Греп Пайп



Гантова зебра

Мыслить на уровне физики диска

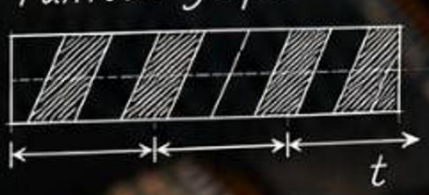
Часть 2. Цена перемещения



- ☑ Механика
- ☑ LBA и NTFS
- ☑ Измерения
- ☑ Эксперимент

№001–№011 — программные измерения на реальном диске

HGST HTS725050B7E630 • HDD 2.5" • 500 ГБ • NTFS • USB 3.x



PEGINELAB

2026

Греп Пайп

Гантова зебра ч2

«Автор»
2026

Пайп Г.

Гантова зебра ч2 / Г. Пайп — «Автор», 2026

В первой части «Гантовой зебры» я предложил идею адаптивного размещения данных на жёстком диске. Теперь пришло время проверить, как HDD ведёт себя в реальных условиях.Почему скорость чтения меняется в зависимости от области диска? Сколько времени занимают переходы между логическими адресами? Как на результаты влияют интерфейс подключения, размер запроса, порядок чтения и кэширование?Чтобы разобраться, я провёл серию лабораторных экспериментов на жёстком диске HGST, исследовал задержки доступа, особенности файловой системы NTFS и поведение накопителя при различных сценариях чтения.Во второй части показываю результаты измерений, объясняю обнаруженные закономерности и продолжаю исследовать возможности «Гантовой зебры».Это не рассказ о гарантированном ускорении HDD, а следующий этап инженерной работы — переход от теоретической идеи к экспериментам, фактам и новым вопросам.

© Пайп Г., 2026
© Автор, 2026

Содержание

От автора. Почему я выбрал жёсткий диск 5
Вступление. Зачем исследовать HDD сегодня? 7
Глава 1. Железо, которое умеет помнить 10
1.1. Что на самом деле лежит в металлической коробке 11
1.2. Пластина — не просто металлический круг 12
1.3. Шпиндельный двигатель — хранитель ритма 13
1.4. Актуатор: почему головка ездит не сама по себе 14
1.5. Головка работает почти без прикосновения 15
1.6. Сервосистема: невидимые дорожные знаки 16
1.7. Контроллер: почему диск — это ещё и компьютер 17
1.8. Что происходит, когда мы открываем фотографию 19
1.9. Время чтения — не только движение каретки 20
1.10. Наш конкретный HDD — и почему паспорт не равен измерению 21
1.11. Что действительно полезно владельцу домашнего архива 22
1.12. Что мы узнали и к чему перейдём дальше 23
Глава 2. Где на самом деле находятся данные? 24
2.1. Фотография в папке и фотография на пластине — не одно и то же 25
2.2. Что означает «намагниченная область» 26
2.3. Дорожка, сектор и зона — что между ними общего? 27
2.4. Что такое LBA, если это не координата? 29
2.5. Почему 512 байт и 4096 байт могут означать один сектор 30
2.6. От LBA к разделу, от раздела к NTFS 31
2.7. Что NTFS знает о файле, а чего не знает 33
2.8. Кластеры, VCN и LCN: два набора номеров 34
2.9. Пример на обычной фотографии: от байтов к адресам 36
2.10. Почему близкие LBA могут вести себя неожиданно 37
2.11. Зачем здесь нужна «Гантова зебра» 38
2.12. Что это означает для владельца домашнего архива 40
2.13. Подведём итог: карта, которую мы видим, и карта, которую скрывает диск 41
Глава 3. Почему один участок диска быстрее другого? 42
3.1. Пять фотографий, одна папка и один неожиданный вопрос 43
3.2. Наш лабораторный образец и что значит «настоящий диск» 44
3.3. Что означает «точка 5 %» и почему это не радиус пластины 45
3.4. Почему размер каждого чтения составил 16 МиБ 46
3.5. Опыт № 001: пять зон, три прохода 47
Конец ознакомительного фрагмента. 48

Греп Пайп
Гантова зебра ч2

От автора. Почему я выбрал жёсткий диск

Иногда большая техническая идея начинается с совершенно обычного желания — сохранить фотографии.

Не создать новый компьютер, не изобрести очередную технологию хранения и даже не ускорить накопитель. Просто сохранить то, что со временем становится дорогим: семейные снимки, домашние видео, документы, собственные проекты и компьютерные программы.

Когда-то для этого хватало бумажного фотоальбома. Я и сейчас не против таких альбомов. У напечатанной фотографии есть преимущество, о котором иногда забывают: её можно показать без компьютера, зарядного устройства и специального приложения. Но попробуйте поместить в такой альбом домашнее видео, исходники своих программ и тысячи снимков за двадцать лет. Бумага здесь уже не справится.

Сегодня накопители выбирают по скорости, ёмкости и цене. Но я поставил вопрос иначе: каким должно быть хранилище, если большую часть времени оно будет молча беречь архив, а спустя годы позволит к нему вернуться? Мне нужен не рекордсмен по скорости запуска игр. Мне нужна возможность сохранить воспоминания и собственную работу.

За десять лет компьютер может смениться несколько раз. Тем временем фотография остаётся фотографией, текстовый файл — текстовым файлом, а исходный код старой программы нередко можно открыть современным редактором. Разумеется, не всякая программа запустится в новой операционной системе и не каждый формат будет поддерживаться вечно. Поэтому я стараюсь думать не только о носителе, но и о понятных форматах, резервных копиях и возможности проверить сохранённые данные.

Когда я сравнивал технологии, в каждой обнаруживалось своё достоинство. Магнитная лента привлекательна для больших архивов, но домашнему пользователю нередко оказывается не по карману сам привод чтения и записи. Картридж — лишь часть стоимости. Нужны совместимый привод, обслуживание, а порой и продуманная система смены поколений оборудования. Для крупной архивной службы это оправданно. Для семейной коллекции фотографий — уже не всегда.

Твердотельные накопители SSD, в том числе распространённые модели M.2, быстры, бесшумны и замечательно подходят для операционной системы, программ и рабочей обработки файлов. Но архив, который открывают редко, не всегда нуждается в такой скорости. При одинаковой вместимости SSD часто обходятся дороже HDD. И отсутствие двигателя не делает цифровые данные бессмертными: старение памяти, качество контроллера и условия хранения по-прежнему имеют значение.

USB-флешка полезна для переноса данных, а компактные карты памяти спасают фотоаппарат или телефон. Однако для большого архива стоимость на единицу объёма и различия в качестве устройств становятся существенными. Хранить единственный экземпляр семейного архива на одной флешке я бы не стал.

Оптические диски — ещё один путь. Существуют обычные CD, DVD, Blu-ray и специальные архивные разновидности. Некоторые из них используют более стойкие неорганические записывающие слои. Но и здесь важно не путать заявленную долговечность с гарантией: всё зависит от носителя, привода, режима записи, условий хранения и того, останется ли через много лет подходящее устройство чтения. Для большого архива потребуется много дисков.

Облако удобно, если нужно получить доступ к файлам из разных мест. Но оно связано с интернетом, правилами поставщика, платежами и контролем доступа. Для меня облако разумнее рассматривать как одну из копий, а не как единственное место существования фотографий.

Так я вернулся к магнитному HDD. Не потому, что он совершенен. У него есть механика, которая изнашивается, электроника, которая может выйти из строя, и вполне реальные риски потери данных. Но он способен хранить значительные объёмы без дорогого внешнего ленточного привода и обычно хорошо подходит для домашнего архива по соотношению ёмкости и стоимости. При этом важную информацию необходимо иметь как минимум в нескольких независимых копиях, желательно в разных местах, и периодически проверять её читаемость.

Выбор носителя привёл меня к неожиданному увлечению. Ведь внутри неприметной коробки вращаются пластины, над ними работают миниатюрные магнитные головки, специальный механизм быстро переводит их в нужную область, а электроника заставляет всё это выглядеть для компьютера как простая последовательность адресов. Пользователь видит файл и дважды щёлкает мышью. Внутри же разворачивается целая механическая и электронная история.

Меня заинтересовало, насколько важны соседство данных, порядок обращений и цена переходов между логическими участками. Так постепенно возникла идея «Гантовой зебры» — представление о чередовании и группировке данных, которое можно нарисовать в виде полос и временных диаграмм. Но красивый рисунок сам по себе не доказывает, что диск станет быстрее.

Чтобы проверить идею, понадобились настоящие эксперименты. Мы использовали физический HDD HGST на 500 ГБ, создавали небольшие программы, сравнивали обращения к различным логическим адресам и фиксировали время выполнения. Это были программные измерения на реальном устройстве — не телеметрия перемещения магнитной головки и не реконструкция скрытой геометрии пластин.

Результаты иногда подтверждали ожидания, иногда нет. Особенно любопытными оказались сверхбыстрые ответы на отдельные запросы и долгие задержки перед первым чтением. Они заставили внимательнее относиться к кэшу, USB-мосту, операционной системе и самой методике измерения. Именно так и должна работать инженерная любознательность: если наблюдение не укладывается в удобное объяснение, надо проверить объяснение, а не выбросить неудобную цифру.

Эта книга написана для человека, который умеет пользоваться компьютером и хочет понимать, что происходит внутри. Здесь не требуется программировать драйверы или знать устройство сервосистемы. Мы начнём с двигателя, пластин и головок; затем перейдём к адресам, файлам и реальным измерениям. Там, где нужна схема, я покажу схему. Там, где остаётся неизвестное, так и скажу.

Мне интересна не гонка с SSD и не обещание чудесного ускорения старого HDD. Мне интереснее научиться разумно обращаться с технологией и видеть за простым действием «открыть файл» физическую работу сложной машины. Всё началось с желания сохранить фотографии и программы. А дальше появилось любопытство: где именно живёт файл и что происходит внутри диска, когда я прошу его прочитать?

Вступление. Зачем исследовать HDD сегодня?

Начнём с простого вопроса: почему исправный жёсткий диск может читать один файл быстро, а другой — заметно медленнее?

Кажется, ответ должен быть очевиден. Диск вращается с постоянной скоростью; значит, достаточно узнать частоту вращения, скорость передачи и время движения головки. Но любая попытка измерить реальную работу HDD показывает, что эти величины переплетены сложнее, чем кажется.

Магнитная пластина — это не длинная лента с файлами, разложенными на поверхности в алфавитном порядке. Современный накопитель хранит данные в дорожках и секторах, пользуется специальными областями для служебной информации и собственными механизмами исправления ошибок. Компьютер не сообщает головке, к какому радиусу подойти. Он передаёт логический номер блока, а контроллер сам переводит запрос в действия, необходимые для его выполнения.

Кроме того, HDD почти никогда не работает в одиночестве. Между программой и пластиной находятся файловая система, операционная система, драйверы, интерфейс подключения, собственный контроллер накопителя и его буфер. Когда программа сообщает, что чтение завершено за десять миллисекунд, это не означает, что ровно десять миллисекунд двигалась каретка. А если запрос завершился быстрее одной миллисекунды, это ещё не доказательство, что головка успела совершить механическое перемещение с такой скоростью.

Мы могли бы начать с теории и придумать схему размещения, в которой обращения становятся «короче» на нарисованной карте. Но это было бы слишком просто: сама карта скрыта от обычной программы. Поэтому в этой книге мы прежде всего уточняем границы наблюдаемого. Что можем измерить? Какие свойства видим лишь косвенно? Что нельзя узнать по одной последовательности логических адресов?

Здесь важно различить три вида утверждений. Первые — факты об устройстве HDD, известные из физики и технической документации. Вторые — то, что мы действительно измерили на своём стенде. Третьи — гипотезы и модели, которые помогают объяснять увиденное, но требуют отдельной проверки. Мы не будем смешивать их, даже если так текст казался бы эффектнее.

Для опытов был выбран портативный магнитный диск HGST Travelstar Z7K500.B, модель HTS725050B7E630. Его паспортная ёмкость — 500 ГБ, скорость вращения — 7200 оборотов в минуту, внутренний интерфейс — SATA. По документации изготовителя это диск семейства Advanced Format: физические сектора размером 4096 байт представлены компьютеру как логические сектора по 512 байт. В наших опытах накопитель использовался как внешний USB-диск через переходные мосты.

Эта деталь важна. USB — не родной язык магнитной пластины. Между ноутбуком и SATA-контроллером накопителя появляется дополнительное устройство, а вместе с ним — собственные ограничения и задержки. Поэтому мы не будем приписывать каждую разницу во времени только головке или мотору. В одной из глав отдельно разберём, как смена USB-моста меняет наблюдаемую скорость.

Наши эксперименты получили номера № 001–011. Под большинством номеров скрываются физические чтения настоящего HDD: разные зоны логического пространства, размеры блоков, близкие и дальние адреса, упорядоченные и перемешанные запросы. Но не все номера означают натурный опыт. Например, программа планирования синтетической нагрузки и исследование метаданных файловой системы относятся к другим уровням наблюдения. В книге это будет отмечено, чтобы не превращать модель в доказанный эксперимент.

Отдельная линия — сама «Гантова зебра». Это рабочее название идеи чередования и группировки участков данных, вдохновлённой графическим представлением последовательностей действий. Но магнитная запись в HDD не обязана подчиняться нашей нарисованной «зебре», а контроллер не раскрывает полной физической карты расположения адресов. На этой границе важно остановиться и честно сказать: оптимизацию физического размещения данных мы пока не проверяли.

Итак, книга пойдёт от простого к сложному. Сначала рассмотрим устройство настоящего HDD. Затем разберём, как программа видит диск и что означают адреса. После этого появятся измерения — с вопросом, графиком и объяснением каждого результата. В конце мы вернёмся к «Гантовой зебре» и посмотрим, что осталось от первоначальной идеи после встречи с реальной машиной.

Это книга не о том, что механический HDD обязательно лучше новых технологий. Это книга о том, как устройство становится понятнее, когда задаёшь ему правильные вопросы. И о том, почему даже неудавшаяся гипотеза может оказаться полезной, если измерение выполнено честно.



Эту цепочку и измеряет программный эксперимент в целом.

Из одного значения задержки нельзя отдельно получить время движения головки.

Важно: LBA — логический адрес, а не карта магнитных дорожек конкретного HDD.

Рисунок B.1. Между файлом и магнитной поверхностью лежит несколько самостоятельных уровней. Стрелки показывают путь запроса, а не точную физическую карту диска.

Вопрос на всю книгу. Когда два логических адреса находятся рядом, знаем ли мы, как далеко друг от друга их реальные магнитные участки? Нет: без внутренней информации контроллера мы можем судить об этом только косвенно и с оговорками.

Глава 1. Железо, которое умеет помнить

Жёсткий диск легко недооценить: снаружи прямоугольный металлический корпус, внутри — целая система точной механики, электроники и управления.

1.1. Что на самом деле лежит в металлической коробке

Представим, что мы положили на стол ноутбук, вынули из него старый 2,5-дюймовый HDD и посмотрели на корпус. Он почти ничего не рассказывает о своих внутренних возможностях. На этикетке есть производитель, модель, объём и служебная информация. Для большинства владельцев этого достаточно: подключили диск — и в проводнике появился новый том.

На самом деле перед нами компактная электромеханическая система. Её работу обеспечивают несколько связанных узлов. Магнитная пластина служит поверхностью для хранения информации. Шпиндельный двигатель вращает её с точно контролируемой скоростью. Подвижный узел головок располагает считывающий элемент над нужной областью. Контроллер выполняет команды, следит за сервосистемой, обработкой сигналов и исправлением ошибок. Буфер сглаживает часть различий между скоростью механики и работой интерфейса.

Важнейшая особенность HDD состоит в разделении двух движений. Пластина вращается, а головка перемещается по радиусу, точнее — по дуге, потому что рычаг актуатора закреплён на поворотной оси. Они не двигаются как единое целое. Вращение подводит под головку нужный сектор, а поворот рычага выбирает дорожку. Чтобы получить данные, контроллер должен согласовать обе задачи.

По документации для выбранного нами HGST HTS725050B7E630 используется одна магнитная пластина. Это не универсальный закон для всех 2,5-дюймовых дисков: модели большей вместимости или другой конструкции могут иметь несколько пластин и набор головок. Но принцип управления остаётся знакомым.

Нарисованные далее схемы условны. Они помогают различать функции деталей, а не изображают устройство нашего конкретного экземпляра с инженерной точностью. Более того, ради фотографии не нужно вскрывать рабочий диск: внутреннее пространство HDD должно оставаться чистым, а неподготовленное вскрытие может лишить накопитель работоспособности.

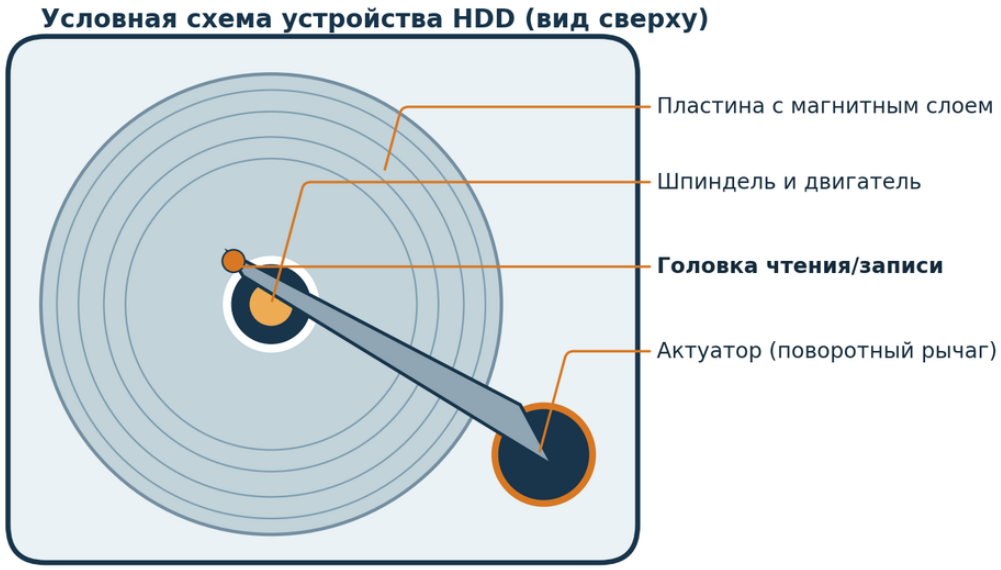


Рисунок 1.1. Основные узлы HDD. Пластина вращается; рычаг актуатора перемещает головку между дорожками. Схема условная.

1.2. Пластина — не просто металлический круг

Когда говорят «магнитный диск», легко представить обычную железную пластинку. В действительности пластина — высокоточная деталь с очень гладкой поверхностью и многослойным покрытием. Основа обеспечивает геометрию и жёсткость. Магнитный слой предназначен для записи информации. Защитные покрытия снижают риск повреждения. Толщины многих функциональных слоёв ничтожны по сравнению с размерами самого диска.

Принцип записи основан на свойствах магнитного материала. Вместо того чтобы механически процарапывать в нём цифры, записывающий элемент создаёт магнитные состояния и переходы, которые затем распознаются электроникой. На бытовой схеме их удобно рисовать цветными полосками. Но это условность: на поверхности нет видимых глазом оранжевых и белых прямоугольников, соответствующих каждому файлу.

В современных HDD используется перпендикулярная магнитная запись — один из способов ориентировать намагниченность внутри слоя. Это позволило повысить плотность хранения информации. Здесь достаточно понять главное: биты представлены физическим состоянием очень маленьких областей, а прочитать их можно лишь при точном согласовании головки, движения пластины и системы обработки сигнала.

Читателю может показаться, что чем меньше область записи, тем проще нарастить ёмкость. Но у плотности есть цена. При уменьшении магнитных областей возрастают требования к стабильности материала, точности позиционирования, качеству сигнала и алгоритмам исправления ошибок. Поэтому развитие HDD — это не просто создание двигателя, который вращается быстрее. Это совместная работа материаловедения, механики, электроники и программ управления.

Почему в разных областях диска скорость бывает разной? Вращение задаёт одинаковую угловую скорость, но линейный путь на внешней части пластины длиннее, чем возле центра. У многих HDD в наружных зонах помещается больше секторов на оборот, что влияет на последовательную скорость. Однако ни один пользовательский номер LBA сам по себе не доказывает, какой именно радиус или дорожка ему соответствует. В опытах мы будем различать эти два утверждения.

1.3. Шпиндельный двигатель — хранитель ритма

Шпиндельный двигатель приводит пластину в постоянное вращение. Для нашего лабораторного HGST паспортная скорость составляет 7200 оборотов в минуту — это 120 оборотов в секунду. Один полный оборот при такой скорости занимает примерно 8,33 миллисекунды. Уже это число помогает почувствовать масштабы времени, с которыми имеет дело накопитель.

Если нужный сектор прошёл мимо головки только что, при определённых условиях приходится ждать следующего появления нужного участка. В среднем при равномерно распределённом угловом положении ожидание составляет около половины оборота, то есть примерно 4,17 миллисекунды. Это только учебная оценка вращательного ожидания. Реальная команда может включать позиционирование, очереди, кэш, служебные действия и время передачи, а контроллер способен планировать запросы сложнее, чем наша простая схема.

Поэтому частота вращения важна, но не определяет всю производительность. Диск с одинаковыми 7200 оборотами в минуту может показывать разные скорости чтения из-за расположения данных, количества секторов в зоне, состояния поверхности, прошивки и интерфейса подключения.

Шпиндель также нужен для правильной работы узла головок: в штатном режиме между движущейся поверхностью и слайдером считывающей головки возникает тонкий газодинамический слой. Не следует представлять, будто головка постоянно трётся о пластину как игла проигрывателя. Когда диск останавливается, конструкция предусматривает безопасное положение головок согласно устройству конкретной модели.

Двигатель требует питания. Запуск и режимы простоя отличаются от устойчивой работы на скорости. Но когда в эксперименте мы видим задержку 350 миллисекунд, нельзя автоматически объявлять её временем раскрутки. Мы не измеряем непосредственно обороты шпинделя и не знаем, что происходило в USB-мосте, ОС или контроллере перед командой. Для вывода о причине нужны отдельные проверки.

Простая физика. 7200 об/мин = 120 оборотов в секунду. Один оборот ≈ 8,33 мс, половина оборота ≈ 4,17 мс. Эти числа помогают понять масштаб вращательной задержки, но не заменяют измерение полной операции чтения.

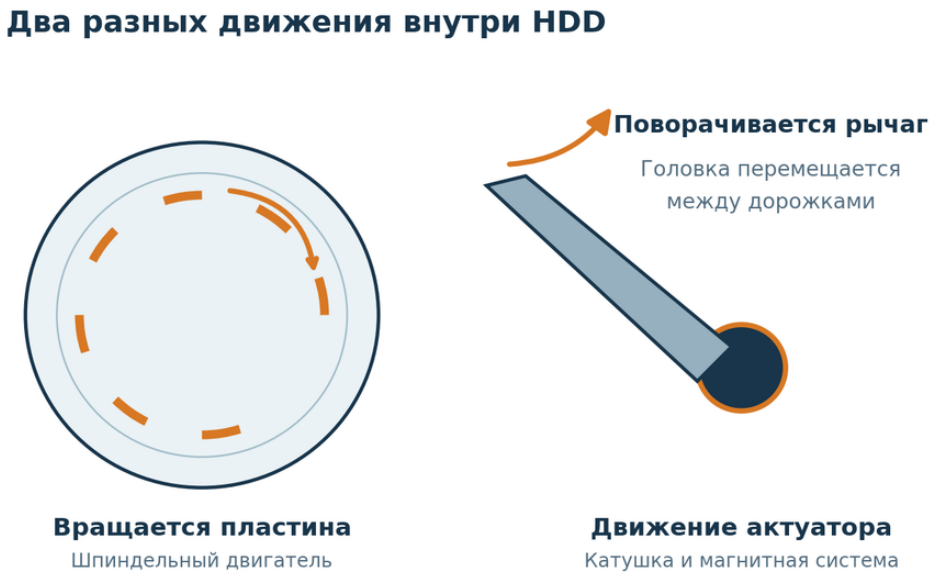


Рисунок 1.2. Вращение пластины и движение актуатора — два разных процесса. Именно их сочетание позволяет головке обратиться к нужной области.

1.4. Актуатор: почему головка ездит не сама по себе

Если бы головка была закреплена неподвижно, она могла бы работать только с узкой кольцевой областью пластины. Но информацию нужно размещать на множестве дорожек. Поэтому в HDD есть актуатор — система, перемещающая узел головок. В распространённой конструкции используется поворотный рычаг с катушкой, работающей в магнитном поле. Отсюда название voice coil motor, или VCM. По принципу управления это вовсе не обычный шаговый двигатель с фиксированными крупными шагами.

Когда через катушку проходит ток, возникает сила, поворачивающая узел. Контроллер изменяет управляющее воздействие, а сервосистема следит, куда именно пришла головка. Размах перемещения невелик, но требования к точности огромны. Главная задача не в том, чтобы как можно быстрее размахнуть рычагом, а в том, чтобы удержать считывающий элемент в нужном положении на конкретной дорожке.

Здесь возникает важное отличие от привычного механического движения. Если передвинуть карандаш по линейке с отметки 3 на отметку 7, мы видим расстояние в сантиметрах. Когда программа просит HDD прочитать два логических блока, она не видит расстояния в миллиметрах. Номера LBA назначены в логическом пространстве, а реальная физическая адресация скрыта за контроллером.

В лаборатории мы сравнивали близкие и далёкие логические адреса. Наблюдаемая разница во времени действительно бывает. Но эти опыты сами по себе не дают чистого времени движения актуатора: к задержке добавляются вращение, работа интерфейса, состояния кэша и служебная логика. Именно поэтому в этой книге выражение «цена перемещения» будет употребляться осторожно. Мы исследуем цену перехода между обращениями, а не снимаем скоростную характеристику каретки датчиком положения.

1.5. Головка работает почти без прикосновения

У большинства читателей первой ассоциацией с диском будет проигрыватель виниловых пластинок. Там игла физически касается дорожки. Но магнитный HDD устроен иначе. Считывающий и записывающий элементы находятся в небольшом узле, который работает над поверхностью вращающейся пластины. Между ними существует крайне малый рабочий зазор. Конкретная высота и устройство такого узла зависят от конструкции, режима и поколения накопителя.

Точность необходима по двум причинам. Во-первых, магнитные участки очень малы, поэтому головка должна находиться достаточно близко, чтобы получить полезный сигнал. Во-вторых, даже небольшой контакт с поверхностью на рабочей скорости способен привести к повреждениям. Пылинка, которая человеку кажется незаметной, для такого зазора может оказаться огромным препятствием.

Это одна из причин, почему не стоит вскрывать HDD на кухонном столе ради фотографии. Внутреннее пространство накопителя не предназначено для попадания бытовой пыли. В некоторых моделях используется воздух с фильтрацией и выравниванием давления, в других — герметизированная среда; конкретная конструкция различается. Поэтому отсутствие видимых повреждений после вскрытия ещё ничего не гарантирует.

При чтении головка не «видит» готовые буквы или числа. Она воспринимает изменения магнитного сигнала. Электроника усиливает и обрабатывает сигнал, преобразует его в последовательность битов и проверяет достоверность с помощью корректирующих кодов. Всё это происходит до того, как программа получит привычные байты фотографии.

Для владельца домашнего архива практический вывод прост: ценность HDD не только в моторе и металле. Его способность читать данные зависит от совместной точной работы физической головки, магнитного слоя, электроники и управляющего кода. Вся эта система должна быть исправной, а резервная копия важнее надежды на вечную механику.

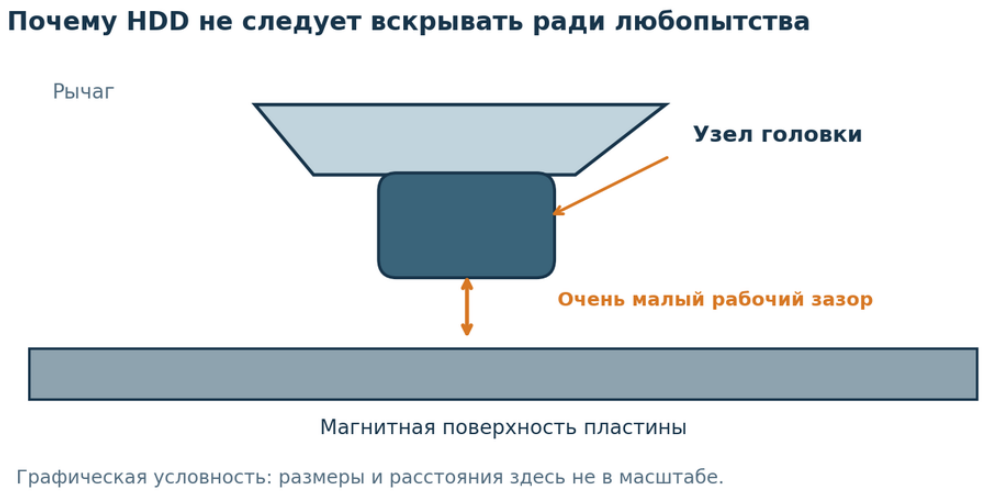


Рисунок 1.3. Головка в рабочем режиме не должна постоянно тереться о пластину. Схематическое изображение зазора, размеры не в масштабе.

1.6. Сервосистема: невидимые дорожные знаки

Можно ли по одной команде электрического тока гарантировать, что головка попала на нужную дорожку? Нет. Сила, положение, колебания и температура изменяются, а сам диск работает с точностями, недоступными для грубого механического управления. Поэтому HDD использует сервосистему — механизм обратной связи, который помогает определять и удерживать положение головки.

На магнитной поверхности записана не только полезная информация пользователя. Существуют специальные серводанные, необходимые для позиционирования. Головка считывает соответствующие сигналы, а контроллер оценивает отклонение и корректирует положение актуатора. Это похоже на водителя, который не просто повернул руль на заданный угол, а смотрит на разметку и постоянно удерживает машину в нужной полосе. Сравнение приблизительное, но смысл обратной связи передаёт верно.

Сервосистема позволяет не только находить дорожку, но и удерживать её во время вращения. Ведь поверхность и механика испытывают микроскопические колебания, а температура и вибрации влияют на взаимное положение деталей. Подобные явления объясняют, почему современный HDD нельзя считать простой механической коробкой с таймером и мотором.

Важный вывод для нашей темы: даже если бы программа точно знала порядковый номер нужной дорожки, это не означало бы возможности напрямую управлять рычагом через обычный запрос чтения. Контроллер остаётся посредником и самостоятельно решает низкоуровневые задачи позиционирования и безопасности.

Именно здесь проходит одна из границ «Гантовой зебры». Мы способны планировать последовательность логических обращений и измерять итоговое время. Но физическую траекторию микродвижений головки, её коррекции по сервосигналу и внутренние решения прошивки мы в обычных опытах не наблюдаем.

1.7. Контроллер: почему диск — это ещё и компьютер

Внешне HDD напоминает механическое устройство, но без платы управления он не сумел бы прочитать ни одного файла. Электроника принимает команды, управляет двигателем и актуатором, обрабатывает сигналы, проверяет ошибки, обслуживает собственный буфер и передаёт результат наружу. В современной конструкции контроллер выполняет сложные программные алгоритмы, а часть служебной информации накопитель может хранить внутри своей системы записи.

Когда Windows обращается к тому, она не управляет катушкой актуатора через отдельный провод. Приложение запрашивает данные; файловая система и драйверы превращают запрос в обращения к логическим блокам; контроллер устройства выполняет низкоуровневую работу. Для компьютера HDD выглядит как адресуемый блоковый накопитель. Это существенно удобнее, чем если бы каждой программе приходилось знать устройство магнитных дорожек.

Но удобство оборачивается потерей наблюдаемости. Если два LBA расположены в числовом порядке рядом, это не доказательство их физического соседства на одном радиусе, особенно с учётом скрытой трансляции, переназначения дефектных секторов и особенностей внутренней организации. Поэтому нельзя просто нарисовать на бумаге шкалу LBA и объявить её схемой намотанных дорожек.

Ещё одна роль контроллера — работа с ошибками. Магнитный сигнал не всегда идеален. Часть повреждений и слабых чтений может исправляться автоматически; повторные внутренние операции способны увеличить задержку без явного сообщения приложению. Мы не должны автоматически приписывать любой долгий ответ именно этому механизму, но обязаны помнить, что он существует.

С помощью внешней программы можно измерить, за какое время завершился запрос, сколько байтов получено, какой логический адрес использован и в каком порядке выполнялись обращения. Для этого не нужен собственный драйвер. А вот для чтения скрытой заводской геометрии драйвера тоже может оказаться недостаточно: нужной информации не предоставляет сам стандартный интерфейс накопителя.

Итак, контроллер превращает сложную машину в понятный адресный интерфейс, но одновременно скрывает большинство деталей механической работы. Вторая часть книги как раз посвящена границе между этими двумя слоями.



Рисунок 1.4. Файловая система, LBA и контроллер образуют несколько уровней между приложением и магнитной поверхностью. Схема не раскрывает внутреннюю трансляцию адресов.

1.8. Что происходит, когда мы открываем фотографию

Допустим, на старом HDD лежит семейная фотография. В проводнике Windows мы дважды щёлкаем её значок. С этого момента запускается цепочка действий, хотя на экране всё выглядит как единая операция. Программа просмотра обращается к файлу. Файловая система ищет метаданные и определяет, какие участки тома содержат его байты. Операционная система отправляет накопителю запросы чтения. Контроллер диска обеспечивает получение соответствующих блоков и передаёт данные через интерфейс.

Если нужные данные уже оказались в оперативной памяти или других кэшах, часть физической работы может быть пропущена. Поэтому повторное открытие фотографии иногда быстрее первого. Это не чудо и не доказательство ускорившейся механики: система могла избежать нового чтения с пластины.

Если же требуется физическое обращение, контроллеру может понадобиться найти нужную дорожку, дождаться прохождения сектора и считать сигнал. Затем данные поступят через SATA и USB-мост в компьютер. После этого приложение декодирует формат изображения и покажет картинку. Время, которое пользователь ощущает как «открылась фотография», включает не только работу HDD, но и обработку файла, память, интерфейс и графическое приложение.

В нашей лаборатории мы старались уменьшить число лишних факторов: часть опытов обращалась к физическому диску напрямую, а не открывала JPEG через просмотрщик. Однако даже такой способ не устраняет контроллер, USB-мост, Windows и возможные скрытые буферы. «Прямое чтение» здесь означает обход некоторых уровней обработки файла, а не прямое управление магнитной головкой.

Почему это важно? Потому что хорошие измерения требуют правильно выбранного вопроса. Если хотим узнать, сколько времени человек ждёт фотографии, достаточно измерить работу приложения. Если хотим сравнить последовательность команд к диску, лучше использовать специальные небольшие чтения и регистрировать их время. А чтобы узнать реальную траекторию движения головки, потребуется другая аппаратура и другой уровень доступа. Это три разных исследования.

1.9. Время чтения — не только движение каретки

Попробуем представить процесс запроса как несколько частей: формирование команды, позиционирование головки, возможное ожидание углового положения сектора, передача данных и дополнительные задержки системы. Такой разбор полезен для понимания, но не следует воспринимать его как точный состав любой отдельно измеренной операции. Некоторые процессы частично перекрываются, а в отдельных ситуациях физическое чтение вообще не требуется благодаря кэшу.

Отсюда возникает первая инженерная ловушка. Пусть операция заняла 20 миллисекунд. Нельзя вычесть из этого числа паспортную половину оборота и объявить остаток временем движения актуатора. На такой расчёт влияют очереди, геометрия, размер запроса, USB и внутренние решения контроллера. Он дал бы красивую формулу, но не обязательно правдивую.

Есть и вторая ловушка. Если два запроса выполняются быстро друг за другом, легко решить, что головка почти не двигалась. Однако данные могли быть уже подготовлены во внутреннем буфере или в кэше системы. Поэтому мы проводили специальные опыты с повторениями, перемешиванием и неповторяющимися адресами. Их смысл — уменьшить число возможных объяснений и не назвать эффект кэша победой над механикой.

Третья ловушка связана с интерфейсом. Один и тот же диск может использоваться через разные USB–SATA-переходники. Скорость внешнего канала и особенности моста влияют на итоговую картину. Если сравнить результаты, не указав способ подключения, можно ошибочно приписать различие поверхности магнитной пластины.

Именно поэтому у каждого эксперимента в этой книге будет паспорт: какой диск, какой интерфейс, какой объём, какие адреса, сколько повторений и что мы действительно измеряли. А рядом с выводом будет честное ограничение: чего это измерение не доказывает.



Рисунок 1.5. Виды затрат времени на чтение. Длины участков условны: диаграмма не является результатом конкретного эксперимента.

1.10. Наш конкретный HDD — и почему паспорт не равен измерению

Для второй части книги мы выбрали конкретный стенд, а не абстрактный диск «средней скорости». В лабораторных журналах указан HGST Travelstar Z7K500.B, модель HTS725050B7E630, ёмкостью 500 ГБ. По технической документации он относится к 2,5-дюймовым HDD толщиной около 7 мм, использует SATA 6 Гбит/с, вращение 7200 об/мин и буфер 32 МБ. Магнитная запись — перпендикулярная. Сектора организованы по Advanced Format: 4096 байт физически и 512 байт логически для совместимости.

Важно не перепутать обозначения. «SATA 6 Гбит/с» сообщает о максимальном классе интерфейса, а не о фактической скорости чтения с поверхности. «500 ГБ» — десятичное маркетинговое обозначение ёмкости, не равное числу гибибайт, которое отображает часть программ. «32 МБ буфера» — характеристика внутреннего устройства, но не гарантия, что любой повторный запрос обязательно будет обслуживаться из него.

В наших экспериментах этот HDD находился за USB–SATA-мостом. Первоначально применялся переходник JMicron с более медленным режимом USB, затем — ORICO с контроллером VIA Labs и высокоскоростным подключением. Логический том имел файловую систему NTFS и букву T: в Windows. При этом измерительные программы сохраняли результаты на другом накопителе, чтобы не создавать новые файлы на самом исследуемом HDD.

Мы видим здесь сразу два мира: паспорт изготовителя и наш реальный измерительный тракт. Паспорт показывает заявленные свойства модели в определённых условиях. Эксперимент показывает, что произошло на данном экземпляре, с данным мостом и конкретной системой Windows. Одна таблица не должна заменять другую.

Например, в паспорте HDD можно найти типичное среднее время позиционирования головки, но наши измерения задержки raw-чтения через USB не равны этому числу и не должны сравниваться с ним как две оценки одной величины. У них разный физический смысл.

Параметр	Лабораторный HDD
Модель	HGST HTS725050B7E630
Серия	Travelstar Z7K500.B
Тип / размер	Магнитный HDD / 2,5 дюйма
Ёмкость	500 ГБ
Скорость вращения	7200 об/мин
Интерфейс диска	SATA 6 Гбит/с
Секторная организация	4К физически / 512 Б логически
Подключение в опытах	USB–SATA, мосты JMicron и ORICO
Файловая система стенда	NTFS, том T:

Таблица 1.1. Параметры лабораторного HDD

Паспортные свойства модели взяты из технической документации HGST/Western Digital; параметры подключения и тома — из лабораторных журналов проекта.

1.11. Что действительно полезно владельцу домашнего архива

Если бы целью нашей книги было только выяснить, какой HDD выгоднее купить, хватило бы нескольких сравнительных таблиц. Но мы хотим понимать устройство. Тем не менее уже первая глава даёт практические выводы, которые пригодятся человеку с домашним фотоархивом.

Первый вывод — не стоит оценивать безопасность хранения по скорости вращения или по наличию кэша. Быстрый накопитель может выйти из строя так же неожиданно, как медленный. Ценные файлы должны иметь независимые резервные копии. Долговременное хранение включает не только запись, но и периодическую проверку чтения, контроль целостности и обновление носителей по мере необходимости.

Второй вывод — HDD не нужно без необходимости подвергать ударам и вибрациям, особенно во время работы. Внутри действует точная механика с малым рабочим зазором. Внешне лёгкое падение может иметь серьёзные последствия. Не нужно вскрывать гермоблок ради красивой фотографии и не нужно ставить работающий диск туда, где его легко задеть.

Третий вывод — при выборе внешнего корпуса полезно учитывать не только модель HDD, но и интерфейсный мост, питание, охлаждение и качество соединения. Если накопитель ведёт себя неожиданно, проблема может находиться в кабеле, USB-порту или адаптере, а не на магнитной поверхности. Это не повод самостоятельно разбирать диск: сначала важнее убедиться, что данные сохранены и есть копия.

Четвёртый вывод — не всякая медленная операция означает физический дефект. Кэш, фоновые запросы операционной системы, разрозненность данных и особенности интерфейса способны сильно менять время работы. Но и игнорировать устойчивые ошибки чтения нельзя: при признаках неисправности приоритетом становится сохранение данных, а не серия дополнительных нагрузочных тестов.

Наконец, пятый вывод — выбирать хранилище следует под задачу. Для системы и интенсивной работы удобнее SSD. Для объёмного домашнего архива HDD может оказаться разумным компонентом системы хранения. Для действительно ценных фотографий лучше использовать несколько копий и не зависеть от одного типа носителя. Эта книга не противопоставляет технологии как соперников: она исследует особенности одной из них.

1.12. Что мы узнали и к чему перейдём дальше

Теперь можно иначе посмотреть на обычный процесс открытия файла. Данные живут в магнитном слое вращающейся пластины. Двигатель поддерживает движение. Головка считывает очень слабые сигналы, не касаясь поверхности в нормальном режиме. Актуатор переводит её между дорожками, сервосистема удерживает нужное положение, контроллер скрывает низкоуровневую сложность, а операционная система превращает результат в понятные файлы.

Ни один из этих узлов не существует отдельно от остальных. Именно поэтому время выполнения команды чтения нельзя объяснять единственной причиной, не проверив остальные. Задержка может отражать движение головки, ожидание сектора, работу кэша, интерфейса, контроллера или несколько факторов сразу.

Отдельно мы установили важное различие между физическим и логическим описанием диска. Магнитные области существуют на пластине. LBA существует как адресный интерфейс. NTFS организует файлы на томе. Эти уровни связаны, но не тождественны. Чередование логических номеров не означает того же чередования реальных дорожек.

В следующей главе мы внимательно разберём эту лестницу адресации: как магнитная запись становится сектором, сектор — логическим блоком, а блоки — кластерами и файлами. Только после этого имеет смысл переходить к измерению близких и дальних адресов. Иначе мы рискуем принять аккуратную таблицу чисел за карту физического устройства, которой у нас нет.

Глава 2. Где на самом деле находятся данные?

Между магнитным следом на пластине и фотографией в папке Windows лежит несколько разных карт. Самое сложное — не спутать одну с другой.

2.1. Фотография в папке и фотография на пластине — не одно и то же

Откроем мысленно привычную папку «Семейные фотографии». В ней лежат снимки за разные годы: школьные праздники, поездки, домашние животные, старые фотографии с первого цифрового фотоаппарата. Каждый файл имеет понятное имя, дату, размер и расширение. На экране кажется, что фотография представляет собой единый предмет: значок, который можно переместить, скопировать или удалить.

Но магнитному диску ничего не известно о нашей семейной истории. Его электронная часть не различает отпуск и выпускной. Она принимает команды чтения и записи блоков с числовыми адресами. Внутри сохраняются последовательности кодированных магнитных состояний. Разобраться в имени фотографии и её содержимом должна файловая система вместе с программами, которые умеют читать формат изображения.

Это первый важный шаг к пониманию хранения: «файл» — удобное представление для человека и операционной системы, а «логический сектор» — часть языка общения с накопителем. Между ними находится система учёта, похожая на каталог большой библиотеки. Библиотекарь знает название книги, в каком шкафу искать том и на каких полках лежат отдельные его части. Но сами страницы не обязаны лежать там, где их мысленно разместил читатель.

Здесь стоит уточнить одну деталь. Даже аналогия с библиотекой не полностью описывает HDD. Библиотекарь может показать пальцем на конкретный шкаф. Обычная программа Windows такого права не имеет: контроллер диска показывает адресный номер, а реальное преобразование в положение на поверхности остаётся внутри накопителя. Поэтому наша карта будет состоять из нескольких этажей, а не из одной линии от имени JPG до точки на пластине.

Зачем вообще разбирать эту цепочку? Потому что вся будущая работа с «Гантовой зеброй» связана со словом «рядом». Два файла могут лежать рядом в списке папки, их логические адреса могут идти подряд, а магнитная организация может быть устроена сложнее. Пока мы не определили, на каком этаже рассматриваем соседство, сравнивать расстояния преждевременно.

Главный вопрос главы. Если компьютер запросил два последовательных номера LBA, можно ли утверждать, что обе порции данных расположены на одной дорожке и головка почти не двигалась? Нет. Логическая последовательность не раскрывает внутреннюю адресную карту HDD.

2.2. Что означает «намагниченная область»

Мы уже видели в первой главе, что на пластине находится магнитный слой и над ним работает головка. Теперь увеличим мысленно участок поверхности настолько, чтобы вместо круга перед нами появилась узкая полоса записи. На популярной картинке её часто изображают как ряд прямоугольников, одни из которых направлены «вверх», другие «вниз». У рисунка есть достоинство: можно увидеть, что информация хранится не как краска, а как состояние магнитного материала.

Однако такая картинка опасна, если принять её буквально. Один прямоугольник на схеме не обязательно соответствует одному биту файла. Современная запись использует кодирование, синхронизацию, избыточные данные для исправления ошибок и обработку сигнала. Головка воспринимает не подписанные цифры, а физическое изменение поля и соответствующие электрические сигналы. Электроника восстанавливает из них записанную последовательность.

Наш исследуемый HGST относится к семейству дисков с перпендикулярной магнитной записью — PMR. В упрощённом объяснении это означает, что направление намагниченности в записывающем слое имеет существенную составляющую перпендикулярно поверхности. Благодаря особенностям магнитных материалов и конструкции головки такой подход позволил уплотнять запись. Но читателю не нужно представлять магнитные «стрелочки», торчащие из пластины: реальные магнитные структуры намного мельче и сложнее модели на бумаге.

Очень важно не смешивать здесь два вида полосатости. Первая — физическая: чередование состояний магнитного слоя вдоль дорожки. Вторая — наша авторская: «Гантова зебра» как способ мысленно выделять чередующиеся группы обращений или данных на логическом уровне. Между ними нет заранее установленного соответствия. Полосы в книге — учебная метафора, а не снимок поверхности конкретного HDD.

Из этого следует и практическая оговорка. Когда компьютер получает блок данных, он не просит головку показать список магнитных доменов. Внутри накопителя сигнал уже прочитан, обработан и проверен. За пределами корпуса появляется обычная последовательность байтов. Именно поэтому мы можем открыть изображение десятилетней давности, ничего не зная о коде магнитных переходов, из которых была восстановлена информация.

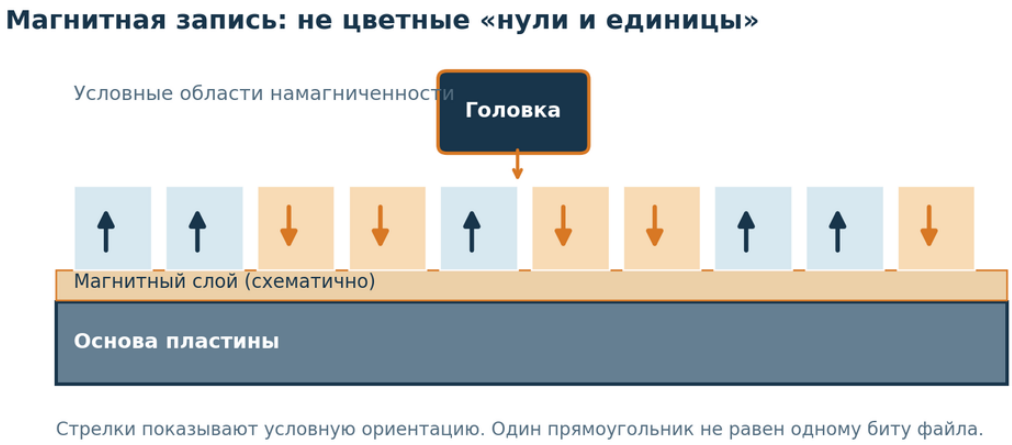


Рисунок 2.1. Условное изображение магнитного слоя и головки. Стрелки показывают принцип намагниченности; размеры и количество областей не соответствуют реальной плотности записи.

2.3. Дорожка, сектор и зона — что между ними общего?

Вернёмся от увеличенного магнитного участка к целой пластине. Если смотреть на диск сверху, поверхность удобно разделить на множество концентрических дорожек. Каждая дорожка напоминает узкую круговую дорогу. Головка выбирает дорожку с помощью поворотного механизма, а вращение пластины подводит под неё нужную часть окружности.

На этой дороге должна существовать возможность отделять одну порцию данных от другой. Для этого используют понятие сектора — адресуемого блока данных на уровне накопителя. Сектор не следует представлять как широкую видимую дольку пирога. Такая долька на школьной схеме лишь помогает увидеть идею разбиения. Реальные сектора располагаются последовательно вдоль дорожек, вместе со служебными участками и структурами синхронизации.

А что такое зона? В механическом диске длина внешней дорожки больше длины внутренней. Если записывать информацию с сопоставимой линейной плотностью, на более длинной окружности потенциально помещается больше данных. Производители используют зонированную организацию: группы дорожек могут отличаться числом секторов на дорожку. Это один из физических факторов, из-за которых последовательная скорость передачи на различных участках диска различается.

Сразу возникает соблазн перевести это утверждение в карту LBA: маленькие номера якобы обязательно означают внешний радиус, а большие — внутренний. В некоторых классических упрощённых моделях непрерывные диапазоны адресов действительно ассоциируются с зонами поверхности. Но современный диск скрывает распределение данных, резервные участки и внутренние действия контроллера. Поэтому по одному номеру LBA нельзя с инженерной строгостью указать точное положение конкретного сектора.

Модель дорожек полезна не для предсказания координат каждого файла, а для понимания того, почему скорость может различаться. Она объясняет, почему наш будущий эксперимент с разными зонами адресного пространства имеет смысл: мы задаём контроллеру запросы к логическим областям и смотрим, как меняется наблюдаемая скорость. Но эксперимент не превращает логический процент заполнения адресного диапазона в датчик положения головки.

Есть и ещё одно отличие: диск может иметь несколько поверхностей и головок. Наш конкретный 2,5-дюймовый HGST по своему семейству относится к однопластинчатым решениям, но учебная модель не должна представлять это обязательным свойством всех HDD. При смене модели число поверхностей, геометрия и организация зон меняются, а интерфейс LBA для программ остаётся столь же удобным.

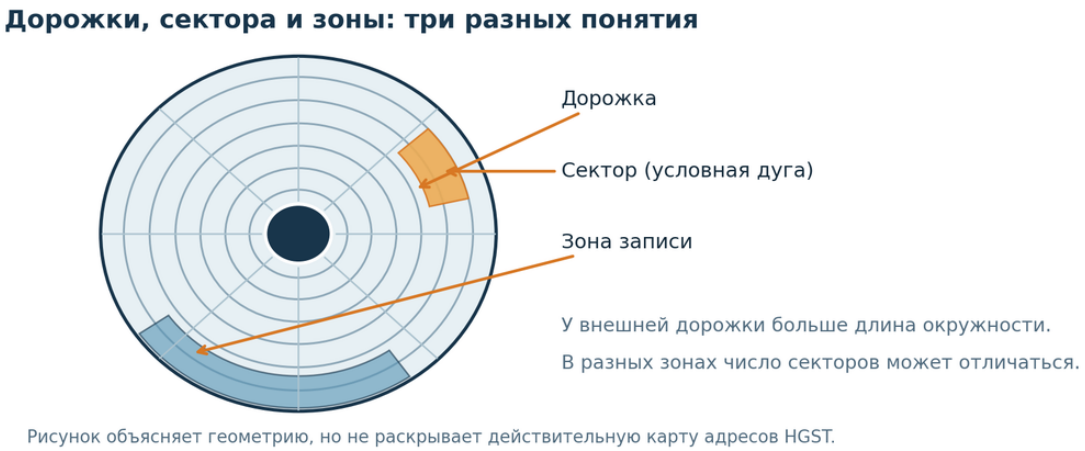


Рисунок 2.2. Принцип дорожек, секторов и зон. На рисунке намеренно не указано, какой LBA соответствует нарисованной дуге: такая информация из наших опытов не следует.

2.4. Что такое LBA, если это не координата?

Аббревиатура LBA расшифровывается как Logical Block Addressing — логическая адресация блоков. В бытовом переводе это способ сказать диску: «Дай мне блок с таким-то порядковым номером». Число служит адресом в пространстве, которое накопитель предоставляет компьютеру. Программе не нужно называть цилиндр, дорожку, головку или сектор в старом геометрическом смысле.

Для представления можно вообразить длинную пронумерованную ленту из небольших ячеек: 0, 1, 2, 3 и так далее. Когда программа запрашивает диапазон, она задаёт стартовый номер и количество блоков. Лента в данном случае не физическая: её номера могут быть преобразованы контроллером во внутренние места хранения и операции с механикой. Это удобная договорённость между компьютером и диском.

Обратите внимание: слово «блок» здесь тоже нуждается в единицах измерения. Накопитель может предоставлять логические блоки по 512 или по 4096 байт в зависимости от модели и режима. Если известно, что логический сектор равен 512 байтам, смещение в байтах для LBA номер N в самом простом случае соответствует произведению N на 512. Но смещение файла внутри раздела — уже другая координатная система, с собственным началом отсчёта.

Почему это удобнее прежней геометрической модели? Потому что производитель волен менять внутреннюю организацию, не заставляя тысячи программ знать устройство нового механизма. Контроллер может управлять дефектными участками, обслуживать служебные зоны и применять свои алгоритмы. Для операционной системы интерфейс остаётся последовательностью адресуемых блоков.

Цена этого удобства — потеря прозрачности. Когда мы видим, что чтение близких номеров LBA выполняется быстрее, можем предположить, что на данном экземпляре контроллер обслуживает их эффективно при данном порядке запросов. Но нельзя автоматически вывести из этого точный путь головки. Точно так же высокая скорость соседних адресов не доказывает, что данные соседствуют на магнитной поверхности именно так, как на нашем рисунке.

Именно этот компромисс между удобным логическим интерфейсом и скрытой механикой лежит в центре второй части книги. Мы не стремимся отменить LBA. Наоборот, хотим понимать, какие вопросы ему допустимо задавать и какие ответы потребуют другой аппаратуры.

Небольшая проверка смысла. «LBA 1000 и LBA 1001 идут подряд» — корректное логическое утверждение. «Значит, головка прочитала их на одной физической дорожке без переходов» — уже гипотеза, которую сами номера не подтверждают.

2.5. Почему 512 байт и 4096 байт могут означать один сектор

Теперь появилась возможность спокойно разобрать необычную характеристику нашего HDD: физический сектор 4096 байт, логический — 512 байт. На первый взгляд кажется, что производитель противоречит сам себе. Но речь идёт о двух разных уровнях интерфейса.

Технология Advanced Format использует более крупные физические сектора, что даёт преимущества по части служебных накладных расходов и организации защиты данных. Для совместимости со старыми системами многие накопители предоставляют компьютеру логические блоки по 512 байт, хотя физически запись организована по 4096 байт. Такой режим называется 512e — 512-byte emulation, эмуляция 512-байтовых логических секторов.

В нашем случае один физический сектор соответствует восьми последовательным логическим секторам по 512 байт. С точки зрения компьютера доступны восемь адресных единиц, а с точки зрения физической организации — одна более крупная единица. Поэтому особое значение приобретают границы выравнивания. Запись, которая затрагивает лишь часть физического сектора, в определённых обстоятельствах может потребовать предварительного чтения и последующей перезаписи более крупного блока. Здесь мы только объясняем возможный принцип; реальные эксперименты записи на исследуемом томе не проводились.

Для чтения также необходимо разделять величины. Запрос 4096 байт в наших опытах — не обязательно команда «прочитать один физический сектор» на уровне электроники. Он проходит через Windows, USB–SATA-мост и внутреннюю реализацию HDD. В идеальной модели 4096 байт совпадают с объёмом одного физического сектора. Но временные измерения такого запроса включают много других затрат.

А ещё рядом существует слово «кластер» — единица выделения места файловой системой. Кластер тоже может быть равен 4096 байтам, но это не делает его физическим сектором. Размер кластера задаётся томом и должен проверяться отдельно; одинаковая цифра на двух этажах не означает одинаковую сущность.

Такая путаница особенно распространена, когда люди говорят: «Я проверил диск блоками по 4K». Нужно спросить: размер чего именно? Запроса приложения, логического сектора устройства, физического сектора на носителе, страницы памяти или кластера NTFS? Часто эти числа совпадают, но смысл каждого остаётся своим.

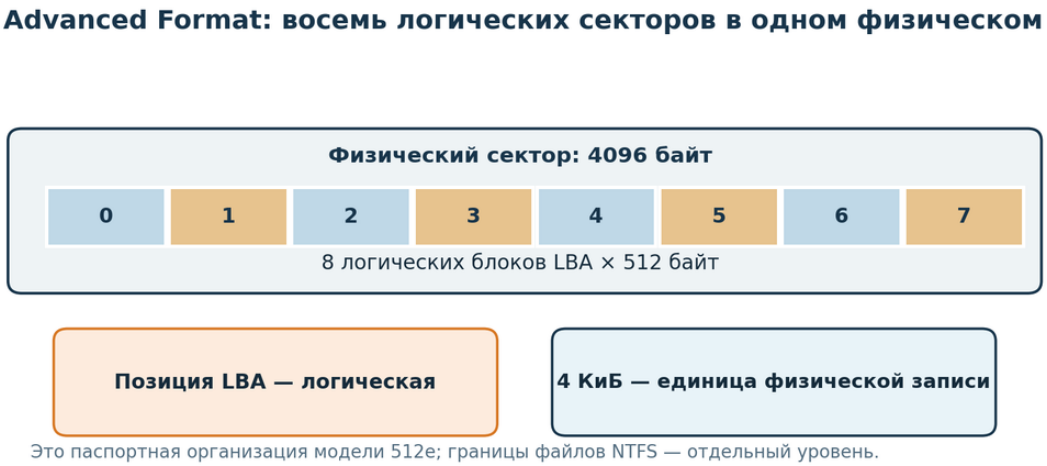


Рисунок 2.3. Принцип режима 512e на исследуемой модели: восемь логических блоков по 512 байт представлены поверх одного физического сектора размером 4096 байт.

2.6. От LBA к разделу, от раздела к NTFS

Когда новый накопитель подключают к Windows, пользователь часто видит букву диска и думает, что она обозначает сам физический прибор. Но буква T: в нашей лаборатории относится к логическому тому, а не к механике головок. У одного накопителя может быть несколько разделов, у нескольких устройств — более сложные схемы хранения, а некоторые тома вообще организованы не так, как привычная единственная область на одном HDD.

Чтобы увидеть разницу, представим диск как длинную линию нумерованных LBA. На этой линии может быть выделен раздел. Начало раздела имеет собственное смещение относительно начала устройства. Файловая система, например NTFS, работает уже с пространством своего тома и управляет тем, какие кластеры заняты файлами, а какие свободны. Номер кластера на томе не обязан совпадать с номером LBA устройства.

В простом случае, если раздел начинается с некоторого логического сектора и кластер равен восьми секторам по 512 байт, положение конкретного кластера можно выразить через смещение раздела и номер кластера. Но это будет вычисление логических адресов, а не координат на поверхности пластины. К тому же простая формула предполагает обычный том без дополнительных слоёв виртуализации и перераспределения.

Именно поэтому лабораторные обозначения нужно читать аккуратно. «HGST Disk 2» — номер физического устройства в Windows во время измерений. «T: NTFS» — буква и файловая система тома. «LBA-позиция 50 %» — участок диапазона логических адресов устройства, если программа построила его именно таким способом. Эти обозначения помогают воспроизвести опыт, но не дают координату дорожки на реальной поверхности.

Такой взгляд объясняет старую на первый взгляд странность. Обычный пользователь без труда может переименовать папку, не перемещая содержимое всех файлов по магнитным дорожкам. Потому что имя и место хранения учитываются разными структурами. Точно так же буква тома может измениться, тогда как сами данные не обязаны физически двигаться. Система адресации устроена именно для того, чтобы отделить удобное для человека от сложного для оборудования.

Для наших исследований это означает, что необходимо точно называть уровень каждой операции. Чтение raw-адреса физического накопителя, чтение файла по имени и запрос карты кластеров — разные действия. Ни одно нельзя без дополнительных проверок заменять другим. В реестре экспериментов они выделены отдельно не ради бюрократии, а ради точности результатов.



Рисунок 2.4. Шесть уровней, разделяющих имя фотографии и физическую запись. Стрелки обозначают логические преобразования, а не прямую линию к одной магнитной точке.

2.7. Что NTFS знает о файле, а чего не знает

Пора познакомиться с файловой системой поближе. NTFS — не просто способ прикрепить букву к накопителю. Она ведёт учёт файлов, каталогов, атрибутов, занятых областей и других сведений, которые делают том удобным для работы. Одной из центральных структур NTFS является Master File Table — MFT, главная файловая таблица. В ней хранятся записи о файлах и их свойствах.

Когда мы открываем фотографию, операционной системе нужно не только прочитать массив байтов. Сначала она находит запись о файле, определяет права доступа, узнаёт его размер и сведения о размещении данных. Если содержимое велико, оно обычно хранится вне самой записи MFT — в кластерах тома. Для некоторых очень небольших потоков данные могут размещаться непосредственно в атрибутах записи MFT. Поэтому не для всякого файла существует обычная карта внешних экстентов.

Здесь кроется важный практический вывод. Название файла и список кластеров — разные сведения. Если переименовать архив «Лето 2015» в «Семейные снимки», файловая система может изменить метаданные, не переписывая все мегабайты архива. И наоборот: при изменении или перемещении содержимого пользователь может сохранить прежнее имя, а распределение кластеров поменяется.

Файловая система может быть организована таким образом, что файл занимает несколько последовательных участков. Это не обязательно повреждение. Иногда свободное место выделяется частями, иногда файл постепенно увеличивается, иногда доступные диапазоны разделены другими файлами. Для владельца домашнего архива всё это может происходить незаметно. Главное — данные читаются, а система знает, откуда брать следующую порцию.

Именно на этом уровне рождается понятие фрагментации. Оно говорит, что не все кластеры потока файла составляют один сплошной непрерывный участок тома. Но фрагментация в терминах NTFS ещё не является прямой фотографией физического расположения магнитных участков. Она представляет разрывы в логическом пространстве тома. Наш HDD дополнительно преобразует LBA во внутренние действия.

Полезно помнить и о безопасности архива. Файловая система помогает находить содержимое, но не заменяет резервного копирования. Исправная таблица файлов не спасёт от полного отказа электроники накопителя, а сохранность магнитных областей не гарантирует, что повреждённые метаданные позволят без труда восстановить имена и структуру каталогов. Ценные фотографии и проекты должны иметь независимые копии.

2.8. Кластеры, VCN и LCN: два набора номеров

Предположим, у нас имеется файл из двенадцати кластеров. Внутри файла они логически идут по порядку: сначала первый, затем второй, третий и так далее. Для описания такой последовательности используется VCN — Virtual Cluster Number, виртуальный номер кластера в пространстве файла или его потока. Он отвечает на вопрос: «На каком месте внутри этого файла находится нужная порция?»

Но том NTFS состоит из множества кластеров, занятых разными файлами. У кластеров тома есть собственные номера — LCN, Logical Cluster Number. Они помогают ответить на другой вопрос: «Какой кластер на этом томе выделен под соответствующую часть файла?» Поэтому при чтении большого файла файловой системе необходимо сопоставить его VCN с LCN.

Для иллюстрации пусть VCN 0, 1, 2 и 3 соответствуют LCN 100, 101, 102, 103. Следующие VCN 4–7 окажутся на LCN 450–453, а последние VCN 8–11 — на LCN 900–903. Внутри самого файла нет дырки: он читается как одна последовательность из двенадцати кластеров. Но на томе они находятся в трёх несмежных логических участках.

Каждый непрерывный участок сопоставления называется экстентом. В нашем примере экстентов три. Если вместо двенадцати кластеров речь идёт о большом видеофайле, экстенты могут состоять из тысяч и миллионов кластеров. Иногда файл представлен одним крупным экстентом; иногда — многими. Точно узнать его карту можно только у файловой системы при наличии соответствующего доступа и применимости метода к данному потоку.

Именно в этом состоит смысл Windows-операции FSCTL_GET_RETRIEVAL_POINTERS, которую мы применяли при исследовании метаданных. Она позволяет запросить сопоставление VCN и LCN для файла. Но её результат не превращает LCN в цилиндр, номер головки или магнитный домен. Более того, разреженные и некоторые сжатые данные могут содержать логические области, для которых обычная физическая выделенная область отсутствует. Такие случаи требуют отдельной обработки.

На уроке эта разница часто кажется пустой терминологией: VCN, LCN, LBA — одни буквы. В исследовании она фундаментальна. VCN относится к позиции внутри файла. LCN — к позиции внутри тома. LBA — к блоку, адресуемому накопителем. И никакой из этих номеров сам по себе не сообщает точную магнитную координату. Мы будем повторять эту мысль не ради сложности, а чтобы потом правильно разобрать реальные результаты.

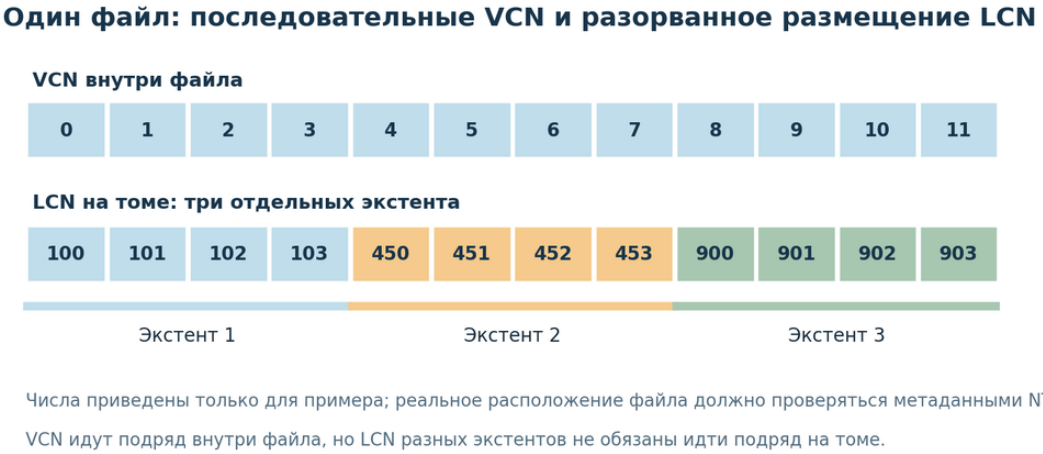


Рисунок 2.5. Пример файла из двенадцати кластеров: непрерывная внутренняя нумерация VCN сопоставлена с тремя раздельными экстентами LCN. Все числа на схеме придуманы для объяснения, это не данные наших экспериментов.

Термин	Где начинается отсчёт	Что позволяет узнать
VCN	Внутри файла или потока	Логическую позицию части содержимого
LCN	Внутри файлового тома	Логическое размещение кластера тома
LBA	В пространстве блоков устройства	Номер адресуемого логического блока
Дорожка	В физической геометрии HDD	Условную окружность записи на поверхности

Таблица 2.1. Похожие слова описывают разные уровни. Не следует переводить между ними номера без знания параметров раздела, тома и внутренней трансляции накопителя.

2.9. Пример на обычной фотографии: от байтов к адресам

Чтобы все предыдущие определения не остались абстрактными, возьмём небольшой условный пример. Пусть изображение занимает ровно 1 МиБ. Это не один миллион байтов, а 1 048 576 байт. Если файловая система выделяет кластеры по 4096 байт, под такое содержимое понадобится 256 кластеров, не считая иных особенностей метаданных. Если диск предоставляет логические блоки по 512 байт, то объём этих данных равен 2048 таким блокам.

Однако эта арифметика отвечает только на вопрос об объёме. Она не говорит, сколько раз головка изменит дорожку, каков будет порядок физических операций, какие данные уже лежат в кэше и сколько потребуется служебных чтений. Даже если все 256 кластеров представлены одним экстентом, наша программа не получит автоматически гарантии, что доступ идёт вдоль одной физической дорожки без скрытых переходов.

Представим теперь, что снимок вместе с другими файлами лежит в папке «Поездка». Открывая его, файловая система сначала ищет сведения о файле, затем получает карту его содержимого, а устройство обрабатывает команды к блокам. В каком-то месте на этом пути может возникнуть ожидание. Например, нужные служебные сведения могут оказаться в памяти операционной системы, но сами блоки изображения — не оказаться; или, напротив, нужный участок данных уже был прочитан ранее. Поэтому два открытия одного файла не обязаны давать одинаковое время.

В учебных примерах полезно рисовать одну стрелку от имени файла к сектору. Но инженерная картина должна включать развилки. Кэш, сжатие, шифрование тома, другие файлы, особенности интерфейса и служебные действия диска способны усложнить путь. Для обычной фотографии это не беда: человек видит результат. Для лаборатории это перечень факторов, которые нельзя забыть при интерпретации миллисекунд.

Если хочется получить честный замер времени чтения именно файла, следует измерять выбранную программную операцию и фиксировать её условия. Если хочется проверить профиль чтения по зонам LBA, лучше работать с диапазонами адресов накопителя, понимая, что полученное время относится ко всему тракту ввода-вывода. А если вопрос заключается в реальном пути головки, потребуются данные о внутреннем состоянии, которые обычная программа не выдаёт. Одно слово «чтение» здесь скрывает три разных задачи.

Пример, а не измерение. Расчёт «1 МиБ = 256 кластеров по 4 КиБ» справедлив только при принятом размере кластера. Он не является утверждением о параметрах конкретного тома T:, которые необходимо проверять отдельно.

2.10. Почему близкие LBA могут вести себя неожиданно

Теперь вернёмся к нашей будущей серии опытов. Представим, что программа последовательно читает два блока с соседними логическими адресами. По простой модели поток должен быть удобен для диска: контроллеру известен порядок запросов, а соседние данные часто удаётся обслуживать эффективнее, чем случайно разбросанные. Но из этой общей тенденции не следует обещание одинаковой задержки на каждой операции.

Во-первых, разные уровни кэша могут сделать некоторые ответы намного быстрее остальных. Когда операция завершается за доли миллисекунды, у нас нет достаточного основания говорить, что за это время головка перескочила между магнитными дорожками. Возможно, необходимое содержимое было доступно из ранее подготовленной памяти. Чтобы разбираться в таких случаях, нужны специально спланированные независимые адреса, контроль повторений и анализ распределения, а не одна средняя цифра.

Во-вторых, накопитель может выполнить собственную внутреннюю работу, которую программа не видит как отдельное действие. Отдельное чтение может задержаться из-за восстановления данных, ожидания состояния устройства, управления очередью команд или других решений контроллера. Windows и USB-мост тоже вносят задержки. Экспериментатор видит начало и конец своей операции, но не каждую микроскопическую причину внутри неё.

В-третьих, адреса LBA могут находиться в разных областях логического пространства. На традиционном HDD это нередко связано с изменением эффективной скорости последовательной передачи, но точное соответствие физическим зонам в современных устройствах является внутренним делом производителя. Даже когда общий профиль скорости похож на ожидаемую картину внешних и внутренних дорожек, в деталях он может быть немонотонным.

Вот почему на следующем этапе нам понадобятся сначала опыты № 001 и № 002, показывающие, как меняется полезная скорость в разных логических областях диска, и только затем опыт № 003 с близкими и далёкими запросами. Без предварительного понимания адресации такое измерение легко назвать «картой головки» — но оно ею не является.

Мы ещё увидим, что порядок запросов бывает важнее одного расстояния. Упорядоченные и перемешанные трассы опыта № 008 показали существенное отличие полного времени чтения, однако в эксперименте были различные наборы адресов и присутствовали необычные стартовые задержки. Эти обстоятельства не позволят приписать всю разницу одному физическому перемещению. Так, постепенно, мы учимся спрашивать диск точнее.

2.11. Зачем здесь нужна «Гантова зебра»

Мы подошли к самому рискованному и одновременно самому любопытному месту книги. Первоначальная идея «Гантовой зебры» связана с группировкой и чередованием. Можно нарисовать крупные полосы данных, разделить их на участки и описать последовательность чтения так, чтобы число логических переходов было удобно рассчитывать. На временной диаграмме это похоже на разноцветные полосы, где одни операции располагаются одна за другой, другие — чередуются.

Такой рисунок полезен для мышления. Он позволяет сравнивать алгоритмы на уровне модели: какой объём данных читается последовательно, как часто запрос переходит в новый диапазон, сколько раз возвращается к уже посещённой области. Для планирования программы этот язык может оказаться нагляднее абстрактной последовательности огромных чисел LBA.

Но схема должна оставаться честной. Если две оранжевые полосы нарисованы рядом, их нельзя автоматически объявлять соседними магнитными секторами. Если линия на диаграмме становится короче, нельзя заключать, что каретка HDD прошла такое же меньшее расстояние. Логическая оптимизация и физическое размещение — не одно и то же. Польза от предложенной схемы требует отдельного сравнения с корректной исходной системой, одинаковыми данными и воспроизводимыми измерениями.

Для этого необязательно сразу писать собственный драйвер. Прежде следует определить, что именно можно улучшить: уменьшить число мелких запросов, упорядочить чтение файлов, сократить накладные расходы каталогов, уменьшить число переходов в пространстве LBA? Эти задачи частично доступны на уровне обычных программ. Но если целью станет управлять точным расположением физических магнитных областей, то драйвер без доступа к внутренней прошивке и трансляции диска не станет магическим инструментом.

Поэтому вторая часть книги не обещает создать «новую физику диска». Она ставит более скромную задачу: отделить полезную рабочую модель от аппаратных фактов и определить, в каком именно масштабе есть смысл говорить о чередовании. Магнитные домены существуют на поверхности. Дорожки существуют как физическая организация. LBA — интерфейс. Кластеры NTFS — уровень файловой системы. А «зебра» — наш способ рассуждать об организации запросов и участков данных.

Чем яснее проведена граница между этими этажами, тем меньше риска обмануть себя красивыми графиками. И, наоборот, тем интереснее становится настоящий эксперимент: он способен подтвердить некоторые ожидания, разрушить другие и подсказать новые вопросы. В инженерии это важнее, чем заранее объявлять победителем понравившийся алгоритм.



Чередование в модели — не снимок магнитных доменов и не известный маршрут головки.

Рисунок 2.6. Три разных изображения: авторское чередование, последовательность логических блоков и неизвестная по нашим замерам физическая магнитная карта. Вопросительные знаки — граница знания, а не пропущенные данные.

2.12. Что это означает для владельца домашнего архива

Теперь можно вернуться туда, откуда началась вся история, — к домашнему фотоальбому. Обычному человеку необязательно знать, где внутри HDD проходит конкретная дорожка с фотографией. Для него важнее, чтобы файл сохранился, копия была доступна и спустя годы фотографию можно было открыть. Но понимание уровней хранения позволяет избежать нескольких типичных ошибок.

Первая ошибка — путать «видно в проводнике» и «надёжно сохранено». Если файл отображается в папке, это означает, что файловая система смогла показать сведения о нём. Это ещё не проверка всех байтов содержимого и не гарантия, что накопитель сохранит работоспособность. Для семейных снимков важнее независимые копии, контроль их читаемости и, по возможности, сравнение контрольных сумм при переносе больших архивов.

Вторая ошибка — считать скорость USB-порта характеристикой магнитных дорожек. Быстрый интерфейс полезен, но не превращает механический HDD в SSD. И наоборот, медленный USB-мост может скрыть часть возможностей самого накопителя. Если внешний диск работает хуже ожидаемого, полезно начать с проверки подключения, кабеля, режима интерфейса и условий питания, а не с немедленной дефрагментации или вскрытия корпуса.

Третья ошибка — бездумно «лечить» любой медленный файл. На скорость влияют размер запросов, размещение кластеров, фоновые операции, кэш и состояние самого диска. При признаках аппаратных неисправностей дополнительные испытания способны увеличить нагрузку. В такой ситуации первоочередная задача — сохранить ценные данные, а уже затем заниматься диагностикой.

Четвёртая ошибка связана с будущей совместимостью. Исходный код программы десятилетней давности может легко читаться как текст, а готовая исполняемая программа — не запускаться в новой системе. Поэтому полезно хранить не только двоичный файл, но и исходные материалы, инструкции, сведения о формате и необходимые данные. Сохранность информации — это не только магнитные свойства носителя, но и возможность её правильно интерпретировать.

Наконец, пятая ошибка — чрезмерное доверие одному типу памяти. HDD хорош для вместительного архива, SSD удобен для рабочих файлов, оптический носитель может быть полезным дополнительным форматом, а облако — ещё одним местом копии. Правильная система хранения определяется не спором о «лучшем диске», а ценностью информации и тем, сколько независимых путей её восстановления остаётся у владельца.

2.13. Подведём итог: карта, которую мы видим, и карта, которую скрывает диск

В этой главе мы прошли путь в обратную сторону — от фотографии в папке до намагниченного слоя. Узнали, что название файла относится к уровню файловой системы, VCN описывает положение внутри потока файла, LCN — размещение кластеров на томе, а LBA — логический адрес устройства. Под ними находится контроллер HDD, который выполняет скрытое для обычной программы сопоставление адресов с внутренней организацией записи.

Теперь становится понятнее, почему крохотная магнитная область и гигабайтный архив не связаны прямой нарисованной линией. Между ними работает целая иерархия. Она сделала современные накопители удобными: пользователю достаточно открыть файл, операционной системе — передать команды, контроллеру — разобраться с деталями исполнения. Но за это удобство исследователь платит ограниченной наблюдаемостью.

На нашем конкретном HGST мы знаем паспортную организацию Advanced Format 512e, объём, скорость вращения и интерфейс. Мы можем программно измерять скорость и задержку, можем читать доступные метаданные NTFS, строить временные трассы и повторять опыты. Не можем только по этим измерениям восстановить достоверную карту магнитных доменов, секторов и фактических перемещений каждой головки. Любая схема, изображающая это как уже измеренный факт, была бы ложным упрощением.

Поэтому следующая глава начинается не с обещания сделать диск быстрее, а с первого проверяемого вопроса: изменяется ли скорость последовательного чтения при обращении к разным частям доступного диапазона LBA? Для этого мы сначала использовали пять логических зон, а затем двадцать одну. Посмотрим, зачем нужны оба шага, какие числа получили и где заканчивается их объясняющая сила.

Именно таким будет путь этой книги: понятный вопрос, ясная схема, честный программный опыт на настоящем магнитном диске и осторожный вывод. «Гантова зебра» при этом остаётся не легендой о чудесной записи, а поводом внимательнее разобраться в устройстве хранилища, которым мы пользуемся каждый день.

Мост к главе 3. Мы знаем, что поверхность HDD имеет дорожки и зоны, а программа видит LBA. Теперь измерим, как меняется наблюдаемая скорость по логическому пространству, не объявляя результаты прямой магнитной картой.

Глава 3. Почему один участок диска быстрее другого?

Опыты № 001 и № 002 — первые измерения на настоящем магнитном диске. Мы хотели увидеть, меняется ли скорость чтения при переходе по логическому пространству HDD. В итоге выяснили кое-что важное: измерительный тракт способен скрыть различия, которые мы надеялись обнаружить.

3.1. Пять фотографий, одна папка и один неожиданный вопрос

Представим домашний фотоархив. Фотографии из летнего путешествия лежат в одной папке, зимние — в другой, рядом находятся семейные видеозаписи. Пользователь видит названия файлов, даты и изображения. Но жёсткий диск получает запросы на чтение блоков с номерами. Для него не существует понятия «зима» или «отпуск». Он знает только, что требуется вернуть определённые байты.

На первый взгляд задача проста: если попросить накопитель прочитать одинаковый объём данных из разных частей его адресного пространства, можно увидеть, отличаются ли эти обращения по скорости. Допустим, на одном участке окажется быстрее. Возникает соблазн сразу решить, что головка находилась ближе к внешнему краю пластины, а на другом — к центру. Но в этом рассуждении пропущено несколько ступеней: от номера LBA до реального расположения данных, от команды программы до ответа диска и от механики до USB-интерфейса.

Именно поэтому первые эксперименты «Гантовой зебры» были не проверкой готового алгоритма размещения, а разведкой местности. Мы начали с вопроса, на который можно ответить измерением: какую полезную скорость чтения показывает компьютер, обращаясь к разным логическим диапазонам одного и того же HDD?

Здесь пригодится правило для всей книги: объяснение допустимо строить лишь настолько подробно, насколько позволяет способ измерения. Если мы наблюдаем скорость передачи через Windows и USB, нельзя без отдельного измерения объявлять её скоростью поверхности диска или временем движения головки.

3.2. Наш лабораторный образец и что значит «настоящий диск»

Работал реальный 2,5-дюймовый HDD HGST HTS725050B7E630, номинальная ёмкость 500 ГБ, частота вращения по паспорту — 7200 оборотов в минуту. Накопитель был подключён к ноутбуку Acer Swift 3 с Windows 11 Pro через USB–SATA-переходник JMicron. Система видела его как физический диск № 2; том T: использовал NTFS. Измерительные программы на Python запрашивали чтение с физического устройства, не создавая и не перезаписывая файлы на исследуемом накопителе.

Словосочетание «программное измерение на физическом диске» в этой книге намеренно точное. Чтение действительно выполнял HDD, но время фиксировала программа. Время прохождения команды через операционную систему, USB-переходник и контроллер не было исключено из результата. Мы не фотографировали поверхность под микроскопом, не управляли кареткой напрямую и не обходили прошивку накопителя.

Контроль личности устройства имеет не только научный, но и практический смысл. В ноутбуке есть собственные системные накопители. Ошибиться номером диска — значит измерить не тот образец, а при неосторожной программе и вовсе рисковать данными. Поэтому наша программа сверяла модель и ёмкость, а запись на T: не входила в исследовательскую процедуру.

В лабораторных журналах отдельно записаны два числа ёмкости, которые возвращали разные интерфейсы Windows: 500 106 780 160 и 500 105 249 280 байт. Их расхождение не превращается в вывод о «потерянных секторах»: оно лишь показывает необходимость сохранять исходные признаки идентификации и не смешивать данные разных системных запросов.

3.3. Что означает «точка 5 %» и почему это не радиус пластины

Чтобы не обследовать сотни миллиардов байтов подряд, мы выбрали контрольные положения: например 5 %, 27,5 %, 50 %, 72,5 % и 95 % от логического диапазона адресов. Процент показывает место внутри последовательности LBA, то есть в условной нумерованной полосе адресов. Он не означает, что головка в этот момент находилась на пяти процентах радиуса магнитной пластины.



Рисунок 3.1. Пять точек опыта № 001 внутри логического диапазона адресов. Каждое чтение охватывало 16 МиБ. Линейка — схема LBA, а не физическая карта дорожек.

Почему выбрали не самый первый и не самый последний адрес? Крайние части пространства могут содержать служебные особенности, а тесту достаточно характерных участков внутри доступного диапазона. Но мы не получаем права утверждать, что каждая из пяти отметок равна строго определённой зоне магнитной поверхности.

Представьте городскую карту и список адресов квартир в компьютерной базе. Даже если номера идут по порядку, из одной нумерации нельзя восстановить все маршруты через подъезды и этажи. Точно так же LBA — удобная логическая последовательность, но внутреннее отображение на дорожки и физические зоны находится под управлением накопителя. В обычном пользовательском интерфейсе оно не раскрывается.

Поэтому вывод «дальше по LBA» и вывод «дальше переместилась головка» относятся к разным уровням. В первых двух опытах мы проверяли первый уровень — последовательность логических диапазонов и доступную нам скорость чтения. С механическими предположениями предстояло обращаться осторожно.

3.4. Почему размер каждого чтения составил 16 МиБ

Во время эксперимента программа могла запрашивать 4 КиБ, 64 КиБ, 1 МиБ или намного больше. В первых двух опытах был выбран участок объёмом 16 МиБ на один замер. Это достаточно большой запрос, чтобы измерять не только единичную задержку маленького обращения, но и общую скорость передачи некоторого массива данных.

МиБ — мебибайт, двоичная мера объёма. Один МиБ равен 1 048 576 байтам. Значит, 16 МиБ — это 16 777 216 байт. В графиках мы используем МБ/с — миллионы байтов в секунду, как было записано в результатах. Путаница между десятичными МБ и двоичными МиБ способна давать небольшие, но реальные расхождения при самостоятельной проверке расчёта.

Если программа передала 16 777 216 байт примерно за 0,44 секунды, вычисленная полезная скорость близка к 38 МБ/с. Такая скорость не включает подробное объяснение того, на какие действия ушла каждая миллисекунда. Она представляет собой итоговое отношение полученного объёма ко времени операции, которое наблюдала программа.

Большой запрос снижает относительное влияние части постоянных накладных расходов по сравнению с очень коротким обращением. Но он не гарантирует, что мы измеряем максимальную скорость поверхности. На итог могут влиять ограничения порта USB, драйвера, кеша или размер внутренних порций данных. Позднее специально появится опыт с разными размерами чтений; здесь важно лишь понимать, почему именно 16 МиБ стало одинаковой единицей сравнения.

3.5. Опыт № 001: пять зон, три прохода

Первое исследование состоялось 12 ноября 2026 года. Программа взяла пять контрольных точек адресного пространства HGST. В каждой читалось по 16 МиБ. Весь маршрут повторили трижды: сначала с возрастанием адресов, затем в обратном направлении, затем снова по возрастанию. Таким образом, было выполнено 15 чтений общим полезным объёмом 240 МиБ.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, купив полную легальную версию на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.