**Практическая работа №9**

**Тема: «Файловые системы ОС Windows»**

**Цель работы:** приобрести навыки анализа физической и логической структуры дисков и закрепление знаний по файловым системам *FAT* и *NTFS*.

**Методические указания**

На физическом уровне работа с устройством внешней памяти всегда проводится через соответствующий контроллер. Именно контроллер знает физические параметры устройства, поэтому физическая модель магнитных дисков не зависит от операционной системы (*Linux* или *Windows*).

Диск *Windows* и *Linux*, может быть разбит на несколько разделов, которые могут быть первичными, расширенными или логическими. Параметры разделов хранятся в таблице разделов диска (*partition table*). В отличие от *Linux*, разделы диска называются логическими дисками и обозначаются латинскими буквами, за которыми следует двоеточие – *A*:, *B*:, *C*:, *D*: и т.д. Файловые системы каждого раздела, в отличие от *Linux*, не связаны между собой и функционируют отдельно друг от друга.

Небольшая часть адресного пространства каждого раздела выделяется под системную область, основная часть – под область данных. Единицей дисковой памяти в области данных является кластер, размер которого кратен размеру сектора и может достигать 64 Кбайт. Все кластеры имеют сквозную нумерацию, причем первый допустимый номер кластера равен 2. Файлы на диске могут храниться в несмежных кластерах, т.е. в различных частях диска.

Файл *Windows* – это поименованная совокупность информации, хранящаяся на ВЗУ. В виде файлов на диске хранятся программы и данные. Каталог – файл специального формата, предназначенный для хранения метаданных о зарегистрированных в этом каталоге файлах и подкаталогах (имя, расширение, атрибуты, размер, дата и время создания или последнего изменения, адрес). Каталоги каждого логического диска организованы в единую древовидную структуру. Имена файлов и каталогов, в отличие от *Linux*, регистронезависимы.

Структура системной области диска зависит от типа файловой системы, которая определяет способы доступа к информации в области данных. Наиболее известными файловыми системами для *Windows* являются *FAT* и *NTFS*.

**Файловая система FAT**

На рис. 1 приведена логическая модель диска с файловой системой FAT. В системной области находятся загрузочная запись, таблица размещения файлов и корневой каталог. Загрузочная запись, иногда называемая начальным загрузчиком, имеет размер 512 байт, всегда хранится в нулевом секторе и используется в процессе загрузки операционной системы.

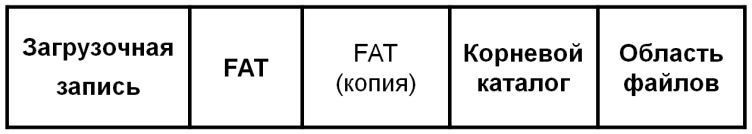


Рис. 1. Структура логического диска *FAT*

***Таблица размещения файлов***, в оригинальной литературе называемая *FAT* (*File Allocation Table*), содержит информацию о размещении файлов в области данных. Она всегда занимает сектора, начиная с первого. На любом диске для обеспечения надежного доступа к данным всегда хранится две копии *FAT*, которые обновляются одновременно.

Таблица размещения файлов содержит информацию о номерах кластеров, выделенных для хранения каждого файла. Она представляет собой карту (образ) области данных, в которой описывается состояние каждого кластера диска. Размер таблицы зависит от объема диска. Номер начального кластера, выделенного файлу, записывается в элемент каталога этого файла.

Каждый элемент таблицы соответствует одному кластеру в области данных. Дефектные кластеры помечаются как "*bad*". Если кластер свободен, то соответствующий ему элемент *FAT* имеет значение "0". Если кластер выделен для какого-либо файла, то возможны два варианта:

* элемент содержит признак конца файла "*EOF*", если этот кластер является последним кластером, выделенным файлу;
* элемент содержит значение номера следующего кластера, выделенного файлу.

Таким образом элементы *FAT*, выделенные одному файлу, связываются в цепочки, позволяющие получить доступ к информации даже в том случае, если файл при записи разбивается на несколько фрагментов, хранящихся в несмежных кластерах диска. Элементы *FAT* могут быть 16- и 32-разрядными, в зависимости от этого файловые системы имеют названия *FAT*-16 и *FAT*-32 и работают с 16-разрядными и 32-разрядными дисковыми адресами соответственно.

Логическое разбиение области данных на кластеры, как совокупность секторов, взамен использования одиночных секторов имеет следующий смысл:

* уменьшается размер *FAT*;
* уменьшается возможная фрагментация файлов;
* ускоряется доступ к файлу, т.к. в несколько раз сокращается длина цепочек фрагментов дискового пространства, выделенных для него

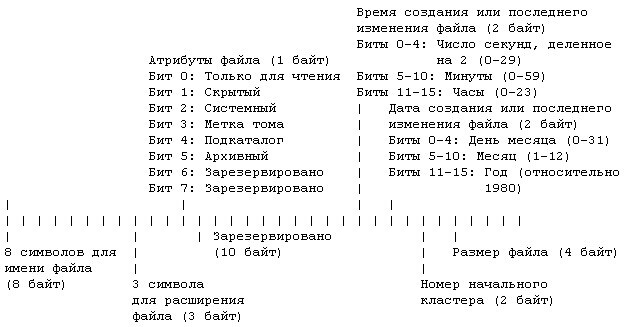
Следует иметь ввиду, что при увеличении размера кластера ухудшается коэффициент использования дисковой памяти за счет увеличения внутренней фрагментации. Минимальный размер кластера на диске с файловой системой FAT зависит от объема диска (*V*диска) и разрядности элемента *FAT*

(*r*): **

***Корневой каталог*** – главный каталог диска, который занимает сектора, следующие за *FAT*. Фиксированное число элементов и размещение в системной области корневого каталога являются принципиальным отличием от прочих каталогов.

Записи корневого каталога имеют длину 32 байта, структура записей представлена в табл. 1. Если файл не имеет расширения, то в соответствующем поле хранятся пробелы. Дата и время используются в виде четырехбайтового значения в операциях сравнения. Номер начального кластера определяет точку входа в *FAT* для данного файла и одновременно дисковый адрес собственно файла.





На рис.2 представлена схема работы и организации *FAT*, а также фрагментация, когда части файла разбросаны по всему диску.

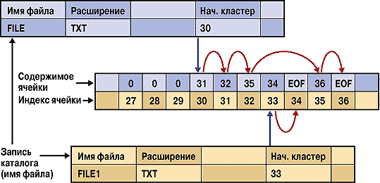


Рис.2. Схема работы и организации *FAT*

Цепочка кластеров для файла *FILE.TXT*: **30, 31, 32, 35, 36, *EOF.***

Цепочка кластеров для файла *FILE*1*.TXT*: **33, 34, *EOF.***

Файл, который занимает на диске более одного непрерывного участка, называется фрагментированным.

***Фрагментация диска*** – это появление на диске множества свободных участков, разделенных занятыми участками.

***Дефрагментация диска*** – это перемещение данных на разделе, после которого, кластеры содержащие части одного файла, размещаются последовательно.

Файловые системы *FAT* выделяют для записываемых на диск файлов некоторое количество кластеров, в зависимости от размера файла. В процессе работы с диском при записи и удалении файлов разного размера на диске появятся свободные и занятые области разной длины.

Такой метод хранения файлов позволяет использовать всё имеющееся на диске свободное место, т.к. если длина записываемого файла больше, чем размеры непрерывных свободных участков, то файл просто расположится в нескольких несмежных участках.

Реально время чтения сильно фрагментированного файла по сравнению с файлом, занимающим непрерывную область на диске, может отличаться в несколько раз! Внешне это выглядит так, как будто все программы стали работать в несколько раз медленнее, при этом наблюдается интенсивное перемещение головок диска от одного участка файла к другому.

***Удаление файлов***

При удалении файла обычно выполняются следующие действия:

* в таблице размещения файлов обнуляются все элементы, выделенные для описания этого файла;
* в соответствующем элементе каталога изменяется имя файла – вместо первого символа в поле имени записывается символ «*х*».

Остальные характеристики файла в элементе каталога, а также содержимое файла в кластерах диска, не изменяются, поэтому всегда есть возможность полностью или частично восстановить удаленный файл.

Полное восстановление возможно, если:

* не перезаписан соответствующий элемент каталога;
* имеется доступ к каталогу;
* кластеры, ранее занимаемые файлом, не выделены другим файлам или каталогам;
* удаленный файл был нефрагментированным.

При несоблюдении последнего условия полное восстановление не гарантируется, т.к. не всегда возможно извлечь данные о том, какие кластеры были выделены файлу.

Современные операционные системы обычно проводят удаление файлов в специальный скрытый каталог, который называется корзиной. Размер корзины может устанавливаться пользователем. Корзина обслуживается специальной программой, что делает восстановление ошибочно удаленных файлов удобным и быстрым.

**Файловая система *NTFS***

***Организация раздела NTFS***

Как и любая другая система*, NTFS* делит все полезное место на кластеры – блоки данных, используемые единовременно. *NTFS* поддерживает почти любые размеры кластеров – от 512 байт до 64 Кбайт, неким стандартом же считается кластер размером 4 Кбайт.

Диск *NTFS* условно делится на две части. Первые 12% диска отводятся под так называемую *MFT* зону – пространство, в которое растет метафайл. Запись каких-либо данных в эту область невозможна. MFT-зона всегда держится пустой – это делается для того, чтобы самый главный, служебный файл (*MFT*) не фрагментировался при своем росте. Остальные 88% диска представляют собой обычное пространство для хранения файлов (рис.3).



Рис. 3. Структура раздела *NTFS*

Основной структурой данных в *NTFS* является главная таблица файлов (*Master File Table, MFT*), которая хранится в системном файле *$MFT* и представляет собой главный каталог, в котором регистрируются все файлы раздела, включая системные файлы. Для *MFT* резервируется 12% от общего объема раздела в виде непрерывной последовательности блоков, которая называется *MFT*-зоной. Запись файлов и каталогов в эту зону не проводится, а ее адрес хранится в загрузочной записи.

*MFT* состоит из множества записей размером 1 Кбайт о файлах, расположенных на томе. В записи *MFT* хранится вся информация о файле (имя, дата и время создания, размер, положение на диске отдельных фрагментов, и т. д). Если не хватает одной записи *MFT*, то используются несколько, причем не обязательно подряд. При этом первая запись называется базовой. Каждая запись *MFT* имеет уникальный номер – индекс, общее количество записей – до 248.

Первые 16 записей файла *$MFT* выделены для хранения информации о системных файлах. Самая первая запись в *MFT* – это запись о самом файле *$MFT*. Во второй записи содержится информация о зеркальной копии *MFT* (файл *$MFTMirr*), в которой дублируются первые 4 записи таблицы *MFT*. В случае возникновения сбоя, если *MFT* окажется недоступным, информация о системных файлах будет считываться из файла *$MFTMirr*, адрес которого также имеется в загрузочной записи.

Ниже приведено назначение некоторых системных файлов *NTFS*:

* *$LogFile* – файл журнала, в котором записывается информация о всех операциях, изменяющих структуру раздела *NTFS*, например, создание файлов и каталогов. Файл журнала используется при восстановлении тома *NTFS* после сбоев;
* *$Volume* – файл информации о томе, в котором содержатся имя тома (Volume label), версия NTFS и набор флагов состояния тома, например, флаг, установка которого означает, что том был поврежден и требует восстановления при помощи системной утилиты Chkdsk;
* *$AttrDef* – таблица определения атрибутов, содержащая возможные на данном томе типы атрибутов файлов;
* *$Root Directory* – файл с информацией о корневом каталоге тома. В нем хранятся ссылки на файлы и каталоги, содержащиеся в корневом каталоге;
* *$BitMap* – файл битовой карты, каждый бит в которой соответствует одному кластеру: единичное значение бита соответствует занятому кластеру, нулевое – свободному;
* *$Boot* – файл загрузочной записи тома;
* *$BadClus* – файл плохих кластеров, содержащий информацию обо всех кластерах, имеющих сбойные секторы.

***Альтернативные потоки NTFS.***

Файловая система *NTFS* обладает интересной возможностью поддержки альтернативных потоков данных (*Alternate Data Stream, ADS*). Технология подразумевает под собой, то, что каждый файл в файловой системе *NTFS* может иметь несколько потоков, в которых могут храниться данные. Проводник и большинство других приложений работают только со стандартным потоком и не могут получить данные их альтернативных. Таким образом с помощью технологии *ADS* можно скрывать данные, которые не удастся обнаружить стандартными способами. Поддержкаальтернативных потоков данных была добавлена в *NTFS* для совместимости с файловой системой *HFS*, использующейся на *MacOS*.

Потоки данных файла описываются атрибутом *$DATA*, одним из свойств которого является имя потока. Основной поток является неименованным, а каждый альтернативный поток должен иметь собственное имя. Альтернативные потоки скрыты от пользователя и не отображаются большинством стандартных программ. Они могут содержать любой тип информации – текстовый, графический, видео и т.д. В альтернативный поток можно даже записать программу, что иногда используется для распространения вредоносного ПО.

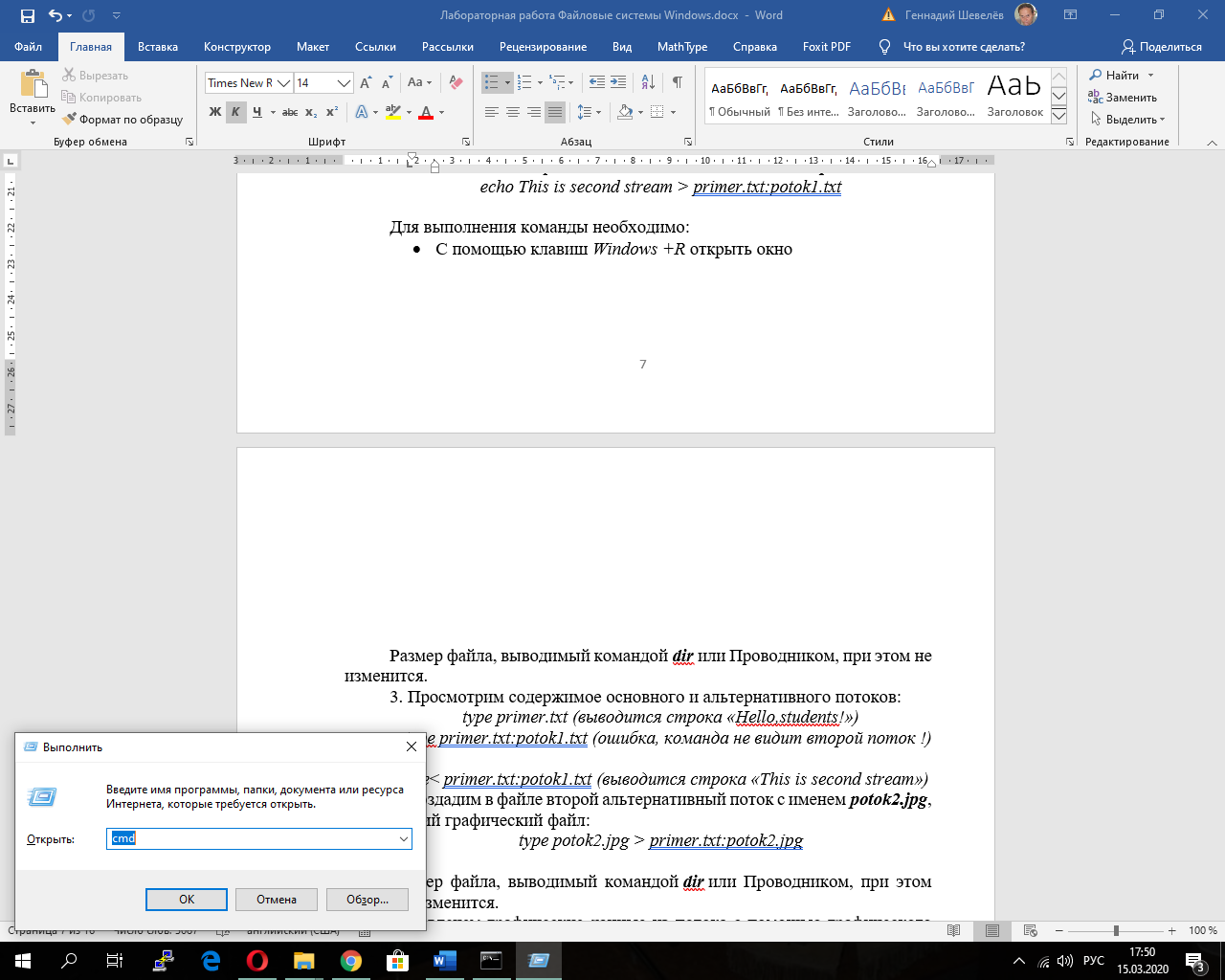
# **[Пример.](#_Пример.)**

1. Создаем в *Блокноте (Notepad)* файл ***primer.txt*** размером 15 байтов, содержащий строку*: «Hello, students!»*
2. Создаем в этом файле именованный поток с именем ***potok1.txt***:

*echo This is second stream > primer.txt:potok1.txt*

Для выполнения этой и следующих команд необходимо:

* Запустить командную строку с использованием диалога ***Выполнить*** (комбинации клавиш ***Win+R***).
* Набрать интерпретатор командной строки***cmd****.*
* *OK.*
* Вставить скопированную команду в командную строку.
* *Enter.*



Размер файла, выводимый командой ***dir*** или Проводником, при этом не изменится.

3. Просмотрим содержимое основного и альтернативного потоков:

*type primer.txt (выводится строка «Hello, students!»)*

*type primer.txt:potok1.txt (ошибка, команда не видит второй поток !)*

*more*< *primer.txt:potok1.txt (выводится строка «This is second stream»)*

4. Создадим в файле второй альтернативный поток с именем ***potok2.jpg***, содержащий графический файл:

*type potok2.jpg > primer.txt:potok2.jpg*

Размер файла, выводимый командой ***dir*** или Проводником, при этом также не изменится.

5*.* Извлечем графические данные из потока с помощью графического редактора:

*mspaint primer.txt:potok2.jpg*

6. Создадим в файле третий альтернативный поток с именем ***calcul.exe***, содержащий программу Калькулятор:

*type c:\windows\system32\ calc.exe > primer.txt:calcul.exe*

Размер файла, выводимый командой ***dir*** или Проводником, при этом опять не изменится.

7. Запустим Калькулятор из текстового файла:

s*tart .\ primer.txt:calcul.exe*

Действительный размер полученных в этом примере файлов можно увидеть командой ***dir/r.***

***Программное обеспечение для выполнения работы***

Основным инструментом исследования файловой системы магнитных дисков является специальная программа – дисковый редактор. Современные дисковые редакторы обладают большим набором возможностей: просмотр и редактирование системной области диска; просмотр и редактирование директорий и файлов; восстановление удаленных директорий и файлов; доступ к любому участку диска по номеру сектора или кластера; работа с образом диска; создание загрузочных дисков и т.д.

Для выполнения работы будем использовать редактор *DMDE* ([**http://dmde.ru**](http://dmde.ru)). Редактор *DMDE* имеет свободно распространяемую версию, которую можно скачать с сайта разработчика, поэтому рекомендуем работать именно с этой программой. Выполнение данной лабораторной работы требует прямого обращения к дискам, что несет потенциальную опасность для вычислительной системы, поэтому для работы необходимо использовать дисковый редактор ***только в режиме чтения***.

Если Вы выполняете работу на домашнем компьютере, то можно работать с реальными дисками или флэш – накопителями. В компьютерном классе в целях безопасности администратор системы может отключить возможность работы с дисками, в этом случае можно работать с заранее подготовленным образом диска или использовать виртуальную машину, работающую в любой среде виртуализации (*Oracle VirtualBox*, *VM Workstation* и т.д.)

После загрузки редактора необходимо выбрать тип диска – физический или логический. При выборе физического диска открывается таблица разделов, в которой хранится список логических дисков с указанием типа файловой системы, объема и границ каждого логического диска (рис. 4). Для отображения имени разделов диска можно нажать кнопку «Меню» и выбрать пункт «Показать буквы томов».

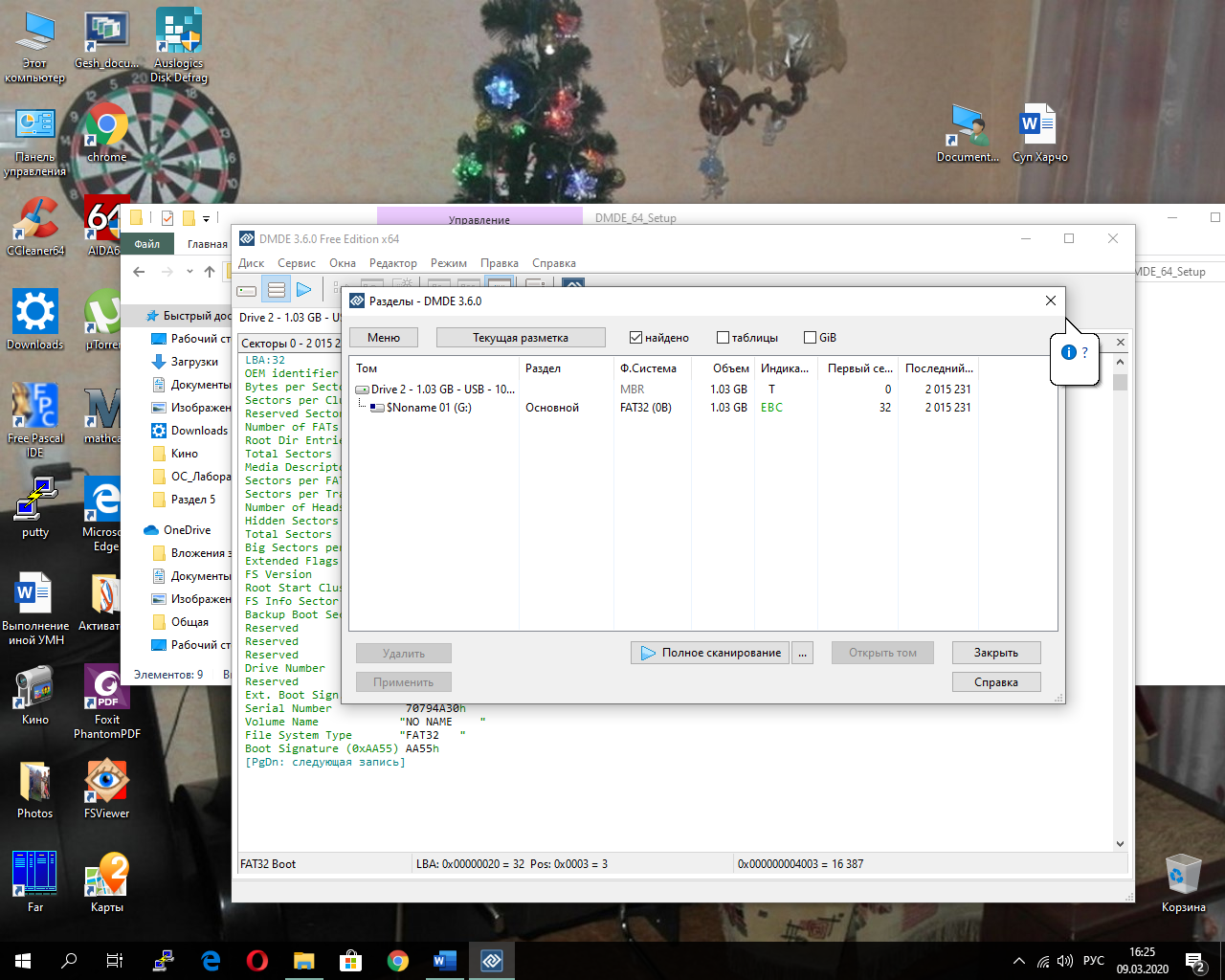


Рис. 4 Объем и другая информация логического диска

С помощью контекстного меню для каждого логического диска можно выполнить следующие действия: открыть, удалить или создать образ. Образ представляет собой файл, содержащий снимок диска, т.е. его точную физическую копию, которую можно использовать для восстановления диска в случае повреждения.

После открытия логического диска редактор выводит его параметры, набор которых зависит от типа установленной файловой системы: размеры сектора и кластера, число элементов корневого каталога или расположение файла *MFT* и т.д. (рис. 5).

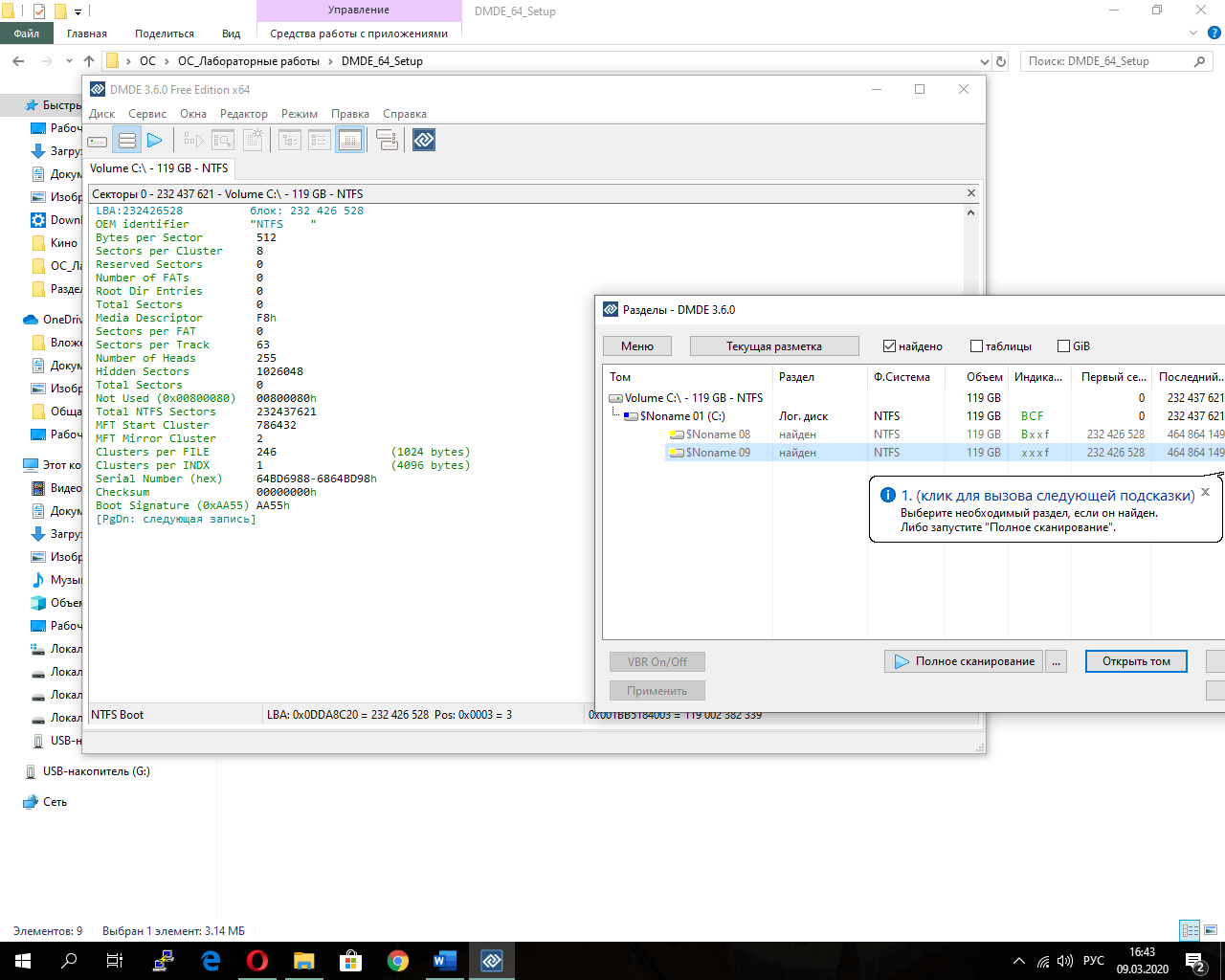


Рис. 5 Расположение файла MFT

Нажатие кнопки «Открыть» переводит редактор в режим просмотра, в котором имеется три панели (просмотр папок, просмотр файлов и панель редактора), показанных на рис. 6. В панель редактора можно выводить содержимое системной области и области данных диска. Управление панелью редактора проводится через меню Редактор.

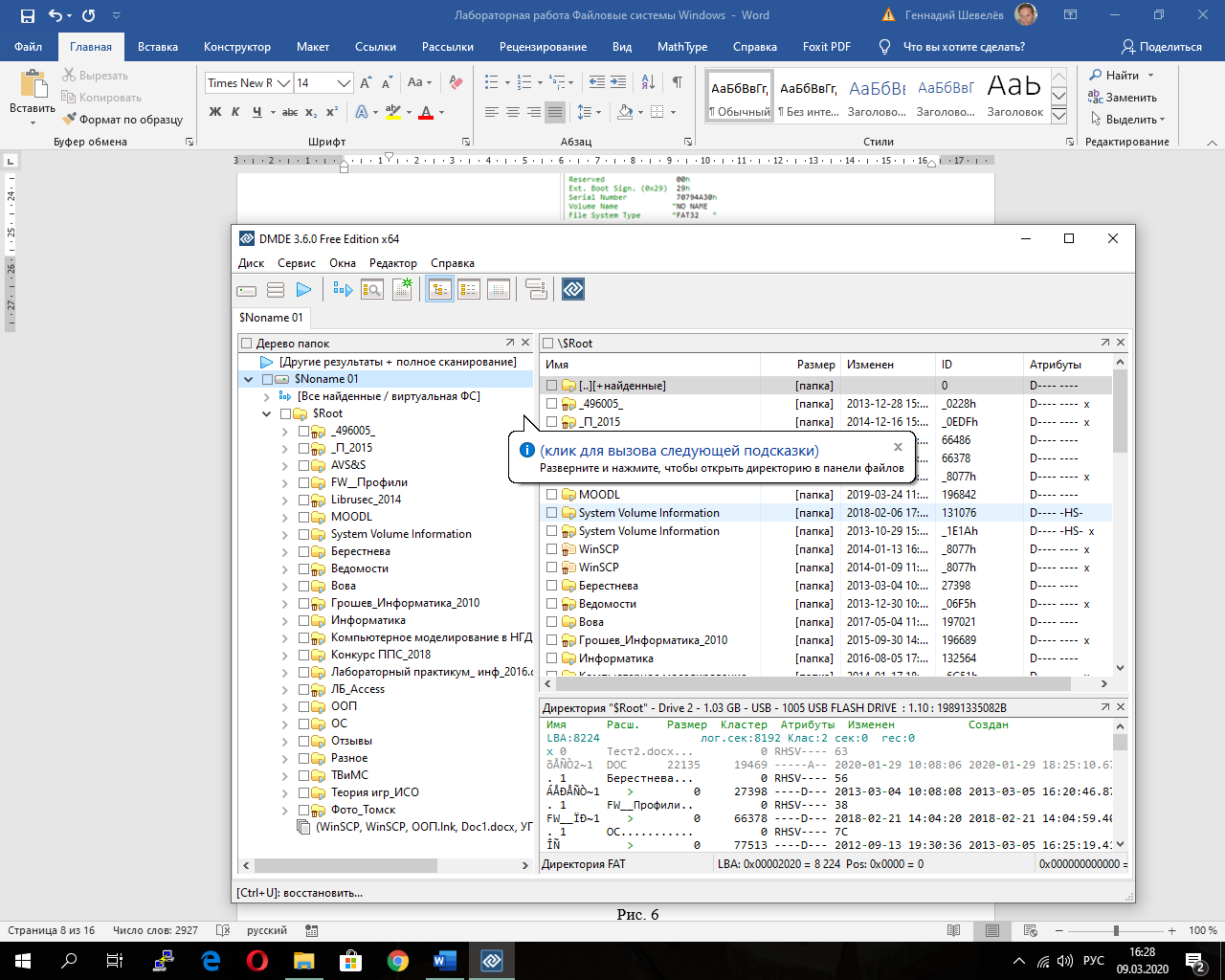


Рис.6 Вывод информации по каждому каталогу и файлу

***Работа с файловой системой FAT*32.**

Для *FAT*32 в панель редактора из системной области можно выводить загрузочную запись, таблицу *FAT* и корневой каталог, а из области данных – каталоги и файлы.

При просмотре таблицы *FAT* элементы, соответствующие свободным кластерам, выводятся символом «0», занятым кластерам – символом «=», а занятым последним кластерам – символом «*Е*». Реальные значения элементов *FAT* выводятся при установке курсора на элемент, при этом в строке статуса отображается название файловой системы и номер кластера, который соответствует текущему элементу *FAT* (например, *FAT*32 [17251]). По каждому каталогу и файлу выводится имя, расширение, размер, номер начального кластера, атрибуты и даты создания и изменения (см. рис. 6).

Просмотр содержимого файла, которое выводится в шестнадцатиричном и символьном виде, проводится двойным щелчком мыши по имени файла; изменение кодировки символов проводится в меню Режим/Кодировка. В этом режиме можно также посмотреть цепочку кластеров, выделенных данному файлу (меню Редактор/ Карта кластеров), как показано на рис.7.

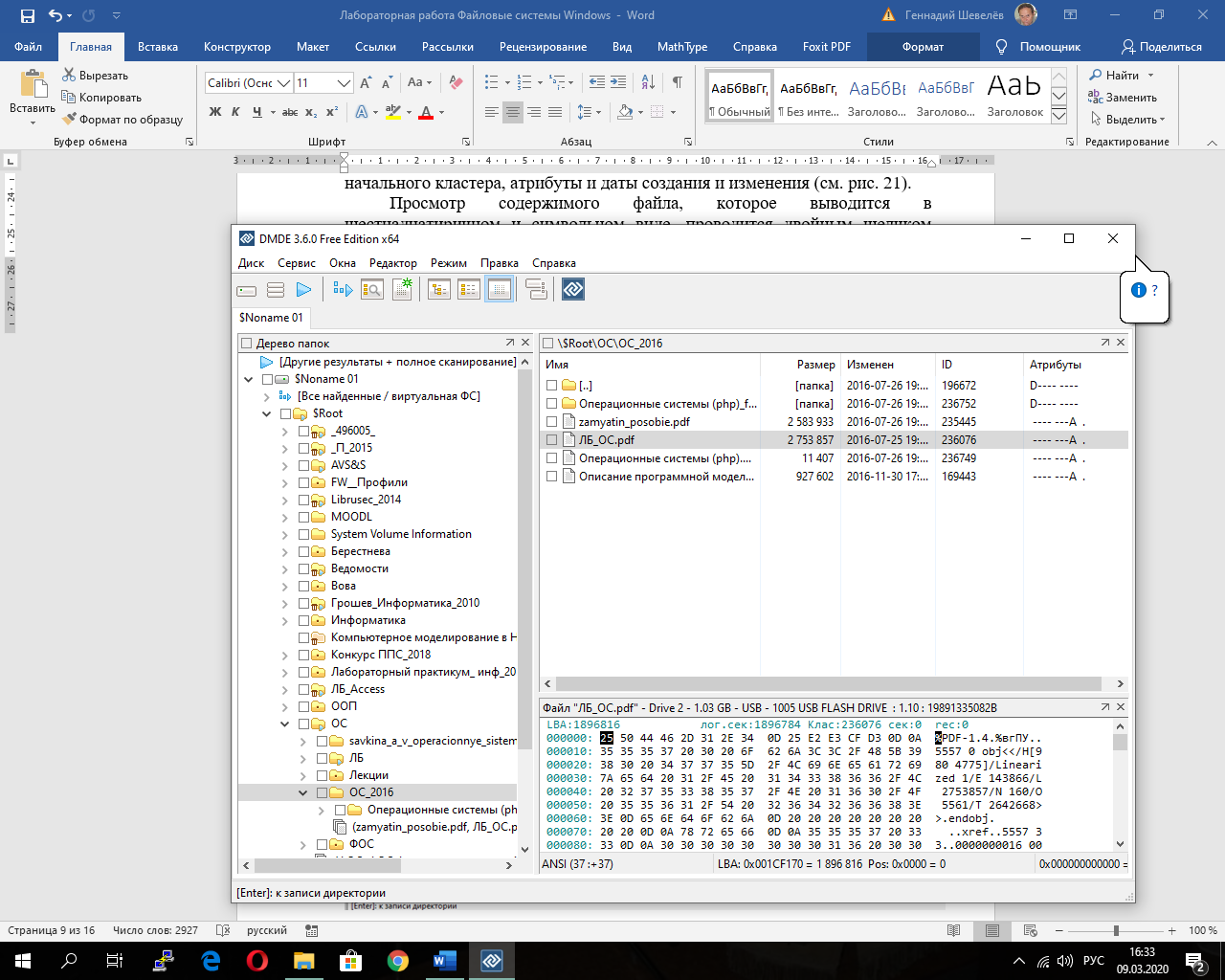


Рис.7 Меню Редактор/ Карта кластеров

Элементы каталога, имена которых начинаются с символа «*х*» соответствуют удаленным файлам. Если в поле имени стоят цифры или символы «е0», то этот элемент предназначен для хранения длинного имени файла.

*DMDE* позволяет осуществить быстрый переход на заданный кластер или сектор диска по их номеру (меню Редактор/Кластер или меню Редактор/Сектор тома), а также восстановить удаленные файлы.

Для восстановления необходимо отметить на панели нужные файлы, выбрать в контекстном меню пункт «Восстановить объект…» и указать каталог, в который надо провести восстановление. Для того, чтобы не испортить файл-оригинал, восстановление желательно проводить на другой логический диск.

***Работа с файловой системой NTFS.***

После выбора логического диска в окне редактора будет выведено содержимое файла *$MFT* (рис. 8).

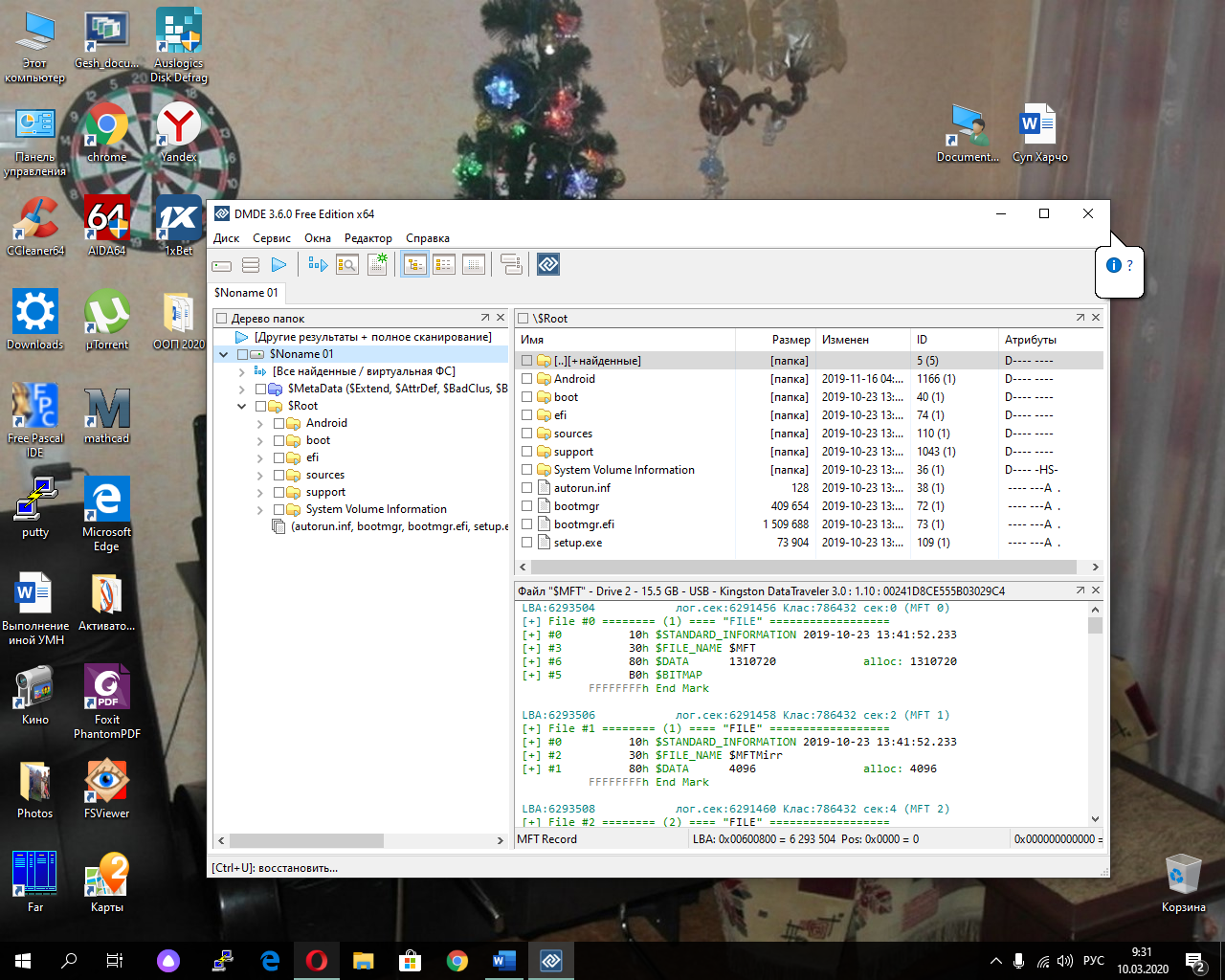


Рис. 8 Вывод содержимого файла $MFT

Первая запись описывает сам файл *$MFT*, а вторая – копию его первых четырех записей (*$MFTMirr*). Для каждой записи выводится ее адрес на диске (номера кластера и сектора), граничные метки, внутренний номер (индекс) и набор атрибутов. Минимальный набор включает атрибуты *$STAN-DART INFORMATION, $FILE NAME* и *$DATA*. Для просмотра содержимого каждого атрибута необходимо в его строке сделать щелчок мыши на символе ‘+’.

На рис. 9 показано содержимое атрибута *$DATA*, указывающего на расположение данных одного из файлов.

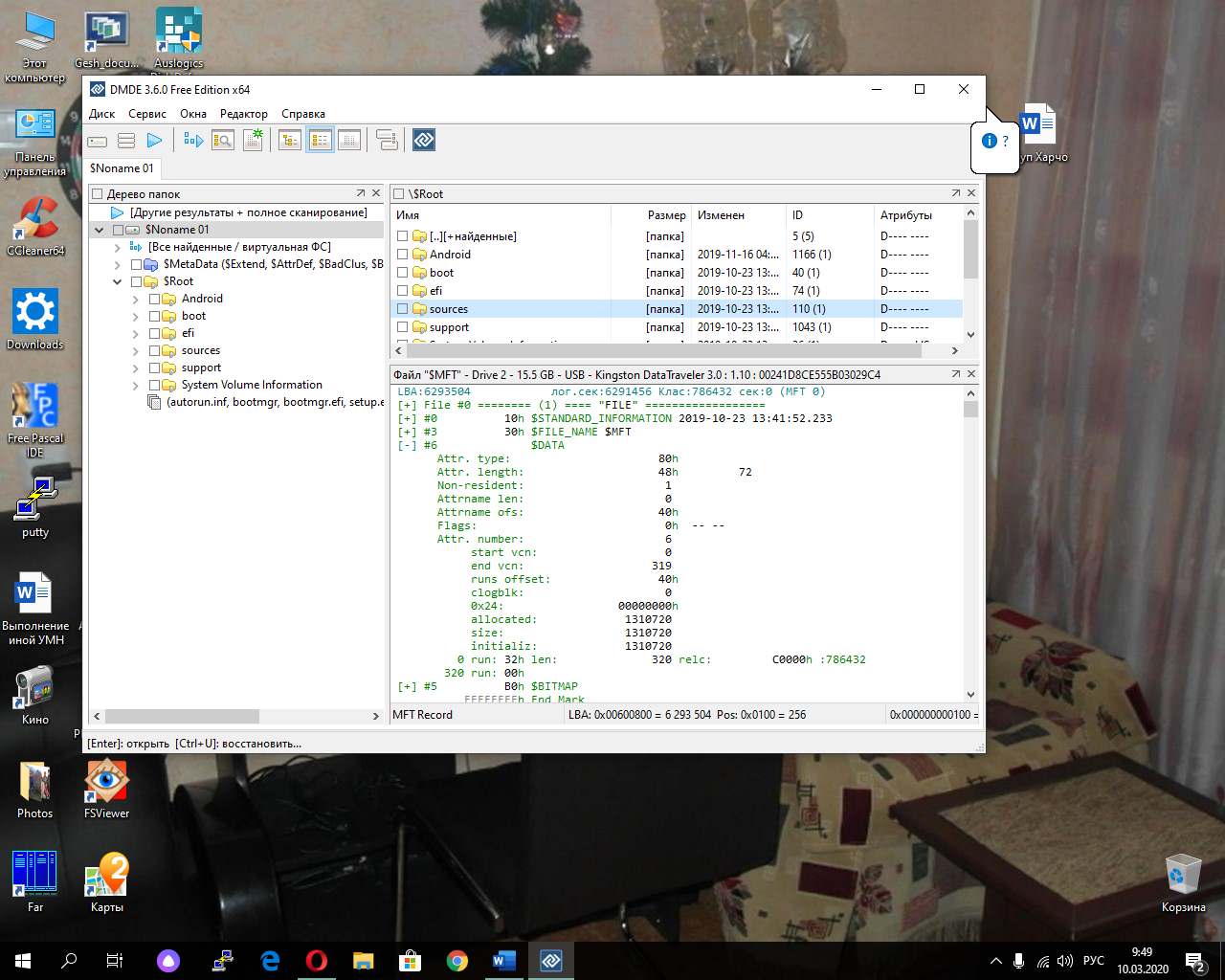


Рис. 9 Содержимое атрибута $DATA, указывающего на расположение данных одного из файлов.

Анализ рисунка позволяет сделать следующие выводы:

* индекс файла в MFT – 110;
* данные файла находятся на диске, т.к. атрибут является нерезидентным;
* данные занимают 320 кластеров (start vcn=0, end vcn=319) или 320\*8\*512 = 1310720 байтов, файл не фрагментирован;
* номер начального кластера файла – 786432;
* длина атрибута – 72 байта.

На рис.10 показан этот файл в режиме просмотра данных

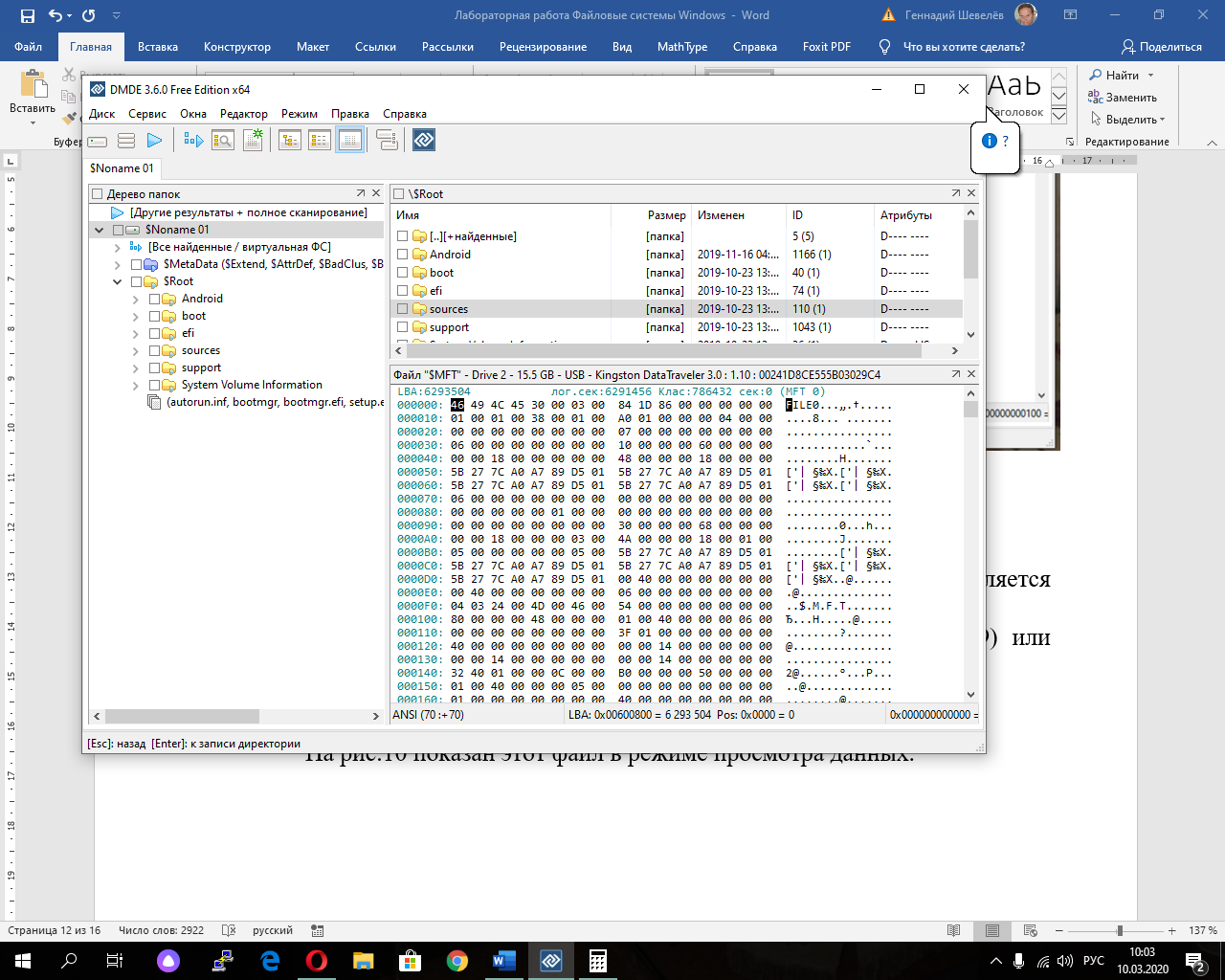


Рис. 10 Файл в режиме просмотра данных

***Порядок выполнения работы***

1. Откройте дисковый редактор DMDE и определите параметры виртуального диска: общий объем, число и типы разделов, тип файловой установленной файловой системы. Для FAT - раздела определите размеры сектора и кластера; число секторов, выделенных для таблицы FAT и размер корневого каталога. Для NTFS - раздела определите размеры сектора и кластера, размер файла $MFT и его адрес, размеры записи MFT и индексной записи. Занесите все параметры в отчет, подтверждая их скриншотами.

3. Откройте логический диск с файловой системой FAT32 и выполните следующие действия, подтверждая их скриншотами:

3.1 Создайте на диске каталог с именем, соответствующим Вашей учетной записи, и в нем создайте структуру каталогов согласно заданию лабораторной работы № 1.

3.2. В каталог *abc\_kk* запишите три файла размером 40 – 60 Кбайт, имеющих форматы *.txt*, *.doc* и *.docx* , имена файлов должны содержать не менее 15 символов, например, *Практическая работа № 9*.

3.3. С помощью программы *Проводник* скопируйте файл *Практическая работа № 9.txt* в каталог *trash\_kk***.**

3.4. Удалите файл *Практическая работа № 9.txt* из каталога *abc\_kk*, проведите анализ изменений в FAT и в каталоге *abc\_kk*. Посмотрите содержимое начального кластера удаленного файла, результат занесите в отчет.

3.5. Восстановите удаленный файл *Практическая работа № 9.txt.*

4. Откройте логический диск с файловой системой NTFS и выполните действия, подтверждая их скриншотами.

4.1. Создайте на диске структуру каталогов и файлов согласно п.3.1 и п.3.2.

4.2. Определите характеристики файла $MFT (начальный адрес, число записей, размер в байтах и кластерах).

4.3. Определите число записей в файле $MFTmirr.

4.4. Проведите полный анализ записи MFT, соответствующей файлу *Практическая работа № 9.txt* и занесите в отчет описания всех атрибутов, включая расположение файла на диске.

4.5. Удалите файл *Практическая работа № 9.txt,* проведите анализ изменений в MFT и в области данных. Результаты занесите в отчет.

4.6. Восстановите удаленный файл.

4.7. С помощью программы Блокнот создайте текстовый файл **primer.txt (см.** [**Пример.**](#_Пример.)**)**, записав в него фразу «Very good weather today!». Проведите анализ соответствующей записи MFT, определить адрес этого файла на диске.

4.8. Запишите в файл **primer.txt** второй поток данных, используя для этого, например, любой текстовый файл размером не менее 50 Кбайт. Проведите анализ соответствующей записи MFT и определите расположение данных этого потока на диске. Определите размер файла, сравните с предыдущим пунктом.

4.9. Запишите в файл **primer.txt** третий поток данных, используя для этого любой графический файл (например, фотографию). Проведите анализ соответствующей записи MFT и определите расположение данных этого потока на диске. Определите размер файла, сравните с предыдущим пунктом.

**Контрольные вопросы:**

1. Каким образом поддерживается древовидная многоуровневая система каталогов в Windows?
2. Назовите основные различия файловых систем FAT и NTFS.
3. Какова структура файла MFT?
4. Алгоритмы восстановления файлов в FAT и NTFS.
5. Каким образом в NTFS увеличена скорость доступа к файлам по сравнению с FAT?

**Содержание отчета**:

1. Название работы.
2. Цель работы.
3. Задание и его решение.
4. Вывод по работе.