МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования

«Белорусский государственный технологический университет»

Кафедра полиграфического оборудования и систем обработки информации

Отчёт по лабораторной работе №5

**ИССЛЕДОВАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ**

Выполнила: Миневич Кристина ПОИТ 4-1

г. Минск 2021

**Цель работы** — изучить существующие способы оценки производительности вычислительных машин и получить базовые навыки сравнения производительности вычислительных машин.

**Теоретическая часть**

Основу для сравнения различных типов компьютеров между собой дают стандартные методики измерения производительности. В процессе развития вычислительной техники появилось несколько таких стандартных методик. Они позволяют разработчикам и пользователям осуществлять выбор между альтернативами на основе количественных показателей, что дает возможность постоянного прогресса в данной области.

Единицей измерения производительности компьютера является время: компьютер, выполняющий тот же объем работы за меньшее время является более быстрым. Время выполнения любой программы измеряется в секундах. Часто производительность измеряется как скорость появления некоторого числа событий в секунду, так что меньшее время подразумевает большую производительность.

Однако в зависимости от того, что мы считаем, время может быть определено различными способами. Наиболее простой способ определения времени называется астрономическим временем, временем ответа (response time), временем выполнения (execution time) или прошедшим временем (elapsed time). Это задержка выполнения задания, включающая буквально все: работу процессора, обращения к диску, обращения к памяти, ввод/вывод и накладные расходы операционной системы. Однако при работе в мультипрограммном режиме во время ожидания ввода/вывода для одной программы, процессор может выполнять другую программу, и система не обязательно будет минимизировать время выполнения данной конкретной программы.

**Производительность** определяется количеством вычислительной работы, выполняемой за единицу времени. Поскольку нет единой меры (единиц) для измерения вычислительной работы, отсутствует общепринятая методика оценки производительности. Для количественных оценок используют понятия номинальной и системной производительности.

При оценках чаще всего выделяют устройства: процессор, оперативную память и дисковую память. Номинальная производительность характеризует только потенциальные возможности устройств. При работе в составе системы эти возможности полностью не используются. Степень их использования зависит от характера и количества решаемых задач, интенсивности входного потока задач, операционной системы, выполняющей функции распределения ресурсов системы и организации управления ими.

Чаще всего показатель производительности требуется не как некоторая величина, измеренная в тех или иных единицах, а как средство для количественного сопоставления производительности различных типов выпускаемых промышленностью вычислительных машин и выбора более быстродействующей, а также для оценки влияния на её производительность вводимых усовершенствований в архитектуре вычислительный машины или системы при комплексировании и разработке. Для этого используется упрощенный подход, основанный на следующих положениях:

1. От абсолютных показателей переходим к относительным. Сравниваем производительность двух вычислительных систем — X, Y. При большем числе альтернативных вариантов (X, Y, Z, ...) одну (например, X) выбираем за базовую и сравниваем попарно X с Y, X с Z и т.д. Определяем относительный показатель k, показывающий, во сколько раз рассматриваемый вариант вычислительной системы производительней базового:

k=< Время испольнения на X >/< Время исполнения на Y >

k=<Производительность Y>/<Производительность X>.

2. Для программ, на которых оценивается производительность, используются:

\* реальная программа, имеющая широкое применение (например, компилятор языка программирования C, программы автоматизированного проектирования CAD, ...);

\* до реальной программы;

\* “игрушечный” тест объемом порядка 100 строк;

\* синтетический тест, содержащий набор реальных программ (каждая содержит 2-10 тыс. строк) из различных областей использования компьютеров (в показателе производительности время выполнения отдельных программ теста учитывается с весовыми коэффициентами).

Для оценки влияния на производительность вносимых в систему усовершенствований (расширения определённых ресурсов) используют подход (закон) Амдаля. Идея подхода заключается в том, чтобы оценку такого комплексного показателя, как относительное увеличение производительности, представить как композицию оценок более простых показателей.

Для измерения времени работы процессора на данной программе используется специальный параметр — **время ЦП** (CPU time), которое не включает время ожидания ввода/вывода или время выполнения другой программы. Очевидно, что время ответа, видимое пользователем, является полным временем выполнения программы, а не временем ЦП. Время ЦП может далее делиться на время, потраченное ЦП непосредственно на выполнение программы пользователя и называемое пользовательским временем ЦП, и время ЦП, затраченное операционной системой на выполнение заданий, затребованных программой, и называемое системным временем ЦП.

В ряде случаев системное время ЦП игнорируется из-за возможной неточности измерений, выполняемых самой операционной системой, а также из-за проблем, связанных со сравнением производительности машин с разными операционными системами. С другой стороны, системный код на некоторых машинах является пользовательским кодом на других и, кроме того, практически никакая программа не может работать без некоторой операционной системы. Поэтому при измерениях производительности процессора часто используется сумма пользовательского и системного времени ЦП.

В большинстве современных процессоров скорость протекания процессов взаимодействия внутренних функциональных устройств определяется не естественными задержками в этих устройствах, а задается единой системой синхросигналов, вырабатываемых некоторым генератором тактовых импульсов, как правило, работающим с постоянной скоростью. Дискретные временные события называются тактами синхронизации (clock ticks), просто тактами (ticks), периодами синхронизации (clock periods), циклами (cycles) или циклами синхронизации (clock cycles). Разработчики компьютеров обычно говорят о периоде синхронизации, который определяется либо своей длительностью (например, 10 наносекунд), либо частотой (например, 100 МГц). Длительность периода синхронизации есть величина, обратная к частоте синхронизации.

Таким образом, время ЦП для некоторой программы может быть выражено двумя способами: количеством тактов синхронизации для данной программы, умноженным на длительность такта синхронизации, либо количеством тактов синхронизации для данной программы, деленным на частоту синхронизации.



Важной характеристикой, часто публикуемой в отчётах по процессорам, является среднее количество тактов синхронизации на одну команду - CPI (clock cycles per instruction). При известном количестве выполняемых команд в программе этот параметр позволяет быстро оценить время ЦП для данной программы.

Таким образом, производительность ЦП зависит от трех параметров: такта (или частоты) синхронизации, среднего количества тактов на команду и количества выполняемых команд. Невозможно изменить ни один из указанных параметров изолированно от другого, поскольку базовые технологии, используемые для изменения каждого из этих параметров, взаимосвязаны: частота синхронизации определяется технологией аппаратных средств и функциональной организацией процессора; среднее количество тактов на команду зависит от функциональной организации и архитектуры системы команд; а количество выполняемых в программе команд определяется архитектурой системы команд и технологией компиляторов. Когда сравниваются две машины, необходимо рассматривать все три компоненты, чтобы понять относительную производительность.

В процессе поиска стандартной единицы измерения производительности компьютеров было принято несколько популярных единиц измерения, вследствие чего несколько безвредных терминов были искусственно вырваны из их хорошо определенного контекста и использованы там, для чего они никогда не предназначались. В действительности единственной подходящей и надежной единицей измерения производительности является время выполнения реальных программ, и все предлагаемые замены этого времени в качестве единицы измерения или замены реальных программ в качестве объектов измерения на синтетические программы только вводят в заблуждение.

Опасности некоторых популярных альтернативных единиц измерения (MIPS и MFLOPS) будут рассмотрены в соответствующих подразделах.

**MIPS**

Одной из альтернативных единиц измерения производительности процессора (по отношению к времени выполнения) является MIPS — (миллион целочисленных команд в секунду). Имеется несколько различных вариантов интерпретации определения MIPS.

В общем случае MIPS есть скорость операций с целыми числами в единицу времени, т.е. для любой данной программы MIPS есть просто отношение количества команд в программе к времени ее выполнения. Таким образом, производительность может быть определена как обратная к времени выполнения величина, причем более быстрые машины при этом будут иметь более высокий рейтинг MIPS.

Положительными сторонами MIPS является то, что эту характеристику легко понять, особенно покупателю, и что более быстрая машина характеризуется большим числом MIPS, что соответствует нашим интуитивным представлениям. Однако использование MIPS в качестве метрики для сравнения наталкивается на три проблемы. Во-первых, MIPS зависит от набора команд процессора, что затрудняет сравнение по MIPS компьютеров, имеющих разные системы команд. Во-вторых, MIPS даже на одном и том же компьютере меняется от программы к программе. В-третьих, MIPS может меняться по отношению к производительности в противоположенную сторону.

Классическим примером для последнего случая является рейтинг MIPS для машины, в состав которой входит сопроцессор плавающей точки. Поскольку в общем случае на каждую команду с плавающей точкой требуется большее количество тактов синхронизации, чем на целочисленную команду, то программы, используя сопроцессор плавающей точки вместо соответствующих подпрограмм из состава программного обеспечения, выполняются за меньшее время, но имеют меньший рейтинг MIPS. При отсутствии сопроцессора операции над числами с плавающей точкой реализуются с помощью подпрограмм, использующих более простые команды целочисленной арифметики и, как следствие, такие машины имеют более высокий рейтинг MIPS, но выполняют настолько большее количество команд, что общее время выполнения значительно увеличивается. Подобные аномалии наблюдаются и при использовании оптимизирующих компиляторов, когда в результате оптимизации сокращается количество выполняемых в программе команд, рейтинг MIPS уменьшается, а производительность увеличивается.

**MFLOPS**

Измерение производительности компьютеров при решении научно-технических задач, в которых существенно используется арифметика с плавающей точкой, всегда вызывало особый интерес. Именно для таких вычислений впервые встал вопрос об измерении производительности, а по достигнутым показателям часто делались выводы об общем уровне разработок компьютеров. Обычно для научно-технических задач производительность процессора оценивается в MFLOPS (миллионах чисел-результатов вычислений с плавающей точкой в секунду, или миллионах элементарных арифметических операций над числами с плавающей точкой, выполненных в секунду).

Как единица измерения, MFLOPS, предназначена для оценки производительности только операций с плавающей точкой, и поэтому не применима вне этой ограниченной области. Например, программы компиляторов имеют рейтинг MFLOPS близкий к нулю вне зависимости от того, насколько быстра машина, поскольку компиляторы редко используют арифметику с плавающей точкой.

Рейтинг MFLOPS зависит от машины и от программы. Этот термин менее безобидный, чем MIPS. Он базируется на количестве выполняемых операций, а не на количестве выполняемых команд. По мнению многих программистов, одна и та же программа, работающая на различных компьютерах, будет выполнять различное количество команд, но одно и то же количество операций с плавающей точкой. Именно поэтому рейтинг MFLOPS предназначался для справедливого сравнения различных машин между собой.

Ливерморские циклы — это набор фрагментов фортран-программ, каждый из которых взят из реальных программных систем, эксплуатируемых в Ливерморской национальной лаборатории им.Лоуренса (США). Обычно при проведении испытаний используется либо малый набор из 14 циклов, либо большой набор из 24 циклов.

Пакет Ливерморских циклов используется для оценки производительности вычислительных машин с середины 60-х годов. Ливерморские циклы считаются типичными фрагментами программ численных задач. Появление новых типов машин, в том числе векторных и параллельных, не уменьшило важности Ливерморских циклов, однако изменились значения производительности и величины разброса между разными циклами.

**Тесты TPC**

TPC определяет и управляет форматом нескольких тестов для оценки производительности OLTP (On-LineTransactionProcessing), включая тесты TPC-A, TPC-B и TPC-C. TPC требует только, чтобы при создании оценочного теста выполнялись определенные условия. Хотя упомянутые тесты TPC не являются характерными тестами для оценки производительности баз данных, системы реляционных баз данных являются ключевыми компонентами любой системы обработки транзакций.

Следует отметить, что как и любой другой тест, ни один тест TPC не может измерить производительность системы, которая применима для всех возможных сред обработки транзакций, но эти тесты действительно могут помочь пользователю справедливо сравнивать похожие системы. Однако, когда пользователь делает покупку или планирует решение о покупке, он должен понимать, что никакой тест не может заменить его конкретную прикладную задачу.

**AIM**

Одной из независимых организаций, осуществляющей оценку производительности вычислительных систем, является частная компания AIM Technology, которая была основана в 1981 году. Компания разрабатывает и поставляет программное обеспечение для измерения производительности систем, а также оказывает услуги по тестированию систем конечным пользователям и поставщикам вычислительных систем и сетей, которые используют промышленные стандартные операционные системы, такие как UNIX и OS/2.

За время своего существования компания разработала специальное программное обеспечение, позволяющее легко создавать различные рабочие нагрузки, соответствующие уровню тестируемой системы и требованиям по ее использованию. Это программное обеспечение состоит из двух основных частей: генератора тестовых пакетов (BenchmarkGenerator) и нагрузочных смесей (LoadMixes) прикладных задач.

Генератор тестовых пакетов представляет собой программную систему, которая обеспечивает одновременное выполнение множества программ. Он содержит большое число отдельных тестов, которые потребляют определенные ресурсы системы, и тем самым акцентируют внимание на определенных компонентах, из которых складывается ее общая производительность. При каждом запуске генератора могут выполняться любые отдельные или все доступные тесты в любом порядке и при любом количестве проходов, позволяя тем самым создавать для системы практически любую необходимую рабочую нагрузку. Все это дает возможность тестовому пакету моделировать любой тип смеси при постоянной смене акцентов (для лучшего представления реальной окружающей обстановки) и при обеспечении высокой степени конфигурирования.

Каждая нагрузочная смесь представляют собой формулу, которая определяет компоненты требуемой нагрузки. Эта формула задается в терминах количества различных доступных тестов, которые должны выполняться одновременно для моделирования рабочей нагрузки.

Используя эти две части программного обеспечения AIM, можно действительно создать для тестируемой системы любую рабочую нагрузку, определяя компоненты нагрузки в терминах тестов, которые должны выполняться генератором тестовых пакетов. Если некоторые требуемые тесты отсутствуют в составе генератора тестовых пакетов, то они могут быть легко туда добавлены.

Генератор тестовых пакетов во время своей работы “масштабирует” или увеличивает нагрузку на систему. Первоначально он выполняет и хронометрирует одну копию нагрузочной смеси. Затем одновременно выполняет и хронометрирует три копии нагрузочной смеси и т.д. По мере увеличения нагрузки, на основе оценки производительности системы, выбираются различные уровни увеличения нагрузки. В конце концов может быть нарисована кривая пропускной способности, показывающая возможности системы по обработке нагрузочной смеси в зависимости от числа моделируемых нагрузок. Это позволяет с достаточной достоверностью дать заключение о возможностях работы системы при данной нагрузке или при изменении нагрузки.

Очевидно, что сам по себе процесс моделирования рабочей нагрузки мало что дал бы для сравнения различных машин между собой при отсутствии у AIM набора хорошо подобранных смесей, которые представляют собой ряд важных для пользователя прикладных задач.

Все смеси AIM могут быть разделены на две категории: стандартные и заказные. Заказные смеси создаются для точного моделирования особенностей среды конечного пользователя или поставщика оборудования. Заказная смесь может быть тесно связана с определенными тестами, добавляемыми к генератору тестовых пакетов. В качестве альтернативы заказная смесь может быть связана с очень специфическим приложением, которое создает для системы необычную нагрузку. В общем случае заказные смеси разрабатываются на основе одной из стандартных смесей AIM путем ее “подгонки” для более точного представления определенной ситуации. Обычно заказные смеси разрабатываются заказчиком совместно с AIM Technology, что позволяет использовать многолетний опыт AIM по созданию и моделированию нагрузочных смесей.

**Практическая часть**

**УСТАНОВКА SISOFTAWARE SANDRA**

**Тесты производительности (benchmarks)**  —это тесты, измеряющие производительность систем, или подсистема на решении заранее определенных задач или наборов задач.

Результаты тестов производительности обычно используются для предсказания производительности незнакомой (не купленной, не установленной) системы при решении (заранее определенных) задач пользователя и для сравнения производительности систем или подсистем для некоторого класса задач.

Тесты производительности также могут быть использованы и при контроле, диагностике и настройке систем для выявления узких мест и плохо работающих (ненастроенных) подсистем.

Тесты производительности чаще всего измеряют обобщенные показатели производительности систем или подсистем, таких как графическая подсистема, подсистема ввода/вывода, процессор (для операций с целыми числами и для операций с числами с плавающей запятой) и т.д., однако, многие тесты измеряют производительность при решении конкретных, но типичных задач, таких как отображение многоугольников, чтение и запись файлов или операции над матрицами.

Каждая организация или подразделение, занимающееся решением своих задач на компьютерах, может (что, в принципе, весьма полезно, если будет сделано грамотно) создать свой набор тестов производительности, которые будут наиболее точно отражать специфику задач, решаемых данным подразделением. Частным случаем такого теста может быть решение реальных задач с фиксированными исходными данными и измерение времени решения. Главный недостаток таких тестов — либо результаты будут известны только для систем, которые доступны авторам, либо авторы должны широко распространить свои тесты и убедить других людей проводить их и публиковать (сообщать) результаты.

В мире существует уже достаточно большой набор разработанных тестов, оценивающих различные аспекты производительности систем и подсистем. Наиболее известные из них: 007 (ODBMS), AIM, Dhrystone, Khornerstone, LFK (Livermore Loops), LINPACK, MUSBUS, NAS Kernels, Nhfsstone, PERFECT, RhosettaStone, SLALOM, SPEC, SSBA, TPC, WPI Benchmark Suite, Whetstone, Xstone, SYSmark, Stanford, IOBENCH, IOZONE, Byte, Netperf, Nettest, Hartstone, EuroBen, Fhourstones, Heapsort, Hanoi, Flops, C LINPACK, TFFTDP, Matrix Multiply (MM), Digital Review.

Производительность вычислительных систем общего назначения оценивается в зависимости от области применения номинальной, комплексной, системной производительностью и производительностью на рабочей нагрузке.

Номинальная производительность характеризует только быстродействие, или производительность устройств, входящих в состав системы. Комплексная производительность учитывает не только быстродействие устройств, но и структуру системы — ее влияние на быстродействие совместно функционирующих устройств.

Системная производительность учитывает как вышеназванные факторы — быстродействие устройств и структуру связей между ними, так и влияние операционной системы.

С производительностью тесно связана такая характеристика качества обслуживания пользователей, как время ответа, т. е. время пребывания задач в системе. Поэтому при оценке производительности определяется не только количество работы, выполняемое системой в единицу времени, но и время ответа для всего множества задач и отдельных классов задач.

Производительность вычислительной системы проявляется, с одной стороны, в скорости обработки задач, а с другой — в степени использования ресурсов системы. Чем больше загружены ресурсы, тем выше производительность системы, и недогрузка ресурсов свидетельствует о наличии резервов для повышения производительности. Поэтому при анализе производительности системы оцениваются не только показатели производительности, но и показатели, характеризующие использование ресурсов.

Производительность вычислительной системы связана с продолжительностью процессов обработки задач, которая зависит от трех факторов:

1) рабочей нагрузки;

2) конфигурации системы;

3) режима обработки задач.

Эти три фактора в совокупности определяют порядок развития вычислительных процессов во времени, и первая задача анализа производительности сводится к поиску компактных и информативных форм представления вычислительных процессов. Эти формы создают концептуальную (понятийную) основу для оценки функционирования вычислительных систем в процессе эксплуатации и при исследовании с помощью моделей производительности. Вторая задача анализа — создание моделей, позволяющих прогнозировать производительность систем для различной конфигурации, режимов обработки и, возможно, разной рабочей нагрузки.

Для исследования производительности вычислительных систем используется программное приложение SiSoftware Sandra.

**SiSoftware Sandra** - это системный анализатор для 32-х и 64-битных версий Windows, включающий в себя тестовые и информационный модули. Sandra старается превзойти другие подобные утилиты и показать реальную картину, объединяя в одной программе возможности для сравнения производительности как на высоком, так и на низком уровне.

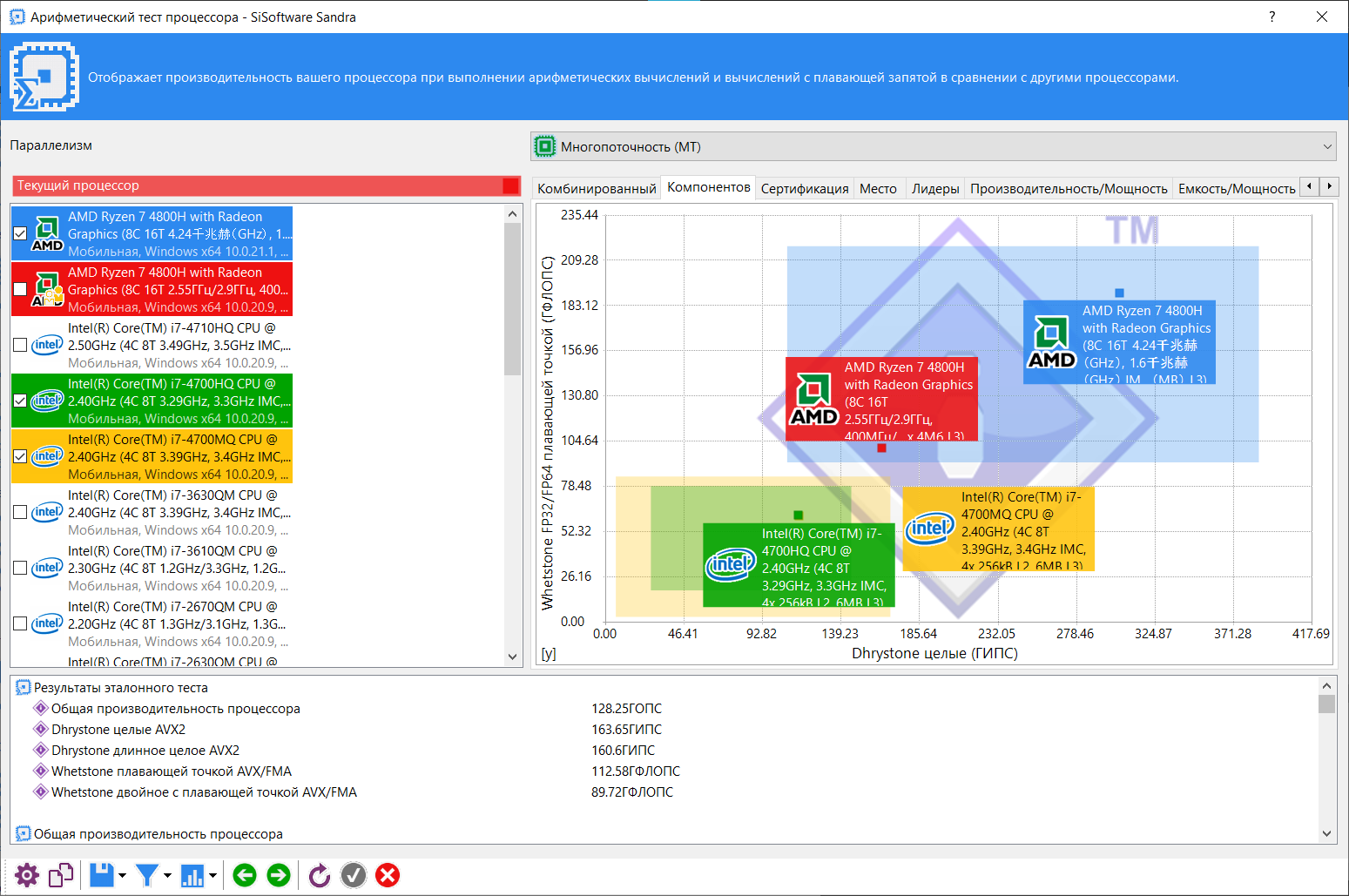
Вы можете получить сведения о процессоре, чипсете, видеоадаптере, портах, принтерах, звуковой карте, памяти, сети, Windows, AGP, соединениях ODBC, USB2, Firewire, и т.д.

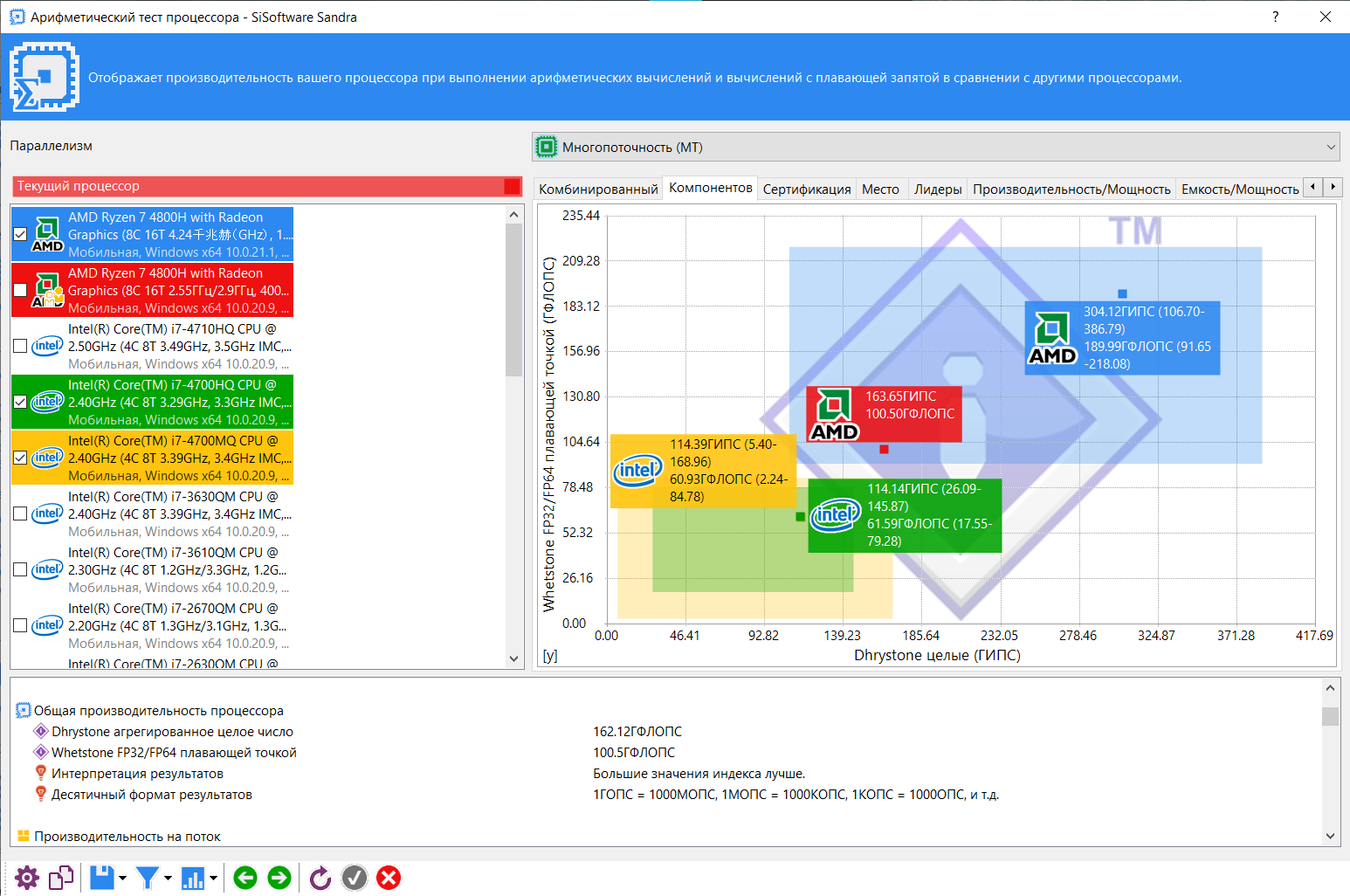
Вы можете сохранять/распечатывать/отправлять по факсу и электронной почте/загружать на сервер или вставлять в базу данных ADO/ODBC отчеты в текстовом, HTML, XML, SMS/DMI или RPT форматах.

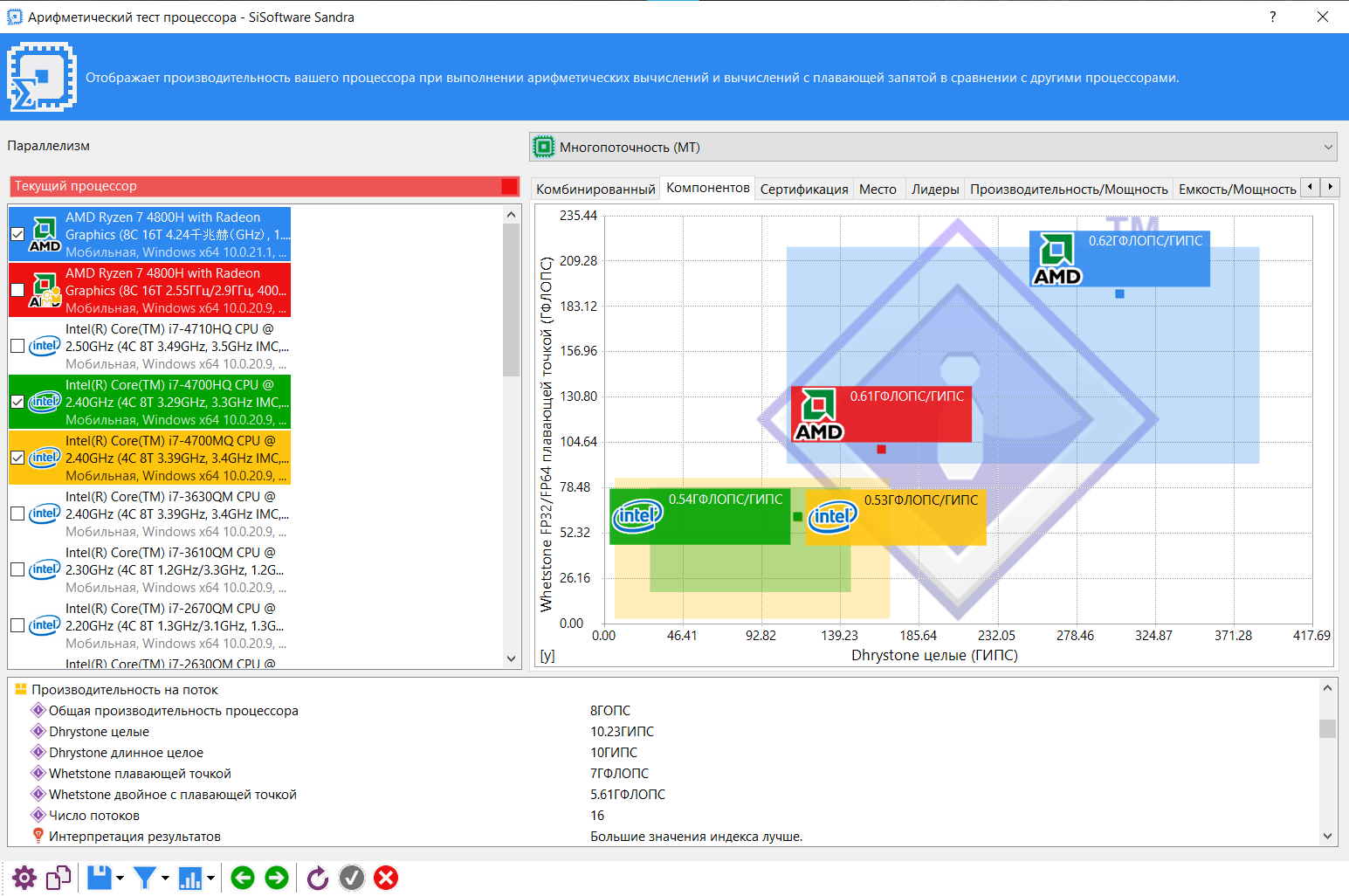
**ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ**

**Арифметический тест**

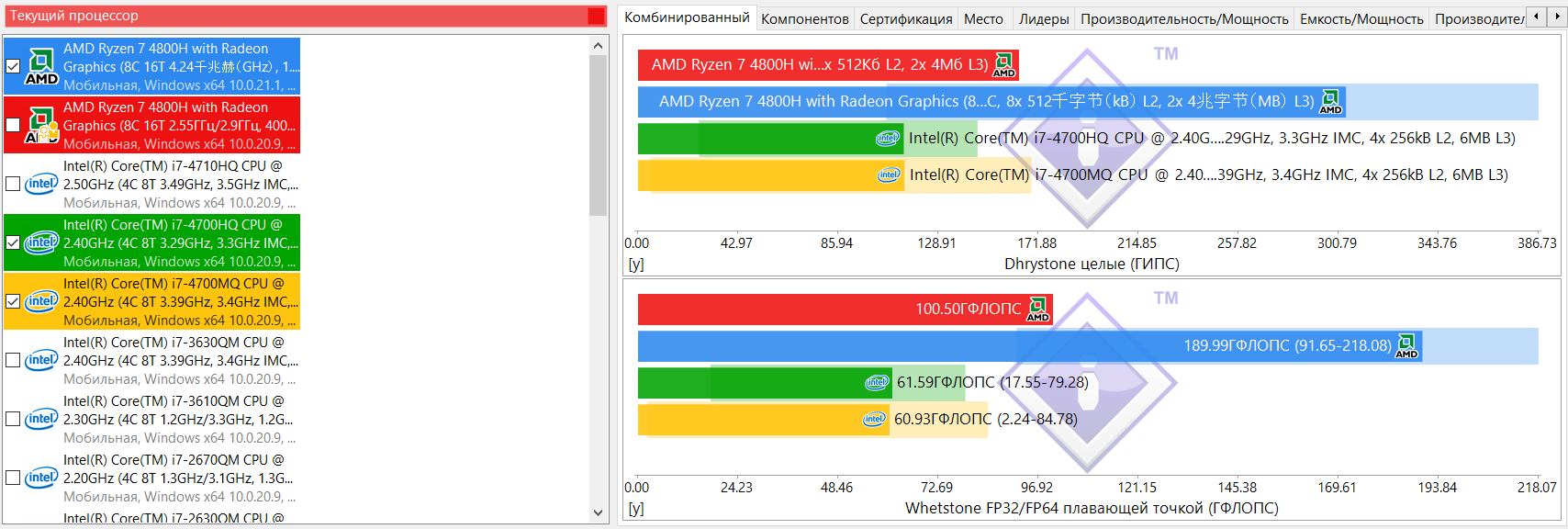
(данный тест отображает производительность вашего процессора при выполнении арифметических вычислений и вычислений с плавающей запятой в сравнении с другими процессорами)

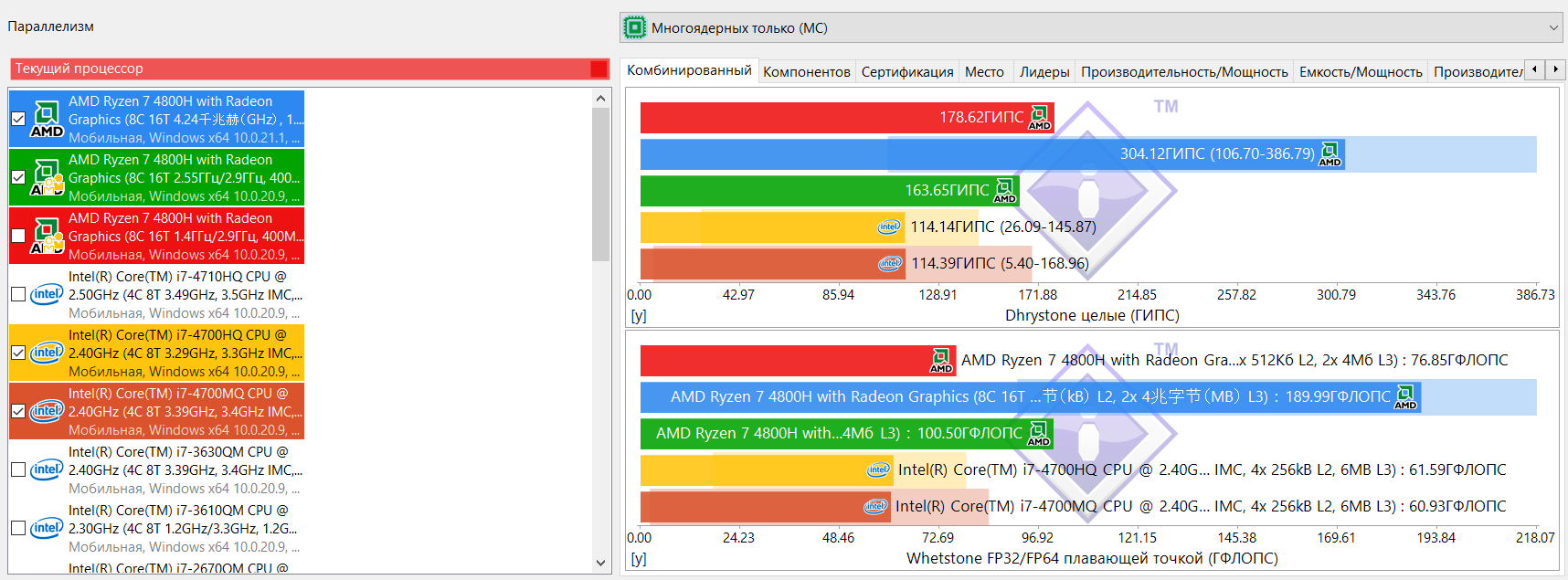
****

****

****

Комбинированный график:

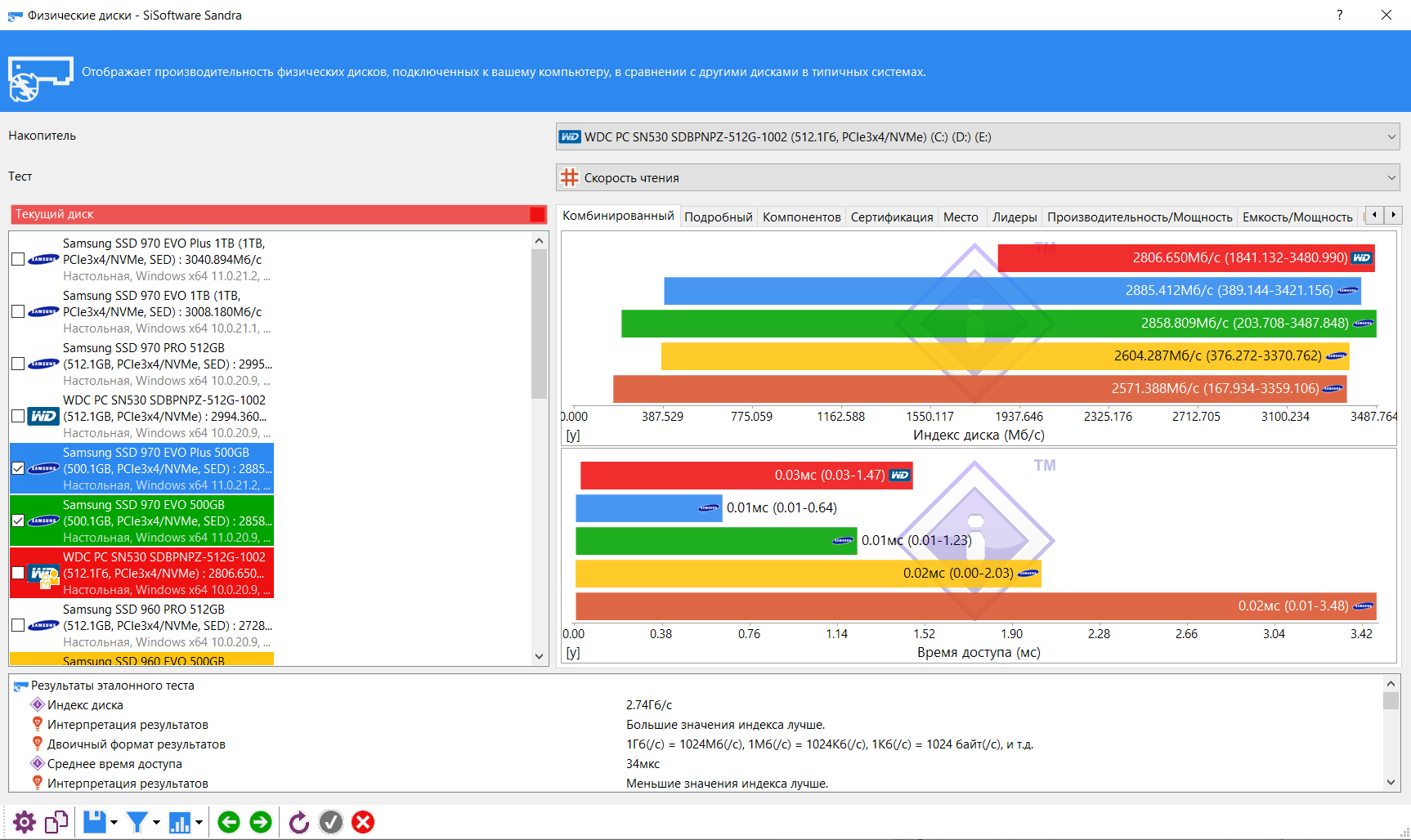
****

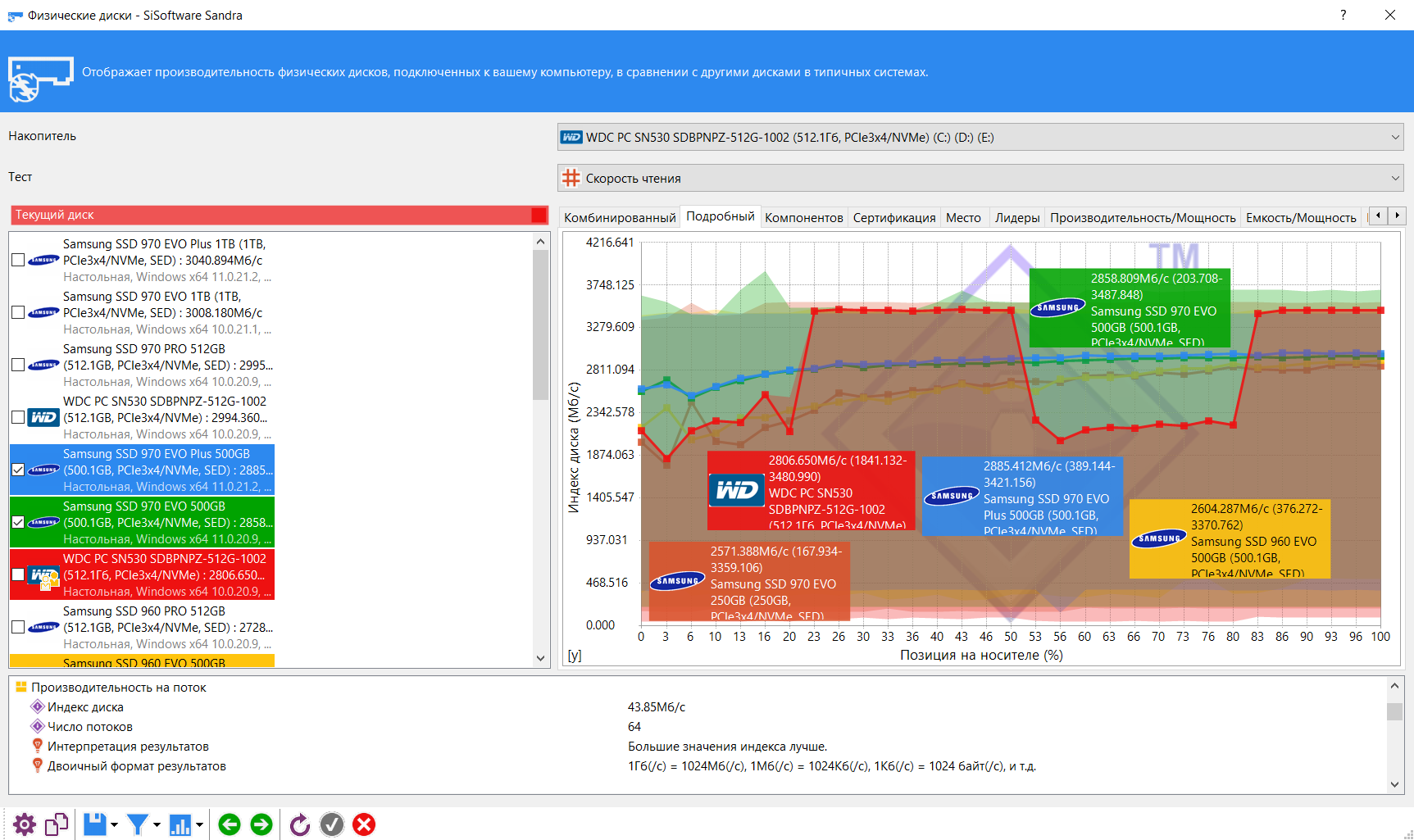
****

****

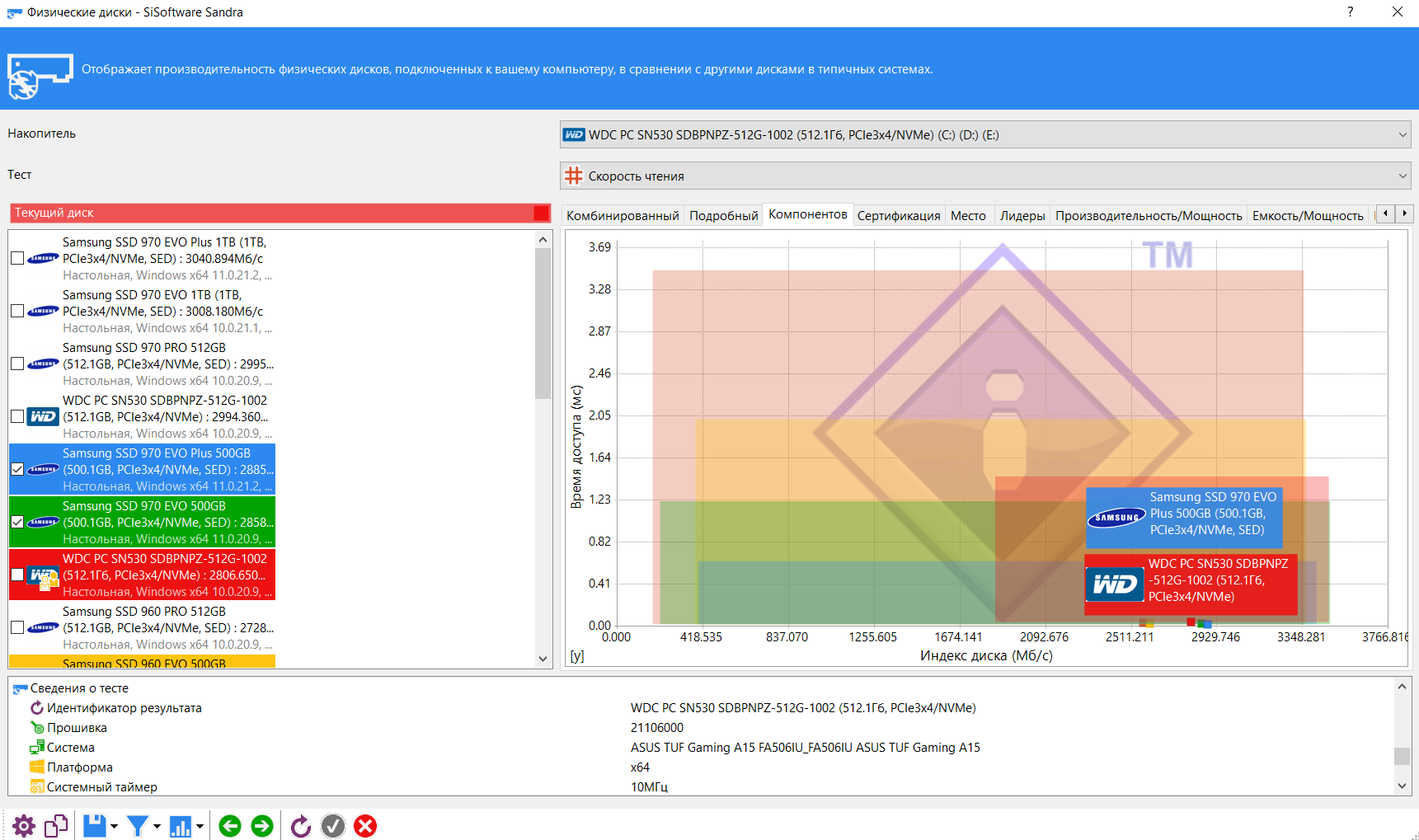
**Тест «Физические диски»**

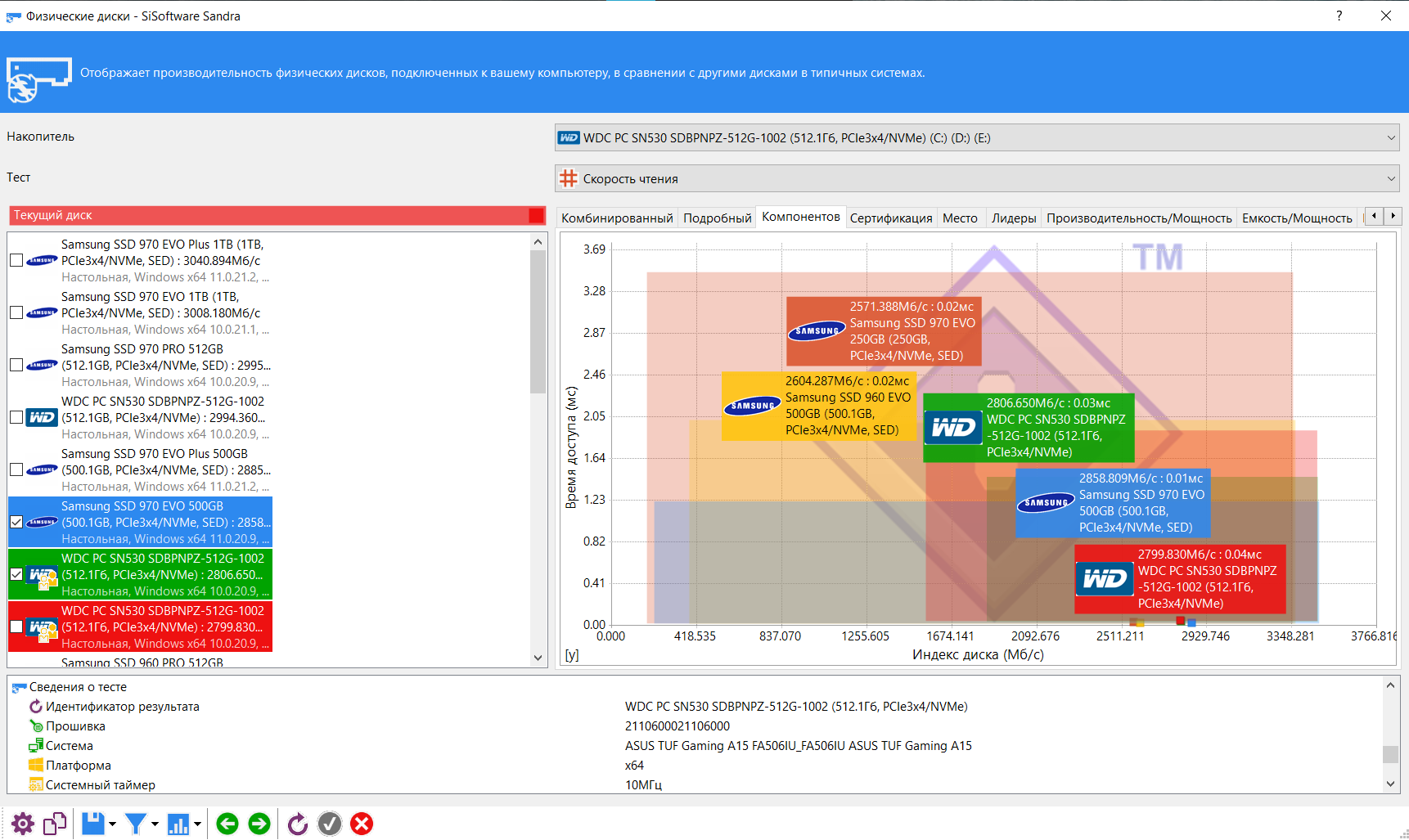
(данный тест отображает производительность физических дисков, подключенных к вашему компьютеру, в сравнении с другими дисками в типичных системах. Скорость чтения измеряется в Мб/с, а среднее время доступа в мс)

****

****

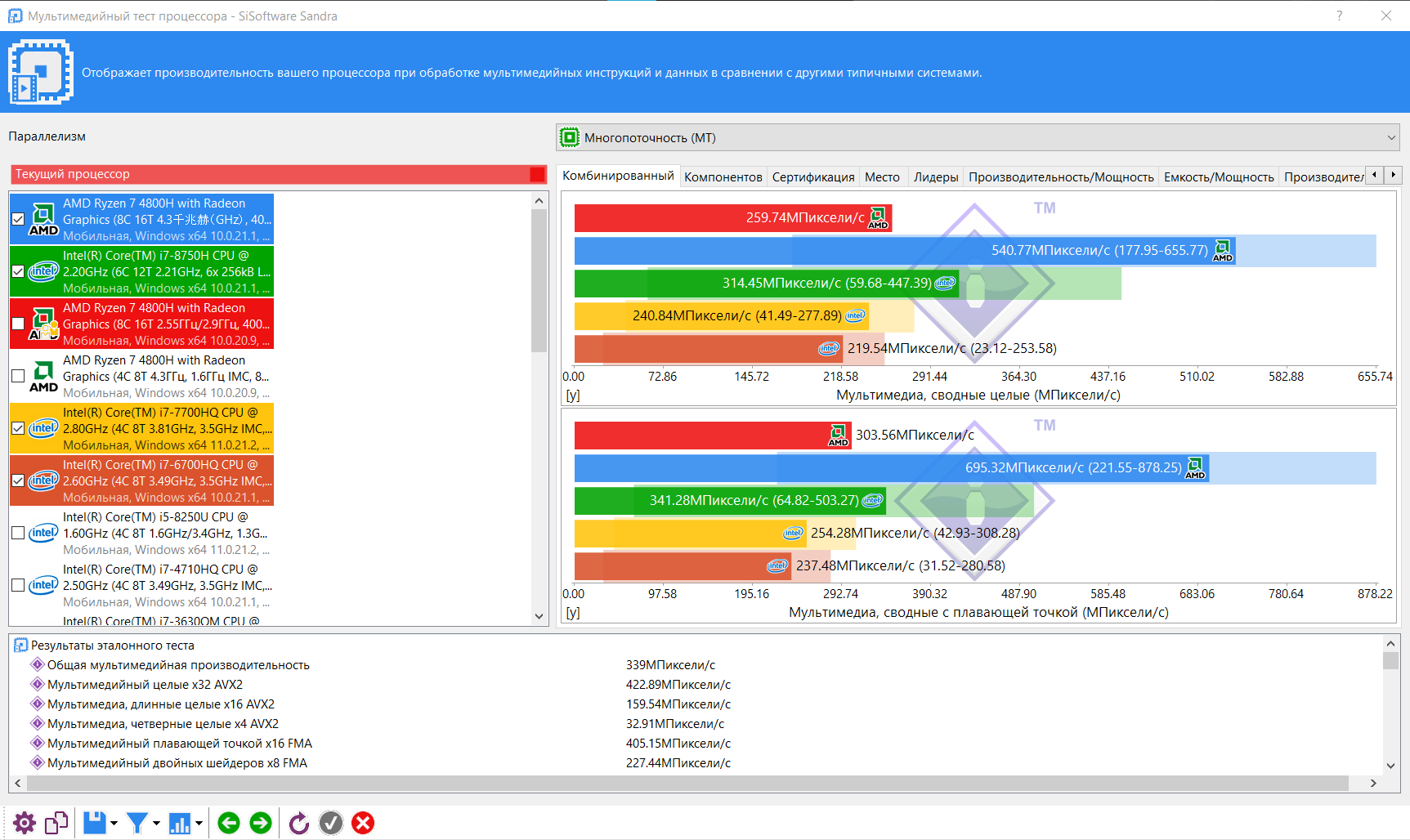
****

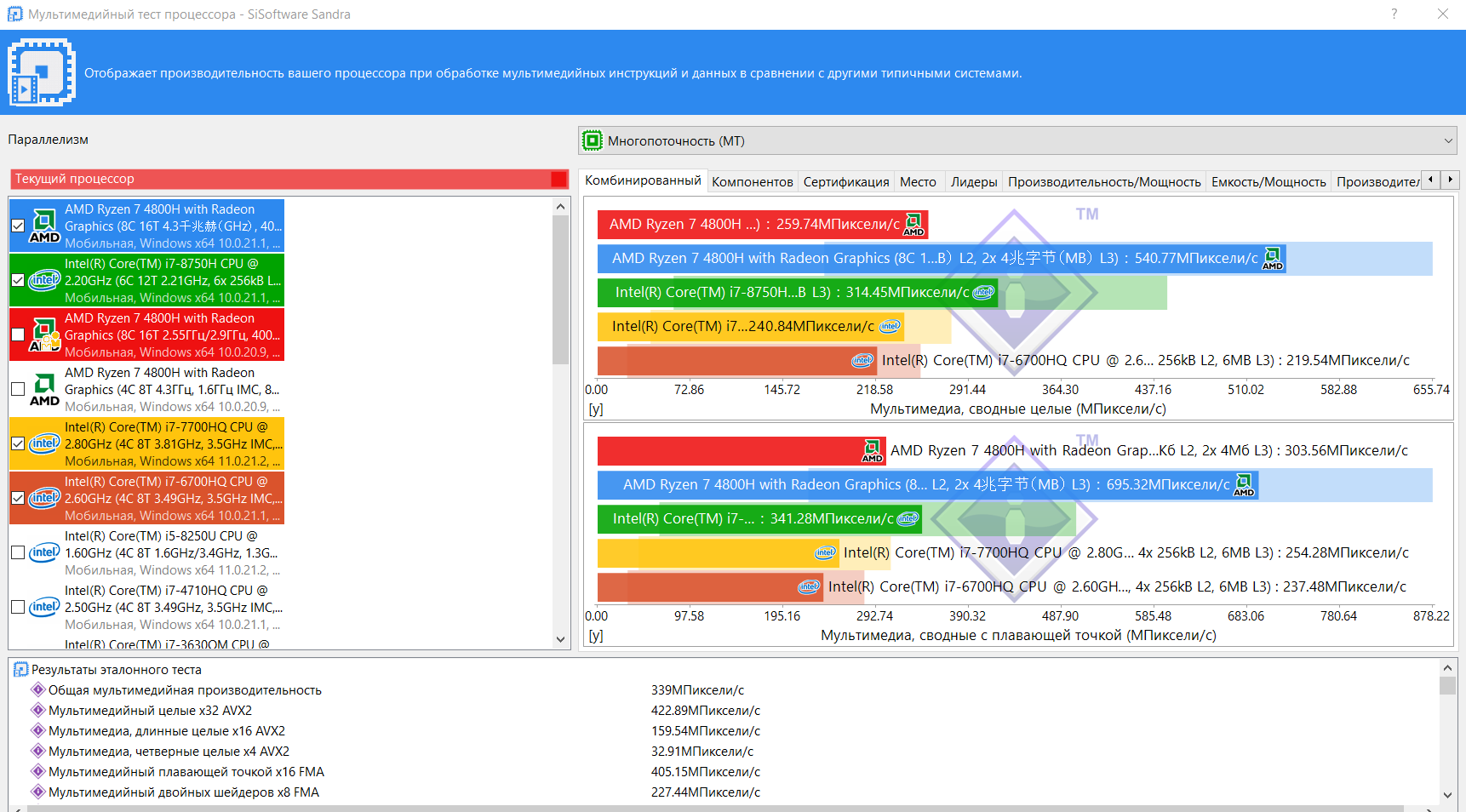
****

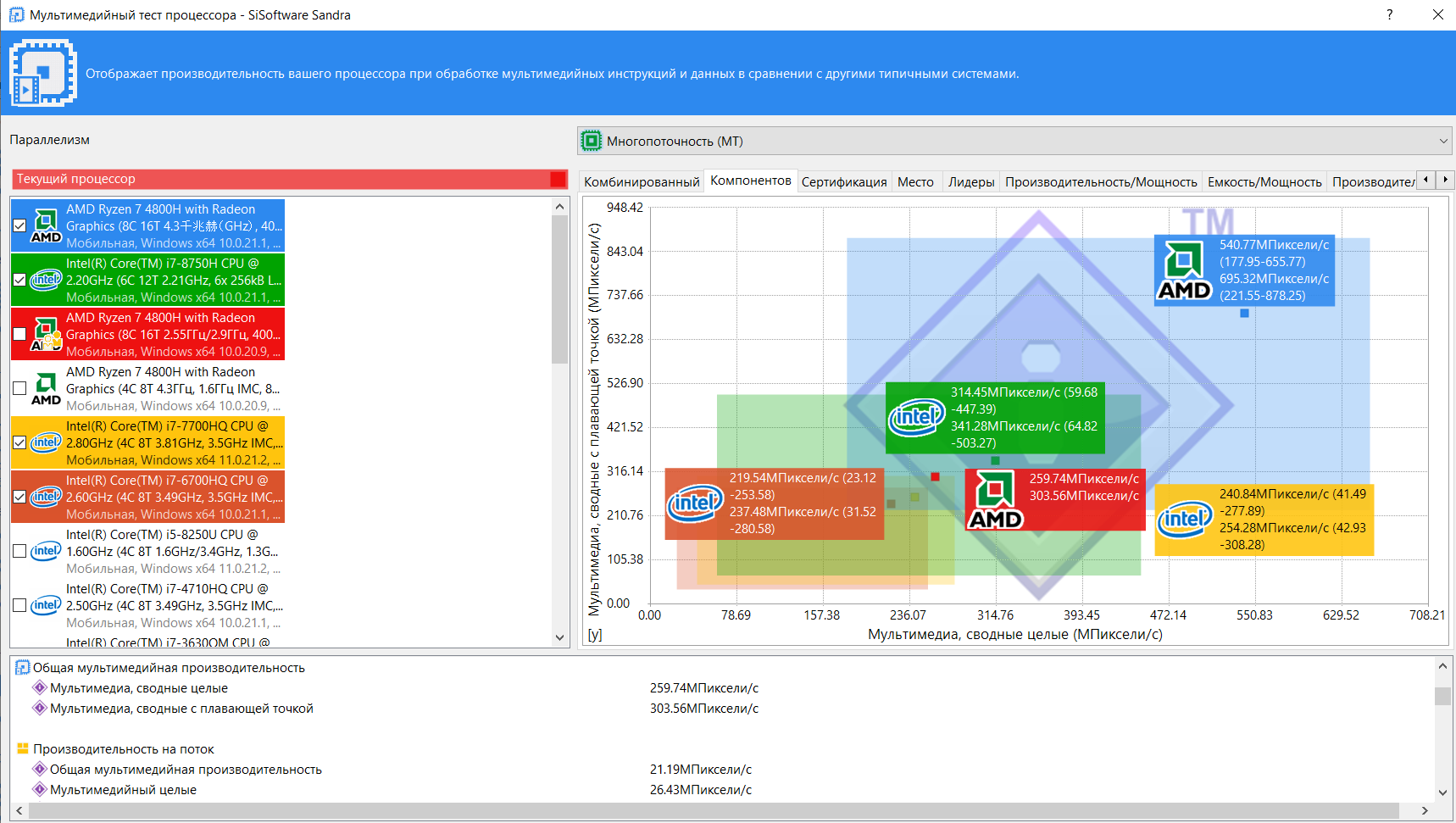
****

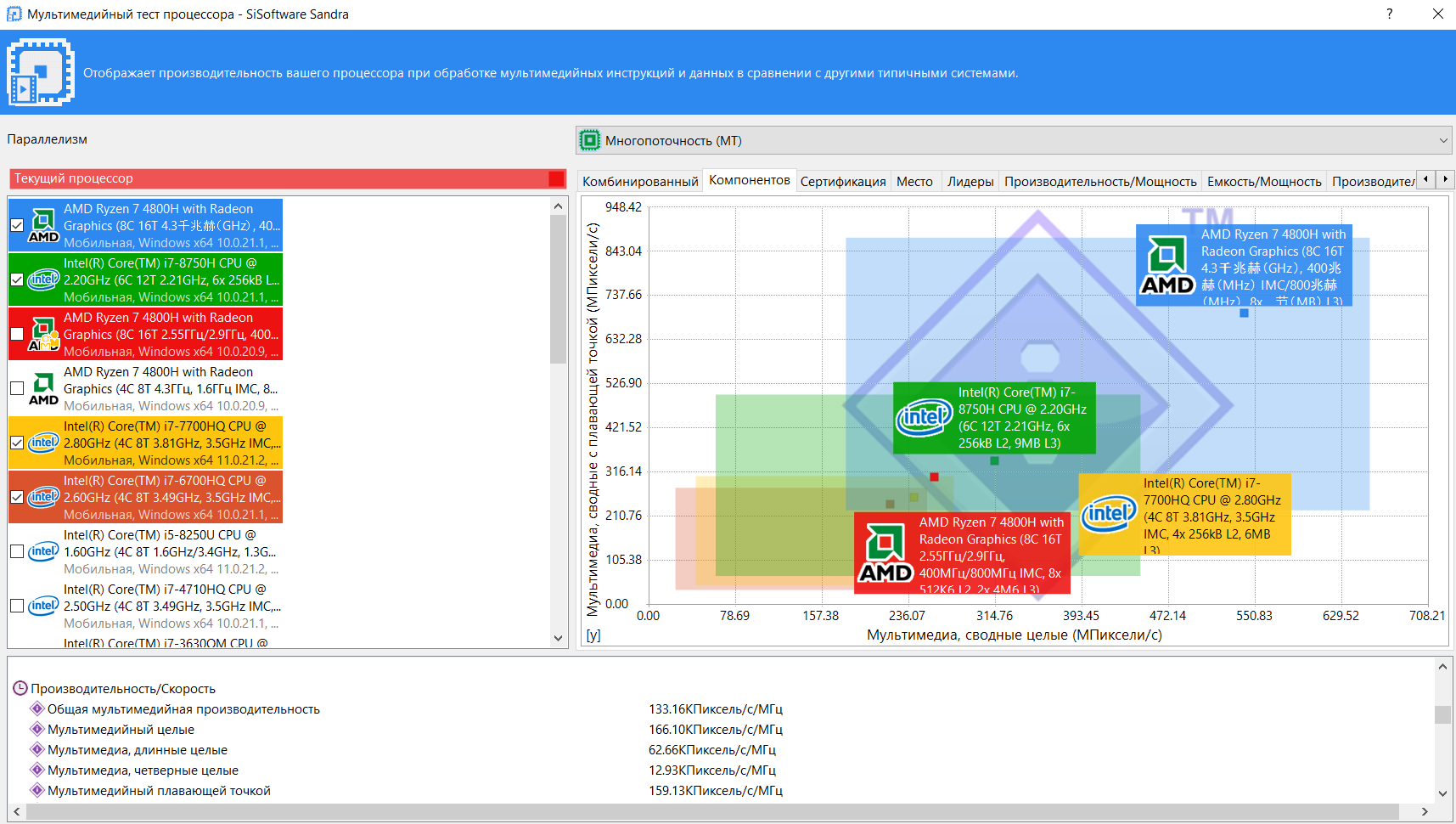
**Мультимедийный тест**

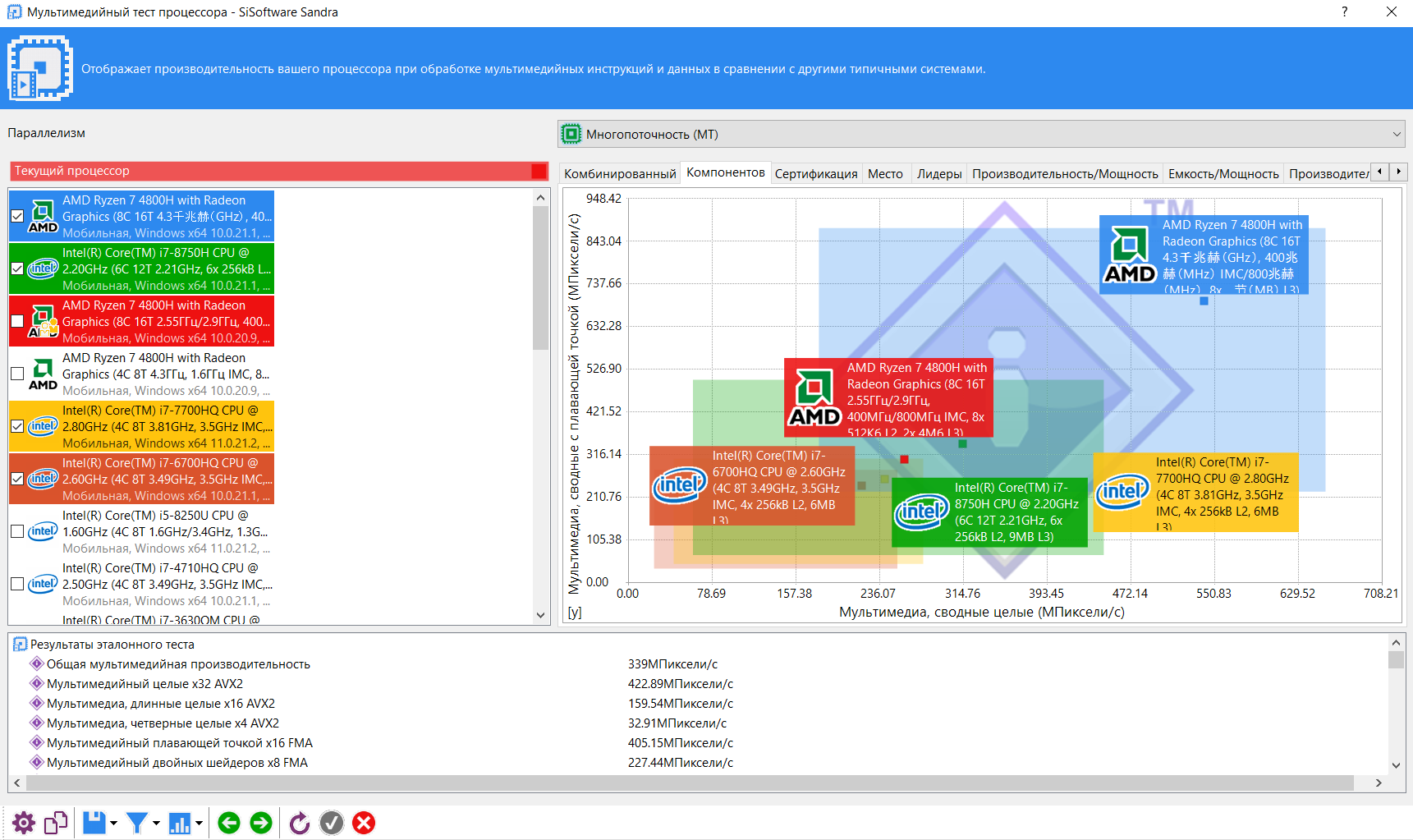
(тест дает возможность оценить производительность системы в работе с мультимедийными данными при использовании поддерживаемых процессором наборов SIMD-инструкций в сравнении с другими эталонными компьютерными системами)

****

****

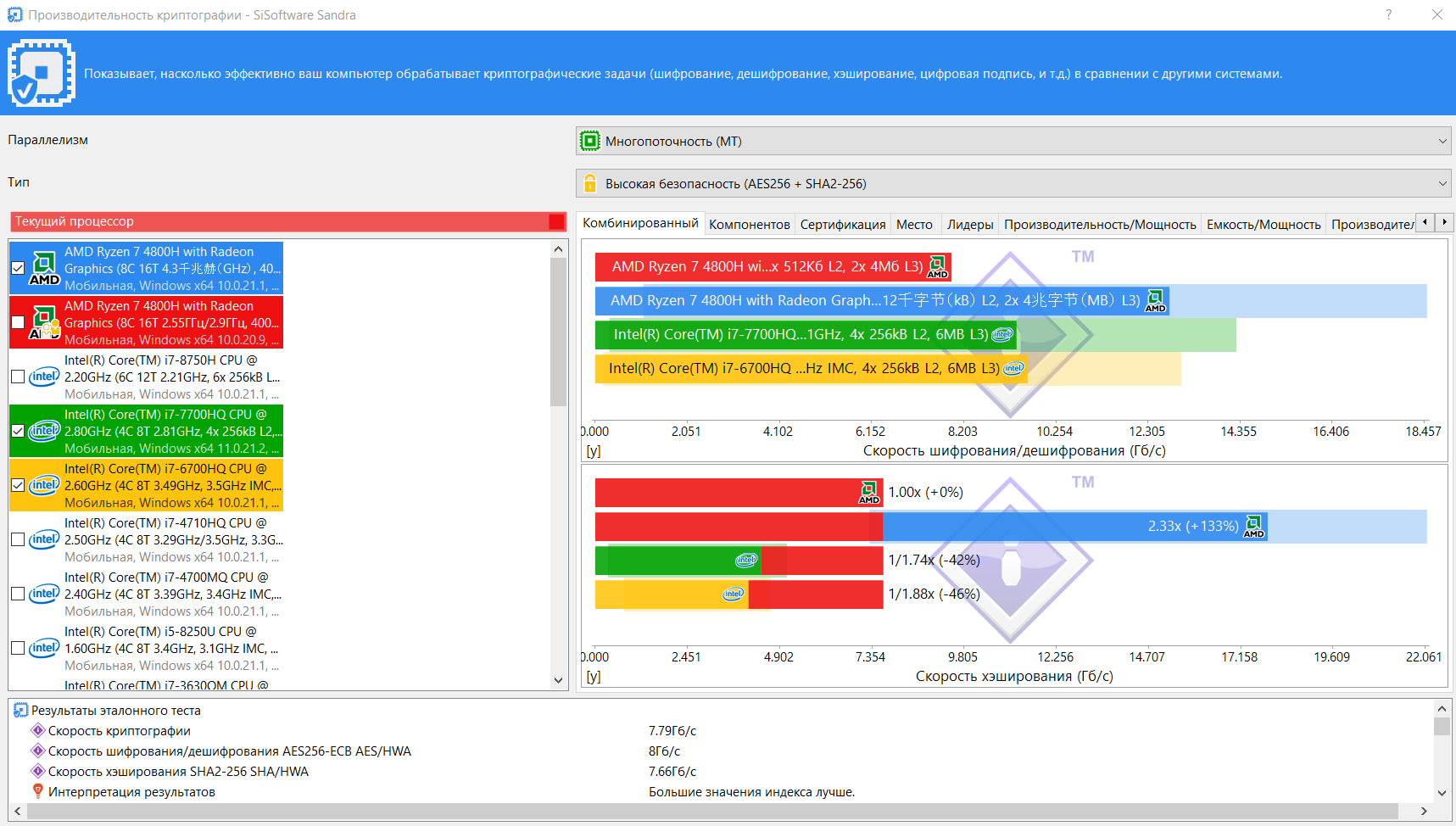
****

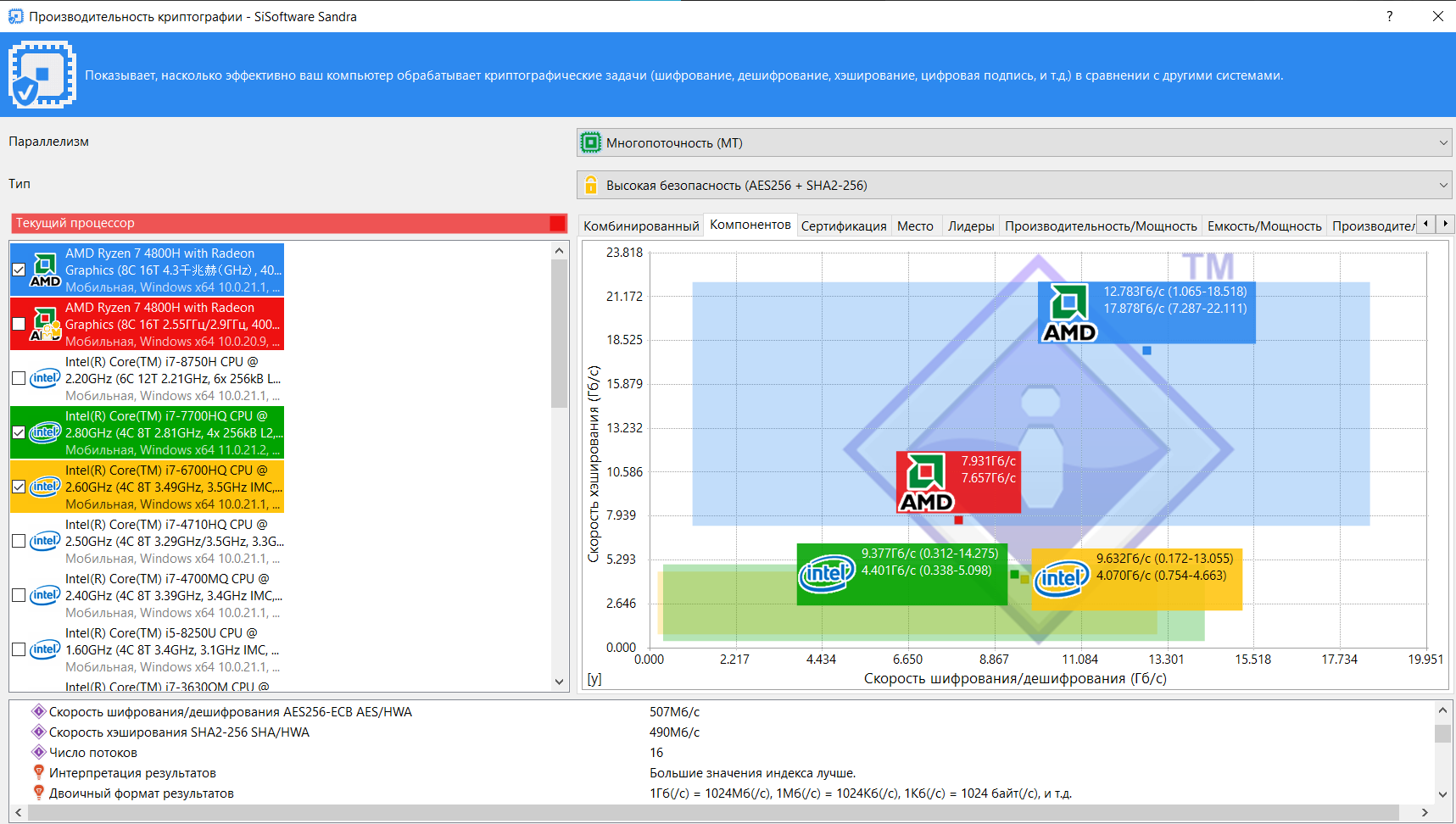
****

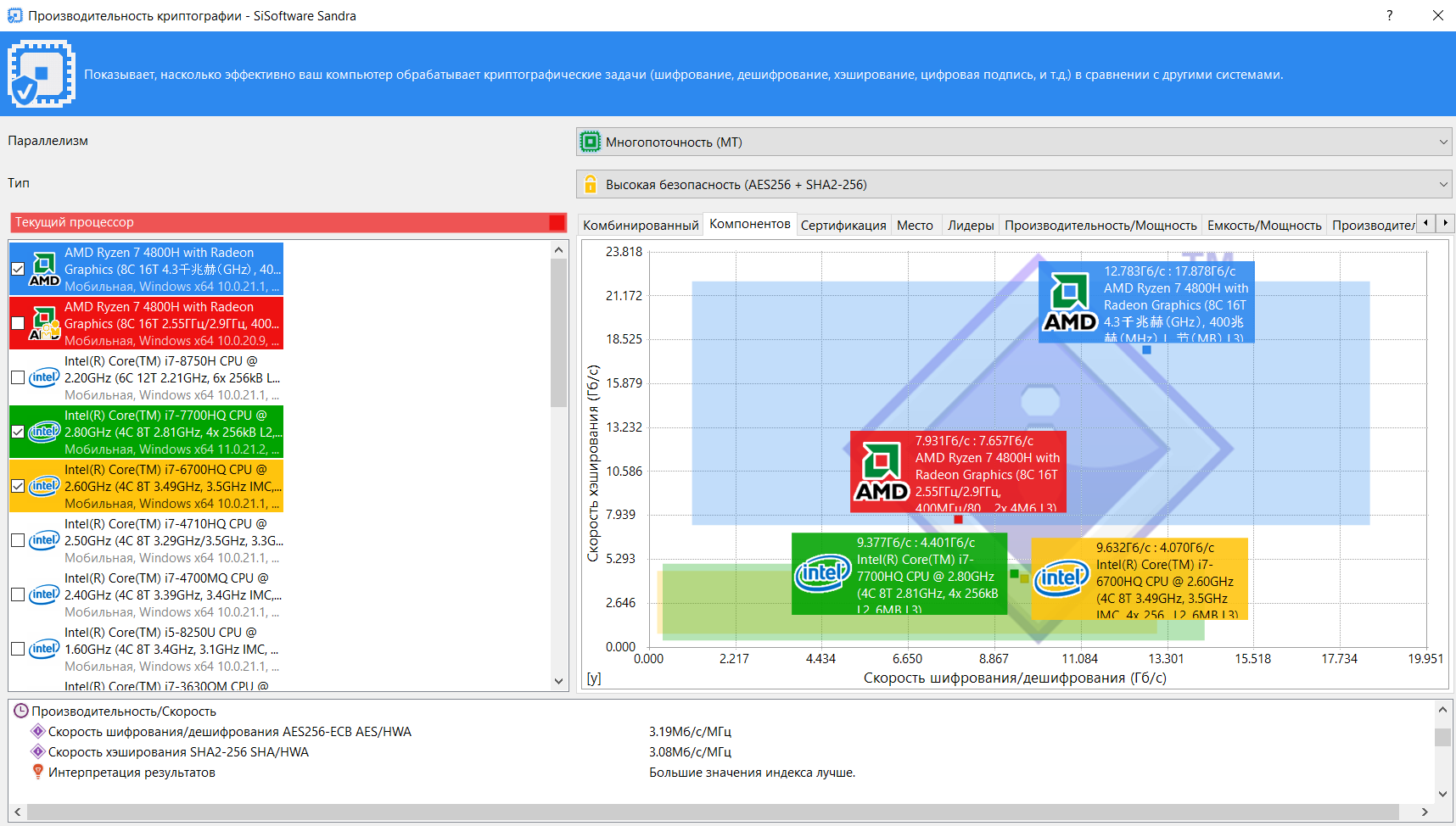
****

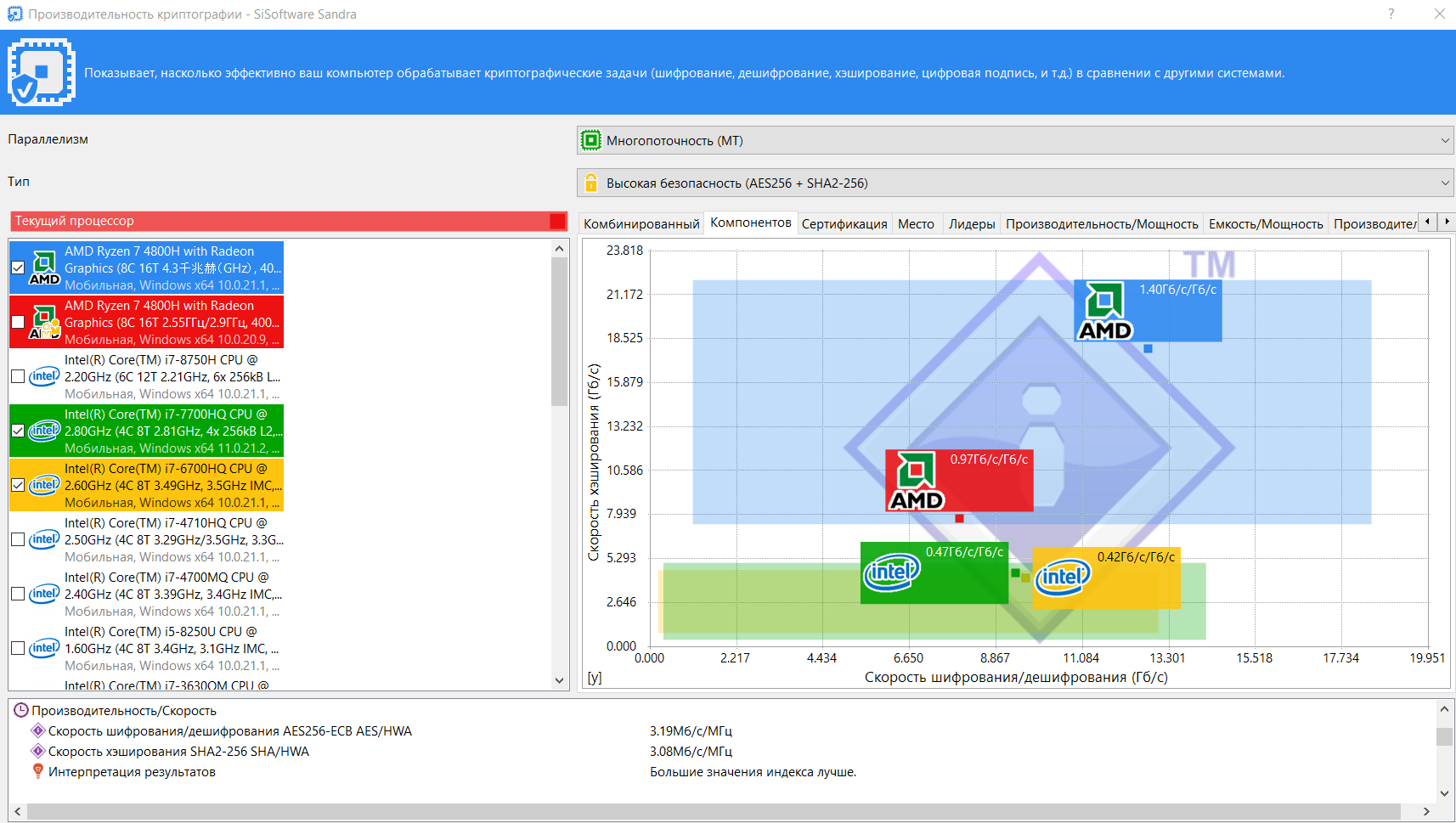
**Тест криптографии**

(тест показывает, насколько эффективно ваш компьютер обрабатывает криптографические задачи (шифрование, дешифрование, хэширование, цифровая подпись))



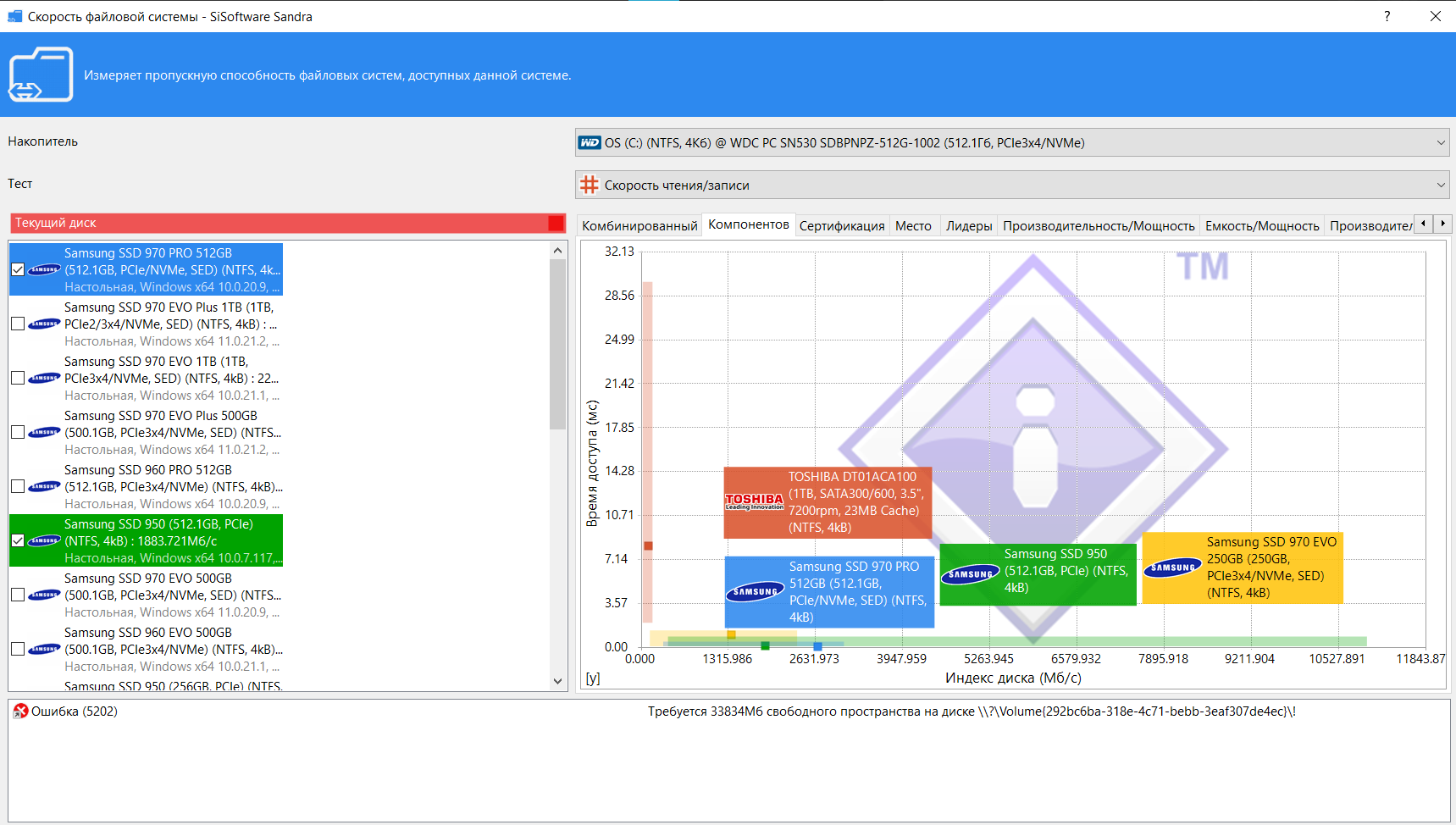


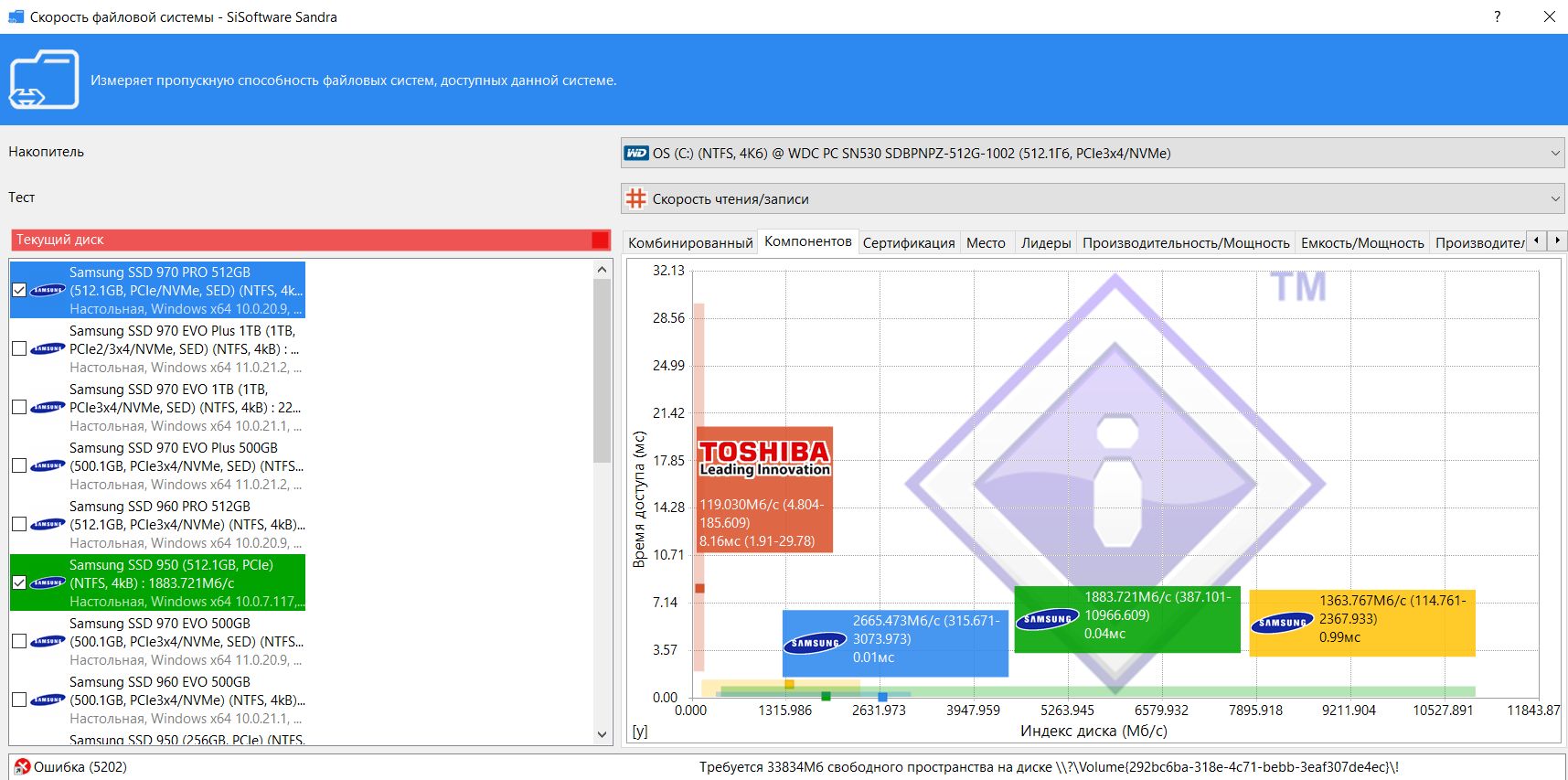
****

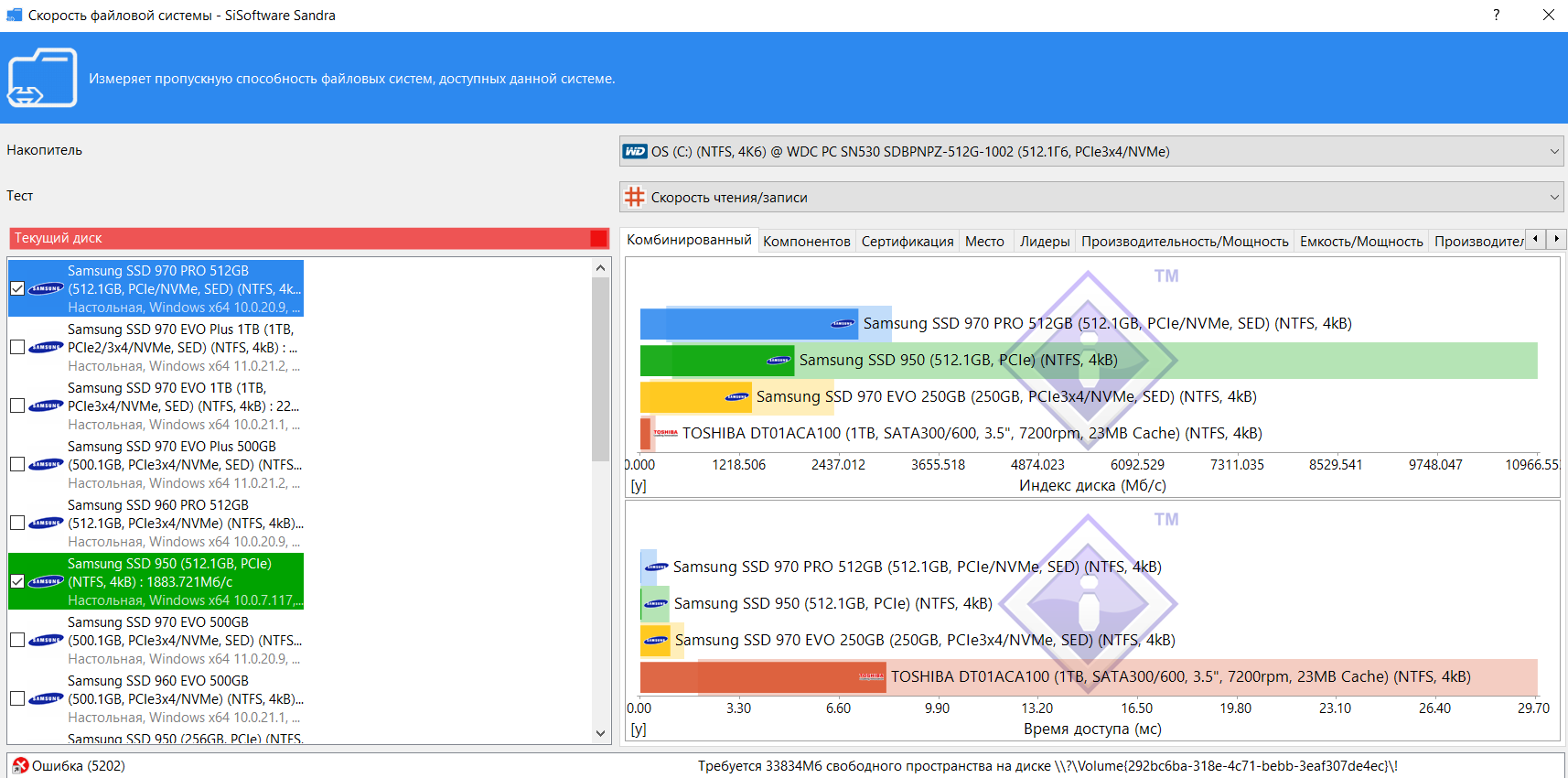
****

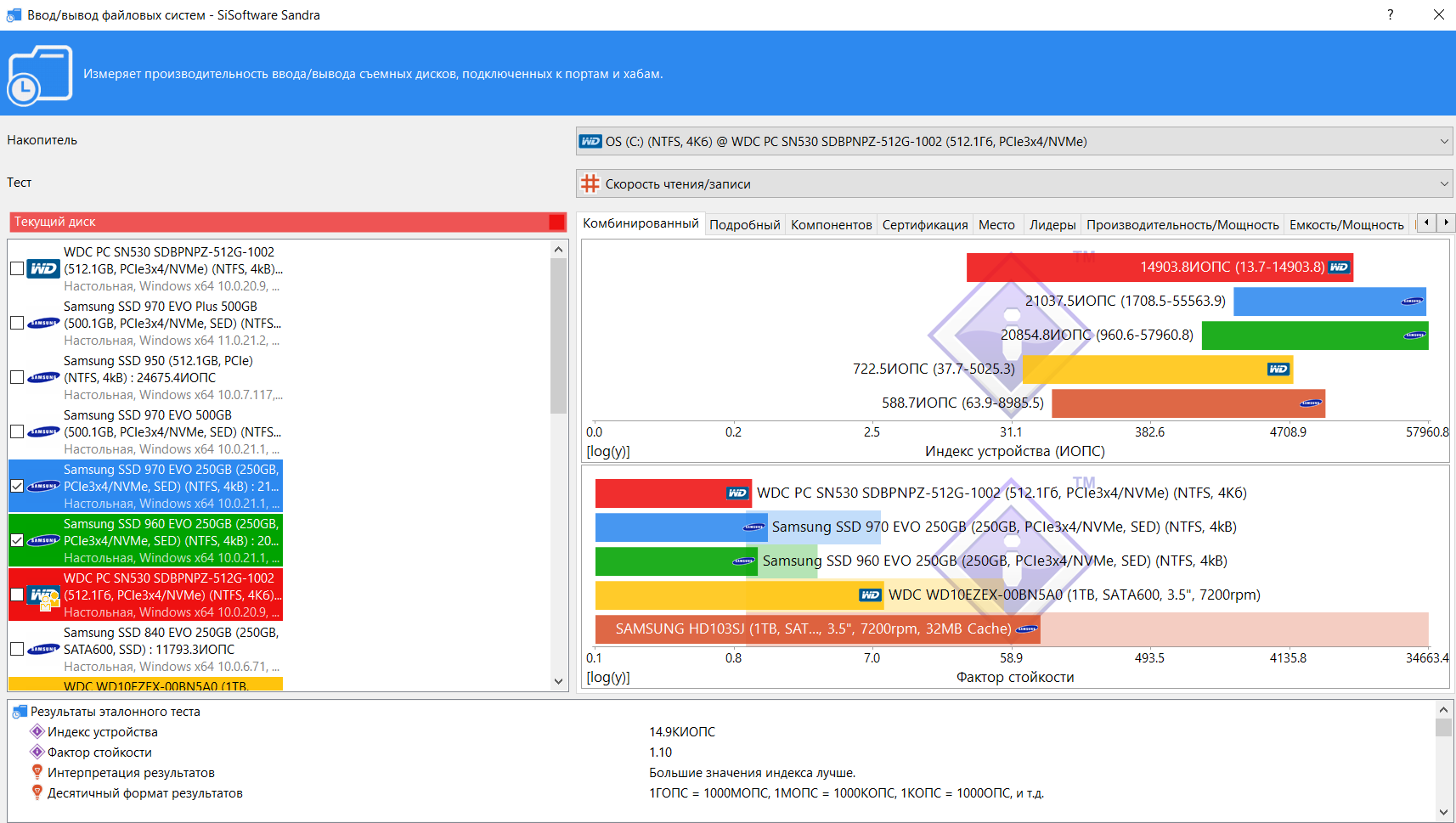
**Тест «Файловые системы»**

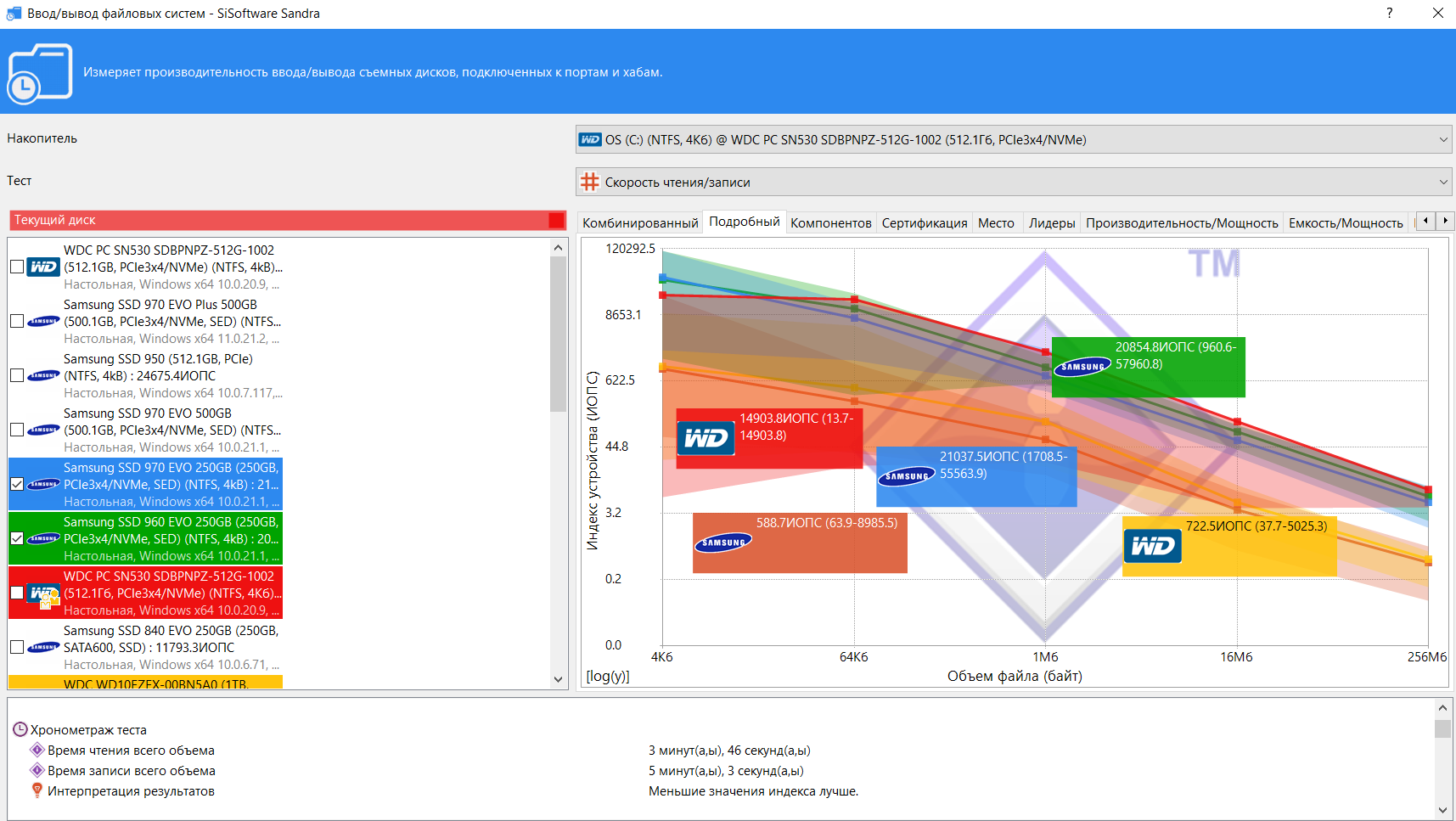
(позволяет определить производительность дисковой (файловой) подсистемы компьютера в сравнении с другими эталонными компьютерными системами)

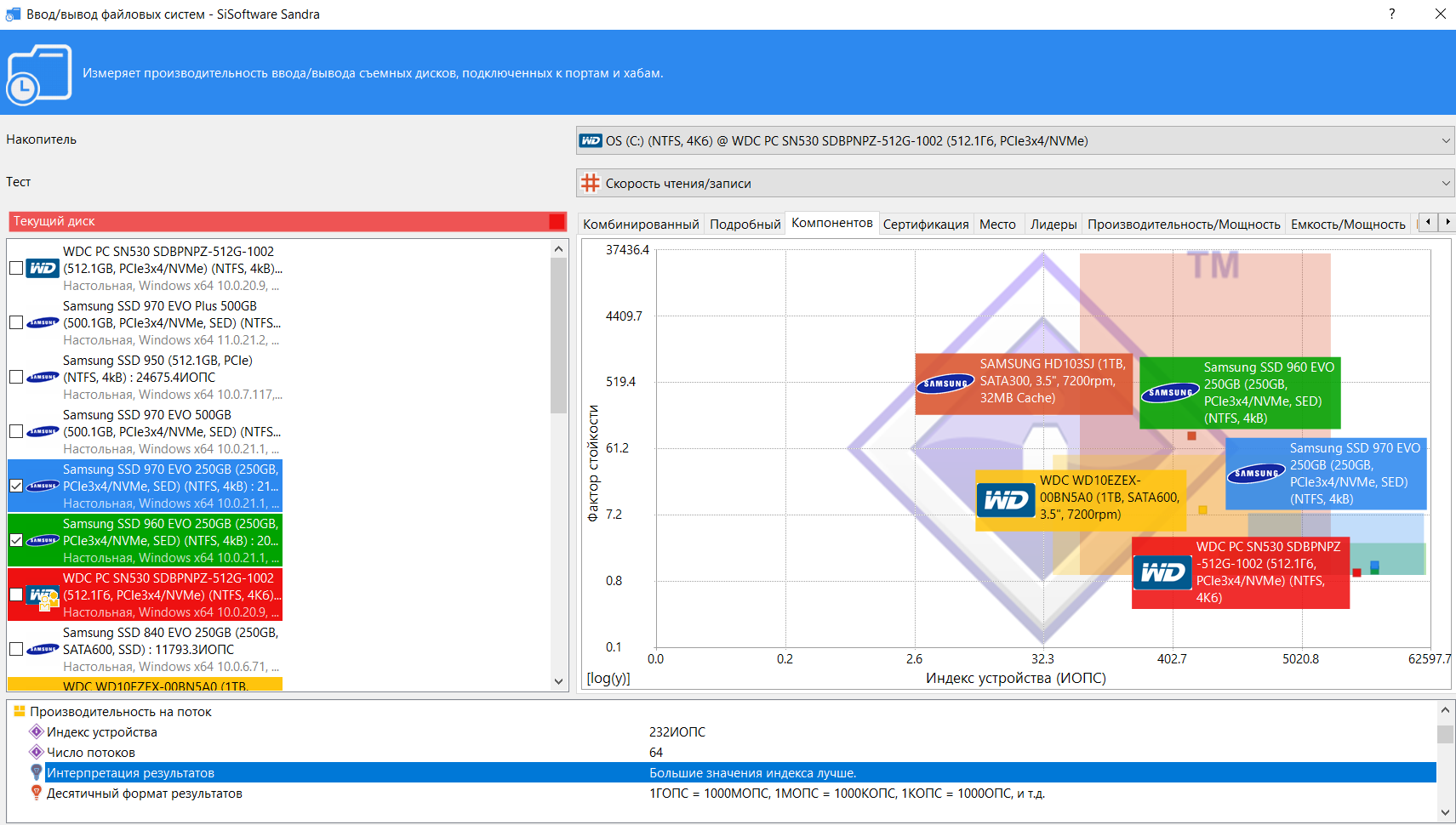
****

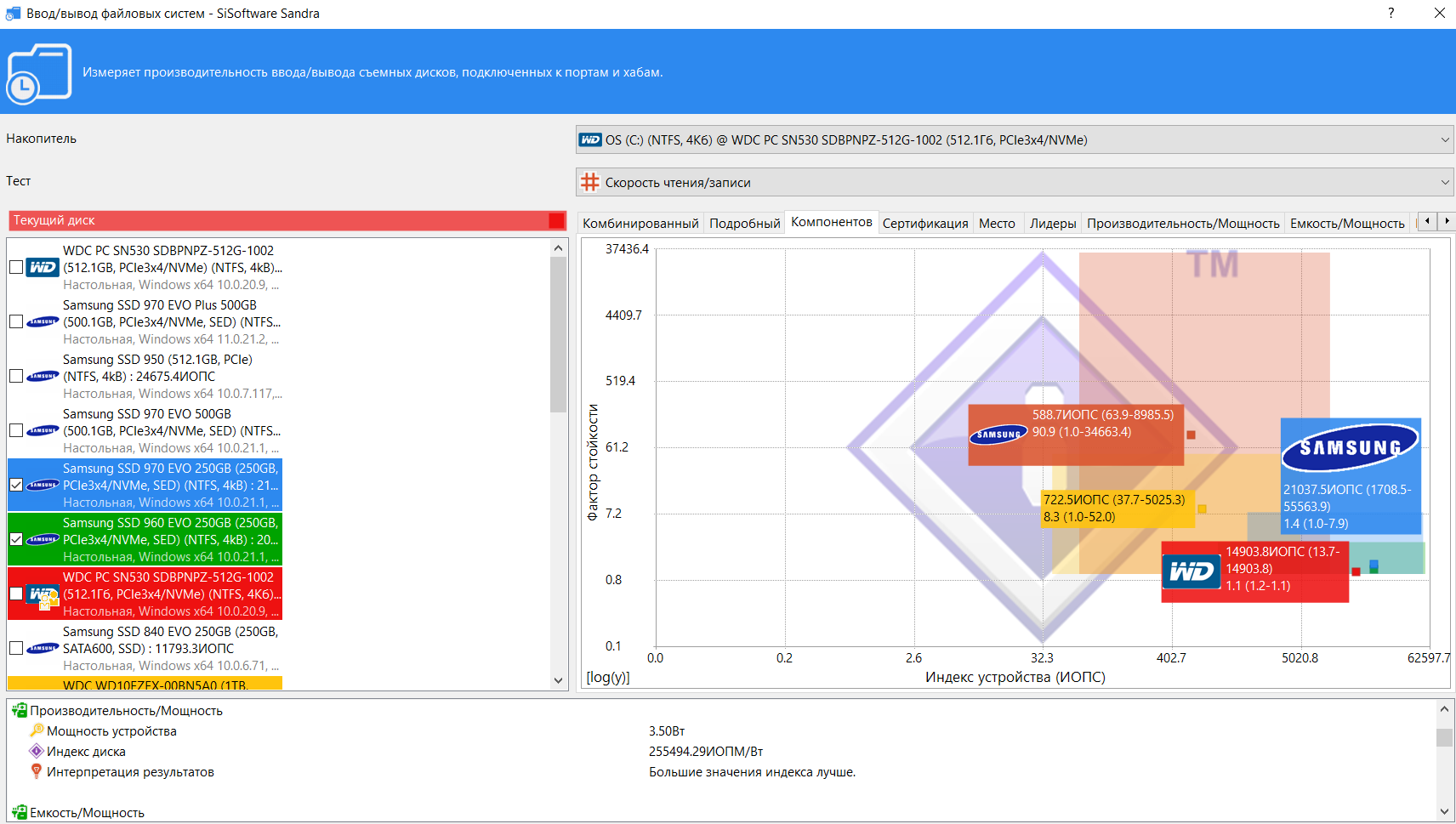
****

****

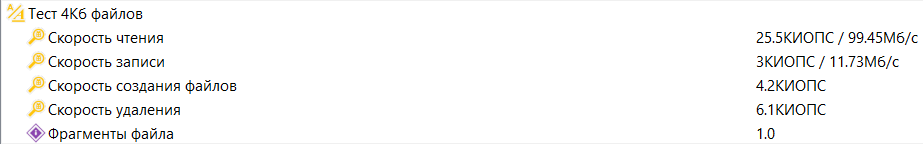
****

****

****

****

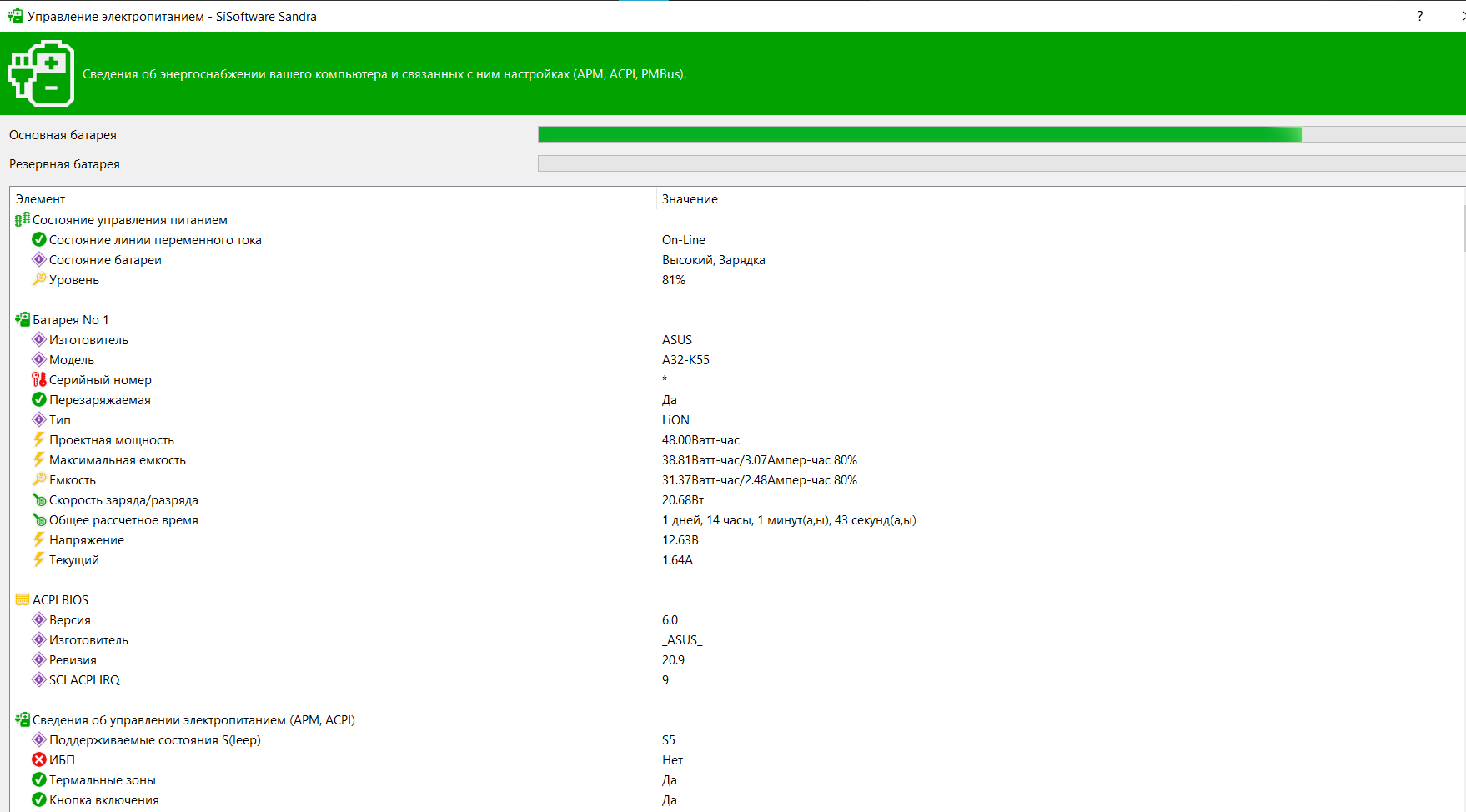
****

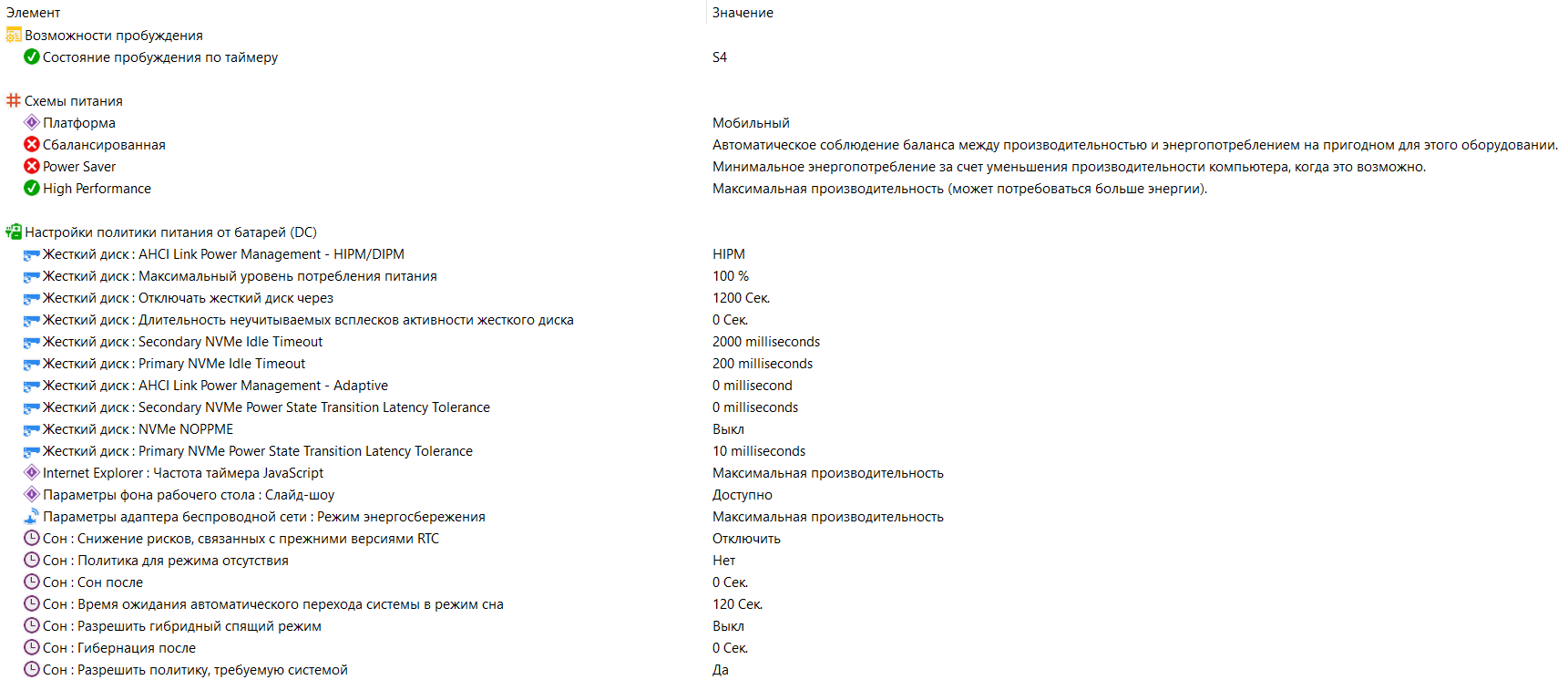
****

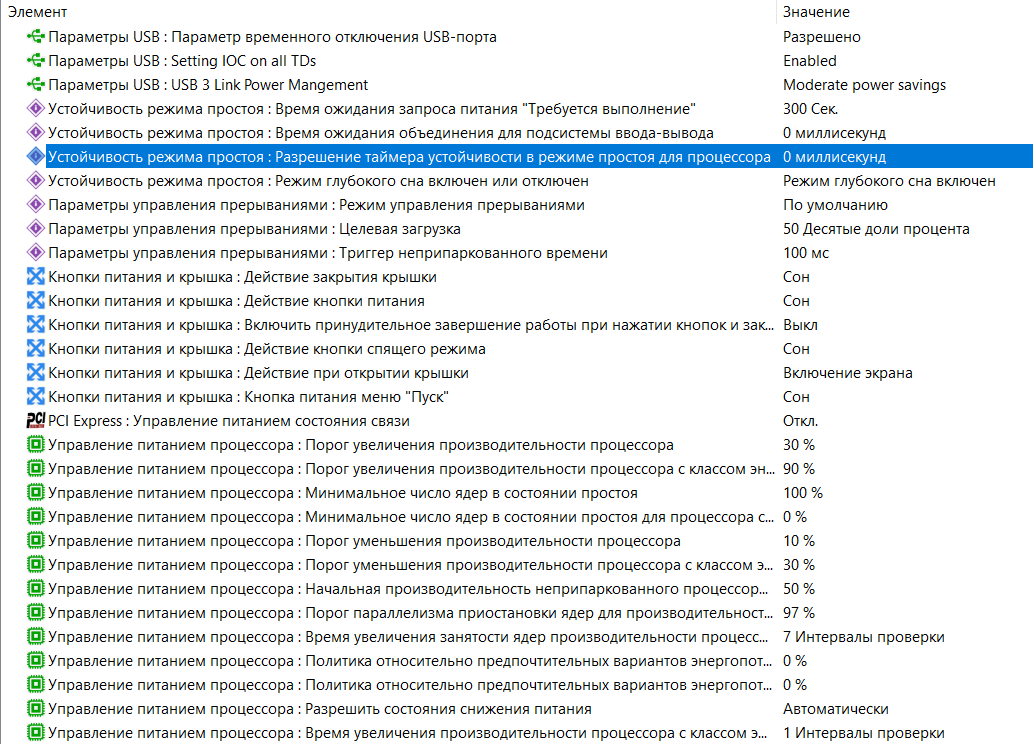
****

**Тест «Управление электропитанием»**

(дает возможность определить характеристики батареи)

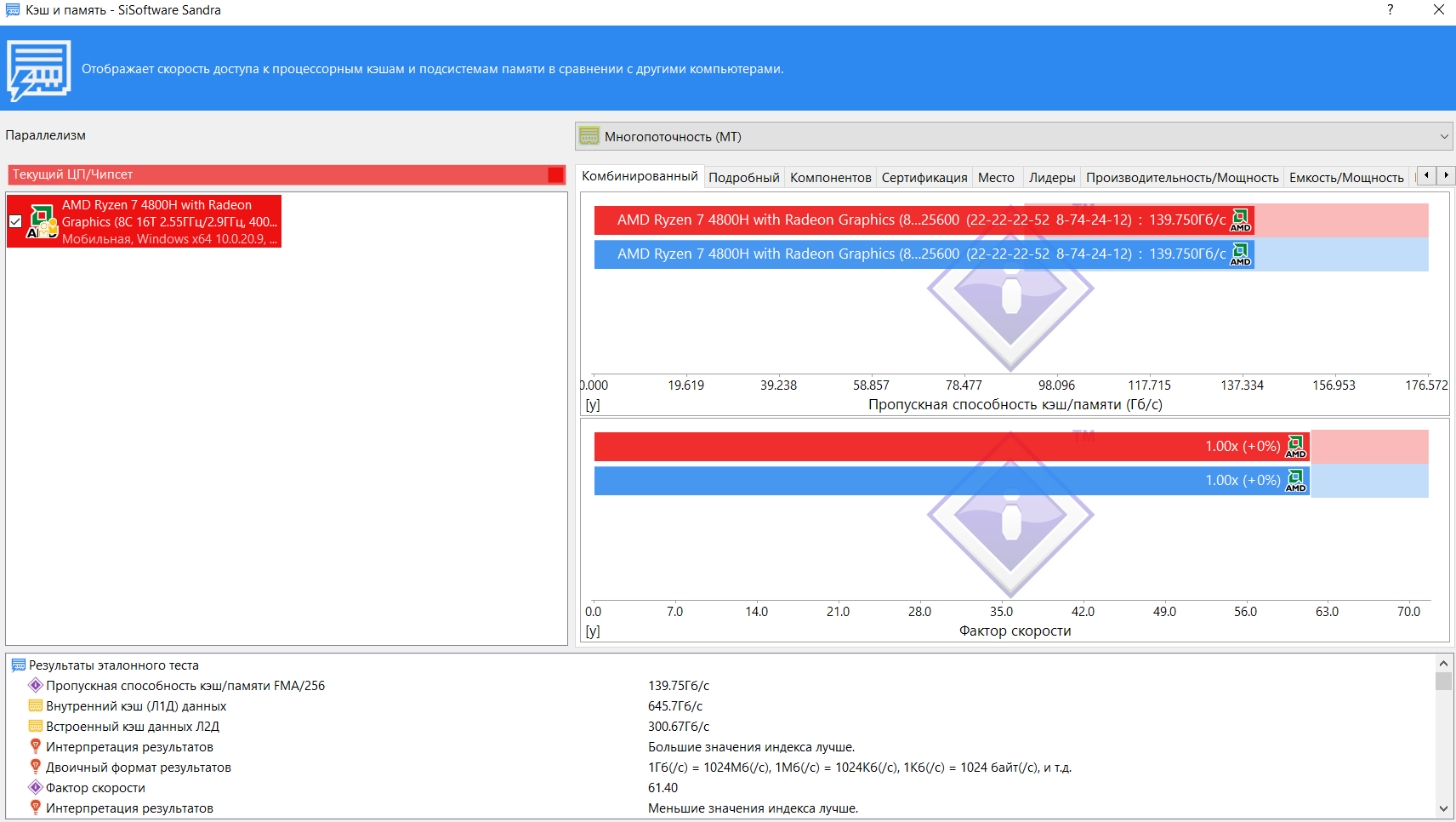
****

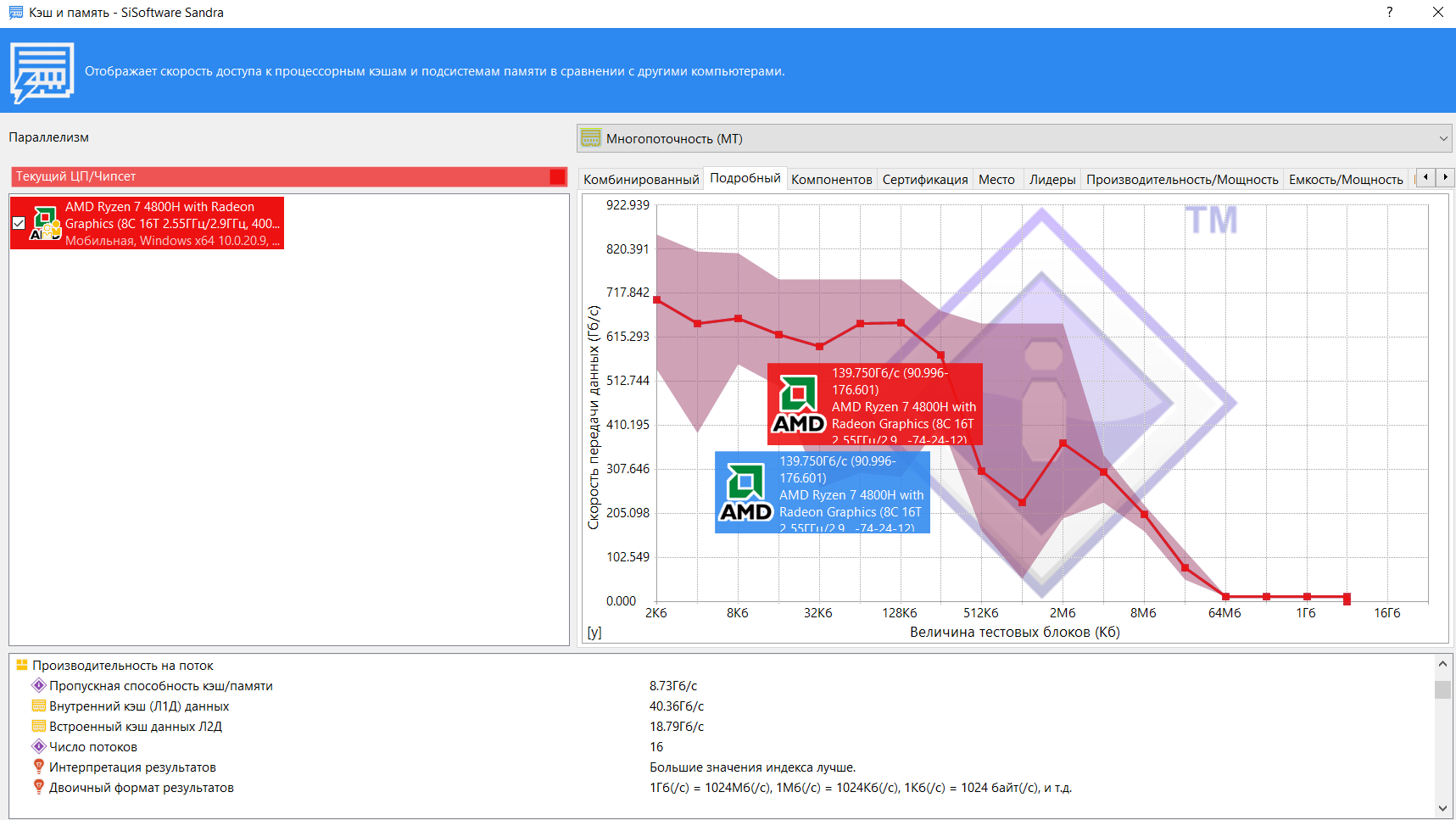
****

****

**Тест «Кэш и память»**

(дает возможность определить пропускную способность подсистемы памяти (связка «процессор — кэш — чипсет — память») в сравнении с другими эталонными компьютерными системами)



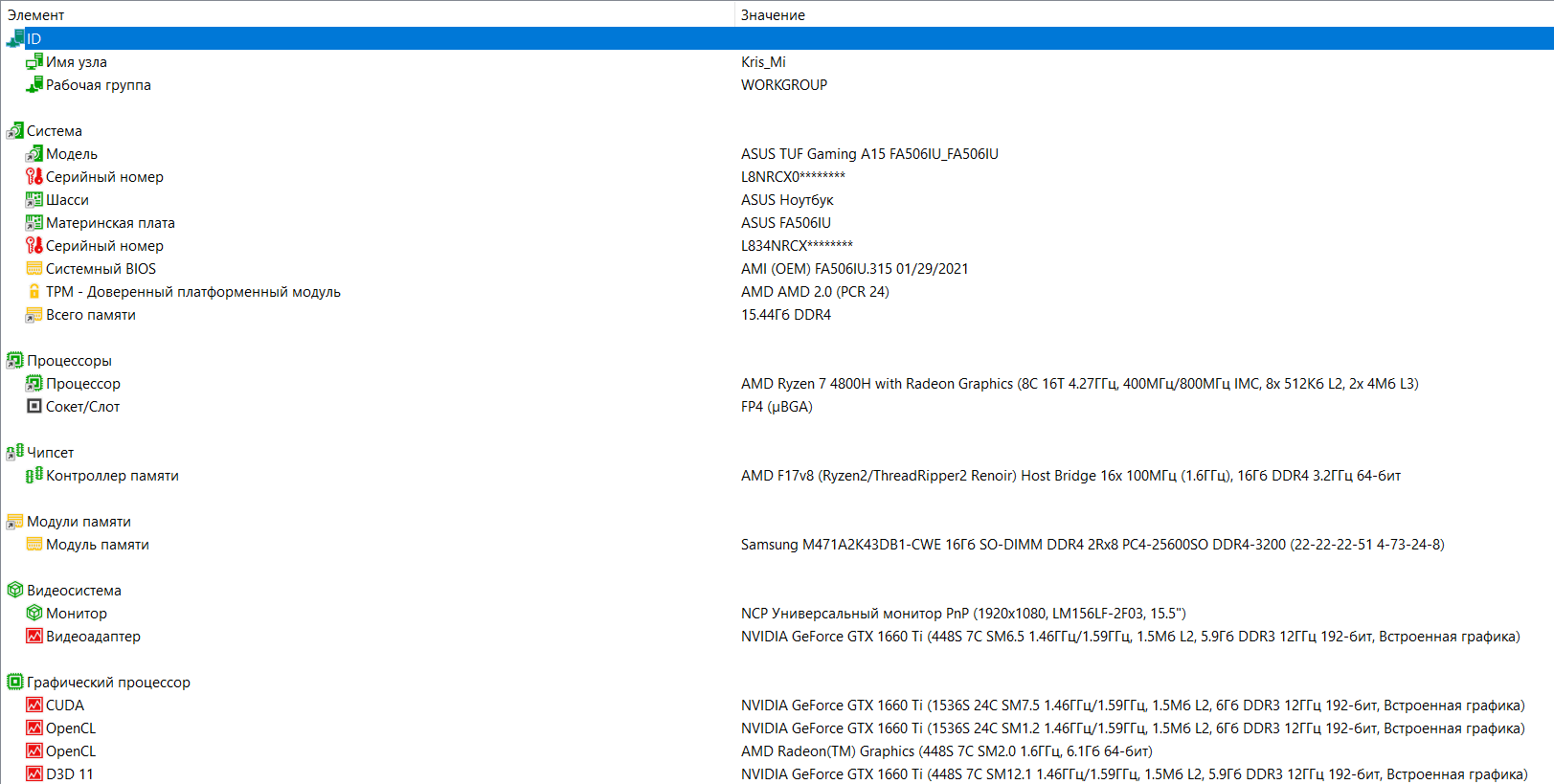


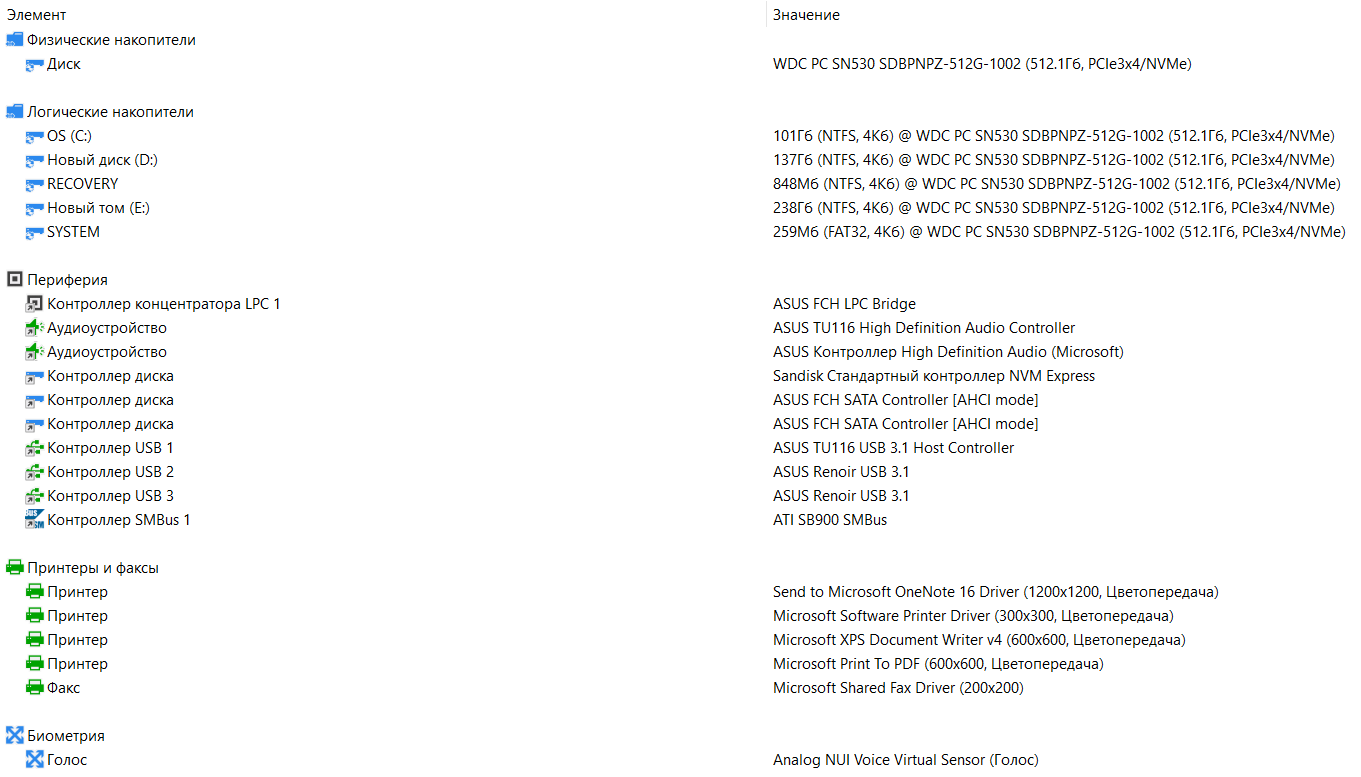


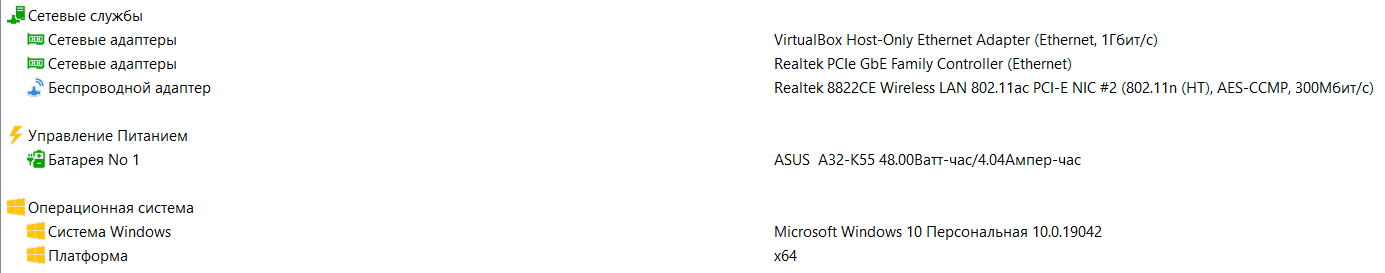




**Информация о системе**

****

****

****

Вывод: в ходе лабораторной работы нами были изучены существующие способы оценки производительности вычислительных машин и получены базовые навыки сравнения производительности вычислительных машин.

**Контрольные вопросы:**

1. Назовите основные факторы, влияющие на производительность ВМ.

Производительность ЦП зависит от трех параметров: такта (или частоты) синхронизации, среднего количества тактов на команду и количества выполняемых команд. Невозможно изменить ни один из указанных параметров изолированно от другого, поскольку базовые технологии, используемые для изменения каждого из этих параметров, взаимосвязаны:

* частота синхронизации определяется технологией аппаратных средств и функциональной организацией процессора;
* среднее количество тактов на команду зависит от функциональной организации и архитектуры системы команд;
* а количество выполняемых в программе команд определяется архитектурой системы команд и технологией компиляторов. Когда сравниваются две машины, необходимо рассматривать все три компоненты, чтобы понять относительную производительность.

Единственной подходящей и надежной единицей измерения производительности является время выполнения реальных программ.

1. Какие существуют тесты для оценки производительности, в чем их отличие?

В мире существует уже достаточно большой набор разработанных тестов, оценивающих различные аспекты производительности систем и подсистем. Наиболее известные из них: 007 (ODBMS), AIM, Dhrystone, Khornerstone, LFK (Livermore Loops), LINPACK, MUSBUS, NAS Kernels, Nhfsstone, PERFECT, RhosettaStone, SLALOM, SPEC, SSBA, TPC, WPI Benchmark Suite, Whetstone, Xstone, SYSmark, Stanford, IOBENCH, IOZONE, Byte, Netperf, Nettest, Hartstone, EuroBen, Fhourstones, Heapsort, Hanoi, Flops, C LINPACK, TFFTDP, Matrix Multiply (MM), Digital Review.

1. Как связана тактовая частота микропроцессора и производительность ВМ?

Быстродействие каждого персонального компьютера определяется производительностью его микропроцессора, т.е. количеством операций, выполняемых процессором в течение одной секунды. На производительность микропроцессора больше всего влияют такие характеристики, как разрядность и тактовая частота.

От тактовой частоты процессора зависит ритм работы всего компьютера. Производительность компьютера прямо пропорциональна тактовой частоте его процессора. **Чем выше тактовая частота, тем выше производительность компьютера.**

Тактовая частота выражает количество тактов, выполняемых микропроцессором за одну секунду. Один такт представляет определенный промежуток времени, необходимый процессору для выполнения одной элементарной операции. Процессор с тактовой частотой 3,2 ГГц выполняет 3,2 млрд. циклов в секунду.

Тактовая частота микропроцессора имеет свои единицы измерения: мегагерцы и гигагерцы.

Поскольку разные архитектуры процессоров обрабатывают команды по разному, лучше всего сравнивать тактовую частоту процессоров одной марки и одного поколения.

1. Что такое MIPS и MFLOPS? В чем их отличия?

**MIPS**

Одной из альтернативных единиц измерения производительности процессора (по отношению к времени выполнения) является MIPS — (*миллион целочисленных команд в секунду, millions of INsTRuctions per second*).

В общем случае MIPS есть скорость операций с целыми числами в единицу времени, т.е. для любой данной программы MIPS есть просто отношение количества команд в программе к времени ее выполнения. Таким образом, производительность может быть определена как обратная к времени выполнения величина, причем более быстрые машины при этом будут иметь более высокий рейтинг MIPS.

Положительными сторонами MIPS является то, что эту характеристику легко понять, особенно покупателю, и что более быстрая машина характеризуется большим числом MIPS, что соответствует нашим интуитивным представлениям. Во-первых, MIPS зависит от набора команд процессора, что затрудняет сравнение по MIPS компьютеров, имеющих разные системы команд. Во-вторых, MIPS даже на одном и том же компьютере меняется от программы к программе. В-третьих, MIPS может меняться по отношению к производительности в противоположенную сторону.

**MFLOPS**

Обычно для научно-технических задач производительность процессора оценивается в MFLOPS (*миллионах чисел-результатов вычислений с плавающей точкой в секунду, или миллионах элементарных арифметических операций над числами с плавающей точкой, выполненных в секунду, millions of floatINg-poINT operations per second*).

Как единица измерения, MFLOPS, предназначена для оценки производительности только операций с плавающей точкой, и поэтому не применима вне этой ограниченной области.

Рейтинг MFLOPS зависит от машины и от программы. Этот термин менее безобидный, чем MIPS. Он базируется на количестве выполняемых операций, а не на количестве выполняемых команд. По мнению многих программистов, одна и та же программа, работающая на различных компьютерах, будет выполнять различное количество команд, но одно и то же количество операций с плавающей точкой. Именно поэтому рейтинг MFLOPS предназначался для справедливого сравнения различных машин между собой.

1. Какая организация занимается разработкой тестовых программ?

**Независимая тестовая организация** — организация, которая проводит тестирование продуктов, материалов, программ и др. на соответствие принятым требованиям. Организация может быть полностью независимой (независимая тестовая лаборатория), либо подчиненной государству или научной организации. Под «независимостью» здесь понимается отсутствие влияния производителя товара на результаты тестирования.

Некоторые крупнейшие независимые тестовые организации:

* РОССТАНДАРТ
* NTA Inc
* IBM Product Test
* Underwriters Laboratories Inc. — тестирование на безопасность. Одна из старейших независимых тестовых организаций (основана в 1894 году)
* Европейское агентство по безопасности продуктов питания
* NSF International — тестирование экологической безопасности продуктов
* QATestLab — тестирование программного обеспечения
* Stiftung Warentest — крупнейшая независимая тестовая организация Германии

1. Назовите два базовых набора тестов? В чем их отличия?

В настоящее время имеется два базовых набора тестов **SPEC**, ориентированных на интенсивные расчеты и измеряющих производительность процессора, системы памяти, а также эффективность генерации кода компилятором. Как правило, эти тесты ориентированы на операционную систему UNIX, но они также импортированы и на другие платформы.

* Набор тестов **CINT92**, измеряющий производительность процессора при обработке целых чисел, состоит из шести программ, написанных на языке Си и выбранных из различных прикладных областей: теория цепей, интерпретатор языка Лисп, разработка логических схем, упаковка текстовых файлов, электронные таблицы и компиляция программ.
* Набор тестов **CFP92**, измеряющий производительность процессора при обработке чисел с плавающей точкой, состоит из 14 программ, также выбранных из различных прикладных областей: разработка аналоговых схем, моделирование методом Монте-Карло, квантовая химия, оптика, робототехника, квантовая физика, астрофизика, прогноз погоды и другие научные и инженерные задачи. Две программы из этого набора написаны на языке Си, а остальные 12 - на Фортране. В пяти программах используется одинарная, а в остальных — двойная точность.

1. Чем занимается организация TPC?

По мере расширения использования компьютеров при обработке транзакций в сфере бизнеса все более важной становится возможность справедливого сравнения систем между собой. С этой целью в 1988 году был создан **Совет по оценке производительности обработки транзакций (TPC — Transaction Processing Performance Council)**, который представляет собой бесприбыльную организацию.

TPC публикует спецификации тестовых пакетов, которые регулируют вопросы, связанные с работой тестов. Эти спецификации гарантируют, что покупатели имеют объективные значения данных для сравнения производительности различных вычислительных систем. Сами спонсоры, объявляя результаты TPC, должны представить TPC детальные отчеты, документирующие соответствие всем спецификациям. Эти отчеты, в частности, включают конфигурацию системы, методику калькуляции цены, диаграммы значений производительности и документацию, показывающую, что тест соответствует требованиям атомарности, согласованности, изолированности и долговечности (ACID — atomicity, consistency, isolation, and durability), которые гарантируют, что все транзакции из оценочного теста обрабатываются должным образом.

1. Деятельность организации AIM.

Одной из независимых организаций, осуществляющей оценку производительности вычислительных систем, является частная компания **AIM Technology**, основанная в 1981 году. Компания разрабатывает и поставляет программное обеспечение для измерения производительности систем, а также оказывает услуги по тестированию систем конечным пользователям и поставщикам вычислительных систем и сетей, которые используют промышленные стандартные операционные системы, такие как UNIX и OS/2.

За время своего существования компания разработала специальное программное обеспечение, позволяющее легко создавать различные рабочие нагрузки, соответствующие уровню тестируемой системы и требованиям по ее использованию. Это программное обеспечение состоит из двух основных частей: генератора тестовых пакетов (Benchmark Generator) и нагрузочных смесей (Load Mixes) прикладных задач.

1. В каких случаях используются тесты Whetstone?

Краткие описания эталонных программ (**benchmarks**) — тестов, используемых для оценки характеристик микропроцессоров и ЭВМ. Большинство описываемых тестов позволяет оценить эффективность векторизующих и оптимизирующих компиляторов.

**Whetstone.** Тест спроектирован для оценки характеристик систем при интенсивном использовании операций с плавающей запятой. Это смесь плавающих и целочисленных операций, вычисления трансцедентных функций, обработки массивов и вызовов процедур. Тест основан на статистике научных программ, написанных на Фортране, имеет небольшой размер и помещается в КЭШ. Результат выражается в Mwhet/s (Millions Whetstone per Second).

**10.** В каких случаях используются тесты Dhrystone?

**Dhrystone**. Этот стандартный тест спроектирован для изменения как характеристик аппаратуры, так и эффективности компиляторов. Он включает с определенным весом вызовы процедур, циклы, логические и целочисленные арифметические операции. результат выражается в D/s (Dhrystone per second).