

КПД

11 К портам А и В , настроенным на режим 1 (оба на ввод) подключены два перифер. Устройства, написать фрагмент программы, который задает конфигурацию, разрешает прерывания от 1-го и запрещает прерывания от 2-го перифер.устройства. (+2)

```
mov dx,303h
mov al,10111111b
out dx,al

mov dx,303h
mov al,00001001b
out dx,al ; разрешает INT , порт А

mov dx,303h
mov al,00000100b ; запрещает INT , порт В
out dx,al
```

ПерУстр1 - к порту А, ПерУстр2 - к порту В.

15 Напишите фрагмент программы, который тестирует состояние 2-го канала КПД и помещает в регистр ВХ единицу если канал занят пересылками и его работа разрешена. (+2)

```
Mov dx,8h
in al,dx
test al,01000000b
jz OK
mov bx,0
jmp out
OK: mov bx,1
OUT:
```

12. Написать фрагмент программы передачи 4Кбайт из области памяти начинающейся с 1А00h в область, начинающейся с 9F000h

```
CLI
MOV AL, 100b ;disable 0-canal
OUT 0Ah, AL
MOV AL, 101b ;disable 1-canal
MOV AL, 10001000b
OUT 0Bh, AL ;0-canal, чтение из памяти, один. передача, increment CAR
;блочная пер.

MOV AL,10000101b
OUT 0Bh, AL ;то же но 1 канал, запись в память
MOV AL, 00h

OUT 0, AL ;адрес source
MOV AL, 1Ah
OUT 0, AL

MOV AL, 0
OUT 2, AL ;Lo, Hi адр
MOV AL, 0F0h
OUT 2, AL

MOV AL, 9 ;ещё «не влезло»
OUT 80h, AL

MOV AL, 00h
OUT 1, AL ;количество байтов
MOV AL, 10h
OUT 1, AL
MOV AL, 0
OUT 3, AL ;количество байтов
MOV AL, 10h
OUT 3, AL
```

```
;включение

MOV AL, 000b
OUT 0Ah, 21      ;enable 0
MOV AL,001b
OUT 0ah, 21      ;enable 0-непол.

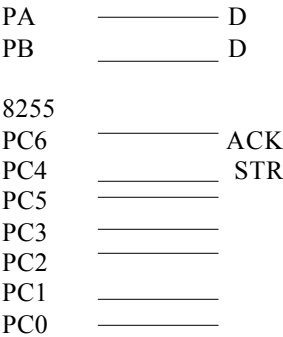
;обращение к PR

MOV AL, 100b
OUT 9h, AL
MOV AL, 101b
OUT 9h, AL      ;enable 0, 1
STI             ;разрешение прерывания

Notfinish:
    IN AL, 8h
    Bt AX, 0
    JNC Notfinish
```

**6. Написать управляющее слово, которое организует один выходной 8-разрядный квитируемый порт, один выходной 8-разрядный неквитируемый порт и один 5-х разрядный выходной порт. Нарисовать схему.**

10100000  
порт А – квитируемый порт  
порт В – неквитируемый порт



**44 Почему, по вашему мнению, фазовая модуляция (и ее модификация) получили более широкое распространение в сравении с частотной модуляцией. (+2)**

ФМ более помехоустойчива, чем ЧМ. Также ФМ позволяет увеличить плотность записи Информации

**#40. Напишите фрагмент программы, который настраивает 2 канал КПД на приём с контроллера гибкого диска 512 байт с адреса 1C000h. Контроллер не имеет буфера.**

```
outp macro P, D
    mov al, D
    out P, al
endm

    mov ax, 1C00h
    mov es, ax
    xor di, di
    mov cx, 512
    .....
    cli
```

```

    outp    0Ah, 011b    ; остановка второго канала DMA
    out     0Ch, al      ; сброс указателя последовательности байт

    xor     ebx, ebx
    mov     bx, ds
    shl     ebx, 4
    xor     eax, eax
    mov     ax, di
    add     ebx, eax      ; Физический адрес в EBX

    outp     4, bl
    outp     4, bh        ; запись адреса

    shr     ebx, 16
    mov     al, bl
    mov     dx, 81h
    out     dx, al        ; Запись страницы

    outp     5, cl
    outp     5, ch        ; Count

    outp     0Bh, 10000110b ; (46h) : FDC to Memory
                                ;   если 4Ah : Memory to FDC

    outp     0Ah, 2        ; Разрешение работы с каналом.
```

**#5. Назовите 4 причины, которые определяют целесообразность использования КПД.**

- 1. Совмещение процессов выполнения команд (когда шина адреса и данных не захвачены процессором) и обмена данных между ПУ и памятью ускоряют работу ЭВМ.
- 2. Режим обмена через DMA позволяет исключить в программе множественные проверки готовности ПУ при передаче каждого байта – это увеличивает производительность.
- 3. Наличие нескольких каналов DMA позволяет вести обмен данными между несколькими ПУ и памятью в разных направлениях (ВУ-ОП, ОП-ВУ) (хотя и последовательно).
- 4. Нет необходимости ожидать готовности ВУ к обмену: периферийное устройство само является инициатором начала обмена.

**18. Написать фрагмент программы настройки пересылки 256 байт из области памяти, начинающейся с адреса 5AF00 в жёсткий диск с использованием 3-го канала КПД. (+2)**

```

MOV     DX, 0Ah
MOV     AL, 00000111b
OUT     DX, AL
MOV     DX, 0Ch
OUT     DX, AL
MOV     DX, 0Bh
MOV     AL, 01101011b    ; 01001011
OUT     DX, AL
MOV     DX, 6
MOV     AL, 0
OUT     DX, AL
MOV     AL, 0AFh
OUT     DX, AL
MOV     DX, 82h
MOV     AL, 5
OUT     DX, AL
MOV     DX, 7
MOV     AL, 0FFh
OUT     DX, AL
MOV     AL, 0
OUT     DX, AL
MOV     DX, 0Ah
MOV     AL, 00000011b
OUT     DX, AL
```

41. Жёсткий диск имеет 4 головки, 33 сектора на дорожке. Определите физический адрес 1233-го сектора.

Дорожка – 9  
Головка – 1  
Сектор – 12  
1 дор. – 132 сект.  
1233/132=9.34  
9\*132=1188  
1233-1188=45  
45-33=12  
9\*132+1\*33+12=1233

Диски

26 Назовите 2 преимущества магнитных лент по сравнению с магнитным диском (+2)

- Одинаковая плотность записи на всех дорожках
- большой объем информации помещается - размер бабины

49. Назовите два преимущества частотной модуляции по сравнению с дифференциально-частотной или квадратурно-амплитудной модуляцией.

- 1 – простота оборудования
- 2 – минимальный стандарт для обязательной поддержки РСЗЛ

#1. Допустимая продольная плотность изменений намагниченности - 200 на 1 мм. Какова плотность хранения информации при использовании частотной модуляции. 100 бит на 1 мм.

#17. Назовите одно преимущество и два недостатка магнитного барабана по сравнению с магнитным диском.

- Недостатки:
- 1) Надёжность хранения информации "чувствительна" к неровностям поверхности и эксцентricности.
  - 2)Громоздкость
- Преимущества
- Одинаковая плотность записи на всех дорожках => одинаковая надёжность хранения информации по всему диску.

#39. Назовите 4 преимущества магнитных дисков по сравнению с оптическими дисками.

- 1. Более низкая стоимость.
- 2. Простота процесса записи и перезаписи.
- 3. Работа с диском не требует сложного оборудования.
- 4. Для Hard-дисков скорость обмена информацией больше.

FDC

#4. Написать фрагмент программы, который выполняет сброс контроллера, включает двигатель 1-го НГМД и, если есть второй НГМД, то его также включает.

```
cli
mov  al, 0001100b          ; Сброс контроллера
mov  dx, 3F2h
out  dx, al

mov  al, 00111100b          ; Включение моторов
out  dx, al
sti
```

#48. Заполните таблицу данных, которые нужно передать при выполнении командной фазы форматирования дорожки, на которой расположен 155-й сектор. Форматировать на 720Кбайт та, чтобы чтение-запись выполнялись бы возможно медленнее. Фаза команды:

	01001101	(4Dh)	- команда форматирования
	00000100		- 155 сектор находится на Head 1
N	2		- код длины сектора 512
SC	9		- к-во секторов
GPL	1Bh		- длина GPL (для 1.44М - 06Ch)
D	0FEh		- Заполнитель

**#37. Написать фрагмент программы, который выполняет фазу чтения результата операции записи на 176-й сектор гибкого диска, отформатированного на 1.44М. Если запись прошла успешно, установить флаг С равным 0, иначе установить этот флаг в 1.**  
; Чтение al из порта с проверкой готовности

```
InAL macro
    local m1
    mov dx, 3F4h

m1:
    in al, dx
    and al, 11000000b
    cmp al, 11000000b
    jne m1
    inc dx
    in al, dx
endm

.....
InAL ; Чтение ST0
and al, 11000000b ; Маска [Нормальное завершение]
mov ah, al
InAL ; Чтение ST1
and al, 10110110b ;
or ax, ax
jnz Error
clc
jmp Next

Error:
    stc

Next:
    InAL ; Чтение ST2
    InAL ; Чтение C
    InAL ; Чтение H
    InAL ; Чтение R
    InAL ; Чтение N
```

**19 Во сколько раз изменится плотность хранения информации на дорожках гибкого диска, отформатированного на 1.44, если диаметр внешней дорожки - 3" , а поперечная плотность составляет 6 дорожек на мм. (+2)**

1.44

$\rho_0 = 6 \text{дор. / мм}$        $1" = 25,4 \text{ мм}$   
 $\rho_{79} / \rho_0 = L_{79} / L_0 = d_{79} / d_0 = (3" * 2,54) / (3" * 25,4 - 80 / [6 * 2]) = 76,2 / 50 = 1,524$

**#60. Написать фрагмент программы, который передвигает головки на 2 дорожки (в сторону увеличения номера) по сравнению с их текущей позицией.**

```
; Вывод al в порт с проверкой готовности
OutAL macro Command
    local m1
    mov dx, 3F4h

m1:
```

FDC

```

        in      al,dx
        and     al,11000000b
        cmp    al,10000000b
        jne    m1
        inc    dx
        mov     al,Command
        out     dx,al
endm
.....
        OutAL   11001111b      ; Относительное позиционирование головки
        OutAL   0
        OutAL   2
```

12 Напишите фрагмент программы, который реализует ожидание конца обработки прерывания контроллером диска (+2)

Имеется ввиду, что сейчас FDC хр. прерывние again:

```

        mov     dx,3F0h
        in      al,dx
        test    al,80h
        jnz     again
```

1 Напишите фрагмент программы, который определяет общее число секторов из области БИОС Boot-сектора и помещает его на регистр AX (+2)

Buffer db 512 dup (?)

```

.....
mov ax,ds
mov es,ax
mov bx,offset Buffer
```

```

mov ah,2
mov dh,0
mov ch,0
mov cl,1
mov al,1
int 13h
mov si,offset Buffer
add si,12
lodsw
```

#1. Какой процент информационной ёмкости используется непосредственно при записи полезной информации в гибком диске 3.5", отформатированной на 720Кбайт?

Количество секторов = 80•9•2  
Начало дорожки: GAP(80) + SYNC(11) + IAM(4) + GAP(50) = 146  
Размер сектора: SYNC(12) + IDAM(4) + CYL(1) + HEAD(1) + SEC(1) + NL(1) + CRC(1) + GAP(22) + SYNC(12) + DATA\_АМ(4) + 512 + CRC(1)+GAP(68)= 640  
Общий объём: (Количество секторов)•(Размер сектора)+(Количество дорожек)•(Начало дорожки)=944960 байт

Процент информационной ёмкости: (720•1024/944960)•100%=78%

#17. Сколько команд обращения процессора к контроллеру гибкого диска (IN и OUT) надо выполнить, чтобы скопировать полностью информацию с одной дискеты на другую?

Дискета 1.44: к-во дорожек 80•2=160  
Чтение: команды чтения дорожки OUT - 9  
  IN - 7  
  Итого Out 9•160=1440  
  Итого IN 7•160=960  
Перемещение головок (Seek): OUT = 3 • 80 = 240  
                                  Sense Interrupt Status OUT = 1 • 80 = 80 IN = 2 • 80 =160  
Итого для ЧТЕНИЯ:  
OUT = 1440+240+80 = 1760  
IN = 960+160 = 1120

Для записи аналогично  
 Ответ:  
 OUT = 1760•2 = 3520  
 IN = 1120•2 = 2240  
 Или 5760

Сколько команд обращения процессора к контроллеру гибкого диска (IN или OUT) надо выполнить, чтобы отформатировать диск на 1.44 М. (+2)

- 1) проверка готовности FDC +1
  - 2) установите гол. Дорожку3б +3
  - 3) командная фаза 6б +6
  - 4) фаза выполнения DMA +0
  - 5) фаза результат 7б +7
- +17  
 17 \* 80 = 1360. Всего 1360 без проверок ошибок

#50. Напишите фрагмент программы, задающий стандартные для дискеты 3.5" (1.44М) временные параметры.

```

  Успользуем макрос OutAL из задачи 60
  OutAL 3 ; Команда Specify
  OutAL 0AFh ; SRT | HUT
  OutAL 2 ; HLT = 2, ND=0
```

#47. Напишите фрагмент программы, который определяет число накопителей, подключённых к КГМД и номер активного в настоящий момент накопителя.

```

Port equ 3F0h
one_mes db 'Один накопитель $'
two_mes db 'Два накопителя $'
fisrt db 'Активен первый $'
second db 'Активен второй $'
```

```

MSG macro S
    lea dx, S
    mov ah, 9
    int 21h
endm

    mov dx, Port
    in al, dx ; SRA
    test al, 40h
    jz One
    MSG two_mes
    jmp @@1
One: MSG one_mes
@@1: mov dx, Port+1
    in al, dx
    test al, 20h
    jz F1
    MSG first
    jmp exit
F1:
    MSG second
exit:
```

#48. Напишите фрагмент программы, который реализует командную фазу чтения 44-го сектора.

```

  Успользуем макрос OutAL из задачи 60
  OutAl 66h ; 65h - запись
  OutAl 0 ;
  OutAl 1 ; cylinder
  OutAl 0 ; head
  OutAl 9 ; логическая нумерация секторов - от 0
  OutAl 2 ; размер сектора
  OutAl 18 ; к-во секторов
```

```
OutAl 1Bh ; GPL
OutAl 0FFh ; DTL

WaitInt

InAl ; st0
inal ; st1
inal ; st2
inal ; cyl
inal ; head
inal ; sect
inal ; byte in sect
```

#5. Напишите фрагмент программы установки на нулевую дорожку головок жёсткого диска с выдачей сообщения, если возникли ошибки. Временной параметр движения принять равным 3-м.

```
cli
mov dx, 1F7h
L1: in al, dx ; порт команд / байта состояния
bt ax, 7 ; проверка
jnc L1 ; готовности

mov al, 0001.0011b ; рекалибровка с временным
Rec Spd ; параметром передвижения 3
out dx, al
sti

; Ожидание выполнения
L2: in al, dx
bt ax, 7
jc L2
bt al, 0
jc Error
.....
Error:
```

#01. Заполните таблицу данных, которые нужно передать в рамках командной фазы форматирования дорожки, на которой расположен 99-й сектор. Форматировать на 1.44 Мбайт.

				01001101b						
				00000100b						
				2						
				18						
				2Ah						
				F6h						
C	2	2	2	2	2	2	2	2	2	2
H	1	1	1	1	1	1	1	1	1	1
R	1	3	5	7	9	11	13	15	17	
N	2	2	2	2	2	2	2	2	2	2
C	2	2	2	2	2	2	2	2	2	2
H	1	1	1	1	1	1	1	1	1	1
R	2	4	6	8	10	12	14	16	18	
N	2	2	2	2	2	2	2	2	2	2

{Первая строка таблицы дописана позднее.}

#05. С использованием BIOS выполнить чтение данных, расположенных начиная с 129 сектора по 155 сектор гибкого диска, отформатированного на 1.44 Мбайта. Если при чтении обнаружена ошибка - выдать сообщение.

```
CODESEG
Mov AH, 02h
Mov AL, 16
Mov DL, 00h
Mov DH, 1
Mov CH, 3
Mov CL, 3
Les BX, [Buf]
Int 13h
Jc Error
Mov AH, 02h
```



```

Mov     AL, 11
Mov     DL, 00h
Mov     DH, 0
Mov     CH, 4
Mov     CL, 1
Les     BX, [Buf1]
Int     13h
Jc      Error
. . .
Error:  Mov     AH, 09h
        Lea     DX, [Msg]
        Int     21h
```

```

DATASEG
Msg      DB      'Ошибка$'
```

**#03. Почему при кодировании номеров дорожек для позиционирования головок используется код Грея, а не простая нумерация.**

Код Грея обеспечивает более надежное кодирование информации.  
При простой нумерации, при переходе от предыдущего номера к следующему может меняться сразу несколько разрядов, а код Грея обеспечивает изменение только одного разряда. Это позволяет улучшить качество чтения при высокой скорости.

***19. На 8-й дорожке 1-й и 4-й сектора имеют одинаковые маркеры: 8-0-1-2. Какой из секторов будет считан при выполнении команды считывания сектора 8-0-1-2? (+2)***

Будет считан тот сектор, который обнаруживается первым, т.е. если головка находится между первым и четвёртым сектором, то обнаруживается первым 4-й сектор, он и будет считан; если же головка находится после четвёртого сектора, то будет обнаружен первым 1-й сектор, он и будет считан.

FAT

**#1. На жёстком диске организовано 10 логических дисков. Сколько передвижений головки надо выполнить, чтобы в наихудшем случае добраться до файла, если последний локализован в корневом каталоге?**

За 4-передвижения считывается последний диск, 5 – доступ к файлу

**#24. Дорожка жёсткого диска содержит 13 секторов, фактор чередования равен 3-м. Укажите порядок расположения секторов на дорожке.**

1	2	3	4	5	6	7	8	9	10	11	12	13
1	10	6	2	11	7	3	12	8	4	13	9	5

**#32. Скорость вращения жёсткого диска составляет 10 обр./с . Файл занимает 8 смежных секторов, расположенных на 17-и секторной дорожке с фактором чередования 3. Определить время считывания файла**

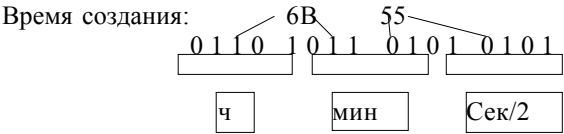
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	7		2	8		3		4		5		6				
1	7	13	2	8	14	3	9	15	4	10	16	5	11	17	6	12

⇒ необходимо сделать 2 оборота.    Время считывания файла

**14       Из корневого каталога считан описатель файла. Определить время создания файла и номер начального кластера**

49 52 5A 31 20 20 20 20 45 58 45 20 00 00 00 00 00 00 00 00 00 00 6B 55 D7 22 0F 00 0C 0A 00 00    (+2)

name



Начальный номер кластера: 000F - переставлены байты

13 часов            26мин    42 сек

#31. Напишите фрагмент программы, которая определяет число файлов, хранящихся в корневом каталоге на гибком диске (1.44М). Считать, что их меньше 32.

```
Data      db      1024 dup (?)
FileCount db      0
        mov     ah, 2      ; Чтение
        mov     dl, 0      ; Дискковод А
        mov     ch, 0      ; Цилиндр
        mov     cl, 2      ; Сектор 19 (абс) - начало корневого каталога
        mov     al, 2      ; Читать два сектора
        lea     bx, Data
        int     13h
; Анализ каталога
        mov     cx, 32
        lea     bx, Data
LL1:
        mov     al, [bx]
        or      al, al
        jz      End
        add     bx, 32
        cmp     al, 0E5h
        je     LL1
        inc     FileCount
        loop    LL1
End:
```

#16. Из корневого каталога считан описатель файла. Определить время создания файла и номер начального кластера.

```
49 52 5F 31 20 20 20 20 45 58 45 20 00 00 00 00
00 00 00 00 00 00 6B 55 D7 22 0F 00 0C 0A 00 00
.....
DirEntry  struc
    FileName db 8 dup(?)
    Ext      db 3 dup(?)
    Attr     db ?
    Reserved db 10 dup(?)
    Time     dw ?
    Date     dw ?
    Cluster  dw ?
    FileSize dd ?
DirEntry  ends
```

I	R		1					E	X	E					
49	52	5F	31	20	20	20	20	45	58	45	20	00	00	00	00
00	00	00	00	00	00	00	6B	55	D7	22	0F	00	0C	0A	00
							Time	Date		Cls	Size				

```
Time Pack 556Bh,      Date Pack 22D7h
Время      01010.101011.01011 = 556Bh      (5.6.5)
           Час   Мин   Сек = 01011•2
Время      10   : 43   : 22

Дата      0010001.0110.10111   = 22D7h   (7.4.5)
      1980 + Год   Мсц   День
Дата      1997   6     23
```

Начальный кластер 0Fh  
 Резмер 0Ah

**#17. Из корневого каталога считано 32 байта - описателя файла. Определить дату создания файла и его объем в байтах: 49 52 5F 31 20 20 20 20 41 58 45 21 00 00 00 00 00 00 00 00 DE 83 3D 16 15 00 00 87 00 00.**

Длина файла 8700h байт. Дата создания 29 января 1991г.

**#58. Какими факторами определяются количество секторов в кластере?**

Первичные факторы: Количество секторов в кластере определяется таким, чтобы, во-первых, максимально заполнить всю таблицу распределения файлов, во-вторых, адресовать через эту таблицу все сектора, отведённые под данные.

Вторичные факторы: Типом файловой системы, установленной на диске (FAT16, FAT32), и допустимым количеством секторов в кластере.

Так, например, для FAT32 размер кластера - 4К (8 секторов).

Для FAT16 размер кластера определяется как:

<(количество секторов, определённых под данные)/65536> - округление в сторону увеличения.

Допустимое количество секторов в кластере: (1К(2) , 2К(4) , 4К(8) , 8К(16) , 16К(32) , 32К(64))

**4. Напишите фрагмент программы, который определяет из области BIOS Boot-сектора число секторов на дорожке и помещает его на регистр AX. (+2)**

```
buf      db 512 dup (?)
-----
MOV     AX, DS
MOV     ES, AX
MOV     BX, Offset buf

MOV     AH, 2
MOV     DL, 1
MOV     DH, 0
MOV     CH, 0
MOV     CL, 1
MOV     AL, 1
INT     13h
MOV     BX, Offset buf
MOV     Si, 17
MOV     AX, [BX+Si] ; AX – число секторов на дорожке
```

**12. С использованием BIOS напишите фрагмент программы, который считывает BOOT RECORD логического диска, на котором хранится операционная система. (+2)**

```
buf      512 dup (?)
-----
MOV     CH, 0
MOV     CL, 0
MOV     DH, 1
MOV     AX, DS
MOV     ES, AX
MOV     BX, Offset buf
MOV     AH, 2
MOV     AL, 1
MOV     DL, 80h
INT     13h
MOV     CX, 512
MOV     Si, Offset buf
MOV     AX, 1BEh
SCASB   ; Si – начало описателя
ADD     Si, 2
MOV     AX, [Si]
```

**FAT**

---

CMP	AX, 80h	; есть OC
JNE	not_OS	
ADD	Si, 2	
MOV	DH, [Si]	
ADD	Si, 4	
MOV	CH, [Si]	
SUB	Si, 2	
MOV	AX, [Si]	
MOV	BX, AX	
SHL	BX, 4	
AND	BX, 11000000b	
OR	CL, BX	
MOV	BX, 6	
AND	BX, 11000000h	
OR	CH, BX	
MOV	DL, 80h	
MOV	AL, 1	
MOV	BX, Offset buf	; в buf – BR
MOV	AH, 2	
INT	13h	