа, пользователь 100 может получить доступ к системе 300, описанной в настоящем документе, например, посредством API геномного анализа, такого как S3 или S3-совместимый API, выгрузить геномные данные, например в формате файла BCL и/или FASTQ или другом формате файла, и тем самым запросить выполнение одной или более геномных операций, таких как картирование, выравнивание, сортировка, удаление дубликатов, определение вариантов, и/или других операций. Система 300 принимает запрос в API диспетчера рабочих потоков, затем система диспетчера рабочих потоков оценивает входящие

запросы, индексирует задания, формирует очередь, выделяет ресурсы, например, выделение экземпляров, и формирует конвейерный поток. Соответственно, после поступления, предварительной обработки и постановки в очередь запроса распределитель экземпляров, например, API, раскрутит различные специфичные для заданий экземпляры, описанные более подробно ниже в настоящем документе, в соответствии с рабочими проектами. Следовательно, после того, как задание индексировано, поставлено в очередь и/или сохранено в соответствующей базе 400 данных, диспетчер рабочих потоков извлечет затем данные из хранилища 400, например, S3 или S3-совместимого хранилища, перезапустит соответствующий экземпляр, которые извлечет файл, и запустит надлежащий процесс на данных, чтобы выполнить одно или более запрошенных заданий.

[00626] Кроме того, в случае запроса на выполнение на данных множества заданий, требующих выполнения множества экземпляров, по завершении выполнения своих операций первым экземпляром, данные результатов могут быть сжаты и сохранены, например, в соответствующем экземпляре памяти, например, в первой базе данных, такой как эластичное или гибкое устройство хранения, чтобы дождаться, когда другие экземпляры конвейера раскрутятся и извлекут данные результатов для дальнейшей обработки, например, в соответствии с системами и способами, описанными в настоящем документе. Далее, по мере поступления новых запросов и/или продолжения выполнения текущих заданий система управления рабочими потоками будет постоянно обновлять очередь, чтобы выделять задания надлежащим экземплярам посредством API распределителя экземпляров, чтобы поддерживать эффективность потока данных через систему и выполнения процессов системы.

[00627] Аналогичным образом система 300 может постоянно принимать данные результатов и сохранять данные 200/400, например, в первой и второй базах данных перед дальнейшей обработкой и передачей, например, передачей обратно первоначальному инициатору запроса 100 или назначенной стороне. В определенных случаях данные результатов могут быть сжаты, как описано в настоящем документе, перед сохранением 400 и/или отправкой. Далее, как указано выше, сформированные файлы данных результатов после сжатия могут содержать надлежащие метаданные и/или другие связанные данные, причем данные результатов могут быть обозначены по-разному по мере протекания через систему, например, переходя от файла F1 к файлу F1C, к файлу F2, к файлу F2C и т.д. по мере того, как данные обрабатываются и перемещаются через конвейер платформы, например, как указано API ассоциаций файлов.

[00628] Соответственно, благодаря патентованным специализированным API, которые описаны в настоящем руководстве, система может иметь общий каркас, к которому могут быть присоединены другие услуги, и/или дополнительные ресурсы, например, экземпляры, могут быть приведены в действие, чтобы обеспечить беспрепятственное и эффективную выполнение операций конвейера. Аналогичным образом при необходимости сжатые и сохраненные файлы данных результатов могут быть вызваны, в результате чего диспетчер рабочих потоков раскрутит надлежащие экземпляры вычисления и/или базы данных, чтобы распаковать данные результатов для представления инициатору запроса. Следует отметить, что в различных случаях указанный экземпляр вычисления и сжатия, как и указанный экземпляр вычисления и распаковки, могут быть одним или множеством экземпляров и могут быть реализованы как ЦПУ, FPGA или жестко связанные ЦПУ/FPGA, жестко связанные ЦПУ/ЦПУ или жестко связанные FPGA/FPGA. В определенных случаях один или более из этих и других

экземпляров, описанных в настоящем документе, могут быть реализованы в виде квантового процессорного устройства.

[00629] Соответственно, с учетом описанного в настоящем документе в соответствии с одним аспектом предложено устройство для осуществления одной или множества функций при выполнении операций геномного анализа последовательности. Например, после того, как данные приняты, например, удаленным пользователем 100, и/или сохранены 400 в облачной системе, входные данным могут быть доступны для WMS и могут быть приготовлены для дальнейшей обработки, например, для вторичного анализа, результаты которой могут быть затем переданы обратно локальному пользователю 100, например, после сжатия, сохранены 400 и/или подвергнуты дополнительной обработке, например, третичной обработке, с помощью системного сервера 300.

[00630] В определенных случаях этапы вторичной обработки, описанные в настоящем документе, к конкретных реализациях могут быть выполнены локальным вычислительным ресурсом 100 и могут быть реализованы программным и/или аппаратным обеспечением, например, дополнительным внешним вычислительным ресурсом 200, где вычислительный ресурс 200 содержит ядро ЦПУ, например от примерно 4 до примерно 14 примерно 24 или более ядер ЦПУ, и может дополнительно содержать одну или более матриц FPGA. Локальный дополнительный внешний ресурс 100 может быть выполнен с возможностью доступа к большому блоку 200 хранения, например, 120 ГБ памяти ОЗУ, причем доступ может быть прямым, например, посредством прямого соединения между ними, или опосредованным, например, путем соединения с возможностью обмена данными между ними по локальной облачной сети 30.

[00631] А именно, в локальной системе данные могут передаваться в память 200 или из нее посредством соответствующим образом сконфигурированных SSD-накопителей, которые выполнены с возможностью записи данных заданий на обработку, например, задач, связанных с геномикой, подлежащих обработке, и чтения обработанных данных результатов из памяти 200. В различных вариантах реализации локальный вычислительный ресурс 100 может быть соединен с возможностью обмена данными с секвенатором 110, откуда может быть получен файл BCL и/или FASTQ, например, из секвенатора, и записан в SSD-накопители напрямую, например, посредством соответствующим образом сконфигурированного межсоединения. После этого локальный вычислительный ресурс 100 может выполнить одну или более операций вторичной обработки на данных. Например, в одном варианте реализации локальный вычислительный ресурс представляет собой сервер LINUX®, имеющий 24 ЦПУ, которые могут быть связаны с соответствующим образом сконфигурированной матрицей FPGA, выполненной с возможностью выполнения одной или более операций вторичной обработки, описанных в настоящем документе.

[00632] Следовательно, в различных случаях локальное вычислительное устройство 100 может быть вычислительным решением в виде «автоматизированного рабочего места», имеющего набор BioIT-микросхем, который выполнен с возможностью осуществления одной или более из вторичной и/или третичной обработки данных генетики. Например, как описано в настоящем документе, вычислительный ресурс 100 может быть связана с платой PCIe, которая вставлена в вычислительное устройство, чтобы тем самым быть связанной с одним или более внутренних ядер ЦПУ, ГПУ, КПУ и/или связанными памятями. В частности, компоненты вычислительного устройства 100, содержащие процессорные устройства, связанные памяти и/или связанные платы

PCIe с имеющимися там один или более наборами микросхем FPGA/ASIC, могут поддерживать обмен данными друг с другом, причем все они могут быть предусмотрены внутри корпуса, например, в виде коробочной версии, которая типична в данной области техники. Более конкретно, коробочная версия может быть выполнена с возможностью использования для автоматизированного рабочего места или, в различных случаях, она может быть выполнена с возможностью использования в удаленно доступной серверной стойке. В других вариантах реализации наборы микросхем ЦПУ/FPGA/памяти и/или связанные платы межсоединения ExpressCard могут быть связаны внутри устройства секвенирования нового поколения с образованием там одного блока.

[00633] Соответственно, в одном конкретном случае настольная коробочная версия может содержать множество ЦПУ/ГПУ/КПУ, связанных с одной или более матриц FPGA, например 4 ЦПУ/ГПУ или 8, или 12, 16, 20, 22, 24 ЦПУ или более, которые могут быть связаны с 1 или 2, или 3, или более матриц FPGA, например, в одном корпусе. А именно, в одном конкретном случае предложена коробочная версия вычислительного ресурса, которая содержит 24 ядра ЦПУ, выполненную с возможностью изменения конфигурации матрицы FPGA, базу данных, например, 128×8 ОЗУ, один или более SSD, например, когда матрица FPGA выполнена с возможностью по меньшей мере частичного изменения конфигурации между операциями, например, между выполнением картирования и выравнивания. Следовательно, в таком случае файлы BCL и/или FASTQ, формируемые устройством 110 секвенирования, могут быть считаны в ЦПУ и/или переданы в матрицу FPGA для обработки, а данные ее результата могут быть возвращены на связанные ЦПУ через SSD-накопители. Соответственно, в данном варианте реализации локальная вычислительная система 100 может быть выполнена с возможностью сброса различных вычислительноемких функций на связанную матрицу FPGA, тем самым улучшая скорость, точность и эффективность биоинформационной обработки. Однако, хотя решение 100 в виде коробочной версии полезно, например, в локальном оборудовании, оно может не подойти для доступа множеством пользователей, которые могут находиться удаленно от коробочной версии.

[00634] В частности, в различных случаях может быть предусмотрен облачный сервер 50, например, когда сервер 300 может быть доступен удаленно. Соответственно, в конкретных случаях могут быть предусмотрены одна или более интегральных схем (ЦПУ, FPGA, КПУ), описанных в настоящем документе, которые могут быть выполнены с возможностью доступа к ним посредством интерфейса на основе облака 50. Следовательно, в конкретных случаях может быть предусмотрена коробочная версия вычислительного ресурса в виде коробочной версии автоматизированного рабочего места, как описано выше, причем конфигурация коробочная версия может быть выполнена с возможностью переноса на облако и удаленного доступа к ней. Однако такой конфигурации может быть недостаточно для того, чтобы справляться с большим объемом трафика от удаленных пользователей. Соответственно, в других случаях одна или более интегральных схем, описанных в настоящем документе, могут быть выполнены в виде сервера на основе решения 300, которое может быть сконфигурировано как часть серверной стойки, например, когда доступная на сервере система выполнена с возможностью удаленного доступа к ней, например посредством облака 50.

[00635] Например, в одном варианте реализации может быть предусмотрен вычислительный ресурс, или локальный сервер 100, имеющий одно или более, например, множество, ядер ЦПУ, и/или ГПУ, и/или КПУ и связанные памяти вместе с одной или более FPGA/ASIC, описанных в настоящем изобретении. В частности, как указано выше,

в одной реализации может быть предусмотрена коробочная версия, которая включает в себя от 18 до 20, до 24 или более ядер ЦПУ/ГПУ, причем коробочная версия имеет SSD, ОЗУ 128×8, и одну или более BioIT-схем FPGA/ASIC, а также включает в себя соответствующим образом сконфигурированный модуль связи, имеющий передатчики, приемники, антенны, а также возможности для обмена данными по WIFI, Bluetooth и/или сотовой связи, которые выполнены с возможностью обеспечения удаленного доступа к коробочной версии. В данной реализации, например, когда предусмотрена одна FPGA, FPGA могут быть выполнены с возможностью изменения конфигурации, например, частичного изменения конфигурации, между одним или множеством различных этапов конвейера геномного анализа.

[00636] Однако, в других случаях предложена серверная система, которая может включать в себя от около 20 до 24, до 30, до 34, до 36 или более ядер ЦПУ/ГПУ и ОЗУ объемом около 972 ГБ или более, которая может быть связана с одной или более, например, около двух, или около четырех, или около шести, или около восьми или более FPGA, которые могут быть выполнены с возможностью конфигурирования, как описано в настоящем документе. Например, в одной реализации одна или более FPGA могут быть выполнены с возможностью изменения конфигурации, например, частичного изменения конфигурации, между одним или множеством различных этапов конвейера геномного анализа. Однако в различных других реализациях может быть предусмотрен набор специализированных матриц FPGA, например, каждая из которых специально предназначена для выполнения определенной BioIT-операции, такой как картирование, выравнивание, определение вариантов и т.д., тем самым избавляя от этапа изменения конфигурации.

[00637] Соответственно, в различных случаях могут быть предусмотрены одна или более матриц FPGA, например, когда матрицы FPGA выполнены с возможностью изменения конфигурации между различными операциями конвейера. Однако в других случаях одна или более матриц FPGA могут быть выполнены с возможностью специализации на выполнении одной или более функций без необходимости частичного или полного конфигурирования их. Например, матрицы FPGA, предложенные в настоящем документе, могут быть выполнены с возможностью специализации на выполнении одной или более вычислительноемких операций в BioIT-конвейере, например, когда предусмотрена одна FPGA, которая специально предназначена для выполнения операции картирования, и предусмотрена другая FPGA, которая выполнена с возможностью выполнения операции выравнивания, хотя, в некоторых случаях, может быть предусмотрена одна FPGA, которая выполнена с возможностью частичного изменения конфигурации между выполнением обоих операций, картирования и выравнивания.

[00638] Кроме того, в число других операций в конвейере, которые тоже могут осуществляться выполненными с возможностью измерения конфигурации или специализированными FPGA, могут входить выполнение операции преобразования/перестановки BCL, операции Смита-Ватермана, операции HMM, операции локального повторного выравнивания и/или различных других операций определения вариантов. Аналогичным образом различные операции конвейера могут быть выполнены с возможностью выполнения одной или более из связанных ЦПУ/ГПУ/КПУ системы. Такие операции могут быть одной или более вычислительноемкими операциями конвейера, например, для выполнения сортировки, удаления дубликатов и других операций определения вариантов. Следовательно, можно выполнить всеобъемлющую систему с возможностью осуществления комбинации операций, частично с помощью

ЦПУ/ГПУ/КПУ, и частично с помощью аппаратного обеспечения, такого как FPGA/ASIC системы.

[00639] Соответственно, как показано на ФИГ. 40B, в различных реализациях облачной системы 50 она может включать в себя множество вычислительных ресурсов, в том числе множество экземпляров и/или слоев экземпляров, например, когда экземпляры и/или слои экземпляров выполнены с возможностью осуществления одной или более операций BioIT-конвейера, описанных в настоящем документе. Например, могут быть предусмотрены различные экземпляры ЦПУ/ГПУ/КПУ и/или жестко смонтированных интегральных схем для выполнения специализированных функций конвейера анализа генома, предложенные в настоящем документе. Например, могут быть предусмотрены различные экземпляры FPGA для выполнения специализированных операций геномного анализа, например, экземпляр FPGA для выполнения картирования, другой для выполнения выравнивания, еще один для выполнения локального повторного выравнивания и/или других операций Смита-Ватермана, следующий для выполнения операций HMM и т.п.

[00640] Аналогичным образом могут быть предусмотрены различные экземпляры ЦПУ/ГПУ/КПУ для выполнения специализированных операций геномного анализа, например, экземпляры ЦПУ/ГПУ/КПУ для выполнения обработки сигнала, сортировки, удаления дубликатов, сжатия, различных операций определения вариантов и т.п. В таких случаях могут быть предусмотрены связанные память или памяти, например, между различными этапами вычисления конвейера, для приема данных результатов по мере их вычисления, компиляции и обработки по всей системе, например, между различными экземплярами ЦПУ и/или FPGA и/или их слоями. Кроме того, необходимо отметить, что размер различных экземпляров ЦПУ и/или FPGA может меняться в зависимости от вычислительных потребностей облачной системы и может измениться в диапазоне от малого до среднего, крупного и очень крупного, как и может меняться количество экземпляров ЦПУ/ГПУ/КПУ и FPGA/ASIC.

[00641] Кроме того, как показано на ФИГ. 40B, система может также содержать диспетчер рабочих потоков, который выполнен с возможностью планирования и направления движения данных по всей системе и от одного экземпляра к другому и/или из одной памяти в другую. В некоторых случаях память может представлять собой множество памятей, которые являются специализированными памятями, специфичными для экземпляров, а в других случаях память может быть одной или более памятями, выполненными с возможностью быть эластичными и потому способными переключаться с одного экземпляра на другой, как переключаемое эластичное блочное устройство хранения. В третьих случаях память может неспецифичной для экземпляров и, следовательно, выполненной без возможности соединения с возможностью обмена данными с множеством экземпляров, например, для эластичного хранения файлов.

[00642] Кроме того, диспетчер рабочих потоков сам может быть специализированным экземпляром, например, ядром ЦПУ/ГПУ/КПУ, которое специально предназначено для и/или выполнено с возможностью определения, какие задания необходимо выполнять, и когда и какие ресурсы будут использованы в при выполнении этих заданий, а также постановки в очередь заданий и направления их от ресурса к ресурсу, например, от экземпляра к экземпляру. Диспетчер рабочих потоков может содержать или может быть иным образом выполнен как оценщик загрузки, и/или образовать эластичный узел управления, представляющий собой специально предназначенный экземпляр, который может выполняться процессором, например ядром ЦПУ/ГПУ/КПУ. В различных случаях диспетчер рабочих потоков может иметь базу данных, соединенную

с ним, которая может быть выполнена с возможностью управления всеми заданиями, которые должны быть обработаны, обрабатываются или были обработаны. Следовательно, диспетчере WMS может быть выполнен с возможностью обнаружения и управления потоками данных по всей системе, определения того, как выделять системные ресурсы, и когда приводить в действие больше ресурсов.

[00643] Как указано выше, в определенных случаях может быть предусмотрено решение для автоматизированного рабочего места и/или сервера одновременно, где вычислительное устройство содержит множество серверов из X ядер ЦПУ, имеющих размер Y, которые выполнены с возможностью подачи данных в одну или более FPGA размера Z, где X, Y и Z являются числами, которые могут меняться в зависимости от потребностей обработки системы, но их следует выбирать и/или иным образом конфигурировать, чтобы они были оптимальными, например 10, 14, 18, 20, 24, 30 и т.д. Например, типичные конфигурации системы оптимизированы для выполнения BioIT-операций системы, описанной в настоящем документе. А именно, определенные конфигурации системы оптимизированы таким образом, чтобы максимально увеличивать поток данных через различные экземпляры ЦПУ/ГПУ/КПУ в различные интегральные схемы, такие как FPGA, системы, причем размер ЦПУ и/или FPGA может меняться в зависимости друг от друга, исходя из потребностей обработки системы. Например, одно или боле из ЦПУ и/или FPGA могут иметь размер, который является относительно маленьким, средним, большим, очень большим или сверхбольшим. Точнее говоря, архитектура системы может быть выполнена таким образом, чтобы аппаратное обеспечение ЦПУ/FPGA имело такой размер и было выполнено таким образом, чтобы оно работало оптимально эффективным образом для поддержания платформ обоих экземпляров занятыми в течение всей работы, например, когда ЦПУ превосходят численно FPGA в соотношении 4 к 1, 8 к 1, 16 к 1, 32 к 1, 64 к 2 и т.д.

[00644] Следовательно, хотя в целом хорошо, когда FPGA имеют большие возможности, однако обработка данных с использованием высокопроизводительных FPGA неэффективна, в систему подаются недостаточные для обработки данные. В таком случае может быть реализована одна FPGA или ее часть. В частности, в идеальном варианте система управления рабочими потоками направляет поток данных в определенные ЦПУ и/или FPGA, которые выполнены с возможностью поддержания системы и ее компонентов постоянно занятыми вычислением. Например, в одном примере конфигурации одно или более, например 2, 3 или 4 и более, ядер ЦПУ/ГПУ/КПУ могут быть выполнены с возможностью подачи данных в небольшую, среднюю, большую или очень большую FPGA или ее часть. А именно, в одном варианте реализации может быть предусмотрен определенный экземпляр ЦПУ, например, для выполнения одной или более операций BioIT-обработки, описанных в настоящем документе, например, когда экземпляр ЦПУ доступен из облака и содержит до 4, 8, 16, 24, 30, 36 ядер ЦПУ, причем эти ядра могут или не могут быть выполнены с возможностью функционального соединения с частью одной или более FPGA.

[00645] Например, может быть предусмотрена доступная из облака серверная стойка 300, где сервер содержит экземпляр ядра ЦПУ, имеющий от около 4 ядер ЦПУ до около 16 или около 24 ядер ЦПУ, которые выполнены с возможностью функционального соединения с экземпляром FPGA. Например, может быть предусмотрен экземпляр FPGA, например, когда средний размер FPGA составляет X, а входящие в состав FPGA могут иметь размер около 1/8X, X, 2,5X и до 8X, или даже около 16X или больше. В различных случаях могут быть включены дополнительные ядра ЦПУ/ГПУ/КПУ или FPGA и/или обеспечены в виде объединенного экземпляра, например, когда имеется

большое количество данных для обработки, и когда множество ядер ЦПУ выбраны для поддержки постоянной занятости FPGA. Следовательно, соотношение между ЦПУ и FPGA можно подбирать, комбинируя их таким образом, чтобы оптимизировать поток данных, и, таким образом, систему можно выполнить с возможностью масштабирования в сторону увеличения или уменьшения в зависимости от потребностей, например, чтобы минимизировать расходы при оптимальном использовании в зависимости от рабочих потоков.

[00646] Однако, когда ЦПУ не создают достаточно работы для поддержания полной занятости и/или полного использования FPGA, конфигурация будет неидеальной. Поэтому в настоящем документе предложена гибкая архитектура из одного или более экземпляров, которые могут быть напрямую связаны вместе, или выполнены с возможностью связывания вместе таким образом, чтобы приспосабливаться для эффективной работы программного/аппаратного обеспечения ЦПУ/FPGA с тем, чтобы ЦПУ/ГПУ/КПУ оптимально снабжали доступные FPGA и/или их часть так, чтобы платформы обоих экземпляров были постоянно заняты. Соответственно, разрешение доступа к такой системе с помощью облака обеспечит, чтобы подаваемые в нее данные помещались в очередь диспетчером рабочих потоков и направлялись в определенные ресурсы ЦПУ/FPGA, которые сконфигурированы и могут принимать и обрабатывать данные оптимально эффективным образом.

[00647] Например, в некоторых конфигурациях доступные из облака экземпляры могут содержать ЦПУ/ГПУ/КПУ разных размеров в разных количествах и, кроме того, могут быть доступные из облака экземпляры, которые содержат FPGA (или ASIC) и/или КПУ разных размеров в разных количествах Это даже могут быть экземпляры, имеющие комбинации данных экземпляров. Однако в различных итерациях предусмотренные ЦПУ/ГПУ/КПУ и/или FPGA/КПУ и/или смешанные экземпляры могут иметь слишком много одного экземпляра и/или слишком мало другого экземпляра для эффективной работы представленных платформ конвейерной BioIT-обработки, описанных в настоящем документе. Соответственно, в настоящем документе предложены системы и архитектуры, их гибкие комбинации и/или способы их реализации для эффективного формирования и использования конвейерных платформ геномной и/или биоинформационной обработки, например, сделанные доступными посредством облака 50.

[00648] В таких системах ЦПУ/ГПУ/КПУ могут быть выбраны в таком количестве и сконфигурированы таким образом, чтобы обрабатывать менее вычислительноемкие операции, а количество и конфигурации FPGA и/или КПУ могут быть адаптированы для решения вычислительноемких задач, например, когда данные беспрепятственно передаются туда и обратно между экземплярами ЦПУ/ГПУ/КПУ и FPGA/КПУ. Кроме того, могут быть предусмотрены одна или более памятей для сортировки данных, например, данных результатов, между всевозможными разными этапами процедур и/или между различными типами экземпляров, чтобы исключить существенные периоды задержки экземпляров. А именно, во время картирования и выравнивания ЦПУ/ГПУ используются очень мало, поскольку вследствие интенсивного характера вычислений эти задачи выполнены с возможностью осуществления аппаратными реализациями. Аналогичным образом во время определения вариантов задачи могут быть разбиты таким образом, чтобы они были примерно одинаково распределены между экземплярами ЦПУ/FPGA в виде их задач, например, когда операции Смита-Ватермана и HMM могут быть выполнены аппаратным обеспечением, а различные другие операции могут быть выполнены программным обеспечением, исполняемым на одном или более

экземплярах ЦПУ/ГПУ/КПУ.

[00649] Соответственно, параметры архитектуры, указанные в настоящем документе, необязательно ограничены однажды установленной архитектурой, скорее, система выполнена с возможностью обладания более большей гибкостью в организации ее реализаций и опирается на диспетчер рабочих потоков, который определят, какие экземпляры активны, когда, как и как долго, и указывает, какие вычисления на каких экземплярах выполняются. Например, количество ЦПУ и/или FPGA, которые нужно привести в действие и оперативно связать вместе, должны быть выбраны и сконфигурированы таким образом, чтобы активированные ЦПУ и FPGA, а также сопровождающее их программное/аппаратное обеспечение, поддерживались оптимально занятыми. В частности, количество ЦПУ и их функционирование должны быть сконфигурированы для обеспечения постоянной занятости данного количества FPGA или их части, чтобы ЦПУ оптимально и эффективно снабжали FPGA для умелого поддержания работы обоих экземпляров и их компонентов.

[00650] Следовательно, таким образом контроллер управления рабочими потоками системы может быть выполнен с возможностью доступа к рабочему процессу и организации и деления его таким образом, чтобы задачи, которые могут быть более оптимально выполнены с помощью ЦПУ/ГПУ, направлялись на ряд ЦПУ, необходимых для оптимального выполнения этих операций, а те задачи, которые могут быть более оптимально выполнены с помощью FPGA/ASIC/КПУ, направлялись на ряд FPGA, необходимых для оптимального выполнения этих операций. Также может быть включена эластичная и/или эффективная память для эффективной передачи данных результатов этих операций из одного экземпляра в другой. Таким образом, комбинация машин и памятей может быть сконфигурирована и объединена так, чтобы она была оптимально масштабирована с учетом объема работы, который нужно выполнить, а также конфигурации и использования экземпляров для лучшего выполнения работы эффективно и без лишних затрат.

[00651] А именно, облачные архитектуры, описанные в настоящем документе, показывают, что различные известные недостатки предыдущих вариантов архитектуры могут привести к проявлениям неэффективности, которые можно преодолеть за счет гибкого предоставления большему количеству ядер ЦПУ/ГПУ/КПУ доступа к всевозможным разным аппаратным экземплярам, например, FPGA, или их частям, которые организованы более целенаправленно, чтобы быть в состоянии специально выделять надлежащий экземпляр для выполнения предназначенных функций оптимально за счет реализации в таком формате. Например, система может быть выполнена с возможностью обеспечения удаленного доступа к более значительной части имеющихся экземпляров ЦПУ/ГПУ, чтобы они были постоянно заняты, создавая данные результатов, которые могут быть оптимально поданы в имеющиеся экземпляры FPGA/КПУ для поддержания постоянной занятости выбранных экземпляров FPGA. Следовательно, желательно обеспечить структурированную архитектуру, которая максимально эффективна и постоянно занята. Необходимо отметить, что конфигурации, где слишком мало ЦПУ снабжают слишком много FPGA, так что одна или более FPGA используются не в полной мере, являются неэффективными и их следует избегать.

[00652] В одной реализации, как показано на ФИГ. 40В, архитектура может быть выполнена с возможностью виртуального включения нескольких различных слоев или уровней, таких как первый уровень, имеющий первое количество X ядер ЦПУ, например, от 4 до около 30 ядер ЦПУ, и второй уровень, имеющий от 1 до 12 и более экземпляров матриц FPGA, размер которых может меняться в диапазоне от малого до среднего или

большого и т.д. Можно включить третий уровень ядер ЦПУ и/или четвертый уровень дополнительных матриц FPGA и т.д. Таким образом, на облачном сервере 300 имеется множество доступных экземпляров, например, экземпляров, которые просто содержат ЦПУ или ГПУ, и/или экземпляров, которые содержат матрицы FPGA, или их комбинации, например, на одном или более уровнях, описанных в настоящем документе. Соответственно, подобным образом архитектура может быть гибко или эластично организованной, чтобы наиболее интенсивные специальные вычислительные функции выполнялись аппаратными экземплярами или КПУ, а те функции, которые могут быть выполнены с помощью ЦПУ, направлялись на соответствующий уровень в целях общей обработки, причем при необходимости количество экземпляров ЦПУ/FPGA может быть увеличено или уменьшено системой по мере надобности.

[00653] Например, размер архитектуры может быть эластичным, чтобы минимизировать издержки системы при максимально оптимальном использовании системы. А именно, архитектура может быть выполнена с возможностью максимального повышения эффективности и сокращения задержки путем сочетания различных экземпляров на различных виртуальных уровнях. В частности, множество, например, значительная часть и/или все, экземпляров ЦПУ/ГПУ уровня 1 могут быть выполнены с возможностью снабжения различных экземпляров FPGA уровня 2, которые специально выполнены с возможностью осуществления специальных функций, таких как FPGA для картирования и FPGA для выравнивания. На другом уровне могут быть предусмотрены одно или более дополнительное (или столько же, сколько на уровне 1) ЦПУ, например, для выполнения операций сортировки и/или удаления дубликатов и/или различных операций определения вариантов. Более того, один или более дополнительных слоев FPGA могут быть выполнены с возможностью осуществления операций определения вариантов Нидлмана-Вунша, Смита-Ватермана и HMM и т.п. Следовательно, первый уровень ЦПУ может быть задействован для формирования начального уровня геномного анализа, например, выполнения этапов общей обработки, включая организацию очереди и подготовку данных для дальнейшего конвейерного анализа, причем данные после обработки одним или множеством ЦПУ, могут быть поданы на один или более дальнейших уровней специализированных FPGA, например, где экземпляр FPGA выполнен с возможностью осуществления вычислительноемких функций.

[00654] Таким образом, в конкретной реализации экземпляры ЦПУ/ГПУ в конвейере направляют свои данные, после их подготовки, в один или два экземпляров FPGA уровня 2, предназначенных для картирования и выравнивания. По выполнении картирования результирующие данные могут быть сохранены в памяти и/или поданы затем в экземпляр выравнивания, где может быть выполнено выравнивание, например, по меньшей мере одним экземпляром FPGA уровня 2, специально предназначенным для выравнивания. Аналогичным образом обработанные картированные и выровненные данные могут быть затем сохранены в память и/или направлены в экземпляр ЦПУ уровня 3 для дальнейшей обработки, который может быть тем же самым, что и на уровне 1, или другим экземпляром, например, для выполнения менее ресурсоемких функций обработки геномного анализа, например, для выполнения функции сортировки. Кроме того, после того, как ЦПУ уровня 3 выполнили свою обработку, полученные в результате данные могут быть направлены либо обратно в экземпляры FPGA уровня 2, либо в экземпляры FPGA уровня 4, например, для ресурсоемких функций дальнейшей геномной обработки, таких как функции обработки Нидлмана-Вунша (NW), Смита-Ватермана (SW), например, в экземпляре FPGA, специально предназначенном для NW

или SW. Аналогичным образом после того, как анализ SW выполнен, например, с помощью FPGA, специально предназначенной для SW, обработанные данные могут быть отправлены в одну или более связанных памятей и/или далее по конвейеру обработки, например, в экземпляр ЦПУ и/или FPGA уровня 4 или 5, или обратно на уровень 1 или 3, например, для выполнения анализа HMM и/или определения вариантов, например, в специализированной FPGA и/или ядре обработки ЦПУ дальнейшего уровня.

[00655] Подобным образом можно преодолеть проблемы задержки и эффективности за счет сочетания всевозможных разных экземпляров на одном или более разных уровнях, чтобы обеспечить конвейерную платформу для геномной обработки. Такая конфигурация может подразумевать больше, чем масштабирование и/или комбинирование экземпляров, экземпляры могут быть выполнены с возможностью специализации на осуществлении специальных функций. В таком случае экземпляр FPGA для картирования выполняет только картирование, и, аналогично экземпляр FPGA для выравнивания выполняет только выравнивание и т.д., вместо того, чтобы один экземпляр выполнял обработку в конвейере от начала до конца. Хотя и в других конфигурациях одна или более FPGA могут быть по меньшей мере частично переконфигурированы, например между выполнением задач конвейера. Например, в определенных вариантах реализации, поскольку подлежащий выполнению в них геномный анализ является многоэтапным процессом, код FPGA может быть выполнен с возможностью изменения посреди процесса обработки, например, когда FPGA завершает операцию картирования, она может быть переконфигурирована для выполнения одного или более из выравнивания, определения вариантов, Смита-Ватермана, HMM и т.п.

[00656] Следовательно, диспетчер конвейера, например, система управления рабочими потоками, может управлять очередью запросов на геномную обработку, создаваемых экземплярами ЦПУ уровня 1, чтобы разбивать их на дискретные задания, агрегировать и направлять в соответствующий специфичны для задания ЦПУ, а затем в специфичные для задания экземпляры FPGA для дальнейшей обработки, например, для картирования и/или выравнивания, например, на уровне 2, причем обработанные картированные и выровненные данные могут быть отправлены назад или вперед на следующий уровень обработки ЦПУ/FPGA данных результатов, например, для выполнения различных этапов в модуле определения вариантов.

[00657] Например, функция определения вариантов может быть разбита на множество операций, которые могут быть выполнены в программном обеспечении, затем направлены на обработку Смита-Ватермана и/или HMM в один или более аппаратных экземпляров FPGA, а затем могут быть отправлены в ЦПУ для продолжения операций определения вариантов, например, когда вся платформа эластично и/или эффективно подобрана по размеру и реализована для сведения к минимуму стоимости дорогих экземпляров FPGA при одновременном максимальном повышении использования, максимальном снижении задержки и, следовательно, оптимизации операций. Соответственно, таким образом требуется меньше аппаратных экземпляров с учетом их абсолютны возможностей обработки и специфичности аппаратной реализации, и поэтому количество FPGA относительно ЦПУ может быть сведено к минимуму, а их использование, например, FPGA, может быть максимально повышено, и поэтому система может быть оптимизирована, чтобы поддерживать постоянную занятость всех экземпляров. Такая конфигурация является оптимальной конструкцией для анализа геномной обработки, особенно для картирования, выравнивания и определения вариантов.

[00658] Дополнительный структурный элемент, который можно включить, например, вспомогательное устройство, в архитектуру конвейера, описанную в настоящем документе, это один или более модулей эффективной памяти, которые могут быть выполнены с возможностью функционирования для обеспечения хранения блоков данных, например, данных результатов, по мере их передачи по всему конвейеру. Соответственно, одно или более эластичных блочных хранилищ данных (EBDS) и/или более эффективных (гибких) модулей хранения блочных данных могут быть вставлены между одним или более уровнями обработки, например, между различными экземплярами и/или уровнями экземпляров. В таком случае устройство хранения может быть сконфигурирован таким образом, чтобы по мере обработки данных и получения информации обработанные результаты могли быть направлены в устройство хранения для хранения перед отправкой на следующий уровень обработки, например, с помощью специализированного модуля обработки FPGA. То же самое устройство хранения может быть использовано между всеми экземплярами, или множество устройств хранения могут быть использованы между различными экземплярами и/или уровнями экземпляров, например, для хранения, и/или компиляции, и/или постановки в очередь данных результатов. Соответственно, могут быть предусмотрены одна или более памятей таким образом, чтобы различные экземпляры системы могли быть связаны и/или имели доступ к одной и той же памяти, чтобы они могли просматривать и получать доступ к одним и тем же или аналогичным файлам. Следовательно, могут присутствовать одна или более эластичных памятей (памятей, выполненных с возможностью последовательного связывания с множеством экземпляров) и/или эффективных памятей (памятей, выполненных с возможностью связывания с множеством экземпляров одновременно), посредством которых различные экземпляры системы могут быть выполнены с возможностью чтения и записи в одну и ту же или аналогичную память.

[00659] Например, в одном примере реализации, имеющем отношение к конфигурациям, использующим такие эластичные памяти, перед отправкой данных непосредственно из одного экземпляра и/или с одного уровня обработки на другой, данные могут быть направлены в EBDS или другое запоминающее устройство или структуру, например, блок эффективной памяти, для хранения и последующего направления в надлежащий модуль аппаратной обработки. А именно, модуль хранения блоков может быть присоединен к узлу в качестве запоминающего устройства, где данные могут быть записаны в блочное хранилище данных (BDS) для хранения на одном уровне, BDS может быть переключено на другой узел, чтобы направить данные на следующий уровень обработки. Таким образом, один или более, например множество, модулей BDS могут быть включены в конвейер и выполнены с возможностью переключения с одного узла на другой для участия в переходе данных по всему конвейеру.

[00660] Кроме того, как указано выше, может быть использовано более гибкое устройство хранения файлов, такое как устройство, которое выполнено с возможностью связывания с одним или более экземплярами одновременно, например, без необходимости переключения с одного на другой. Подобным образом система может быть эластично масштабирована на каждом уровне системы, например, когда каждый уровень в ней может отличаться количеством узлов для обработки данных на этом уровне, а после обработки данные результатов могут быть записаны на одно или более связанных устройств EBDS, которые затем могут быть переключены на следующий уровень системы, чтобы сделать сохраненные данные доступными следующему уровню

процессоров для выполнения их специфических задач на этом уровне.

[00661] Соответственно, в конвейере обработки существует множество этапов, например, на его обслуживающих узлах, на которых данные подготавливают для обработки, например, предварительная обработка, и после подготовки данные направляют в соответствующие экземпляры обработки на одном уровне, где могут быть сформированы данные результатов, затем результирующие данные могут быть сохранены, например, на устройстве EDS, поставлены в очередь и подготовлены для следующей стадии обработки путем переключения на следующий узел экземпляров и направлены в следующий экземпляр для обработки с использованием экземпляров обработки FPGA и/или ЦПУ следующего порядка, где могут быть сформированы дальнейшие данные результатов, и опять после формирования данные результатов могут быть направлены обратно на прежний или вперед на следующий уровень EDS для хранения перед продвижением на следующую стадию обработки.

[00662] В частности, в одной конкретной реализации поток через конвейер может выглядеть следующим образом: ЦПУ (например, 4 ядра ЦПУ, или экземпляр C4): данные подготовлены (поставлены в очередь и/или сохранены); FPGA (например, 2XL FPGA - 1/8 всего сервера, или экземпляр F1): Картирование, временное хранение; FPGA (например, 2XL FPGA - 1/8 всего сервера, или экземпляр F1): выравнивание, временное хранение; ЦПУ: сортировка, временное хранение; ЦПУ: удаление дубликатов, временное хранение; ЦПУ: определение вариантов 1, временное хранение; FPGA (например, F1 или 16XL, или экземпляр F2): Смит-Ватерман, временное хранение; FPGA (например, экземпляр F1 или F2): HMM, временное хранение; ЦПУ: определение вариантов 2, временное хранение; ЦПУ: VCGF, временное хранение и т.д. Кроме того, может быть включена система управления рабочими потоками для управления потоком данных или направления его через систему, например, когда WMS может быть реализована в ядре ЦПУ, например 4-ядерное ЦПУ, или экземпляр C4. Следует отметить, что один или более из этих этапов могут быть выполнены в любом логическом порядке и могут быть реализованы любым подходящим образом сконфигурированным ресурсом, например, реализованы в программном и/или аппаратном обеспечении во всевозможных разных сочетаниях. И необходимо отметить, что любая из этих операций может быть выполнена в одном или более экземплярах ЦПУ и одном или более экземплярах FPGA на одном или более теоретических уровнях обработки, например, для формирования BioIT-обработки, описанной в настоящем документе.

[00663] Как было указано, может быть включен диспетчер рабочих потоков, например, когда WMS реализован в одном или более ядрах ЦПУ. Следовательно, в различных случаях WMS может иметь базу данных, связанную с ней в процессе работы. В таком случае база данных содержит различные операции и задания, которые нужно поставить в очередь, ожидающие задания, а также историю всех заданий, предыдущих или подлежащих выполнению в настоящее время. Поэтому WMS контролирует систему и базу данных с целью выявления любых новых заданий, подлежащих выполнению. Следовательно, при выявлении ожидающих заданий WMS инициирует новый протокол анализа на данных и отправляет их на соответствующих узел экземпляра. Соответственно, диспетчер рабочих потоков отслеживает и знает, где находятся все входные файлы, или хранятся, обрабатываются, или должны быть сохранены, и поэтому направляет и дает команды экземплярам различных узлов обработки на получение доступа к соответствующим файлам в данном месте, на начало считывания файлов, на начало реализации инструкций по обработке и куда записывать данные результатов. И, следовательно, WMS руководит системами в отношении передачи данных на

последующие узлы обработки. WMS также определяет, где нужно запустить новые экземпляры и привести в действие, чтобы обеспечить динамическое масштабирование каждого этапа или уровня обработки. В результате WMS выявляет, организуется и направляет дискретные задания, которые должны быть выполнены на каждом уровне и далее направляет данные результатов, записываемые в память для хранения, а по завершении задания запускает другой узел, считывает следующее задание и выполняет следующую итеративную операцию.

[00664] Подобным образом входные задания могут распространяться по множеству различных экземпляров, которые могут быть масштабированы, например, независимо или коллективно, путем включения меньше или все больше экземпляров. Эти экземпляры могут быть использованы для создания узлов, чтобы эффективнее балансировать использование ресурсов, когда такие экземпляры могут представлять собой частичный или полный ресурс. Диспетчер рабочих потоков может также руководить и управлять использованием одной или более памятей, например, между этапами обработки, описанными в настоящем документе. Различные экземпляры могут также включать в себя комплементарное программирование, чтобы они могли обмениваться данными друг с другом и/или различными памятями для виртуализации сервера. WMS может также включать в себя также оценщик загрузки для эластичного управления использованием узлов.

[00665] Кроме того, что касается использования памятей, одно или более EBDS или другие соответствующим образом сконфигурированные устройства хранения данных и/или файлов могут быть присоединены к одному или более из различных узлов, например, между различными уровнями экземпляров, например, для временного хранения между всевозможными разными этапами обработки. Следовательно, устройство хранения может быть одним устройством хранения, выполненным с возможностью связывания со всеми различным экземплярами, например, блоком эффективной памяти, таким как эластичное хранилище файлов, или может быть множеством устройств хранения, например по одному устройству на тип экземпляра, которое выполнено с возможностью переключения между экземплярами, например, эластичное блочное устройство хранения. Соответственно, подобным образом каждый уровень экземпляров обработки и/или память могут быть эластично масштабированы по мере надобности, например, между каждым из разных узлов или уровней узлов, например, для обработки одного или нескольких геномов.

[00666] Виду архитектуры, описанной в настоящем документе, один или множество геномов могут быть введены в систему для обработки, например, из одной или более полос проточной кюветы секвенатора нового поколения, как показано на ФИГ. 1. А именно, обеспечение облачной серверной системы 300, которая описана в настоящем документе, позволит накапливать и/или выстраивать в очередь на обработку множество заданий, которые могут быть обработаны всевозможными разными экземплярами системы одновременно или последовательно. Следовательно, конвейер может быть выполнен с возможностью поддержки множества заданий, обрабатываемых виртуальной матрицей процессором, которые связаны с соответствующим образом сконфигурированными запоминающими устройствами для облегчения эффективной обработки данных из одного экземпляра в другой. Кроме того, как было указано, может быть предусмотрено простое запоминающее устройство, которое выполнено с возможностью связывания с множеством разных экземпляров, например, одновременно. В других случаях запоминающее устройство может представлять собой запоминающее устройство эластичного типа, которое может быть выполнено с возможностью

связывания с первым экземпляром, например, в одно время, с последующими изменением конфигурации и/или устранения связи с первым экземпляром и переключением на второй экземпляр.

[00667] Поэтому в одной реализации могут быть включены одно или более эластичных блочных устройств хранения и система может быть выполнена с возможностью включения в себя механизма управления переключением. Например, может быть включен контроллер переключения, выполненный с возможностью управления функционированием таких запоминающих устройств по мере переключения их с одного экземпляра на другой. Конфигурация может быть выполнена с возможностью обеспечения перемещения данных по конвейеру специализированных процессоров, тем самым повышая эффективность системы, например, среди всех экземпляров, например, путем пропускания потока данных через систему, позволяя каждому уровню масштабироваться независимо и по мере необходимости приводя в действие для эффективного масштабирования.

[00668] Кроме того, алгоритм системы управления рабочими потоками может быть выполнен с возможностью определения количества заданий, количества ресурсов для обработки этих заданий, порядка обработки, и направляет поток данных из одного узла в другой путем перекидывания или переключения одного или более гибких переключающих устройств, и там, где требуется, может приводить в действие дополнительные ресурсы, чтобы справляться с ростом в рабочем процессе. Необходимо отметить, что данная конфигурация может быть выполнена с возможностью избежания копирования данных из одного экземпляра в следующий, которое является неэффективным и занимает слишком много времени. Вместо этого за счет переключения эластичного хранилища от одного набора экземпляров на другой, например, путем перетаскивания его с одного узла и прикрепления ко второму узлу, можно значительно улучшить эффективность системы. Кроме того, в различных случаях вместо EBSD можно использовать одно или более эластичных устройств хранения файлов, например, одно запоминающее устройство, выполненное с возможностью связывания с множеством экземпляров без необходимости переключения с одного на другой, чтобы еще больше улучшить передачу данным между экземплярами, делая систему еще более эффективной. Кроме того, необходимо отметить, что, как указано ранее в настоящем документе, в другой конфигурации ЦПУ архитектуры могут быть напрямую связаны друг с другом. Аналогичным образом различные FPGA могут быть напрямую связаны вместе. И, как указано выше, ЦПУ могут быть напрямую связаны с FPGA, например, когда связывание осуществляется посредством интерфейса жесткого связывания, как описано выше.

[00669] Соответственно, что касается хранения пользователем сформированных данных результатов и получения доступа к ним, то все сформированные данные результатов не требуется сохранять, причем это относится ко всей системе. Например, сформированные данные результатов обычно будут в конкретном формате файла, например, BCL, FASTQ, SAM, BAM, CRAM, VCF. Однако каждый из этих файлов имеет большой размер, и хранение всех их займет уйму памяти, что чревато огромными расходами. Тем не менее, преимуществом представленных устройств, систем и способов их использования является то, что все эти файлы не нужно хранить. Вместо этого, учитывая высокие скорости обработки и быстрые темпы сжатия и распаковки, достижимые компонентами и способами системы, хранить нужно всего один формат файла, например, сжатый формат файла, например, например, в облачной базе 400 данных. А именно, хранить нужно только один формат файла данных, из которого

реализуя устройства и способы системы, можно получить все остальные форматы файла. И, ввиду быстрых темпов сжатия и распаковки, достигаемых системой, как правило, это сжатый файл, например файл CRAM.

[00670] В частности, как показано на ФИГ. 40A, в одной реализации пользователь локального вычислительного ресурса 100 может выгружать данные, такие как геномные данные, например, файл BCL и/или FASTQ, в систему посредством облака 50 для приема облачным вычислительным ресурсом, например сервером 300. Затем сервер 300 либо временно сохранит данные 400, либо начнет обработку данных в соответствии с запросом на задания от пользователя 100. Во время обработки входных данных вычислительный ресурс 300 будет тем самым формировать данные результатов, например в файле SAM или BAM и/или VCF. После этого система может сохранить один или более из этих файлов или может сжать один или более из этих файлов и сохранить их. Однако, чтобы снизить издержки и эффективнее использовать ресурсы, система может сохранить один, например, сжатый, файл, из которого могут быть сформированы все остальные форматы файла, например, с помощью устройств и способов, описанных в настоящем документе. Соответственно, система выполнена с возможностью формирования файлов данных, например, данных результатов, который может быть сохранен в связанной с сервером 300 базе 400 данных, которая доступна посредством облака 50 способом, эффективным с точки зрения затрат.

[00671] Соответственно, используя локальный вычислительный ресурс 100, пользователь системы может выполнить вход и через облако 50 получить доступ к серверу 300, выгрузить данные на сервер 300 или в базу 400 данных и запросить выполнение одного или более заданий на этих данных. Затем система 300 выполнит запрошенные задания и сохранит данные результатов в базе 400 данных. Как было отмечено, в конкретных случаях система 300 сохранить сформированные данные результатов в одном формате файла, таком как файл CRAM. Кроме того, нажав кнопку, пользователь может получить доступ к сохраненному файлу, и после этого еще одним нажатием кнопки может получить доступ ко всем другим форматам файлов. Например, в соответствии со способами, описанными в настоящем документе, и учитывая возможности системы по быстрой обработке, эти файлы будут затем обработаны и сформированы за кадром, например, на лету, тем самым сокращая и время обработки, и нагрузку, а также издержки на хранение, например, когда функции вычисления и хранения связаны друг с другом.

[00672] В частности, существуют две части этого эффективного и быстрого процесса хранения, которые обеспечены выполнением ускоренных операций, описанных в настоящем документе. Более конкретно, поскольку различные операции обработки картирования, выравнивания, сортировки, удаления дубликатов и/или определения вариантов могут быть реализованы в аппаратной и/или квантовой конфигурации обработки, можно достичь быстрого создания данных результатов в одном или более форматах файлов. Кроме того, благодаря тесно связанным архитектурам, описанным в настоящем документе, достигнуто также беспрепятственное сжатие и сохранение данных результатов, например, в формате файла FASTQ, SAM, BAM, CRAM, VCF.

[00673] Более того, за счет ускоренной обработки, обеспечиваемой устройствами системы, и за счет их беспрепятственной интеграции со связанными устройствами хранения, данные, получаемые в результате операций обработки системы, которые подлежат сохранению, могут быть эффективно сжаты перед сохранением и распакованы после хранения. Ввиду такой эффективности снижаются расходы на хранение и/или потери, связанные с распаковкой файлов перед использованием. Соответственно,

благодаря этим преимуществам система может быть выполнена с возможностью обеспечения беспрепятственного сжатия и сохранения только одно типа файла с восстановлением на лету любого другого типа файла по мере надобности или требованию пользователя. Например, можно сохранить файл BAM, или связанный с ним сжатый файл SAM или CRAM, и из этого файла могут быть сформированы остальные, например, в прямом или обратном направлении, например, чтобы воспроизвести файл VCF или файл FASTQ, или файл BCL, соответственно.

[00674] Например, в одном варианте реализации в систему может быть первоначально введен или иным образом сформирован в ней файл FASTQ и затем сохранен. В таком случае при переходе в прямом направлении можно получить контрольную сумму файла. Аналогичным образом после получения результирующих данных при переходе назад можно сформировать другую контрольную сумму. Затем эти контрольные суммы можно использовать для гарантии того, что любые дальнейшие форматы файлов, подлежащие формированию и/или воссозданию системой в прямом и обратном направлении, будут в точности совпадать с еще одним и/или его сжатым форматом файла. Подобным образом можно убедиться, что все необходимые данные сохранены максимально эффективным образом, и WMS знает точно, где эти данные сохранены, в каком формате файла, каким был исходный формат файла, и из этих данных система может восстановить любой формат файла идентичным образом, переходя вперед или назад между форматами файлов (после того, как первоначально сформирован шаблон).

[00675] Следовательно, преимущество в скорости компиляции «точно во время» обеспечивается частично за счет аппаратной или квантовой реализации формирования соответствующих файлов, например, при формировании файла BAM из ранее сформированного файла FASTQ. В частности, сжатые файлы BAM, в том числе файлы SAM и CRAM, как правило, не хранят в базе данных в связи с увеличением времени за счет распаковки сжатого сохраненного файла перед обработкой. Однако, система JIT позволяет делать это без существенных потерь. Более конкретно, благодаря реализации устройств и процессов, описанных в настоящем документе, можно не только быстро сжимать и распаковывать сформированные данные последовательности, например, почти мгновенно, но и можно эффективно хранить их. Кроме того, из сохраненного файла, в каком бы формате он ни был сохранен, можно восстановить любой другой формат файла в считанные секунды.

[00676] Следовательно, как показано на ФИГ. 40C, когда платформы с аппаратным или квантовым ускорением выполняют различные процедуры вторичной обработки, такие как картирование и выравнивание, сортировка, удаление дубликатов и определение вариантов, может быть выполнен дополнительный этап сжатия, например, в виде процесса «все в одном», перед сохранением в сжатой форме. После этого, когда пользователю требуется проанализировать или иным образом использовать сжатые данные, файл может быть извлечен, распакован и/или преобразован из одного формата файла в другой, и/или проанализирован, например, с помощью движков JIT, загружаемых в жестко смонтированный процессор или сконфигурированный в квантовом процессоре, и применения к сжатому файлу одной или более процедур конвейера JIT.

[00677] Соответственно, в различных случаях, когда система содержит связанную FPGA, можно полностью или частично переконфигурировать FPGA и/или организовать движок квантовой обработки, чтобы выполнить процедуру JIT. В частности, модуль JIT может быть загружен в систему и/или выполнен в виде одного или более движков, которые могут включать в себя один или более движком 150 сжатия, которые выполнены

с возможностью работы в фоновом режиме. Следовательно, при вызове данного формата файла система типа JIT может выполнить необходимые операции на запрошенных данных, чтобы создать файл в запрошенном формате. Эти операции могут включать в себя сжатие и/или распаковку, а также преобразование для получения запрошенных данных в опознанном формате файла.

[00678] Например, при формировании генетических данных их обычно создают в формате необработанных данных, таком как файл BCL, который затем может быть преобразован в файл FASTQ, например, секвенатором нового поколения, который формирует данные. Однако, с помощью представленной системы файлы необработанных данных, такие как BCL или другой формат файла необработанных данных, может быть передан в потоковом режиме или иным образом в модуль JIT, где он затем может быть преобразован в данные в файле формата FASTQ и/или другой формат файла. Например, после формирования файла FASTQ он может быть обработан, как описано в настоящем документе, и может быть сформирован соответствующий файл BAM. И, аналогичным образом, из файла BAM может быть сформирован соответствующий файл VCF. Кроме того, во время соответствующих этапов могут быть также сформированы файлы SAM и CRAM. Каждый из этих этапов может быть выполнен очень быстро, особенно если соответствующий формат файла уже был сформирован однажды. Следовательно, после получения файла BCL, например, прямо из секвенатора, файл BCL может быть преобразован в файл FASTQ или непосредственно преобразован в файл SAM, BAM, CRAM и/или VCF, например, с помощью реализованной аппаратным или квантовым способом процедуры картирования/выравнивания/сортировки/определения вариантов.

[00679] Например, в одной модели использования в типичном приборе для секвенирования на отдельную полосу может быть загружено большое количество различных геномов субъекта для параллельной обработки одним прибором для секвенирования. Поэтому по завершении анализа формируется, в виде мультиплексного комплекса, большое количество различных файлов BCL, полученных из всех различных полос и представляющих полные геномы каждого из разных субъектов. Соответственно, эти мультиплексированные файлы BCL можно затем демультиплексировать и сформировать соответствующие файлы FASTQ, представляющие генетический код для каждого отдельного субъекта. Например, если за один прогон секвенирования формируют N файлов BCL, эти файлу нужно будет демультиплексировать, разделить на слои и сшить вместе для каждого субъекта. Это сшивание является сложным процессом, где каждый генетический материал субъекта преобразуют в файлы BCL, которые могут быть затем преобразованы в файлы FASTQ или использованы непосредственно для картирования, выравнивания и/или сортировки, определения вариантов и т.п. Этот процесс может быть автоматизирован, чтобы значительно ускорить его различные этапы.

[00680] Кроме того, как показано на ФИГ. 40A, после того, как эти данные сформированы 110, и, следовательно, должны быть сохранены, например, независимо от того, какой формат выбран, они могут быть сохранены в защищенной паролем и/или шифрованием кэш-памяти, например в специализированной геномной памяти 400 типа «Dropbox». Соответственно, по мере того, как сформированные и/или обработанные генетические данные покидают секвенатор, они могут быть обработаны и/или сохранены и сделаны доступными другим пользователям в других системах, например, в кэш-памяти 400 типа «Dropbox». В таком случае автоматическая конвейерная система биоинформационного анализа может затем получить доступ к данным в кэше и

автоматически начать их обработку. Например, система может включать в себя систему управления, например, систему 151 управления рабочими потоками, имеющую контроллер, такой как микропроцессор или другая интеллектуальная система, например, искусственный интеллект, который управляет извлечением файлов BCL и/или FASTQ, например, из кэш-памяти, и затем направляет обработку этой информации на формирование файла BAM, CRAM, SAM и/или VCF, тем самым автоматически формируя и выводя различные результаты обработки и/или сортируя их в памяти 400 типа «Dropbox».

[00681] Уникальное преимущество обработки JIT, которая реализуется в данной модели использования, состоит в том, что JIT позволяет сжимать различные созданные генетические файла, например, перед сохранением данных, и быстро распаковывать перед использованием. Следовательно, JIT может компилировать и/или сжимать и/или сохранять данные по мере их выхода из секвенатора, причем сохраняет их в защищенной геномной кэш-памяти типа «Dropbox». Эта защищенная геномная кэш-память 400 типа «Dropbox» может представлять собой доступную из облака 50 кэш-память, которая выполнена с возможностью хранения геномных данных из одного или более автоматизированных секвенаторов 110, например, когда секвенаторы расположены удаленно от кэш-памяти 400.

[00682] В частности, после того, как данные последовательности сформированы 110, например, с помощью СНП, они могут быть сжаты 150 для передачи и/или сохранения 400, чтобы сократить количество данных, выгружаемых на облако 50 и сохраняемых там. Такие выгрузка, передача и сохранение могут быть выполнены быстро за счет сжатия 150 данных, которое происходит в системе, например перед передачей. Кроме того, после выгрузки и сохранения в облачной кэш-памяти 400 данные могут быть затем извлечены, локально 100 или удаленно 300, для обработки в соответствии устройствами, системами и способами BioIT-конвейера, описанными в настоящем документе, с целью создания файла картирования, выравнивания, сортировки и/или определения вариантов, такого как файл SAM, BAM и/или CRAM, который может быть затем сохранен вместе с метафайлом, который указывает информацию о том, как был создан сформированный файл, например SAM, BAM, CRAM и т.д.

[00683] Следовательно, в совокупности с метаданными сжатый файл SAM, BAM и/или CRAM может быть затем обработан для получения других форматов файлов, таких как файлы FASTQ и/или VCF. Соответственно, как отмечалось, выше, с помощью JIT можно на лету восстанавливать файл FASTQ или VCF из сжатого файла BAM и наоборот. Файл BCL тоже может быть восстановлен аналогичным образом. Необходимо отметить, что, что файлы SAM и CRAM могут быть подобным образом сжаты и/или сохранены и использоваться для создания одного или более других форматов файлов. Например, файл CRAM, который может быть преобразован обратно из формата CRAM, может быть использован для создания файла определения вариантов и, аналогичным образом для создания файла SAM. Следовательно, нужно сохранить файл SAM, BAM и/или CRAM, из этих файлов могут быть воспроизведены другие форматы файлов, например, VCF, FASTQ, BCL.

[00684] Соответственно, как показано на ФИГ. 40A, прибор 110 для картирования, и/или выравнивания, и/или сортировки, и/или определения вариантов, например, компьютер автоматизированного рабочего места, может находиться в месте эксплуатации 100, и/или второй соответствующий прибор 300 может быть расположен удаленно и быть доступным из облака 50. Данная конфигурация вместе с устройствами и способами, описанными в настоящем документе, выполнены с возможностью

обеспечения пользователю быстрого выполнения BioIT-анализа «на облаке», как описано в настоящем документе, чтобы получить данные результатов. Данные результатов могут быть затем обработаны для сжатия, и после сжатия они могут быть сконфигурированы для передачи, например, обратно на локальный вычислительный ресурс 100 или могут быть сохранены на облаке 400 и посредством облачного интерфейса сделаны доступными для локального вычислительного ресурса 100. В таком случае сжатые данные могут быть файлом SAM, BAM, CRAM и/или VCF.

[00685] В частности, второй вычислительный ресурс 300 может быть другим решением для автоматизированного рабочего места или он может быть серверным конфигурируемым ресурсом, например, когда вычислительный ресурс доступен из посредством облака 50 и выполнен с возможностью осуществления картирования, и/или выравнивания, и/или сортировки, и/или определения вариантов. В таком случае пользователь может запрашивать на облачном сервере 300 выполнение BioIT-заданий на выгруженных данных, например данных в формате BCL и/или FASTQ. В этом случае сервер 300 затем получит доступ к сохраненным и/или сжатым файлам, чтобы быстро обработать эти данные и сформировать одни или более данные результатов, которые могут быть после этого сжаты и/или сохранены. Кроме того, из файла данных результатов могут быть сформированы один или более других файлов BCL, FASTQ, SAM, BAM, VCF или файлы других форматов на лету с помощью обработки JIT. Тем самым данная конфигурация расширяет типичное «узкое место» скорости передачи.

[00686] Поэтому в различных вариантах реализации система 1 может содержать первый прибор 100 картирования, и/или выравнивания, и/или сортировки, и/или определения вариантов, который может быть расположен локально 100, например, для локального получения данных, сжатия 150 и/или хранения 200; а второй прибор 300 может находиться удаленно и быть связан с облаком 50, при этом второй прибор 300 выполнен с возможностью приема сформированных и сжатых данных и сохранения их, например посредством связанного устройства 400 хранения. После сохранения данные могут быть доступны для распаковки и преобразования сохраненных файлов в один или более других форматов файлов.

[00687] Поэтому в одной реализации системы данные. например, необработанные данные последовательности, например, в формате файла BCL или FASTQ, которые сформированы устройством формирования данных, например, секвенатором 110, могут быть выгружены и сохранены на облаке 50, например, в связанной геномной кэш-памяти 400 типа «Dropbox». Эти данные могут быть затем непосредственно доступны для первого прибора 100 картирования, и/или выравнивания, и/или сортировки, и/или определения вариантов, как описано в настоящем документе, или могут быть доступны опосредованно с помощью серверного ресурса 300, которые могут после этого обработать данные последовательности для получения картированных, выровненных, сортированных и/или подвергнутых определению вариантов данных результатов.

[00688] Соответственно, в различных вариантах реализации одно или более устройств хранения, описанных в настоящем документе, могут быть выполнены с возможностью предоставления к себе доступа, при наличии надлежащих полномочий, через облако. Например, различные данные результатов системы могут быть сжаты и/или сохранены в памяти или другой соответствующим образом сконфигурированной базе данных, где база данных выполнена в виде геномного кэша 400 типа «Dropbox», например, где различные данные результатов могут быть сохранены в файле SAM, BAM, CRAM и/или VCF, который может быть доступен удаленно. В частности, необходимо отметить, что, как показано на ФИГ. 40A, может быть предусмотрен локальный прибор 100,

причем этот локальный прибор может быть связан с сами прибором 110 для секвенирования или может быть удален от него, но связан с прибором 110 для секвенирования посредством локального облака 30, и локальный прибор 100 может быть также связан с локальным хранилищем 200 или удаленной кэш-памятью 400, например, когда удаленная кэш-память выполнена в виде хранилища генома типа «Dropbox». Кроме того, в различных случаях второй прибор 300 картирования, и/или выравнивания, и/или сортировки, и/или определения вариантов, например, облачный прибор, при наличии надлежащих полномочий может быть соединен с хранилищем 400 генома типа «Dropbox» для получения доступа к файлам, например, сжатым файлам, сохраненным там локальным вычислительным ресурсом 100, и может затем распаковать эти файлы, чтобы сделать результаты доступными для дальнейшей, например, вторичной или третичной, обработки.

[00689] Соответственно, в различных случаях система может быть рационализирована так, чтобы по мере формирования данных и выхода их из секвенатора 110, например, в формате необработанных данных, они могли быть либо немедленно выгружены на облако 50 и сохранены в хранилище 400 генома типа «Dropbox», либо переданы в систему 300 BioIT-обработки для дальнейшей обработки и/или сжатия перед началом выгрузки и сохранения 400. После сохранения в кэш-памяти 400 система может сразу же поставить данные в очередь на извлечение, сжатие, распаковку и/или дальнейшую обработку, например, с помощью другого связанного устройства 300 BioIT-обработки, которые после обработки которых в данные результатов могут быть сжаты и/или сохранены 400 для использования позже. В этом момент может быть инициирован конвейер третичной обработки, и тем самым сохраненные данные результатов из вторичной обработки могут быть распакованы и использованы, например, для третичного анализа, в соответствии со способами, описанными в настоящем документе.

[00690] Поэтому в различных вариантах реализации система может быть конвейеризована таким образом, что все данные, которые выходят из секвенатора 110, могут быть либо сжаты, например, локальным вычислительным ресурсом 100, перед передачей и/или сохранением 200, либо данные могут быть переданы непосредственно в папку генома «Dropbox» для хранения 400. После приема там сохраненные данные могут быть затем по существу немедленно поставлены в очередь на извлечение и сжатие и/или распаковку, например удаленным вычислительным ресурсом 300. После распаковки данные могут быть по существу немедленно стать доступными для обработки, такой как картирование, выравнивание, сортировка и/или определение вариантов для создания обработанных данных результатов, которые могут быть затем снова сжаты для хранения. После этого сжатые данные вторичных результатов могут быть сделаны доступными, например, в хранилище 400 генома типа «Dropbox», распакованы и/или использованы в одной или более процедур третичного анализа. Поскольку данные могут быть сжаты при хранении и по существу немедленно распакованы при извлечении, они доступны для использования множеством разных систем и в множестве разных биоаналитических протоколов в разное время - просто нужно получить доступ к ним в кэше хранилища 400 типа «Dropbox».

[00691] Следовательно, подобным образом конвейеры BioIT-платформы, представленные в настоящем документе, могут быть выполнены с возможностью обеспечения невероятной гибкости формирования и/или анализа данных и выполнены с возможностью приема вводимых в конкретных формах генетических данных во множестве форматов для обработки с целью создания выходных форматов, которые совместимы с различным последующим анализом. Соответственно, как показано на

ФИГ. 40C, в настоящем документе предложены устройства, системы и способы для выполнения анализа генетического секвенирования, который может включать в себя один или более из следующих этапов: Во-первых, принимают входной файл, который может быть в одном или более форматах файла FASTQ или BCL или других форматах файла генетической последовательности, таком как сжатый формат файла, после чего этот файл может быть распакован и/или обработан посредством ряда этапов, описанных в настоящем документе, для формирования файла VCF/gVCF, который может быть затем сжат, и/или обработан, и/или передан. Такие сжатие и распаковка могут происходить на любой подходящей стадии по всему процессу.

[00692] Например, после приема файла BCL он может быть подвергнут обработке в конвейере анализа, например, последовательно, как описано в настоящем документе. Например, после приема файл BCL может быть преобразован и/или демультиплексирован, например, в формат файла FASTQ и/или FASTQgz, который может быть затем отправлен в модуль картирования и/или выравнивания, например, сервера 300, чтобы быть картированным и/или выровненным в соответствии с устройством и его способами использования, описанными в настоящем документе. Кроме того, в различных вариантах картированные и выровненные данные, например, в формате файла SAM или BAM, могут быть отсортированы по позиции и/или в них могут быть маркированы и удалены дубликаты. Затем файлы могут быть преобразованы, например, с получением файла CRAM, для передачи и/или хранения, или могут быть пересланы в модуль определения вариантов, например, HMM, для обработки с целью получения файла определения вариантов, VCF или gVCF.

[00693] Точнее говоря, как показано на ФИГ. 40C и 40D, в определенных случаях файл, предназначенный для приема системой, может быть передан в потоковом режиме или иным образом перенесен в систему прямо из устройства секвенирования, например, СНП 110, и поэтому переданный файл может быть в формате файла BCL. Когда принимаемый файл представлен в формате файла BCL, он может быть преобразован и/или иным образом демультиплексирован в файл FASTQ для обработки системой, или файл BCL может быть обработан непосредственно. Например, процессоры конвейера платформы могут быть выполнены с возможностью приема данных BCL, которые передаются в потоковом режиме прямо из секвенатора, как описано со ссылкой на ФИГ. 1, или они могут принимать данные в формате файла FASTQ. Однако, прием данных последовательности сразу по выходе из секвенатора полезен, поскольку он позволяет переводить данные прямо из необработанных данных секвенирования в непосредственно обрабатываемые данные, например, в один или более из форматов SAM, BAM и/или VCF/gVCF, для вывода.

[00694] Соответственно, после приема файла BCL и/или FASTQ, например, вычислительным ресурсом 100 и/или 300, он может быть картирован и/или выровнен вычислительным ресурсом, причем картирование и/или выравнивание могут быть выполнены на ридах с одинарными или спаренными концами. Например, после приема данные последовательности могут быть компилированы в риды для анализа, например, с длинами рида, которые могут меняться в диапазоне от около 10 или около 20, например, 26, или 50, или 100, или 150 пар оснований или меньше, до около 1K, или около 2,5K, или около 5K, даже около10K пар оснований или больше. Аналогичным образом после картирования и/или выравнивания последовательность может быть затем сохранена, например, отсортированной по позиции, например, посредством распределения по группа в соответствии с референсным диапазоном и сортировки групп по позициям референса. Кроме того, секвенированные данные могут быть

обработаны посредством маркировки дубликатов, например, на основе начальной позиции и строки CIGAR, для формирования высококачественного отчета о дубликатах, и все маркированные дубликаты могут быть удалены на этом этапе. В результате может быть сформирован картированный и выровненный файл SAM, который может быть сжат для формирования файла BAM/CRAM, например, для хранения и/или дальнейшей обработки. Кроме того, после того, как файл BAM/CRAM извлечен, картированные и/или выровненные данные последовательности могут быть направлены в системный модуль определения вариантов, такой как определитель вариантов гаплотипов с повторной сборкой, который, в некоторых случаях, может использовать одно или более из выравнивания Смита-Ватермана и/или скрытой марковской модели, которые могут быть реализованы в комбинации программного и/или аппаратного оборудования, для формирования файла VCF.

[00695] Поэтому, как показано на ФИГ. 40D, система и/или один или более из ее компонентов выполнены с возможностью преобразования данных BCL в форматы данных FASTQ или SAM/BAM/CRAM, которые могут быть затем отправлены по всей системе для дальнейшей обработки и/или реконструкции данных. Например, после того, как данные BCL приняты и/или преобразованы в файл FASTQ и демультиплексированы и/или освобождены от дубликатов, они могут быть отправлены в один или более модулей конвейера, описанных в настоящем документе, например, для картирования и/или выравнивания, что в зависимости от количества обрабатываемых образцов приведет к созданию одного или более, например, нескольких, файлов SAM/BAM. Эти файлы могут быть затем отсортированы, освобождены от дубликатов и направлены в модуль определения вариантов для получения одного или более файлов VCF. Эти этапы могут быть повторены для получения больше контекста и улучшения точности. Например, после того, как данные последовательности картированы или выровнены, например, с получением файла SAM, файл SAM может быть затем сжат в один или более файлов BAM, которые после этого могут быть переданы в движок VCF для преобразования посредством обработки системой в файл VCF/gVCF, который может быть сжат в файл CRAM. В результате прохождения по всей системе на выходе может быть получен файл Gzip и/или CRAM.

[00696] В частности, как показано на ФИГ. 40C и 40D, один или более из сформированных файлов может быть сжат и/или передан из одного компонента системы в другой, например, из локального ресурса 100 в удаленный ресурс 300, и после приема может быть распакован, например если он ранее был сжат, или преобразованы/демультиплексирован. Более конкретно, после приема файла BCL либо локальным ресурсом 100, либо удаленным ресурсом 300, он может быть преобразован в файл FASTQ, который может быть затем обработан интегральными схемами системы с целью картирования и/или выравнивания, или может быть передан на удаленный ресурс 300 для такой обработки. После картирования и/или выравнивания полученные в результате данные последовательности, например, в формате файла SAM, могут пройти дальнейшую обработку, такую как сжатие один или более раз, например, в файл BAM/CRAM, после чего эти данные могут быть обработаны путем сортировки по позиции, маркировки дубликатов и/или определения вариантов, результаты чего, например в формате VCF, могут быть затем сжаты еще раз, и/или сохранены, и/или переданы, например, из удаленного ресурса 300 на локальный ресурс 100.

[00697] Более конкретно, система может быть выполнена с возможностью обработки данных BCL напрямую, тем самым избавляя от этапа преобразования в файл FASTQ. Аналогичным образом данные BCL могут быть поданы прямо в конвейер для создания

уникального выходного файл VCF для образца. Промежуточные файлы SAM/BAM/CRAM тоже могут быть сформированы по требованию. Следовательно, система может быть выполнена с возможностью приема и/или передачи одного или более файлов данных, таких как BCL или FASTQ, содержащих информацию о последовательности, и обработки их для создания сжатого файла данных, такого как файл данных SAM/BAM/CRAM.

[00698] Соответственно, как показано на ФИГ. 41A, пользователь может по желанию получить сжатый файл и преобразовать его в исходную версию сформированного файла BCL 111c и/или FASTQ 111d, например, чтобы подвергнуть данные дальнейшей, например, более совершенной, обработке 111b сигнала, например, для исправления ошибок. В альтернативном варианте реализации пользователь может получать доступ к необработанным данным последовательности, например в формате файла BCL или FASTQ 111, и подвергать эти данные дальнейшей обработке, такой как картирование 112, и/или выравнивание 113, и/или другие связанные функции 114/115. Например, данные результатов этих процедур могут быть затем сжаты, и/или сохранены, и/или подвергнуты дальнейшей обработке 114, такой как сортировка 114a, удаление дубликатов 114b, перекалибровка 114c, локальное повторное выравнивание 114d и/или сжатие/распаковка 114e. Тот же или другой пользователь может после этого захотеть получить доступ к сжатой форме картированных и/или выровненных данных результатов и затем выполнить на них другой анализ, например чтобы выполнить одно или более из определений вариантов 115, например, посредством HMM, алгоритма Смита-Ватермана, преобразования и т.д., а результаты затем сжать и/или сохранить. Еще один пользователь системы может затем получить доступ к сжатому файлу VCF 116, распаковать его и подвергнуть данные одному или более протоколам третичной обработки.

[00699] Далее, пользователь, возможно, захочет выполнить конвейерное сравнение. Картирование/выравнивание/сортировка/определение вариантов полезны для выполнения различного геномного анализа. Например, если впоследствии требуется дальнейший анализ ДНК или РНК или анализа несколько иного рода, пользователь может по желанию прогнать данные через другой конвейер и, следовательно, наличие доступа к восстановленному исходному файлу данных очень полезно. Аналогичным образом этот процесс может быть полезным, например, когда может потребоваться создать или воссоздать файл SAM/BAM/CRAM, например, когда сформирован новый или другой референсный геном и, следовательно, может потребоваться повторить картирование или выравнивание на новый референсный геном.

[00700] Хранение сжатых файлов SAM/BAM/CRAM тоже полезно, так как оно позволят пользователю системы 1 использовать с выгодой тот факт, что референсный геном образует остов данных результатов. В таком случае важны не данные, которые согласуются с референсом, а, скорее, как данные не согласуются с референсом. Следовательно, важно сохранить только данные, которые не согласуются с референсом. Следовательно, система 1 может извлечь пользу из этого факта сохранения только того, что важно и/или полезно для пользователей системы. Таким образом, весь геномный файл (показывающие соответствие и несоответствие с референсом) или его часть (показывающая только соответствие или несоответствие с референсом) может быть выполнена с возможностью сжатия и сохранения. Поэтому можно заметить, что поскольку для исследования наиболее полезны только отличия и/или вариации между референсом и исследуемым геномом, в различных вариантах реализации нужно сохранять только эти отличия, так как все, что совпадает с референсом, не нужно

рассматривать заново. Соответственно, так как любой данный геном слабо отличается от референса, например, 99% человеческих геномов, как правило, идентичны, то после создания файла BAM рассматривать и/или сохранять нужно только вариации относительно референсного генома.

[00701] Соответственно, как показано на ФИГ. 41B, другим полезным компонентом доступной из облака системы 1, предложенной в настоящем документе, является контроллер 151 управления рабочими потоками, который может быть использован для автоматизации потока системы. Такая автоматизация системы может включать в себя использование различных компонентов системы для доступа к данным, либо локально 100, либо удаленно 300, когда и/или где они становятся доступными с последующим по существу автоматическим применением к этим данным дальнейших этапов обработки, например, относящихся к BioIT-конвейерам, описанным в настоящем документе. Соответственно, контроллер 151 управления рабочими потоками является основополагающей технологией автоматизации для руководства различными конвейерами системы, например, 111, 112, 113, 114 и/или 115, и в различных случаях может использовать компонент 121a искусственного интеллекта.

[00702] Например, система 1 может содержать модуль искусственного интеллекта (ИИ), который выполнен с возможностью анализа различных данных системы и в ответ на них передает свои результаты с помощью системы 151 управления рабочими потоками. В частности, в различных случаях модуль ИИ может быть выполнен с возможностью анализа различных геномных данных, представленных в системе, а также данных результатов, которые формируются обработкой данных, для выявления и определения различных взаимосвязей между этими данными и/или любыми другими данными, которые могут быть введены в систему. Более конкретно, модуль ИИ может быть выполнен с возможностью анализа различных геномных данных в соответствии с множеством других факторов с целью определения любой взаимосвязи, например, причинно-следственной связи, между различными факторами, например, точками данных, которые могут быть информативными в отношении воздействий рассматриваемых факторов на определенные геномные данные, например, вариацию данных, и наоборот.

[00703] А именно, как описано более подробно ниже в настоящем документе, модуль ИИ может быть выполнен с возможностью коррелирования геномных данных субъекта, формируемых системой, с любыми электронными медицинскими записями для данного субъекта или других людей, с целью определения взаимосвязей между ними и/или другими уместными факторами и/или данными. Соответственно, в число таких других данных, которые могут быть использованы при определении любых уместных воздействий и/или взаимосвязей, которые могут иметь эти факторы в отношении субъекта и/или его геномных данных и/или здоровья, входят: данные НИПТ, данные ОРИТН, относящиеся к раку данные, данные LDT, относящиеся к окружающей среде и/или аграрно-биологические данные и/или другие такие данные. Например, дальнейшие данные для анализа могут быть получены с помощью таких других факторов, как данные об окружающей среде, данные клады, данные микробиома, данные метилирования, структурные данные, например, данные химерного или сопряженного рида, данные генеративных вариантов, данные аллелей, данные РНК и другие такие данные, относящиеся к генетическим материалам субъекта. Следовательно, модуль ИИ может использоваться для связывания различных относящихся к делу данных, протекающих через систему, с вариантами, определенными в геноме одного или более субъектов, наряду с другими возможными связанными последствиями, основанными

на факторах.

[00704] В частности, движок ИИ может быть выполнен с возможностью выполнения в ЦПУ/ГПУ/КПУ и/или он может быть выполнен с возможностью выполнения в виде ускоренного движка ИИ, который может быть реализован в FPGA и/или квантовом процессорном устройстве. А именно, движок ИИ может быть связан с одной или более, например, всеми, различными базами данных системы, чтобы движок ИИ мог использовать и обрабатывать различные данные, протекающие через систему. Кроме того, когда субъект, геном которого обрабатывают, дает соответствующее разрешение на доступ к геномным и данным и истории болезни, система выполнена с возможностью коррелирования различных наборов данных друг с другом и может затем извлекать данные для определения различных значимых соответствий, ассоциаций и/или взаимосвязей.

[00705] Точнее говоря, модуль ИИ может быть выполнен с возможностью реализации протокола машинного обучения применительно к входным данным. Например, геномные данные множества субъектов, которые формируют на основе выполняемых анализов, описанных в настоящем документе, могут быть сохранены в базе данных. Аналогичным образом при наличии соответствующих разрешений и подтверждений подлинности можно получить доступ к электронным медицинским записям/электронному учету здоровья (EMR) субъекта, геномную ДНК которого обрабатывают, и можно подобным образом сохранить в базе данных. Как описано более подробно ниже, движки обработки могут быть выполнены с возможностью анализа геномных данных субъектов, а также данных их EMR, с целью определения любых корреляций между ними. Эти корреляции будут затем использованы для подтверждения наблюдаемых взаимосвязей, а их результаты могут быть использованы для более эффективного и более успешного выполнения различных функций системы.

[00706] Например, движок обработки ИИ может получать доступ к геномным данным субъектов, коррелировать их с известными болезнями и состояниями этих субъектов и на основе данного анализа модуль ИИ может учиться выполнять прогнозные корреляции, опираясь на эти данные, чтобы повышать свои возможности предсказания наличия болезни и/или других подобных состояний у других индивидов. В частности, определяя такие корреляции между геномами других людей с их EMR, например, в отношении наличия маркеров болезней, модуль ИИ может научиться выявлять такие корреляции, например, определяемые системой маркеры болезней, в геномах других людей, тем самым приобретая способность предсказывать возможность болезни или других поддающихся опознанию состояний. Более конкретно, анализируя геном субъекта в сравнении с известными или определенными генетическими маркерами, и/или путем определения вариаций в геноме субъекта, и/или, дополнительно, путем определения потенциальной взаимосвязи между геномными данными и состоянием здоровья субъекта, например, EMR, модуль ИИ может быть в состоянии делать выводы не только в отношении изучаемого субъекта, но и других людей, образцы которых будут брать в будущем. Это можно делать, например, на систематической основе, для каждого субъекта отдельно, или в пределах популяций и/или в разных географических районах.

[00707] Более конкретно, применительно к представленным системам, создают скопление ридов. Скопление может перекрывать области, о которых известно, что в них более высокая вероятность значительной вариации. Соответственно, с одной стороны, система будет анализировать скопление для определения наличия вариации, и, в то же самое время, исходя из предыдущих результатов, она уже будет знать

правдоподобие присутствия или отсутствия вариации там, например, она будет иметь первоначальный прогноз в отношении того, каким должен быть ответ. Вне зависимости от того, имеется ли там ожидаемая вариация или нет, эта информация будет полезна при анализе данной области геномов других субъектов. Например, это может быть одна точка данных в сумме точек данных, используемых системой для улучшения определений вариантов и/или более хорошего связывания этих вариантов с одним или более болезненных состояний или других состояний здоровья.

[00708] Например, в примере протокола изучения анализ ИИ может включать в себя получение электронного изображения скопления одной или более областей в геноме, например, для тех областей, на которые падает подозрение в кодировании одного или более состояний здоровья, и связывания этого изображения с известными определениями вариаций из других скоплений, например, когда отношение этих вариаций к болезненным состояниями может быть известно или не известно. Это можно проделывать снова и снова, обучая систему обрабатывать информацию, создавать соответствующие ассоциации и выполнять правильные определения вариантов быстрее и быстрее с более высокой точностью. Сделав это для различных, например, всех, известных областей генома, подозреваемых в качестве причины болезни, то же самое можно повторить с остальной частью генома, например, пока не будет изучен весь геном. Аналогичным образом это можно повторять снова и снова для множества образцов геномов раз за разом, чтобы натренировать систему, например, определитель вариантов, выполнять определение вариантов точнее, быстрее и эффективнее, и/или позволить модулю третичной обработки лучше выявлять болезненные состояния.

[00709] Соответственно, система принимает множество входных данных с известными ответами, выполняет анализ и вычисляет ответ, и тем самым обучается в ходе этого процесса, например, воспроизводит изображение скопления, относящегося к одному геному, и затем учится выполнять определение на основе другого генома, быстрее и быстрее, так как ей уже легче определять, что последующие скопления напоминают ранее полученные изображения с известной взаимосвязью с болезненными состояниями. Таким образом, система может быть выполнена с возможностью обучения прогнозированию наличия вариантов, например, на основе распознаваний образов, и/или прогнозированию взаимосвязи между наличием этой вариации с одним или более медицинских состояний.

[00710] Точнее говоря, чем больше система выполняет анализов частичного или полного генома и определяет взаимосвязь между вариациями и различными состояниями, например, в множестве образцов, тем лучше она делает прогнозы, например, на основе частичных или полных изображений генома скоплений. Это полезно при прогнозировании болезненных состояний на основе изображений скопления и/или другого анализа ридов, и может включать в себя создание корреляции между одной или более EMR (содержащими фенотипические данные), изображением скопления и/или известными вариантами (генотипические данные) и/или болезнями и болезненными состояниями, например, на основе которых можно делать прогнозы. В различных случаях система может включать в себя функцию транскрипции, чтобы быть в состоянии транскрибировать любые физические примечания, которые могут быть частью медицинской записи пациента, для включения этих данных в ассоциации.

[00711] В одной модели использования субъект может иметь мобильное средство отслеживания и/или датчик, например, мобильный телефон или другое вычислительное устройство, которое может быть выполнено с возможностью как отслеживания местоположения субъекта, так и восприятия условий окружающей среды и/или

физиологических условий пользователя в этом месте. Возможен сбор и других воспринимаемых данных. Например, мобильное вычислительное устройство может содержать GPS-трекер и/или его местоположение может быть определено с помощью триангуляции вышек сотовой связи, и может быть выполнено с возможностью передачи собранных им данных, например, посредством сотовой связи WIFI, Bluetooth или другого соответствующим образом сконфигурированного протокола связи. Таким образом, мобильное устройство может отслеживать и определять категорию данных окружающей среды, относящихся к географическим местоположениям, условий окружающей среды, физиологического состояния и других воспринимаемых данных субъекта, владеющего мобильным компьютером, с которыми он сталкивается в своей повседневной жизни. Собранные данные о местоположении, окружающей среде, физиологическом состоянии, здоровье и/или другие связанные данные, например, данные ZNA, могут быть затем переданы, например, на регулярной или периодической основе, в оду или более баз данных, описанных в настоящем руководстве, причем собранные данные ZNA могут быть коррелированы с историей болезни субъекта, например, с записями EMR, и/или геномными данными, как определено системой, описанной в настоящем документе.

[00712] Аналогичным образом в различных случаях одни или более из этих данных могут быть направлены с платформы сбора и анализа данных ZNA в центральный репозиторий, например, в правительственном учреждении, для анализа в большем, например, общегосударственном, масштабе, например, в соответствии с искусственным интеллектом, описанным в настоящем документе. Например, база данных, например, управляемая государством база данных, может иметь записанные данные об окружающей среде, с которыми можно сравнить данные об окружающей среде субъекта. Например, в одном иллюстративном случае тест НИПТ может быть выполнен для матери, отца и их ребенка, и затем в течении всех их жизней можно собирать их данные об окружающей среде, геномные данные и данные медицинских записей могут и коррелировать друг с другом и/или одной или более моделями, например, на протяжении жизни индивидов, особенно в отношении возникновения мутаций, например, обусловленных факторами воздействия окружающей среды. Этот сбор данных может осуществляться на протяжении всей жизни индивида и может выполняться на основе семьи в целом, чтобы лучше построить базу данных сбора данных и лучше предсказывать воздействия таких факторов на генетические вариации и наоборот.

[00713] Соответственно, контроллер 151 управления рабочими потоками позволяет системе 1 принимать входные данные из одного или более источников, таких как множество приборов для секвенирования, например, 110a, 110b, 110c и т.д., и множество входных данных из одного прибора 110 секвенирования, где принимаемые данные представляют геномы множества субъектов. В таких случаях контроллер 151 управления рабочими потоками не только отслеживает все входящие данные, но и эффективно организовывает и облегчает вторичную и/или третичную обработку принимаемых данных. Соответственно, контроллер 151 рабочих потоков обеспечивает системе 1 возможность беспрепятственного подключения как небольшим, так и к крупным центрам секвенирования, куда может поступать генетический материал любого рода посредством одного или более аппаратов 110 для секвенирования одновременно, причем все они могут быть переданы в систему 1 посредством облака 50.

[00714] Точнее говоря, как показано на ФИГ. 41A, в различных случаях один или более из множества образцов может быть принят системой 1, и поэтому система 1 может быть выполнена с возможностью приема и эффективной обработки образцов, либо последовательно, либо параллельно, например, в режиме обработки множества

образцов. Соответственно, чтобы рационализировать и/или автоматизировать обработку множества образцов, управление системой может осуществляться всеобъемлющей системой 151 управления рабочими потоками (WMS) или системой управления лабораторной информацией (LIMS) WMS 151 позволяет пользователям без труда планировать выполнение множества рабочих потоков для любого конвейера, а также корректировать или ускорять алгоритмы анализа СНП, конвейеров платформы и их сопровождающих приложений.

[00715] В таком случае каждая обрабатываемая последовательность может иметь штрихкод, указывающий тип последовательности, формат файла и/или какие этапы обработки были выполнены, и какие этапы обработки должны быть выполнены. Например, штрихкод может содержать объявление, указывающее «это обработка генома субъекта X в формате файла Y, поэтому эти данные нужно пропустить через конвейер Z», или, аналогичным образом, может быть указано «это результирующие данные A, которые должны поступить в систему составления отчетов». Соответственно, по мере приема, обработки и передачи через систему данных, штрихкоды и результаты будут загружаться в систему 151 управления рабочими потоками, такую как LIMS (система управления лабораторной информацией). В данном примере LIMS может быть стандартным средством, которое используют для управления лабораториями, или это может быть специально разработанное средство, используемое для управления технологическим процессом.

[00716] В любом случае контроллер 151 управления рабочими потоками отслеживает снабженный штрихкодом образец с момента поступления на данный участок, например, для хранения и/или обработки, до отправки результатов пользователю. В частности, контроллер 151 управления рабочими потоками выполнен с возможностью отслеживания всех данных по мере их прохождения через систему от начала до конца. Более конкретно, при поступлении образца связанный с ним штрикод считывается, и на основе этого считывания система определяет порядок выполнения запрошенной работы и подготавливает образец для обработки. Такая обработка может быть простой, например, выполняется одним геномным конвейером, или она может быть более сложной, например, выполняется множеством конвейеров, например пятью конвейерами, которые должны быть сшиты вместе. В одной конкретной модели формируемые или принимаемые данные можно прогнать через систему, чтобы получить обработанные данных, которые можно затем прогнать через эквивалентный GATK модуль, и результаты сравнить, а затем образец можно передать в другой конвейер для дальнейшей, например, третичной, обработки 700 (см. ФИГ. 41B).

[00717] Следовательно, система в целом может работать в соответствии с несколькими различными конвейерами обработки. Действительно, многие процессы системы могут быть связаны между собой, причем диспетчер 151 рабочих потоков уведомляется или иным образом определяет, что задание ожидает, количественно оценивает матрицы задания, выявляет свободные ресурсы для выполнения требуемого анализа, загружает задание в систему, принимает поступающие данные, например, из секвенатора 110, загружает их и затем обрабатывает. В частности, после того, как рабочий процесс настроен, его можно сохранить, а затем этот рабочий процесс получает измененный штрихкод, и автоматизированный процесс происходит в соответствии с указаниями рабочих потоков.

[00718] До появления настоящей автоматизированной системы 151 управления рабочими потоками множеству специалистов в области биоинформатики приходилось в течение длительного времени конфигурировать и настраивать систему и ее

составляющие части, а затем нужно было потратить еще время на фактическое выполнение анализа. Дело еще больше усложняется тем, что перед приемом следующего образца на анализ нужно было изменять конфигурацию системы, причем изменение конфигурации системы для анализа нового набора образцов требовало еще больше времени. Благодаря технологии, описанной в настоящем документе, система может быть полностью автоматизирована. Представленная система, в частности, выполнена с возможностью автоматического приема множества образцов, отображения их на множество различных рабочих потоков и конвейеров и обработки их на одних и тех же или множестве разных системных плат.

[00719] Соответственно, система 151 управления рабочими потоками считывает требования к заданиям из штрихкодов, выделяет ресурсы для выполнения заданий, например, вне зависимости от местоположения, обновляет штрихкод образца и направляет образцы в выделенные ресурсы, например, блоки обработки, для обработки. Следовательно, именно диспетчер 151 рабочих потоков определяет протоколы вторичного 600 анализа и/или третичного 700 анализа, которые будут выполнены на принятых образцах. Эти блоки обработки представляют собой ресурсы, которые доступны для разграничения и выполнения операций, назначенных каждому набору данных. В частности, контроллер 151 рабочих потоков управляет различными операциями, связанными с приемом и считыванием образца, определением заданий, выделением ресурсов для выполнения этих заданий, например, вторичной обработки, соединение всех компонентов системы и продвижение набора образцов по системе от одного компонента к другому. Таким образом, контроллер 151 совершает действия по управлению всей системой, от начала до конца, например, от приема образца до формирования файла VCF и/или до третичной обработки включительно (см. ФИГ. 41B).

[00720] В дополнительных случаях, как показано на ФИГ. 41C, система 1 может включать в себя дополнительный ярус модулей 800 обработки, например, выполненных с возможностью осуществления дополнительной обработки, например, данных результатов вторичной и/или третичной обработки, например, для диагностики, открытия новых болезней и/или способов лечения и/или профилактики их. Например, в различных случаях может быть предусмотрен дополнительный уровень обработки 800, например, для диагностики болезней, терапевтического воздействия и/или профилактического предупреждения 70, например, включая НИПТ 123a, ОРИТН 123b, рак 123c, LDT 123d, аграрно-биологические 123e и другие такие данные диагностики болезней, профилактики и/или терапий, используя эти данные сформированные одним или более представленными первичными, и/или вторичными, и/или третичными конвейерами.

[00721] Соответственно, в настоящем документе представлена система 1 для создания и использования локальной сети 30 и/или глобальной гибридной сети 50. Например, в настоящее время локальное облако 30 используют в основном в качестве частного хранилища данных, например, в удаленном месте 400 хранения. В таком случае вычисление данных выполняют локально 100 с помощью локального вычислительного ресурса 140, и когда потребности в хранение большие, можно получить доступ к локальному облаку 30, чтобы сохранить данные, сформированные локальным вычислительным ресурсом 140, например, с помощью удаленного частного ресурса 400 хранения. Следовательно, управление формируемыми данными, как правило, осуществляют полностью в месте эксплуатации локально 100. В других вариантах реализации формирование, вычисление и управление данными осуществляют полностью

извне за счет защищенного подключения к удаленному вычислительному ресурсу 300 посредством интерфейса частного облака 30.

[00722] В частности, реализацию платформы биоинформационного анализа, локальные функции вычисления 140 и/или хранения 200 поддерживают, как правило, локально в месте эксплуатации 100. Однако, когда потребности хранения превышают емкость локального хранилища, данные можно выгрузить, получив доступ к локальному облаку 30 для приватного хранения вне места 400 эксплуатации. Кроме того, когда возникает потребность обеспечить другим удаленным пользователям доступ к сохраненным данным 400, такие данные могут быть переданы и сделаны доступными посредством интерфейса глобального облака 50 для удаленного хранилища 400, но для глобального доступа. В таком случае, когда вычислительные ресурсы 140, требуемые для выполнения функций вычисления минимальные, но требования к хранению большие, вычислительную функцию 140 можно поддерживать локально 100, тогда как функцию хранения 400 можно поддерживать удаленно, например, либо для частного, либо для глобального доступа, причем полностью обработанные данные перемещаются туда и обратно между локальной функцией 140 обработки, например, только для локальной обработки, и функцией хранения 400, например, для удаленного хранения 400 обработанных данных, например, путем использования протоколов JIT, описанных выше в настоящем документе.

[00723] Например, это можно показать на примере функции секвенирования 110, такой как типичный СНП, где ресурс 100 формирования данных и/или вычисления выполнен с возможностью осуществления функций, необходимых для секвенирования генетического материала для получения генетических секвенированных данных, например ридов, причем эти данные создают на месте 100 эксплуатации и/или передают на место эксплуатации локально 30. После формирования этих ридов, например, с помощью СНП в месте эксплуатации, могут быть затем переданы, например, в виде файла BCL или FASTQ, по облачной сети 30, например, для хранения 400, в удаленное место 300 таким образом, чтобы их можно было вызвать с облака 30 при необходимости, например, для дальнейшей обработки. Например, после того, как данные последовательности сформированы и сохранены, например, 400, они могут быть после этого вызваны, например, для локального использования, например, для выполнения одного или более из функций вторичной 600 обработки и/или третичной 700 обработки, то есть в месте, удаленном от хранилища 400, например локально 100. В таком случае локальный ресурс 200 хранения служит просто в качестве кэша хранилища, куда данные помещают в ожидании перемещения на облако 30/50 или с него, например, в удаленное хранилище 400 или из него.

[00724] Аналогичным образом, когда вычислительная функция объемная, например, требует одно или более ядер 300 удаленных вычислительных серверов или вычислительных кластеров для обработки данных, и когда потребности в хранении для хранения обработанных данных 200 относительно минимальные по сравнению с вычислительными ресурсами 300, требуемыми для обработки данных, подлежащие обработке данные могут быть отправлены, например, посредством облака 30, для обработки удаленным вычислительным ресурсом 300, который может содержать одно или более ядер или кластеров вычислительных ресурсов, например, один или более супервычислительных ресурсов. В таком случае после того, как данные обработаны облачным ядром 300 компьютера, их можно переместить посредством облачной сети 30, чтобы сохранить локально 200 и сделать легко доступными для использования локальным вычислительным ресурсом 140, например, для локального анализа и/или

диагностики. Конечно, удаленно сформированные данные 300 могут быть также сохранены удаленно 400.

[00725] Это также можно показать на примере типичной функции вторичной 600 обработки, например, когда предварительно обработанные секвенированные данные, например данные рида, хранятся локально 200 и к ним получает доступ, например, локальный вычислительный ресурс 100, и передает по облачной сети 30 в Интернете на удаленное вычислительное средство 300 для обработки тем самым, например, с помощью функции вторичной 600 или функции третичной 700 обработки, с целью получения обработанных данных результатов, которые могут быть затем отправлены обратно на локальное средство 100 для хранения 200 тем самым. Это может иметь место, когда местный практикующий врач формирует секвенированные данные рида с помощью локального ресурса 110 формирования данных, например, автоматизированного секвенатора, чтобы получить файл BCL или FASTQ, и затем отправляет эти данные по сети 50 в удаленное вычислительное средство 300, которое после этого применяет к этим данным последовательности одну или более функций, например, функцию преобразования Барроуза-Уилера, или функцию выравнивания Нидлмана-Вунша и/или Смита-Ватермана, чтобы сформировать данные результатов, например, в формате файла SAM, которые могут быть затем сжаты и переданы по Интернету 30/50, например, в виде файла BAM, на локальный вычислительный ресурс 100 для исследования тем самым в одном или более локально управляемых протоколах обработки, например, для создания файла VCF, который может быть затем сохранен локально 200. В различных случаях данные могут быть также сохранены удаленно 400.

[00726] Однако необходима бесшовная интеграция для взаимодействия между локальной 100 и удаленной 300 компьютерной обработкой, а также между локальным 200 и удаленным сохранением 400, такая как в гибридном облачной системе 50, представленной в настоящем документе. В таком случае система может быть выполнена таким образом, чтобы локальный 100 и удаленный 300 вычислительные ресурсы были выполнены с возможностью беспрепятственной совместной работы таким образом, чтобы данные, подлежащие при этом обработке, могли выделяться в реальном времени либо локальному 100, либо уделенному 300 вычислительному ресурсу, без существенных издержек вследствие скорости передачи и/или эксплуатационной эффективности. Это может иметь место, например, когда программная, и/или аппаратная, и/или квантовая обработка должны быть развернута или иным образом осуществлена с помощью вычислительных ресурсов 100 и 300, которые выполнены с возможностью соответствия друг другу и/или наличия одинаковых или похожих функциональных возможностей, например, аппаратное и/или программное обеспечение выполнено одинаковым образом, чтобы одинаково выполнять одни и те же алгоритмы на сформированных и/или принятых данных.

[00727] Например, как показано на ФИГ. 41A, локальный вычислительный ресурс 100 может быть выполнен с возможностью формирования или приема сформированных данных и, следовательно, может содержать механизм 110 формирования данных, например, для формирования первичных данных, и/или анализа 500, например, для создания файла последовательности BCL и/или FASTQ. Этот механизм 110 формирования данных может представлять собой или быть связанным с локальным компьютером 100, как описано повсюду в настоящем документе, имеющим процессор 140, который может быть выполнен с возможностью выполнения одного или более программных приложений и/или может быть реализован аппаратно для выполнения одного или более алгоритмов, например, в монтажной конфигурации, на

сформированных и/или полученных данных. Например, механизм 110 формирования данных может быть выполнен с возможностью получения одних или более формируемых данных, например, секвенирования 111 данных. В различных вариантах реализации формируемые данные могут быть обнаруженными данными 111a, такими как данные, которые могут быть обнаружены как изменение напряжения, концентрации ионов, электромагнитного излучения и т.п.; и/или механизм 110 формирования данных может быть выполнен с возможностью формирования и/или обработки сигнала, например, данных аналогового или цифрового сигнала, таких как данные, представляющие одну или более нуклеотидных идентичностей в последовательности или цепочке связанных нуклеотидов. В таком случае механизм 110 формирования данных, например, секвенатор 111, может быть также выполнен с возможностью осуществления предварительной обработки на сформированных данных с целью обработки 111b сигнала или выполнения одной или более операций 111c определения оснований, например, на данных, для получения данных идентичности последовательности, например, файла 111d BCL и/или FASTQ.

[00728] Необходимо отметить, что в этом случае создаваемые данные 111 могут быть сформированы локально и непосредственно, например, с помощью локального ресурса 110 формирования данных и/или вычислительного ресурса 140, например СНП или секвенатора на микросхеме. В альтернативном варианте реализации данные могут быть созданы локально и опосредованно, например, с помощью удаленного вычислительного и/или формирующего ресурса, такого как удаленный СНП. После того, как данные 111, например, в формате файла BCL и/или FASTQ, созданы, они могут быть переданы опосредованно через локальное облако 30 на локальный вычислительный ресурс 100, например, для вторичной обработки 140 и/или сохранения в локальном ресурсе 200 хранения, например, на время ожидания дальнейшей локальной обработки 140. В таком случае, когда ресурс формирования данных удален от локальных ресурсов обработки 100 и/или хранения 200, соответствующие ресурсы могут быть выполнены таким образом, чтобы удаленное и/или локальное сохранение, удаленная и локальная обработка и/или протоколы обмена данными, используемые каждым ресурсом, могли быть выполнены с возможностью плавной и/или бесшовной интеграции друг с другом, например, за счет выполнения одинакового, подобного и/или эквивалентного программного обеспечения, и/или наличия одинаковых, подобных и/или эквивалентных аппаратных конфигураций, и/или использования одинаковых протоколов связи и/или передачи, которые, в некоторых случаях, могли быть реализованы во время изготовления или позже этого.

[00729] В частности, в одной реализации эти функции могут быт реализованы в аппаратной конфигурации, например, где функция секвенирования и функция вторичной обработки поддерживаются на одной и том же или связанной микросхеме или наборе микросхем, например, когда секвенатор и процессор вторичной обработки взаимно соединены напрямую на микросхеме, как описано в настоящем руководстве. В других реализациях эти функции могут быть реализованы на двух или более отдельных устройствах посредством программного обеспечения, например, на квантовом процессоре, ЦПУ или ГПУ, которые оптимизированы для обеспечения беспрепятственного обмена данными друг с другом между эти двумя удаленными устройствами. В других реализациях для выполнения перечисленных функций может быть использована также комбинация аппаратных и программных реализаций.

[00730] Точнее говоря, одна и та же конфигурация может быть реализована для выполнения картирования, выравнивания, сортировки, определения вариантов и/или

других функций, которые могут быть развернуты локальным 100 и/или удаленным 300 вычислительным ресурсами. Например, локальный вычислительный 100 и/или удаленный 300 ресурсы могут включать в себя программное и/или аппаратное обеспечение, выполненное с возможностью осуществления одной или более функций 112-115 обработки вторичного яруса 600 и/или функций обработки третичного яруса 700/800 на локально и/или удаленно формируемых данных, таких как данные генетической последовательности таким образом, что обработка и ее результаты могут беспрепятственно совместно использоваться друг с другом и/или храниться благодаря этому. В частности, локальная вычислительная функция 100 и/или удаленная вычислительная функция 300 могут быть выполнены с возможностью формирования и/или приема первичных данных, таких как данные генетической последовательности, например, в формате файла BCL и/или FASTQ, и выполнения одного или более протоколов вторичной 600 и/или третичной 700 обработки на этих сформированных и/или полученных данных. В таком случае один или более из этих протоколов может быть реализован в программном обеспечении, аппаратном обеспечении или в комбинированном формате, например, выполняться на квантовом процессоре, ЦПУ и/или ГПУ, Например, ресурс 110 формирования данных и/или локальный ресурс 100 и/или удаленный ресурс 300 обработки могут быть выполнены с возможностью осуществления одной или более из операции 112 картирования, операции 113 выравнивания, операции 115 определения вариантов или другой связанной функции 114 на полученных или сформированных данных в программном и/или аппаратном обеспечении.

[00731] Соответственно, в различных вариантах реализации ресурс формирования данных, такой как секвенатор 111, например, СНП или секвенатор на микросхеме, будь то реализованный в программно и/или аппаратном обеспечении или их комбинации, может быть также выполнен с возможностью включения в себя начального яруса процессоров 500, таких как планировщик, различная аналитика, средства сравнения, средства построения графов, средства выпуска и т.п., для оказания помощи формирователю 111 данных, например, секвенатору, в преобразовании биологической информации в необработанные данные рида в формате файлов 111d BCL или FASTQ. Кроме того, локальный вычислительный ресурс 100, будь то реализованный в программном и/или аппаратном обеспечении или их комбинации, может быть также выполнен с возможностью включения в себя дополнительного яруса процессоров 600, например, может содержать движок 112 картирования или может иным образом включать в себя программирование для выполнен алгоритма на данных генетической последовательности, например, для выполнения преобразования Барроуза-Уилера и/или другого алгоритма для построения хэш-таблицы и/или выполнения хэш-функции 112a на указанных данных, например, для картирования затравки хэша, чтобы сформировать картированные данные последовательности. Более того, локальный вычислительный ресурс 100, будь то реализованный в программном и/или аппаратном обеспечении или их комбинации, может быть также выполнен с возможностью включения в себя начального яруса процессоров 600, например, может также содержать движок 113 выравнивания, как описано в настоящем документе, или может иным образом включать в себя программирование для выполнения алгоритма выравнивания на данных генетической последовательности, например, картированной секвенированной последовательности, например, для выполнения выравнивания Смита-Ватермана с гэпами и/или без гэпов, и/или алгоритма Нидлмана-Вунша или подобного алгоритма 113a оценки, чтобы сформировать выровненные данные последовательности.

[00732] Локальный вычислительный ресурс 100 и/или ресурс 110 формирования данных могут быть также выполнены с возможностью включения в себя одного или более других модулей 114, будь то реализованных в программном и/или аппаратном обеспечении или их комбинации, которые могут быть выполнены с возможностью осуществления одной или более других функций обработки на данных генетической последовательности, например, на картированной и/или выровненной последовательности данных. Таким образом, один или более других модулей могут содержать соответствующим образом сконфигурированный движок 114 или иным образом включать в себя программирование для выполнения одной или более других функций обработки, таких как функции сортировки 114a, удаления 114b дубликатов, перекалибровки 114c, локального повторного выравнивания 114d, маркировки 114f дубликатов, перекалибровки 114g оценки качества оснований и/или функция 114e сжатия (например, для получения файла SAM, редуцированного BAM и/или сжатия и/или распаковки CRAM) в соответствии со способами, описанными в настоящем документе. В различных случаях одна или более из этих функций обработки могут быть выполнены в виде одного или более конвейеров системы 1.

[00733] Аналогичным образом система 1 может быть выполнена с возможностью включения в себя модуля 115, будь то реализованного в программно и/или аппаратном обеспечении или их комбинации, который может быть выполнен с возможностью обработки данных, например, секвенированных, картированных, выровненных и/или сортированных данных, таким образом, чтобы получать файл 116 определения вариантов. В частности, система 1 может содержать модуль 115 определения вариантов для выполнения одной или более функций определения вариантов, таких как функция 115a скрытой марковской модели (HMM) и/или GATK, например, в монтажной конфигурации и/или посредством одного или более программных приложений, например, либо локально, либо удаленного, и/или преобразователь 115b для них. В различных случаях этом модуль может быть выполнены в виде одного или более конвейеров системы 1.

[00734] В конкретных вариантах реализации, как показано на ФИГ. 41B, система 1 может включать в себя локальную вычислительную функцию 100, которая может быть выполнена с возможностью использования ресурса 150 компьютерной обработки для выполнения одной или более функций дальнейшей обработки на данных, например, данных BCL и/или FASTQ, сформированных системным генератором 110 данных и/или полученных системным механизмом 120 получения данных (как описано в настоящем документе), например, переданных в него, например, третьей стороной 121, например, посредством облака 30 или гибридной облачной сети 50. Например, сторонний анализатор 121 может использовать удаленный вычислительный ресурс 300 для формирования соответствующих данных, нуждающихся в дальнейшей обработки, таких как данные генетической последовательности и т.п., которые могут быть переданы в систему 1 по сети 30/50 для дальнейшей обработки. Это может быть полезно, например, когда удаленный вычислительный ресурс 300 представляет собой СНП, выполненный с возможностью получения необработанных биологических данных и преобразования их в цифровое представление этих данных, например, в форме одного или более файлов FASTQ, содержащих риды данных генетической последовательности; и когда требуется дальнейшая обработка, например, для определения того, как сформированная последовательность индивида отличается от одной или более референсных последовательностей, как описано в настоящем документе, и/или желательно результаты этой обработки подвергнуть дальнейшей, например, третичной, обработке.

[00735] В таком случае система 1 выполнена с возможностью предоставления одной или более сторон, например, основному, и/или второстороннему, и/или третьестороннему пользователю, доступа к связанных локальным ресурсам 100 обработки и/или соответствующим образом связанным с ней сконфигурированным удаленным ресурсам 300 обработки таким образом, чтобы позволить пользователю выполнять одну или более количественных и/или качественных функций 152 обработки на сформированных и/или полученных данных. Например, в одной конфигурации система 1 может включать в себя, например, в дополнение к конвейерами первичной 500 и/или вторичной 600 обработки, третий ярус модулей 700/800 обработки, которые могут быть выполнены с возможностью осуществления одной или более функций обработки на сформированных и/или полученных данных и/или вторичных обработанных данных.

[00736] В частности, в одном варианте реализации система 1 может быть выполнена с возможностью формирования и/или приема обработанных данных 111 генетической последовательности, которые удаленно или локально картированы 112, выровнены 113, отсортированы 114a и/или подвергнуты дальнейшей обработке 114 для формирования файла 116 определения вариантов, который может быть затем подвергнут третичной обработке, например, в системе 1, например, в ответ на запросы 121 второсторонней и/или третьесторонней аналитики. Более конкретно, система 1 может быть выполнена с возможностью приема запросов на обработку от третьей стороны 121 и дополнительно выполнена с возможностью осуществления такой запрошенной вторичной 600 и/или третичной 700/800 обработки на сформированных и/или полученных данных. В частности, система 1 может быть выполнена с возможностью создания и/или получения данных 111 генетической последовательности, может быть выполнена с возможностью взятия этих данных генетической последовательности и картирования 112, выравнивания 113 и/или сортировки 114a их и обработки для создания одного или более файлов 116 определения вариантов (VCF), и дополнительно система 1 может быть выполнена с возможностью осуществления функции 700/800 третичной обработки на данных, например, применительно к одному или более файлов VCF, сформированных или принятых системой 1.

[00737] В частности, система 1 может быть выполнена с возможностью осуществления в той или иной форме третичной обработки 700 на сформированных и/или полученных данных, например, применения к ним одной или более функций 700 конвейерной обработки для формирования данных 122a генома, например, полного, данных122b эпигенома, данных 122c метагенома и т.п., включая данные 122d генотипирования, в том числе совместного генотипирования, данные анализов вариантов, включая GATK 122e и/или данные анализа MuTect2 122f , среди других потенциальных данных аналитических конвейеров, таких как конвейер микроматричного анализа, конвейер анализа экзома, конвейер анализа микробиома, конвейер анализа секвенирования РНК и конвейеры других генетических анализов. Кроме того, система 1 может быть выполнена с возможностью осуществления дополнительного яруса обработки 800 на сформированных и/или обработанных данных, в том числе включая одного или более из неинвазивного пренатального тестирования (НИПТ) 123a, ОРИТН 123b, связанной с раком диагностики и/или терапевтических воздействий 123c, различных разработанных в лаборатории тестов (LDT) 123d, сельскохозяйственных биологических (Ag Bio) приложений 123e или других таких относящихся к здравоохранению 123f функций обработки (см. ФИГ. 41C).

[00738] Поэтому в различных вариантах реализации, где основной пользователь может получать доступ и/или конфигурировать систему 1 и ее различные компоненты

напрямую, например, путем прямого доступа к ним, такого как посредством локального вычислительного ресурса 100, как представлено в настоящем документе, система 1 может быть также выполнена с возможностью предоставления доступа второй стороне, например, соединенной с системой 1 посредством подключения 10 по локальной сети или внутренней сети, для конфигурирования и использования системы 1 в локальной среде. Кроме того, в определенных вариантах реализации система может быть выполнена с возможностью предоставления доступа и/или конфигурирования ее третьей стороной 121, например, посредством связанной гибридной облачной сети 50, соединяющей третью сторону 121 с системой 1, например, с помощью интерфейса прикладных программ (API), к которому можно получить доступ с помощью одного или более компонентов графического пользовательского интерфейса (ГПИ). Такой ГПИ может быть выполнен с возможностью обеспечения третьестороннему пользователю доступа к системе 1 и использования API для конфигурирования различных компонентов системы, модулей, связанных с конвейерами, и других связанных функциональных возможностей формирования и/или обработки данных, чтобы использовать только те компоненты системы, которые необходимы и/или полезны для третьей стороны и/или выполнение которых запрошено или желательно для нее.

[00739] Соответственно, в различных случаях система 1, которая представлена в настоящем документе, может быть выполнена с возможностью конфигурирования ее основным, второсторонним или третьесторонним пользователем системы. В таком случае система 1 может быть выполнена с возможностью разрешения пользователю конфигурировать систему 1 и тем самым компоновать ее компоненты таким образом, чтобы разворачивать один, все или часть ресурсов аналитической системы, например, 152, для выполнения на данных, которые сформированы, получены или иным образом переданы в систему, например, основным, второсторонним или третьесторонним пользователем, таким образом, чтобы система 1 использовала только те части системы, которые необходимы или полезны для выполнения аналитики, запрошенной пользователем для получения при этом требуемых результатов. Например, для этих и других целей в систему 1 может быть включен API, который выполнен с возможностью включения или иным образом функционального связывания с графическим пользовательским интерфейсом (ГПИ), содержащим действующее меню и/или сопутствующий список вызовов системных функций, в котором пользователь может выбирать и/или выполнять иные действия, чтобы сконфигурировать и использовать систему и ее компоненты требуемым образом.

[00740] В таком случае с помощью меню ГПИ и/или вызовов системных функций можно управлять выбираемыми пользователем операциями из числа одной или более операций первого яруса операций 600: секвенирование 111, картирование 112, выравнивания 113, сортировки 114a, определение вариантов 115 и/или других связанных функций 114 в соответствие с идеями, изложенными в настоящем документе, например, в отношении функций первичной и/или вторичной обработки, описанных в настоящем документе. Кроме того, при желании с помощью меню ГПИ и/или вызовов системных функций можно управлять операциями из числа одной или более операций второго яруса операций 700: конвейер 122a анализа генома, например, полногеномного анализа, конвейер 122b эпигенома, конвейер 122c метагенома, конвейер 122d генотипирования, например, совместного, конвейеры анализа вариантов, например, 122e GATK и/или 122f MuTect2, включая конвейеры структурных вариантов, конвейер анализа экзома, конвейер анализа микробиома, конвейеры секвенирования РНК и конвейеры других генетических анализов. Кроме того, если требуется, с помощью меню ГПИ и вызовов

функций системы можно руководить операциями, которые могут быть выбраны пользователем, из числа одной или более операций 800 третьего яруса, включая: неинвазивное пренатальное тестирование (НИПТ) 123a, ОРИТН 123b, связанную с раком диагностику и/или терапевтические воздействия 123c, различные разработанные в лаборатории тесты (LDT) 123d, сельскохозяйственные биологические (Ag Bio) приложения 123e или другие такие относящиеся к здравоохранению 123f функции обработки.

[00741] Соответственно, меню и вызовы системных функций могут содержать одну или более функций первичной, вторичной и/или третичной обработки, чтобы можно было сконфигурировать систему и/или ее составляющие части для выполнения одного или более конвейеров анализа данных, которые выбраны и сконфигурированы пользователем. В таком случае локальный вычислительный ресурс 100 может быть выполнен с возможностью соответствия и/или зеркального отражения удаленного вычислительного ресурса 300 и/или, аналогичным образом, локальный ресурс 200 хранения может быть выполнен с возможностью соответствия и/или зеркального отражения удаленного ресурса 400 хранения, чтобы различные компоненты системы могли быть выполнены и/или сформированные таким образом данные могли быть сохранены либо локально, либо удаленно беспрепятственным образом по выбору с помощью системы 1. Кроме того, в конкретных вариантах реализации система 1 может быть выполнена с возможностью предоставления доступа третьим сторонам для выполнения запатентованных протоколов 121a анализа на сформированных и/или обработанных данных, например, путем применения интерфейса искусственного интеллекта, предназначенного для поиска корреляций между ними.

[00742] Система 1 может быть выполнена с возможностью осуществления в той или иной форме третичной обработки на сформированных и/или полученных данных. Следовательно, в различных вариантах реализации основной, второсторонний или третьесторонний пользователь может получать доступ и/или конфигурировать любой уровень системы 1 и ее различные компоненты либо напрямую, например, путем прямого доступа с помощью вычислительного ресурса 100 либо опосредованно, например, с помощью соединения локальной сети 30 или связанной гибридной облачной сети 50, соединяющей эту сторону с системой 1, например, посредством соответствующим образом сконфигурированного API, при наличии надлежащих полномочий. В таком случае компоненты системы могут быть представлены в меню, ГПИ с возможностью выбора, где пользователь может выбирать из всех различных вариантов обработки и хранения те, которые нужно выполнить на представленных пользователем данных. Кроме того, в различных случаях пользователь может выгружать свои собственные системные протоколы, чтобы они были приняты и выполнены системой для обработки различных данных способом, предусмотренным и выбранным пользователем. В таком случае ГПИ и связанный API позволят пользователю получать доступ к системе 1 и использовать API для конфигурирования различных компонентов системы, модулей, связанных с конвейерами, и других связанных функциональных возможностей формирования и/или обработки данных, чтобы использовать только те компоненты системы, которые необходимы и/или полезны для этой стороны и/или выполнение которых запрошено или желательно для нее.

[00743] Как показано на ФИГ. 41C, один или более из вышеобозначенных модулей и их соответствующих функций и/или связанных ресурсов могут быть выполнены с возможностью осуществления их удаленно, например, с помощью удаленного вычислительного ресурса 300, и также могут быть выполнены с возможностью передачи

в систему 1, например, с помощью протокола беспрепятственной передачи по глобальному облачному соединению 50 через Интернет, например, посредством соответствующим образом сконфигурированного механизма 120 получения данных. Соответственно, в таком случае локальный вычислительный ресурс 100 может содержать механизм 120 получения данных, например, выполненный с возможностью передачи и/или приема таких получаемых данных и/или связанной информации.

[00744] Например, система 1 может включать в себя механизм 120 получения данных, который выполнен с возможностью обеспечения непрерывной обработки и/или сохранения данных беспрепятственным и устойчивым образом, например посредством облачной сети 50, где функции обработки распределяются локально 100 и/или удаленно 300. Аналогичным образом, когда один или более результатов такой обработки могут быть сохранены локально 200 и/или удаленно 400, система беспрепятственно назначает, в какой локальный или удаленный ресурс отправлять данное задание для обработки и/или сохранения вне зависимости от того, где физически находится этот ресурс. Такие распределенные обработка, передача и получение могут включать в себя одну или более из функций секвенирования 111, картирования 112, выравнивания 113, сортировки 114a, маркировки 114c дубликатов, удаления дубликатов, перекалибровки 114d, локального повторного выравнивания 114e, перекалибровки 114f оценок качества оснований и/или функцию 114g сжатия, а также функцию 116 определения вариантов, как описано в настоящем документе. В случае хранения локально 200 или удаленно 400 обработанные данные, в каком бы состоянии они ни были в процессе, могут быть сделаны доступными для либо локального 100, либо для удаленного 300 ресурсов обработки, например, для дальнейшей обработки перед повторной передачей и/или повторным сохранением.

[00745] В частности, система 1 может быть выполнена с возможностью создания и/или получения данных 111 генетической последовательности, может быть выполнена с возможностью взятия этой последовательности генетических данных и обработки их локально 140 или передачи этих данных посредством соответствующим образом сконфигурированного облака 30 или гибридной облачной сети 50, например, на средство удаленной обработки для удаленной обработки 300, Кроме того, система 1 может быть выполнена так, чтобы после обработки сохранять обработанные данные дистанционно 400 или передавать их обратно для локального хранения 200. Соответственно, система 1 может быть выполнена с возможностью либо локального либо дистанционного формирования и/или обработки данных, например, когда этапы формирования и/или обработки могут быть из первого яруса функций 600 первичной и/или вторичной обработки, который может включать в себя одну или более функций: секвенирования 111, картирования 112, выравнивания 113 и/или сортировки 114a для создания одного или более файлов 116 определения вариантов (VCF).

[00746] Кроме того, система 1 может быть выполнена с возможностью либо локального, либо удаленного формирования и/или обработки данных, например, когда этапы формирования и/или обработки могут быть из второго яруса функций 700 третичной обработки, который может включать в себя одно или более из формирования и/или получения данных, относящихся к конвейеру 122a генома, конвейеру 122b эпигенома, конвейеру 122c метагенома, конвейеру 122d генотипирования, конвейеру анализа вариантов, например, 122e GATK и/или 122f MuTect2, а также конвейерам других третичных анализов, таких как конвейер микроматричного анализа, конвейер анализа микробиома, конвейер анализа экзома, а также конвейеры секвенирования РНК и конвейеры других генетических анализов. Кроме того, система 1 может быть

выполнена с возможностью либо локального, либо удаленного формирования и/или обработки данных, например, когда этапы формирования и/или обработки могут быть из третьего яруса функций 800 третичной обработки, который может включать в себя одно или более из формирования и/или получения данных, которые имеют отношение или включают в себя: неинвазивное пренатальное тестирование (НИПТ) 123a, ОРИТН 123b, связанную с раком диагностику и/или терапевтические воздействия 123c, различные разработанные в лаборатории тесты (LDT) 123d, сельскохозяйственные биологические (Ag Bio) приложения 123e или другие такие относящиеся к здравоохранению 123f функции обработки.

[00747] В конкретных вариантах реализации, как показано на ФИГ. 41C, система 1 может быть также выполнена с возможностью предоставления одной или более сторонам доступа к системе и передаче информации в связанные ресурсы локальной обработки 100 и/или удаленной обработки 300 или из них, а также к сохранению информации либо локально 200, либо удаленно 400 таким образом, который позволяет пользователю выбирать, какую информацию обрабатывать и/или сохранять, и в каком месте системы. В таком случае пользователь может не только решать, какие функции первичного, вторичного и/или третичного анализа выполнять на сформированных и/или полученных данных, но и как эти ресурсы разворачивать, и/или где сохранять результаты такой обработки. Например, в одной конфигурации пользователь может выбирать, формировать ли данные локально или удаленно либо использовать ли комбинированный подход, подвергать ли их вторичной обработке, и если да, то с задействованием каких модулей вторичной обработки и/или с использованием каких ресурсов в этих процессах, а также может определять, подвергать ли после этого сформированные или полученные данные третичной обработке, и если да, то с задействованием каких модулей и/или каких ярусов третичной обработки и/или с использованием каких ресурсов в этих процессах, и, аналогичным образом, где сохранять результаты этих процессов для каждого этапа операций.

[00748] В частности, в одном варианте реализации пользователь может конфигурировать систему 1, изображенную на ФИГ. 41A, чтобы формирование данных 111 генетической последовательности происходило удаленно, например, с помощью СНП, но вторичная обработка 600 данных происходила локально 100. В таком случае пользователь может тогда определять, какие из функций вторичной обработки выполнять локально 100, например, путем выбора функций обработки, таких как картирование 112, выравнивание 113, сортировка 114a и/или создание VCF 116, в меню доступных вариантов обработки. Затем пользователь может определять, подвергать ли локально обработанные данные третичной обработке, и если да, какие модули активировать для дальнейшей обработки данных, и должна ли эта третичная обработка происходить локально 100 или удаленно 300. Аналогичным образом пользователь может выбирать различные варианты для различных ярусов вариантов третичной обработки и где любые сформированные и/или полученные данные нужно сохранять, локально 200 или удаленно 400, на каждом данном этапе или моменте операции.

[00749] Более конкретно, основной пользователь может конфигурировать систему для приема запросов на обработку от третьей стороны, причем третья сторона может конфигурировать систему для выполнения такой запрошенной первичной, вторичной и/или третичной обработки на сформированных и/или полученных данных. В частности, пользователь или вторая и/или третья сторона могут конфигурировать систему 1 для создания и/или получения данных генетической последовательности либо локально 100, либо удаленно 300. Кроме того, пользователь может конфигурировать систему 1

для взятия этих данных генетической последовательности и картирования, выравнивания и/или сортировки ее, либо локально, либо удаленно, чтобы создать один или более файлов определения вариантов (VCF). Кроме того, пользователь может конфигурировать систему для выполнения функции третичной обработки на данных, например, применительно к одному или более файлам VCF, либо локально, либо удаленно.

[00750] Еще более конкретно, пользователь или другая сторона могут конфигурировать систему 1 для выполнения той или иной формы третичной обработки на сформированных и/или полученных данных, и где в системе должна происходить обработка. Следовательно, в различных вариантах реализации основной, второсторонний и/или третьесторонний пользователь121 может получать доступ к системе 1 и/или конфигурировать саму систему и ее различные компоненты, например, путем прямого доступа к локальной вычислительной функции 100 посредством соединения локальной сети 30 или соединения связанной гибридной облачной сети 50, соединяющей сторону 121 с системой 1, например, с помощью интерфейса прикладных программ (API), к которому можно получить доступ с помощью одного или более компонентов графического пользовательского интерфейса (ГПИ). В таком случае третьесторонний пользователь может получать доступ к системе 1 и использовать API для конфигурирования различных компонентов системы, модулей, связанных с конвейерами, и других связанных функциональных возможностей формирования и/или обработки данных, чтобы использовать только те компоненты системы, которые необходимы и/или полезны для третьей стороны и/или выполнение которых запрошено или желательно для нее, и также выделять вычислительные ресурсы, которые обеспечат обработку, и где будут сохранены результаты обработки.

[00751] Соответственно, в различных случаях система 1 может быть выполнена с возможностью конфигурирования ее основным, второсторонним или третьесторонним пользователем системы, который может конфигурировать систему 1 так, чтобы компоновать ее компоненты таким образом, чтобы развертывать один, все или выбранные аналитические системные ресурсы для выполнения на данных, которые пользователь формирует непосредственно, или поручает их формирование системе 1, или дает команду на передачу их в систему 1, например, по сети, связанной с ней, например, посредством механизма 120 получения данных. Таким образом, система 1 выполнена с возможностью использования только тех частей системы, которые необходимы или полезны для аналитики, требуемой для запрашивающей стороны или запрошенной ею. Например, для этих и других целей может быть включен API, который выполнен с возможностью включения в себя действующего меню ГПИ и/или сопутствующего списка вызовов системных функций, в котором пользователь может выбирать, чтобы сконфигурировать и использовать систему требуемым образом.

[00752] Кроме того, в конкретных вариантах реализации система 1 может быть выполнена с возможностью предоставления доступа основному пользователю и/или третьим сторонам, таким как правительственные регулятивные органы, например, Федеральное управление по лекарственным средствам (FDA) 70b, или предоставления основным пользователям или третьим сторонам возможности сличать, компилировать и/или получать доступ к базе данных генетической информации, извлеченной и/или иным образом полученной и/или скомпилированной системой 1, для формирования базы 70a данных электронных медицинских записей (EMR), и/или предоставления правительственным органам возможности доступа к системе и/или надзора за ней, например, для FDA с целью оценки разработки лекарственных средств. Система 1 может

быть также настроена для сбора, компиляции и/или аннотирования данных 70c и/или предоставления возможности доступа к ним пользователям высокого уровня.

[00753] Соответственно, система 1 или ее компоненты могут быть выполнены с возможностью предоставления доступа к ним удаленному пользователю, например, основному пользователю или третьей стороне, и, следовательно, один или более из компьютерных ресурсов 100 и/или 300 могут содержать пользовательский интерфейс и/или могут также содержать устройство отображения, имеющее графический пользовательский интерфейс для обеспечения потенциальному пользователю системы возможности доступа к системе для передачи данных образца для ввода в один или более из BioIT-конвейеров, описанных в настоящем документе, и/или для приема данных результатов из них. ГПИ или другой интерфейс могут быть выполнены с возможностью предоставления пользователю возможности управления компонентами системы, например, посредством соответствующим образом сконфигурированного веб-портала, и отслеживания хода выполнения обработки вне зависимости от того, являются ли задействуемые вычислительные ресурсы доступными локально 100 или удаленно 300. Соответственно, ГПИ может содержать список наборов заданий, которые могут быть выполнены, например, картирование 112, выравнивание 113 и т.д., и/или наборов ресурсов для выполнения заданий, причем пользователь может сам выбирать, какие задания он хочет выполнить, и с использованием каких ресурсов. Следовательно, в подобном случае каждый отдельный пользователь может поэтому создать уникальный, или может использовать заданный, рабочий процесс анализа, например, с помощью щелчков мышью, перетаскивания или выбора иным образом, конкретные рабочие проекты, которые он желает выполнить.

[00754] Например, в одной модели использования предложена информационная панель с интерфейсом ГПИ, которая может содержать множество значков, представляющих различные процессы, которые могут быть реализованы и выполнены системой. В таком случае пользователь может щелкать или перетаскивать значки выбранных рабочих процессов в интерфейс рабочих потоков, чтобы построить требуемый процесс рабочего потока, который после его построения можно сохранить и использовать для создания управляющих инструкций для штрихкодов набора образцов. После того, как требуемые рабочие проекты выбраны, контроллер 151 управления рабочими потоками может сконфигурировать требуемые процессы рабочего потока (например, вторичный анализ) и затем определить и выбрать ресурсы для выполнения выбранного анализа.

[00755] После начала процесса анализа рабочего потока можно следить за его продвижением по системе, наблюдая за ним на информационной панели. Например, панель информационная мониторинга указывать, сколько данных проходят через систему, какие процессы выполняются на этих данных, что уже сделано, сколько осталось обработать, какие рабочие потоки завершены, а к каким еще нужно получить доступ, последние проекты, подлежащие выполнению, а выполнение каких уже завершено. По существу на рабочем столе можно получить доступ ко всему, что выполняется в системе или ее части.

[00756] Кроме того, в различных случаях рабочий стол может содержать всевозможные разные пользовательские интерфейсы, к которым можно получить доступ с помощью одной или более вкладок. Например, одной из вкладок для доступа к элементам управления системы может быть вкладка «локальные ресурсы 100», которая, если она выбрана, позволяет пользователю выбирать функции управления, которые могут быть реализованы локально. Другая вкладка может быть выполнена

с возможностью доступа к «облачным ресурсам 300», которая, если она выбрана, позволяет пользователю выбирать другие функции управления, которые могут быть реализованы удаленно. Соответственно, взаимодействуя с информационной панелью, пользователь может выбирать, каким ресурсам какие задачи выполнять, и тем самым может повышать или снижать использование ресурса по мере надобности для удовлетворения требований к проекту.

[00757] Следовательно, по мере возрастания вычислительной сложности и/или потребности в увеличении скорости пользователь (или сама система, например, WMS 151) может приводить в действие больше и больше ресурсов по мере надобности, например, простым нажатием кнопки, указывающим диспетчеру рабочих потоков привести в действие дополнительные локальные 100 и/или облачные 300 ресурсы, которые необходимы для выполнения задачи в пределах требуемых временных рамок. Таким образом, хотя система автоматизирована и/или управляется контроллером 151 рабочих потоков, пользователь системы все же может устанавливать параметры управления и при необходимости может приводить в действие облачные ресурсы 300. Соответственно, контроллер 151 может распространиться на облако 50/300 при необходимости, чтобы привести в действие дополнительные ресурсы обработки и/или хранения 400.

[00758] В различных случаях интерфейс рабочего стола может быть выполнен в виде мобильного приложения или «приложения», которое доступно посредством мобильного устройства и/или настольного компьютера. Следовательно, в соответствии с одним аспектом может быть предусмотрена геномная рыночная площадка, или когорта, чтобы предоставить множеству пользователей возможность сотрудничать в одном или более исследовательских проектах для создания электронной когортной рыночной площадки, которая доступна посредством приложения информационной панели, например, интерфейса веб-браузера. Поэтому система может предоставлять форум в Интернете для выполнения совместного исследования и/или рыночную площадку для разработки различных аналитических средств анализа генетических данных, причем система может быть доступна непосредственно через системный интерфейс или через приложение для обеспечения удаленного управления системой пользователем.

[00759] Соответственно, в различных вариантах реализации, как показано на ФИГ. 42А, предусмотрено гибридное облако 50, которое выполнено с возможностью соединения локального вычислительного ресурса 100 и/или ресурса 200 хранения с удаленным вычислительным ресурсом 300 и/или ресурсом хранения 400, например, когда локальные и удаленные ресурсы отделены друг друга расстоянием, пространством, географически и т.п. В таком случае локальные и удаленные ресурсы могут быть выполнены с возможностью обмена данными друг с другом таким образом, чтобы совместно использовать информацию, например, цифровые данные, беспрепятственно между собой. В частности, локальные ресурсы могут быть выполнены с возможностью осуществления одного или более видов обработки на данных, например, до передачи по гибридной сети 50, а удаленные ресурсы могут быть выполнены с возможностью осуществления одно или более других видов обработки на данных.

[00760] Например, в одной конкретной конфигурации система 1 может быть выполнена с таким образом, чтобы функция 152 формирования и/или анализа была выполнена с возможностью осуществления локально 100 локальным вычислительным ресурсом, например, в целях выполнения функции первичной и/или вторичной обработки для формирования и/или обработки данных генетической последовательности, как описано в настоящем документе. Кроме того, в различных вариантах реализации

локальные ресурсы могут быть выполнены с возможностью осуществления одной или более функций третичной обработки на данных, например, одного из анализов генома, экзома и/или эпигенома, или рака, микробиома и/или других анализов обработки ДНК/РНК. Кроме того, когда такие обработанные данные предназначены для передачи, например, на удаленный вычислительный ресурс 300 и/или ресурс хранения 400, данные могут быть преобразованы, например, с помощью соответствующим образом сконфигурированного преобразователя, который может быть выполнен с возможностью индексации, конвертации, сжатия и/или шифрования данных, например, до передачи по гибридной сети 50.

[00761] В конкретных случаях, например, когда сформированные или обработанные данные передают на удаленный вычислительный ресурс, например, сервер 300, для дальнейшей обработки, такая обработка может носить глобальный характер и может включать в себя прем данных из множества локальных вычислительных ресурсов 100, упорядочение таких множеств данных, аннотирование данных и сравнение их, например, для интерпретации данных, определения их тенденций, анализа данных на различные биомаркеры и оказания помощи в развитии диагностики, терапевтики и/или профилактики. Соответственно, в различных случаях удаленный вычислительный ресурс 300 может быть выполнен в виде концентратора обработки данных, например, когда данные из различных источников могут передаваться, обрабатываться и/или сохраняться в ожидании преобразования и/или передачи, например, благодаря доступности локальному вычислительному ресурсу 100. Более конкретно, удаленный концентратор 300 обработки может быть выполнен с возможностью приема данных из множества ресурсов 100, обработки этих данных и распределения обработанных данных обратно различным локальным ресурсам 100, чтобы обеспечить возможность сотрудничества между исследователями и/или ресурсами 100. Такое сотрудничество может включать в себя различные протоколы совместного использования и может дополнительно включать в себя подготовку данных, подлежащих отправке, например, путем предоставления пользователю системы 1 выбора среди различных протоколов защиты и/или настроек конфиденциальности для управления подготовкой данных для передачи.

[00762] В одном конкретном случае, который представлен на ФИГ. 42B, предусмотрены локальные вычислительный ресурс 100 и/или ресурс 200 хранения, например, непосредственно в месте нахождения пользователя. Вычислительный ресурс 100 и/или ресурс 200 хранения могут быть связаны с ресурсом 121 формирования данных, таким как СНП или секвенатор на микросхеме, как описано в настоящем документе, например, посредством прямого или осуществляемого по внутренней сети соединения 10, причем секвенатор 121 выполнен с возможностью формирования данных генетической последовательности, таких как файлы BCL и/или FASTQ. Например, секвенатор 121 может быть частью или находиться внутри того же устройства, что и вычислительный ресурс 100 и/или ресурс 200 хранения, чтобы иметь прямой обмен данными и/или оперативной соединение с ними, или секвенатор 121 и вычислительный ресурс 100 и/или ресурс 200 хранения могут быть частями отдельных друг от друга устройств, но находиться в одном и том же учреждении и тем самым быть связанными с помощью кабеля или соединения внутренней сети 10. В некоторых случаях секвенатор 121 может находиться не в одном учреждении с вычислительным ресурсом 100 и/или ресурсом 200 хранения и поэтому может быть соединен посредством Интернета 30 или гибридной облачной сети 50.

[00763] В таких случаях данные генетической последовательности могут

обрабатываться 100 и храниться 200 локально перед преобразованием с помощью соответствующим образом сконфигурированного преобразователя, или сформированные данные последовательности могут передаваться непосредственно в один или более преобразователей и/или анализаторов 152, например, посредством соответствующим образом сконфигурированного соединения локальной сети 10, внутренней сети 30 или гибридной облачной сети 50, как описано выше, например перед обработкой локально. В частности, подобно ресурсу 121 формирования данных, преобразователь 151 и/или анализатор 152 могут быть частью или находиться внутри того же устройства, что и вычислительный ресурс 100 и/или ресурс 200 хранения, чтобы иметь прямой обмен данными и/или оперативной соединение с ними, или преобразователь и/или анализатор 152 и вычислительный ресурс 100 и/или ресурс 200 хранения могут быть частями отдельных друг от друга устройств, но находиться в одном и том же учреждении и тем самым быть связанными с помощью кабеля или соединения внутренней сети 10. В некоторых случаях преобразователь 151 и/или анализатор 152 могут находиться не в одном учреждении с вычислительным ресурсом 100 и/или ресурсом 200 хранения и поэтому может быть соединен посредством Интернета 30 или гибридной облачной сети 50.

[00764] Например, преобразователь может быть выполнен с возможностью подготовки подлежащих передаче данных либо до анализа, либо после анализа, например, с помощью соответствующим образом сконфигурированного вычислительного ресурса 100 и/или анализатора 152. Например, анализатор152 может выполнять функцию вторичной и/или третичной обработки на данных, как описано в настоящем документе, например, для анализа сгенерированной последовательности данных с точки зрения определения ее геномных и/или экзомных характеристик 152a, ее эпигеномных характеристик 152b, любых различных представляющих интерес маркеров ДНК и/или РНК и/или индикаторов рака 152c, и их взаимосвязей с одним или более микробиомами 152d, а также один или более других процессов вторичной и/или третичной обработки, как описано в настоящем документе.

[00765] Как было указано, сформированные и/или обработанные данные могут быть преобразованы, например, с помощью соответствующим образом сконфигурированного преобразователя, например, до передачи по всей системе 1 из одного ее компонента в другой, например, по прямому соединению, соединению локальной сети 10, Интернета 30 или гибридной облачной сети 50. Такое преобразование может включать в себя одно или более из конвертации 151d, например, когда данные конвертируют из одной формы в другую; затруднения понимания 151c, включая кодирование, декодирование и/или превращение иным образом данных из непонятной формы в понятную форму; индексирования 151b, например, включая компиляцию и/или упорядочение сформированных данных из одного или более ресурсов и приведение их в состояние, пригодное для определения местоположения и/или поиска их элементов, например, с помощью сформированного индекса; и/или шифрования 151a, например, создания блокируемых и неблокируемых защищенных паролем наборов данных, например, до передачи посредством Интернета 30 и/или гибридного облака 50.

[00766] Следовательно, как показано на ФИГ. 42C, в этих и/или других таких случаях гибридное облако 50 может быть выполнено с возможностью обеспечения возможности беспрепятственной и защищенной передачи данных всем компонентам системы, например, когда гибридное облако 50 выполнено с возможностью разрешения различным пользователям системы конфигурировать ее составляющие части и/или саму систему для удовлетворения потребностей пользователей в области

исследовательских, диагностических, терапевтических и/или профилактических открытий и/или разработок. В частности, гибридное облако 50 и/или различные компоненты системы 1 могут быть выполнены с возможностью функционального соединения с совместимыми и/или соответствующими интерфейсами API, которые выполнены с возможностью разрешения пользователям удаленного конфигурирования различных компонентов системы 1 для развертывания требуемых ресурсов нужным образом, причем локальным, удаленным или комбинированным способом, например, на основе потребностей системы и особенностей выполняемых анализов, обеспечивая при этом обмен данными в защищенной среде с возможностью шифрования.

[00767] В конкретных случаях система 1 может включать в себя архитектуру 310 обработки, например, интерпретатор, который выполнен с возможностью осуществления функции 310 интерпретации. Интерпретатор 310 выполняет одну или серию аналитических функций на сформированных данных, таких как аннотирование 311, интерпретация 312, диагностика 313 и/или функция обнаружения и/или анализа для определения наличия одного или более биомаркеров, например, в генетических данных. Интерпретатор 310 может быть частью локального вычислительного ресурса 100 или отделен от него, например, когда интерпретатор 313 связана с вычислительным ресурсом 100 посредством интерфейса облака, такого как гибридное облако 50.

[00768] Кроме того, может быть включена дополнительная архитектура 320 обработки, например, когда архитектура 320 выполнена в виде коллаборатора. Коллаборатор 320 сможет быть выполнен с возможностью осуществления одной или более функций, относящихся к обеспечению безопасности и/или конфиденциальности данных, подлежащих передаче. Например, коллаборатор может быть выполнен с возможностью защиты процесса 321 совместного использования данных, обеспечения конфиденциальности передачи 322, установки параметров 323 управления и/или инициирования протокола 324 защиты. Коллаборатор 320 выполнен с возможностью обеспечения совместного использования данных, например, для облечения совместного осуществления обработки, например, коллаборатор 320 может быть частью локального вычислительного ресурса 100 или отделен от него, например, когда коллаборатор связан с вычислительным ресурсом 100 посредством интерфейса облака, такого как гибридное облако 50. Интерпретатор 310, коллаборатор 320 и/или локальный вычислительный ресурс 100 могут быть также связаны с удаленным вычислительным ресурсом 300, например, для улучшения эффективности системы за счет сброса функций вычисления 300 и/или хранения 400 на облако 50. В различных случаях система 1 может быть выполнена с возможностью разрешения выполнения защищенного анализа третьей стороной 121, например, когда третья сторона может подключаться к системе и задействовать ее, например, посредством соответствующим образом сконфигурированного API.

[00769] Как показано на ФИГ. 43, система 1 может быть многоярусной и/или мультиплексированной платформой биоаналитической обработки, которая содержит уровни блоков формирования данных и/или обработки данных, каждый из которых имеет один или более конвейеров обработки, которые могут быть развернуты систематически и одновременно или последовательно для обработки генетической информации, начиная со стадии ее первичной обработки и заканчивая стадией вторичной и/или третичной обработки. В частности, в настоящем документе представлены устройства, выполненные с возможностью осуществления биоанализа в одной или более из аппаратной, и/или программной, и/или квантовой реализаций обработки, а также способы их использования и содержащих их системы. Например, в одном варианте

реализации может быть предусмотрена платформа геномного анализа, выполненная в виде множестве интегральных схем, которые могут быть выполнены в виде или иным образом включены в одно или более из центрального или графического процессорного устройства, такого как ЦПУ и/или ГПУ общего назначения, аппаратная реализация и/или квантовое процессорное устройство. В частности, в различных вариантах реализации один или более из конвейеров платформы геномной обработки могут быть выполнены с возможностью конфигурирования с помощью одной или более интегральных и/или квантовых схем квантового процессорного устройства.

[00770] Соответственно, платформы, представленные в настоящем документе, могут быть выполнены таким образом, чтобы использовать огромную мощь оптимизированных программных, и/или аппаратных, и/или квантовых реализаций обработки для выполнения различных функций генетического секвенирования и/или вторичной и/или третичной обработки, описанных в настоящем документе, которые могут быть выполнены на одной или более интегральных схем. Такие интегральные схемы могут быть бесшовно связаны вместе и могут быть также бесшовно связаны с различными другими интегральными схемами, например, ЦПУ, и/или ГПУ, и/или КПУ системы, которая выполнена с возможностью исполнения различных программных и/или аппаратных приложений третичных биоаналитических функций.

[00771] В частности, в различных вариантах реализации эти процессы могут быть выполнены оптимизированным программным обеспечением, исполняемым на ЦПУ, ГПУ и/или КПУ, и/или могут быть реализованы в виде интегральных схем, конфигурируемых прошивкой, например, FPGA, которые могут быть частью одного и того же устройства или отдельных устройств, которые могут быть расположены на одной и той же материнской плате, разных платах PCIe внутри одного и того же устройства, отдельных устройств в одном и том же учреждении и/или в разных учреждениях. Соответственно, одно или более процессорных устройств и/или интегральных схем могут быть непосредственно связаны вместе, например, жестко, например, путем физического включения в одну и ту же материнскую плату или отдельные материнские платы, расположенные в одном и том же корпусе и/или иным образом связанные вместе, или могут быть расположены на отдельных материнских платах или платах PCIe, которые могут обмениваться данными друг с другом удаленно, например, без проводов и/или посредством сетевого интерфейса, например, через локальное облако 30, и в различных вариантах реализации одно или более процессорных устройств и/или интегральных схем могут быть расположены географически удаленно друг от друга, но обмениваться данными посредством гибридного облака 50. В конкретных случаях интегральные схемы, образующие ЦПУ, ГПУ и/или КПУ или являющиеся их частью, могут быть выполнены в виде и/или быть частью платформы вторичной и/или третичной аналитики, могут быть выполнены таким образом, чтобы образовывать один или более конвейеров анализа, где различные формируемые данные могут вводиться и выводиться вперед-назад между различными процессорными устройствами и/или интегральными схемами, например, бесшовно или в потоковом режиме, чтобы обеспечивать возможность быстрой передачи данных между множеством интегральных схем и, более конкретно, ускорять анализы в них.

[00772] Например, в некоторых случаях различные устройства для использования в соответствии со способами, описанными в настоящем документе, могут содержать или быть иным образом связанными с одним или более устройствами секвенирования для выполнения протокола секвенирования, который может быть осуществлен путем исполнения программного обеспечения на удаленном секвенаторе, таком как секвенатор

нового поколения, например, HiSeq Ten производства компании Illumina, расположенном в главном центре секвенирования, чтобы сделать его доступным посредством облачного интерфейса. В других случаях секвенирование могут выполнять в жестко смонтированной конфигурации на микросхеме секвенирования, например, реализованной в секвенаторе Ion Torrent производства компании Thermo Fisher, или в других технологиях секвенатора на микросхеме, где секвенирование осуществляется с помощью полупроводниковой технологии, которая обеспечивает настольное секвенирование следующего поколения, и/или с помощью интегральной схемы, выполненной в виде или иным образом включающей в себя полевой транзистор, использующий графеновый канальный слой. В таких случаях, когда секвенирование выполняют с помощью одной или более интегральных схем, выполненных в виде полупроводниковой микросхемы секвенирования, или содержащих ее, они могут быть расположены удаленно от одного или более процессорных устройств и/или интегральных схем, описанных в настоящем документе, который могут быть выполнены с возможностью осуществления вторичной и/или третичной аналитики на секвенированных данных. В альтернативном варианте реализации микросхемы и/или процессорные устройства могут быть расположены относительно близко друг к другу, чтобы быть напрямую связанными вместе, или по меньшей в общей близости друг от друга, например в одном учреждении. В этом и других случаях конвейеры секвенирования и/или BioIT-аналитики могут быть сформированы таким образом, чтобы необработанные данные секвенирования, сформированные секвенатором, могли быть быстро переданы, например, в потоковом режиме, в другие аналитические компоненты конвейера для непосредственного анализа, например в потоковом режиме.

[00773] Кроме того, после создания прибором для секвенирования необработанных данных секвенирования (например, данных BCL) или данных рида (например, данных FASTQ) они могут быть переданы в интегральную схему и приняты ей, где интегральная схема выполнена с возможностью осуществления различных биоаналитических функций на генетической и/или белковой последовательностях, например, с целью анализа сформированных и/или принятых данных последовательностей ДНК, РНК и/или белков. Анализ последовательности может включать в себя сравнение сформированной или принятой последовательности нуклеиновых кислот или белков с одной или более баз данных последовательностей, например, для выполнения вторичного анализа на принятых данных, и/или в некоторых случаях для выполнения диагностики заболевания, например, когда база данных известных последовательностей для выполнения сравнения может представлять собой базу данных, содержащую данных морфологически отличающихся друг от друга и/или несовместимых последовательностей, которые являются данными генетических образцов, относящихся или считающихся относящимися к одному или более болезненных состояний.

[00774] Соответственно, в различных случаях изолированные и секвенированные генетические, например, ДНК и/или РНК, данные могут быть подвергнуты вторичному анализу, который может быть выполнен на принятых данных, например, для выполнения картирования, выравнивания, сортировки, определения вариантов и т.п. с целью формирования картированных и/или выровненных данных, которые могут быть затем использованы для получения одного или более файлов VCF, подробно описывающих разницу между картированной и/или выровненной генетической последовательностью и референсной последовательностью. В частности, по завершении вторичной обработки генетическая информация может быть передана в один или более модулей третичной обработки системы, например, для дальнейшей обработки там, например, для получения

результатов, касающихся терапии и/или профилактики. Более конкретно, после определения вариантов сопоставитель/выравниватель/определитель вариантов могут выдать стандартный файл VCF, который готов и может быть передан в дополнительную интегральную схему для выполнения третичного анализа , такого как анализы, относящиеся к геному, например, полногеномный анализ, анализ генотипирования, например, совместного генотипирование, микроматричный анализ, анализ экзома, анализ микробиома анализ эпигенома, анализ метагенома, анализ совместного генотипирования, анализ вариации, например, анализ GATK, анализ структурных вариантов, анализ соматических вариантов и т.п., а также анализ секвенирования РНК и другие геномные анализы.

[00775] Следовательно, биоаналитическая, например, BioIT, платформа, представленная в настоящем документе, может содержать высокооптимизированные алгоритмы для картирования, выравнивания, сортировки, маркировки дубликатов, определения вариантов гаплотипов, сжатия и/или распаковки, например, в конфигурации жестко смонтированной и/или квантовой обработки. Например, хотя одна или более из этих функций могут быть выполнены с возможностью осуществления в полностью или частично жестко смонтированной конфигурации, в конкретных случаях платформа вторичной и/или третичной обработки может быть выполнена с возможностью выполнения одного или более программного приложения и/или приложения квантовой обработки, например, одной или более функций, описанных в настоящем документе. В частности, секвенированные, и/или картированные, и/или выровненные, и/или иным образом обработанные данные могут быть затем подвергнуты дополнительной обработке с помощью одного или более оптимизированных алгоритмов для одного или более из полногеномного анализа, анализа генотипирования, микроматричного анализа, анализа экзома, анализа микробиома, анализа эпигенома, анализа метагенома, анализа совместного генотипирования и/или анализа вариантов, например, GATK, например, реализованного с помощью программного обеспечения, выполняемого на ЦПУ, и/или ГПУ, и/или КПУ общего назначения, хотя и в определенных случаях одна или более из этих функций могут быть, по меньшей мере частично, реализованы в аппаратном обеспечении.

[00776] Соответственно, как показано на ФИГ. 43, в различных вариантах реализации мультиплексированные платформы биоаналитической обработки выполнены с возможностью осуществления одной или более из первичной, вторичной и/или третичной обработки. Например, на стадии первичной обработки создают данные генетической последовательности, например, в одном или более файлов BCL и/или FASTQ, для передачи в систему 1. Оказавшись в системе 1, секвенированные данные, включая связанные метаданные, могут быть продвинуты на стадию 600 вторичной обработки для создания одного или более файлов определения вариантов. Следовательно, система может быть также выполнена с возможностью взятия одного или более файлов определения вариантов вместе со связанными метаданными и/или другими связанными обработанными данными, и выполнения одной или более других операций на одной или более стадиях третичной обработки, например, с целью выполнения на них одной или более диагностических, и/или профилактических, и/или терапевтических процедур.

[00777] В частности, анализ данных может быть инициирован, например, в ответ на запрос 120 пользователя, например, сделанный из удаленного вычислительного ресурса 100, и/или в ответ на данные, поданные третьей стороной 121, и/или данные, автоматические извлеченные из локального хранилища 200 и/или удаленного хранилища 400. Такая дальнейшая обработка может включать в себя первый ярус обработки, где

различные конвейеры, выполняющие протоколы 700, выполнены с возможностью осуществления аналитики на определенных генетических данных, например, вариации, одного или более субъектов. Например, устройства первого яруса третичной обработки могут включать в себя платформу геномной обработки, которая выполнена с возможностью осуществления анализа генома, эпигенома, метагенома, генотипирования и/или различных анализов вариантов, и/или другого анализа на основе биоинформатики. Кроме того, на втором ярусе третичной обработки могут выполняться различные диагностирующие заболевания, исследовательские и/или аналитические протоколы 800, причем анализ может включать в себя одно или более из приложений, относящихся к НИПТ, ОРИТН, раку, LDT, биологии, AgBio и т.п.

[00778] Система 1 может быть также выполнена с возможностью приема и/или передачи различных данных 900, относящихся к процедурам и процессам, описанным в настоящем документе, например, имеющим отношение к данным электронных медицинских записей (EMR), данным испытаний и/или структурирования Федерального управления по лекарственным средствам (США), относящимся к аннотации данными и т.п. Такие данные могут быть полезными ввиду предоставления пользователю возможности использования и/или доступа к сформированным медицинским, диагностическим, терапевтическим и/или профилактическим методам, разработанным посредством использования системы 1 и/или доступны с ее помощью. Соответственно, в различных случаях устройства, способы и системы, представленные в настоящем документе, позволяют безопасно выполнять генетический и биоаналитический анализ, а также безопасно передавать его результаты на форум, который может быть легко доступным для обработки далее по цепочке. Кроме того, в различных случаях устройства, способы и системы, представленные в настоящем документе, позволяют безопасно передавать данные в систему, например, из одного или более учреждений мониторинга здоровья и/или хранения данных и/или правительственного учреждения, такого как FDA или NIH. Например, система может быть выполнена с возможностью безопасного приема данных электронной медицинской записи/персонального учета здоровья (EMR/PHR), например, которые могут быть переданы из учреждения и/или хранилища здравоохранения, для использования в соответствии со способами, описанными в настоящем документе, например, для выполнения генетического и биоаналитического анализа, а также безопасной передачи его результатов на форум, который может быть легко доступным для обработки далее по цепочке.

[00779] В частности, первый ярус 700 третичной обработки может включать в себя одну или более платформ геномной обработки, например, для выполнения генетического анализа, например, на картированных и/или выровненных данных, например, в формате файла SAM или BAM, и/или для обработки данных вариантов, например в формате VCF. Например, первая платформа третичной обработки может включать в себя одну или более из конвейера генома, конвейера метагенома, конвейера совместного генотипирования, а также один или более конвейеров анализа вариантов, в том числе: конвейер GATK, конвейер структурных вариантов, конвейер определения соматических вариантов и, в некоторых случаях, может включать в себя конвейер анализа секвенирования РНК. Также могут быть включены один или более конвейеров геномного анализа.

[00780] Точнее говоря, как показано на ФИГ, 43, в различных случаях многоярусная и/или мультиплексированная платформа биоаналитической обработки включает в себя дополнительный уровень устройств формирования и/или обработки данных. Например, в определенных случаях платформа биоаналитической обработки содержит один или

более конвейеров обработки в одной или более программных и/или аппаратных реализаций, которые относятся к выполнению одного или более протоколов третичной обработки. Например, в конкретных случаях платформа конвейеров 700 третичного анализа может включать в себя один или более из конвейеров генома, конвейера эпигенома, конвейера метагенома, конвейера совместного генотипирования, конвейера вариации, такого как конвейер GATK, и/или другие конвейеры, такие как конвейер РНК. Кроме того, второй уровень платформы анализа третичной обработки может включать в себя ряд конвейеров обработки, таких как один или более из конвейера микроматричного анализа, конвейера анализа генома, например, полногеномного анализа, конвейера анализа генотипирования, конвейера анализа экзома, конвейера анализа микробиома, конвейера анализа генотипирования, включая совместное генотипирование, конвейера анализа вариантов, включая конвейеры структурных вариантов, конвейеры соматических вариантов, и конвейеры GATK и/или MuTect2, а также конвейеры секвенирования РНК и конвейеры других генетических анализов.

[00781] Соответственно, в одном варианте реализации многоярусная платформа биоаналитической обработки включает в себя метагеномный конвейер. Например, метагеномный конвейер может быть включен, например, для выполнения одного или более процессов экологической геномики. В частности, в различных вариантах реализации метагеномный анализ может быть выполнен с возможностью определения, развилась ли группа организмов от общего предка, например, вид или другая клада. Более конкретно, в различных вариантах реализации может быть получен экологический образец, содержащий множество живых и/или мертвых организмов, из которого можно изолировать присутствующие ДНК/РНК, секвенировать и обработать с помощью одной или более платформ обработки, описанных в настоящем документе, чтобы идентифицировать присутствующий вид и/или один или более других геномных факторов, относящихся к нему. Такие «экологические» образцы могут включать в себя множество микробиомов человека (например, относящихся к микроорганизмам, которые обнаруживают в связи со здоровыми и больными людьми, в том числе микроорганизмы, обнаруживаемые в образцах кожи, крови, мокроты, стула), а также внешних факторов окружающей среды.

[00782] Существуют множество способов получения секвенированных генетических образцов для выполнения метагеномной обработки. Первый способ включает в себя протокол направленного клонирования 16S рибосомальной РНК и/или секвенирование генов. Например, 16S рибосомальная РНК сильно варьируется в виде (или даже штаммах вида). Соответственно, эта РНК может быть изолирована и секвенирована для создания генетического профиля биоразнообразия, получаемого из биологических образцов естественного происхождения, который может быть использован для информирования ИИ или других баз данных системы. Однако проблема с таким секвенированием заключается в том, что большая часть микробиального биоразнообразия может быть утеряна просто вследствие способа, которым оно было культивировано.

[00783] Соответственно, второй способ включает в себя протокол методом «дробовика» и/или ориентированный на ПЦР протокол, который может быть использован для формирования образцов множества, например, всех, генов из всех биологических факторов сообществ, из которых взяты образцы, причем однократное секвенирование может выявить генетическое разнообразие микроскопической жизни. В частности, при секвенировании методом «дробовика» может быть сформирована агрегированная референсная последовательность, например, из множества (например, десятков тысяч) референсных геномов различных видов. Однако совокупный размер

этих множественных геномов гигантский. Поэтому для построения агрегированного референсного генома целесообразно выбирать одну или более отличающихся друг от друга подпоследовательностей из каждого референсного генома.

[00784] Например, такая подпоследовательность может быть длиной от нескольких сотен оснований до нескольких тысяч оснований, причем идеально подходят уникальные последовательности, отсутствующие в других видах (или штаммах). Эти подпоследовательности могут быть затем агрегированы для построения референсных последовательностей. Соответственно, изолированные, секвенированные, картированные и выровненные метагеномные последовательности можно сравнить с частичными или полными референсными геномами для многих видов и определить биоразнообразие.

[00785] Таким образом, метагеномика предоставляет мощное «увеличительное стекло» для просмотра мира микробов, которое может коренным образом изменить наше понимание живого мира. Следовательно, в любом из этих двух случаев, когда в образце значительное присутствие ДНК организмов, эти виды могут быть идентифицированы как обитающие в данной среде. В идеале подобным образом можно выявить виды, которые необычны для видов, как правило, присутствующих в данной среде. В частности, когда для полученных экологических проб охват всех видов нормализован, можно определить генетическое многообразие присутствующих видов и сравнить со всем охватом, например, путем сравнения части ДНК определенного организма с соответствующей частью сформированной биологически многообразной референсной генетической последовательностью.

[00786] Значимость этих анализов можно определить с помощью байесовских методов, например, путем оценки вероятности наблюдения секвенированных ридов конкретного организма в предположении присутствия или отсутствия данного организма. Методы байесовской вероятности направлены на описание вероятности события на основе условий, которые могли быть связаны с этим событием. Например, если требуется определить наличие рака у субъекта, и возраст субъекта известен, а заболевание раком, который нужно определить, связано с возрастом, то с помощью теоремы Байеса информацию о возрасте субъекта можно использовать для более точной оценки вероятности рака.

[00787] А именно, с помощью интерпретации байесовской вероятности теорема выражает, как субъективная степень уверенности может рационально измениться с учетом наблюдаемых доказательств. Байесовская теорема математически сформулирована в следующем уравнении: $P^{(A/B)} = P^{(B/A)} P^{(A)}/P^{(B)}$, где A и B являются событиями и $P(B) \neq 0$. P(A) и P(B) - вероятности наблюдения A и B безотносительно друг друга. P(A|B) - условная вероятность, являющаяся вероятностью наблюдения события A при условии, что B истинное. P(B|A) - вероятностью наблюдения события B при условии, что A истинное.

[00788] Соответственно, один или более этапов выполнения анализов байесовской вероятности в данном контексте могут включать в себя одно или более из следующего: можно выполнить определения наличия клад на различных таксономических уровнях: царство, тип, класс, отряд, семейство, род, вид и/или штамм. Однако это осложняется тем, что, как правило, чем ниже таксономические уровни, занимаемые организмами, тем более похожи их ДНК. Кроме того, часто проба может совпадать с референсным геномом для множества видов с более высоким таксономическим уровнем (или множестве штаммов одного вида), и, следовательно, во многих случаях может быть

определена только более общая клада (такая как род или семейство), а не конкретный вид или штамм. Тем не менее, для преодоления этих и других таких трудностей можно использовать устройства, системы и способы их использования, описанные в настоящем документе.

[00789] А именно, в одном варианте реализации предусмотрен способ определения присутствия двух или более видов или клад организмов в предоставленной пробе. Например, на первом этапе из пробы можно получить риды данных геномной последовательности, например, когда риды могут быть в формате FASTQ или BCL. Можно выполнить картирование геномной поверхности, чтобы картировать риды на множество геномных референсных последовательностей. В данном случае геномные референсные последовательности могут быть полным геномом, или могут быть частичным геномом, чтобы сократить объем данных, требуемых для каждого вида, штамма или клады. Однако использование более крупного генома повысит чувствительность обнаружения, и каждую используемую референсную последовательность следует выбирать для представления вида, штамма или клады, которые будут отличаться друг от друга.

[00790] Для этой цели можно использовать всю или часть геномной последовательности из 16S рибосомы. Таким образом, можно построить две или более геномные референсные последовательности видов, штаммов или клад организмов, наличие которых предполагается в образце, чтобы обнаружить членов этих групп в образце. После построения геномных референсных последовательностей для каждой из них можно также создать индекс. Индексы могут быть хэш-таблицами или древовидными индексами, такими как древовидный индекс префиксов или суффиксов. После того, как индекс построен, риды геномной последовательности образца можно сравнить с каждым из двух или более индексов. После этого можно определить, картируются ли риды геномной последовательности образа на каждый из индексов.

[00791] Аналогичным образом риды геномной последовательности можно также выровнять на геномные последовательности, на которые они картирована. В результате будет сформирована оценка выравнивания в соответствии со способами, описанными в настоящем документе, которую можно использовать при анализе вероятности того, что рид указывает на присутствие или отсутствие вида или клады организма в образце. В частности, картирование и/или выравнивание можно осуществит с помощью представленных программных и/или аппаратных модулей, как описано в настоящем документе. В некоторых вариантах реализации картированные и выровненные данные могут быть затем переданы вычислительный ресурс 100/300 для дальнейшего анализа и обработки.

[00792] Например, картированные и/или выровненные риды геномной последовательности могут быть проанализированы для определения правдоподобия того, что организм, имеющий данную геномную референсную последовательность, присутствует в образце. Аналогичным образом может быть сообщен список видов, штаммов или клад, присутствие которых в экологической пробе определено. В определенных вариантах реализации список может быть сообщен вместе метрикой достоверности (например, Р-значением), чтобы указать статистическую достоверность оценки. Может быть также сообщен полный список проанализированных видов, штаммов или клад организмов вместе с указанием каждого вида, штамма или клады, которые присутствовали, и метрики достоверности. Необходимо отметить, что различные методы и процедуры, описанные в настоящем документе, хотя они описаны применительно к анализу микробиомов, могут быть использованы в анализе всех других

протоколов третичной обработки, где уместно.

[00793] Например, на ФИГ. 43В показан пример реализации способа выполнения экологического анализа, например, микробиомов в экологическом образце. Например, в первом случае можно получить экологический образец и из него изолировать разнообразный генетический материал. Этот разнообразный материал можно затем обработать и секвенировать, например, с помощью подходим образом сконфигурированного СНП.

[00794] В результате на первом этапе 1000 после того, как разнообразный генетический материал секвенирован, например, с помощью СНП, его можно передать в систему 1, описанную в настоящем документе. На этапе 1010 можно построить одну, две или более геномных референсных последовательностей, представляющих интерес, например, которые нужно обнаружить в образце. На этапе 1020 можно построить индекс для одной, двух или более геномных референсных последовательностей. Далее, на этапе 1030 полученные секвенированные риды геномного образца можно сравнить с одним, двумя или более индексами, например, с помощью соответствующим образом сконфигурированного модуля картирования. На этапе 1040 можно определить, картируются ли секвенированные риды геномного образа на каждый из двух или более индексов.

[00795] В это время, если требуется, на этапе 1050 картированные риды можно выровнять с геномными референсными последовательностями, чтобы сформировать выравнивание и/или оценку выравнивания. Соответственно, после того, как полученные генетические материалы в образце картированы и/или выровнены, на этапе 1060 можно определить правдоподобие того, что данный организм, имеющий референсную последовательность, присутствует в образце. И после обработки можно определить и/или сообщить список видов, штаммов и/или клад, которые присутствуют в образце.

[00796] Платформа третичной обработки, описанная в настоящем документе, может быть также включена в эпигеномный конвейер. В частности, эпигенетика изучает генетические воздействия, которые не кодированы в последовательности ДНК организма Этот термин относится также с самим изменениям: относящимся к функциональным свойствам изменения в геноме, которые не влекут за собой изменения в нуклеотидной последовательности. Тем не менее, эпигенетические изменения являются стабильно наследуемыми фенотипами в результате изменений в хромосоме, которые не изменяют последовательность ДНК, Эти изменения могут быть наследуемыми или ненаследуемыми. В частности, эпигенетические изменения модифицируют активацию определенных генов, но не генетический код последовательности ДНК. Модифицированы могут быть именно сама микроструктура (не код) ДНК или связанные белки хроматина, вызывающие активацию или умолкание.

[00797] Эпигеном участвует в регулировании экспрессии генов, развитии. дифференциации тканей и подавлении транспозонов. В отличие от лежащего в основе генома, который в значительной мере статичен у индивида, эпигеном может динамически изменяться условиями окружающей среды. Эта область аналогична геномике и протеомике, которые изучают геном и протеом клетки. Кроме того, эпигеномика включает в себя изучение полного набора эпигенетических модификаций на генетическом материале клетки, известном как эпигеном, состоящий из записи химических изменений в ДНК и гистоновые белки организма. Эти изменения могут передаваться потомкам организма посредством трансгенерационного эпигенетического наследования. Изменения в эпигеноме могут привести к изменениям структуры хроматина и изменениям функции генома.

[00798] Данный эпигенетический механизм позволяет дифференцированным клеткам в многомолекулярном организме выражать только гены, необходимые для собственной активности. При делении клеток эпигенетические изменения сохраняются. В частности, большинство эпигенетических изменений могут происходить только в течение срока жизни отдельного организма. Однако, если инактивация гена происходит в сперматозоидах или яйцеклетках, которые приводят к оплодотворению, то некоторые эпигенетические изменения могут быть переданы следующему поколению. В том, что стало известно как клеточная память, могут участвовать несколько типов систем эпигенетического наследования. Например, различные ковалентные модификации либо ДНК (например, метилирование и гидроксиметилирование цитозина) или гистоновых белков (например, ацетилирование лизина, метилирование лизина и аргинина, фосфорилирование серина и треонина и убиквитинирование и сумоилирование лизина) могут играть центральные роли во многих типах эпигенетического наследования. Поскольку на фенотип клетки или индивида влияет, какие из их генов траскрибируются, наследуемые состояния транскрипции могут привести к эпигенетическими последствиям. Такие последствия для клеточных и физиологических фенотипических признаков могут быть результатом внешних или экологических факторов, которые включают и выключают гены и влияют на то, как клетки экспрессируют гены.

[00799] Например, повреждение ДНК может вызвать эпигенетические изменения. ДНК повреждаются очень часто. Эти повреждения в значительной степени репарируются, но в месте репарации ДНК эпигенетические изменения могут сохраниться. В частности, двухнитевый разрыв ДНК может инициировать незапрограммированное эпигенетическое умолкание гена, как за счет вызова метилирования ДНК, так и за счет стимулирования умолкания типов модификаций гистона (ремоделирования хроматина). К другим примерам механизмов, которые создают такие изменения, можно отнести метилирование ДНК и модификацию гистона, каждое из которых изменяет экспрессирование генов без изменения лежащей в основе последовательности ДНК, Было установлено, что ремоделирование нуклеосомы вызывает эпигенетическое умолкание репарирования ДНК. Более того, повреждающие ДНК химикаты могут также вызывать значительное гипометилирование ДНК, например, за счет активации путей окислительного стресса. Кроме того, экспрессия генов может управляться действием репрессорных белков, которые прикрепляются к областям глушителя ДНК.

[00800] Эти эпигенетические изменения могут длиться в течение делений клетки на протяжении всей жизни клетки, и могут также длиться в течение множества поколений, даже если они не влекут изменений в лежащей в основе последовательности ДНК организма; зато негенетические факторы заставляют гены организма менять поведение (или «экспрессировать себя»). Одним из примеров эпигенетического изменения в эукариотической биологии является процесс клеточной дифференциации. Во время морфогенеза тотипотентные стволовые клетки становятся различными плюрипотентными клеточными линиями эмбриона, которые, в свою очередь, становятся полностью дифференцированными клетками. Другими словами, по мере того, как одна оплодотворенная яйцеклетка - зигота - продолжает делиться, получающиеся в результате дочерние клетки превращаются во все различные типы клеток в организме, включая нейроны, мышечные клетки, эпителий, эндотелий кровяных сосудов и т.д., за счет активирования некоторых генов при подавлении экспрессии других генов.

[00801] Существуют несколько уровней регуляции экспрессии генов. Один путь регуляции генов лежит через ремоделирование хроматина. Хроматин представляет собой комплекс ДНК и гистоновых белков, с которыми она связана. Если порядок

обертывания ДНК вокруг гистонов изменяется, экспрессия генов тоже может измениться. Во-первых это происходит вследствие посттрансляционной модификации аминокислот, которая компенсирует нехватку гистоновых белков. Гистоновые белки слагаются из длинных цепочек аминокислот. Если аминокислоты в цепочке изменяются, форма гистона может быть модифицирована. Во время репликации ДНК не полностью разматываются. Поэтому возможно, что модифицированные гистоны могут проникнуть в каждую новую копию ДНК. Оказавшись там, эти гистоны могут действовать в качестве шаблонов, инициирующих окружение новых гистонов с образованием новой формы. Благодаря изменению формы гистонов вокруг них, эти модифицированные гистоны обеспечат поддержание программы линиеспецифической транскрипции после деления клетки.

[00802] Во-вторых, это происходит за счет добавления метиловых групп в ДНК, в основном на сайтах CpG, для преобразования цитозина в 5-метилцитозин. 5-метилцитозин действует во многом подобно нормальному цитозину, спариваясь с гуанином в двухнитевой ДНК. Однако, некоторые области генома метилируются сильнее других, и сильно метилированные области, как правило, отличаются меньшей транскрипциональной активностью за счет механизма, который не до конца понятен. Метилирование цитозинов может также переходить из зародышевой линии одного из родителей в зиготу с образованием хромосомы, которая наследуется от одного родителя или другого (генетический импринтинг). Хотя модификации гистона происходят по все последовательности, неструктурированные N-концы гистонов (называемые гистоновыми хвостами) модифицируются особенно сильно. В число этих модификаций входят ацетилирование, метилирование, убиквитинирование, фосфорилирование, сумоилирование, рибозилирование и цитруллинирование.

[00803] Соответственно, метилирование ДНК - это присутствие метиловой группы на некоторых нуклеотидах ДНК, особенно на основаниях «G» или динуклеотидах «CpG». Метилирование в областях промотора, как правило, подавляет экспрессию генов. Анализ метилирования представляет собой процесс обнаружения того, какие основания «C» метилированы в геноме данного образца. Бисульфитное секвенирование (MethylC-seq) является наиболее распространенным способом обнаружен метилирования с использованием секвенирования всего генома, причем неметилированные основания цитозина («C») химически преобразуются в основания урацила («U»), которые становятся основаниями тимина «T» после ПЦР-амплификации. Метилированные основания «C» устойчивы к преобразованию.

[00804] Поэтому в соответствии с устройствами и способами, описанными в настоящем документе, предлагается обнаружение модификаций молекул ДНК, где модификации не влияют на последовательность ДНК, но сказываются на экспрессии генов, например, путем выполнения одной или более операций картирования и/или выравнивания на эпигенетическом генетическом материале. В таких способах полученные риды могут быть картированы и выровнены на референсный геном таким образом, который допускает выравнивание преобразованных оснований «T» на позиции «C» референса, а перед картированием/выравниванием основания «C» в референсной последовательности могут быть заменены основаниями «T». Это позволяет точно картировать и выравнивать риды, у которых основания C преобразованы бисульфитом (теперь T), в результате чего в ридах геномной последовательности обнаруживаются не преобразованные сульфитом (метилированные) основания C. Для обратно-комплементарных выравниваний можно использовать комплементарные замены, например, все «G» можно заменить на «A».

[00805] Аналогичным образом построитель индекса референса (например, хэш-таблицы) и сопоставитель/выравниватель могут быть модифицированы для выполнения этих замен автоматически для использования секвенирования MethylC-seq. В альтернативном варианте реализации сопоставитель/выравниватель могут быть модифицированы для обеспечения возможности прямого выравнивания «T» рида на «C» референса и обратно-комплементарного выравнивания «A» рида на «G» референса. Способы, описанные в настоящем документе, улучшают точность и предотвращают ошибочное прямое выравнивание «C» рида на «T» референса или ошибочное обратно-комплементарное выравнивание «G» рида на «A» референса.

[00806] Кроме того, в настоящем документе предложены способы для определения состояния метилирования оснований цитозина в ридах геномной последовательности. Например, на первом этапе можно получить риды геномной последовательности из образцов с нуклеотидами, обработанными бисульфитом. В этой связи, в частности, для формирования рида для вторичной обработки можно использовать один или более модифицированных протоколов. А именно, для выявления метилирования ДНК по всем частям генома при меняющихся уровнях разрешения до уровня пар оснований можно использовать одно или более из: бисульфитного секвенирования при сниженном представительстве; секвенирования посредством иммунопреципитации метилированной ДНК; и секвенирование с использованием метил-чувствительных рестриктаз. Кроме того, можно получить доступ к хроматину там, где он доступен, например, когда можно выполнить секвенирование сайта гиперчувствительности к ДНКазе I, например, когда с помощью фермента ДНКазы I можно найти открытые или доступные области в геноме. Кроме того, можно использовать матрицы секвенирования и экспрессии РНК для выявления уровней экспрессии генов, кодирующих белки. В частности, для определения экспрессии малых некодирующих ДНК, прежде всего миРНК, можно использовать секвенирование ммРНК.

[00807] Следовательно, после секвенирования для создания ридов можно построить геномную референсную последовательность для сравнения с ридами. Затем можно отметить местоположения CpG в геномной референсной последовательности. Далее, можно предварительно обработать геномную референсную последовательность, заменив в ней все «C» на «T». Для геномной референсной последовательности можно построить индекс. И после того, как индекс построен, риды геномной последовательности образца можно сравнить с индексом и определить, картируются ли риды эпигеномной последовательности на индекс.

[00808] Далее, картированные риды можно выровнять с геномной референсной последовательностью, чтобы сформировать оценку выравнивания. В определенных вариантах реализации можно выполнить замену оснований в последовательности рида, и рид можно снова сравнить и повторно выровнять с индексом. В некоторых вариантах реализации во время картирования и/или выравнивания рида можно использовать ограничение ориентации выравнивания, таким образом, чтобы разрешены были только прямое выравнивание с заменами C на T в риде и геномной референсной последовательности, и только обратно-комплементарные выравнивание с заменами G на A в риде и референсной геномной последовательности.

[00809] Эти процедуры картирования и выравнивания можно осуществит с помощью различных программных и/или аппаратных модулей, описанных в настоящем документе. В некоторых вариантах реализации картированные и выровненные данные можно затем передать в ЦПУ/ГПУ/КПУ для дальнейшего анализа и обработки. Например, картированные и выровненные риды можно отсортировать по позиции их картирования

на референс. В некоторых вариантах реализации можно маркировать и удалить дубликаты. Для каждого отмеченного местоположения CpG в референсе можно проанализировать перекрывающиеся риды из скопления ридов. В таком случае тиамин (T), который заменил цитозин (C) указывает на неметилированный цитозин и помечен как таковой. А цитозин, который остается в последовательности рида, может быть помечен как метилированный цитозин. Можно также пометить обратно-комплементарные выравнивания CpG как метилированные или неметилированные. Например, гуанин (G), который заменил аденин (A), помечают как обратный комплемент неметилированного цитозина (C), тогда как гуанин (G), который остается в последовательности рида, помечают как обратный комплемент метилилированного цитозина (C). Может быть сообщено вероятное состояние метилирования каждого местоположения CpG на каждой нуклеотидной нити, и можно создать связанную метрику достоверности (например, p-значения) определении метилирования. В некоторых вариантах реализации можно также указать состояние метилирования отмеченных местоположений CpG для каждой хромосомы диплоидной пары хромосом.

[00810] Что касается модификации гистона, она включает в себя различные возникающие естественным образом химические модификации гистоновых белков, вокруг которых обернута ДНК, приводящие к более или мене плотному обвертыванию ДНК. Неплотно обернутая ДНК, например, связана с более высокими скоростями экспрессии генов. Такие модификации гистонов можно определить методом секвенирования после иммунопреципитации хроматина (ChIP-Seq), который можно использовать для выявления по всему геному паттернов модификаций гистонов, например, с использованием антител к модификациям. Кроме того, ChIP-seq является способом, который можно использовать для изолирования и секвенирования ДНК, которая тесно связана с гистонами (или другими выбранными белками). После выполнения ChIP-seq можно подготовить образец, изолировать и секвенировать ДНК, и секвенированную ДНК можно затем картировать/выровнять на референсный геном, как описано в настоящем документе, и картированное покрытие можно использовать для выводов об уровне связывания гистонов на различных локусах в геноме. Кроме того, в настоящем документе предложены способы анализа полученных методом ChIP нуклеотидных последовательностей, который аналогичны способам, описанным ниже для анализа структурных вариантов.

[00811] Особо следует отметить, что эпигенетика полезна в исследованиях и диагностике рака. Например, опухоли человека подвергают сильному разрушению метилировнием ДНК и исследуют паттерны модификаций гистонов. Фактически аберрантный эпигенетический ландшафт раковой клети характеризуется глобальным геномным гипометилированием, гиперметилированием промотора CpG-островков генов-супрессоров опухоли, измененным кодом гистонов для критических генов и глобальной потерей моноацетилированного и триметилированного гистона H4. Соответственно, способы, описанные в настоящем документе, можно использовать в целях исследования и/или диагностики рака.

[00812] Кроме того, способы, описанные в настоящем документе, можно использовать для формирования одной или более эпигеномных баз данных и/или референсных геномов. Например, описанные в настоящем документе способы, например, использования протоколов обучения ИИ системы, могут быть полезны для формирования референса эпигеномов для человека, например, с использованием нормальных, здоровых индивидов по всему широкому спектру клеточных линий, первичных клеток и/или первичных тканей. После создания такие данные могут быть

использованы для улучшения протоколов картирования и/или выравнивания, описанных в настоящем документе. Кроме того, после формирования базы данных эпигеномных различий в ней можно выполнить интеллектуальный анализ, например, с помощью модуля ИИ, чтобы лучше охарактеризовать и определить соответствующие факторы, которые имеют место в различных болезненных состояниях, таких как рак, болезнь Альцгеймера и другие неврологические заболевания.

[00813] Соответственно, в различных случаях можно выполнить эпигеномный анализ, чтобы выявить одну или более или полный набор эпигенетических модификаций, которые произошли на генетическом материале клетки. В частности, с помощью способов, описанных в настоящем документе, можно определить эпигеном организма и/или его клеток, чтобы каталогизировать и/или записать химические изменения в ДНК и гистоновых белках клеток организма. Например, пример эпигеномного анализа показан в настоящем документе на ФИГ. 43С.

[00814] Например, на первом этапе можно получить геномный образец из организма, изолировать из него генетический материал и секвенировать. В результате после секвенирования на этапе 1000 секвенированные риды образца можно передать в систему 1, где он будет принят. В данном случае риды могут быть получены из обработанных бисульфитом нуклеотидов образца. Аналогичным образом на этапе 1010 можно построить геномный референс последовательностей, например, для организма, например, для выполнения сравнения эпигеномных ридов образца. На этапе 1012 можно выявить любые различные местоположения CpG в геномных референсных последовательностях.

[00815] После выявления на этапе 1014 «С» в местоположениях CpG в референсе можно заменить на «Т», и на этапе 1020 можно сформировать индекс для модифицированной геномной референсной последовательности. После формирования индекса для модифицированного референса на этапе 1030 можно сравнить риды геномной последовательности образца с индексом, а на этапе 1040 можно определить, картируются ли риды геномной последовательности образца на индекс, например, картируются в соответствии со способами и устройствами, описанными в настоящем документе. Картированные риды можно затем выровнять с геномной референсной последовательностью и сформировать оценку выравнивания, например, путем выполнения одной или более операций выравнивания, рассмотренных в настоящем документе.

[00816] На этой стадии можно выполнить один из множества различных анализов. Например, на этапе 1051, если требуется больше контекста, можно скорректировать замены оснований в ридах, которые обработаны выше, и/или ориентацию совмещения, и/или ограничения параметров и можно повторить этапы сравнения 1030-1050. Сам этот процесс можно повторять по мере надобности до тех по, пока не будет достигнут достаточный уровень контекста. Соответственно, после достижения достаточного уровня контекста картированные и/или выровненные риды на этапе 1080 можно отсортировать, например, в процессе, описанном в настоящем документе, по картированной/выровненной позиции референса. На этапе 1081 можно маркировать и/или удалить любые дубликаты ридов.

[00817] Далее, на этапе 1082 можно проанализировать риды из скопления ридов, перекрывающих каждое отмеченное местоположение CpG референса. Там, где «Т» заменено на «С», на этапе 1083 можно отметить как неметилированный «С»; а там, где «С» остается в последовательности, на этапе 1084 «С» можно отметить как метилированный «С». Наконец, на этапе 1086 можно также выполнить определение/

составление отчета о вероятном состоянии метилирования каждого местоположения CpG на каждой нуклеотидной нити, а также достоверности определения метилирования.

[00818] Кроме того, в настоящем документе предложены способы анализа геномного материала, где часть генетического материала может иметь структурный вариант или быть иным образом связанной с ним. В частности, структурная вариация является вариацией в структуре хромосомы организма. Структурные вариации включают в себя множество видов вариаций в геном вида, в том числе микроскопические и субмикроскопические типы, такие как делеции, дупликации, вариации числа копий, инсерции, инверсии и транслокации. Многие структурные варианты связаны с генетическими болезнями. Действительно, около 13% генома человека определено как структурный вариант в нормальной популяции, и имеются по меньшей мере 240 генов, которые существуют в виде полиморфизмов гомозиготной делеции в человеческих популяциях. Такие структурные вариации могут содержать миллионы нуклеотидов гетерогенности в каждом геноме, и, вероятно, вносят важный вклад в склонность к заболеваниям человека.

[00819] Вариация числа копий представляет собой большую категорию структурной вариации, которая включает в себя инсерции, делеции и дупликации. Существуют несколько версий известных тем, что они связаны с болезнью человека. Например, рекуррентная инверсия 400kb в гене фактора VIII является распространенной причиной гемофилии A, а меньшие инверсии, влияющие на идуронат-2-сульфатазу, вызовут синдром Хантера. В число других примеров входят синдром Ангельмана и синдром Сотоса. Наиболее распространенными типами сложной структурной вариации являются нетандемные дупликации, где последовательность дублирована и вставлена в инвертированной или прямой ориентации в другую часть генома. Другой класс сложного структурного варианта включает комбинации делеция-инсерция-делеции, дупликация-инсерция-дупликации и тандемные дупликации с вложенными делециями. Существуют также криптические транслокации и сегментная однородительская дисомия (UPD).

[00820] Однако обнаружение аномальных структур ДНК проблематично и выходит за рамки определений вариантов, известных до настоящего времени. В число структурных вариантов, которые трудно обнаруживать, входят те, которые имеют: большие инсерции и делеции (например, инделы размером свыше 50-100 пар оснований); дупликации и другие вариации числа копий (CNV); инсерции и транслокации вместе с анеуплоидией (аномальное количество копий хромосомы: моносомия, дисомия, трисомия и т.д.). В определенных случаях, описанных в настоящем документе, выявленные вариации количества копий можно проверить субъектах, не имеющих генетических болезней, например, с помощью количественного генотипирования ОНП.

[00821] Обнаружение структурной вариации обычно начинают с выполнения операции картирования выравнивания с использованием устройств и способов, описанных в настоящем документе. Например, риды геномного образца, подлежащие анализу, могут быть картированы и выровнены на референсный геном, например, в протоколе, который поддерживает химерные выравнивания. А именно, некоторые структурные варианты (например, CNV и анеуплоидия) могут быть обнаружены с помощью анализа относительно картированного покрытия. Однако другие структурные варианты (например, большие инделы, инверсии, транслокации) могут быть обнаружены с помощью анализа обрезанных и химерных выравниваний.

[00822] А именно, каждый структурный вариант включает в себя одну или более позиций «разрыва», где рид не картируется на референсный геном, например, когда геометрия изменяется между образцом и референсом. В таком случае скопление можно

сконфигурировать таким образом, чтобы риды в нем, которые слегка перекрывают разрывы структурного варианта, могли быть обрезаны в месте разрыва, а риды, существенно перекрывающие разрывы структурного варианта, могли быть химерически выровнены. Однако, пары ридов, перекрывающих разрывы структурных вариантов, могут быть несовместимо выровнены, когда два сопряженных рида картируются на совершенно разные местоположения референса и/или с аномальной относительной ориентацией парных ридов. Такие препятствия можно преодолеть с помощью способов, описанных в настоящем документе.

[00823] Например, в определенных случаях данные, относящиеся к известным структурных вариантам, могут быть использованы для более хорошего определения последовательности структурного варианта. Например, можно составить базу данных, имеющую список структурных вариаций в геноме человека, например, с упором на CNV, и такие данные могут быть использованы при определении последовательности конкретных вариантов, например, в соответствующим образом сконфигурированном протоколе взвешивания. В частности, когда структурный вариант известен, его «внутренние» и «наружные» координаты можно использовать как минимальный и максимальный диапазон последовательности, на который может влиять структурная вариация. Кроме того, известные вариации в виде инсерции, потери, приобретения, инверсии, ПГЗ, выворачивания, транслокации и ОРД можно классифицировать и ввести в базу знаний представленной системы.

[00824] В различных случаях определение структурного варианта можно выполнить с помощью соответствующим образом сконфигурированного программного обеспечения, исполняемого на ЦПУ/ГПУ/КПУ, например, путем использования ранее определенных данных секвенирования, а в других случаях можно выполнить анализ структурных вариантов, например, в аппаратном обеспечении, описанном в настоящем документе. Соответственно, в конкретных случаях предусмотрен способ анализа геномных последовательностей для структурных вариантов. Например, на первом этапе можно получить риды геномной последовательности из образца нуклеотидов. В определенных случаях секвенированные риды могут быть получены из протоколов спаренных концов или сопряженных парных ридов для обнаружения структурных вариантов. Далее можно построить индекс геномной референсной последовательности, например, когда индекс может быть хэш-таблицей или деревом, таким как дерево префиксов или суффиксов. После того, как индекс построен, риды геномной последовательности образца можно сравнить с индексом, чтобы определить, картируются ли риды геномной последовательности на индекс. Если да, риды геномной последовательности образца можно затем выровнять на геномную референсную последовательность, на которую они картированы, и можно определить оценку выравнивания.

[00825] Как указано выше, картирование и/или выравнивание можно осуществит с помощью аппаратного модуля, как описано в настоящем документе. В некоторых вариантах реализации картированные и выровненные данные можно затем передать в связанные ЦПУ/ГПУ/КПУ для дальнейшего анализа и обработки. Эти риды можно отсортировать по картированным позициям референса, а дубликаты ридов можно маркировать и удалить. Можно определить выравнивания химерных ридов и/или необычные относительные выравнивания двух сопряженных ридов и на основе обнаруженных выравниваний химерных ридов и/или необычных относительных выравниваний можно определить возможные структурные варианты (например, большой индел, инверсию или транслокацию). Аналогичным образом можно вычислить

апостериорные вероятности каждого возможного структурного варианта. В некоторых вариантах реализации можно определить гаплотипы структурных вариантов, например, с помощью анализа HMM на выравниваниях химерных ридов и/или необычных относительных выравниваниях. Например, для такого определения можно использовать парную HMM. Парную HMM можно осуществить с помощью аппаратного модуля.

[00826] Соответственно, в различных случаях, как показано на ФИГ. 43D, представлен способ определения вариантов в структуре хромосом организма. Например, в соответствии со способами, описанными в настоящем документе, на этапе 1000 можно принять риды геномной последовательности. На этапе 1010 можно построить одну или более референсных последовательностей, чтобы выполнить сравнение между ридами и референсными последовательностями. А именно, на этапе 1010 можно построить геномную референсную последовательность, что обеспечить возможность сравнения принятых ридов со сформированным референсом. Точнее говоря, для этих целей на этапе 1020 можно сформировать индекс для геномной референсной последовательности, например, на этапе 1020 можно сформировать хэш-таблицу или дерево префиксов/суффиксов. В результате на этапе 1030 можно сравнить риды геномной последовательности образца со сформированным индексом, например, в соответствии с программными и/или аппаратными реализациями, описанными в настоящем документе.

[00827] Если на этапе 1040 определено, что риды геномной последовательности образца картируются на индекс, то на шаге 1050 картированные риды можно выровнять с геномной референсной последовательностью и сформировать оценку выравнивания. На этапе 1080 риды образца можно отсортировать по их картированным позициям референса. В это время на этапе 1081 можно маркировать и удалить дубликаты ридов. Далее, на этапе 1090 можно обнаружить выравнивания химерных ридов и/или необычные относительные выравнивания, например, могут быть обнаружены два сопряженных рида, и на этапе 1092 можно определить возможные структурные варианты, например, на основе обнаруженных выравниваниях химерных ридов и/или необычных относительных выравниваниях. К тому же можно вычислить апостериорные вероятности каждого возможного структурного варианта и, необязательно, на этапе 1096 можно определить гаплотипы структурных вариантов, например, с помощью анализа HMM, как описано в настоящем документе, выравниваний химерных ридов и/или необычных относительных выравниваний.

[00828] Кроме того, устройства, системы и способы, описанные в настоящем документе, могут быть использованы для обработки последовательностей РНК. В частности, в настоящем документе представлены способы анализа последовательности РНК, например, с использованием протокола сплайсированного картирования и выравнивания (например, с помощью соответствующим образом сконфигурированного сопоставителя/выравнивателя РНК). Например, в одном варианте реализации может быть предусмотрен конвейер транскриптома, например для сверхбыстрого анализа данных последовательности РНК, В частности, этот конвейер может быть выполнен с возможностью осуществления вторичного анализа РНК-транскриптов, например, применительно к выравниванию только на референс, а также выравнивают с помощью аннотаций).

[00829] Соответственно, в первом способе с помощью прибора для секвенирования можно создать необработанные данные рида, например в формате файла BCL и/или FASTQ, и ввести в систему, где можно выполнить картирование, выравнивание и определение вариантов. Однако, в различных случаях в систему можно ввести один или более файлов аннотаций генов (GTF), например, чтобы направить сплайсированные

выравнивания, например, можно построить и использовать LUT границы сплайсинга. Например, можно использовать таблицы точности выравнивания и границы сплайсинга. Соответственно, можно выполнить 2-фазовое выравнивание, например, когда на первой фазе выравнивания можно использовать новые границы сплайсинга, которые затем могут быть затем использованы для направления выполнения второй фазы картирования/выравнивания. После определения вариантов система выведет стандартный файл VCF, готовый для третичного анализа.

[00830] В частности, после того, как входной файл принят, можно выполнить сплайсированное картирование и выравнивание, например, на ридах с одинарными и спаренными концами. Как было указано, для вывода одной границы можно использовать фильтры границ, выполненные с возможностью конфигурирования. Можно выполнить сортировку по позиции, которая может включать в себя распределения по группам в соответствии с референсом, и затем сортировку групп по позиции референса, и можно маркировать дубликаты, например, на основе начальной позиции и строки CIGAR, для формирования высококачественного отчета о дубликатах, с помощью которого можно удалить все дубликаты. Затем можно выполнить определение вариантов гаплотипов, например, с помощью движка обработки SW и HMM.

[00831] Кроме того, устройства, системы и способы, описанные в настоящем документе, могут быть использованы для выполнения определения соматических вариантов. Например, можно использовать протокол определения соматических вариантов, чтобы обнаружить варианты, имеющие место в раковых клетках. В частности, можно получить геномные образцы для определения соматических вариантов из или биопсий одной или множества опухолей. Необязательно, можно также получить «нормальный» (неопухолевый) образец, например, для сравнения во время определения вариантов, например, когда соматические варианты будут возникать в опухолевых клетках, но не в клетках нормального образца. Из образцов можно изолировать ДНК/РНК и секвенировать, например, с помощью секвенатора нового поколения. Секвенированные данные, например, из каждого образца, можно затем передать на платформу вторичной обработки, и риды можно картировать и выровнять. Далее, риды можно подвергнуть множеству процедур определения вариантов, включая обработку с использованием одного или обоих движков SW и парной HMM.

[00832] Однако система должна быть выполнена с возможностью обнаружения вариантов с низкой частотой аллеля, такой как от 3% до 10% (или выше). Более конкретно, можно использовать модель вероятности генотипирования, которая выполнена с возможностью обеспечения произвольных частот аллелей. Один из способов обеспечения такой возможности состоит в назначении частот аллелей каждого генотипа варианта, соответствующих наблюдаемым частотам аллей в перекрывающихся ридах. Например, если 10% перекрывающихся ридов проявляют определенный вариант, можно проверить генотип, состояний на 90% из референсного аллеля и на 10% из другого аллеля. В случае двойных образцов опухоль/нормальная ткань можно оценить апостериорную вероятность того, что вариант присутствует в образце опухоли, но отсутствует в нормальном образце.

[00833] Например, конвейер определителя соматических вариантов может быть выполнен с возможностью обеспечения информации о гетерогенности опухоли, например, что произошла серия различных событий мутации, например, когда выявлены одна или более секций опухоли разных генотипов (подклон). Такую информацию о подклоне можно получить из определения частот аллелей вариантов и их распределений,

и/или исключительно с помощью дифференциального определения вариантов среди множества образцов опухоли.

[00834] Соответственно, предложены способы обнаружения вариантов последовательности раковых клеток из образца. На первом этапе из раковых и/или нормальных клеток можно получить риды геномной последовательности из образца нуклеотидов. Риды последовательности могут быть из протоколов спаренных концов или сопряженных парных ридов, аналогичных используемым для обнаружения структурных вариантов. Можно построить индекс геномной референсной последовательности, например, когда индекс может быть хэш-таблицей или деревом, таким как дерево префиксов или суффиксов. Риды геномной последовательности образца, например, образца опухоли и/или нормального образца, можно сравнить с индексом и определить, картируются ли риды эпигеномной последовательности на индекс.

[00835] Риды геномной последовательности образца можно затем выровнять на геномную референсную последовательность, на которую они картированы, и можно сформировать оценку выравнивания. Картирование и/или выравнивание можно осуществит с помощью программных и/или аппаратных модулей, как описано в настоящем документе. В некоторых вариантах реализации картированные и выровненные данные можно затем передать в ЦПУ/ГПУ/КПУ для дальнейшего анализа и обработки. Эти риды можно отсортировать по картированным позициям референса, а любые дубликаты ридов можно маркировать и удалить. Варианты можно обнаружить с помощью байесовского анализа, который модифицирован для ожидания произвольных частот аллелей вариантов и для обнаружения и сообщения возможных низких частот аллелей (например, от 3% до 10%).

[00836] В некоторых вариантах реализации генеративные варианты могут быть обнаружены как в нераковых, так и в раковых образцах, в соматические варианты могут быть обнаружены только в раковых образцах. Например, генеративные и соматические мутации могут отличаться относительной частотой. Для каждого возможного ракового варианта можно вычислить апостериорные вероятности, и в некоторых вариантах реализации можно определить гаплотипы структурных вариантов с помощью анализа HMM выравниваний химерических ридов и/или необычных относительных ридов. Например, для такого определения можно использовать парную HMM. Парную HMM можно осуществить с помощью аппаратных модулей, как описано в настоящем документе.

[00837] Соответственно, в различных вариантах реализации можно выполнять процедуру определения соматических вариантов, которая пояснена на ФИГ, 43E, например, для вычисления вероятности того, что вариант является раковым вариантом. Например, на этапе 1000 можно сформировать риды геномной последовательности образцов, например, путем секвенирования с помощью СНП, и/или получить их, например, посредством передачи с помощью соответствующим образом сконфигурированной облачной сетевой системы, например, из одного или обоих ракового и нераковых генетических образцов. На этапе 1010 можно сформировать геномную референсную последовательность, например, для сравнения ридов, на этапе 1020 можно построить индекс геномной референсной последовательности, а на этапе 1030 геномную последовательность образца можно сравнить с индексом, например, с помощью программных и/или аппаратных реализаций, описанных в настоящем документе, чтобы на этапе 1040 картировать риды геномной последовательности на индекс. Далее, на этапе 1050 картированные риды можно выровнять с геномной

референсной последовательностью, чтобы сформировать оценку выравнивания. На этапе 1080 картированные и/или выровненные риды можно затем отсортировать по позиции референса, и, необязательно, на этапе 1081 можно маркировать и удалить любые дубликаты ридов.

[00838] Кроме того, после того, как риды маркированы, и/или выровнены, и/или отсортированы, и/или избавлены от дубликатов, на этапе 1100 можно обнаружить варианты, например, с помощью байесовского анализа, а на этапе 1101 можно обнаружить генеративные варианты, как в нераковых, так и в раковых образцах, а также, необязательно, соматические варианта в этих образцах. Аналогичным образом на этапе 1094 можно вычислить апостериорные вероятности каждого возможного ракового варианта. Далее, на этапе 1096 можно, необязательно, определить гаплотипы раковых вариантов, например, путем реализации анализа HMM в программном и/или аппаратном обеспечении, как описано в настоящем документе.

[00839] Кроме того, устройства, системы и способы, описанные в настоящем документе, могут быть выполнены с возможностью осуществления операции совместного генотипирования. В частности, операцию совместного генотипирования можно использовать для улучшения точности определения вариантов, например, за счет совместного рассмотрения ридов из когорты множества субъектов. Например, в различных случаях геномные вариации могут быть сильно коррелированными в определенных популяциях, например, когда определенные варианты распространены среди множества субъектов. В таких случаях чувствительность и специфичность определения вариантов можно улучшить за счет совместного рассмотрения подтверждающих данных для каждого варианта из множества образцов ДНК (или РНК). А именно, чувствительность можно улучшить ввиду того, что слабые подтверждающие данные для варианта у одного субъекта могут быть усилены подтверждающими данными для того же самого варианта в других образцах. Точнее говоря, чувствительность можно улучшить потому, что умеренные подтверждающие данные для ложноположительного варианта могут быть ослаблены отсутствием подтверждающих данных для того же самого варианта в других образцах. Вообще говоря, чем больше образцов участвуют в совместном генотипировании, тем более точным будет определение вариантов для любого данного субъекта.

[00840] Совместное генотипирование включает в себя оценку апостериорных вероятностей для различных подмножеств всех субъектов, имеющих данный вариант с помощью априорных вероятностей, которые выражают наблюдаемые корреляции в генетической вариации. В различных случаях совместное генотипирование можно осуществить за одно выполнение определения вариантов, где выровненные риды из множества образцов исследуются определителем вариантов. Обычно это практично только для небольшого количества образцов, так как при участии десятков, сотен или тысяч образцов общий размер данных становится практически нереальным для быстрого доступа и манипулирования.

[00841] В альтернативном варианте реализации совместное генотипирование можно осуществить, сначала выполнив определение вариантов отдельно для каждого образца, объединив затем результаты с помощью средства совестного генотипирования, которое обновляет вероятности вариантов для каждого субъекта с использованием объединенной информации. В этом способе используют дополнительные выходные данные из каждого определения вариантов одного образца, чтобы лучше изменить области слабых подтверждающих данных для вариантов и/или в областях, где без совместной обработки варианты не были бы определены. Хотя для представления определенных вариантов

при определении вариантов одного образца обычно используют формат VCF, для представления определений вариантов первой стадии (и не вариантов) при подготовке к объединению можно использовать специальный формат gVCF. Формат gVCF содержит записи для местоположений и/или блоков из множества местоположений, где скорее всего нет варианта, поэтому данную информацию можно объединят с gVCF других определений или отсутствия определений в тех же местоположениях, чтобы получать улучшенные совместные определения генотипов для каждого субъекта.

[00842] Соответственно, конвейер совместного генотипирования может быть выполнен с возможностью определения вариантов для множества проб быстрее и с большей точностью. Кроме того, конвейер совместного генотипирования может быть также выполнен с возможностью поддержки определения вариантов генеалогии, а также популяции из когорты образцов. Например, конвейер можно выполнить с возможностью обработки до 10, 15, 20, 25, даже 50 или более образцов одновременно. В различных случаях конфигурация для определения вариантов популяции может быть выполнена с возможностью одновременной обработки образцов размером в многие тысячи. Кроме того, сочетание скорости и иерархического группирования множества образцов обеспечивает эффективное с вычислительной точки зрения решение для анализа совместного генотипирования. Кроме того, секвенирование образцов для совместного генотипирования можно выполнять в одной и той же проточной кювете секвенатора нового поколения, что позволяет системе одновременно картировать/выравнивать входные данные множества образцов, тем самым ускоряя общий процесс совместного определения, например, когда данные BCL можно подавать прямо в конвейер для создания уникальных файлов gVCF для каждого образца.

[00843] Поэтому в настоящем документе предложен способ улучшения точности определения вариантов за счет совместного рассмотрения ридов когорты из множества субъектов. На первом этапе принимают риды геномной последовательности из двух или более образцов. Строят геномную референсную последовательность для сравнения с ридами, а из нее формируют индекс геномной референсной последовательности. Затем риды геномной последовательности каждого образца сравнивают с индексом и определяют, картируются ли риды геномной последовательности каждого образца на индекс.

[00844] Затем картированные риды можно выровнять с геномной референсной последовательностью, и можно сформировать оценку выравнивания. Эти риды можно отсортировать по картированным позициям референса, а дубликаты ридов можно маркировать и/или удалить. Кроме того, после этого можно проанализировать перекрывающиеся риды из скопления ридов, чтобы определить, согласуются ли большинство ридов с референсной геномной последовательностью. Для каждого возможного варианта вычисляют апостериорные вероятности, и можно объединить данные определения вариантов из всех образов, чтобы улучшить точность определения вариантов для каждого отдельного образца. Это улучшает точность определения вариантов (например, чувствительность и специфичность) для каждого образца и может осуществить в виде этапа обработки после того, как образцы были подвергнуты анализу определения вариантов, или это можно делать накопительно после определения вариантов для каждого образца. Затем можно определить правдоподобие нереференсных аллелей в областях, где не определено вариантов, и сообщить полученное правдоподобие нереференсных аллелей в областях, где не определено вариантов.

[00845] Соответственно, в различных вариантах реализации можно выполнять процедуру определения соматических вариантов, которая пояснена на ФИГ, 43F,

например, для вычисления вероятности того, что вариант является раковым вариантом. Например, на этапе 1000 можно сформировать риды геномной последовательности образцов, например, путем секвенирования с помощью СНП, и/или получить их, например, посредством передачи с помощью соответствующим образом сконфигурированной облачной сетевой системы, например, из одного или обоих ракового и неракового генетических образцов. На этапе 1010 можно сформировать геномную референсную последовательность, например, для сравнения ридов, на этапе 1020 можно построить индекс геномной референсной последовательности, а на этапе 1030 геномную последовательность образца можно сравнить с индексом, например, с помощью программных и/или аппаратных реализаций, описанных в настоящем документе, чтобы на этапе 1040 картировать риды геномной последовательности на индекс. Далее, на этапе 1050 картированные риды можно выровнять с геномной референсной последовательностью, чтобы сформировать оценку выравнивания. На этапе 1080 картированные и/или выровненные риды можно затем отсортировать по позиции референса, и, необязательно, на этапе 1081 можно маркировать и удалить любые дубликаты ридов.

[00846] Аналогичным образом на этапе 1082 можно проанализировать перекрывающиеся риды из скопления ридов, чтобы определить, согласуются одно или более, например, большинство, ридов с референсными геномными последовательностями, а на этапе 1094 можно вычислить апостериорные вероятности каждого возможного варианта. В этом момент на этапе 1096 можно определить гаплотипы вариантов, если требуется, например, путем выполнения анализа HMM, и/или на этапе 1120 можно, необязательно, объединить данные определения вариантов, например, из всех, образцов, чтобы улучшить точность определения вариантов для каждого отдельного образца. Далее, на этапе 1122 можно определить и сообщить правдоподобие нереференсных аллелей, например, в областях, где не определено вариантов.

[00847] Кроме того, как показано на ФИГ, 43, в соответствии с одним аспектом предложен магазин приложений в сети, чтобы пользователи могли разрабатывать, продавать и использовать геномные средства, которые могут быть внедрены в систему и использоваться для анализа геномных данных, передаваемых и вводимых в систему. В частности, магазин геномных приложений позволяет клиентам, у которых есть желание, разрабатывать генетические тесты, например, вроде теста ОРИТН, которые после разработки могут быть выгружены в систему, например, генетическую торговую площадку, для приобретения и использования в качестве платформы системы, чтобы все, кто используют вновь разработанную платформу системы, могли использовать выгруженные тесты через веб-портал. Более конкретно, пользователь может перейти на веб-портал магазина приложений, найти требуемый тест, например, тест ОРИТН, загрузить его и/или сконфигурировать систему для его реализации, например, на своих пригодных для выгрузки генетических данных. Поэтому «когортная» торговая интернет-площадка представляет быстрый и эффективный способ развертывания новых генетических аналитических приложений, которые позволяют получать идентичные результаты с любой из представленных платформ, которые исполняют загруженное приложение. Более конкретно, рыночная интернет-площадка обеспечивает для любого, кто работает с системой, механизм разработки приложений генетического анализа, которые удаленный пользователь может загрузить и сконфигурировать для использования в соответствии с представленными моделями рабочего потока.

[00848] В соответствии с другим аспектом, когортная торговая площадка, описанная

в настоящем документе, позволяет безопасно делиться данными. Например, передача и хранение геномных данных должны быть хорошо защищенными. Однако зачастую такие генетические данные большого объема и сложны для передачи надежным защищенным образом, например, когда идентификационные данные субъекта ограничена. Соответственно, представленная генетическая торговая площадка позволяет участникам когорты совместно использовать генетические данные без необходимости идентификации субъекта. На такой торговой площадке участники когорты могут делиться вопросами и процессами для продвижения своих исследований в защищенной и безопасной среде, не подвергая риску идентификационные данные геномов своих соответствующих субъектов. Кроме того, пользователь может заручиться помощью остальных исследователей в анализе своих наборов образцов, не раскрывая личности тех, кому эти геномы принадлежат.

[00849] Например, пользователь может идентифицировать субъектов, обладающих определенным генотипом и/или фенотипом, таким как рак молочной железы 3-й стадии, и/или прохождение терапии конкретным лекарственным средством. Можно сформировать когорту, чтобы посмотреть, как эти лекарственные препараты влияют на рост раковых клеток на генетическом уровне. Таким образом, эти характеристики, среди прочих, могут формировать критерии отбора в когорту, которые позволят другим исследователям, например, находящихся на удалении, выполнять стандартные генетические анализы на генетических данных, используя единообразные аналитические процедуры, на тех доступных для них субъектах, которые удовлетворяют критериям когорты. Таким образом, данному исследователю не нужно отвечать за идентификацию и безопасность тех, чьи образцы входят в набор образцов, например, субъектов, удовлетворяющих критериям, при аргументировании своего научного анализа.

[00850] В частности, исследователь А может создать исследовательскую когорту на торговой площадке и определить надлежащие критерии выбора для субъектов, геномных тестов, которые будут выполняться, и параметры для выполнения тестов. Исследователи В и С, находящиеся на удалении от исследователя А, могут затем подписаться на когорту, определить и выбрать субъектов, отвечающих критериям, и затем выполнить указанные тесты на своих субъектах, используя единообразные процедуры, описанные в настоящем руководстве, чтобы помочь исследователю А оперативно достичь своих исследовательских целей, или сделать это более эффективно. Это выгодно, так как передается только часть генетических данных, идентификационные данные субъекта защищены, и, поскольку данные анализируются при помощи одной и той же системы генетического анализа с использованием одинаковых параметров, данные результатов будут одними и теми же вне зависимости от того, где и на какой машине выполнялись тесты. Следовательно, когортная торговая площадка позволяет пользователям формировать и создавать когорты, просто публикуя критерии выбора и параметры выполнения на информационной панели. Также можно публиковать ставки вознаграждения и исполненные платежи, используя соответствующим образом сконфигурированную программу для коммерческой деятельности, например для денежного обмена.

[00851] Любой, кто решает принять участие в когорте, может загрузить критерии и файлы данных и/или использовать генетические данные, уже сформированные и/или сохраненные им, при выполнении запрошенного анализа. Например, каждый участник когорты будет должен, или иметь возможность, формировать базу данных файлов BCL и/или FASTQ, которые хранятся на его индивидуальных серверах. Эти генетические файлы будут получены для субъектов, которые окажутся соответствующими критериям

отбора. А именно, эти сохраненные генетические и/или другие данные субъекта могут быть отсканированы, чтобы определить пригодность для включения в рамках критериев выбора в когорту. Такие данные могли быть уже сформированы в нескольких целях, но вне зависимости от причин их формирования, они могут после этого быть выбраны и подвергнуты запрошенным конвейерным анализам и использованы для включения в когорту.

[00852] Соответственно, в различных вариантах реализации когортная система может быть форумом для соединения исследователей, чтобы позволить им объединять свои ресурсы и данные, например данные генетической последовательности. Например, вступление в когорту позволит первому исследователю ввести проект, запрашивающий анализы генетических данных, требующие интеллектуального анализа и/или исследования ряда геномов различных субъектов, например, касающихся картирования, выравнивания, определения вариантов и т.п. Поэтому вместо того, чтобы собирать субъектов и отбирать наборы образцов самому, инициатор когорты может объявить о потребности в выполнении заданной процедуры анализов на наборах образцов, собранных другими, или которые будут собраны другими, и такой коллективный подход формированию наборов образцов и анализа их обеспечивается организацией когорты, описанной в настоящем документе. В частности, инициатор когорты может установить выбор в когорту, создать файл конфигурации для совместного использования потенциальными участниками когорты, создать параметры рабочего потока, например, в папке рабочего потока, и может тем самым автоматизировать формирование и анализ данных, например посредством системы управления рабочими потоками. Система может также обеспечить коммерческую сторону транзакции, например, обработку платежа для компенсации участникам когорты предоставления ими наборов генетических данных, которые могут быть проанализированы, например, в отношении картирования, выравнивания, определения вариантов и/или с точки зрения третичного анализа.

[00853] В различных вариантах реализации когортный структурированный анализ может быть направлен на первичную обработку, например, либо ДНК, либо РНК, например, такую как обработка изображения и/или перекалибровка оценки качества оснований, анализ метилирования и т.п.; и/или может быть направлен на выполнение вторичного анализа, например, в отношении картирования, выравнивания, сортировки, определения вариантов и т.п.; и/или может быть направлен на третичный анализ, например, в отношении матричного, геномного, эпигеномного, метагеномного анализа, анализа генотипирования, вариантов и/или других форм третичного анализа. Кроме того, необходимо понимать, что хотя многие из конвейеров и анализов, выполняемых тем самым, могут включать в себя первичную и/или вторичную обработку, различные платформы анализа, описанные в настоящем документе, могут не относиться непосредственно к первичной или вторичной обработке. Например, в определенных случаях платформа анализа может относиться исключительно к выполнению третичного анализа, например, на генетических данных, или других форм геномных и/или биоинформационных анализов.

[00854] Например, в конкретных вариантах реализации, что касается конкретных аналитических процедур, которые будут выполняться, подлежащие выполнению анализы могут включать в себя одно или более из картирования, выравнивания, сортировки, определения вариантов и т.п., для создания данных результатов, которые могут быть подвергнуты оной или более процедур вторичного и/или третичного анализа в зависимости от определенных конвейеров, выбранных для выполнения. Рабочий поток

может быть простым или он может быть сложным, например, он может требовать использования одного модуля конвейера, например картирования, или множества модулей, таких как картирование, выравнивание, сортировка, определение вариантов и/или другие, но важным параметром является то, что рабочий поток должен быть идентичным для каждого участника когорты. В частности, уникальным признаком системы является то, что инициатор запроса, создающий когорту, указывает параметры управления, чтобы обеспечить выполнение анализа одинаковым образом вне зависимости от того, где эти процедуры выполняются, и на каких машинах.

[00855] Соответственно, при настройке когорты инициатор запроса выгрузит критерии выбора вместе с файлом конфигурации. Затем другие участники когорты просмотрят критерии выбора, чтобы определить, имеются ли у них наборы данных генетической информации, попадающие в пределы установленных критериев отбора, и если да, выполнят запрошенный анализ на этих данных на основе настроек файла конфигурации. Исследователи могут подписаться на выбор в качестве участника когорты, и в случае большого числа подписчиков, может быть организован розыгрыш или выбор участников на конкурентной основе. В различных случаях может быть инициирована система открытых торгов. Данные результатов, сформированные участниками когорты, могут быть обработаны в месте эксплуатации или на облаке, и если при этом используют файл конфигурации, обработка данных будет одинаковой. В частности, файл конфигурации указывает, как нужно сконфигурировать устройство BioIT-аналитики, и после того, как устройство настроено в соответствии с заданной конфигурацией и связано с системой, оно будет выполнять запрошенный генетический анализ одинаковым образом вне зависимости от того, где оно расположено, локально или удаленно. Затем данные результатов могут быть выгружены на когортную торговую площадку, и с учетом полученных данных результатов переводится и принимается оплата.

[00856] Например, анализ генетических данных может быть выполнен локально, а результаты выгружены на облако, или выгружены могут быть сами генетические данные, а анализ может быть выполнен на облаке, например, на сервере или в сети серверов, например, на квантовой платформе обработки, связанной с облаком. В различных случаях, возможно, будет полезно выгружать только данные результатов, чтобы лучше защищать идентификационные данные субъектов. В частности, выгрузка только данных результатов не только надежно защищена, то и избавляет от необходимости передачи больших количеств данных, тем самым повышая эффективность системы.

[00857] Более конкретно, в различных случаях может быть выгружен сжатый файл, содержащий данные результатов из одного или более конвейеров, и в некоторых случаях нужно выгружать только файл, содержащий описание вариаций. В некоторых случаях нужно лишь дать ответ, такой как, например, «да» или «нет». Такие ответы предпочтительнее, так как они не указывают идентификационные данные субъекта. Однако, если анализ нужно выполнять в сети, например, в облаке, выбранные файлы BCL и/или FASTQ могут быть выгружены, анализ выполнен, и затем данные результатов могут быть возвращены обратно исходному отправителю, который затем может выгрузить данные результатов через интерфейс когорты. После чего исходные необработанные данные могут быть удалены из памяти в сети. Благодаря этому и прочим подобным вещам инициатор запроса когорты не будет иметь доступа к идентифицирующим данным субъектов.

[00858] Для повышения эффективности когорты особенно полезно сжатие, например,

с использованием технологии анализа «точно в срок» (JIT). Например, перемещение данных в когортную систему и обратно с использованием типичных процедур обходится очень дорого. Соответственно, хотя в различных конфигурациях необработанные и/или несжатые данные, выгружаемые в систему, могут храниться там, в конкретных случаях данные могут быть сжаты перед выгрузкой, и затем эти данные могут быть обработаны в системе, а результаты после этого могут быть сжаты перед передачей из системы, например, когда сжатие совершается в соответствии с протоколом JIT. В этом случае хранение таких данных, например, в сжатой форме, обходится дешевле, и, следовательно, когортная система весьма экономически эффективна.

[00859] Кроме того, в различных случаях на одной торговой интернет-площадке могут быть представлены множество когорт, и при наличии процессов сжатия, описанных в настоящем документе, данные могут быть переданы из одной когорты в другую, чтобы исследователи всевозможных разных когорт могли совместно использовать данные между собой, что без применения способов сжатия, описанных в настоящем документе, могло бы оказаться непозволительно дорогим. В частности, без скорости и эффективности сжатия JIT данные, однажды переданные на облако, будут, как правило, оставаться на облаке, хотя и будут доступны там для просмотра и манипулирования. Однако JIT позволяет быстро передавать данные на облако и обратно для локальной и/или облачной обработки. Кроме того, как показано на ФИГ. 41B и 43, в конкретных случаях система 1 может быть выполнена с возможностью применения к сформированным и/или подвергнутым вторичной обработке данным дальнейшей обработки, например, посредством локального 100 и/или удаленного 300 вычислительного ресурса, например, путем пропускания их через один или более конвейеров третичной обработки, таких как один или более из конвейера микроматричного анализа, конвейера анализа генома, например, полногеномного анализа, конвейера анализа генотипирования, конвейера анализа экзома, конвейера анализа микробиома, конвейера анализа генотипирования, включая совместное генотипирование, конвейера анализа вариантов, включая конвейеры структурных вариантов, конвейеры соматических вариантов, и конвейеры GATK и/или MuTect2, а также конвейеры секвенирования РНК и/или другой конвейер третичной обработки. Данные результатов такой обработки могут быть затем сжаты и/или сохранены удаленно 400 и/или переданы для сохранения локально 200.

[00860] В частности, одна или более, например, все, из этих функций могут быть выполнены локально, например, в месте 10 эксплуатации, на локальном облаке 30 или посредством контролируемого доступа при помощи гибридного облака 50. В таком случае создают среду разработчика, которая позволяет пользователю управлять функциональными возможностями системы 1 для удовлетворения своих индивидуальных потребностей и/или для предоставления доступа к ней другим пользователям, которые ищут такие же или подобные результаты. Следовательно, различные компоненты, процессы, процедуры, средства, ярусы и иерархии системы могут быть выполнены с возможностью конфигурирования через интерфейс ГПИ, который позволяет пользователю выбирать, какие компоненты системы использовать, на каких данных, в какое время и в каком порядке в соответствии с установленными требованиями пользователя и протоколами, чтобы сформировать соответствующие данные и соединения между данными, которые могут быть безопасно переданы по всей системе, будь то локально или удаленно. Как было указано, эти компоненты могут быть выполнены с возможностью беспрепятственного обмена данными между собой, например, независимо от местоположения и/или вида соединения, например, за счет

конфигурации с жестким связыванием и/или бесшовного связывания посредством облака, и/или возможности конфигурирования, например, посредством протокола JIT, чтобы выполнять одни и те же или подобные процессы одинаковым или аналогичным образом, например, путем использования соответствующих интерфейсов API, рассредоточенных по всей системе, применение которых позволяет различным пользователям конфигурировать различные компоненты для выполнения различных процедур аналогичным образом.

[00861] Например, API может быть определен в заголовочном файле в отношении процессов, которые должны выполняться каждым конкретным компонентом системы 1, причем заголовок описывает выполняемые функции и определяет, как вызывать функции, например, параметры, которые передаются, принимаемые входные и передаваемые выходные данные, и способ, каким это происходит, что поступает и как, что выдается и как, что возвращается и каким образом. Например, в различных вариантах реализации один или более компонентов и/или их элементов, которые могут образовывать один или более конвейеров одного или более ярусов системы, могут быть выполнены с возможностью конфигурирования их с помощью инструкций, вводимых пользователем и/или одним или более приложений второй и/или третьей стороны. Эти инструкции могут передаваться в систему через соответствующие API, которые обмениваются данными с одним или более различных драйверов системы, указывая драйверам, какие части системы, например, модули, и/или какие процессы в них нужно активировать, когда и в каком порядке, с учетом предварительной выбранной конфигурации параметров, которая может быть определена с помощью интерфейса, который может быть выбран пользователем, например, ГПИ,

[00862] В частности, один или более драйверов DMA системы 1 могут быть выполнены с возможностью работы соответствующим образом, например, на уровне ядра каждого компонента и системы 1 в целом. В таком случае одно или более из предусмотренных ядер может иметь свой собственный базовый API очень низкого уровня, который предоставляет доступ к аппаратному обеспечению и функциям различных компонентов системы 1, чтобы иметь доступ к соответствующим регистрам и модулям для конфигурирования и руководства процессами и тем, как они выполняются в системе 1. А именно, сверху этого слоя может быть построен виртуального слой служебных функций для формирования строительных блоков, которые используются для множества функций, которые отправляют файлы вниз в ядра и получают обратно результаты, кодирую, шифруют и/или передают соответствующие данные и далее выполняют на них функции более высокого уровня. Поверх этого слоя может быть построен дополнительный слой, использующий эти служебные функции, которые могут быть уровня API, с которым может взаимодействовать пользователь, причем этот слой может быть выполнен с возможностью функционирования в основном для конфигурирования системы 1 в целом или ее составляющих частей, загружая файлы и выгружая результаты, при этом файлы и/или результаты могут быть переданы по всей системе, либо локально, либо глобально. Могут быть сконфигурированы и включены дополнительные API, более подробно описанные выше в связи с безопасным хранением данных.

[00863] Такое конфигурирование различных API, памятей и/или прошивки системы может включать в себя обмен данными с регистрами, а также выполнение вызовов функций. Например, как описано выше в настоящем документе, один или более вызовов функций, необходимых и/или полезных для выполнения этапов, например, последовательно, с целью осуществления картирования, и/или выравнивания, и/или сортировки, и/или определения вариантов или других функций вторичной и/или

третичной обработки, которые описаны в настоящем документе, могут быть реализованы в соответствии с операциями аппаратного обеспечения и/или связанных алгоритмов для формирования необходимых процессов и выполнения требуемых этапов.

[00864] А именно, ввиду того, что в определенных вариантах реализации одна или более из этих операций могут быть основаны на одной или более структурах, возможно, потребуется построить различные структуры, необходимые для реализации этих операций. В этой связи потребуется вызов функции, которая выполнит данные действия, причем вызов функции приведет к построению нужной структуры для выполнения операции, и поэтому данный вызов примет имя файла, где хранятся файлы параметров структуры, и затем сформирует один или более файлов данных, которые содержат и/или конфигурируют нужную структуру. Другой вызов функции может быть предназначен для загрузки структуры, которая была сформирована посредством соответствующего алгоритма, и передачи ее вплоть до памяти на микросхеме и/или в систему 1 и/или помещения ее в нужное место, где ее нахождение предполагается аппаратным обеспечением. Конечно, для выполнения различных других выбранных функций системы 1 потребуется загружать различные данные в микросхему и/или иным образом передавать в системный генератор, и эти функции может выполнять диспетчер конфигураций, например, путем загрузки всего, что необходимо для того, чтобы модули конвейеров в ярусах платформ микросхемы и/или системы в целом выполняли свои функции, в память, прикрепленную или иным образом связанную с микросхемой и/или системой.

[00865] Кроме того, система может быть выполнена таким образом, чтобы обеспечивать различным компонентам системы возможность обмена данными друг с другом, например, чтобы позволять одной или более микросхемам системы 1 взаимодействовать с печатной платой секвенатора 121, вычислительного ресурса 100/300, преобразователя 151, анализатора 152, интерпретатора 310, коллаборатора 320 или других компонентов системы, когда они включены в нее, чтобы принимать FASTQ и/или другие файлы сформированной и/или обработанной генетической последовательности непосредственно из секвенатора или других компонентов обработки, например, немедленно после их формирования и/или обработки, и затем передавать эту информацию в диспетчер конфигураций, который после этого направляет данную информацию в соответствующие банки памяти в аппаратном и/или программно обеспечении, которые делают эту информацию доступной соответствующим модулям аппаратного обеспечения, программного обеспечения и/или системе в целом, чтобы они могли выполнять свои назначенные функции на этой информации для определения оснований, картирования, выравнивания, сортировки и т.д. образца ДНК/РНК относительно референсного генома и/или выполнения на ней связанных операций вторичной и/или третичной обработки.

[00866] Соответственно, в различных вариантах реализации может быть включен интерфейс уровня клиентов (CLI), которые может обеспечивать пользователям возможность непосредственного вызова одной или более из этих функций. В различных вариантах реализации CLI может быть программным приложением, например, имеющим ГПИ, которое выполнено с возможностью конфигурирования доступности и/или использования аппаратных и/или различных других программных приложений системы. Следовательно, CLI может быть программой, которая принимает инструкции, например, аргументы, и делает функциональные возможности доступными путем вызова прикладной программы. Как указано выше, CLI может быть основан на командной

строке или ГПИ (графическом пользовательском интерфейсе). Уровень командной строки ниже уровня ГПИ, причем ГПИ включает в себя диспетчер файлов на базе Windows с возможностью выбора щелчком мыши функциональных блоков, которые изображают, какие модули, какие конвейеры, какие ярусы каких платформ будут использованы, а также параметры для их использования. Например, во время работы, если предписано, CLI будет находить местоположение ссылки, будет определять, нужно ли формировать хэш-таблицу и/или индекс, или, если они уже сформированы, находить, где они хранятся, и руководить выгрузкой формируемой хэш-таблицы и/или индекса и т.д. Инструкции этих видов могут появляться в виде вариантов на ГПИ, которые пользователь может выбирать для выполнения связанными микросхемами/системой 1.

[00867] Кроме того, может быть включена библиотека, которая может содержать уже существующие редактируемые файлы конфигурации, такие как файлы, ориентированные на типичные выбираемые пользователем функциональные возможности аппаратного и/или связанного программного обеспечения, например, относящиеся к анализу части или всего генома и/или белков, например, для различных анализов, таких как анализ персональных медицинских историй и родословной, или диагностика болезней, или открытие новых лекарственных средств, терапевтика и/или одна или более других аналитик и т.д. Параметры этих типов могут предварительно установлены, например, для выполнения таких анализов, и могут быть сохранены в библиотеке. Например, если описанную в настоящем документе платформу используют, например, для исследований в области НИПТ, ОРИТН, рака, LDT, AgBio и связанных исследований на коллективном уровне, настоящие параметры могут быть сконфигурирован иначе, чем если бы платформа была направлена на проведение только геномного и/или генеалогического исследования, например на индивидуальном уровне.

[00868] Более конкретно, в случае определенной диагностики индивида точность может быть важным фактором. Поэтому, параметры системы могут быть установлены так, чтобы гарантировать точность, хотя и в обмен на возможное снижение скорости. Однако для других областей применения геномики скорость может быть основным определяющим фактором, и, следовательно, параметры системы могут быть установлены на максимальное повышение скорости, однако при этом придется пожертвовать точностью. Соответственно, в различных вариантах реализации часто используемые настройки параметров для выполнения разных задач могут быть предварительно установлены в библиотеку, чтобы облегчить их использование. Такие настройки параметров могут также включать в себя необходимые программные приложения и/или аппаратные конфигурации, используемые при эксплуатации системы 1. Например, библиотека может содержать код, который исполняет API и может также включать в себя файлы образов, сценарии и любую другую вспомогательную информацию, необходимую для работы системы 1. Следовательно, библиотека может быть выполнена с возможностью компиляции программного обеспечения для выполнения API, а также различных исполняемых объектов.

[00869] Кроме того, как показано на ФИГ. 42C и 43 система может быть выполнена таким образом, чтобы один или более системных компонентов могли быть выполнены удаленно, например, когда компонент системы выполнен с возможностью осуществления одной или более функций сравнения на данных, например, функции 310 интерпретации и/или функции 320 совместной работы. Например, когда к данным применяют протокол интерпретации, протокол 312 интерпретации может быть выполнен с возможностью анализа и делания выводов о данных и/или определения различных

взаимосвязей в них, могут также выполняться один или более других аналитических протоколов, которые включают в себя аннотирование 311 данных, выполнение 313 диагностики на данных и/или анализ данных с целью определения присутствия или отсутствия одного или боле биомаркеров 314. Как было указано одна или более из этих функций могут руководиться WMS и/или выполняться модулем ИИ, описанным в настоящем документе.

[00870] Кроме того, при использовании протокола совместной работы система 1 может быть выполнена с возможностью обеспечения электронного форума, где можно делиться 321 данными, причем протокол совместного использования данных может включать в себя настройки безопасности 324 и/или конфиденциальности 322, которые могут быть выбраны пользователем и позволяют шифровать данные и/или защищать их паролем, чтобы можно было скрыть идентификаторы и источники данных от пользователей системы 1. В конкретных случаях система 1 может быть выполнена с возможностью разрешения анализатору 3-й стороны 121 выполнять виртуальные моделирования на данных. Кроме того, после формирования данные, интерпретированные и/или подвергнутые одному или более совместно осуществляемым анализам, могут быть сохранены либо удаленно 400, либо локально 200, чтобы сделать их доступными для удаленного 300 или локального 100 вычислительных ресурсов.

[00871] В соответствии с другим аспектом, как показано на ФИГ. 44, предложен способ использования системы для формирования одного или более файлов данных, на которых можно выполнять один или более протоколов вторичной и/или третичной обработки. Например, способ может включать в себя обеспечение геномной инфраструктуры, например, для одного или более из локального, облачного и/или гибридного формирования, и/или обработки, и/или анализа в области геномики и/или биоинформатики.

[00872] В таком случае геномная инфраструктура может включать в себя биоинформационную платформу обработки, имеющую одну или более памятей, которые выполнены с возможностью хранения одной или более выполненных с возможностью конфигурирования структур для конфигурирования системы с целью обеспечения возможности выполнения одной или более функций аналитической обработки на данных, таких как данные, содержащие геномную последовательность, представляющую интерес, или относящиеся к ней обработанные результирующие данные. Память может содержать интересующую геномную последовательность, которую нужно обработать, например, после того, как она сформирована и/или получена, одну или более референсных генетических последовательностей и/или может дополнительно содержать индекс одной или более генетических референсных последовательностей и/или список относящихся к ним границ сплайсинга. Система может также включать в себя устройство ввода, имеющее интерфейс прикладных программ (API) платформы для выбора из списка вариантов одной или более структур обработки, выполненных с возможностью конфигурирования, например, путем выбора функций обработки системы, которые будут выполняться на данных, например, предварительной или последующей обработки геномных последовательностей, представляющих интерес. Возможно также наличие графического пользовательского интерфейса (ГПИ), которые выполнен с возможностью функционального связывания с API, например, для предоставления меню, с помощью которого пользователь может выбирать, какие из имеющихся вариантов требуется выполнить на данных.

[00873] Следовательно, в этих и/или других таких случаях гибридное облако 50 может быть выполнено с возможностью обеспечения возможности беспрепятственной и

защищенной передачи данных всем компонентам системы, например, когда гибридное облако 50 выполнено с возможностью разрешения различным пользователям системы конфигурировать ее составляющие части и/или саму систему, например, с помощью WMS, для удовлетворения потребностей пользователей в области исследовательских, диагностических, терапевтических и/или профилактических открытий и/или разработок. В частности, гибридное облако 50 и/или различные компоненты системы 1 могут быть выполнены с возможностью функционального соединения с совместимыми и/или соответствующими интерфейсами API, которые выполнены с возможностью разрешения пользователям удаленного конфигурирования различных компонентов системы 1 для развертывания требуемых ресурсов нужным образом, причем локальным, удаленным или комбинированным способом, например, на основе потребностей системы и особенностей выполняемых анализов, обеспечивая при этом обмен данными в защищенной среде с возможностью шифрования.

[00874] Как описано выше, система может быть реализована на одной или более интегральных схем, которые могут быть сформированы из одного или более наборов выполненных с возможностью конфигурирования, например, предварительного конфигурирования, или жестко смонтированных цифровых логических схем, которые могут быть взаимно соединены с помощью множества физических электрических межсоединений. В таком случае интегральная схема может иметь вход, например, интерфейс памяти, для приема одного или множества протоколов структуры, выполненных с возможностью конфигурирования, например, из памяти, и может быть также выполнена с возможностью реализации одной или более структур на интегральной схеме в соответствии с протоколами структуры обработки, выполненными с возможностью конфигурирования. Интерфейс памяти входа может быть также выполнен с возможностью приема данных геномной последовательности, которые могут быть в виде множества ридов геномных данных. Интерфейс может быть также выполнен с возможностью доступа к одной или более генетических референсных последовательностей и индексу (-ам).

[00875] В различных случаях цифровые логические схемы могут выполнены в виде набора движков обработки, каждый из которых сформирован из подмножества цифровых логических схем. Цифровые логические схемы и/или движки обработки могут быть выполнены с возможностью осуществления одного или более предварительно конфигурируемых этапов протокола первичной, вторичной и/или третичной обработки для формирования множества ридов данных геномной последовательности и/или обработки множества ридов геномных данных, например, в соответствии с генетическими референсными последовательностями или другой информацией, полученной из генетической последовательности. Интегральная схема может также иметь выход, чтобы выводить результирующие данные первичной, вторичной и/или третичной обработки, например, в соответствии с интерфейсом прикладных программ (API) платформы.

[00876] В частности, в различных вариантах реализации цифровые логические схемы и/или наборы движков обработки могут образовывать множество конвейеров геномной обработки, например, когда каждый конвейер может иметь вход, который определен в соответствии интерфейсом прикладных программ платформы, для приема платформой биоинформационной обработки результирующих данных первичной и/или вторичной обработки и для выполнения на них одного или более аналитических процессов с целью создания результирующих данных. Кроме того, множество конвейеров геномной обработки могут иметь общий API конвейера, который определяет операцию вторичной

и/или третичной обработки, которую нужно выполнить на результирующих данных первичной и/или вторичной обработки, например, когда каждый из множества конвейеров геномной обработки выполнен с возможностью осуществления подмножества операций вторичной и/или третичной обработки и вывода результирующих данных вторичной и/или третичной обработки в соответствии с API конвейера.

[00877] В таких случаях в памяти и/или связанном репозитории приложений, выполненном с возможностью поиска, могут храниться множество приложений геномного анализа, например, когда каждое из множества приложений геномного анализа может быть доступно компьютеру посредством электронного носителя, например, для исполнения процессором компьютера, с целью осуществления намеченного анализа геномных данных из предварительной или последующей обработки результирующих данных первичной, вторичной и/или третичной обработки, например одним или более из множества конвейеров геномной обработки. В конкретных случаях каждое из множество приложений геномного анализа может быть определено интерфейсом API и может быть выполнено с возможностью приема результирующих данных первичной, вторичной и/или третичной обработки и/или осуществления намеченного анализа геномных данных предварительной или последующей обработки и вывода результирующих данных намеченного анализа в одну или более геномных баз данных.

[00878] Способ может дополнительно включать в себя выбор, например, в меню ГПИ, одного или более конвейеров геномной обработки из множества имеющихся конвейеров геномной обработки системы; выбор одного или более приложений геномного анализа из множества приложений геномного анализа, которые хранятся в репозитории приложений; и исполнение с помощью процессора компьютера одного или более выбранных приложений геномного анализа для осуществления намеченного анализа геномных данных из результирующих данных первичной, вторичной и/или третичной обработки.

[00879] Кроме того, в различных вариантах реализации все из картирования, выравнивания, сортировки и определения вариантов может происходить на микросхеме, и в различных вариантах реализации локальное повторное выравнивание, маркировка дубликатов, перекалибровка оценки качества оснований и/или один или более протоколов и/или конвейеров третичной обработки тоже могут выполняться на микросхеме или в программном обеспечении, и в различных случаях различные протоколы сжатия, такие как SAM, и/или BAM, и/или CRAM, тоже могут выполняться на микросхеме. Однако после того, как данные в результате первичной, вторичной и/или третичной обработки созданы, они могут быть сжаты, например, перед передачей, такой как оправка по всей системе, отправка на облако, например, для выполнения модуля определения вариантов, платформы вторичной, третичной или другой обработки, например, включая протокол анализа интерпретации и/или совместной работы. Это может быть полезно, особенно с учетом того факта, что определение вариантов, в том числе их третичная обработка, может быть «стрельбой по движущейся мишени», например, стандартизованного согласованного алгоритма, используемого в данной отрасли, нет.

[00880] Поэтому для достижения различных типов результатов могут использоваться различные алгоритмы, например, удаленным пользователем, и, следовательно, полезно иметь облачный модуль для выполнения данной функции, чтобы обеспечить гибкость при выборе алгоритма, полезного в данный конкретный момент, а также

последовательной или параллельной обработки. Соответственно, любой из модулей, описанных в настоящем документе, может быть реализован либо аппаратно, например, на микросхеме, либо программно, например, на облаке, но в определенных вариантах реализации все модули могут быть выполнены с возможностью осуществления их функций только на микросхеме, или все модули могут быть выполнены с возможностью осуществления их функций удаленно, например, на облаке, или может быть смесь модулей, где некоторые из них находятся на одной или более микросхем, а другие расположены на облаке. Кроме того, как было указано, в различных вариантах реализации сами микросхемы могут быть выполнены с возможностью функционирования совместно, а в некоторых вариантах реализации, в непосредственном взаимодействии с генетическим секвенатором, таким как СНП и/или секвенатор на микросхеме.

[00881] Точнее говоря, в различных вариантах реализации устройство по настоящему изобретению может быть микросхемой, такой как микросхема, которая выполнена с возможностью обработки геномных данных, например, путем использования конвейера модулей анализа данных. Соответственно, как показано на ФИГ. 45, предложена микросхема 100 процессора геномного конвейера вместе со связанным аппаратным обеспечением геномной конвейерной процессорной системы 10. Микросхема 100 имеет одно или более соединений 102 с внешней памятью («Контроллер памяти DDR3») и соединение 104 (например, интерфейс PCIe или QPI) с внешним миром, таким как, например, главный компьютер 1000. Коммутатор 108 (например, переключатель) обеспечивает доступ к интерфейсам памяти различным инициаторам запросов. Движки 110 DMA передают данные с высоко скоростью между главным устройством и внешними памятями 102) процессора 100 микросхемы (через коммутатор 108) и/или между указанным главным устройством и центральным контроллером 112. Центральный контроллер 112 управляет операциями микросхемы, в частности, координирует действия множества движков 13 обработки. Движки обработки сформированы из набор жестко смонтированных цифровых логических схем, которые взаимно связаны физическими электрическими соединениями и организованы в кластеры 11/114 движков. В некоторых реализациях движки 13 в одном кластере 11/114 совместно используют один порт коммутатора посредством арбитра 115. Центральный контроллер 112 имеет соединения с каждым кластером движков. Каждый кластер 11/114 движков имеет ряд движков 13 обработки для обработки геномных данных, в том числе сопоставитель 120 (или модуль картирования), выравниватель 122 (или модуль выравнивания) и сортировщик 124 (или модуль сортировки) и могут быть предусмотрены один или более движков обработки для выполнения других функций, таких как определение вариантов. Следовательно, кластер 11/114 движков может содержать также другие движки, такие как модуль определителя вариантов.

[00882] В соответствии с моделью потока данных, согласующейся с реализациями, описанными в настоящем документе, главное ЦПУ 1000 посылает команды и данные через движки 110 DMA в центральный контроллер 112, который равномерно распределяет данные между движками 13 обработки. Движки обработки возвращают обработанные данные в центральный контроллер 112, который передает их в потоковом режиме обратно в главное устройство посредством движков 110 DMA. Эта модель потока данных приспособлена для картирования, и выравнивания и определения вариантов. Как было указано, в различных случаях обмен данными с главным ЦПУ может осуществляться посредством относительно слабого или жесткого связывания, например, с помощью широкополосного межсоединения с малой задержкой, такого

как QPI, для поддержания когерентности кэша между связанными элементами памяти двух или более устройств.

[00883] Например, в различных случаях ввиду различных ограничений по питанию и/или пространству, например, при выполнении анализа больших данных, такого как картирование/выравнивание/определение вариантов в среде с гибридным программным/аппаратным ускорением, как описано в настоящем документе, где требуется быстро и беспрепятственно перемещать данные между устройствами системы, для выполнения таких передач по всей системе туда и обратно между связанными устройствами, например, туда и обратно в секвенатор, цифровой сигнальный процессор (ЦСП), ЦПУ и/или ГПУ или гибридный ЦПУ/ГПУ, ускоренную интегральную схему, например, FPGA, ASIC (на сетевой карте), а также другие интеллектуальные сетевые ускорители, быстро и с поддержанием когерентности кэша может быть полезен интерфейс тесного связывания с поддержанием когерентности кэша. В таких случаях подходящее межсоединение жесткого связывания с поддержанием когерентности кэша может быть межсоединением по одной или более технологиям одинарного межсоединения, выполненным с возможностью обеспечения обработки, например, между множеством платформ обработки, использующих разные архитектуры набора команд (АНК), при когерентном совместном использовании данных между разными платформами и/или с одним или более связанных ускорителей, таким как, например, жестко смонтированный ускоритель на FPGA, для обеспечения эффективного разнородного вычисления и тем самым значительного улучшения вычислительной эффективности системы, которая в различных случаях может быть выполнена в виде облачной серверной системы. Следовательно, в определенных случаях может быть использован протокол широкополосного межсоединения с малой задержкой и поддержанием когерентности кэша, такой как QPI, когерентный интерфейс ускорителя процессора (CAPI), NVLink/ГПУ или любой другой подходящий протокол межсоединения для ускорения различных передач данных между различными компонентами системы, например, относящимися к вычислительным функциям картирования, выравнивания и/или определения вариантов, которые могут включать в себя использование движков ускорения, для функционирования которых требуются получение доступа, обработка и беспрепятственное перемещение данных между различными компонентами системы вне зависимости от места нахождения различных подлежащих обработке данных в системе. И, когда такие данные хранятся в связанном запоминающем устройстве, таком как ОЗУ или DRAM, действия по передаче данных могут также включать в себя ускоренный и когерентный поиск и обработку базы данных в памяти.

[00884] В частности, в конкретных вариантах реализации такие разнородные вычисления могут включать в себя множество архитектур обработки и/или ускорения, которые могут быть взаимно соединены в формате вычислений с сокращенным набором команд. В таком случае это устройство межсоединения может быть устройством когерентное соединение-6 (CCVI), которое выполнено таким образом, что позволяет всем вычислительным компонентам внутри системы применять операции адресации, чтении и/или записи к одной или более связанным памятям единым согласованным и когерентным образом. Более конкретно, межсоединение CCVI может быть использовано для соединения различных устройств системы, таких как ЦПУ и/или ГПУ или гибридное ЦПУ/ГПУ, FPGA, и/или связанных памятей и т.д. друг с другом, например, с широкой полосой пропускания, и выполнено с возможностью повышения скоростей передачи между различными компонентами, о чем свидетельствует резкое сокращение уровней задержки. А именно, межсоединение CCVI может быть использовано и

сконфигурировано таким образом, чтобы все компоненты системы получали доступ данным и обрабатывали их независимо от того, где находятся данные, и без необходимости в сложных средах программирования, которые в противном случае пришлось бы реализовывать для обеспечения когерентности данных. В число других таких межсоединений, которые могут быть использованы для ускорения, например, уменьшения времени, и повышения точности обработки, входят QPI, CAPI, NVLink или другие межсоединения, которые могут быть выполнены с возможностью взаимного соединения различных компонентов системы и/или работать поверх связанного периферийного межсоединения PCI-express.

[00885] Поэтому в соответствии с альтернативной моделью потока данных, согласующейся с реализациями, описанными в настоящем документе, главное ЦПУ 1000 в потоковом режиме передает данные во внешнюю память 1014 либо непосредственно через движки 110 DMA и коммутатор 108, либо через центральный контроллер 112. Главное ЦПУ 1000 посылает команды в центральный контроллер 112, который посылает в движки 13 обработки команды, указывающие движкам обработки, какие данные обрабатывать. Ввиду жесткого связывания движки 13 обработки вводят данные прямо из внешней памяти 1014 или связанного с ней кэша, обрабатывают их и записывают результаты обратно во внешнюю память 1014, например, посредством жестко связанного межсоединения 3, сообщая статус в центральный контроллер 112. Центральный контроллер 112 либо отправляет результирующие данные в потоковом режиме обратно в указанное главное устройство 1000 из внешней памяти 1014, либо уведомляет главное устройство, чтобы оно само извлекло результирующие данные посредством движков 110 DMA.

[00886] На ФИГ. 46 приведены процессор конвейера геномной обработки и система 20, показывающие полный комплект движков 13 обработки внутри кластера 11/214 движков. Конвейерная процессорная система 20 может содержать один или более кластеров 11/214 движков. В некоторых реализациях конвейерная процессорная система 20 может содержать четыре или более кластеров 11/214 движков. В число движков 13 обработки или типов движков обработки могут входить, без ограничений, сопоставитель, выравниватель, сортировщик, локальный повторные выравниватель, перекалибровщик оценки качества оснований, маркировщик дубликатов, определитель вариантов, средство сжатия и/или распаковки. В некоторых реализациях каждый кластер 11/214 движков имеет по одному движку обработки каждого типа. Соответственно, все движки 13 обработки одного типа могут получать доступ к коммутатору 208 одновременно через разные порты коммутатора, так кластеры 11/214, в которых находится каждый из них, разные. Формирование в каждом кластере 11/214 движка обработки каждого типа не требуется. Типы движков обработки, которые требуют огромной параллельной обработки или пропускной способности памяти, такие как сопоставитель (с прикрепленными к нему выравнивателями) и сортировщик, могут оказаться в каждом кластере движков конвейерной процессорной системы 20. Движки других типов могут появляться только в одном или нескольких кластерах 214 движков по мере необходимости для удовлетворения требований к их производительности или производительности конвейерной процессорной системы 20.

[00887] На ФИГ. 47 приведена конвейерная процессорная система 30 для генома, показывающая в дополнение к кластерам 11 движков, описанным выше, одно или более внедренных центральных процессорных устройств (ЦПУ) 302. В число таких примеров внедренных ЦПУ входят ядра Snapdragon® или стандартные ядра ARM®, или в других случаях это могут быть FPGA. Эти ЦПУ исполняют полностью программируемые

биоинформационные алгоритмы, такие как улучшенное определение вариантов, например, построение DBG или выполнение HMM. Такую обработку ускоряют с помощью вычислительных функций в различных кластерах 11 движков, которые могут быть вызваны ядрами 302 ЦПУ по мере надобности. Кроме того, даже ориентированная на движки обработка, такая как картирование или выравнивание, может выполняться ядрами 302 ЦПУ, обеспечивая их повышенную программируемость.

[00888] На ФИГ. 48 показан поток обработки для конвейерной процессорной системы и способа для генома. В некоторых предпочтительных реализациях данные обрабатывают в три прохода. Первый проход включает в себя картирование 402 и выравнивание 404, причем через движки 13 прогоняют полный набор ридов. Второй проход включает в себя сортировку 406, где один большой блок, подлежащий сортировке (например, существенную часть всех ридов, ранее картированных на одну хромосому) загружают в память, сортируют с помощью движков обработки и возвращают в главное устройство. Третий проход включает в себя следующие по цепочке стадии (локальное повторное выравнивание 408, маркировку 410 дубликатов, перекалибровку 412 оценки качества оснований (BQSR), вывод 414 SAM, вывод 416 редуцированного BAM и/или сжатие 418 CRAM). Этапы и функции третьего прохода могут быть выполнены в любой комбинации или подкомбинации и в любом порядке за один проход.

[00889] Следовательно, таким образом данные проходят относительно беспрепятственно из одного или более движков обработки в главное ЦПУ, например, в соответствии с одним или более методами, описанными в настоящем документе. Таким образом, виртуальную конвейерную архитектуру, например, описанную выше, используют для потоковой передачи ридов из главного устройства в циклические буферы в памяти через один движок обработки за другим последовательно и обратно в главное устройство. В некоторых реализациях распаковка CRAM может быть отдельной функцией потоковой передачи. В некоторых реализациях вывод 414 SAM, вывод 416 редуцированного BAM и/или компрессия 418 CRAM могут быть заменены определением вариантов, сжатием и распаковкой.

[00890] В различных случаях описана аппаратная реализация конвейера анализа последовательности. Это можно сделать множеством различных способов, например при помощи варианта реализация с использованием FPGA, ASIC или структурированной ASIC. Функциональные блоки, которые реализованы с помощью FPGA, ASIC или структурированной ASIC, показаны на ФИГ. 49. Соответственно, система включает в себя ряд блоков или модулей для выполнения анализа последовательности. Входными данными аппаратной реализации может быть файл FASTQ, но этот формат не является ограничением. Помимо файла FASTQ входные данные FPGA, или ASIC, или структурированной ASIC содержат вспомогательную информацию, например, Информацию об объеме потока (Flow Space Informatio)n, относящуюся к технологии, такой как СНП. В число блоков или модулей могут входить следующие блоки: исправление ошибок, картирование, выравнивание, сортировка, локальное повторное выравнивание, маркировка дубликатов, перекалибровка оценки качества оснований, сокращение BAM и побочной информации и/или определение вариантов.

[00891] Эти блоки или модули могут присутствовать внутри, или могут быть реализованы аппаратно, но для достижения цели реализации конвейера анализа последовательности некоторые из указанных блоков могут быть опущены, а другие блоки добавлены. Блоки 2 и 3 описывают два альтернативных варианта платформы с конвейером анализа последовательности. Платформа с конвейером анализа

последовательности содержит FPGA, ASIC или структурированную ASIC и программное обеспечение, поддерживаемое главным устройством (например, ПК, сервером, кластером или средствами облачного вычисления) с помощью облачного и/или кластерного хранилища. Блоки 4-7 описывают различные интерфейсы, которые может иметь конвейер анализа последовательности. В блоках 4 и 6 интерфейс может представлять собой интерфейс PCIe и/или QPI/CAPI/CCVI/NVLink, но не ограничиваясь PCIe, QPI или другим интерфейсом. В блоках 5 и 7 аппаратное обеспечение (FPGA, или ASIC, или структурированная ASIC) может быть непосредственно интегрировано в секвенатор. Блоки 8 и 9 описывают интеграцию аппаратного конвейера анализа последовательности, встроенного в главную систему, такую как ПК, кластер серверов или секвенатор. Аппаратное обеспечение FPGA, ASIC или структурированной ASIC окружают множество элементов памяти DDR3 и интерфейс PCIe/QPI/CAPI/CCVI/NVLink. Плата с FPGA/ASIC/sASIC соединена с главным компьютером, состоящим из главного ЦПУ и/или ГПУ, которое может быть маломощным ЦПУ, таким ка ARM®, Snapdragon® или любой другой процессор. Блок 10 иллюстрирует API аппаратного конвейера анализа последовательности, который может быть доступен приложениям третьей стороны для выполнения третичного анализа.

[00892] На ФИГ. 50A и 50B изображена плата 104 расширения, имеющая микросхему 100, например, FPGA, по данному изобретению, а также один или более соответствующих элементов 105 для связывания FPGA 100 с г главным ЦПУ/ГПУ, например, для передачи данных, таких как данные, подлежащие обработке, и результирующие данные, туда и обратно из ЦПУ/ГПУ в FPGA 100. На ФИГ. 50B изображена плата расширения, показанная на ФИГ. 50A, которая имеет множество, например 3 гнезд, содержащих множество, например, 3, микросхемы обработки по данному изобретению.

[00893] В частности, как показано на ФИГ. 50A и 50B, в различных вариантах реализации устройство по настоящему изобретению может содержать вычислительную архитектуру, например, внедренную в кремниевую программируемую пользователем вентильную матрицу (FPGA) или специализированную интегральную схему (ASIC) 100. FPGA 100 может быть интегрирована в печатную плату (ПП) 104, такую как плата интерфейса периферийных компонентов типа экспресс (PCIe), которую можно вставить в вычислительную платформу. В различных случаях, как показано на ФИГ. 50A, плата 104 PCIe может содержать одну матрицу FPGA 100, которая может быть окружена локальными памятями 105, однако в различных вариантах реализации, как показано на ФИГ. 50B, плата 104 PCIe может содержать множество матриц FPGA 100A, 100B и 100C. В различных случаях плата PCI может также содержать шину PCIe. Плата 104 PCIe может быть добавлена на вычислительную платформу для исполнения алгоритмов на чрезвычайно больших наборах данных. В альтернативном варианте реализации, который упомянут выше со ссылкой на ФИГ. 34, в различных вариантах реализации матрица FPGA может быть выполнена с возможностью прямого соединения с ЦПУ/ГПУ, например, посредством перемычки, и жестко связана с ним, например посредством интерфейса QPI, CAPI, CCVI. Соответственно, в различных случаях весь рабочий поток секвенирования генома, задействующий матрицу FPGA, может включать в себя следующее: подготовку образца, выравнивание (включая картирование и выравнивание), анализ вариантов, биологическую интерпретацию и/или специальные приложения.

[00894] Следовательно, в различных вариантах реализации устройство по настоящему изобретению может содержать вычислительную архитектуру, которая достигает высокоэффективного исполнения алгоритмов, таких как алгоритмы картирования и выравнивания, которые работают на чрезвычайно больших наборах данных, например,

когда наборы данных проявляют плохую локальность ссылок (LOR). Эти алгоритмы предназначены для реконструкции всего генома из миллионов последовательностей коротких ридов из современных, так называемых, секвенаторов нового поколения, требующей многогигабайтные структуры данных с произвольным доступом. По достижении реконструкции, как описано выше в настоящем документе, используют дальнейшие алгоритмы с аналогичными характеристиками для сравнения одного генома с библиотеками других, выполнения функционального анализа генов и т.д.

[00895] Существуют две другие типичные архитектуры, которые, как правило, могут быть построены для выполнения одной или более операций, подробно описанных в настоящем документе, в том числе, например, многоядерные ЦПУ общего назначения и графические процессорные устройства общего назначения (ГПУОН). В таком случае каждый ЦПУ/ГПУ в многоядерной системе может иметь классическую архитектуру на основе кэша, где инструкции и данные извлекаются из кэша уровня 1 (кэш L1), который мал, но обладает чрезвычайно быстрым доступом. Множество кэшей L1 могут быть соединены с более крупным, но более медленным кэшем L2. Кэш L2 может быть соединен с большой, но более медленной системой памяти DRAM (динамическое оперативное запоминающее устройство), или может быть соединен с еще более крупным, но более медленным кэшем L3, который может быть затем соединен с DRAM. Преимущество такой компоновки может заключаться в том, что приложения, в которых программы и данные проявляют локальность ссылок, ведут себя почти так, как если бы они исполнялись на компьютере с одним запоминающим устройством, большим как DRAM, но быстрым как кэш L1. Поскольку полностью заказные, в высшей степени оптимизированные ЦПУ работают при очень высоких тактовых частотах, например от 2 до 4 ГГц, эта архитектура может быть существенна для достижения хороших рабочих характеристик. Кроме того, как подробно обсуждалось со ссылкой на ФИГ. 33, в различных вариантах реализации ЦПУ может быть жестко связано с FPGA, например, с FPGA, выполненной с возможностью осуществления одной или более функций, относящихся к различным операциям, описанным в настоящем документе, например, посредством QPI, CCVI, CAPI, для дальнейшего улучшения рабочих характеристик, а также скорости и когерентности данных, передаваемых по всей системе. В таком случае между двумя устройствами может поддерживаться когерентность кэша, как отмечено выше.

[00896] Кроме того, эту архитектуру можно расширить с помощью ГПУОН, например, за счет реализации очень большого количества малых ЦПУ, каждый со своим кэшем L1, причем каждый ЦП исполняет одни и те же инструкции на различных подмножествах данных. Это, так называемая архитектура ОКМД (один поток команд, много потоков данных). Экономии можно достичь за счет совместного использования логики выборки и декодирования команды по всем большому количеству ЦПУ. Каждый кэш имеет доступ ко множеству больших внешних DRAM через сеть межсоединений. В предположении, что вычисление должно выполняться с возможностью высокого распараллеливания, ГПУОН имеют значительное преимущество над ЦПУ общего назначения благодаря большому количеству вычислительных ресурсов. Тем не менее, они все равно имеют архитектуру с кэшированием и их рабочие характеристики ухудшаются приложениями, которые не обладают достаточной высокой степенью локальности ссылок. Это приводит к высокому коэффициенту непопадания при обращении к кэшу и простою процессора в ожидании поступления данных из внешнего DRAM.

[00897] Например, в различных случаях в качестве системной памяти могут

использоваться динамические ОЗУ, поскольку они более экономичные, чем статические ОЗУ (СОЗУ). Приблизительно можно считать, что при одинаковой стоимости объем DRAM в 4 раза превосходит объем СОЗУ. Однако ввиду падения спроса на СОЗУ в пользу DRAM, разрыв между ними увеличился вследствие значительной экономии пространства, которую предлагают DRAM, которые пользуются большим спросом. Независимо от стоимости DRAM в 4 раза плотнее, чем СОЗУ, размещенные на одинаковой площади кремния, так как для них требуются только один транзистор и одна емкость на бит по сравнению с 4 транзисторами на бит для реализации триггера в СОЗУ. DRAM представляет один бит информации как наличие или отсутствие заряда в емкости.

[00898] Проблема с такой структурой состоит в том, что заряд ослабевает со временем, поэтому его нужно периодически обновлять. Такая необходимость привела к архитектурам, организующим память в виде независимых блоков и механизмов доступа, которые выдают множество слов памяти на каждый запрос. Это компенсирует время, когда данный блок недоступен во время обновления. Идея состоит в перемещении огромного количества данных, пока данный блок доступен. В этом отличие от СОЗУ, в котором любое место памяти доступно за одно обращение в течение постоянного количества времени. Данная характеристика позволяет при обращении к памяти ориентироваться на одно слово, а не блок. DRAM работают хорошо в архитектуре с кэшированием, так как каждое непопадание в кэш приводит к блоку памяти, считываемому из DRAM. Теория локальности ссылок состоит в том, что сразу после обращения к слову N, вероятно, последует обращение к словам N + 1, N + 2, N + 3 и т.д.

[00899] На ФИГ. 51 приведен пример реализации системы 500 по настоящему изобретению, содержащей одну или более плат расширения, изображенных на ФИГ. 50, например, для биоинформационной обработки 10. Система включает в себя микросхему 100 биоинформационной обработки, выполненную с возможностью осуществления одной или более функций в конвейере обработки, такой как определение оснований, исправление ошибок, картирование, выравнивание, сортировка, сборка, определение вариантов и т.п., как описано в настоящем документе.

[00900] Система 500 также содержит диспетчер конфигураций, который выполнен с возможность конфигурирования на плате одного или более процессором 100. А именно, в различных вариантах реализации диспетчер конфигураций выполнен с возможностью передачи инструкций во внутренний контроллер FPGA, например в прошивку, например, посредством соответствующим образом сконфигурированного драйвера по слабо или жестко связанному межсоединению, для конфигурирования одной или более функций обработки системы 500. Например, диспетчер конфигураций может быть выполнен с возможностью конфигурирования внутренних кластеров 11 и/или связанных с ним движков 13 обработки для выполнения одной или более требуемых операций, таких как картирование, выравнивание, сортировка, определение вариантов и т.п. в соответствии с принятыми инструкциями. Таким образом, только кластеры 11, содержащие движки 13 обработки для выполнения запрошенных операций обработки на данных, предоставленных из главной системы 1000 в микросхему 100, могут быть задействованы для обработки этих данных в соответствии с принятыми инструкциями.

[00901] Кроме того, в различных вариантах реализации диспетчер конфигураций может быть также выполнен с возможностью адаптации его самого, например, пользователем третьей стороны, например, посредством соединения API, как более подробно описано выше в настоящем документе, например, с помощью пользовательского интерфейса (ГПИ), представленного приложением системы 500.

Кроме того, диспетчер конфигураций может быть подключен к одной или более внешних памятей, такой как память, формирующая или иным образом содержащая базу данных, например, базу данных, содержащую один или более референсов, или секвенированных по отдельности геномов, и/или их индекс, и/или один или более ранее картированных, выровненных и/или отсортированных геномов или их частей. В различных случаях база данных может также содержать один или более генетических профилей, характеризующих болезненное состояние, например, для выполнения одного или более протоколов третичной обработки, например, на вновь картированных, выровненных генетических последовательностях или относящемся к ним файле VCF.

[00902] Система 500 может включать в себя веб-доступ для обеспечения удаленного обмена данными, например, через Интернет, с целью формирования платформы обмена данными посредством облака или по меньшей мере гибридного облака 504. Подобным образом обработанная информация, сформированная биоинформационным процессором, например, данные результатов, могут быть зашифрованы и сохранены в виде электронной записи здоровья, например, во внешней, такой как удаленная, базе данных. В различных случаях база данных EMR может быть выполнена с возможностью поиска в ней, например, в хранящейся там генетической информации, для выполнения одного или более статистических анализов на данных, например, определения болезненных состояний или тенденций, или в целях анализа эффективности в отношении них одной или более профилактических или терапевтических мер. Такая информация вместе с данными EMR может быть затем дополнительно обработана и/или сохранена в еще одной базе 508 данных таким образом, чтобы гарантировать конфиденциальность источника генетической информации.

[00903] Более конкретно, на ФИГ. 51 показана система 500 для исполнения конвейера анализа последовательности на данных генетической последовательности. Система 500 содержит диспетчер 502 конфигураций, который включает в себя вычислительную систему. Вычислительная система диспетчера 502 конфигураций может содержать персональный компьютер или другую компьютерную рабочую станцию, или может быть реализована комплектом сетевых компьютеров. Диспетчер 502 конфигураций может также включать в себя одно или более третьесторонних приложений, соединенных с вычислительной системой посредством одного или более API, которые вместе с одним или более частных приложений формируют конфигурацию для обработки данных генома из секвенатора или другого источника данных генома. Диспетчер 502 конфигураций также включает в себя драйверы, которые загружают конфигурацию в конвейерную процессорную систему 10 для генома. Конвейерная процессорная система 10 для генома может выводить результирующие данные в сеть Интернет 504 или другую сеть, или быть доступна через них, для сохранения результирующих данных в электронной записи 506 о здоровье или другой базе 508 данных знаний.

[00904] Как не раз отмечалось выше в настоящем документе, микросхем, реализующая конвейерный процессор для генома, может быть соединена или интегрирована с секвенатором. Эта микросхема может быть также соединена или интегрирована, например, напрямую посредством перемычки, или опосредованно, например, может находиться на плате расширения, такой как плата PCIe, а плата расширения может быть соединена или интегрирована с секвенатором. В других реализациях микросхема может быть соединена или интегрирована с серверным компьютером, который соединен с секвенатором, чтобы передавать риды геномов из секвенатора на сервер. В еще одних реализациях микросхема может быть соединена или интегрирована с сервером в облачном вычислительном кластере компьютеров и серверов. Система может включать

в себя один или более секвенаторов, подключенных (например, посредством Ethernet) к серверу, содержащему микросхему, причем риды геномов формируются множеством секвенаторов, передаются на сервер, и затем картируются и выравниваются в микросхеме.

[00905] Например, обычно в конвейерах данных секвенатора ДНК нового поколения обработка стадии первичного анализа, как правило, специфична для данной технологии секвенирования. Эту стадию первичного анализа выполняют для перевода физических сигналов, обнаруживаемых внутри секвенатора, в «риды» нуклеотидных последовательностей вместе с соответствующими оценками качества (достоверности), например в файлах формата FASTQ или других форматов, содержащих последовательность и, как правило, информацию о качестве. Первичный анализ, как упоминалось выше, по своей природе часто довольно специфичен для используемой технологии секвенирования. В различных секвенаторах нуклеотиды обнаруживаются путем измерения изменений в флуоресценции и/или электрических зарядах, электрических токах или излучаемом свете. Некоторые конвейеры первичного анализа часто включают в себя: обработку сигнала для усиления, фильтрации, разделения и измерения выходного сигнала датчика; уменьшение объема данных, например, путем разбиения на подгруппы, прореживания, усреднения, преобразования и т.д.; обработку изображения или цифровую обработку с целью выявления и усиления имеющих значение сигналов и связывания их с конкретными ридами и нуклеотидами (например, вычисление смещения изображения, идентификацию кластера); алгоритмическую обработку и эвристическую процедуру для компенсации артефактов технологии секвенирования (например, оценку фазирования, матрицы перекрестных помех); вычисления байесовской вероятности; скрытые марковские модели; определение оснований (выбор наиболее вероятного нуклеотида в каждой позиции в последовательности); оценку качества (достоверности) определения оснований и т.п. Как описано в настоящем документе выше, один или более из этих этапов выиграют, если одну или более необходимых функций обработки реализовать в аппаратном обеспечении, например реализовать с помощью интегральной схемы, такой как FPGA. Кроме того, после получения такого формата переходят к вторичному анализу, как описано в настоящем документе, чтобы определить содержимое секвенированного образца ДНК (или РНК и т.д.), например, с помощью картирования и выравнивания ридов на референсный геном, сортировки, маркировки дубликатов, перекалибровки оценки качества оснований, локального повторного выравнивания и определения вариантов. Затем может следовать третичный анализ для извлечения медицинских и исследовательских заключений из определенного содержимого ДНК.

[00906] Соответственно, учитывая последовательный характер описанных выше функций обработки, было бы целесообразно объединить ускорение первичной, вторичной и/или третичной обработки в одной интегральной схеме и множестве интегральных схем, расположенных на одно плате расширения. Это может быть выгодно, поскольку секвенаторы создают данные, которые, как правило, требуют и первичного, и вторичного анализа, чтобы они были полезны и в дальнейшем можно было использовать в различных протоколах третичной обработки, и объединение их в одном устройстве является наиболее эффективной с точки зрения стоимости, пространства, электропитания и совместного использования ресурсов. Поэтому в соответствии с одним конкретным аспектом изобретение относится к системе, например, к системе, для исполнения конвейера анализа последовательности на данных генетической последовательности. В различных случаях система может включать в себя электронный источник данных, такой как источник данных, который обеспечивает

цифровые сигналы, например, цифровые сигналы, представляющие множество ридов геномных данных, где каждой из множества ридов геномных данных содержит нуклеотидную последовательность. Система может включать в себя одни или более памятей, таких как память, хранящая одну или более генетических референсных последовательностей и/или индекс одной или более генетических референсных последовательностей; и/или система может содержать микросхему, такую как ASIC, FPGA или sASIC.

[00907] Один или более аспектов или признаков объекта изобретения, описанного в настоящем документе, могут быть реализованы в цифровой электронной схеме, интегрированной схеме, специально разработанных специализированных интегральных схемах (ASIC), программируемых пользователем вентильных матрицах (FPGA) или структурированной ASIC, компьютерном аппаратном обеспечении, прошивке, программном обеспечении и/или их комбинациях.

[00908] Эти различные аспекты или признаки могут включать в себя реализацию в одной или более компьютерных программ, которые выполнены с возможностью исполнения или интерпретации в программируемой системе, включающей в себя по меньшей мере программируемый процессор, который может быть специальным или общего назначения и связан с системой хранения, для получения оттуда данных и инструкций и передачи туда данных и инструкций, по меньшей мере одно устройство ввода и по меньшей мере одно устройство вывода. Программируемые система или вычислительная система может включать в себя клиенты или серверы. Клиент и сервер обычно удалены друг от друга и, как правило, взаимодействуют посредством сети связи. Взаимоотношение клиента и сервера появляется благодаря компьютерным программам, выполняемым на соответствующих компьютерах и имеющих взаимосвязь клиент-сервер друг с другом.

[00909] Эти компьютерные программы, которые также можно назвать программами, программным обеспечением, программными приложениями, приложениями, компонентами или кодом, содержат машинные команды для программируемого процессора и могут быть реализованы на процедурном или объектно-ориентированном языке программирования высокого уровня и/или на ассемблере/машинном языке. Используемый в настоящем документе термин «машиночитаемый носитель информации» относится к любому компьютерному программному продукту, прибору и/или устройству, к такому как, например, магнитные диски, оптические диски, память, и программируемые логические устройства (ПЛУ), используемые для предоставления машинных команд и/или данных в выполненный с возможностью программирования процессор, в том числе машиночитаемый носитель информации, который принимает машинные команды как машиночитаемый сигнал. Термин «машиночитаемый сигнал» относится к любому сигналу, используемому для предоставления машинных инструкций и/или данных в программируемый процессор. Машиночитаемый носитель информации может хранить такие машинные команды некратковременно, например, как это делают некратковременная твердотельная память или накопитель на жестких магнитных дисках или любой другой эквивалентный носитель информации. Машиночитаемый носитель информации может в качестве альтернативы или дополнительно хранить такие машинные команды кратковременно, например, как это делает кэш процессора или другая память с произвольным доступом, связанная с одним или более физическими ядрами процессора.

[00910] Кроме того, ввиду колоссального роста в создании и получении данных в 21-ом веке появилась потребность в увеличении мощности обработки, которая в

состоянии справляться с выполнением анализов с постоянно растущей вычислительной интенсивностью, на которых основывается современное развитие. Появились суперкомпьютеры, которые помогли поступательному развитию технологии целого ряда платформ. Однако, хотя супервычисление полезно, его оказалось недостаточно для ряда очень сложных вычислительных проблем, с которыми сегодня сталкиваются технологические компании. В частности, со времени секвенирования генома человека произошло экспоненциальное развитие технологии в биологических областях. Тем не менее, поскольку сложность ежедневно создаваемых необработанных данных растет высокими темпами, в обработке и анализа формируемых данных образовалось трудноразрешимое узкое место. Поэтому для устранения этого узкого места были разработаны квантовые компьютеры. Квантовое вычисление представляет передний край в области вычислений, обеспечивая абсолютно новый подход к удовлетворению самых сложных вычислительных потребностей в мире.

[00911] Квантовое вычисление известно с 1982 г. Например, Richard Feynman в журнале International Journal of Theoretical Physics теоретически описал систему для выполнения квантового вычисления. А именно, Richard Feynman предложил квантовую систему, которую можно было бы сконфигурировать для использования в моделировании других квантовых систем таким образом, чтобы традиционные функции компьютерной обработки могли бы выполняться более быстро и эффективно. См. работу Feynman, 1982, International Journal of Theoretical Physics 21, pp. 467-488, которая полностью включена в настоящий документ путем ссылки. В частности, квантовая компьютерная система может быть выполнена с возможностью экспоненциальной экономии времени при сложных вычислениях. Такие управляемые квантовые системы общеизвестны как квантовые компьютеры и были успешно разработаны в компьютерах для обработки общего назначения, которые можно использовать не только для имитации квантовых систем, но и выполнить с возможностью выполнения специализированных квантовых алгоритмов. Более конкретно, сложные проблемы можно моделировать в вид уравнения, такого как функция Гамильтона, которое может быть представлено в квантовой системе таким образом, что поведение системы обеспечивает информацию о решении уравнения. См. работу Deutsch, 1985, Proceedings of the Royal Society of London A 400, pp. 97-117, которая полностью включена в настоящий документ путем ссылки. В таких случаях решение модели за счет поведения квантовой системы может быть выполнено таким образом, чтоб оно включало в себя решение дифференциального уравнения, относящегося к квантово-механическому описанию частицы, например, гамильтониана, квантовой системы.

[00912] По существу квантовое вычисление представляет собой вычислительную систему, которая использует квантово-механические явления, например, суперпозицию и/или запутанность, для выполнения различных вычислений на больших объемах данных с чрезвычайной скоростью. Поэтому квантовые компьютеры являются огромных улучшением по сравнению с традиционными цифровыми логическими компьютерами. В частности, традиционные цифровые логические схемы функционируют за счет использования двоичных цифровых логических вентилей, которые формируют путем жесткого монтажа электронной схемы на проводящем субстрате. В цифровой логической схеме состояние «включено/выключено» транзистора служит в качестве базовой единицы информации, например, бита. В частности, обычный цифровой компьютерный процессор для кодирования данных использует двоичные цифры, например, биты, в состоянии «включено» или «выключено», например, как 0 или 1. С другой стороны, квантовое вычисление для кодирования данных применяет

информационное устройство, которое использует суперпозиции запутанных состояний, называемые квантовыми битами или кубитами.

[00913] Основу выполнения таких квантовых вычислений составляет информационное устройство, которое формирует квантовый бит. Кубит является аналогом цифрового «бита» в традиционных цифровых компьютерах за исключением того, что вычислительный потенциал кубит намного больше, чем у цифрового бита. В частности, как описано более подробно в настоящем документе, вместо кодирования только двух дискретных состояний, вроде «0» или «1», как в случае цифрового бита, кубит может быть также приведен в состояние суперпозиции «0» или «1». А именно, кубит может существовать в обоих состояниях, «0» или «1», одновременно. Следовательно, кубит позволяет выполнять квантовое вычисление на обоих состояниях одновременно. Вообще, N кубитов могут быть в суперпозиции $2^N$ состояний. Поэтому в квантовых алгоритмах можно использовать это свойство суперпозиции для ускорения определенных вычислений.

[00914] Следовательно, кубит аналогичен биту в традиционном цифровом компьютере, и представляет собой тип информационного устройства, которое проявляет когерентность. В частности, квантовое вычислительное устройство построено из множества строительных блоков, например, кубитов, информационных устройств. Например, вычислительная мощность квантового компьютера возрастает по мере того, как информационные устройства, которые образуют его стандартные блоки, связываются, например, запутываются, вместе управляемым образом. В таком случае квантовое состояние одного информационного устройства влияет на квантовое состояние каждого из других информационных устройств, с которыми оно связано.

[00915] Соответственно, как и бит в классическом цифровом вычислении, кубит в квантовом вычислении служит базовой единицей для кодирования информации, такой как квантовая информация. Подобно биту кубит кодирует данные в системе из двух состояний, которая в данном случае является квантово-механической системой. В частности, для кубита два квантовых состояния подразумевают запутывание, например, включающее в себя поляризацию одного фотона. Поэтому, когда в классической системе бит должен находиться в одном или другом состоянии, в квантовой вычислительной платформе кубит может быть суперпозицией обоих состояний одновременно, и это свойство является фундаментальным для квантовой обработки. Следовательно, отличительным признаком между кубитом и классическим битом является то, что множество кубитов проявляют квантовую запутанность. Такая запутанность является нелокальным свойством, которое позволяет набору кубитов выражать более высокую корреляцию, чем возможно в классической системе.

[00916] Чтобы такие информационные устройства, например, квантовые биты, работали, они должны удовлетворять нескольким требованиям. Во-первых, информационное устройство должно быть выполнено с возможностью сведения его к квантовой двухуровневой системе. Это означает, что информационное устройство должно иметь два различимых квантовых состояния, которые могут быть использованы для выполнения вычислений. Во-вторых, информационные устройства должны быть в состоянии создавать квантовые эффекты вроде запутывания и суперпозиции. Кроме того, в определенных случаях информационное устройство может быть выполнено с возможностью хранения информации, например, квантовой информации, например в когерентной форме. В таких случаях когерентное устройство может иметь квантовое состояние, которое сохраняется без существенного ухудшения в течение длительного периода времени, например в течение порядка микросекунд или более.

[00917] В частности, квантовое запутывание представляет собой физическое явление, которое возникает при формировании или конфигурировании иным образом пары или группы частиц для взаимодействия таким образом, чтобы квантовое состояние одной частицы невозможно было описать независимо от других, несмотря на разделяющее их пространство. Следовательно, вместо описания состояния одной частицы отдельно от остальных квантовое состояние нужно описывать для системы в целом. В таких случаях измерения различных физических свойств, таких как положение, момент, спин и/или поляризация, выполняемые на запутанных частицах, являются коррелированными. Например, если пара частиц сформирована таким образом, что их общий спин известен и равен нулю, и обнаруживается, что одна частица имеет направленный по часовой стрелке спин на определенной оси, то спин другой частицы, измеряемые на этой же оси, окажется направленным против часовой стрелки, как и следовало ожидать ввиду их запутанности.

[00918] Следовательно, одна частицы запутанной пары просто «знает», какое измерение было выполнено на другой и с каким результатом, даже если не известно средств, при помощи которых такая информация должна была быть передана между частицами, которые на момент измерения могли быть удалены друг от друга на произвольно большие расстояния. Благодаря такой взаимосвязи, в отличие от классических битов, которые могут иметь только одно значение одновременно, запутанность позволяет действовать множеству состояний одновременно. Именно эти уникальные взаимосвязи и квантовые состояния были использованы при разработке квантового вычисления.

[00919] Соответственно, существуют различные виды физических операций, использующих чистые состояния кубита, которые могут быть выполнены. Например, можно сформировать и сконфигурировать квантовый логический вентиль для работы на основе кубита, где кубит подвергают унитарному преобразованию, например, когда унитарные преобразования соответствуют вращениям или иным квантовым явлениям кубита. Действительно, любая двухуровневая система может быть использована как кубит, например, фотоны, электроны, ядерные спины, состояния когерентного излучения, оптические решетки, переходы Джозефсона, квантовые точки и т.п. В частности, квантовый вентиль является основой для квантовой схемы, работающей на малом количестве кубитов. Например, квантовая схема содержит квантовые вентили, которые действуют на фиксированном количестве кубитов, например, двух, трех или более. Следовательно, кубиты являются строительными блоками квантовых схем подобно классическим логическим вентилям для традиционных цифровых схем. А именно, квантовая схема является моделью для квантового вычисления, где вычисление представляет собой последовательность квантовых вентилей, которые являются обратимыми преобразованиями на квантовом механическом аналоге n-битового регистра. Такие аналогичные структуры называют n-кубитовыми регистрами. Однако в отличие от классических логических вентилей квантовые логические вентили всегда обратимы.

[00920] В частности, как описано выше, цифровой логический вентиль является физическим, монтажным устройством, которое может быть реализовано с помощью одного или более диодов или транзисторов, которые действуют как электронные переключатели для выполнения логических операций, например, булевы функций, на одном или более двоичных входных сигналах с тем, чтобы создать один двоичный выходной сигнал. С помощью усиления логические вентили могут быть расположены каскадом точно также, как булевы функции, что позволяет строить физическую модель

всей булевой логики и, следовательно, все алгоритмы и математические соотношения, которые могут быть описаны с помощью булевой логики, могут быть выполнены с использованием цифровых логических вентилей. Подобным образом можно сформировать каскад квантовых логических вентилей для выполнения операций булевой логики.

[00921] Квантовые вентили обычно представляют в виде матриц. В различных реализациях квантовый вентиль действует на k кубитах, которые могут быть представлены унитарной матрицей размером 2k × 2k. В таких случаях количество кубитов на входе и выходе вентиля должно быть одинаковым, и действие вентиля на конкретном квантовом состоянии находят путем умножения вектора, представляющего состояние, на матрицу, представляющую вентиль. Следовательно, при такой конфигурации квантовые вычислительные операции могут быть исполнены на очень маленьком количестве квантовых битов. Например, существуют квантовые алгоритмы, которые выполнены с возможностью осуществления значительно более сложных вычислений быстрее, чем с помощью любого возможного вероятностного классического алгоритма. В частности, квантовый алгоритм - это алгоритм, который выполняется на модели квантовой схемы вычисления.

[00922] Тогда как классический алгоритм представляет собой конечную последовательность пошаговых команд или процедур, которые могут быть выполнены цифровыми логическими схемами классического компьютера, квантовый алгоритм является пошаговой процедурой, где каждый шаг может быть выполнен на квантовом компьютере. Однако, несмотря на существование квантовых алгоритмов, например, алгоритмов Шора, Гровера и Саймона, все классические алгоритмы тоже могут быть выполнены на квантовом компьютере с надлежащими конфигурациями. Квантовые алгоритмы обычно используют в качестве алгоритмов, которые квантовые по своей сути, например такие, которые используют суперпозицию или квантовую запутанность. Квантовые алгоритмы могут сформулированы в различных моделях квантового вычисления, такой как модель гамильтонова оракула.

[00923] Соответственно, в то время как классический компьютер имеет память, построенную из битов, где каждый бит представлен либо «1», либо «0», квантовый компьютер поддерживает последовательность кубитов, где один кубит может представлять единицу, нуль или квантовую суперпозицию этих двух состояний кубита. Соответственно, пара кубитов может быть любой квантовой суперпозицией 4 состояний, а три кубита могут быть любой суперпозицией 8 состояний. В целом квантовый компьютер с n кубитами может быть в произвольной суперпозиции $2^n$ различных состояний одновременно по сравнению с обычным компьютером, который может быть только в одном из $2^n$ состояний одновременно. Следовательно, кубиты могут содержать экспоненциально больше информации, чем их классические аналоги. Во время работы квантовый компьютер совершает операции путем установки кубитов в состояние дрейфа, что решает проблему путем манипулирования этими кубитами с помощью фиксированной последовательности квантовых логических вентилей. В этой последовательности квантовых логических вентилей, которая формирует операции квантовых алгоритмов. Вычисление завершается измерением, схлопывающим систему кубитов в одно из $2^n$ чистых состояний, где каждый кубит является «0» или «1», тем самым разлагая на классические состояния. Следовательно, традиционные алгоритмы тоже могут быть выполнены на квантовой вычислительной платформе, где результатом будет, как правило, n-битовая классическая информация.

[00924] В стандартном представлении базовые состояния кубита называют состояниями «0» и «1». Однако во время квантового вычисления состояние кубита, как правило, может быть суперпозицией базовых или базисных состояний, такой что кубит имеет ненулевую вероятность нахождения в базисном состоянии «0» и одновременно ненулевую вероятность нахождения в базисном состоянии «1». Соответственно, квантовая природа кубита в большой степени является производной от его способности существовать в когерентной суперпозиции базисных состояний и того, что состояние кубита должно иметь фазу. Кубит будет сохранять свою способность существования в когерентной суперпозиции базисных состояний до тех пор, пока он достаточно изолирован от источников декогерентности.

[00925] Поэтому, чтобы завершить вычисление с помощью кубита, измеряют состояние кубита. Как указано выше, при выполнении измерения кубита квантовый характер кубита может быть временно утерян, и суперпозиция базисных состояний может быть схлопнуто либо в базисное состояние «0», либо в базисное состояние «1». Поэтому подобным образом кубит снова становится похожим на традиционный цифровой «бит». Однако фактическое состояние кубита после схлопывания будет зависеть от различных вероятностных состояний, имевших место непосредственно перед операцией измерения. Таким образом, кубиты можно использовать для формирования квантовых схем, которые сами могут конфигурироваться образованием квантового компьютера.

[00926] Существуют несколько общих подходов к проектированию и эксплуатации квантового компьютера. Одним подходом, который был сформулирован, является схемная модель квантового вычисления. Схемная модель квантового вычисления требует длительной квантовой когеренции, поэтому типом информационного устройства, используемого в квантовых компьютерах, которые поддерживают такой подход, может быть кубит, который, по определению, обладает длительным временем когеренции. Соответственно, схемная модель квантового вычисления основана на предпосылке, что кубиты могут формироваться и подвергаться действию логических вентилей, во многом напоминая биты, и могут быть запрограммированы с помощью квантовой логики для выполнения вычислений, таких как булевы вычисления. Для разработки кубитов, которые могут быть запрограммированы для выполнения квантовых логических функций таким образом было проведено исследование. Например, см. работу Shor, 2001, arXiv.org:quant-ph/0005003, которая полностью включена в настоящий документ путем ссылки. Аналогичным образом, компьютерный процессор может принять форму квантового процессора в виде сверхпроводящего квантового процессора.

[00927] Сверхпроводящий квантовый процессор может содержать ряд кубитов и связанных смещающих устройств, например, два, три или более сверхпроводящих кубитов. Соответственно, хотя в различных вариантах реализации компьютерный процессор может быть выполнен в виде нетрадиционного сверхпроводящего процессора, в других вариантах реализации компьютерный процессор может быть сконфигурирован как сверхпроводящий процессор. Например, в некоторых вариантах реализации нетрадиционные сверхпроводящий процессор может быть выполнен так, чтобы он не фокусировался на квантовых эффектах, таких как суперпозиция, запутанность и/или квантовое туннелирование, а, скорее, мог действовать за счет подчеркивания различных принципов, таких как принципы, охватывающие работу классических компьютерных процессоров. В других вариантах реализации компьютерный процессор может быть выполнен в виде традиционного сверхпроводящего процессора, например такого, который может адаптироваться к процессу за счет различных квантовых эффектов,

таки как суперпозиция, запутанность и/или квантовое туннелирование.

[00928] Соответственно, в различных случаях реализация таких сверхпроводящих процессоров может иметь определенные преимущества. В частности, ввиду их естественных физических свойств сверхпроводящие процессоры, как правило, могут быть выполнены с возможностью более высоких скоростей переключения и более короткого времени вычисления, чем несверхпроводящие процессоры, и поэтому, возможно, более практично решать определенные проблемы на сверхпроводящих процессорах. Дальнейшие подробности и варианты реализации примеров квантовых процессоров, которые могут быть использованы совместно с представленными устройствами, системами и способами их использования, описаны в USSN: 11/317,838; 12/013,192; 12/575,345; 12/266,378; 13/678,266 и 14/255,561; а также в различных разделенных заявках, продолжающихся заявках и/или в заявках в частичное продолжение их, включая патенты США №№7,533,068; 7,969,805; 9,026,574; 9,355,365; 9,405,876; и все их зарубежные аналоги, которые полностью включены в настоящий документ путем ссылки.

[00929] Кроме того, помимо вышеупомянутого предложены также квантовые устройства, системы и способы их применения для решения сложных вычислительных задач. Например, квантовые устройства и системы, описанные в настоящем документе, могут быть использованы для управления квантовым состоянием одного или более информационных устройств и/или систем когерентным образом для выполнения одного или более этапов конвейера бионформационной и/или геномной обработки, например, для выполнения одной или более операций при обработке изображений, определения оснований, картирования, выравнивания, сортировки, определения вариантов и/или других геномных и/или биоинформационных конвейеров. В конкретных вариантах реализации одна или более операций могут представлять собой выполнение операций Барроуза-Уилера, Смита-Ватермана и/или HMM.

[00930] В частности, решение сложных геномных и/или биоинформационных вычислительных проблем с помощью квантового вычислительного устройства может включать в себя формирование одного или более кубитов и использование их для формирования квантовых логических схем, представляющих вычислительную проблему, кодирование представления логической схемы как проблемы дискретной оптимизации и решение проблемы дискретной оптимизации с помощью квантового процессора. Представление может быть арифметической и/или геометрической проблемой для решения с помощью схем сложения, вычитания, умножения и/или деления. Проблема дискретной оптимизации может содержать набор миниатюрных проблем оптимизации, где каждая миниатюрная проблема оптимизации кодирует соответствующий логический вентиль из представления логической схемы. Например, в математической схеме можно использовать двоичное представление факторов, и эти двоичные представления можно разложить для сокращения общего количества переменных, требуемых для представления математической схемы. Соответственно, согласно идеям, изложенным в настоящем документе, компьютерный процессор может принимать форму цифрового и/или аналогового процессора, например, квантового процессора, такого как сверхпроводящий квантовый процессор. Сверхпроводящий квантовый процессор может содержать ряд кубитов и связанные локальные смещающие устройства, например, два или более сверхпроводящих кубита, которые могут быть сформированы в одно или более представлений квантовой логической схемы.

[00931] Более конкретно, в различных вариантах реализации может быть предусмотрена сверхпроводящая интегральная микросхема. А именно, в конкретных

вариантах реализации такая сверхпроводящая интегральная схема может содержать первый путь сверхпроводящего тока, который размещен в металле, например, в первом металлическом слое. Может быть также включен диэлектрический, например, первый диэлектрический, слой, например, где по меньшей мере часть диэлектрического слоя связана с первым металлическим слоем и/или выполнена на нем. Второй путь сверхпроводящего тока тоже может быть включен и расположен во втором металлическом слое, например, в металлическом слое, выполненном на первом диэлектрическом слое или иным образом связанным с ним. В таком варианте реализации по меньшей мере часть второго пути сверхпроводящего тока может перекрывать по меньшей мере часть первого пути сверхпроводящего тока. Аналогичным образом может быть также включен второй диэлектрический слой, например, где по меньшей мере часть второго диэлектрического слоя связана со вторым металлическим слоем и/или выполнена на нем. Кроме того, третий путь сверхпроводящего тока может быть включен и расположен в третьем металлическом слое, который может быть связан со вторым диэлектрическим слоем или выполнен на нем, например, кода по меньшей мере часть третьего пути сверхпроводящего тока может перекрывать по меньшей мере часть одного или обоих из первого и второго путей сверхпроводящего тока. Также могут быть включены и сконфигурированы соответствующим образом один или более дополнительных слоев, диэлектрических слое и/или путей тока

[00932] Кроме того, между первым путем сверхпроводящего тока и третьим путем сверхпроводящего тока может быть расположено первое сверхпроводящее соединение, например, где первое сверхпроводящее соединение проходит сквозь оба диэлектрических слоя, первый и второй. Между первым путем сверхпроводящего тока и третьим путем сверхпроводящего тока может быть расположено второе сверхпроводящее соединение, например, где второе сверхпроводящее соединение может проходить сквозь оба диэлектрических слоя, первый и второй. Кроме того, по меньшей мере часть второго пути сверхпроводящего тока может быть окружена наружным путем сверхпроводящего тока, который может быть сформирован по меньшей мере частью одного или более из первого пути сверхпроводящего тока по меньшей мере части второго пути сверхпроводящего тока и/или первого и второго сверхпроводящих соединений. Соответственно, в таких случаях второй путь сверхпроводящего тока может быть выполнен с возможностью соединения, например, индуктивного соединения, сигнала с наружным путем сверхпроводящего тока.

[00933] В некоторых вариантах реализации взаимная индуктивность между вторым путем сверхпроводящего тока и наружным путем сверхпроводящего тока может быть почти линейно пропорциональна толщине первого диэлектрического слоя и толщине второго диэлектрического слоя. Каждое из второго и первого сверхпроводящих соединений могут также могут также включать в себя по меньшей мере одно сверхпроводящее сквозное межсоединение. Кроме того, в различных вариантах реализации второй путь сверхпроводящего тока может быть частью линии входного сигнала, а один или оба из первого и третьего путей сверхпроводящего тока могут быть соединены с программируемым сверхпроводящим устройством. В других вариантах реализации второй путь сверхпроводящего тока может быть частью программируемого сверхпроводящего устройства, а первый и третий пути сверхпроводящего тока оба могут соединены с линией выходного сигнала. В конкретных вариантах реализации программируемое сверхпроводящее устройство может быть сверхпроводящим кубитом, который может быть также соединен, например, квантовым образом, с одним или более другими кубитами с образованием квантовой схемы, такой

как квантовое обрабатывающее устройство.

[00934] Соответственно, в настоящем документе предложены устройства, системы и способы решения вычислительных проблем, особенно проблем, связанных с разрешением узких мест в геномике и/или биоинформатике, которые описаны выше в настоящем документе. В различных вариантах реализации эти устройства, системы и способы вводят метод, посредством которого представление логической схемы вычислительной проблемы может быть решено непосредственно и/или может быть закодировано как проблема дискретной оптимизации, и затем проблема дискретной оптимизации может быть решена с помощью компьютерного процессора, такого как квантовый процессор. Например, в конкретных вариантах реализации решение таких проблем дискретной оптимизации может включать в себя исполнение логической схемы для решения исходной вычислительной проблемы.

[00935] Следовательно, устройства, системы и способы, описанные в настоящем документе, могут быть реализованы с использованием любой формы компьютерного процессора, в том числе, например, традиционных логических схем и/или представлений логических схем, например, выполненных с возможностью использования в качестве квантового процессора и/или сверхпроводящего процессора. В частности, различные этапы выполнения обработки изображений, определения оснований, картирования, выравнивания и/или определения вариантов биоинформационного конвейера могут быть закодированы как проблемы дискретной оптимизации, и поэтому могут особенно подходить для решения с помощью квантовых процессоров, описанных в настоящем документе. В других случаях такие вычисления могут быть разрешены более обычным путем с помощью компьютерного процессора, который приспосабливает квантовые эффекты для достижения таких вычислений; и/или в других случаях такие вычисления могут быть выполнены с помощью специализированной интегрированной схемы, такой как FPGA, ASIC или структурированная ASIC, как подробно описано в настоящем документе. В некоторых вариантах реализации проблема дискретной оптимизации трактуется как проблема конфигурирования логических схем, кубитов и/или ответвителей в квантовом процессоре. В некоторых вариантах реализации квантовый процессор может быть специально выполнен с возможностью решения таких проблем дискретной оптимизации.

[00936] По всему данному описанию изобретения и в прилагаемой формуле изобретения часто упоминается «представление логической схемы», например вычислительной проблемы. В зависимости от контекста логическая схема может содержать набор логических входов, набор логических выходов и набор логических вентилей (например, вентили «И-НЕ», вентили «исключающее ИЛИ» и т.п.), которые преобразуют логические входные сигнал в логические выходные сигналы посредством набора промежуточных логических входов и промежуточных логических выходов. Полная логическая схема может содержать представление входов в вычислительную проблему, представление выходов из вычислительной проблемы и представление последовательности промежуточных этапов между входами и выходами.

[00937] Таким образом, при использовании в различных целях представленных устройств, систем и способов вычислительная проблема может быть определена своими входами, своими выходами и промежуточными этапами, которые преобразуют входы в выхода, и «представление логической схемы» может включать в себя все эти элементы. Специалистам в данной области будет понятно, что кодирование «представления логической схемы» вычислительной проблемы как проблемы дискретной оптимизации и последующим отображением проблемы дискретной оптимизации на квантовый

процессор может привести к любому количеству участвующих слоев и любому количеству кубитов на слой. Кроме того, такое отображение может реализовывать какую-либо схему межкубитного связывания для обеспечения какой-либо схемы межслойного связывания (например, связывание между кубитами различных слоев) и внутрислойного связывания (например, связывания между кубитами в пределах конкретного слоя).

[00938] Соответственно, как было указано, в некоторых вариантах реализации структура логической схемы может быть разделена на слои. Например, логические входы могут быть представлены первым слоем, каждая последующая логическая (или арифметическая) операция может представлять соответствующий дополнительный слой, а логические выходы могут представлять еще один слой. Как описано ранее, логическая операция может быть исполнена одним логическим вентилем или комбинацией логических вентилей в зависимости от исполняемой конкретной логической операции. Таким образом, «слой» в логической схеме может включать в себя один логический вентиль или комбинацию логических вентилей в зависимости от реализуемой конкретной логической схемы.

[00939] Следовательно, в различных вариантах реализации, таких где структура логической схемы разбита на слои (например, где логические входы представляют первый слой, каждая последующая логическая операция представляет соответствующий дополнительный слой, а логические выходы представляют еще один слой), каждый слой может быть осуществлен с помощью соответствующего набора кубитов в квантовом и/или сверхпроводящем процессоре. Например, в одном варианте реализации квантового процессора один или более, например, каждый ряд, кубитов могут быть запрограммированы для представления соответствующего слоя квантовой логической схемы. То есть, конкретные кубиты могут быть запрограммированы для представления входов в логическую схему, другие кубиты могут быть запрограммированы для представления первой логической операции (исполняемой либо одним, либо множеством логических вентилей), еще одни кубиты могут быть запрограммированы для представления второй логической операции (аналогичным образом исполняемой либо одним, либо множеством логических вентилей), а следующие кубиты могут быть запрограммированы для представления выходов логической схемы.

[00940] Кроме того, когда различные наборы кубитов представляют различные слои проблемы, может оказаться целесообразным обеспечение независимого динамического управления каждым соответствующим набором. Кроме того, в различных вариантах реализации различные последовательные логические схемы могут быть отображены в квантовый процессор, а соответствующие кубиты отображены так, чтобы облегчить функциональные взаимодействия для квантовой обработки таким образом, который подходит для обеспечения независимого управления ими. Учитывая вышесказанное, специалистам в данной области будет понятно, как подобную целевую функцию можно определить для любого логического вентиля. Таким образом, в некоторых вариантах реализации проблема, представляющая логическую схему, может по существу содержать множество миниатюрных проблем оптимизации, где каждый вентиль в логической схеме соответствует конкретной миниатюрной проблеме оптимизации.

[00941] Следовательно, с помощью систем и способов, которые известны данной области техники, можно сформировать пример представлений логической схемы. В одном примере представление логической схемы вычислительной проблемы, например, проблемы геномики и/или биоинформатики, может быть сформировано и/или закодировано с помощью классического цифрового компьютерного процессора, и/или

квантового, и/или сверхпроводящего процессора, как описано в настоящем документе. Соответственно, представление логической схемы вычислительной проблемы может быть сохранено на по меньшей мере одном читаемым компьютером или процессором носителе информации, таком как некратковременный машиночитаемый носитель информации или память (например, энергозависимая или энергонезависимая). Следовательно, как отмечалось выше, представление логической схемы вычислительной проблемы может быть закодировано как проблема дискретной оптимизации или набор целей оптимизации, и в различных вариантах реализации, таких где для решения проблемы сконфигурирована модель обработки классическим цифровым компьютером, система может быть выполнена таким образом, чтобы битовые сроки, которые удовлетворяют логической схеме, имели нулевую энергию, а все другие битовые строки имели энергию больше нуля, и тогда проблема дискретной оптимизации может быть решена таким образом, чтобы решить исходную вычислительную проблему.

[00942] Кроме того, в других вариантах реализации проблема дискретной оптимизации может быть решена с помощью компьютерного процессора, такого как квантовый процессор. В таком случае решение проблемы дискретной оптимизации может включать в себя, например, развертывание квантового процессора до конфигурации, которая минимизирует энергию системы для установления битовой строки, которая удовлетворяет целям оптимизации. Соответственно, в некоторых вариантах реализации акт решения проблемы дискретной оптимизации может включать в себя три действия. Во-первых, проблему дискретной оптимизации можно отобразить на компьютерный процессор. В некоторых вариантах реализации компьютерный процессор может представлять собой квантовый и/или сверхпроводящий процессор, а отображение проблемы дискретной оптимизации на компьютерный процессор может включать в себя программирование элементов (например, кубитов и ответвителей) квантового и/или сверхпроводящего процессора. Отображение проблемы дискретной оптимизации на компьютерный процессор может включать в себя проблему дискретной оптимизации в по меньшей мере одном читаемым компьютером или процессором носителе информации, таком как некратковременный машиночитаемый носитель информации или память (например, энергозависимая или энергонезависимая).

[00943] Соответственно, ввиду вышеизложенного, в различных случаях предусмотрены устройство, система и способ для исполнения конвейера анализа последовательности, например, на геномном материале. Например, геномный материал может включать в себя множество ридов геномных данных, например, в файле изображения BCL, FASTQ и т.п. В различных вариантах реализации устройство и/или система могут быть использованы для исполнения анализа последовательности на геномных данных, таких как риды геномных данных, например, с помощью индекса одной или более генетических референсных последовательностей, например, хранящихся в памяти, например, когда каждый рид геномных данных и каждая референсная последовательность представляют нуклеотидные последовательности.

[00944] В частности, в различных вариантах реализации устройство может быть квантовым вычислительным устройством, например, сформированным из набора квантовых логических схем, например, жестко смонтированных квантовых логических схем, таких как логические схемы, соединенные друг с другом. В различных случаях квантовые логические схемы могут быть взаимно соединены друг с другом с помощью одного или более сверхпроводящих соединений. Кроме того, одно или более сверхпроводящих соединений могут включать в себя интерфейс памяти, например для получения доступа к памяти. Вместе логические схемы и межсоединения могут быть

выполнены с возможностью обработки информации, представляющей квантовое состояние, которое само представлено как набор из одного или более кубитов. Более конкретно, набор жестко смонтированных квантовых логических схем может быть выполнен в виде набора движков обработки, например, где каждой движок обработки может быть сформирован подмножеством жестко смонтированных квантовых логических схем и может быть выполнен с возможностью осуществления одного или более этапов в конвейере анализа последовательностей на ридах геномных данных.

[00945] Например, набор движков обработки может быть выполнен с возможностью включения в себя модуля обработки изображений, определения оснований, картирования, выравнивания, сортировки, определения вариантов и/или другого модуля геномики и/или биоинформатики. Например, в различных вариантах реализации может быть включен модуль картирования, например, в первой жестко смонтированной конфигурации. Кроме того, в других вариантах реализации может быть включен модуль выравнивания, например, во второй жестко смонтированной конфигурации. Далее, может быть включен модуль сортировки, например, в третьей жестко смонтированной конфигурации. И в дополнительных вариантах реализации может быть включен модуль определения вариантов, например, во четвертой жестко смонтированной конфигурации. Более того, в различных вариантах реализации может быть включен модуль обработки изображений и/или определения вариантов в дополнительных жестко смонтированных конфигурациях, например, когда одна или более из этих жестко смонтированных конфигураций могут содержать жестко смонтированные квантовые логические схемы, причем этот модуль может быть организован в виде набора движков обработки.

[00946] А именно, в конкретных случаях квантовое вычислительное устройство и/или система могут содержать модуль картирования, где модуль картирования включает в себя множество квантовых логических схем, которые организованы как набор движков обработки, один или более из которых выполнены с возможностью осуществления одного или более этапов процедуры картирования. Например, один или более квантовых движков обработки могут быть выполнены с возможностью приема рида геномных данных, например, посредством одного или более из множества сверхпроводящих соединений. Далее, один или более квантовых движков обработки могут быть выполнены с возможностью выделения части рида для формирования затравки, например, когда затравка может быть представлять подмножество последовательности нуклеотидов, представленной ридом. Кроме того, один или более квантовых движков обработки могут быть выполнены с возможностью вычисления первого адреса в индексе на основе затравки и обращения по адресу в индексе к памяти, чтобы принять записи с адреса, например, когда запись представляет информацию о позиции в генетической референсной последовательности. Кроме того, один или более квантовых движков обработки могут быть выполнены с возможностью определения, например, на основе записи, одной или более совпадающих позиций рида с генетической референсной последовательностью и вывода по меньшей мере одной из совпадающих позиций в память посредством интерфейса памяти.

[00947] Более того, модуль картирования может включать в себя набор квантовых логических схем, которые организованы в виде набора движков обработки, выполненных с возможностью вычисления второго адреса в индексе, например, на основе записи и второго подмножества последовательности нуклеотидов, которые не содержатся в первом подмножестве последовательности нуклеотидов. Затем движки обработки могут обратиться по второму адресу в индексе в память, чтобы принять вторую запись со второго адреса, например, когда вторая запись, или последующая

запись, содержит информацию о позиции в генетической референсной последовательности. Движок обработки может быть также выполнен с возможностью определения на основе информации о позиции одной или более совпадающих позиций рида с генетической референсной последовательностью.

[00948] Кроме того, в различных случаях квантовое вычислительное устройство и/или система могут содержать модуль выравнивания, где модуль выравнивания включает в себя множество квантовых логических схем, которые организованы как набор движков обработки, один или более из которых выполнены с возможностью осуществления одного или более этапов процедуры выравнивания. Например, один или более квантовых движков обработки могут быть выполнены с возможностью приема множества картированных позиций из памяти и обращения к памяти с целью извлечения сегмента генетической референсной последовательности, соответствующего каждой картированной позиции. Один или более движков обработки, сформированных в виде модуля выравнивания, могут также быть выполнены с возможностью вычисления выравнивания рида для каждого извлеченного сегмента генетической референсной последовательности с целью формирования оценки каждого выравнивания. Далее, после того, как одна или более оценок сформированы, может быть выбрано по меньшей мере один рид с лучшей оценкой выравнивания. В конкретных случаях квантовое вычислительное устройство может содержать набор квантовых логических схем, организованных в виде набора движков обработки, которые выполнены с возможностью осуществления выравнивания с гэпами или без гэпов, например выравнивание Смита-Ватермана.

[00949] Также, в определенных случаях квантовое вычислительное устройство и/или система могут содержать модуль определения вариантов, где модуль определения вариантов включает в себя множество квантовых логических схем, которые организованы как набор движков обработки, один или более из которых выполнены с возможностью осуществления одного или более этапов процедуры определения вариантов. Например, квантовый вычислительный модуль определения вариантов может содержать набор квантовых логических схем, которые выполнены с возможностью исполнения анализа на множестве ридов геномных данных, например, с использованием одного или более гаплотипов-кандидатов, например, хранящихся в памяти, где каждый рид геномных данных и каждый гаплотип-кандидат представляют последовательность нуклеотидов.

[00950] В частности, набор квантовых логических схем может быть сформирован как один или более квантовых движков обработки, которые выполнены с возможностью приема одного или более ридов геномных данных и формирования и/или приема одного или более гаплотипов-кандидатов, например, из памяти, например, посредством одного или более из множества сверхпроводящих соединений. Кроме того, один или более из квантовых движков обработки может быть выполнен с возможностью приема одного или более ридов геномных данных и одного или более гаплотипов-кандидатов из памяти, а также сравнения нуклеотидов в каждом из одного или более ридов с нуклеотидами одного или более гаплотипов-кандидатов с целью определения вероятности представления каждым гаплотипом-кандидатом правильного определения вариантов. Кроме того, один или более квантовых движков обработки могут быть выполнены с возможностью формирования выходных данных на основе определенной вероятности.

[00951] Кроме того, в различных случаях набор квантовых логических схем может быть сформирован как один или более квантовых движков обработки, которые

выполнены с возможностью определения вероятности наблюдения каждого рида из множества ридов на основе по меньшей мере одного гаплотипа-кандидата, являющегося действительной последовательностью нуклеотидов, например организма-источник множества ридов. В частных случаях, что касается определения вероятности, один или более квантовых движков обработки могут быть выполнены для исполнения скрытой марковской модели. Более конкретно, в дополнительных вариантах реализации один или более квантовых движков обработки могут быть выполнены с возможностью объединения множества ридов в одну или более непрерывных нуклеотидных последовательностей и/или формирования одного или более гаплотипов-кандидатов из одной или более непрерывных нуклеотидных последовательностей. Например, в различных вариантах реализации объединение множества ридов включает в себя один или более квантовых движков обработки, строящих граф де Брейна.

[00952] Соответственно, ввиду вышесказанного, предложена система для выполнения различных вычислений при решении проблем, связанных с геномной и/или биоинформационной обработкой. Например, система может включать в себя один или более автоматизированных секвенаторов в месте эксплуатации, например, СНП, и/или сервер обработки, причем один из них или оба могут содержать одно или более ЦПУ/ГПУ и/или другую интегральную схему, в частности, такую как FPGA, ASIC и/или структурированная ASIC, которые, выполнены с возможностью осуществления одного или более этапов в конвейере анализа последовательности, как описано в настоящем документе. В частности, секвенатор нового поколения может быть выполнен с возможностью секвенирования множества последовательностей нуклеиновых кислот для формирования одного или более файлов изображений BCL и/или FASTQ, представляющих секвенированные последовательности нуклеиновых кислот, причем последовательности нуклеиновых кислот могут быть последовательностями ДНК и/или РНК. Эти файлы последовательностей могут быть обработаны самим секвенатором или связанным серверным устройством, например, когда секвенатор и/или связанный сервер содержат встроенную схему, такую как FPGA или ASIC, выполненную, как описано в настоящем документе, с возможностью осуществления одного или более этапов в конвейере вторичного анализа последовательности.

[00953] Однако в различных случаях, таких когда автоматизированный секвенатор и/или связанный сервер не выполнены с возможностью осуществления вторичного анализа последовательности на данных, сформированных секвенатором, сформированные данные могут быть переданы на удаленный сервер, который выполнен с возможностью осуществления вторичного и/или третичного анализа на данных, например посредством облачного интерфейса. В таком случае доступный из облака сервер может быть выполнен с возможностью приема сформированных данных последовательности, таких как изображение в виде файла BCL и/или FASTQ, и может также быть выполнен с возможностью осуществления анализа первичной, например обработки изображений, и/или вторичной, и/или третичной обработки, например, конвейера анализа последовательности, на принятых данных. Например, доступный из облака сервер может быть одним или более серверами, содержащими ЦПУ и/или ГПУ, из которых одно или оба могут быть связаны с интегральной схемой, такой как FPGA или ASIC, как описано в настоящем документе. В частности, в определенных случаях доступный из облака сервер может быть квантовым вычислительным сервером, как описано в настоящем документе.

[00954] А именно, доступный из облака сервер может быть выполнен с возможностью осуществления первичного, вторичного и/или третичного геномного и/или

биоинформационного анализа на принятых данных, причем анализ может включать в себя выполнение одного или более этапов в одном или более из протоколов обработки изображений, определения оснований, картирования, выравнивания, сортировки и/или определения вариантов. В определенных случаях некоторые этапы могут быть выполнены одной платформой обработки, такой как ЦПУ или ГПУ, а другие могут быть выполнены другой платформой обработки, такой как связанная, например, жестко связанная, интегральная схема, такая как FPGA или ASIC, которая специально выполнена с возможностью осуществления различных этапов в конвейере анализа последовательности. В таких случаях, когда данные и результаты анализа подлежат передаче с одной платформы на другую, система и ее компоненты могут быть выполнены с возможностью сжатия данных перед передачей и распаковкой данных после передачи, и поэтому такие компоненты системы могут быть выполнены с возможностью формирования одного или более файлов SAM, BAM или CRAM, например для передачи. Кроме того, в различных вариантах реализации доступный из облака сервер может быть квантовой вычислительной платформой, которая, как описано в настоящем документе, выполнена с возможностью осуществления одного или более этапов в конвейере анализа последовательности, как описано в настоящем документе, и может включать в себя выполнение одного или более этапов вторичной и/или третичной обработки в соответствии с одним или более способами, описанными в настоящем документе.

[00955] Кроме того, что касается квантового вычисления, подробные сведения и варианты реализации примеров квантовых процессоров и способов их применения, которые могут быть использованы вместе с представленным устройствами, системами и способами, описаны в патентах США №№7,135,701; 7,533,068; 7,969,805; 8,560,282; 8,700,689; 8,738,105; 9,026,574; 9,355,365; 9,405,876; а также в различных их аналогах, которые полностью включены в настоящий документ путем ссылки.

[00956] Кроме того, что касается модуля искусственного интеллекта, описанного выше, в соответствии с одним аспектом предложено доступный из облака модуль искусственного интеллекта, который выполнен с возможностью функционального соединения с возможностью обмена данными с одним или более другими компонентами BioIT-конвейера, описанного в настоящем документе. Например, модуль ИИ может тесно сотрудничать с WMS для эффективного руководства и/или управления различными процессами системы, описанной в настоящем руководстве. Соответственно, в различных вариантах реализации предусмотрен модуль ИИ, который выполнен с возможностью действия в качестве интерфейса между геномной и клинической сферами деятельности.

[00957] Например, в различных случаях BioIT-система может быть выполнена с возможностью приема клинических данных. В таком случае система диспетчера рабочих потоков может быть выполнена с возможностью анализа клинических данных и других таких данных и реализации одной или более систем детерминированных правил для получения данных результатов в соответствии с ее анализом клинических данных. Например, в определенных вариантах реализации различные базы данных системы могут быть выполнены таким образом, чтобы они имели реляционную архитектуру.

[00958] Эти конструкции могут быть представлены одной или более табличных структур. Тогда WMS может использовать ряд таблиц, например, для корреляции итеративным образом. Например, в различных моделях использования первая корреляция может быть выполнена в отношении имени субъекта с медицинским состоянием. После этого другая таблица может быть использована для корреляции медицинского состояния субъекта с его консервативным лечением. Аналогичным

образом еще одна таблица может быть использована для корреляции прогресса консервативного лечения со смягчением симптомов и/или самой болезни. Для корреляции таблиц может использоваться ключ, доступ к которому предоставляется в ответ на заданный вопрос или команду. Ключ может быть каким-либо общим идентификатором, таким как имя, номер например, номер социального страхования, индивидуальный номер налогоплательщика, код сотрудника, номер телефона и т.п., по которому можно получить доступ к одной или более таблицам, выполнить корреляцию и/или получить ответ на вопрос. Соответственно, без ключа будет намного труднее строить корреляции между информацией в одной таблице с информацией в другой таблице.

[00959] Однако в других случаях модуль ИИ может быть выполнен с возможностью обеспечения более всеобъемлющего анализа на сформированных и/или предоставленных данных. Например, модуль ИИ может быть выполнен с возможностью реализации одного или более протоколов машинного обучения на данных системы, которые разработаны для обучения модуля ИИ созданию корреляций между геномными данными, например, сформированными системой, и накопленными клиническими данными одного или более субъектов, например, в связи с вводом EMR и других клинически уместных данных в систему.

[00960] В частности, модуль ИИ может включать в себя программирование, направленное на убыстрение тренировки системы, например, мгновенное, на распознавание того, каким образом были получены выходные данные на основе типа или характеристик принятых входных данных. Поэтому система выполнена с возможностью обучения на основе входных данных, которые она принимает, и результатах ее выходных данных, чтобы быть в состоянии быстрее и точнее устанавливать корреляции на основе принимаемых входных данных. Как правило, входные данные могут быть двух общих типов. В первом случае данные могут быть типа, в котором выходные данные, например, ответ, известны. Данные этого типа можно вводить в систему и использовать в целях тренировки. Данные второго типа могут быть данными, где ответ не известен, и, следовательно, должен быть определен; эти данные, вероятно, будут геномными данными, на которых нужно выполнить анализ, или клинические данные, для которых нужно определить клинически уместные результаты. В частности, эти способы можно использовать для улучшения возможностей обучения модулей ИИ на входных данных первого типа, чтобы лучше прогнозировать результат для входных данных второго типа. А именно, на основании прошлых подтверждающих данных модуль ИИ может быть выполнен с возможностью обучения прогнозированию результатов, исходя из ранее наблюдаемых данных.

[00961] Точнее говоря, в настоящем документе представлена клиническая геномная платформа, которая выполнена с возможностью коррелирования клинических исходов заболеваний с геномними данными. В таком случае в систему можно вводить клинические профили субъектов и оценивать вместе с их определенным геномным профилем. В частности, модуль ИИ выполнен с возможностью определения на основе объединения эти двух наборов данных различных взаимозависимостей между ними. Соответственно, на первом этапе можно построить графовую базу данных или граф знаний. Например, в данном случае граф знаний может содержать элементы трех типов, в число которых входят субъект, предикат и объект, и эти элементы образуют узлы, и между узлами нужно определить взаимосвязь. Любую конкретную точку данных можно выбрать в качестве узла, и узлы могут меняться на основе выполняемых запросов. Существуют несколько различных типов взаимосвязей, которые можно определить.

Например, взаимосвязи можно определить на основе их последствий, например, на основе их влияния; или их можно определить на основе умозаключений, например, взаимосвязи, которые неизвестны, но поддаются определению.

[00962] Соответственно, что касается построения графа знаний, любая конкретная точка данных может образовывать узел. Например, на одной стороне графа узлом может быть болезненное состояние, а на другой стороне графа узлом может быть генотип, например, последовательность вариаций. Между этими двумя узлами может быть третий узел, например, ряд узлов, таких как один или более симптомов, один или более лекарственных препаратов, одна или более аллергий, одно или более других состояний или фенотипических признаков, например, кровяное давление, холестерин и т.д. Кроме того, между этими узлами существуют взаимосвязи, которые могут быть определены.

[00963] В частности, при построении графа знаний в систему вводят клинические данные, например, из учреждения, ведущего медицинские записи, например, электронные медицинские записи, семейный анамнез медицинских состояний и т.д., которые могут быть зашифрованы и безопасно переданы с помощью электронных средств. Аналогичным образом геномные данные субъекта могут быть секвенированы и сформированы в соответствии с этапами вторичной обработки, описанными в настоящем документе. Кроме того, после того, как эти два узла созданы, в систему можно ввести один или более узлов третьего типа, на наличии которых можно определить взаимосвязь (-и) между двумя исходными узлами.

[00964] Например, в одном примере первый узел может быть представлен медицинскими записями человека или популяции людей, а второй узел может быть представлен характеристикой болезни. В таком случае в систему можно ввести и сформировать на графе один или более третьих узлов, где, например, третий узел может быть лекарственным препаратом; физическим, биологическим, ментальным состоянием и/или характеристикой; аллергией; географической областью; режимом питания, продуктом питания и/или ингредиентом; условием окружающей среды; географическим условием; линиями электропередачи, вышками сотовой связи и т.п. Затем можно определить ряд взаимосвязей путем анализа различных точек соединения между этими тремя элементами. В частности, в конкретном случае один узел может представлять пациента, страдающего от болезненного состояния, вторым узлом могут быть геномные данные пациента, в числе третьих узлов могут быть геномные вариации пациента, например, мутации субъекта, хромосома за хромосомой, его лекарственные препараты, физиологические состояния и т.п. Аналогичным образом этот процесс может быть повторен для множества субъектов, имеющих тот же самый диагноз или состояние. Следовательно, подобным образом можно определить корреляцию между клинической и геномной сферами активности.

[00965] Соответственно, этапом в построении клинического геномного графа является определение узлов привязки, представляющих собой два граничных элемента, между которыми все определяют и изучают все различные общности. Следовательно, следующим этапом является определение всех возможных известных соответствий между двумя узлами привязки, которые могут быть представлены в графе как третий узел. Эти известные соответствия могут быть построены путем детализации последствий, вызванных одним или другим узлом и/или их характеристиками. Это могут быть известные и/или наблюдаемые взаимосвязи между узлами. На основе этих известных взаимосвязей можно изучить второй тип взаимосвязи и/или определить, какие взаимосвязи могут быть построены на умозаключениях. Далее, чтобы лучше определять

причинные и/или прогнозируемые результаты, всевозможным разным взаимосвязям можно присвоить веса, например, на основе степени определенности, количества общностей, количества экземпляров, совместно использующих узел, количества общих взаимосвязей и т.п.

[00966] Таким образом, построение и реализация динамического графа знаний лежат в основе клинической геномной платформы обработки. Как было указано, различные платформы обработки глобальной системы можно связать вместе, например, для беспрепятственной передачи данных между ее различными компонентами. Например, как было указано, конвейеры картирования, выравнивания и/или определения вариантов могут быть выполнены с возможностью передачи данных, например, данных результатов, в модуль искусственного интеллекта. В частности, модуль ИИ может быть выполнен с возможностью приема входных данных из одного или более компонентов платформы вторичной обработки и/или одного или более других компонентов системы. Более конкретно, модуль ИИ выполнен с возможностью приема данных картирования, выравнивания и/или определения вариантов из движков обработки сопоставителя, выравнивателя и/или определения вариантов и взятия в оборот этих данных и использования их для формирования одного или более узлов на графе знаний. Кроме того, модуль ИИ может быть выполнен с возможностью приема входных данных из одного или более источников, например, из медицинского учреждения, от поставщика услуг здравоохранения, из исследовательской лаборатории, средства хранения записей и т.п., например, когда записи содержат данные, относящиеся к физическому, ментальному и/или эмоциональному благополучию одного или более субъектов, и взятия в оборот этих данных и использования их для формирования одного или более узлов на графе знаний.

[00967] Кроме того, после того, как архитектура графа знаний построена, она может непрерывно обновляться и расти за счет добавления все больше и больше соответствующих данных в структуру знаний, построения все больше и больше потенциальных узлов и/или взаимосвязей. В таком случае граничные узлы могут быть любой комбинацией узлов, например, в определенных случаях могут выбираться пользователем. Например, в различных вариантах реализации система может быть выполнена с возможностью доступа к ней третьей стороной. В таком случае пользователь может получать доступ к модулю ИИ, например, посредством соответствующим образом сконфигурированного пользовательского интерфейса, выгружать относящуюся к делу информацию в систему и/или определять соответствующие узлы, которыми нужно ограничить запрос, например, щелчком или перетаскиванием их, и может формулировать вопрос для получения ответа от модуля ИИ. Соответственно, пользователь может просматривать и/или выбирать граничные узлы и затем предоставлять системе формирование соответствующей карты знаний с использованием выбранных узлов и определение взаимосвязей между этими узлами, а на основании этих взаимосвязей могут быть заданы вопросы и получены ответы, по меньшей мере умозрительные, например, от системы ИИ.

[00968] Например, в одной модели использования пользователь может быть врачом, которому требуется узнать, как определенная дозировка лекарственного средства влияет на пациента применительно к данной болезни. Соответственно, врач может выгрузить EMR пациента, болезненное состояние и дозировку лекарственного средства, и с помощью этих данных модуль ИИ может сформировать подходящий граф знаний (и/или добавить к уже существующему графу знаний), а на основании этого графа знаний можно выбрать граничные узлы и определить взаимосвязи. Кроме того, в

различных случаях пользователь может выгрузить генетические данные пациента, которые могут быть подвергнуты вторичной обработке, а ее результаты, например, картированные, выровненные и/или подвергнутые определению вариантов результирующие данные, могут быть выгружены в модуль ИИ. В таком случае данные о болезни, и/или EMR, и/или семейного анамнеза могут быть коррелированы с геномными данными, на основании чего могут быть определены различные взаимосвязи, оценены логические выводы и сделаны прогнозы.

[00969] В частности, в систему можно ввести файл VCF субъекта, например, можно выгрузить все установленные хромосомные свойства, например, в виде группы узлов, и эти узлы могут быть использованы для определения различных взаимосвязей, имеющих отношение к субъекту, например, путем ввода запроса в систему и предоставления ей формирования соответствующих связей, из которых могут быть логически выведены ответы. Точнее говоря, в систему можно выгрузить одну или более фенотипических характеристик субъекта, например, онтологию фенотипа человека, чтобы сформировать еще одну группу узлов. Например, когда в систему вводят геномные и/или медицинские истории двух людей, модулем ИИ могут быть определены любые взаимосвязи между ними, например в отношении общих генотипов, фенотипов, условий, окружающей среды, географических положений, аллергий, этническо-культурного происхождения, лекарственных препаратов и т.п.

[00970] Кроме того, могут быть определены взаимосвязи между двумя или более характеристиками субъекта. Например, система может определить взаимосвязь между систолическим и диастолическим кровяным давление субъекта. В частности, в систему можно ввести ряд прошлых показаний систолического и диастолического давления, на основании чего машиннообучаемая платформа системы может проанализировать эти показания и/или определить одну или более взаимосвязей между этими двумя давлениями, так что при вводе в систему данного систолического давления может быть выдано предполагаемое диастолическое давление с учетом предсказуемых соотношений между ними. Необходимо отметить, что хотя предыдущий пример приведен для кровяного давления одного субъекта, то же самое справедливо в отношении любых данных узлов, которые математически связаны друг с другом, например, в отношении множества субъектов и/или множества условий.

[00971] Кроме того, хотя в некоторых случаях взаимосвязи могут быть сконфигурированы в линейном массиве, например, с образованием нейронной сети информации, в различных других случаях взаимосвязи могут быть сформированы в виде множества стадий, таких как в протоколе глубокого обучения. Например, система ИИ может быть выполнена с возможностью послойной или многостадийной обработки информации, например, в целях глубокого обучения. Соответственно, система может быть выполнена с возможностью оценки данных на стадиях. В частности, модуль ИИ может быть выполнен таким образом, чтобы по мере исследования различных данных, например, при выполнении протокола обучения, стадия за стадией, каждая связи между данными система присваивала вес, например, на основе прошлых подтверждающих данных и/или характеристик взаимосвязей.

[00972] Чем больше стадий обучения инициировано в системе, тем лучше будет присвоение весов связям и глубже обучение. Кроме того, выгрузка данных на стадиях обеспечивает улучшение схождения данных в системе. В частности, можно также использовать различные парадигмы извлечения признаков, чтобы лучше организовывать, присваивать веса и анализировать наиболее характерные признаки данных, которые будут выгружены. Кроме того, чтобы лучше коррелировать данные,

один или более пользователей могут вводить и/или модулировать основные функции взвешивания, хотя система сама может использовать более совершенную функцию взвешивания на основе протоколов активного обучения.

[00973] Для обеспечения взаимодействия с пользователем можно реализовать один или более аспектов или признаков объекта изобретения, описанных в настоящем документе, на компьютере, имеющем устройство отображения, такое как, например, электронно-лучевая трубка (ЭЛТ), жидкокристаллический дисплей (ЖКД) или монитор на светоизлучающих диодах (СИД), для отображения информации пользователю, и клавиатуру или указательное устройство, такое как, например, мышь или трекбол, с помощью которого пользователь может осуществлять ввод в компьютер. Для обеспечения взаимодействия с пользователем можно также использовать устройства других видов. Например, обратная связь с пользователем может быть любой формой сенсорной обратной связи, такой как, например, визуальная обратная связь, звуковая обратная связь или тактильная обратная связь; а ввод пользователя может приниматься в любой форме, включая, без ограничений, звуковой, речевой или тактильный ввод. В число других возможных устройств ввода входят, без ограничений, сенсорные экраны или другие сенсорные устройства, такие как одно- или многоточечные резистивные или емкостные сенсорные панели, аппаратное или программное обеспечение распознавания голоса, оптические сканеры, оптические указатели, устройства захвата цифровых изображений со связанным программным обеспечением для интерпретации и т.п.

[00974] Объект изобретения, описанный в настоящем документе, может быть реализован в системах, устройствах, способах и/или изделиях в зависимости от требуемой конфигурации. Реализации, указанные в вышеизложенном описании, представляют не все реализации, соответствующие объекту изобретения, описанному в настоящем документе. Напротив, это всего лишь некоторые примеры, соответствующие аспектам, связанным с описанным предметом изобретения. Хотя выше приведено подробное описание нескольких вариантов, возможны другие модификации и дополнения. В частности, описанные в настоящем документе признаки и/или варианты могут быть дополнены другими признаками и/или вариантами. Например, реализации, описанные выше, могут относиться к различным комбинациям и подкомбинациям описанных признаков и/или к комбинациям и подкомбинациям нескольких дополнительных признаков, описанных выше. Кроме того, логические потоки, изображенные на прилагаемых фигурах и/или описанные в настоящем документе, не обязательно требуют соблюдения конкретного показанного порядка или последовательного порядка для достижения требуемых результатов. Возможны другие реализации в объеме следующей формулы изобретения.

## (57) Формула изобретения

1. Способ улучшения точности определения вариантов посредством совместной оценки ридов, которые картируют две или более области референсной последовательности, которые являются гомологичными, причём способ включает:

обращение посредством одного или более компьютеров к совместному скоплению множества ридов последовательности, причём совместное скопление содержит первое скопление ридов, выровненное с первой областью референсной последовательности, и по меньшей мере второе скопление ридов, выровненное со второй областью референсной последовательности, при этом первая область и вторая область гомологичны друг другу;

определение посредством одного или более компьютеров набора вариантов-кандидатов из совместного скопления, причем каждый вариант-кандидат в наборе вариантов-кандидатов соответствует варианту-кандидату между определением основания одного из указанного множества ридов последовательности и определением основания соответствующего местоположения референсной последовательности;

установление посредством одного или более компьютеров порядка обработки вариантов-кандидатов;

оценку посредством одного или более компьютеров каждого варианта-кандидата из набора вариантов-кандидатов на основе установленного порядка обработки; и

формирование посредством одного или более компьютеров и на основе оценки вариантов-кандидатов файла определения вариантов, идентифицирующего один или более вариантов-кандидатов.

2. Способ по п. 1, также включающий

получение множества гомологичных областей референсной последовательности от одного или более запоминающих устройств.

3. Способ по п. 1, согласно которому определение набора вариантов-кандидатов с использованием совместного скопления включает

использование графа де Брёйна для выделения вариантов-кандидатов из совместного скопления.

4. Способ по п. 3, согласно которому узлы в указанном графе представляют список кандидатов,

причём использование графа де Брёйна включает формирование графа де Брёйна с использованием каждой области указанной референсной последовательности в качестве остова и выравнивание каждой позиции варианта-кандидата по универсальным координатам.

5. Способ по п. 1, согласно которому установление посредством одного или более компьютеров порядка обработки вариантов-кандидатов включает

установление посредством одного или более компьютеров порядка обработки вариантов-кандидатов как функции длины рида или размера инсерции.

6. Способ по п. 5, согласно которому установление порядка обработки вариантов-кандидатов как функции длины рида или размера инсерции включает

формирование матрицы связности, устанавливающей порядок обработки вариантов-кандидатов как функции длины рида и размера инсерции.

7. Способ по п. 1, согласно которому оценка посредством одного или более компьютеров каждого варианта-кандидата из набора вариантов-кандидатов на основе установленного порядка обработки включает:

для каждого варианта-кандидата из набора вариантов-кандидатов: формирование совместных диплотипов-кандидатов, расчет апостериорной вероятности каждого совместного диплотипа,

вычисление матрицы генотипа, обрезание совместных диплотипов-кандидатов и включение следующей активной позиции в качестве подтверждающих данных для текущей позиции.

8. Система для улучшения точности определения вариантов посредством совместной оценки ридов, которые картируют две или более области референсной последовательности, которые являются гомологичными, причём система содержит

один или более компьютеров и одно или более устройств хранения, хранящих инструкции, выполненные с возможностью инициирования, при выполнении одним или более компьютерами, выполнения одним или более компьютерами операций,

включающих:

обращение посредством одного или более компьютеров к совместному скоплению множества ридов последовательности, причём совместное скопление содержит первое скопление ридов, выровненное с первой областью референсной последовательности, и по меньшей мере второе скопление ридов, выровненное со второй областью референсной последовательности, при этом первая область и вторая область гомологичны друг другу;

определение посредством одного или более компьютеров набора вариантов-кандидатов из совместного скопления, причем каждый вариант-кандидат в наборе вариантов-кандидатов соответствует варианту-кандидату между определением основания одного из указанного множества ридов последовательности и определением основания соответствующего местоположения референсной последовательности;

установление посредством одного или более компьютеров порядка обработки вариантов-кандидатов;

оценку посредством одного или более компьютеров каждого варианта-кандидата из набора вариантов-кандидатов на основе установленного порядка обработки; и

формирование посредством одного или более компьютеров и на основе оценки вариантов-кандидатов файла определения вариантов, идентифицирующего один или более вариантов-кандидатов.

9. Система по п. 8, в которой операции также включают

получение множества гомологичных областей референсной последовательности от одного или более запоминающих устройств.

10. Система по п. 8, в которой определение набора вариантов-кандидатов с использованием совместного скопления включает

использование графа де Брёйна для выделения вариантов-кандидатов из совместного скопления.

11. Система по п. 10, в которой узлы в указанном графе представляют список кандидатов, причём использование графа де Брёйна включает формирование графа де Брёйна с использованием каждой области указанной референсной последовательности в качестве остова и выравнивание каждой позиции варианта-кандидата по универсальным координатам.

12. Система по п. 8, в которой установление посредством одного или более компьютеров порядка обработки вариантов-кандидатов включает

установление посредством одного или более компьютеров порядка обработки вариантов-кандидатов как функции длины рида или размера инсерции.

13. Система по п. 12, в которой установление порядка обработки вариантов-кандидатов как функции длины рида или размера инсерции включает

формирование матрицы связности, устанавливающей порядок обработки вариантов-кандидатов как функции длины рида и размера инсерции.

14. Система по п. 8, в которой оценка посредством одного или более компьютеров каждого варианта-кандидата из набора вариантов-кандидатов на основе установленного порядка обработки включает:

для каждого варианта-кандидата из набора вариантов-кандидатов: формирование совместных диплотипов-кандидатов, расчет апостериорной вероятности каждого совместного диплотипа,

вычисление матрицы генотипа, обрезание совместных диплотипов-кандидатов и включение следующей активной позиции в качестве подтверждающих данных для текущей позиции.

15. Машиночитаемое устройство хранения, на котором сохранены инструкции, которые, при выполнении устройством обработки данных, инициируют выполнение устройством обработки данных операций для улучшения точности определения вариантов посредством совместной оценки ридов, которые картируют две или более области референсной последовательности, которые являются гомологичными, причём операции включают:

обращение посредством одного или более компьютеров к совместному скоплению множества ридов последовательности, причём совместное скопление содержит первое скопление ридов, выровненное с первой областью референсной последовательности, и по меньшей мере второе скопление ридов, выровненное со второй областью референсной последовательности, при этом первая область и вторая область гомологичны друг другу;

определение посредством одного или более компьютеров набора вариантов-кандидатов из совместного скопления, причем каждый вариант-кандидат в наборе вариантов-кандидатов соответствует варианту-кандидату между определением основания одного из указанного множества ридов последовательности и определением основания соответствующего местоположения референсной последовательности;

установление посредством одного или более компьютеров порядка обработки вариантов-кандидатов;

оценку посредством одного или более компьютеров каждого варианта-кандидата из набора вариантов-кандидатов на основе установленного порядка обработки; и

формирование посредством одного или более компьютеров и на основе оценки вариантов-кандидатов файла определения вариантов, идентифицирующего один или более вариантов-кандидатов.

16. Машиночитаемое устройство хранения по п. 15, в котором операции также включают

получение множества гомологичных областей референсной последовательности от одного или более запоминающих устройств.

17. Машиночитаемое устройство хранения по п. 15, в котором определение набора вариантов-кандидатов с использованием совместного скопления включает

использование графа де Брёйна для выделения вариантов-кандидатов из совместного скопления.

18. Машиночитаемое устройство хранения по п. 17, в котором узлы в указанном графе представляют список кандидатов, причём использование графа де Брёйна включает формирование графа де Брёйна с использованием каждой области указанной референсной последовательности в качестве остова и выравнивание каждой позиции варианта-кандидата по универсальным координатам.

19. Машиночитаемое устройство хранения по п. 15, в котором установление посредством одного или более компьютеров порядка обработки вариантов-кандидатов включает

формирование матрицы связности, устанавливающей порядок обработки вариантов-кандидатов как функции длины рида и размера инсерции.

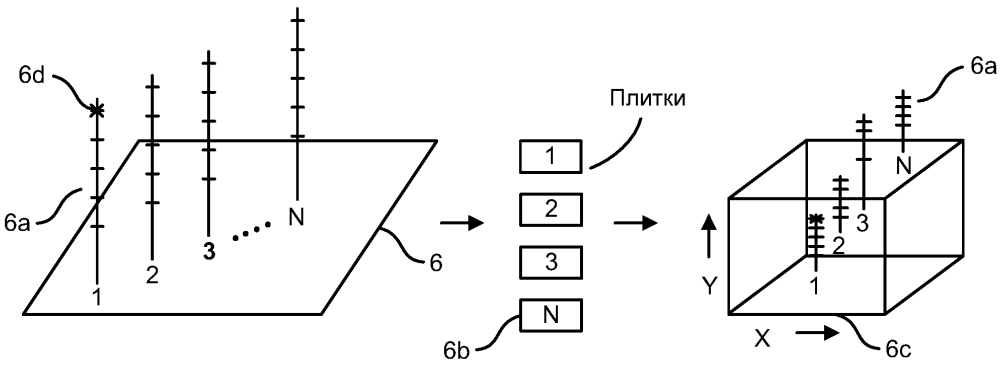
20. Машиночитаемое устройство хранения по п. 15, в котором оценка посредством одного или более компьютеров каждого варианта-кандидата из набора вариантов-кандидатов на основе установленного порядка обработки включает:

для каждого варианта-кандидата из набора вариантов-кандидатов: формирование совместных диплотипов-кандидатов, расчет апостериорной вероятности каждого совместного диплотипа,

вычисление матрицы генотипа, обрезание совместных диплотипов-кандидатов и включение следующей активной позиции в качестве подтверждающих данных для текущей позиции.

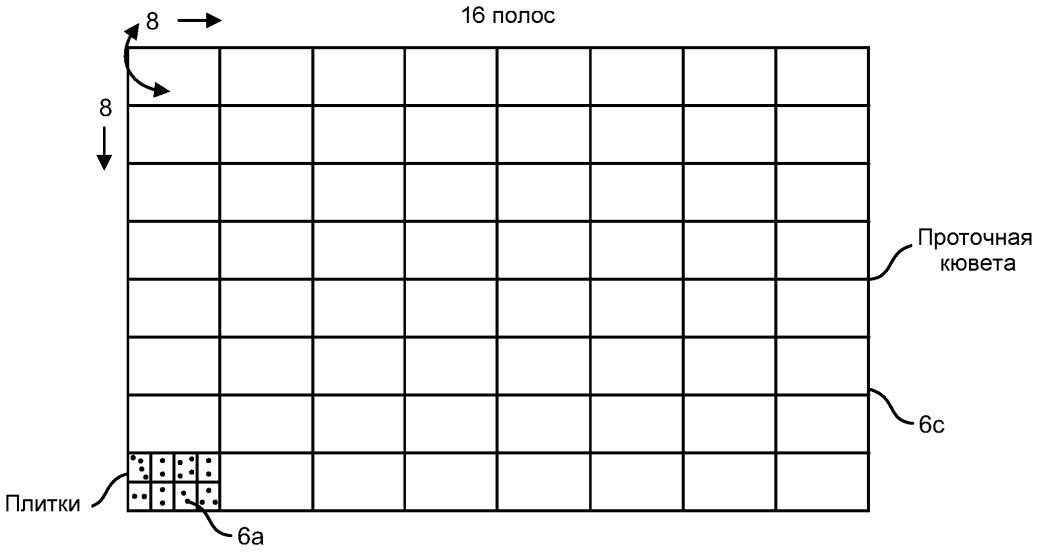
1



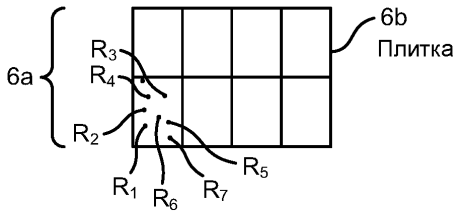



ФИГ. 1А

2

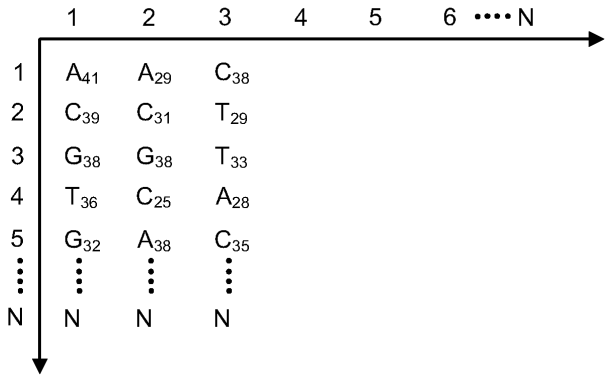



ФИГ. 1B

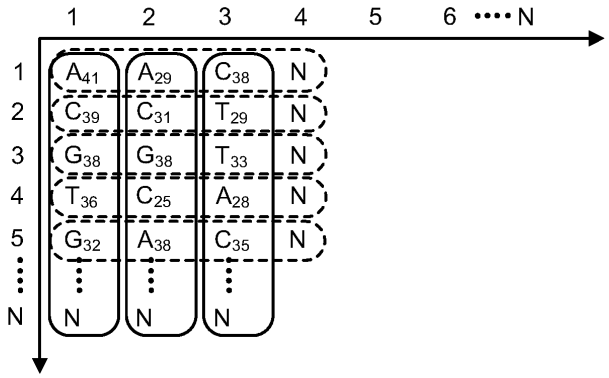



ФИГ. 1C

3




ФИГ. 1D

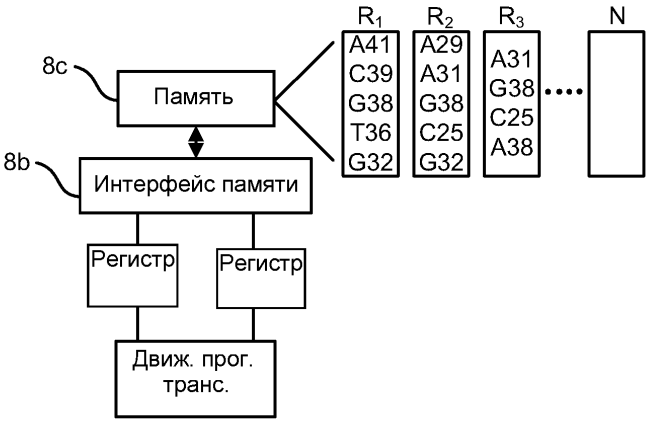



ФИГ. 1E

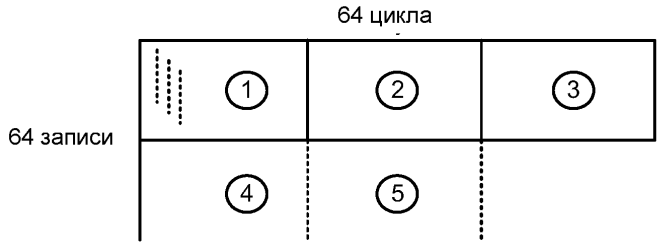
| | | | | |
|---|---|---|---|---|
| Read 1 | $A_{41}$ | $A_{29}$ | $C_{38}$ | N |
| Read 2 | $C_{39}$ | $C_{31}$ | $T_{29}$ | N |
| Read 3 | $G_{38}$ | $G_{38}$ | $T_{33}$ | N |
| Read 4 | $T_{36}$ | $C_{25}$ | $A_{28}$ | N |
| Read 5 | $G_{32}$ | $A_{38}$ | $C_{35}$ | N |

ФИГ. 1F

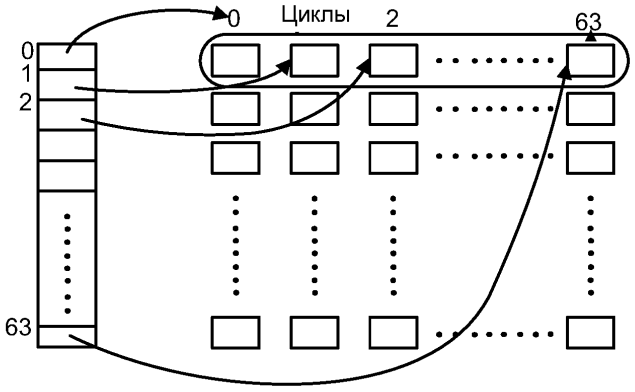
4




ФИГ. 1G

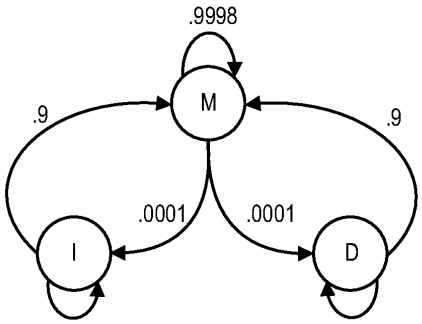



ФИГ. 1H



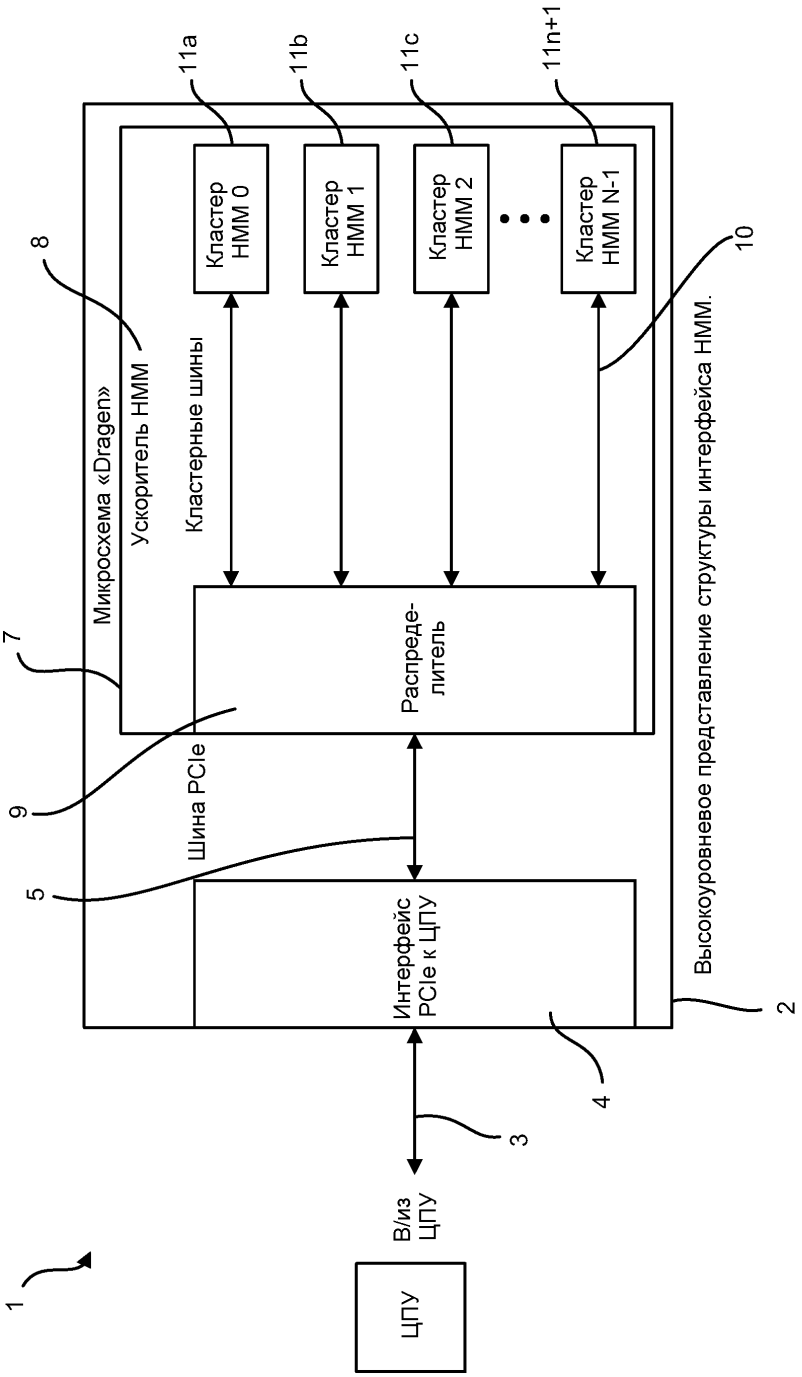

ФИГ. 1I

5



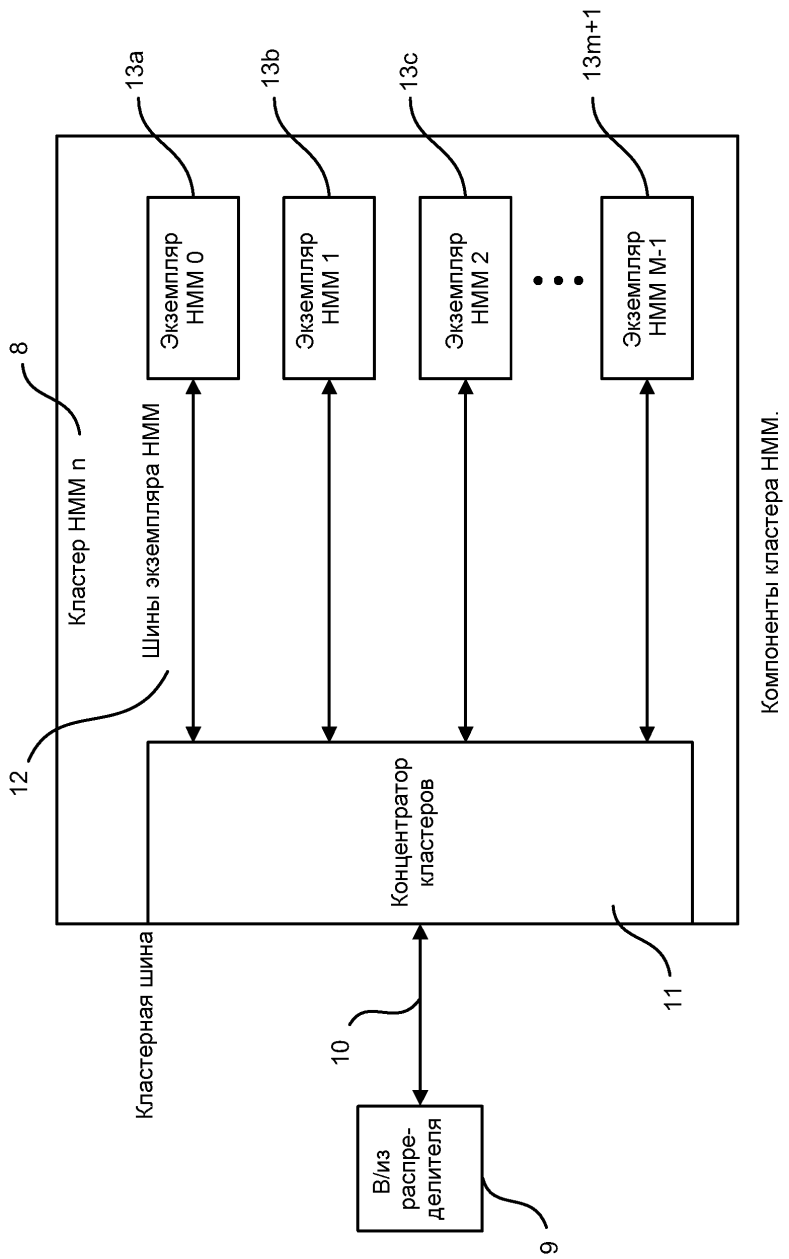

ФИГ. 2

6

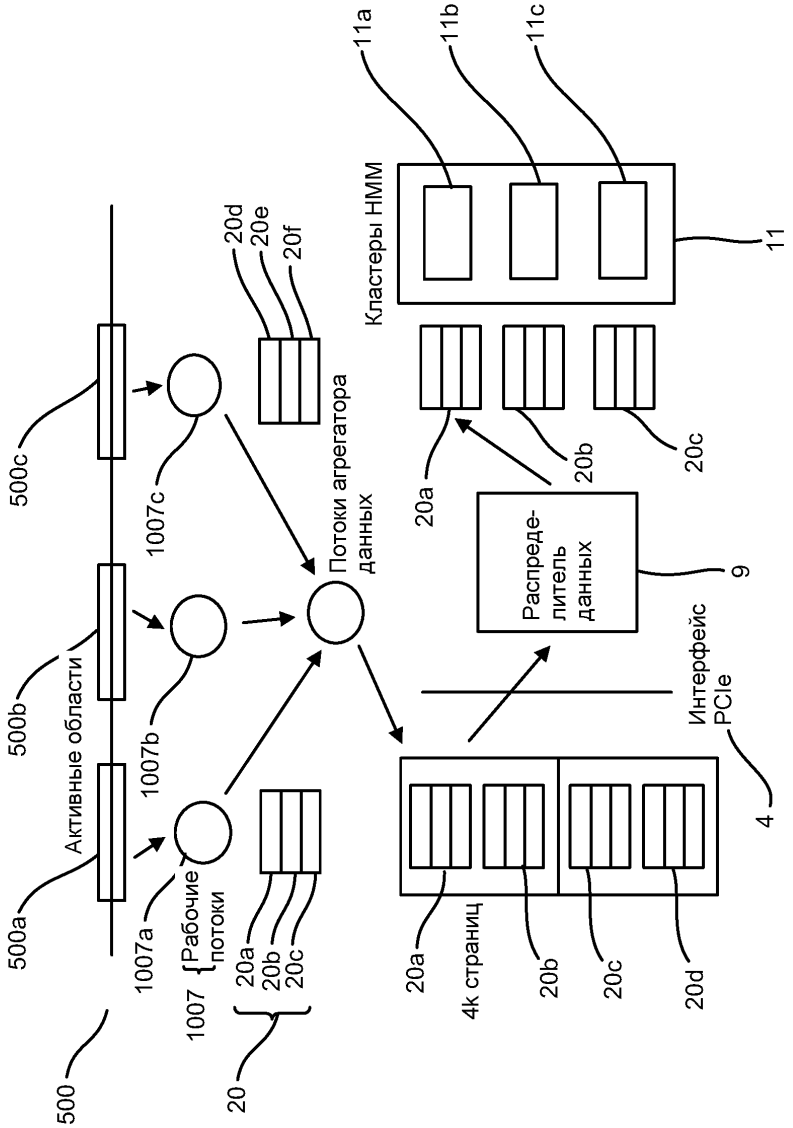

1
ЦПУ
В/из ЦПУ
3
4
2
Высокоуровневое представление структуры интерфейса HMM.
10
Интерфейс PCIe к ЦПУ
5
Шина PCIe
9
Распреде-литель
7
Микросхема «Dragen»
Ускоритель HMM
8
Кластерные шины
Кластер HMM 0
Кластер HMM 1
Кластер HMM 2
Кластер HMM N-1
11a
11b
11c
11n+1

ФИГ. 3А

7


13a
13b
13c
13m+1
Экземпляр НММ 0
Экземпляр НММ 1
Экземпляр НММ 2
Экземпляр НММ М-1
8
Кластер НММ n
Шины экземпляра НММ
12
Концентратор кластеров
Компоненты кластера НММ.
Кластерная шина
10
11
В/из распре-делителя
9

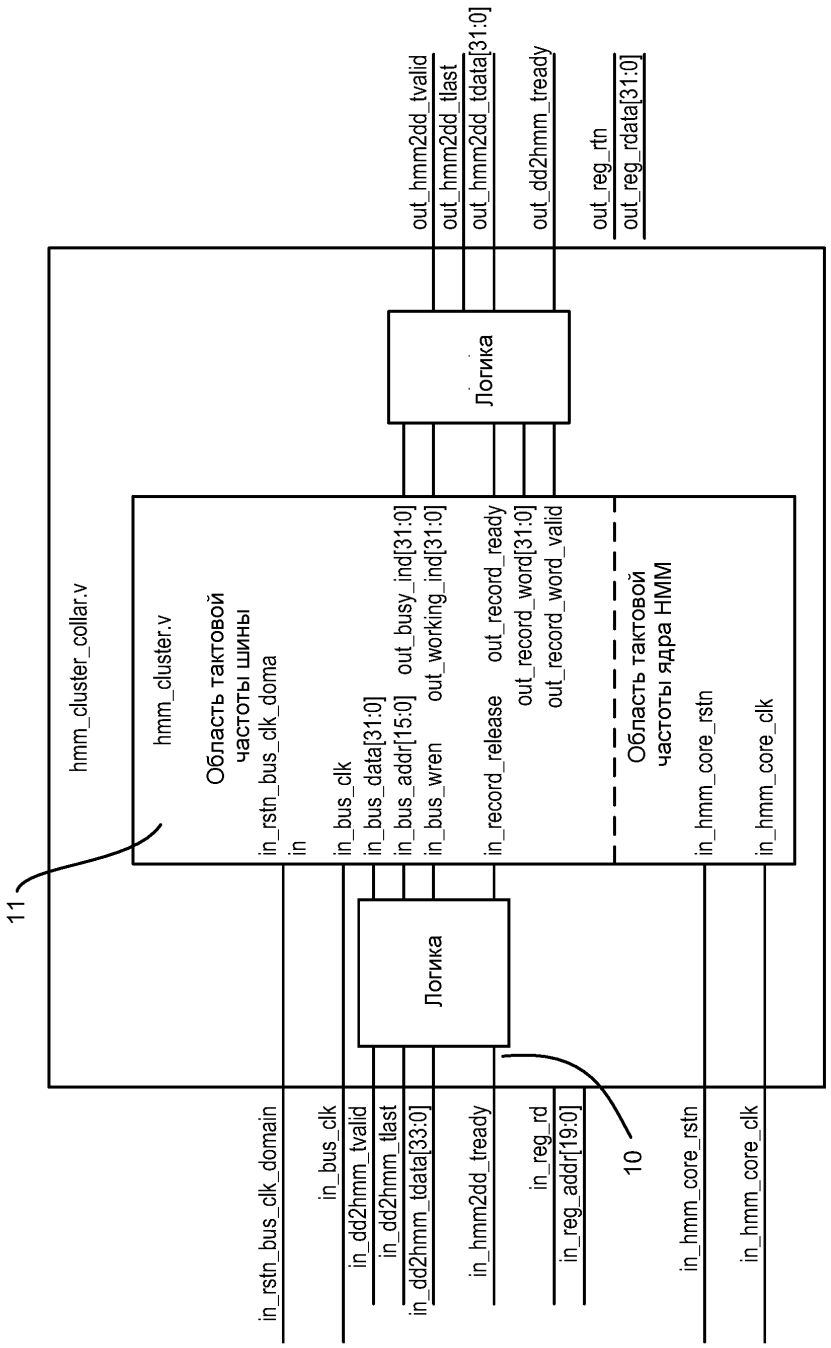
ФИГ. 3В




Обзор потока данных HMM и взаимодействия аппаратного/программного обеспечения.

ФИГ. 4

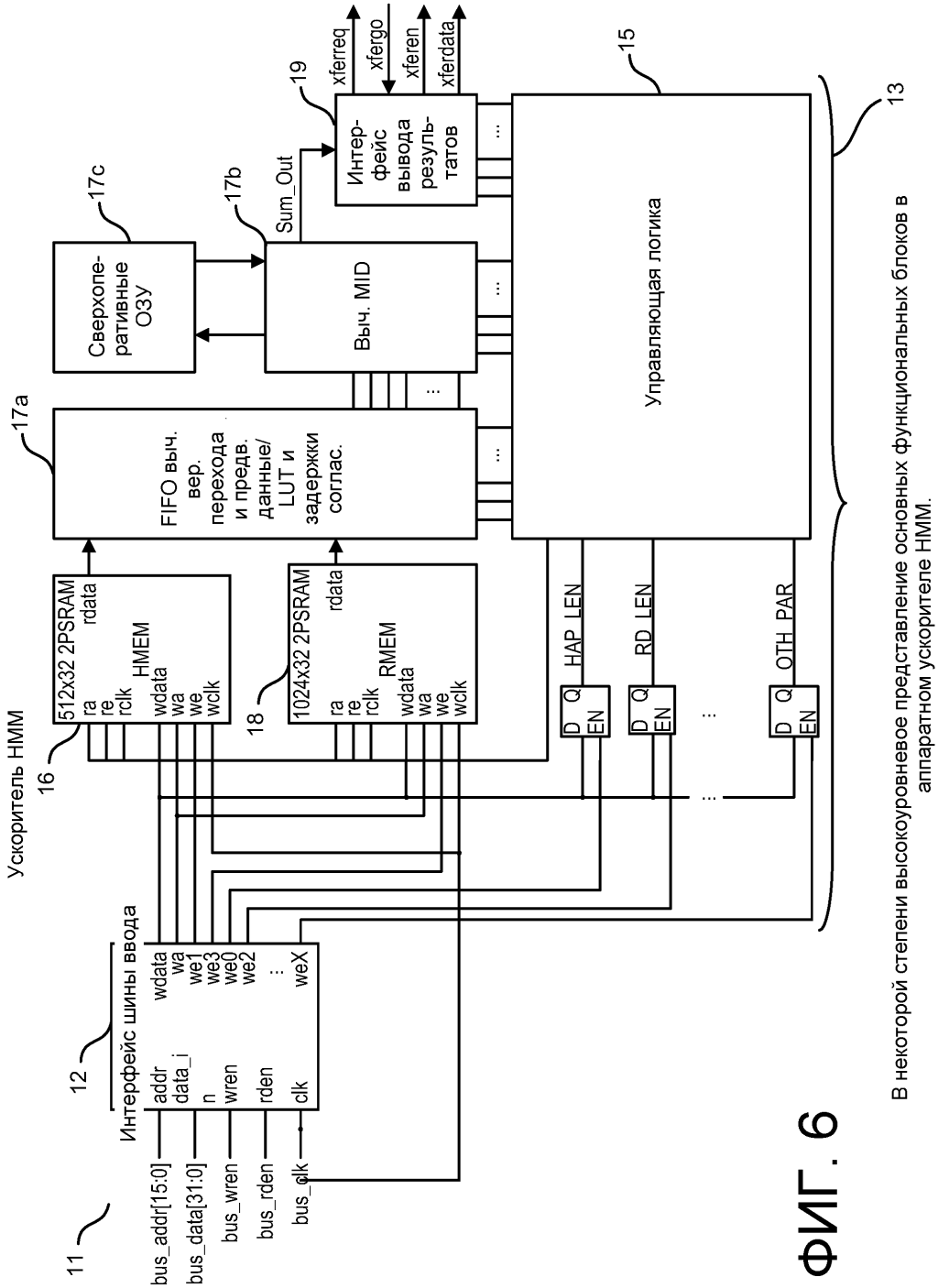
9




Соединения манжеты кластера HMM.

ФИГ. 5

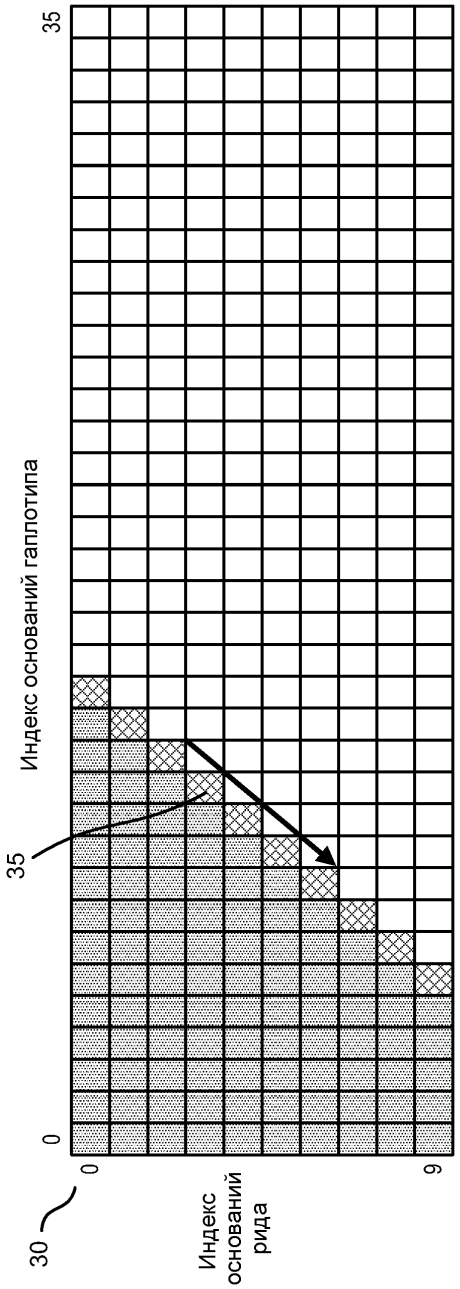
10




ФИГ. 6

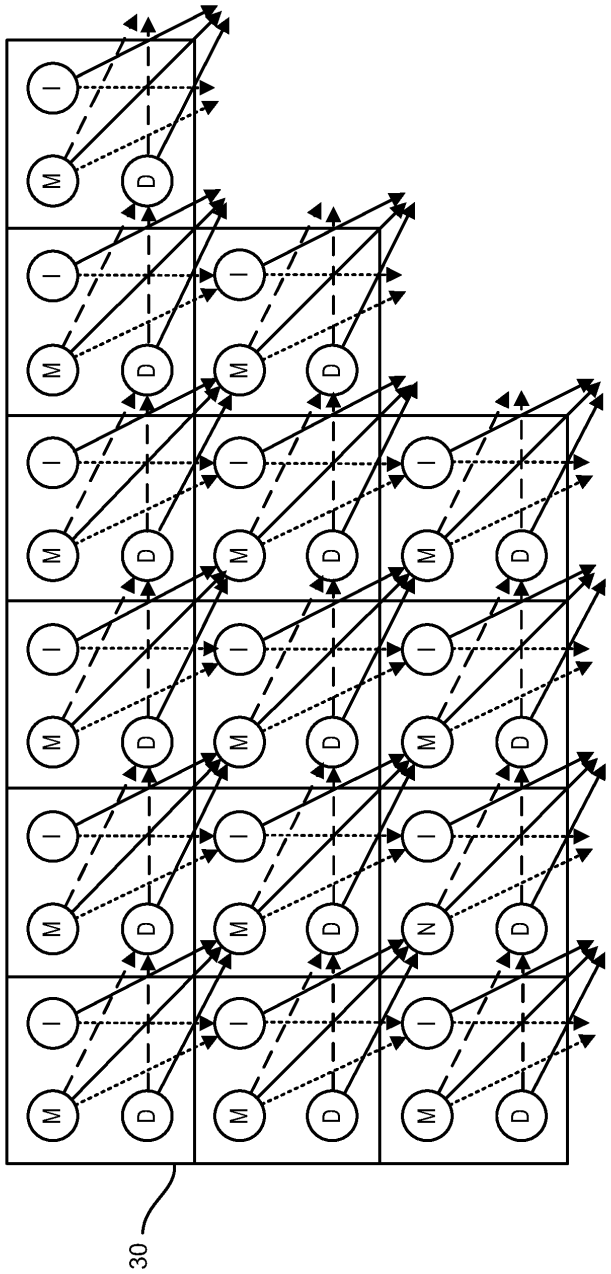
В некоторой степени высокоуровневое представление основных функциональных блоков в аппаратном ускорителе HMM.

11



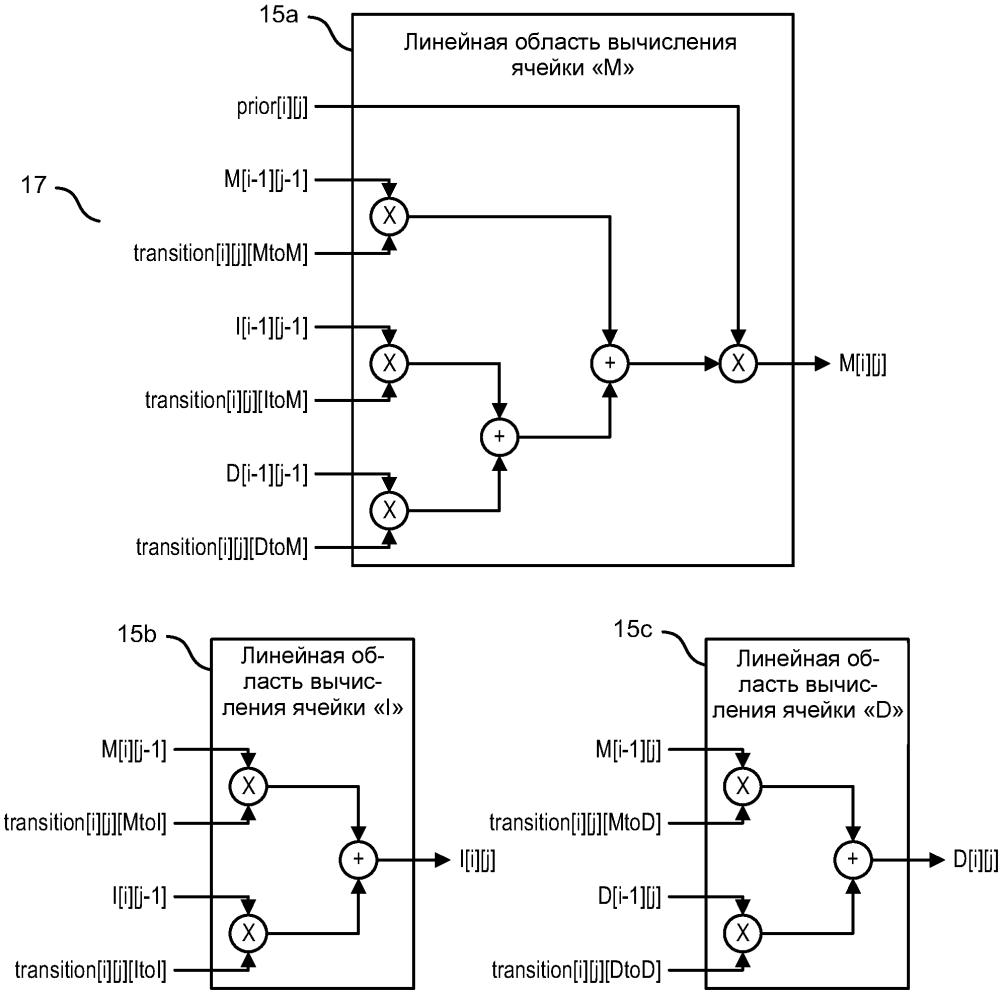

Пример структуры матрицы HMM и аппаратной обработки.

ФИГ. 7



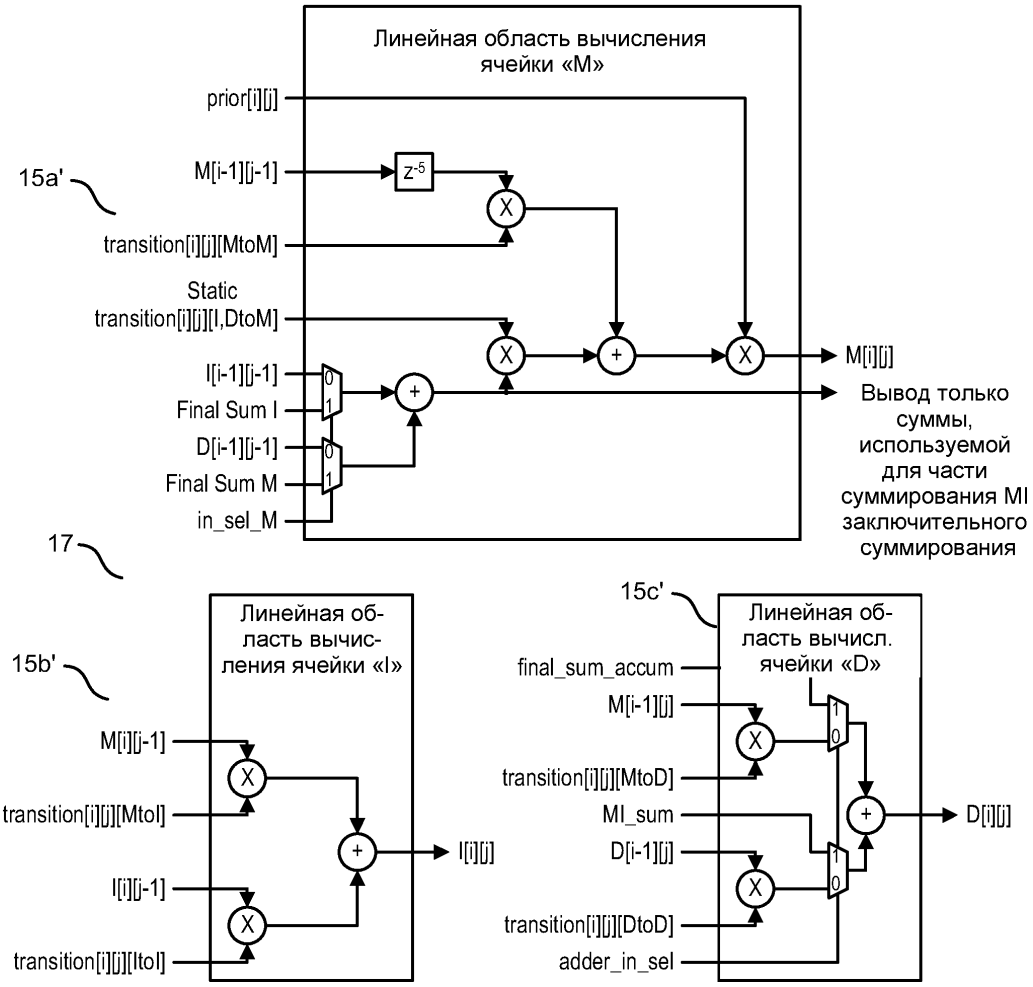

Увеличенный вид части ФИГ. 8, показывающий поток данных и зависимости между соседними ячейками при вычислении состояний M, I и D матрицы HMM.

ФИГ. 8




Вычисления, необходимые для обновления состояний M, I, D.

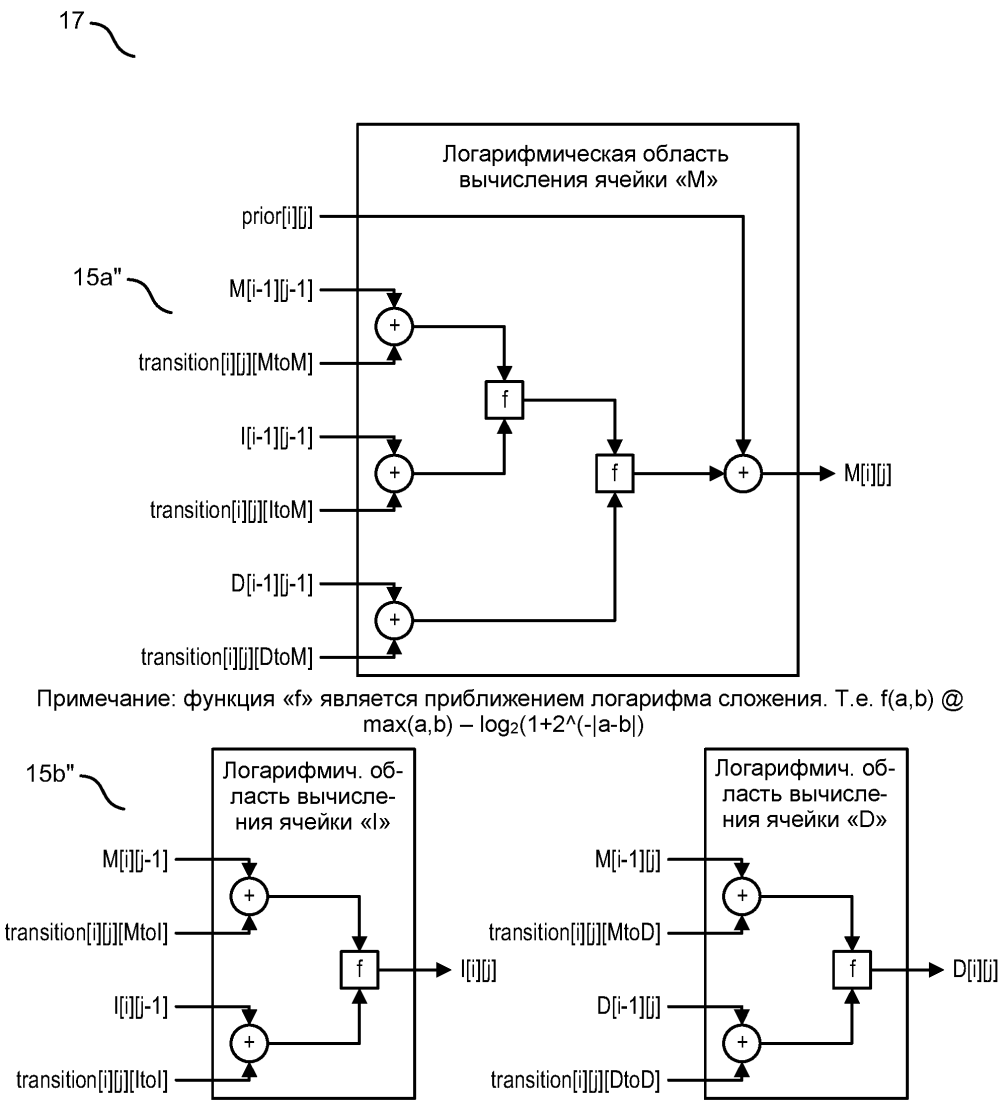
ФИГ. 9




M,1, and D state update circuits, including effects of simplifying assumptions related to transition probabilities and the effect of sharing some M, I, D adder resources with the final sum operations.

ФИГ. 10

15




Подробности вычисления состояний M, I, D в логарифмической области.

ФИГ. 11

16

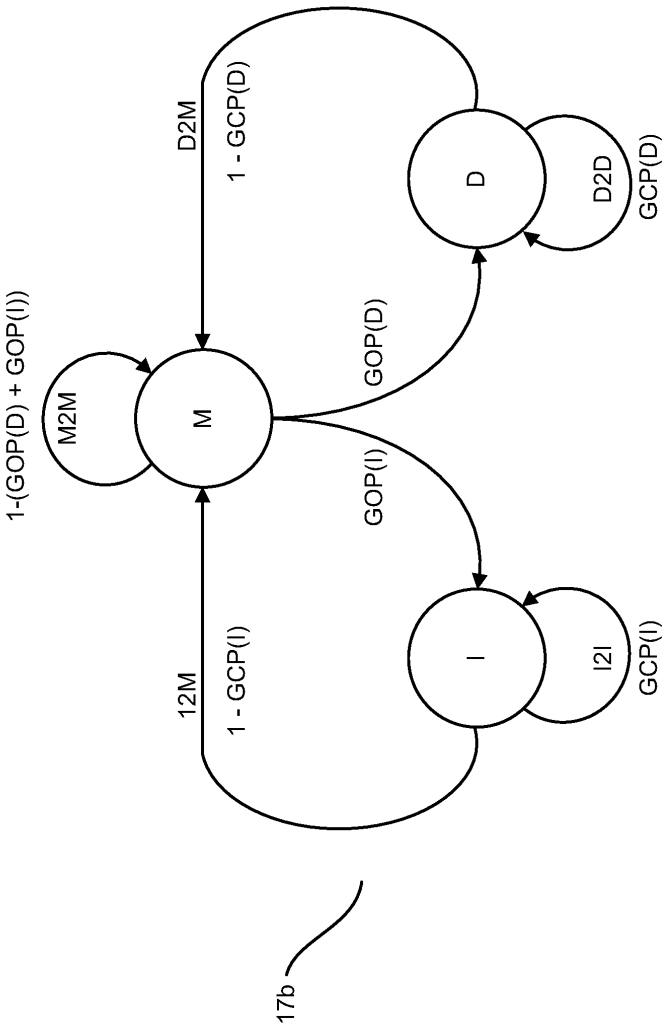



Диаграмма перехода состояний HMM, показывающая взаимосвязь между GOP, GCP и вероятностями перехода.

ФИГ. 12

17

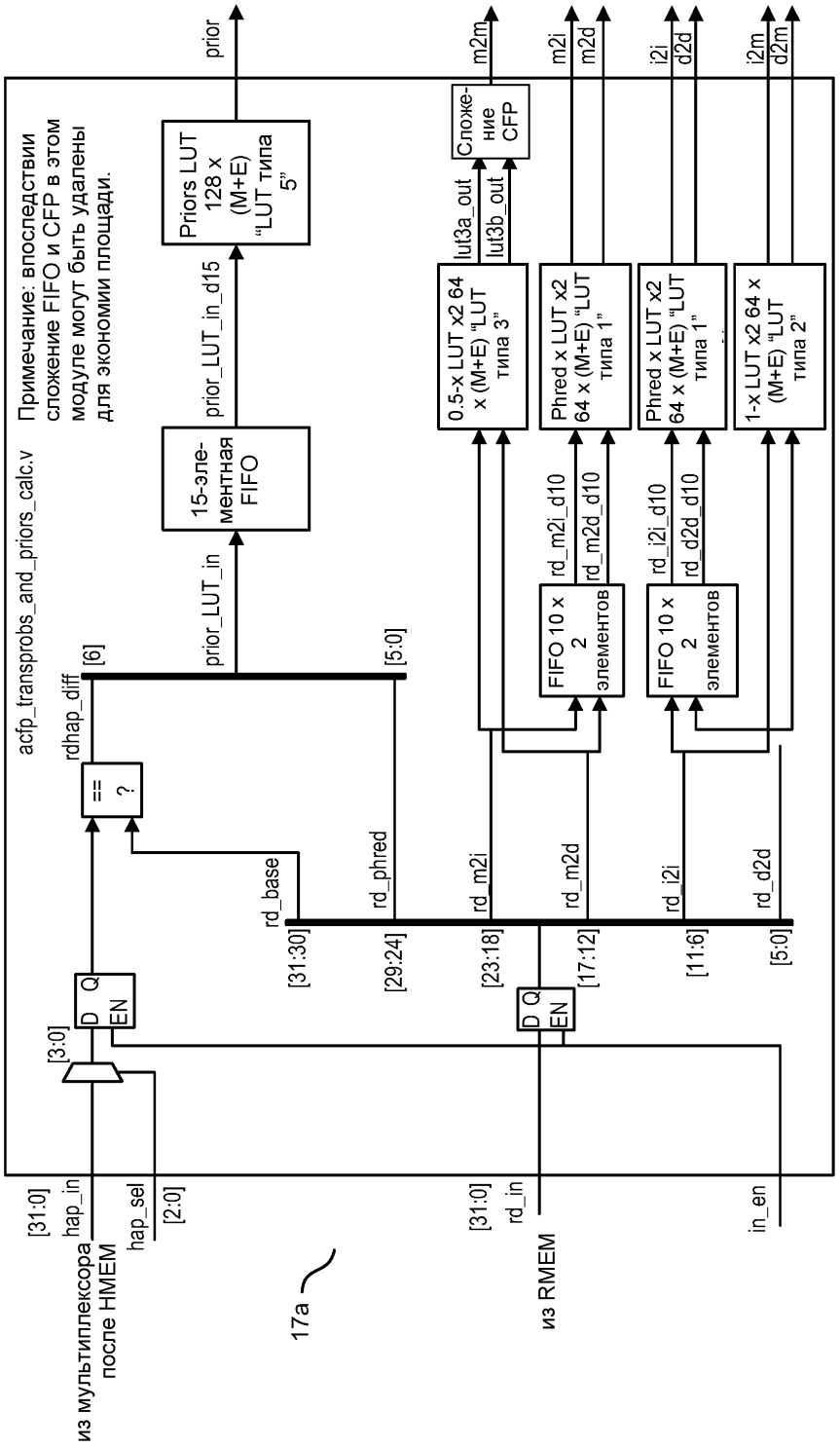
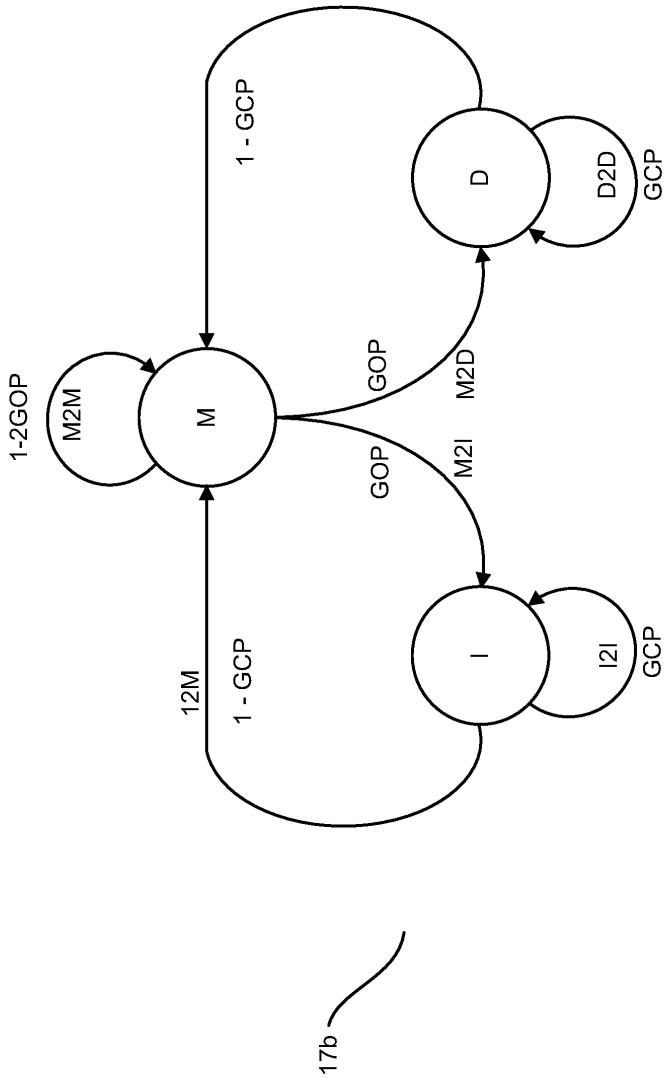



Схема формирования вероятностей перехода HMM и предварительных данных для поддержки общей диаграммы перехода состояний, изображенной на ФИГ. 17.

ФИГ. 13

18




Упрощенная диаграмма перехода состояний HMM, показывающая взаимосвязь между GOP, GCP и вероятностями перехода.

ФИГ. 14

19

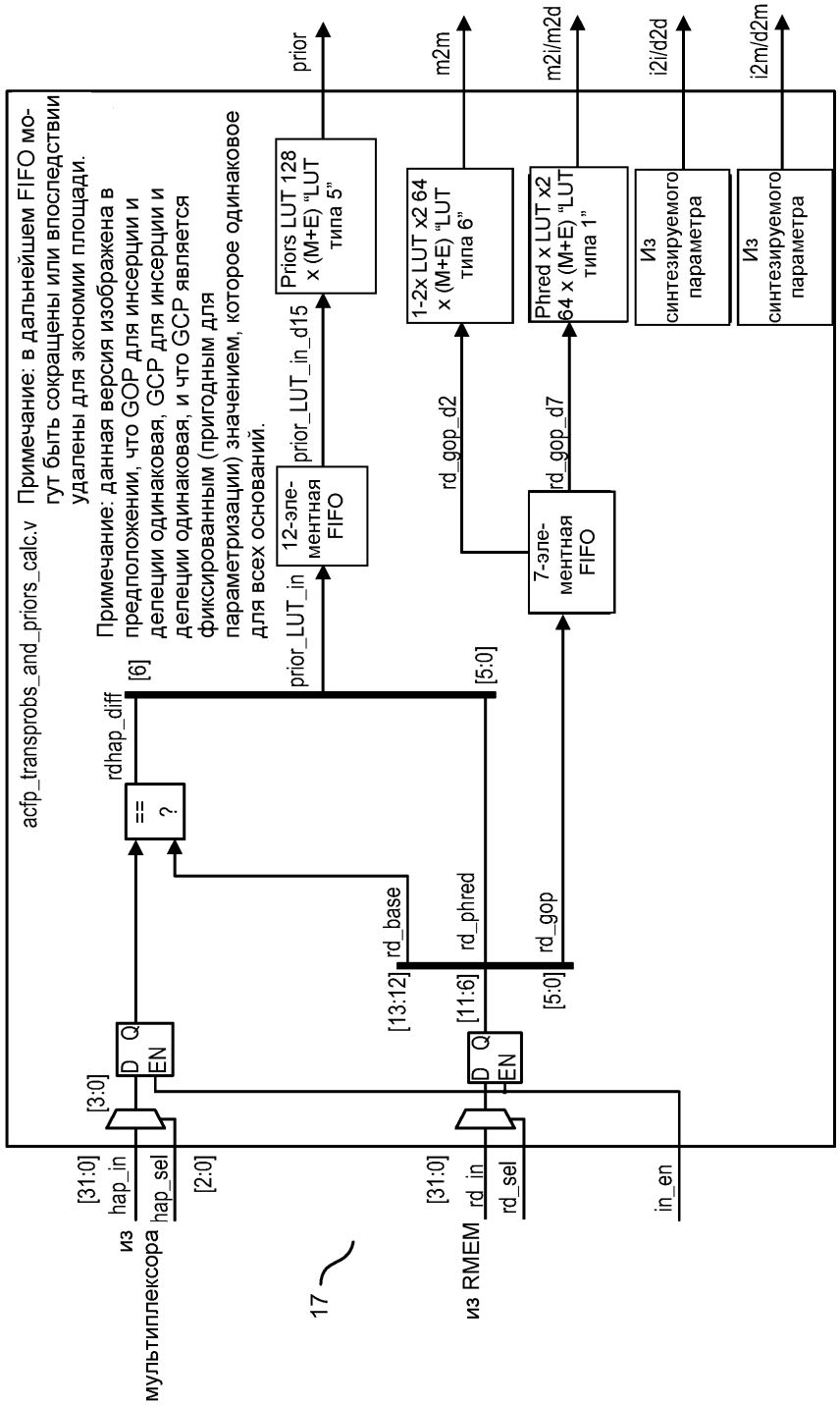
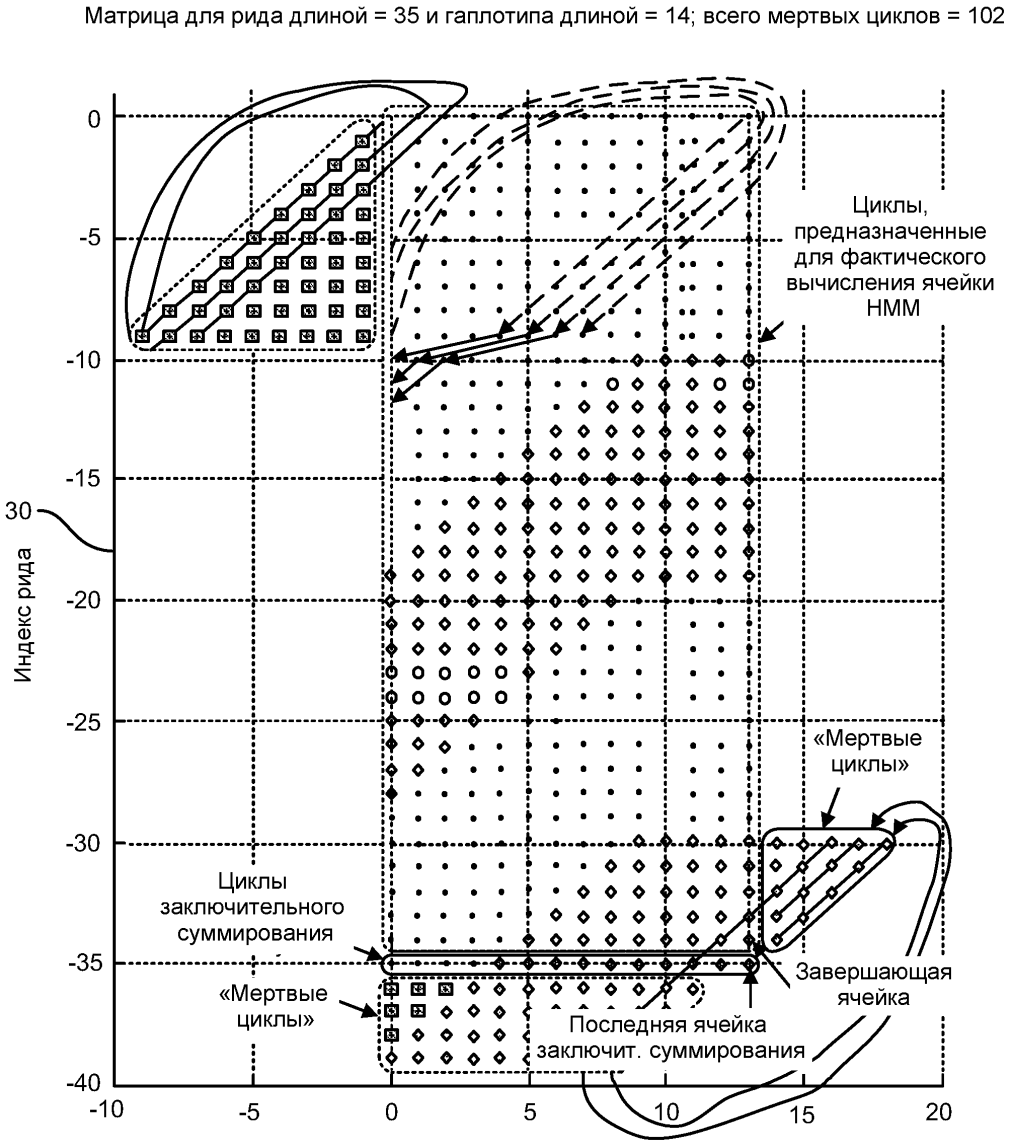


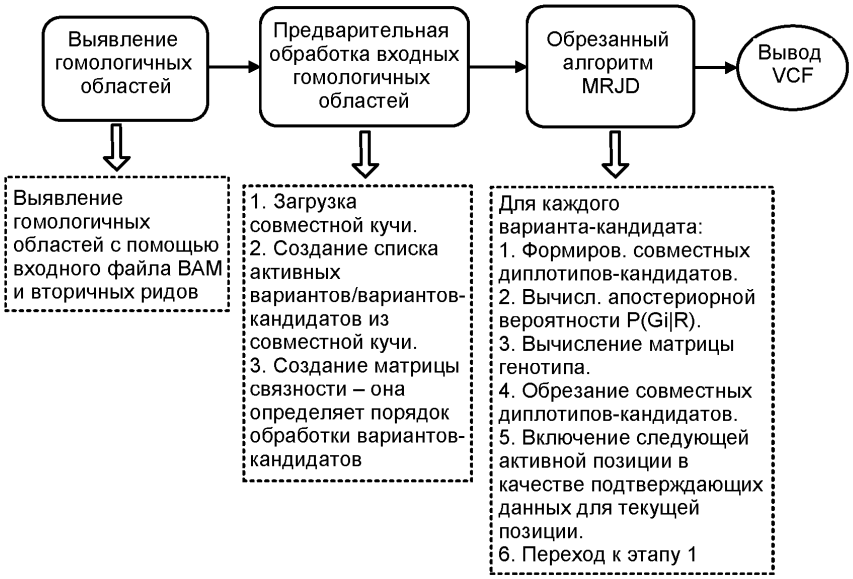

Схема формирования вероятностей перехода HMM и предварительных данных для поддержки упрощенной диаграммы перехода состояний, изображенной на ФИГ. 19.

ФИГ. 15

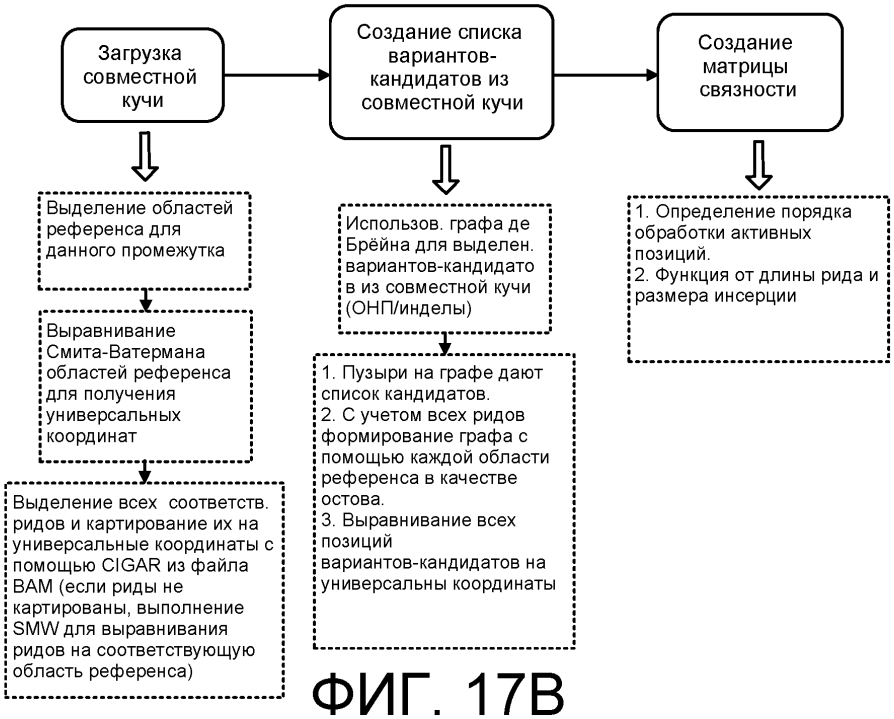
20


Матрица для рида длиной = 35 и гаплотипа длиной = 14; всего мертвых циклов = 102
Индекс рида
0
-5
-10
-15
30
-20
-25
-30
-35
-40
-10
-5
0
5
10
15
20
Циклы, предназначенные для фактического вычисления ячейки HMM
«Мертвые циклы»
Циклы заключительного суммирования
«Мертвые циклы»
Завершающая ячейка
Последняя ячейка заключит. суммирования

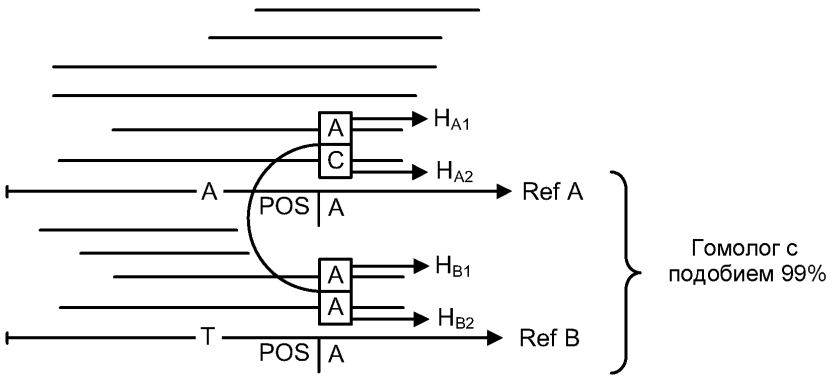
ФИГ. 16



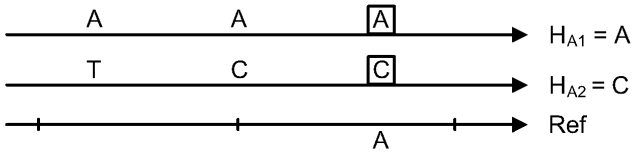

ФИГ. 17A



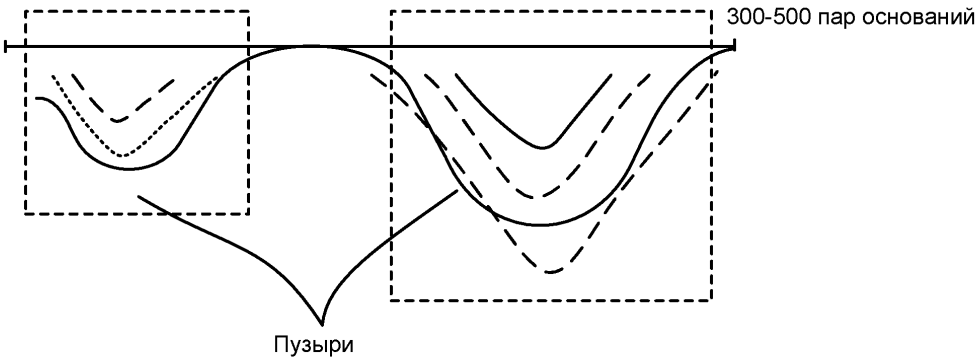

ФИГ. 17B



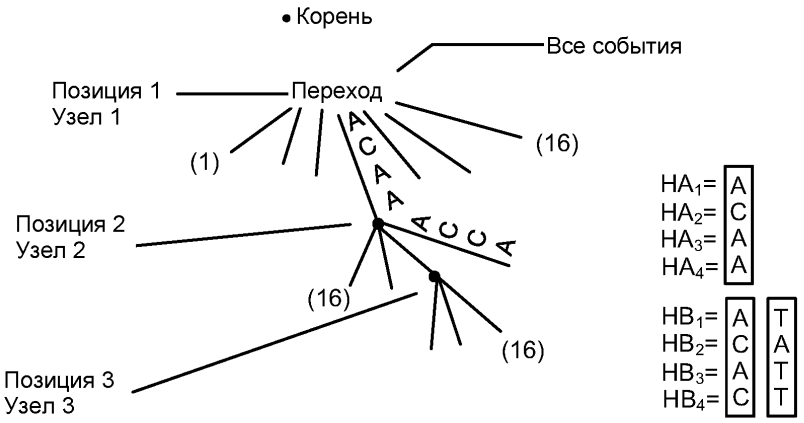

ФИГ. 18A



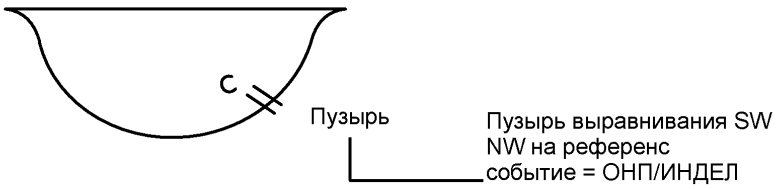

ФИГ. 18B



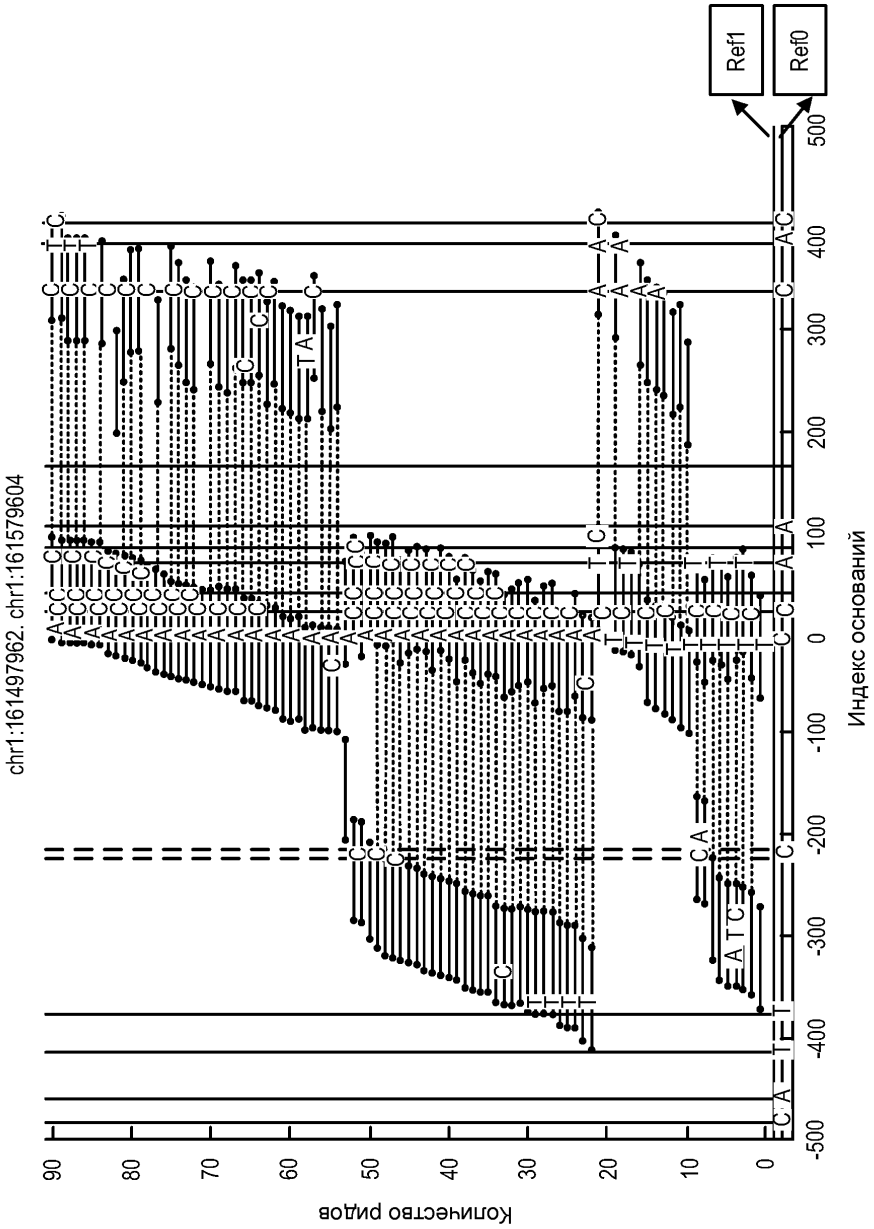

ФИГ. 18C



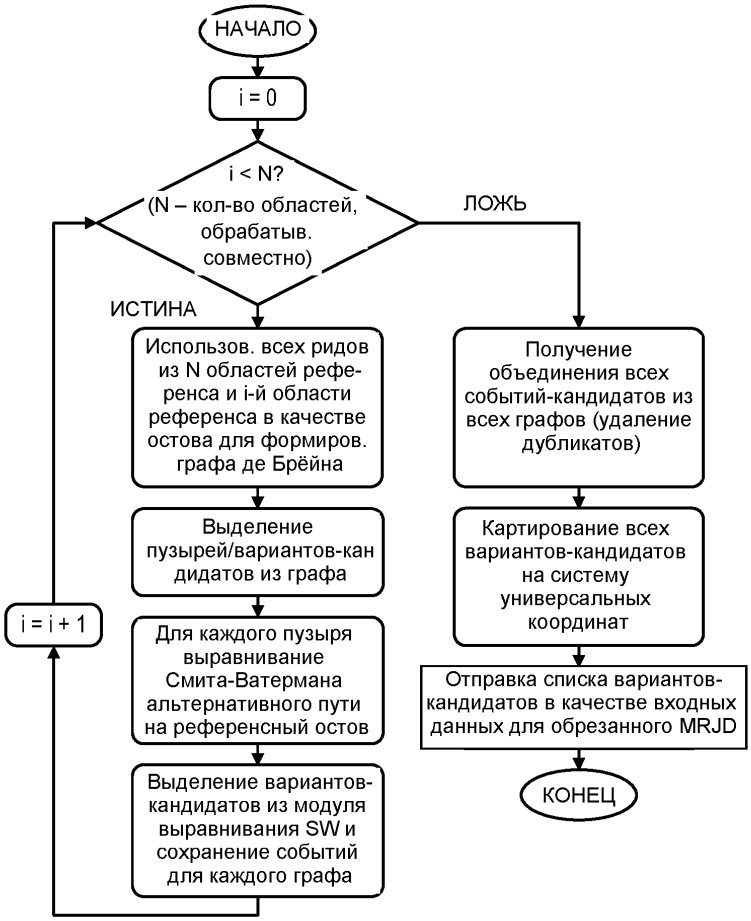

ФИГ. 18D



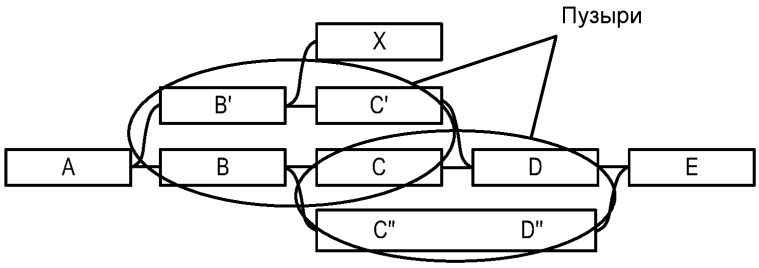

ФИГ. 18E


Ref1
Ref0
chr1:161497962. chr1:161579604
Индекс оснований
Количество рядов
500
400
300
200
100
0
-100
-200
-300
-400
-500
90
80
70
60
50
40
30
20
10
0

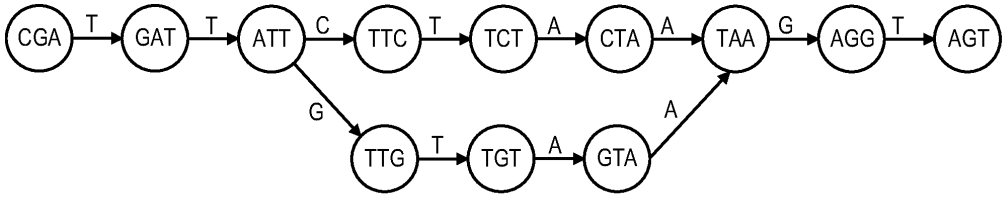
ФИГ. 19


НАЧАЛО
i = 0
i < N?
(N – кол-во областей, обрабатыв. совместно)
ЛОЖЬ
ИСТИНА
Использов. всех ридов из N областей референса и i-й области референса в качестве остова для формиров. графа де Брёйна
Выделение пузырей/вариантов-кандидатов из графа
Для каждого пузыря выравнивание Смита-Ватермана альтернативного пути на референсный остов
Выделение вариантов-кандидатов из модуля выравнивания SW и сохранение событий для каждого графа
i = i + 1
Получение объединения всех событий-кандидатов из всех графов (удаление дубликатов)
Картирование всех вариантов-кандидатов на систему универсальных координат
Отправка списка вариантов-кандидатов в качестве входных данных для обрезанного MRJD
КОНЕЦ

ФИГ. 20

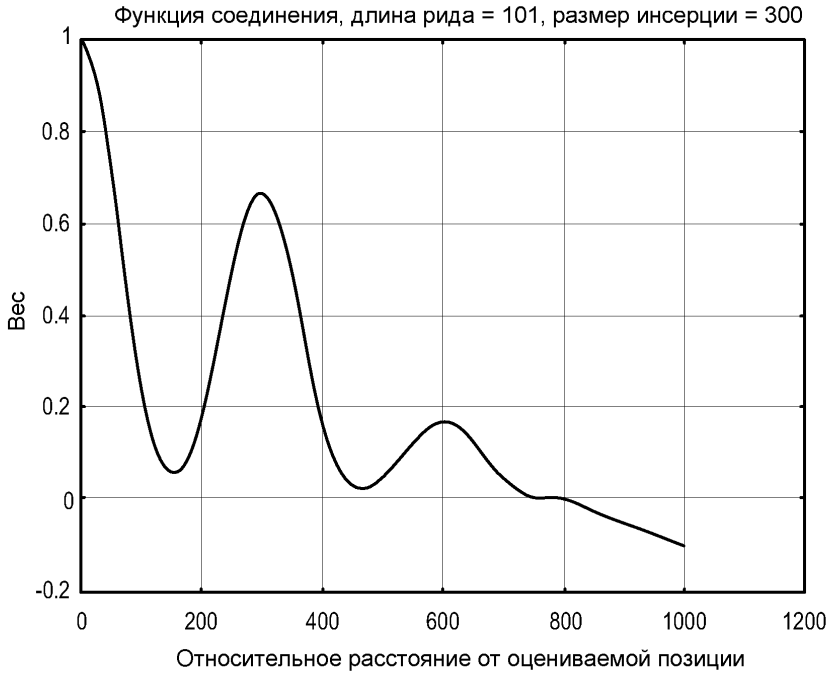

Пузыри
X
B'
C'
A
B
C
D
E
C"
D"

ФИГ. 21

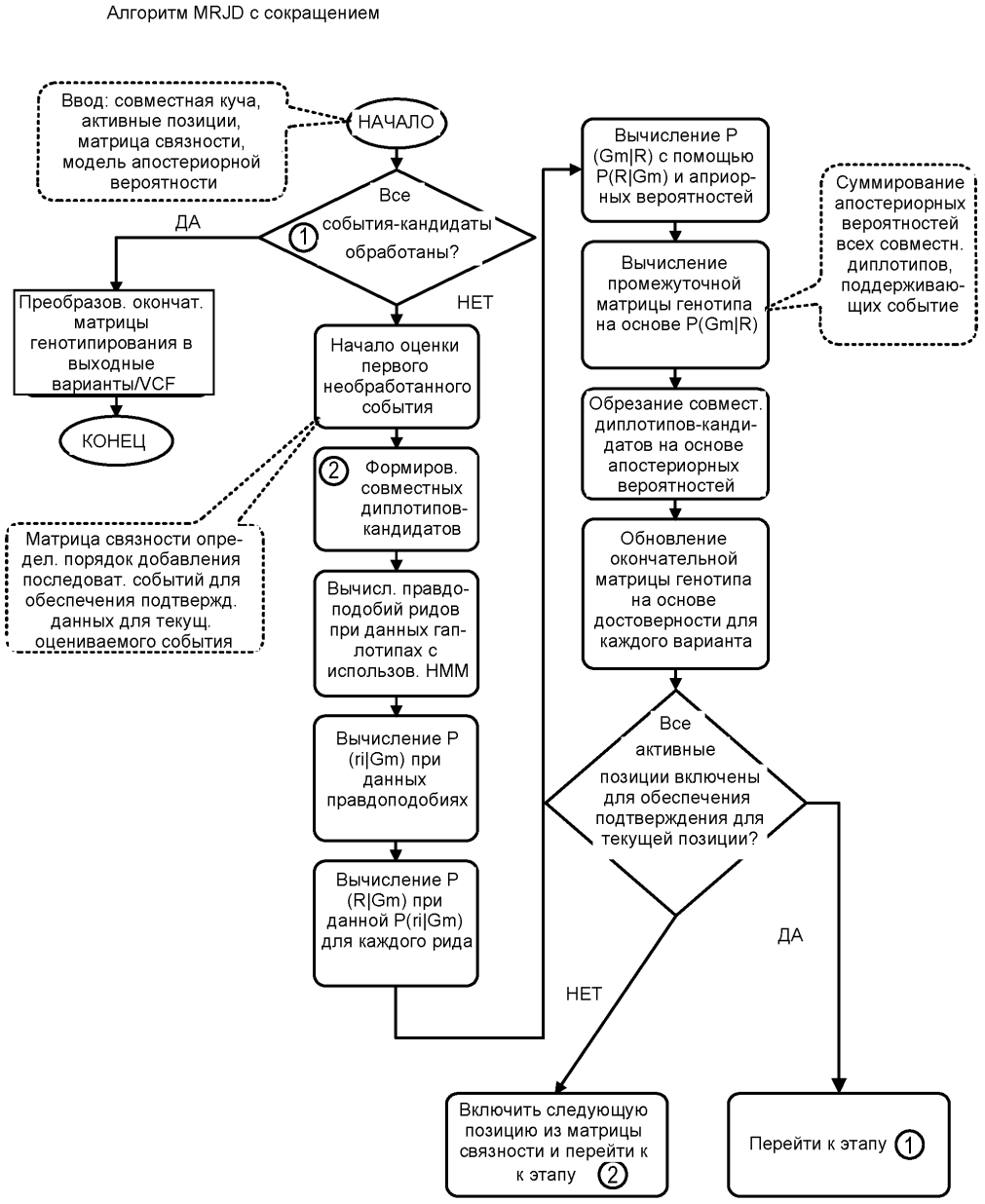



| Гаплотипы-кандидаты | Выделение пузырей |
|---|---|
| CGATT**C**TAAGT | TT**C**TAA |
| CGATT**G**TAAGT | TT**G**TAA |

ФИГ. 22

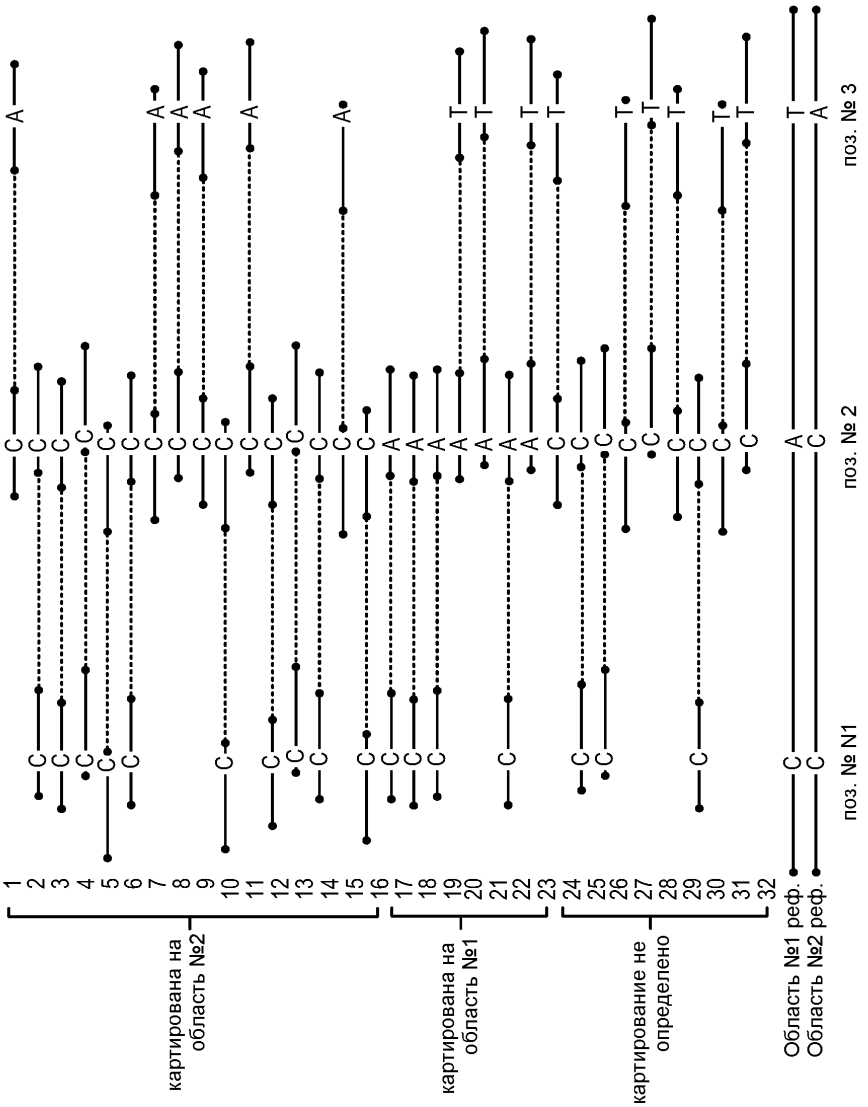



ФИГ. 23


Алгоритм MRJD с сокращением
Ввод: совместная куча, активные позиции, матрица связности, модель апостериорной вероятности
НАЧАЛО
Все события-кандидаты обработаны?
1
ДА
НЕТ
Преобразов. окончат. матрицы генотипирования в выходные варианты/VCF
КОНЕЦ
Начало оценки первого необработанного события
2
Формиров. совместных диплотипов-кандидатов
Матрица связности опред. порядок добавления последоват. событий для обеспечения подтвержд. данных для текущ. оцениваемого события
Вычисл. правдоподобий ридов при данных гаплотипах с использов. HMM
Вычисление P (ri|Gm) при данных правдоподобиях
Вычисление P (R|Gm) при данной P(ri|Gm) для каждого рида
Вычисление P (Gm|R) с помощью P(R|Gm) и априорных вероятностей
Вычисление промежуточной матрицы генотипа на основе P(Gm|R)
Суммирование апостериорных вероятностей всех совместн. диплотипов, поддерживающих событие
Обрезание совмест. диплотипов-кандидатов на основе апостериорных вероятностей
Обновление окончательной матрицы генотипа на основе достоверности для каждого варианта
Все активные позиции включены для обеспечения подтверждения для текущей позиции?
ДА
НЕТ
Включить следующую позицию из матрицы связности и перейти к к этапу 2
Перейти к этапу 1

ФИГ. 24

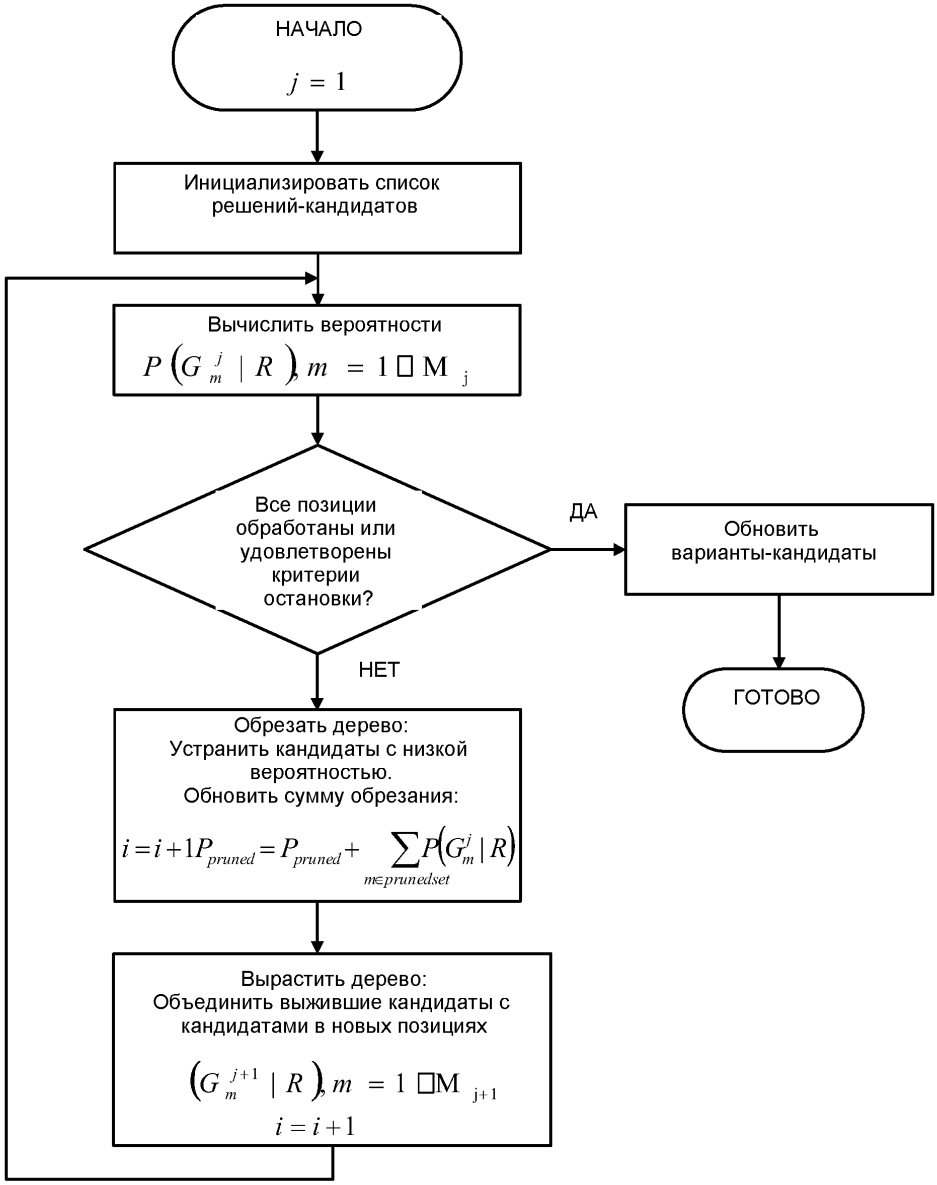
28


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
картирована на область №2
картирована на область №1
картирование не определено
Область №1 реф.
Область №2 реф.
поз. № N1
поз. № 2
поз. № 3

ФИГ. 25

| i | $H_1$=CAT | $H_2$=CAA | $H_3$=CCT | $H_4$=CCA | $H_5$=GAT | $H_6$=GAA | $H_7$=GCT | $H_8$=GCA | CAT GCA CAT GCA | CAT GCA CCT GCA |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.4901 | 0.4909 |
| 2 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 3 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 4 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 5 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 6 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 7 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.4901 | 0.4909 |
| 8 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.4901 | 0.4909 |
| 9 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.4901 | 0.4909 |
| 10 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 11 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.4901 | 0.4909 |
| 12 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 13 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 14 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 15 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.0000 | 0.0033 | 0.0033 | 0.9801 | 0.4901 | 0.4909 |
| 16 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.4901 | 0.4909 |
| 17 | 0.9801 | 0.9801 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.0000 | 0.0000 | 0.4901 | 0.2459 |
| 18 | 0.9801 | 0.9801 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.0000 | 0.0000 | 0.4901 | 0.2459 |
| 19 | 0.9801 | 0.9801 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.0000 | 0.0000 | 0.4901 | 0.2459 |
| 20 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.4901 | 0.2459 |
| 21 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.4901 | 0.2459 |
| 22 | 0.9801 | 0.9801 | 0.0033 | 0.0033 | 0.0033 | 0.0033 | 0.0000 | 0.0000 | 0.4901 | 0.2459 |
| 23 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.4901 | 0.2459 |
| 24 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.2475 |
| 25 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.2475 |
| 26 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.2475 |
| 27 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.2475 |
| 28 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.2475 |
| 29 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.2475 |
| 30 | 0.0033 | 0.0033 | 0.9801 | 0.9801 | 0.0000 | 0.0000 | 0.0033 | 0.0033 | 0.0033 | 0.2475 |
| 31 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.2475 |
| 32 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.0000 | 0.9801 | 0.0033 | 0.0033 | 0.2475 |
| | | | | | | | | | 3.5e-30 | 2.2e-15 |

ФИГ. 26


НАЧАЛО
$j = 1$
Инициализировать список решений-кандидатов
Вычислить вероятности
$P\left(G_m^j \mid R\right), m = 1 \square \mathrm{M}_j$
Все позиции обработаны или удовлетворены критерии остановки?
ДА
Обновить варианты-кандидаты
ГОТОВО
НЕТ
Обрезать дерево:
Устранить кандидаты с низкой вероятностью.
Обновить сумму обрезания:
$i = i+1 P_{pruned} = P_{pruned} + \sum_{m \in prunedset} P\left(G_m^j \mid R\right)$
Вырастить дерево:
Объединить выжившие кандидаты с кандидатами в новых позициях
$\left(G_m^{j+1} \mid R\right), m = 1 \square \mathrm{M}_{j+1}$
$i = i+1$

ФИГ. 27

31

| | | |
|---|---|---|
| $G^1_1 = \begin{bmatrix} C & G \\ C & G \end{bmatrix}$ | $P(G_1\|R) = 0.99892171$ (phred 0.00) |
| $G^1_2 = \begin{bmatrix} G & G \\ C & G \end{bmatrix}$ | $P(G_2\|R) = 0.00103398$ (phred 29.85) |
| $G^1_3 = \begin{bmatrix} C & G \\ C & C \end{bmatrix}$ | $P(G_3\|R) = 0.00003934$ (phred 44.05) |
| $G^1_4 = \begin{bmatrix} G & G \\ C & C \end{bmatrix}$ | $P(G_4\|R) = 0.00000251$ (phred 56.00) |
| $G^1_5 = \begin{bmatrix} G & C \\ G & C \end{bmatrix}$ | $P(G_5\|R) = 0.00000126$ (phred 59.00) |
| $G^1_6 = \begin{bmatrix} G & G \\ G & C \end{bmatrix}$ | $P(G_6\|R) = 0.00000116$ (phred 59.35) |
| $G^1_7 = \begin{bmatrix} G & C \\ C & C \end{bmatrix}$ | $P(G_7\|R) = 0.00000004$ (phred 73.55) |
| $G^1_8 = \begin{bmatrix} G & G \\ G & G \end{bmatrix}$ | $P(G_8\|R) = 0.00000000$ (phred 151.67) |
| $G^1_8 = \begin{bmatrix} C & C \\ C & C \end{bmatrix}$ | $P(G_9\|R) = 0.00000000$ (phred 225.85) |

ФИГ. 28

| | | |
|---|---|---|
| $G^2_1$ = | CC GC<br>CA GC | P($G_1$IR) = 0.99799456 (phred 0.01) |
| $G^2_2$ = | CA GC<br>CA GC | P($G_2$IR) = 0.00101673 (phred 29.93) |
| $G^2_3$ = | CA GC<br>CA CC | P($G_3$IR) = 0.00097796 (phred 30.10) |
| $G^2_4$ = | CC GC<br>CA CC | P($G_4$IR) = 0.00000913 (phred 50.39) |
| $G^2_5$ = | GC GC<br>CA CC | P($G_5$IR) = 0.00000125 (phred 59.02) |
| $G^2_6$ = | GC GC<br>CA GC | P($G_6$IR) = 0.00000015 (phred 68.29) |
| $G^2_7$ = | GA GC<br>CA CC | P($G_7$IR) = 0.00000007 (phred 71.45) |
| $G^2_8$ = | CC GC<br>CA GA | P($G_8$IR) = 0.00000007 (phred 71.45) |
| $G^2_9$ = | GA GC<br>CA GC | P($G_9$IR) = 0.00000006 (phred 72.06) |
| $G^2_{10}$ = | CC GC<br>CC CA | P($G_{10}$IR) = 0.00000001 (phred 79.89) |
| $G^2_{11}$ = | GC CC<br>GC CA | P($G_{11}$IR) = 0.00000000 (phred 88.51) |
| $G^2_{12}$ = | GC GC<br>CC CA | P($G_{12}$IR) = 0.00000000 (phred 88.51) |
| $G^2_{13}$ = | CC GC<br>CA CA | P($G_{13}$IR) = 0.00000000 (phred 89.10) |
| $G^2_{14}$ = | GA GC<br>CC GC | P($G_{14}$IR) = 0.00000000 (phred 116.35) |
| $G^2_{15}$ = | CA GC<br>CA GA | P($G_{15}$IR) = 0.00000000 (phred 120.60) |

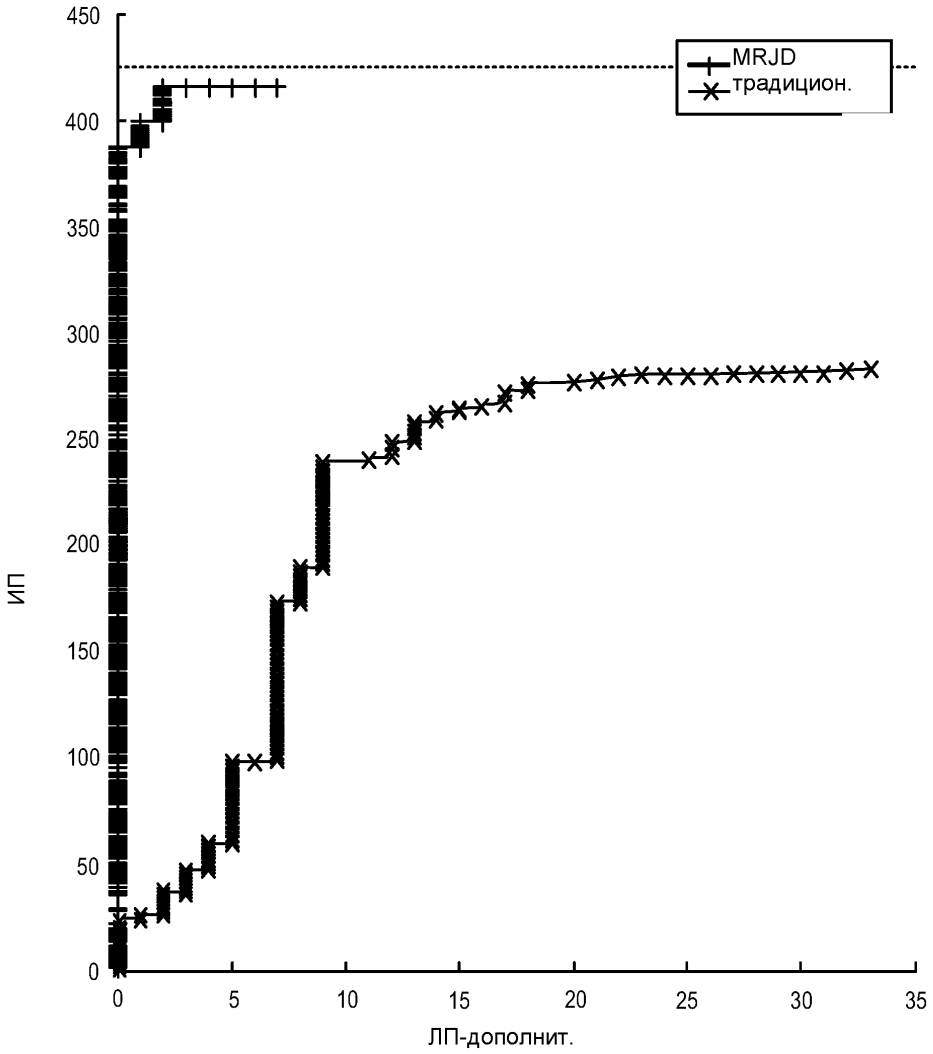
(кандидаты 16-70 не показаны)

ФИГ. 29

| | | |
|---|---|---|
| $G^3_1$ = | CCT GCA<br>CAT GCA | P($G_1$IR) = 0.99885810 (phred 0.00) |
| $G^3_2$ = | CCT GCT<br>CAT GCA | P($G_2$IR) = 0.00111991 (phred 29.51) |
| $G^3_3$ = | CCT GCA<br>CAT CCA | P($G_3$IR) = 0.00000914 (phred 50.39) |
| $G^3_4$ = | CAT GCT<br>CAT GCA | P($G_4$IR) = 0.00000462 (phred 53.36) |
| $G^3_5$ = | CAT GCA<br>CAT CCT | P($G_5$IR) = 0.00000222 (phred 56.53) |
| $G^3_6$ = | CAT GCT<br>CAT CCA | P($G_6$IR) = 0.00000222 (phred 56.53) |
| $G^3_7$ = | CCA GCT<br>CAT GCT | P($G_7$IR) = 0.00000126 (phred 59.01) |
| $G^3_8$ = | GCT GCA<br>CAT CCA | P($G_8$IR) = 0.00000126 (phred 59.01) |
| $G^3_9$ = | CCA GCT<br>CAT GCA | P($G_9$IR) = 0.00000126 (phred 59.01) |
| $G^3_{10}$ = | CCT GCA<br>CAT CCT | P($G_{10}$IR) = 0.00000001 (phred 79.89) |
| $G^3_{11}$ = | CCT GCT<br>CAT CCA | P($G_{11}$IR) = 0.00000001 (phred 79.89) |
| $G^3_{12}$ = | GCT GCA<br>CAT CCT | P($G_{12}$IR) = 0.00000000 (phred 88.52) |
| $G^3_{13}$ = | GCT GCT<br>CAT CCA | P($G_{13}$IR) = 0.00000000 (phred 88.52) |
| $G^3_{14}$ = | CAT GCT<br>CAA GCA | P($G_{14}$IR) = 0.00000000 (phred 91.82) |
| $G^3_{15}$ = | CCT GCA<br>CAA GCA | P($G_{15}$IR) = 0.00000000 (phred 94.70) |

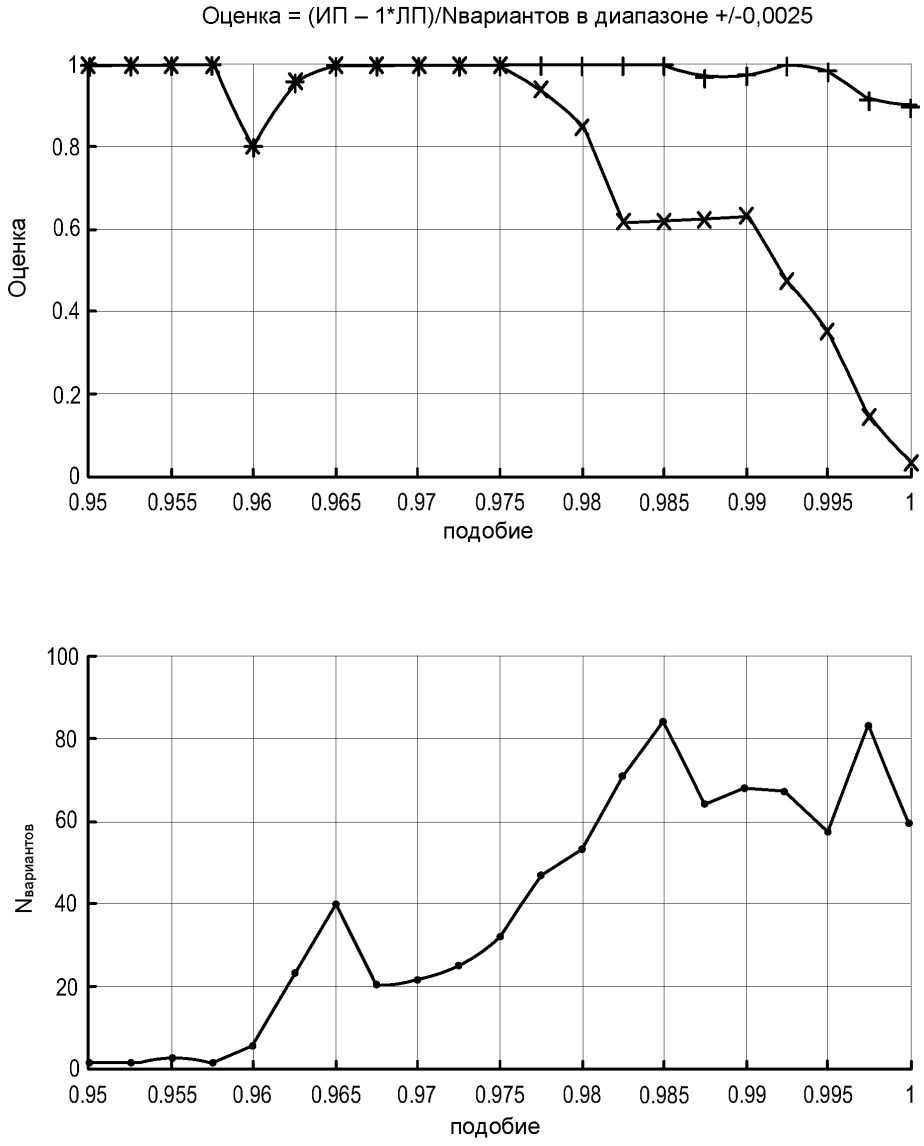
(кандидаты 16-65 не показаны)

ФИГ. 30

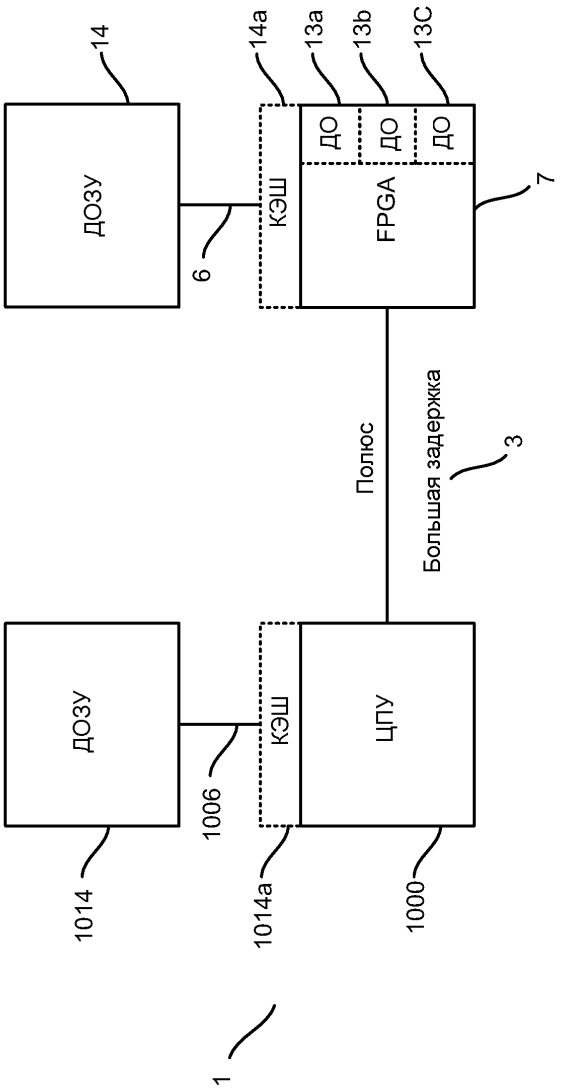

450
400
350
300
250
200
150
100
50
0
ИП
0
5
10
15
20
25
30
35
ЛП-дополнит.
MRJD
традицион.

ФИГ. 31

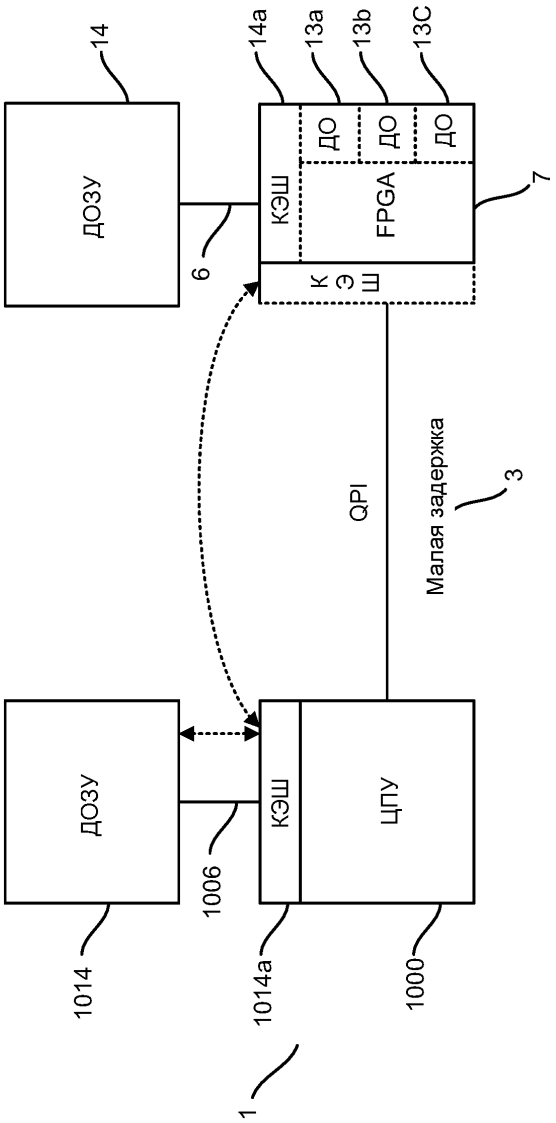
35


Оценка = (ИП – 1*ЛП)/Nвариантов в диапазоне +/-0,0025
Оценка
1
0.8
0.6
0.4
0.2
0
0.95
0.955
0.96
0.965
0.97
0.975
0.98
0.985
0.99
0.995
1
подобие
Nвариантов
100
80
60
40
20
0
0.95
0.955
0.96
0.965
0.97
0.975
0.98
0.985
0.99
0.995
1
подобие

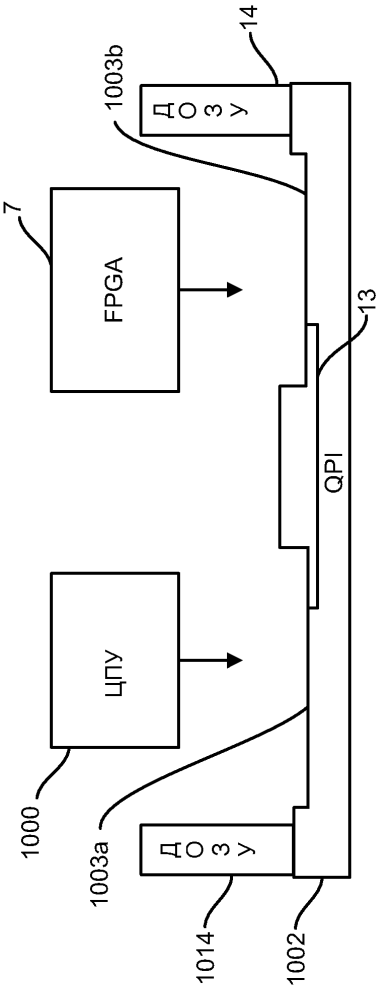
ФИГ. 32


14
14a
13a
13b
13C
ДО
ДО
ДО
ДОЗУ
КЭШ
FPGA
7
6
Полюс
Большая задержка
3
ДОЗУ
КЭШ
ЦПУ
1006
1014
1014a
1000
1

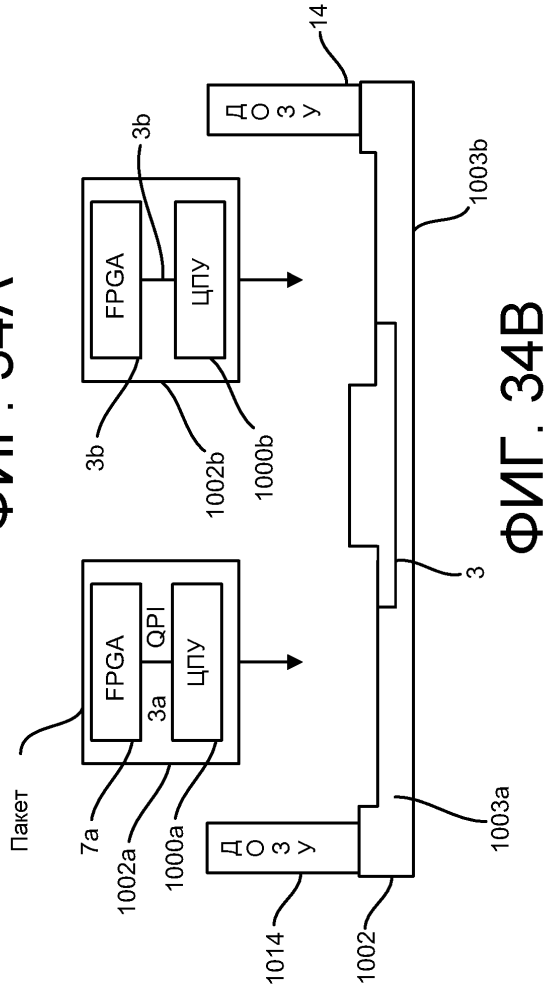
ФИГ. 33А


14
14a
13a
13b
13C
ДОЗУ
КЭШ
ДО
ДО
ДО
FPGA
7
6
К
Э
Ш
QPI
Малая задержка
3
ДОЗУ
КЭШ
ЦПУ
1006
1014
1014a
1000
1

ФИГ. 33В

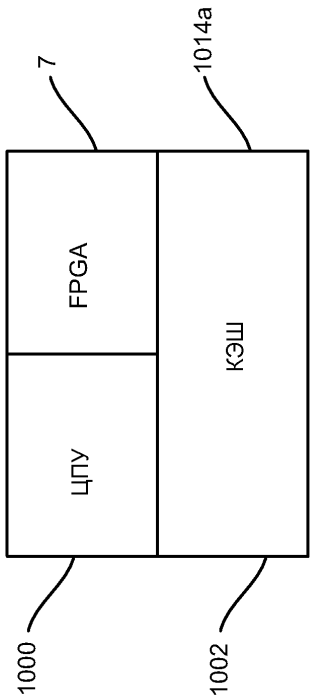



ФИГ. 34A



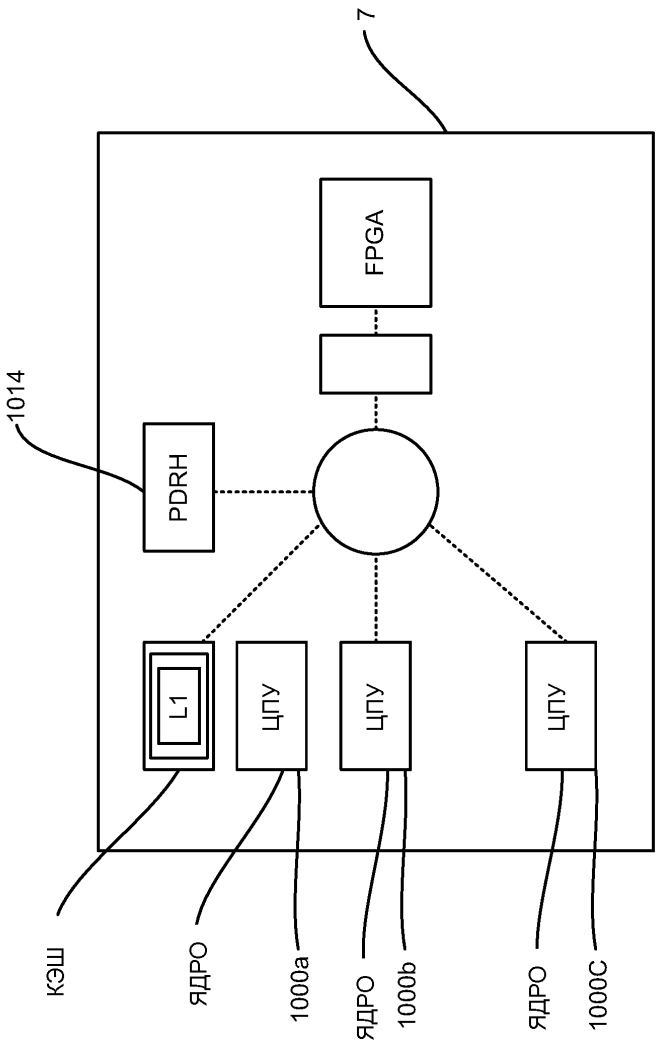

ФИГ. 34B

39

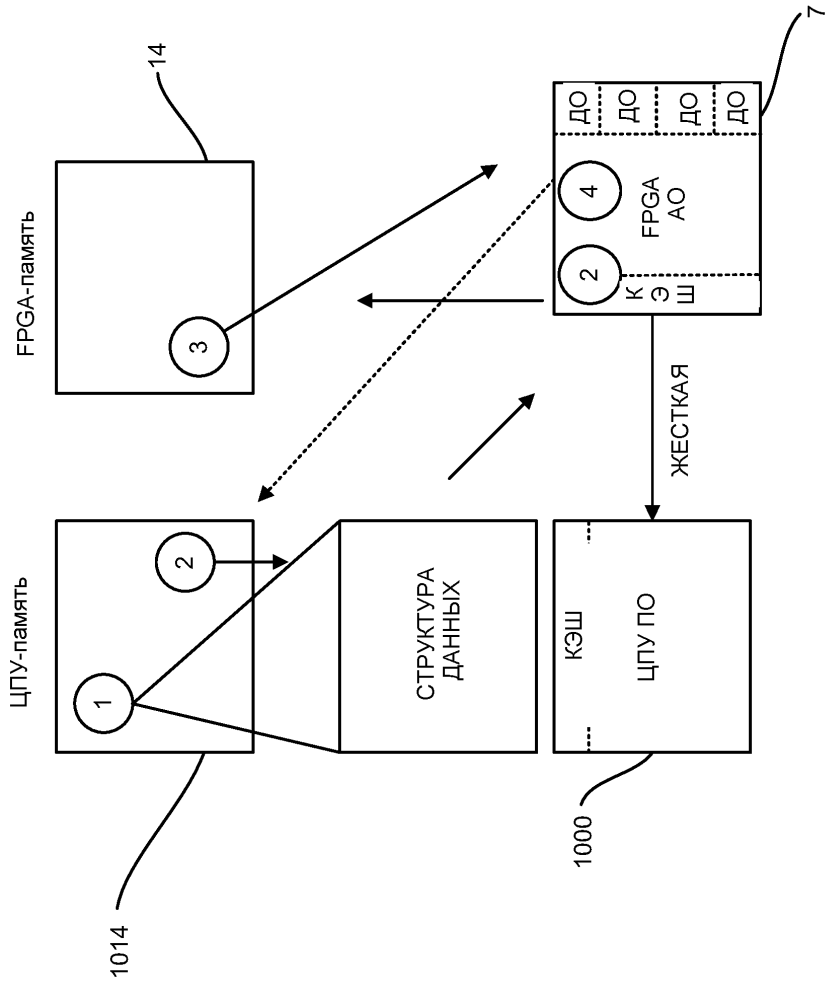



ФИГ. 35

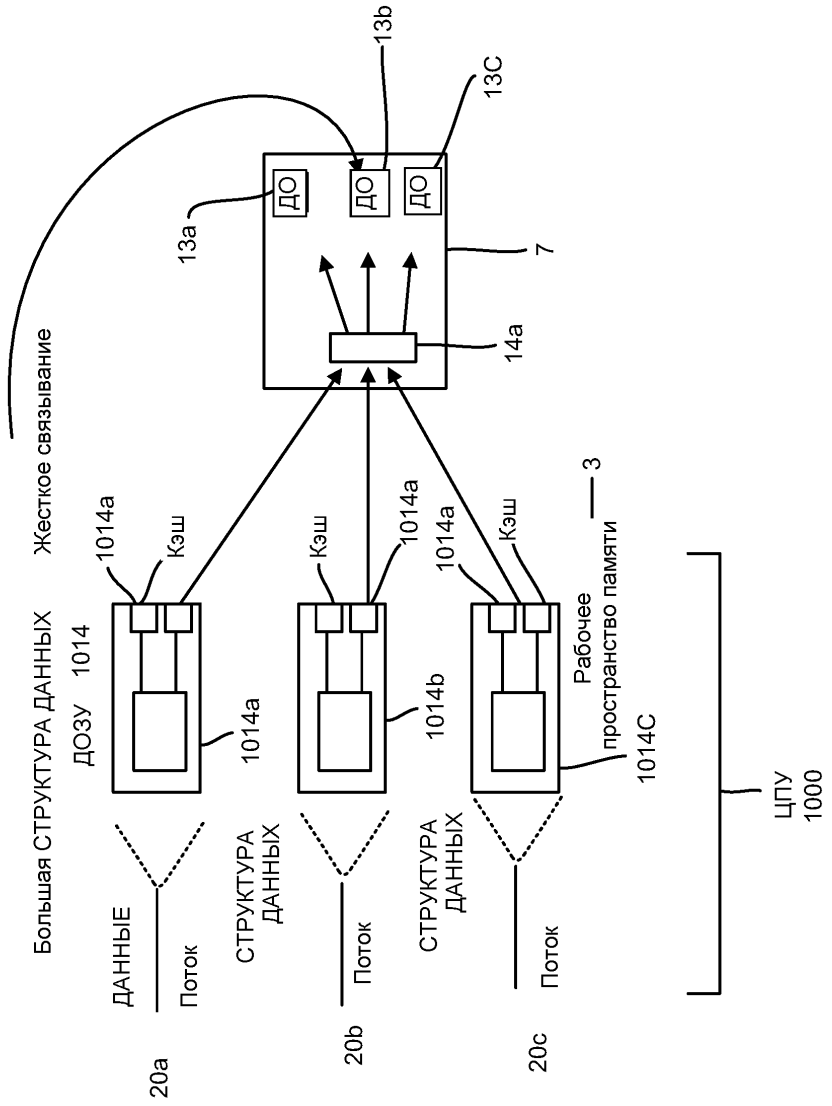
40


7
FPGA
1014
PDRH
L1
ЦПУ
ЦПУ
ЦПУ
КЭШ
ЯДРО
1000a
ЯДРО
1000b
ЯДРО
1000C

ФИГ. 36

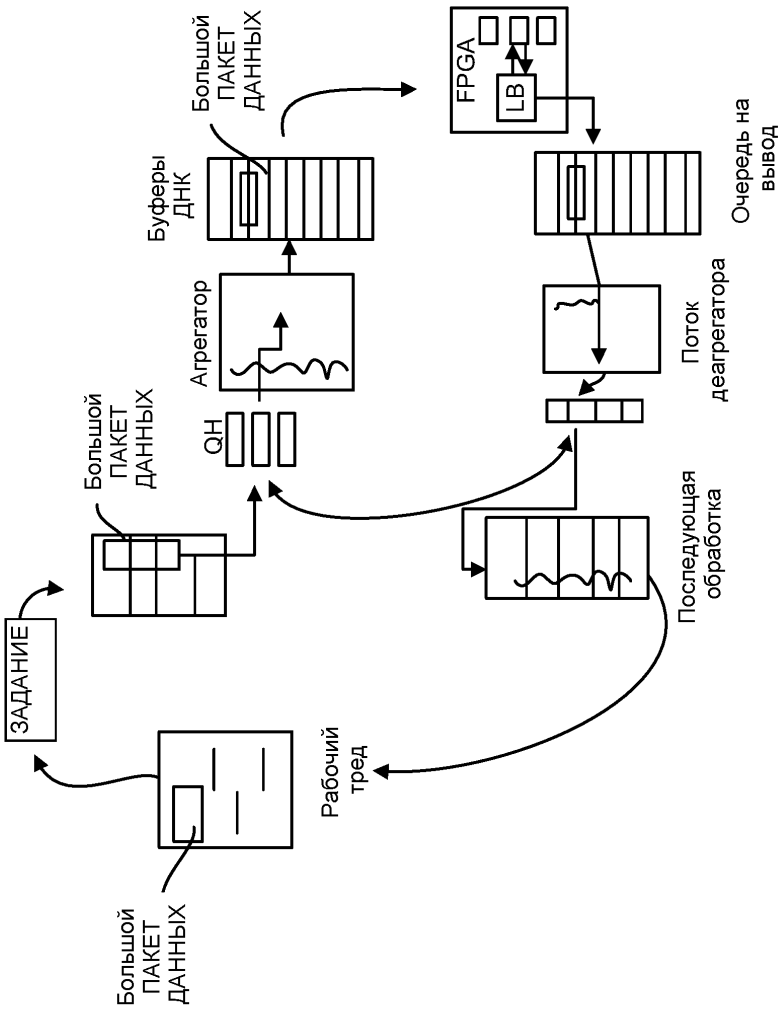

14
FPGA-память
3
7
ДО
ДО
ДО
ДО
4
FPGA
АО
2
К
Э
Ш
ЖЕСТКАЯ
ЦПУ-память
2
1
СТРУКТУРА
ДАННЫХ
КЭШ
ЦПУ ПО
1014
1000

ФИГ. 37

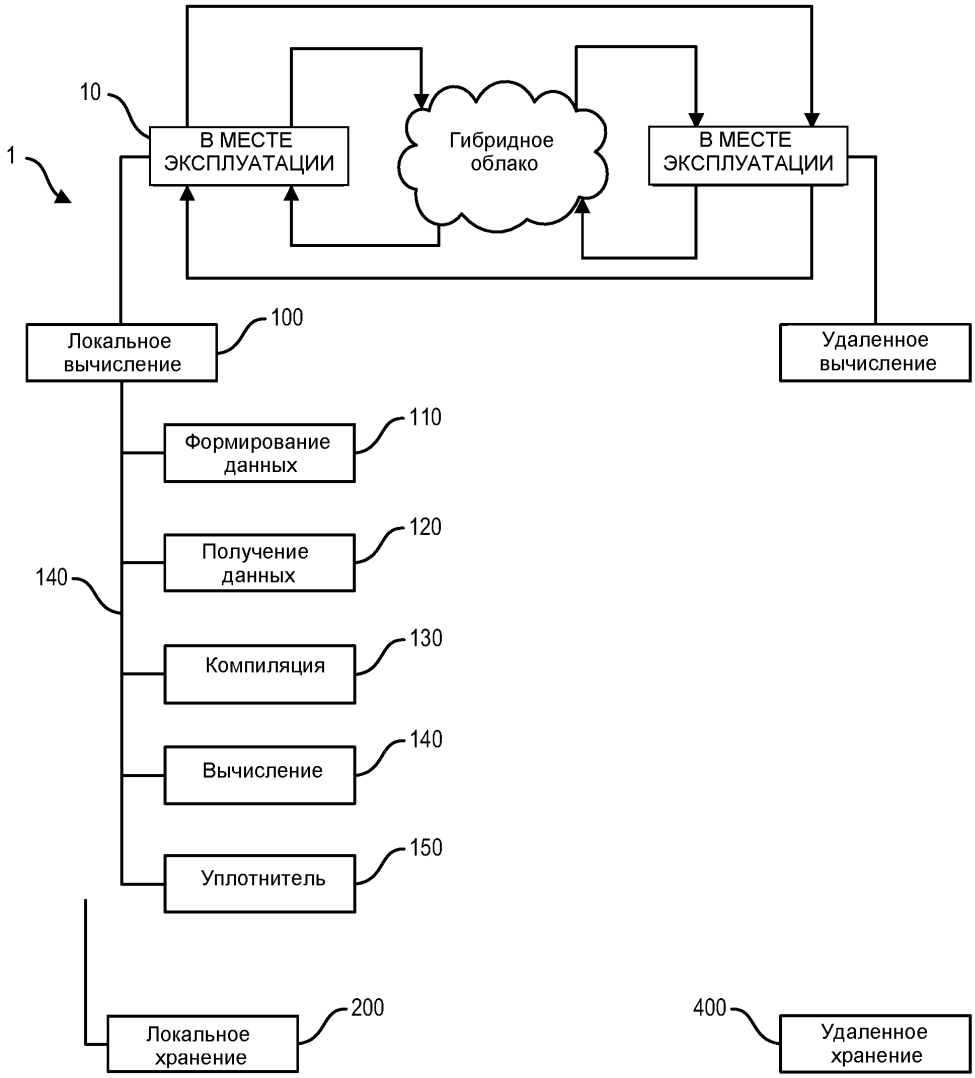

13b
13C
13a
ДО
ДО
ДО
7
14a
Жесткое связывание
Большая СТРУКТУРА ДАННЫХ
ДОЗу 1014
1014a
Кэш
1014a
Кэш
1014a
Кэш
3
Рабочее пространство памяти
1014a
1014b
1014C
ЦПУ 1000
ДАННЫЕ
Поток
20a
СТРУКТУРА ДАННЫХ
Поток
20b
СТРУКТУРА ДАННЫХ
Поток
20c

ФИГ. 38

43

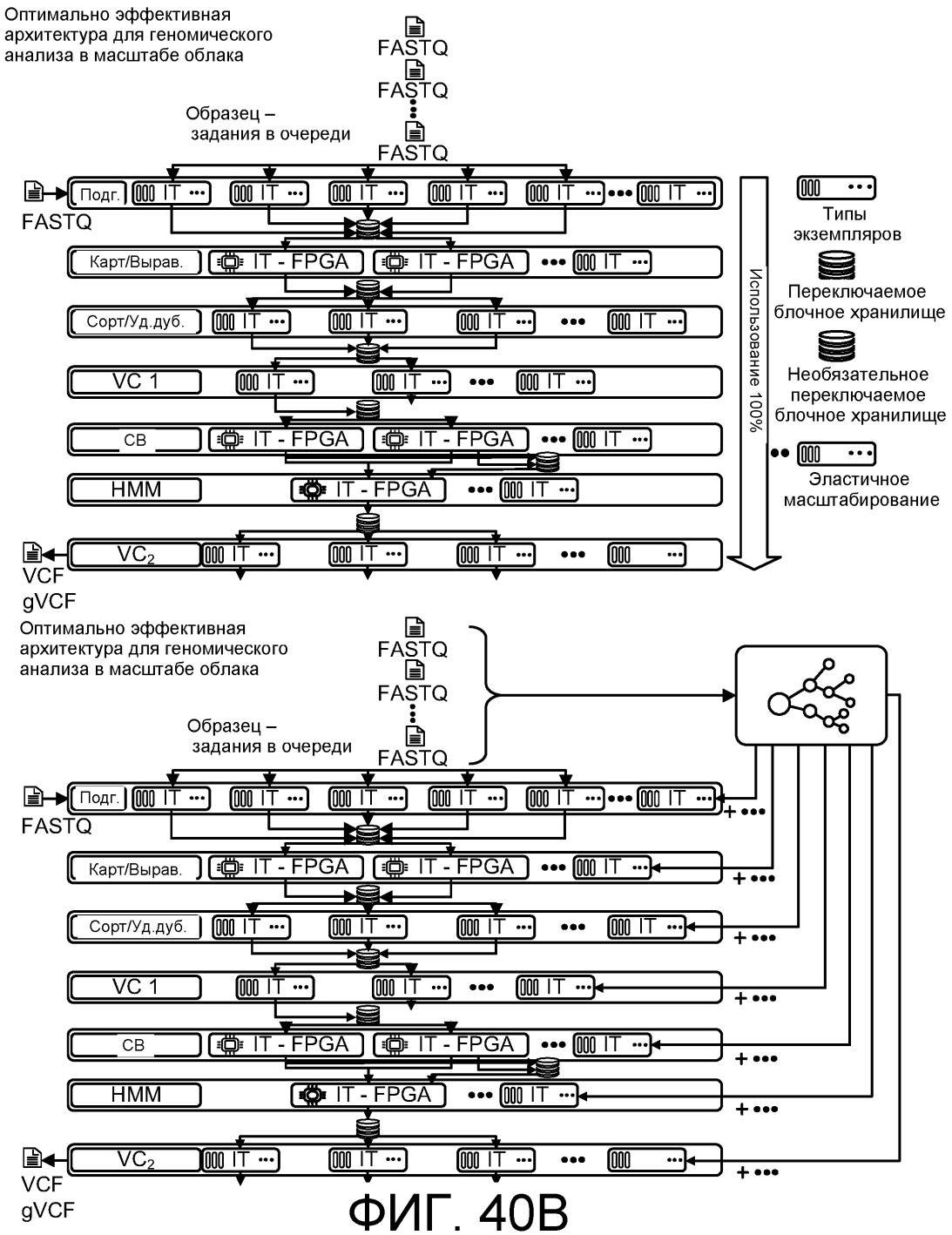

ЗАДАНИЕ
Большой ПАКЕТ ДАННЫХ
Рабочий тред
Большой ПАКЕТ ДАННЫХ
QH
Агрегатор
Буферы ДНК
Большой ПАКЕТ ДАННЫХ
FPGA
LB
Очередь на вывод
Поток деагрегатора
Последующая обработка

ФИГ. 39

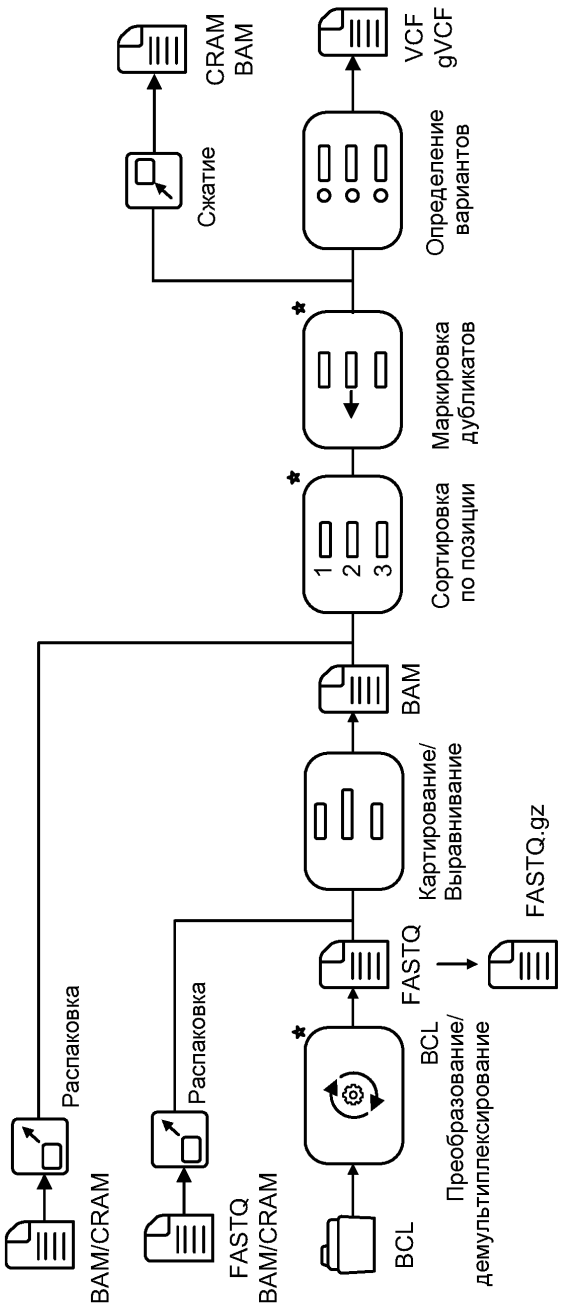

10
1
В МЕСТЕ ЭКСПЛУАТАЦИИ
Гибридное облако
В МЕСТЕ ЭКСПЛУАТАЦИИ
Локальное вычисление
100
Удаленное вычисление
Формирование данных
110
Получение данных
120
140
Компиляция
130
Вычисление
140
Уплотнитель
150
Локальное хранение
200
400
Удаленное хранение

ФИГ. 40А

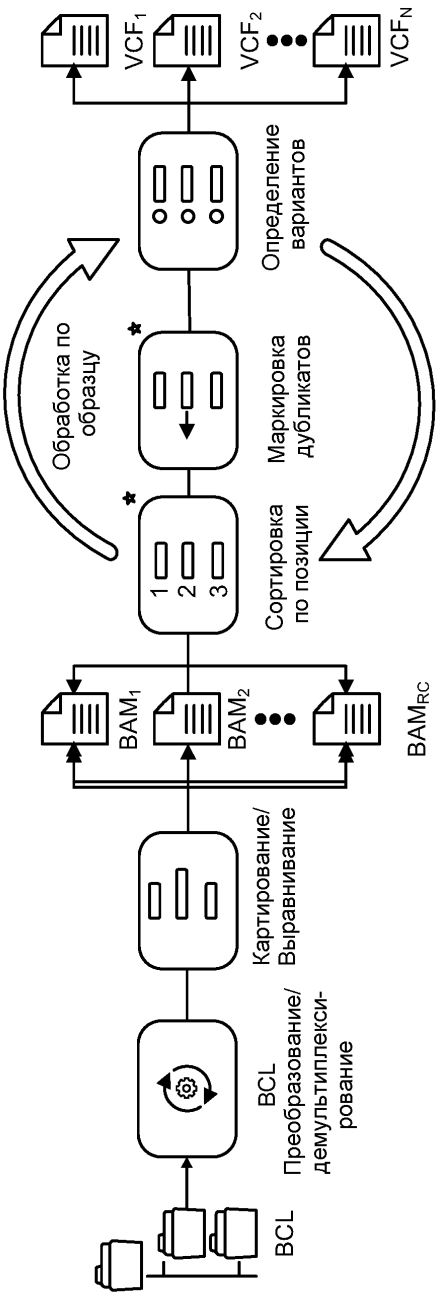
45


Оптимально эффективная архитектура для геномического анализа в масштабе облака
FASTQ
FASTQ
FASTQ
Образец – задания в очереди
Подг.
IT
FASTQ
Карт/Вырав.
IT - FPGA
Сорт/Уд.дуб.
VC 1
CB
HMM
VC2
VCF
gVCF
Использование 100%
Типы экземпляров
Переключаемое блочное хранилище
Необязательное переключаемое блочное хранилище
Эластичное масштабирование
Оптимально эффективная архитектура для геномического анализа в масштабе облака
FASTQ
FASTQ
FASTQ
Образец – задания в очереди
Подг.
IT
FASTQ
Карт/Вырав.
IT - FPGA
Сорт/Уд.дуб.
VC 1
CB
HMM
VC2
VCF
gVCF

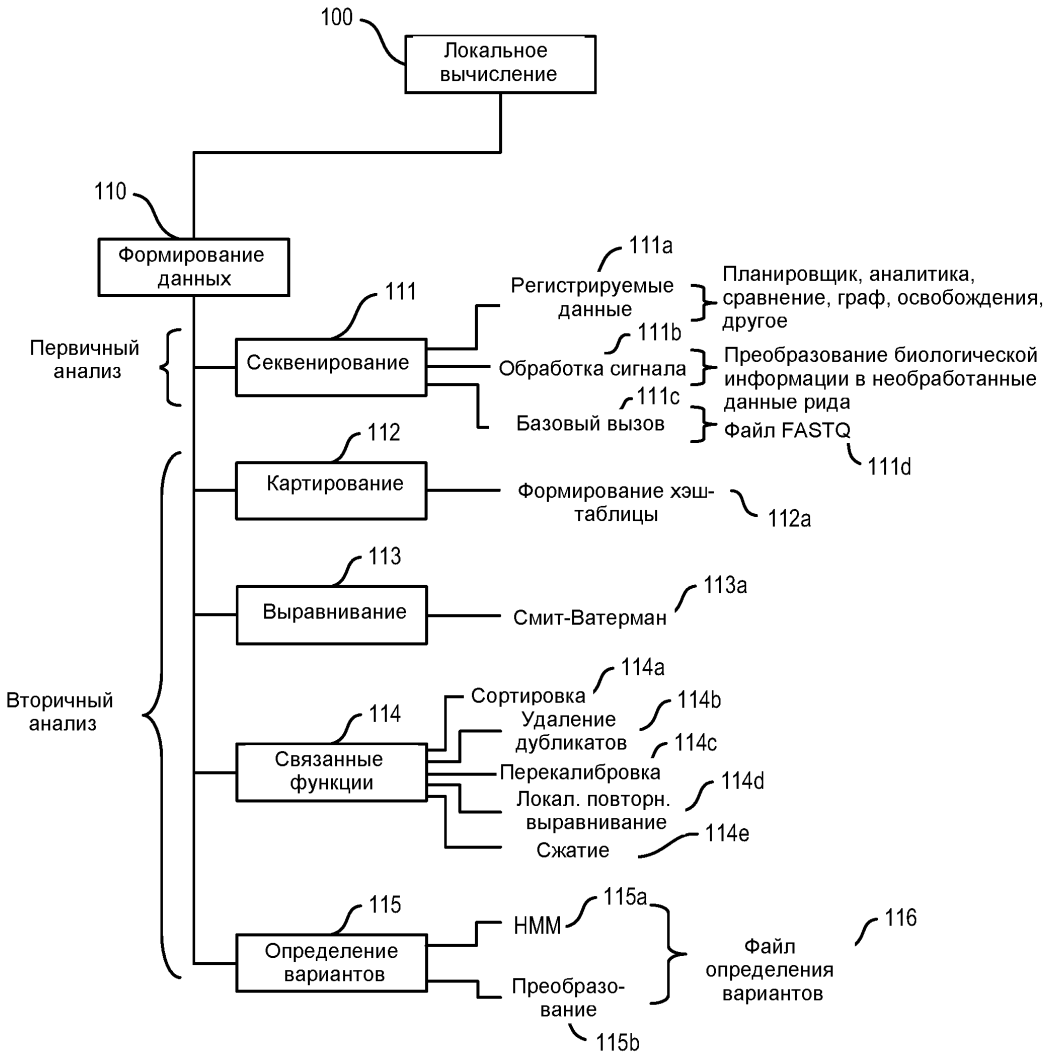
ФИГ. 40B


BAM/CRAM
Распаковка
FASTQ
BAM/CRAM
Распаковка
BCL
Преобразование/
демультиплексирование
FASTQ
FASTQ.gz
Картирование/
Выравнивание
BAM
1
2
3
Сортировка
по позиции
Маркировка
дубликатов
Определение
вариантов
VCF
gVCF
Сжатие
CRAM
BAM

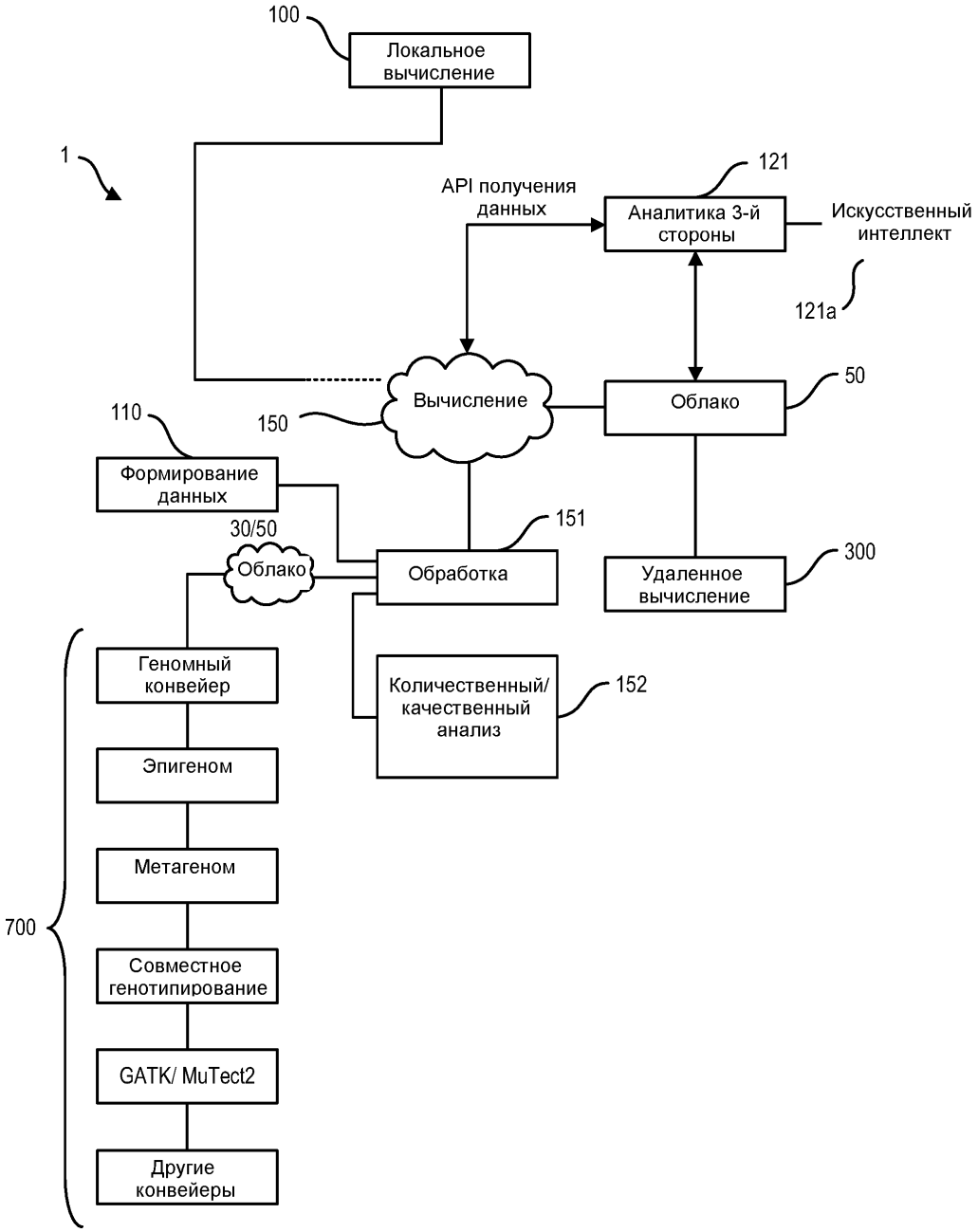
ФИГ. 40С


BCL
BCL
Преобразование/
демультиплекси-
рование
Картирование/
Выравнивание
BAM1
BAM2
BAMRC
1
2
3
Сортировка
по позиции
Маркировка
дубликатов
Определение
вариантов
VCF1
VCF2
VCFN
Обработка по
образцу

ФИГ. 40D

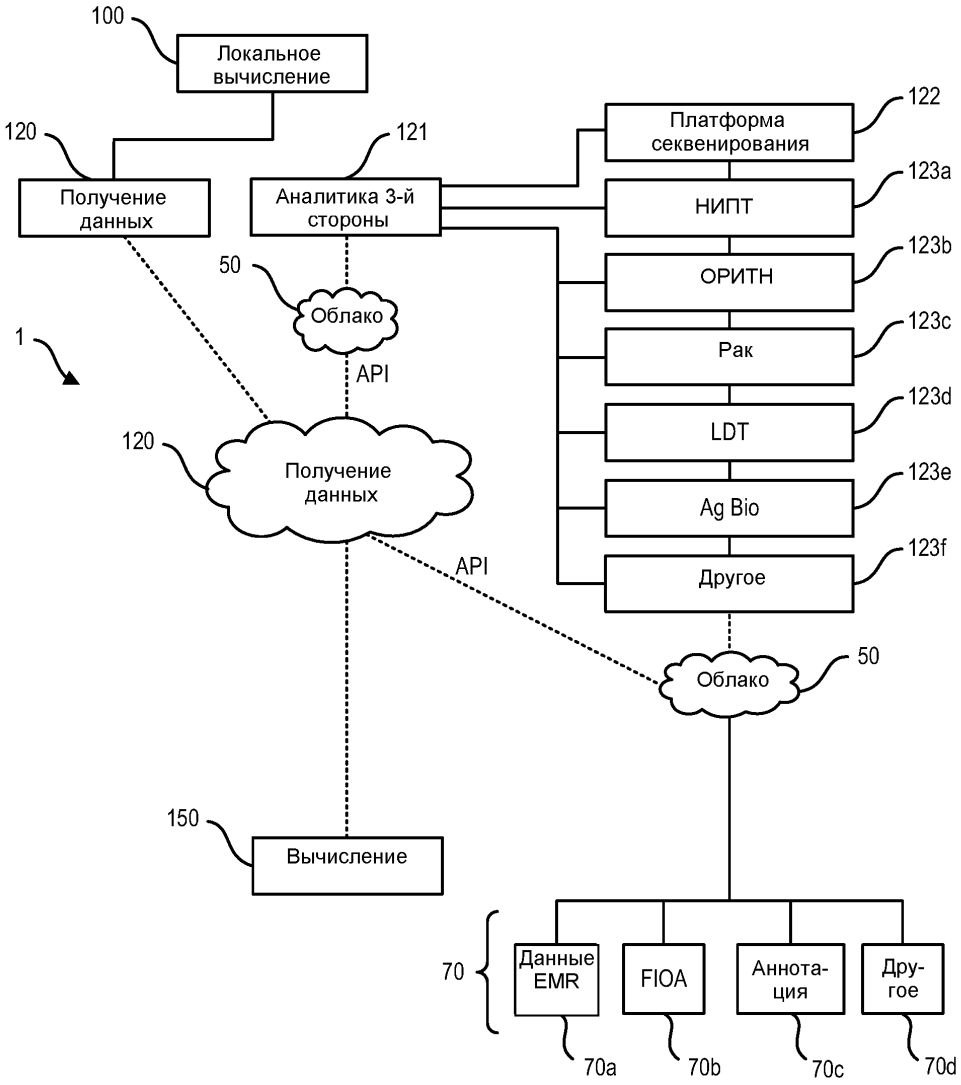

100
Локальное вычисление
110
Формирование данных
Первичный анализ
111
Секвенирование
111a
Регистрируемые данные
Планировщик, аналитика, сравнение, граф, освобождения, другое
111b
Обработка сигнала
Преобразование биологической информации в необработанные данные рида
111c
Базовый вызов
Файл FASTQ
111d
112
Картирование
Формирование хэш-таблицы
112a
113
Выравнивание
Смит-Ватерман
113a
Вторичный анализ
114
Связанные функции
114a
Сортировка
114b
Удаление дубликатов
114c
Перекалибровка
114d
Локал. повторн. выравнивание
114e
Сжатие
115
Определение вариантов
115a
HMM
Преобразо-вание
115b
116
Файл определения вариантов

ФИГ. 41A


100
Локальное вычисление
1
API получения данных
121
Аналитика 3-й стороны
Искусственный интеллект
121a
Вычисление
Облако
50
110
150
Формирование данных
30/50
Облако
151
Обработка
Удаленное вычисление
300
Геномный конвейер
Количественный/ качественный анализ
152
Эпигеном
Метагеном
700
Совместное генотипирование
GATK/ MuTect2
Другие конвейеры

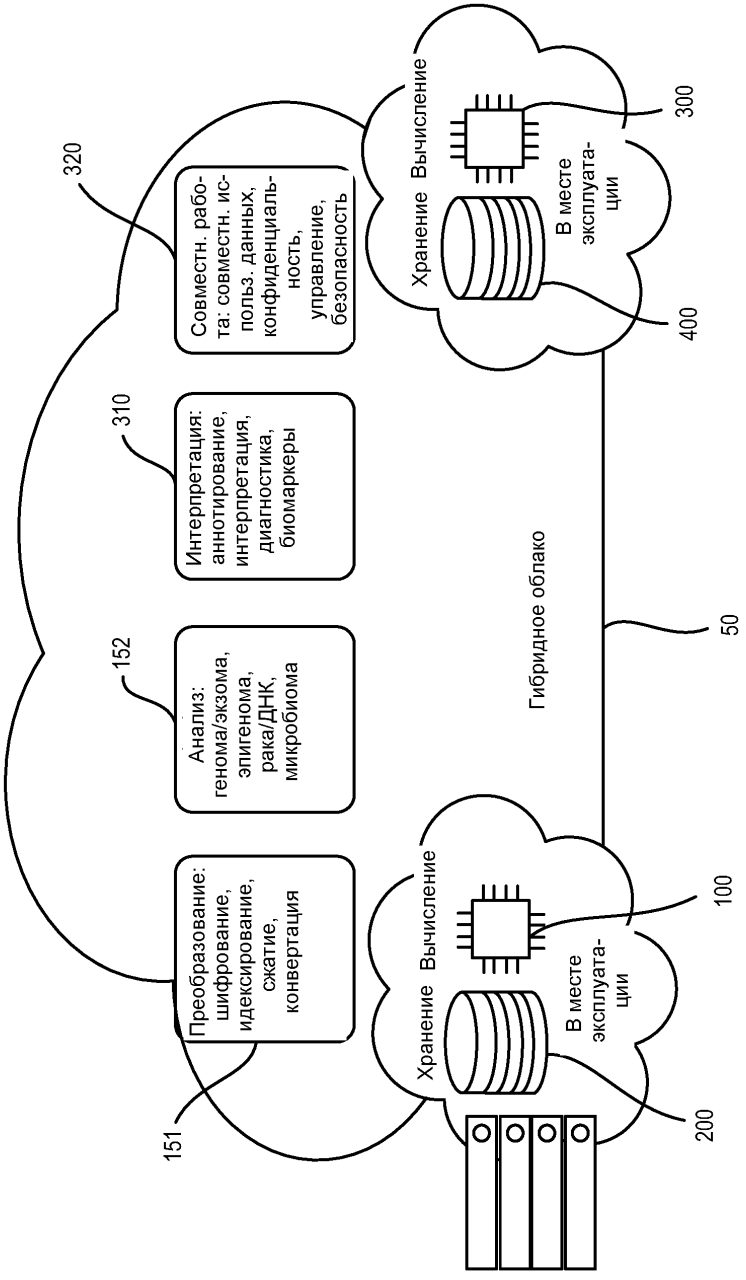
ФИГ. 41B

50


100
Локальное вычисление
120
Получение данных
121
Аналитика 3-й стороны
122
Платформа секвенирования
123a
НИПТ
123b
ОРИТН
123c
Рак
123d
LDT
123e
Ag Bio
123f
Другое
50
Облако
1
API
120
Получение данных
API
50
Облако
150
Вычисление
70
Данные EMR
FIOA
Аннота-ция
Дру-гое
70a
70b
70c
70d

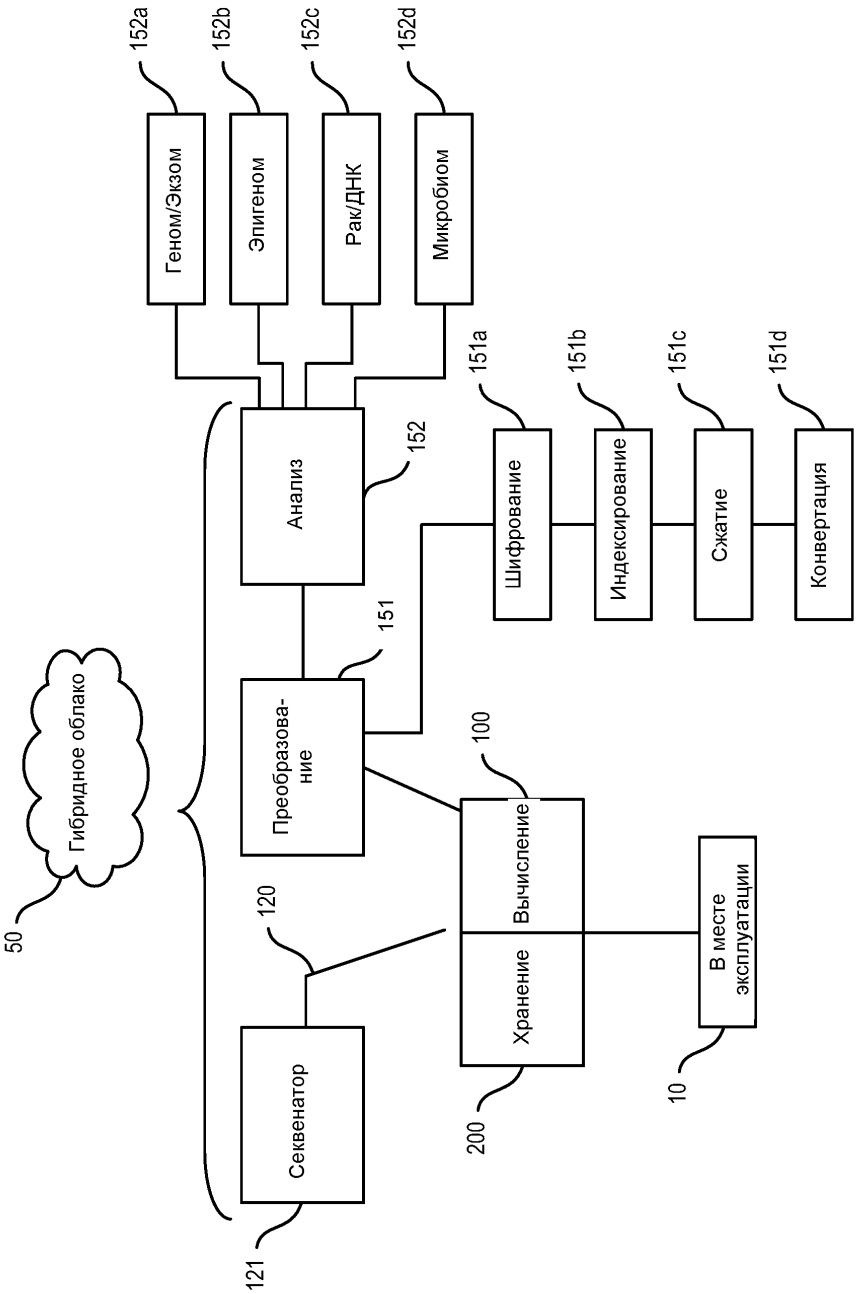
ФИГ. 41C

51

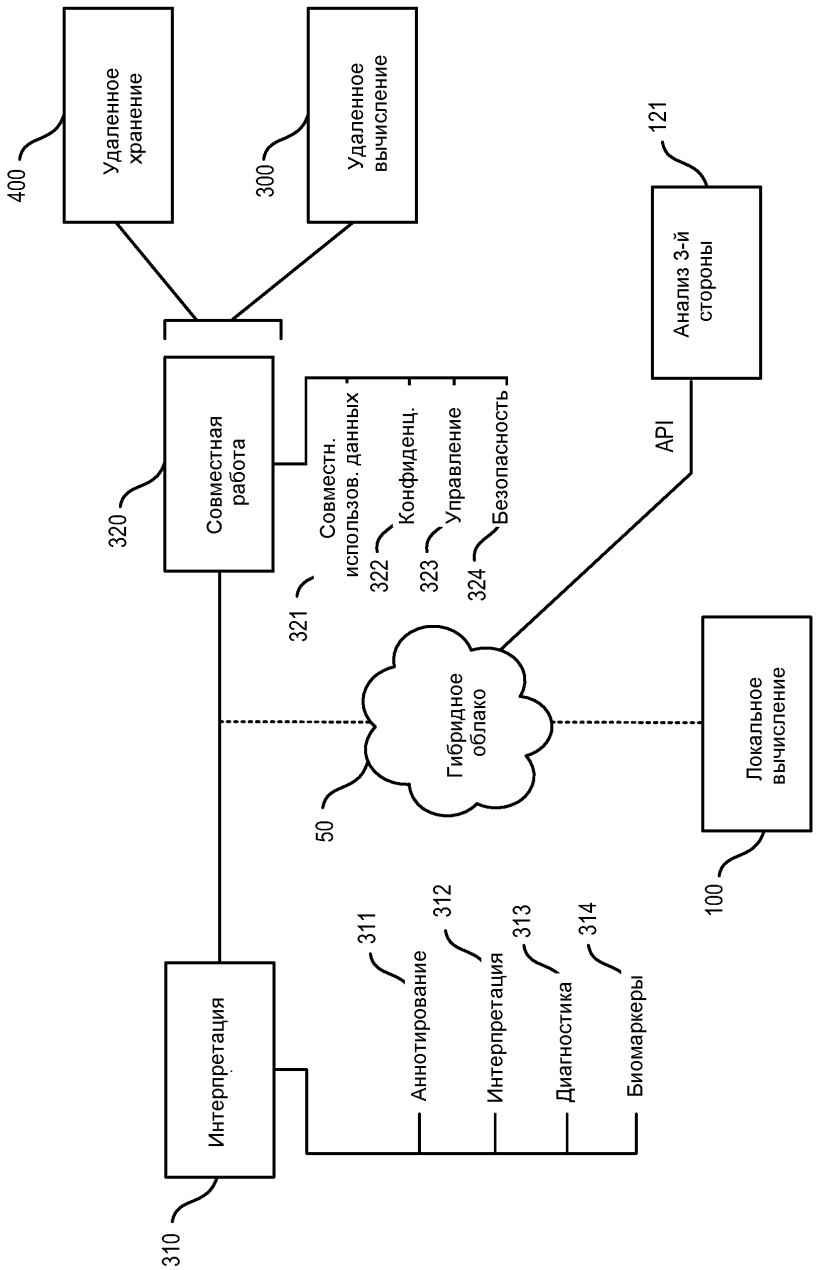

320
Совместн. работа: совместн. использ. данных, конфиденциальность, управление, безопасность
310
Интерпретация: аннотирование, интерпретация, диагностика, биомаркеры
152
Анализ: генома/экзома, эпигенома, рака/ДНК, микробиома
Преобразование: шифрование, идексирование, сжатие, конвертация
151
Хранение
Вычисление
В месте эксплуатации
200
100
Гибридное облако
50
Хранение
Вычисление
В месте эксплуатации
400
300

ФИГ. 42А

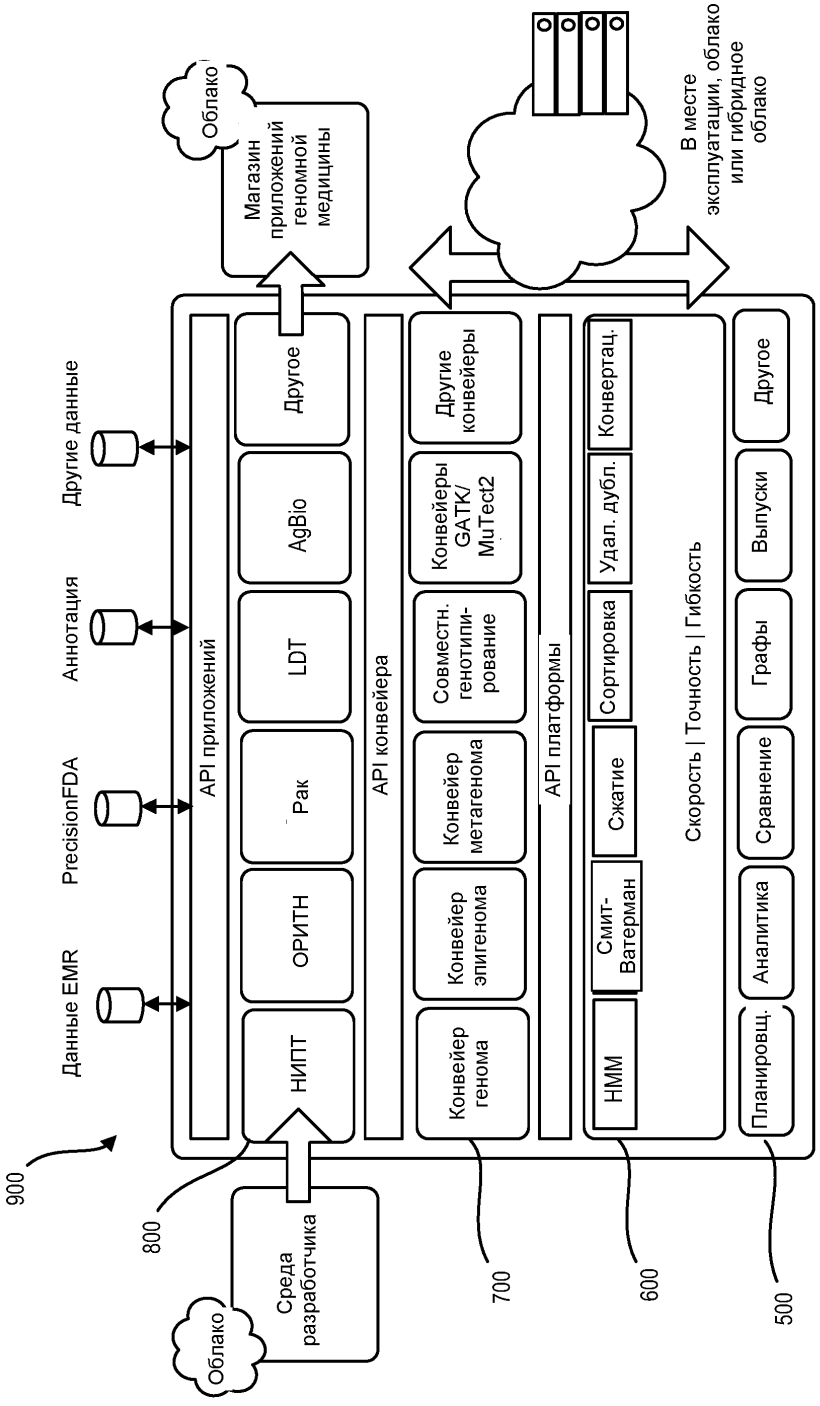
52


152a
152b
152c
152d
Геном/Экзом
Эпигеном
Рак/ДНК
Микробиом
151a
151b
151c
151d
Шифрование
Индексирование
Сжатие
Конвертация
Анализ
152
151
Преобразова-
ние
100
Вычисление
Хранение
200
120
Секвенатор
121
50
Гибридное облако
В месте
эксплуатации
10

ФИГ. 42B

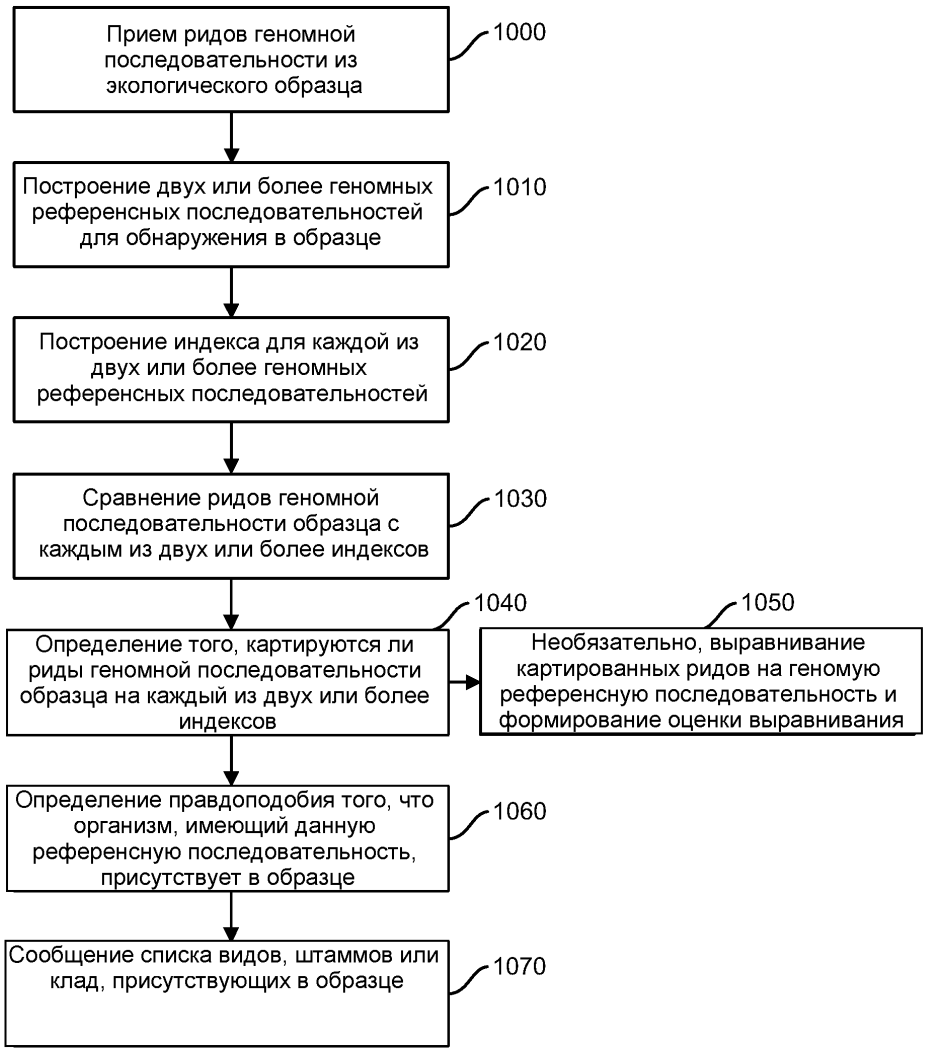

Удаленное хранение
400
Удаленное вычисление
300
121
Анализ 3-й стороны
Совместная работа
320
Совместн. использов. данных
321
Конфиденц.
322
Управление
323
Безопасность
324
API
Гибридное облако
50
Локальное вычисление
100
Интерпретация
310
311
Аннотирование
312
Интерпретация
313
Диагностика
314
Биомаркеры

ФИГ. 42С

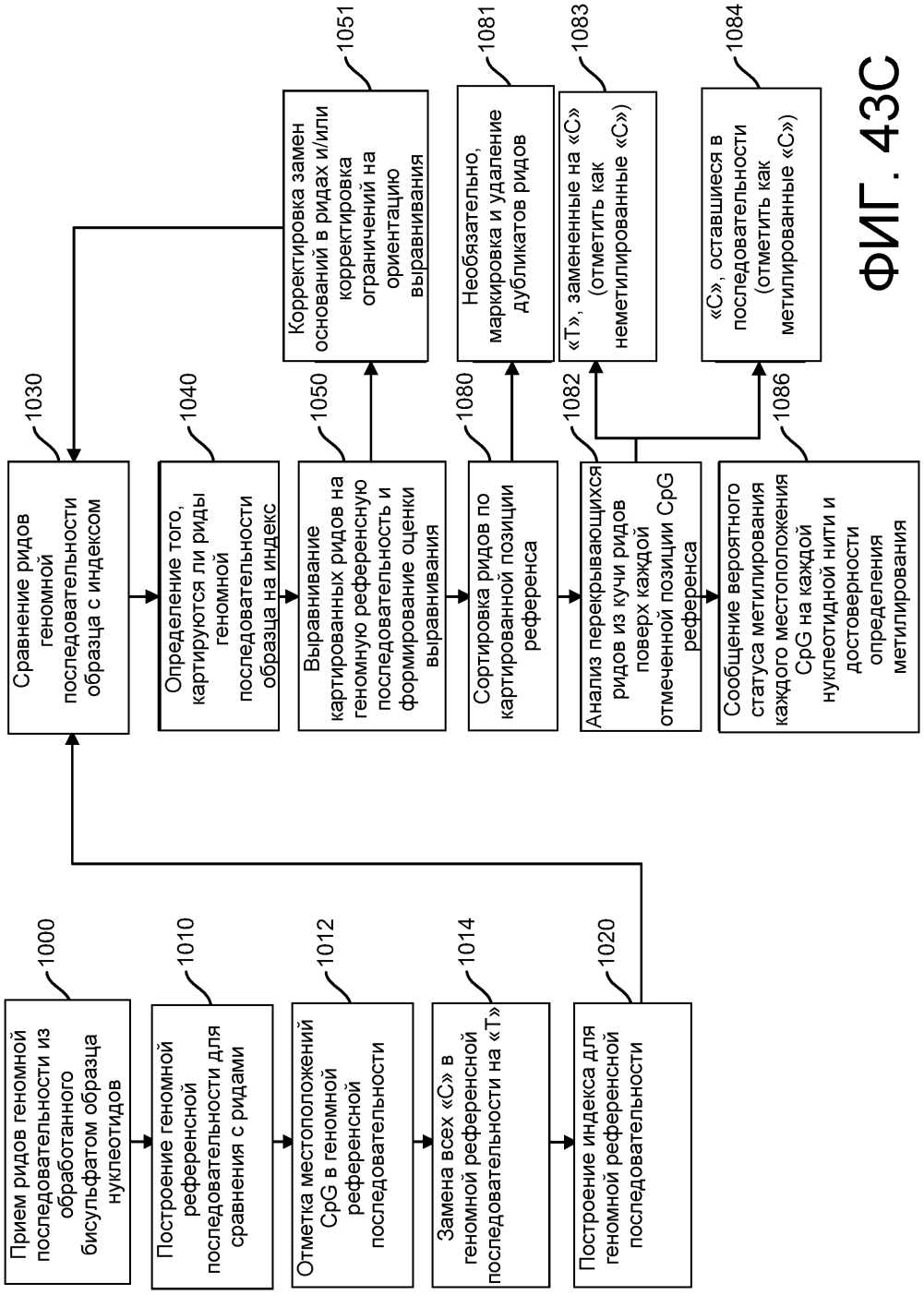

Облако
Магазин приложений геномной медицины
В месте эксплуатации, облако или гибридное облако
Другие данные
Аннотация
PrecisionFDA
Данные EMR
API приложений
Другое
AgBio
LDT
Рак
ОРИТН
НИПТ
API конвейера
Другие конвейеры
Конвейеры GATK/ MuTect2
Совместн. генотипи-рование
Конвейер метагенома
Конвейер эпигенома
Конвейер генома
API платформы
Конвертац.
Удал. дубл.
Сортировка
Сжатие
Смит-Ватерман
HMM
Скорость | Точность | Гибкость
Другое
Выпуски
Графы
Сравнение
Аналитика
Планировщ.
900
800
700
600
500
Облако
Среда разработчика

ФИГ. 43А

55

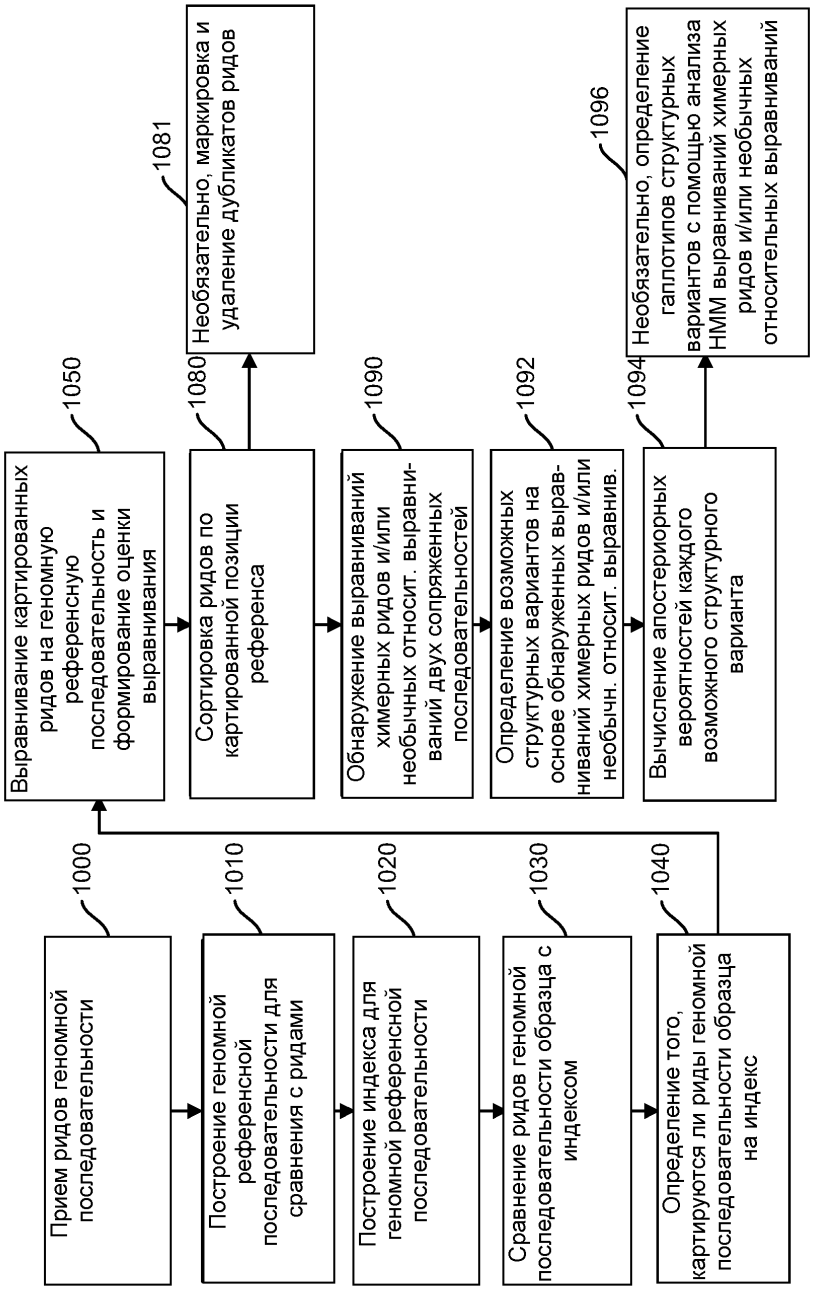

Прием ридов геномной последовательности из экологического образца
1000
Построение двух или более геномных референсных последовательностей для обнаружения в образце
1010
Построение индекса для каждой из двух или более геномных референсных последовательностей
1020
Сравнение ридов геномной последовательности образца с каждым из двух или более индексов
1030
Определение того, картируются ли риды геномной последовательности образца на каждый из двух или более индексов
1040
Необязательно, выравнивание картированных ридов на геномую референсную последовательность и формирование оценки выравнивания
1050
Определение правдоподобия того, что организм, имеющий данную референсную последовательность, присутствует в образце
1060
Сообщение списка видов, штаммов или клад, присутствующих в образце
1070

ФИГ. 43B


1000
Прием ридов геномной последовательности из обработанного бисульфатом образца нуклеотидов
1010
Построение геномной референсной последовательности для сравнения с ридами
1012
Отметка местоположений CpG в геномной референсной последовательности
1014
Замена всех «C» в геномной референсной последовательности на «T»
1020
Построение индекса для геномной референсной последовательности
1030
Сравнение ридов геномной последовательности образца с индексом
1040
Определение того, картируются ли риды геномной последовательности образца на индекс
1050
Выравнивание картированных ридов на геномную референсную последовательность и формирование оценки выравнивания
1051
Корректировка замен оснований в ридах и/или корректировка ограничений на ориентацию выравнивания
1080
Сортировка ридов по картированной позиции референса
1081
Необязательно, маркировка и удаление дубликатов ридов
1082
Анализ перекрывающихся ридов из кучи ридов поверх каждой отмеченной позиции CpG референса
1083
«T», замененные на «C» (отметить как неметилированные «C»)
1084
«C», оставшиеся в последовательности (отметить как метилированные «C»)
1086
Сообщение вероятного статуса метилирования каждого местоположения CpG на каждой нуклеотидной нити и достоверности определения метилирования

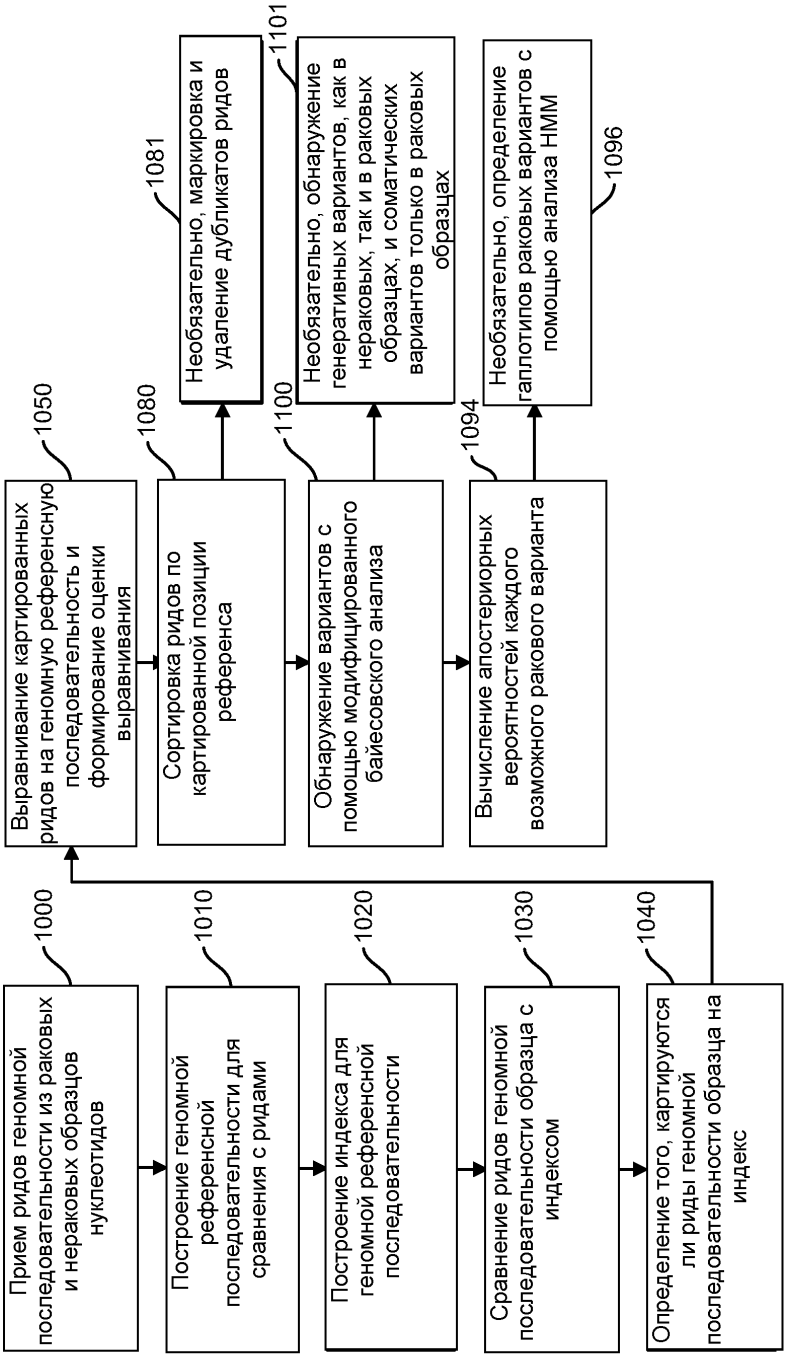
ФИГ. 43C

57

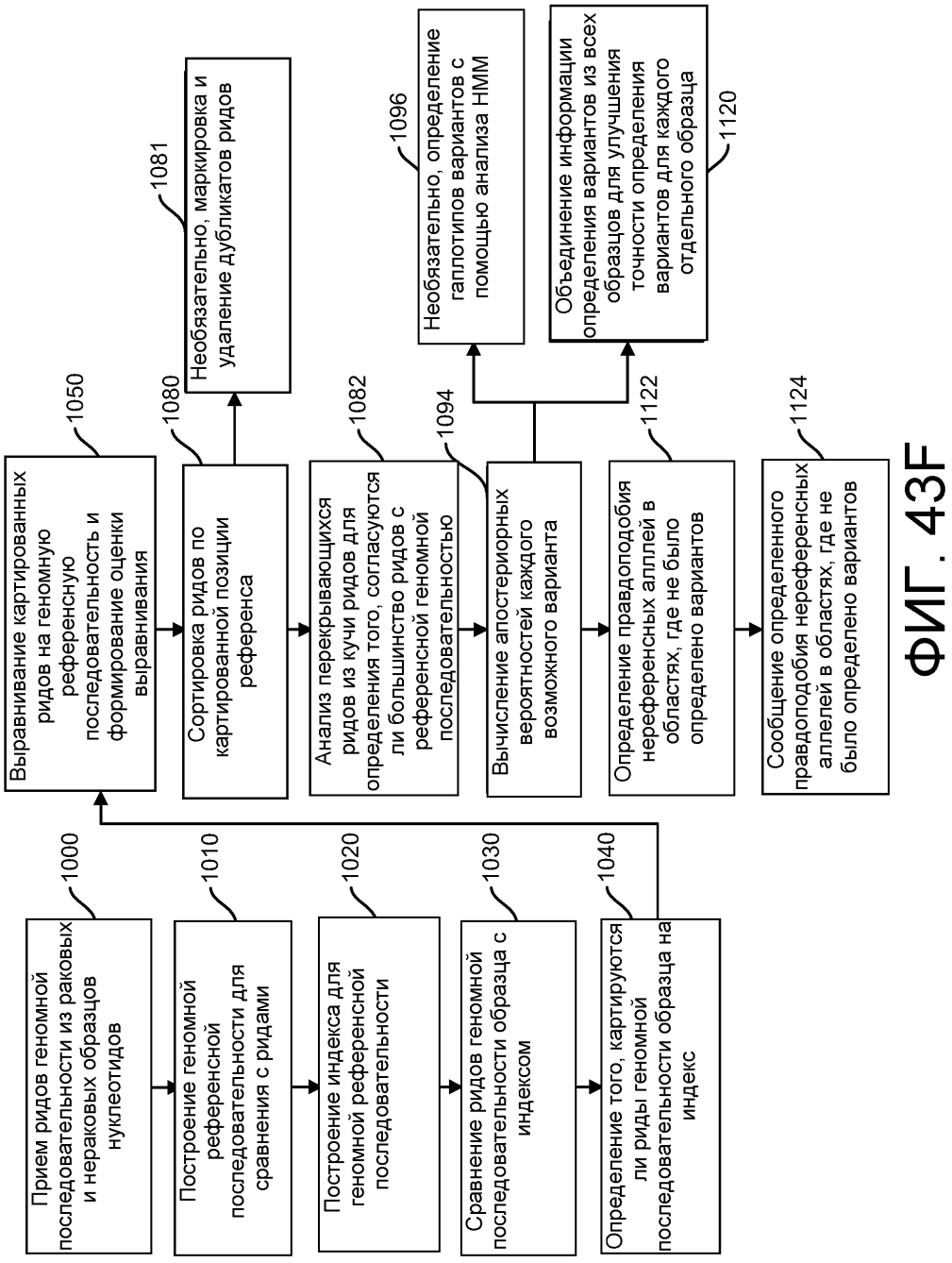

1000
Прием ридов геномной последовательности
1010
Построение геномной референсной последовательности для сравнения с ридами
1020
Построение индекса для геномной референсной последовательности
1030
Сравнение ридов геномной последовательности образца с индексом
1040
Определение того, картируются ли риды геномной последовательности образца на индекс
1050
Выравнивание картированных ридов на геномную референсную последовательность и формирование оценки выравнивания
1080
Сортировка ридов по картированной позиции референса
1081
Необязательно, маркировка и удаление дубликатов ридов
1090
Обнаружение выравниваний химерных ридов и/или необычных относит. выравниваний двух сопряженных последовательностей
1092
Определение возможных структурных вариантов на основе обнаруженных выравниваний химерных ридов и/или необычн. относит. выравнив.
1094
Вычисление апостериорных вероятностей каждого возможного структурного варианта
1096
Необязательно, определение гаплотипов структурных вариантов с помощью анализа HMM выравниваний химерных ридов и/или необычных относительных выравниваний

ФИГ. 43D

58

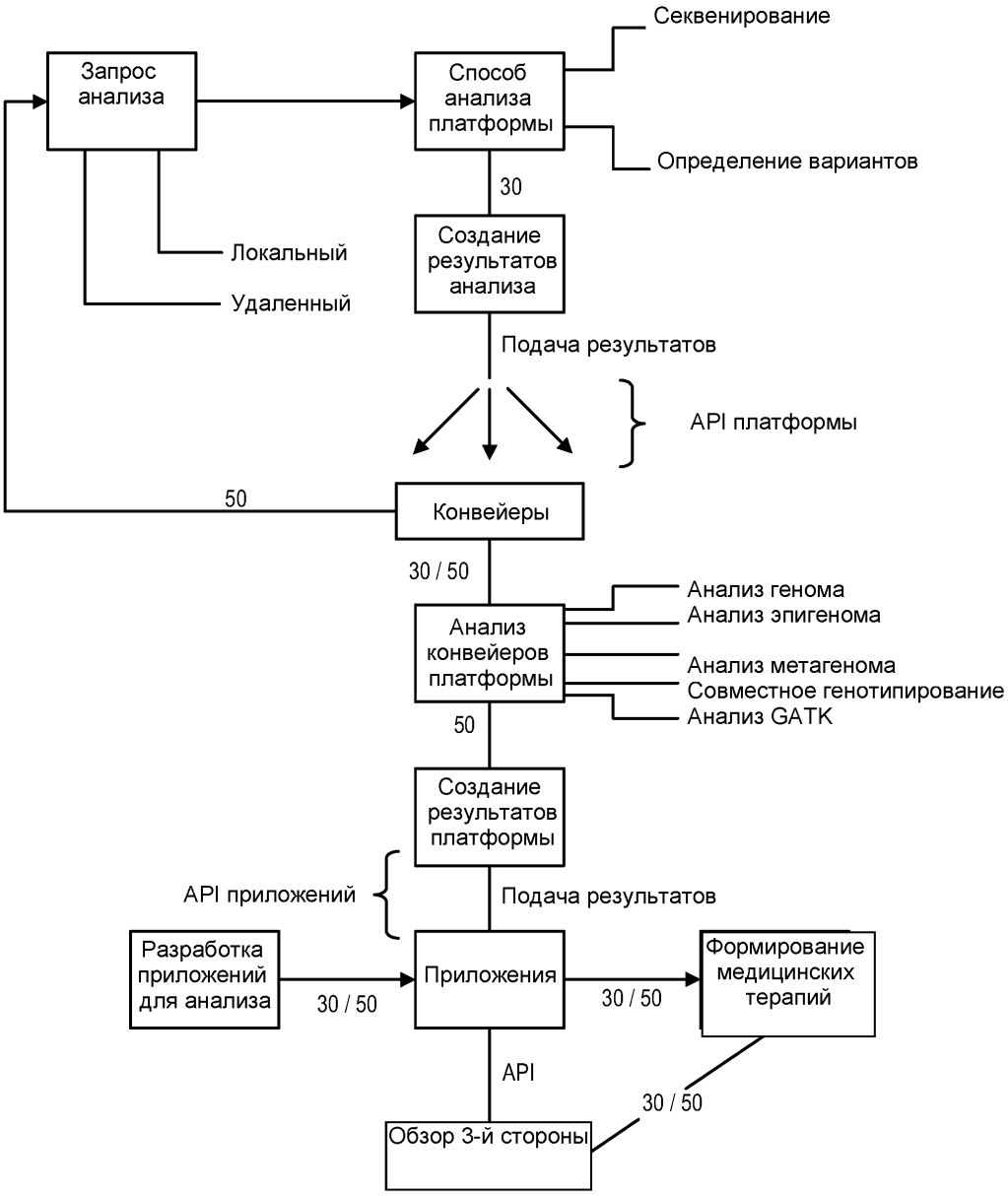

Прием ридов геномной последовательности из раковых и нераковых образцов нуклеотидов
1000
Построение геномной референсной последовательности для сравнения с ридами
1010
Построение индекса для геномной референсной последовательности
1020
Сравнение ридов геномной последовательности образца с индексом
1030
Определение того, картируются ли риды геномной последовательности образца на индекс
1040
Выравнивание картированных ридов на геномную референсную последовательность и формирование оценки выравнивания
1050
Сортировка ридов по картированной позиции референса
1080
Необязательно, маркировка и удаление дубликатов ридов
1081
Обнаружение вариантов с помощью модифицированного байесовского анализа
1100
Необязательно, обнаружение генеративных вариантов, как в нераковых, так и в раковых образцах, и соматических вариантов только в раковых образцах
1101
Вычисление апостериорных вероятностей каждого возможного ракового варианта
1094
Необязательно, определение гаплотипов раковых вариантов с помощью анализа HMM
1096

ФИГ. 43E


1000
Прием ридов геномной последовательности из раковых и нераковых образцов нуклеотидов
1010
Построение геномной референсной последовательности для сравнения с ридами
1020
Построение индекса для геномной референсной последовательности
1030
Сравнение ридов геномной последовательности образца с индексом
1040
Определение того, картируются ли риды геномной последовательности образца на индекс
1050
Выравнивание картированных ридов на геномную референсную последовательность и формирование оценки выравнивания
1080
Сортировка ридов по картированной позиции референса
1081
Необязательно, маркировка и удаление дубликатов ридов
1082
Анализ перекрывающихся ридов из кучи ридов для определения того, согласуются ли большинство ридов с референсной геномной последовательностью
1094
Вычисление апостериорных вероятностей каждого возможного варианта
1096
Необязательно, определение гаплотипов вариантов с помощью анализа HMM
1120
Объединение информации определения вариантов из всех образцов для улучшения точности определения вариантов для каждого отдельного образца
1122
Определение правдоподобия нереференсных аллелей в областях, где не было определено вариантов
1124
Сообщение определенного правдоподобия нереференсных аллелей в областях, где не было определено вариантов

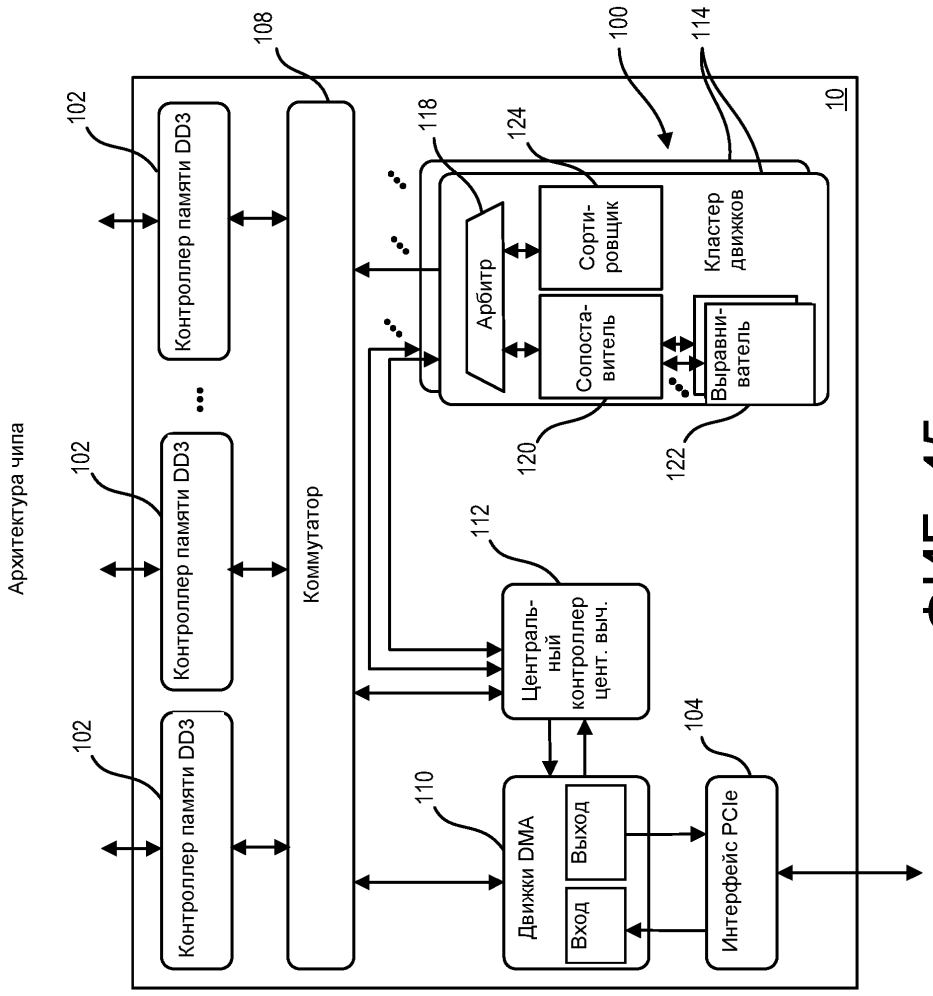
ФИГ. 43F

60


Секвенирование
Запрос анализа
Способ анализа платформы
Определение вариантов
30
Локальный
Создание результатов анализа
Удаленный
Подача результатов
API платформы
50
Конвейеры
30 / 50
Анализ генома
Анализ эпигенома
Анализ конвейеров платформы
Анализ метагенома
Совместное генотипирование
Анализ GATK
50
Создание результатов платформы
API приложений
Подача результатов
Разработка приложений для анализа
30 / 50
Приложения
30 / 50
Формирование медицинских терапий
API
30 / 50
Обзор 3-й стороны

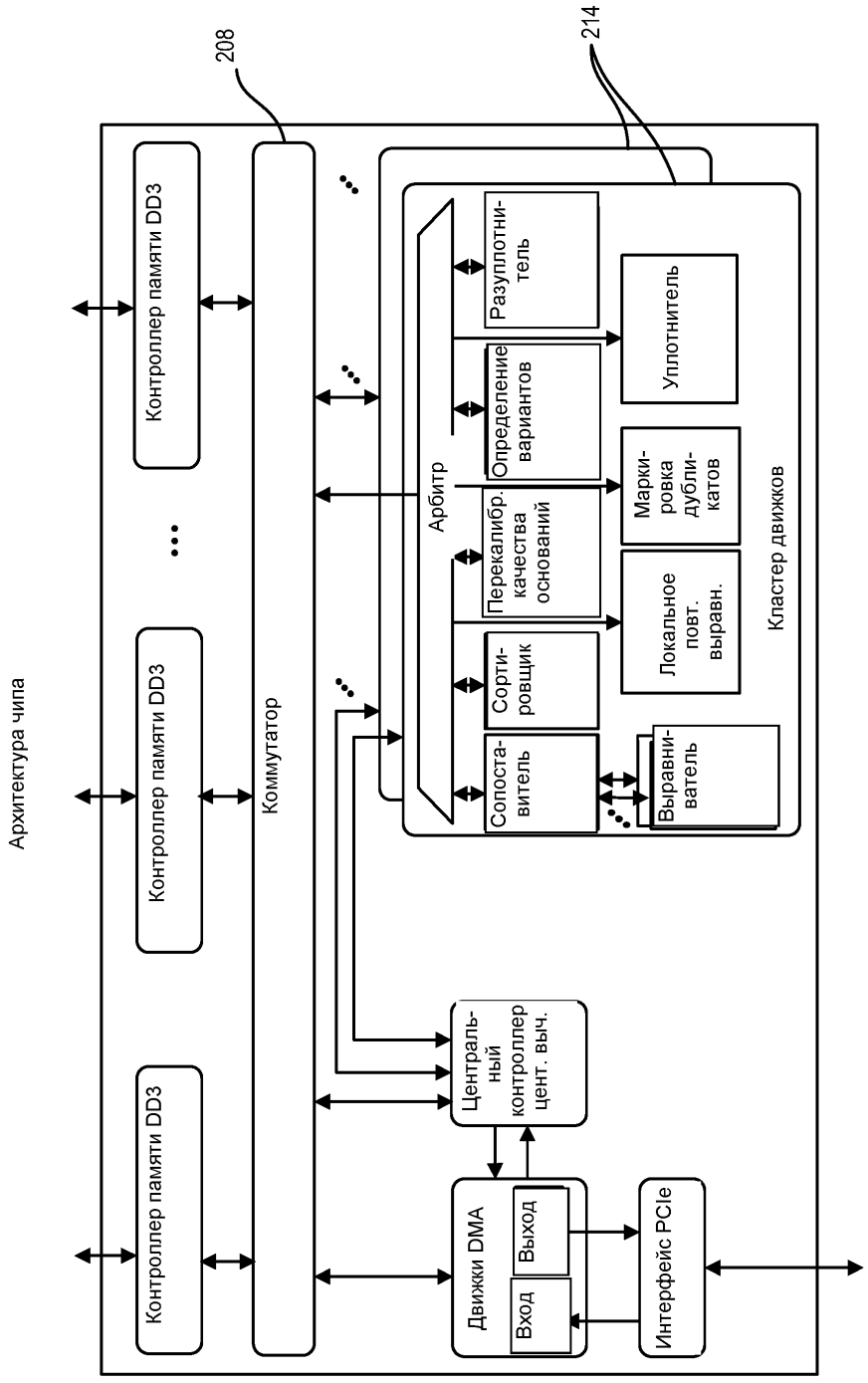
ФИГ. 44

61


Архитектура чипа
10
102
Контроллер памяти DD3
102
Контроллер памяти DD3
102
Контроллер памяти DD3
108
Коммутатор
100
114
118
124
Арбитр
Сорти-
ровщик
Сопоста-
витель
Выравни-
ватель
Кластер
движков
120
122
112
Централь-
ный
контроллер
цент. выч.
110
Движки DMA
Выход
Вход
104
Интерфейс PCIe

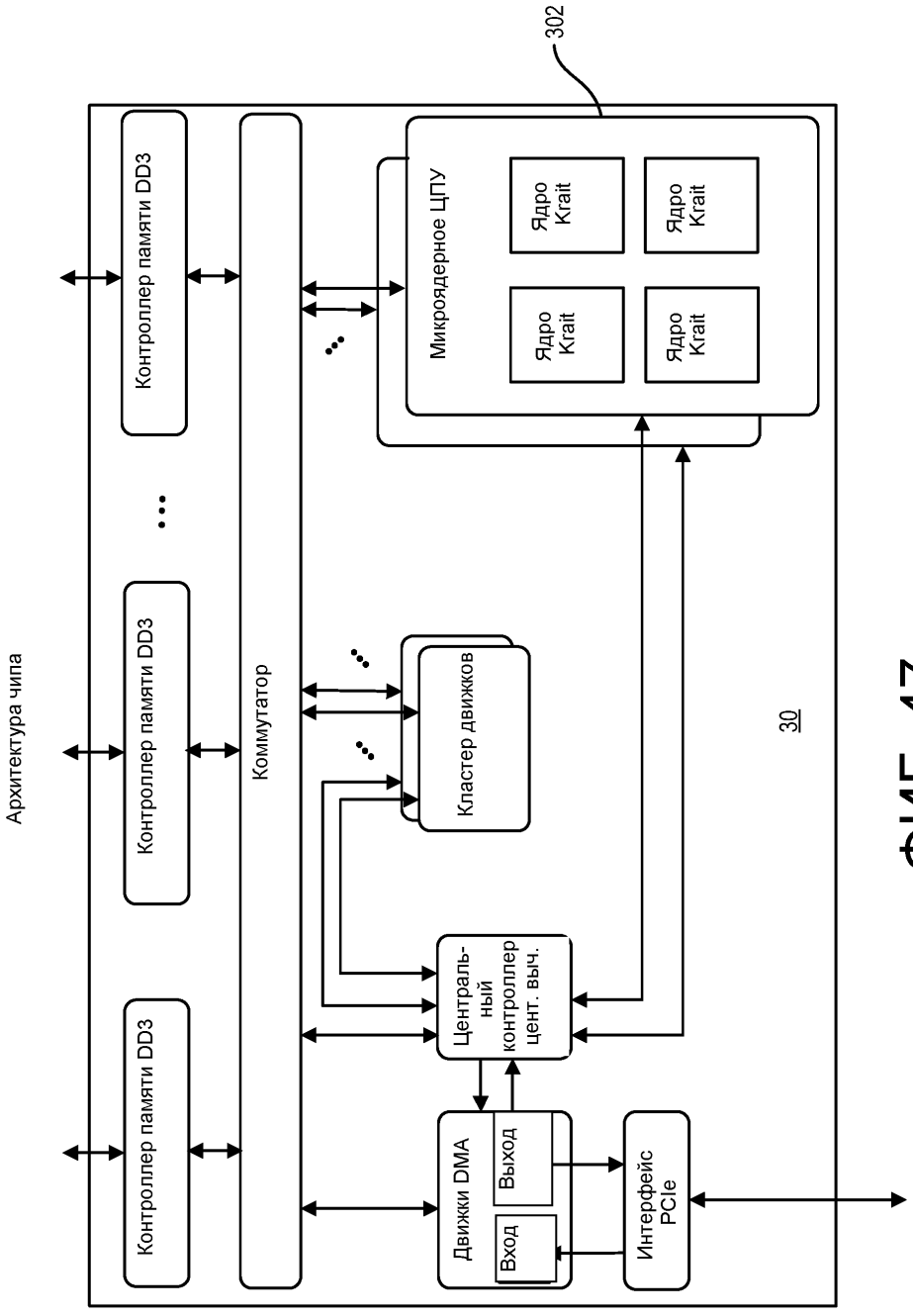
ФИГ. 45

62


Архитектура чипа
208
214
Контроллер памяти DD3
Контроллер памяти DD3
Контроллер памяти DD3
Коммутатор
Арбитр
Разуплотни-тель
Определение вариантов
Перекалибр. качества оснований
Сорти-ровщик
Сопоста-витель
Выравни-ватель
Уплотнитель
Марки-ровка дубли-катов
Локальное повт. выравн.
Кластер движков
Централь-ный контроллер цент. выч.
Движки DMA
Вход
Выход
Интерфейс PCIe

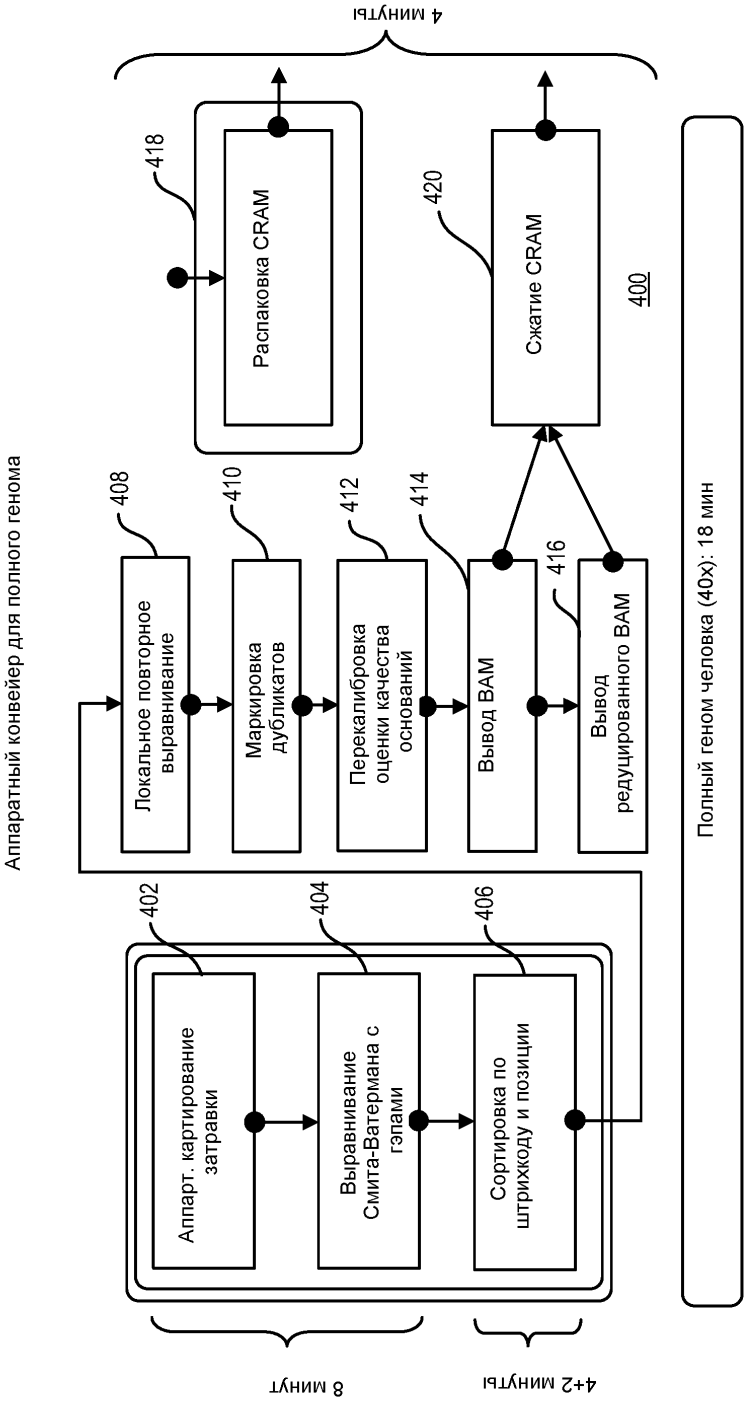
ФИГ. 46

63

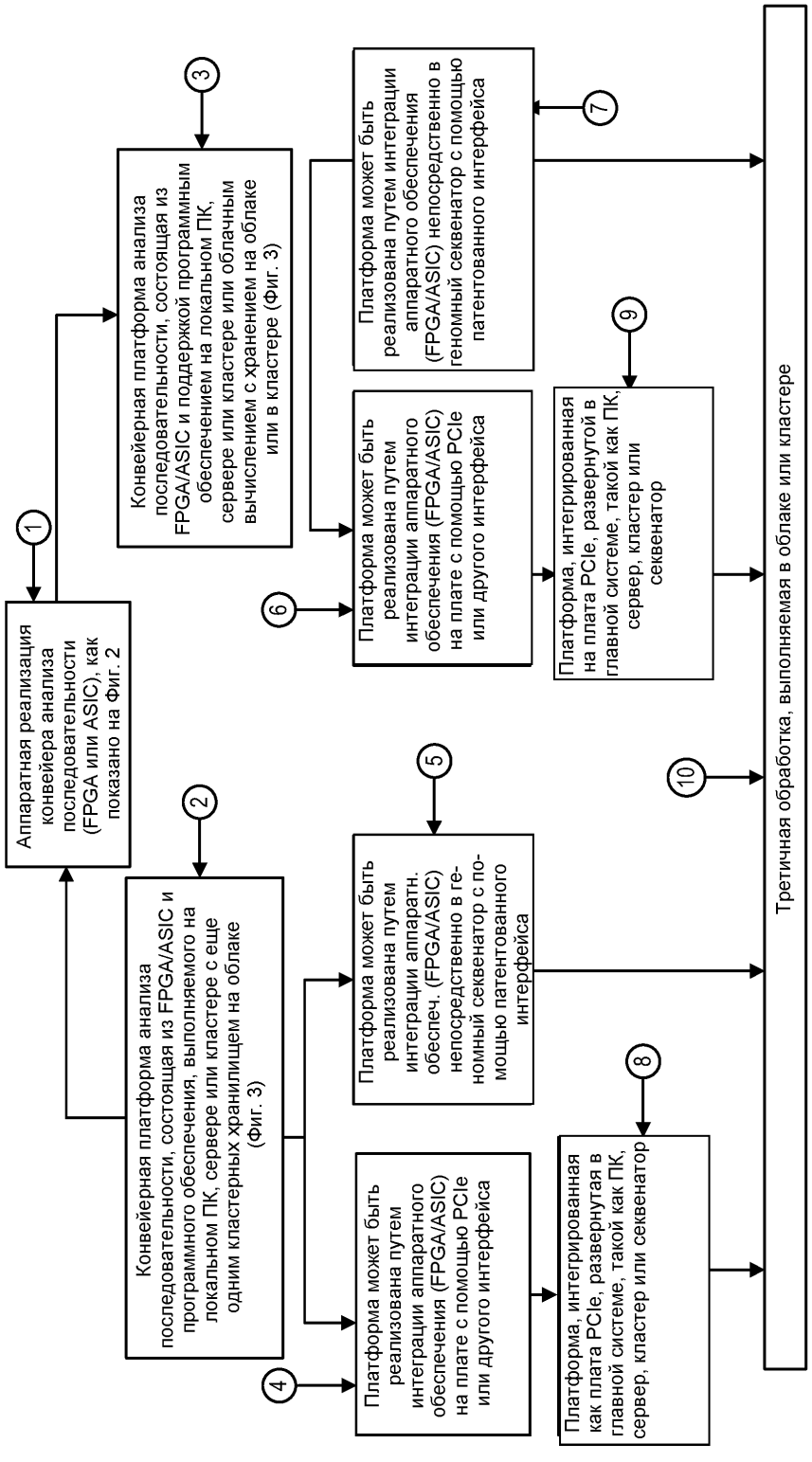

Архитектура чипа
Контроллер памяти DD3
Контроллер памяти DD3
Контроллер памяти DD3
Коммутатор
Кластер движков
Централь-ный контроллер цент. выч.
Движки DMA
Выход
Вход
Интерфейс PCIe
Микроядерное ЦПУ
Ядро Krait
Ядро Krait
Ядро Krait
Ядро Krait
302
30

ФИГ. 47

64

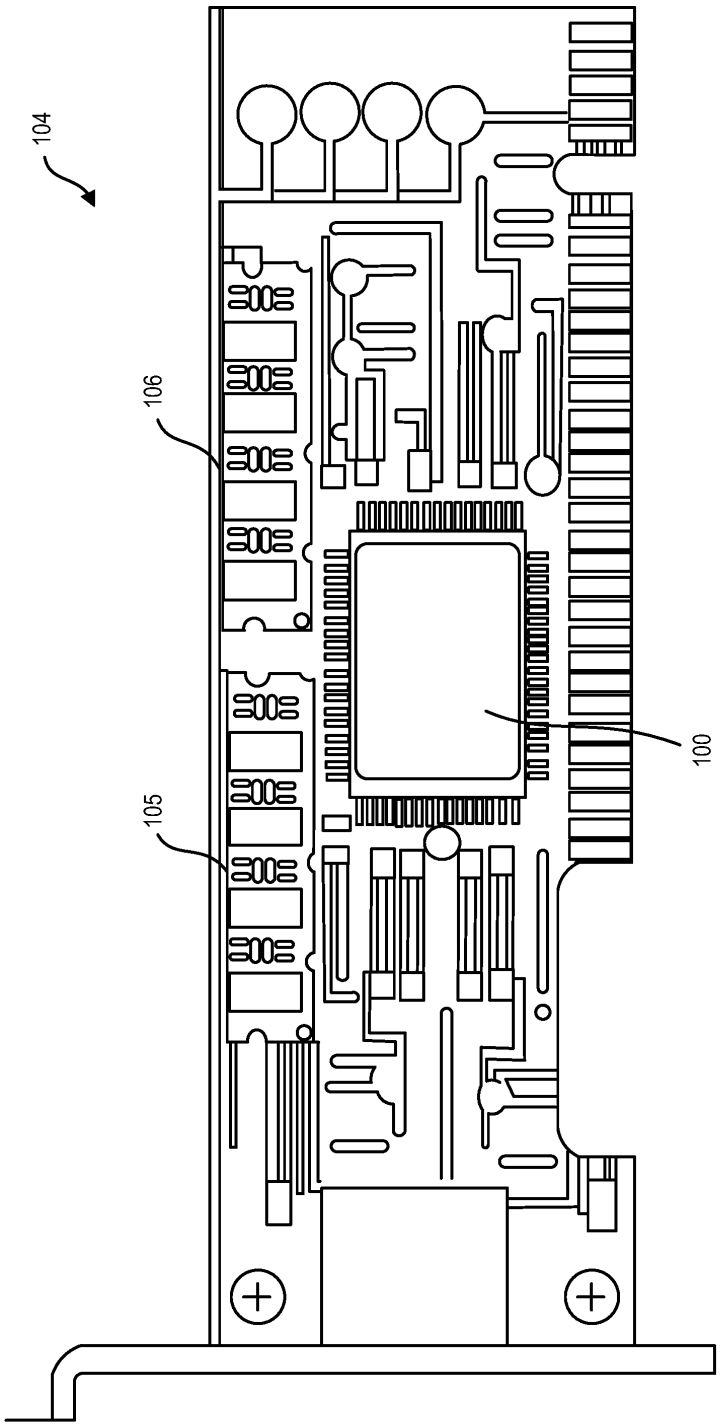

Аппаратный конвейер для полного генома
Аппарат. картирование затравки
Выравнивание Смита-Ватермана с гэпами
Сортировка по штрихкоду и позиции
Локальное повторное выравнивание
Маркировка дубликатов
Перекалибровка оценки качества оснований
Вывод BAM
Вывод редуцированного BAM
Распаковка CRAM
Сжатие CRAM
400
402
404
406
408
410
412
414
416
418
420
8 минут
4+2 минуты
4 минуты
Полный геном человека (40x): 18 мин

ФИГ. 48

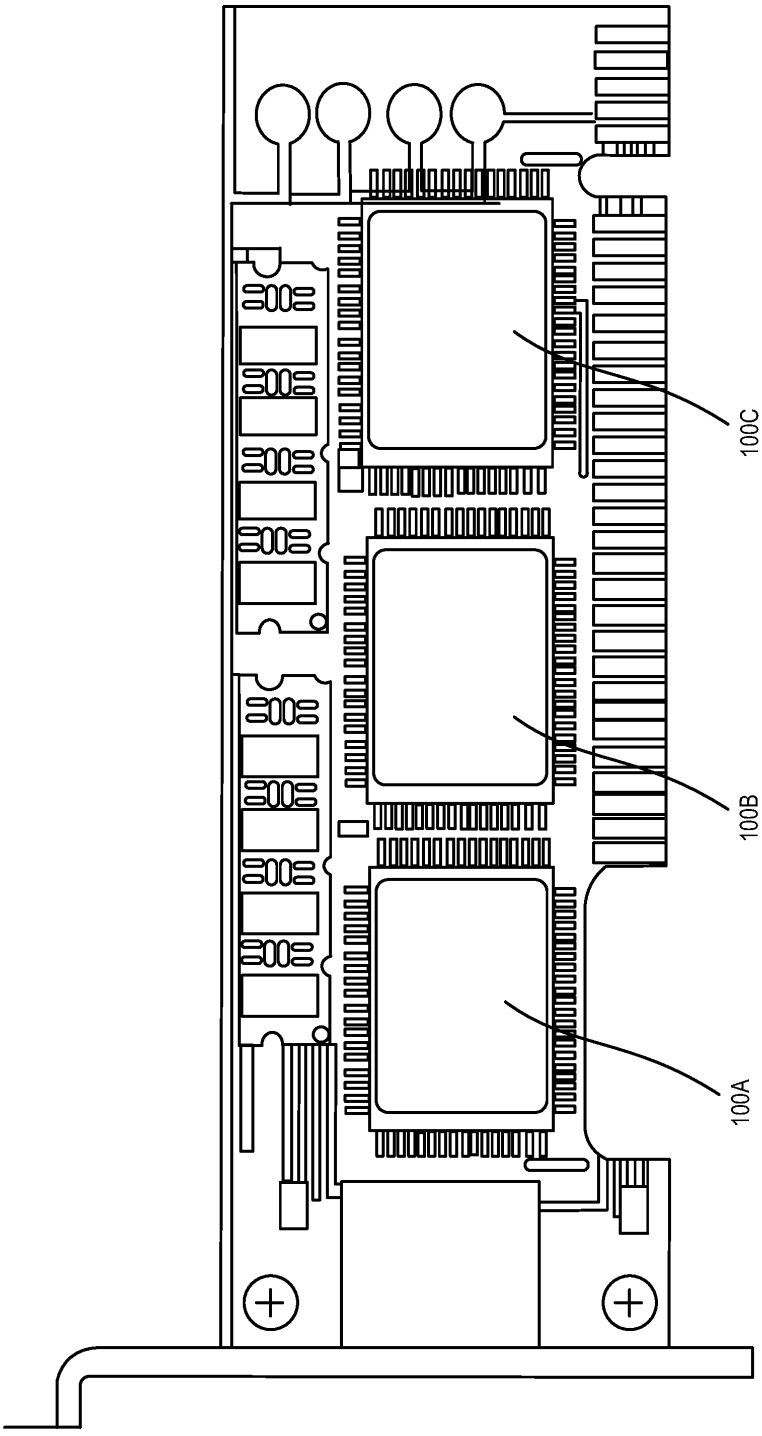

Аппаратная реализация конвейера анализа последовательности (FPGA или ASIC), как показано на Фиг. 2
Конвейерная платформа анализа последовательности, состоящая из FPGA/ASIC и программного обеспечения, выполняемого на локальном ПК, сервере или кластере с еще одним кластерных хранилищем на облаке (Фиг. 3)
Конвейерная платформа анализа последовательности, состоящая из FPGA/ASIC и поддержкой программным обеспечением на локальном ПК, сервере или кластере или облачным вычислением с хранением на облаке или в кластере (Фиг. 3)
Платформа может быть реализована путем интеграции аппаратного обеспечения (FPGA/ASIC) на плате с помощью PCIe или другого интерфейса
Платформа может быть реализована путем интеграции аппаратного обеспечения (FPGA/ASIC) непосредственно в геномный секвенатор с помощью патентованного интерфейса
Платформа может быть реализована путем интеграции аппаратн. обеспеч. (FPGA/ASIC) непосредственно в геномный секвенатор с помощью патентованного интерфейса
Платформа, интегрированная как плата PCIe, развернутая в главной системе, такой как ПК, сервер, кластер или секвенатор
Платформа, интегрированная как плата PCIe, развернутая в главной системе, такой как ПК, сервер или секвенатор
Третичная обработка, выполняемая в облаке или кластере
1
2
3
4
5
6
7
8
9
10

ФИГ. 49

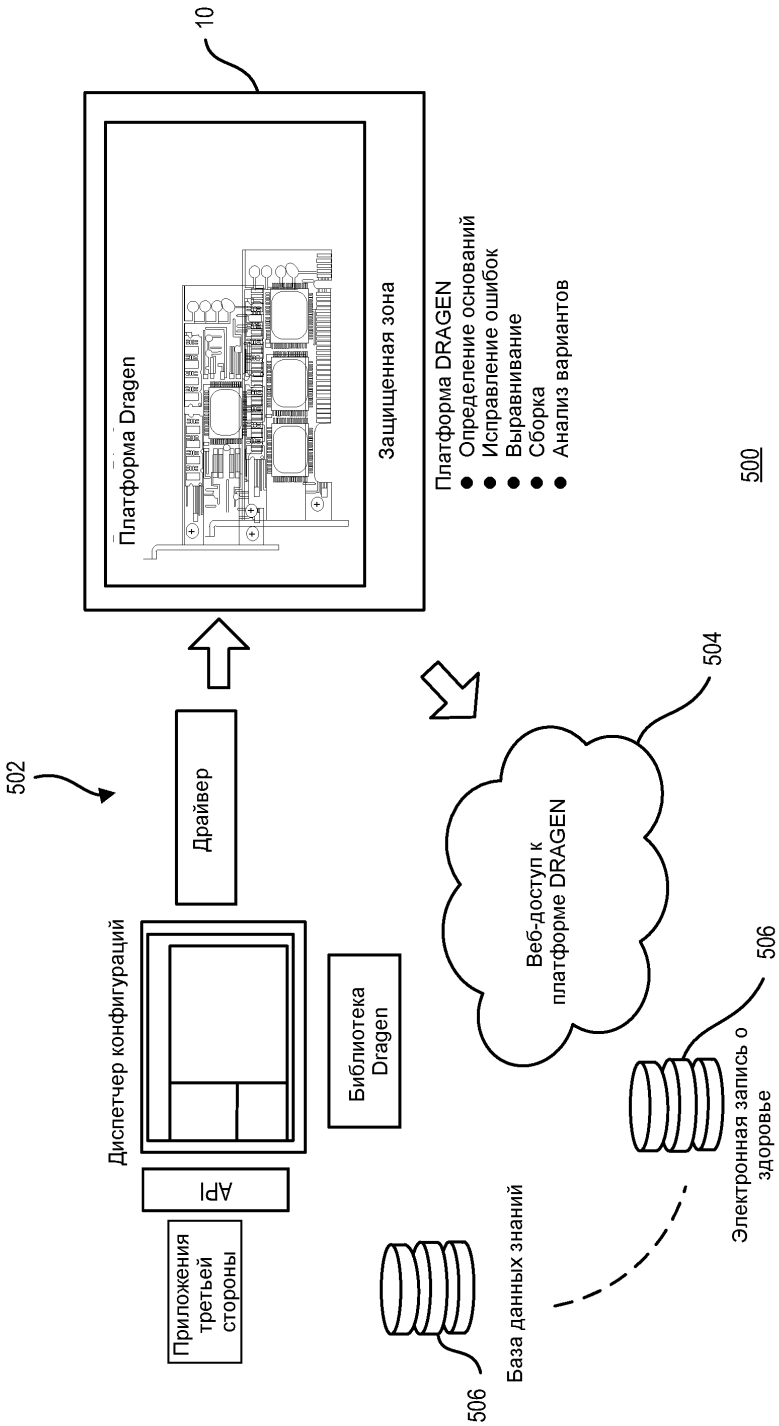
66




ФИГ. 50А




ФИГ. 50В


10
Платформа Dragen
Защищенная зона
Платформа DRAGEN
Определение оснований
Исправление ошибок
Выравнивание
Сборка
Анализ вариантов
500
502
Драйвер
Диспетчер конфигураций
API
Приложения третьей стороны
Библиотека Dragen
504
Веб-доступ к платформе DRAGEN
506
Электронная запись о здоровье
506
База данных знаний

ФИГ. 51