

Лабораторная работа №1

Структура программы. Прерывания

- Научиться использовать функции прерываний 10h (BIOS) и 21h (DOS).
- Написать программу согласно заданию.
- [Ответить на контрольные вопросы.](#)

Постановка задачи

1. В центре чистого экрана вывести изречение:

Труден лишь первый шаг.	Hard is the first step.
Варрон Марк Теренций	Varro, Mark Terence
116-27 гг. до н.э.	116-27 years. BC

2. Вывести по углам экрана:

- Фамилию автора программы;
- Номер группы;
- Факультет;
- Символ '!', повторив его 5 раз.

3. Задать видеоатрибуты, определяющие цвет символов и цвет фона при выводе символов на экран. *Выбор цветового оформления зависит от вкуса.*

Теоретические сведения

Прерывание

Прерывание означает временное прекращение основного процесса вычислений для выполнения некоторых запланированных или незапланированных действий, вызванных работой устройств или программы.

Событие прерывания состоит в том, что процессор прекращает выполнять инструкции программы в нормальной последовательности, а вместо этого начинает выполнять другую программу, предназначенную для обработки этого события. После окончания обработки прерывания процессор продолжит выполнение прерванной программы.

Прерывание - специальный набор готовых процедур, постоянно доступный для использования.

Вектор прерываний – адрес процедуры обработки прерываний. Адреса размещаются в специальной области памяти доступной для всех подпрограмм. Процессоры семейства x86 и совместимые с ними могут порождать 256 прерываний. Адреса всех 256 функций обработки прерываний хранятся в специальной таблице векторов прерываний.

Прерывания могут быть программными и аппаратными.

Программные прерывания генерируются с помощью специальной команды в теле программы, то есть их порождает программист. Обычно программные

прерывания используются для «общения» вашей программы с операционной системой или возникают по запросу периферийных устройств.

Аппаратные прерывания позволяют использовать предоставляемые системой MS-DOS¹ и BIOS² большой набор подпрограмм, выполняющих различные полезные действия и оформленные как программные прерывания.

INT (interrupt) — инструкция на языке ассемблера для процессора архитектуры x86, генерирующая программное прерывание.

Синтаксис инструкции:

INT n ,

где n — номер прерывания, которое будет сгенерировано.

Как правило, номер прерывания записывается в виде шестнадцатеричного числа с суффиксом *h* (от англ. hexadecimal - шестнадцатеричный).

Общий алгоритм применения функций любого прерывания

1. Поместить номер вызываемой функции в регистр АН;
2. Подготовить, если необходимо, входные данные для функции;
3. Вызвать прерывание;
4. Сохранить или обработать выходные данные.

Обзор прерываний MS-DOS. Операционная система MS-DOS предоставляет программе набор системных вызовов, реализованных с использованием механизма программных прерываний. Эти вызовы открывают прикладной программе доступ к системной информации, к системе консольного ввода/вывода, файловой системе, к системе управления программами и памятью, позволяют организовать обращение к драйверам периферийных устройств компьютера и т. д.

Все основные функции MS-DOS вызываются с помощью прерывания INT 21h, однако MS-DOS использует и другие прерывания.

Функции прерывания INT 21h можно разделить на следующие группы:

- получение системной информации;
- символьный ввод/вывод;
- работа с файловой системой;
- управление программами;
- управление памятью;
- связь с драйверами устройств;
- прочий системный сервис.

¹ Дисковая операционная система фирмы Microsoft MS DOS (Microsoft Disk Operating System).

² Базовая система ввода – вывода (BIOS) - модуль операционной системы MS DOS.

Символьный ввод и вывод. Эти функции применяются для работы со всеми символьными устройствами, такими как консоль, принтер, последовательный порт. Они называются функциями стандартного ввода/вывода.

Основных функций стандартного символьного ввода/вывода в виде таблицы.

Код	Назначение	Описание
01h	Ввод с клавиатуры	Выполняется ввод символа с устройства стандартного ввода, а также эхо-вывод символа на стандартное устройство вывода. Пользователь может прервать работу программы при помощи комбинации клавиш <Ctrl+C> и <Ctrl+Break>
06h	Ввод с клавиатуры без ожидания	Ввод символа с устройства стандартного ввода без ожидания, а также вывод его на устройство стандартного вывода. Комбинации клавиш <Ctrl+C> и <Ctrl+Break> не проверяются
07h	Прямой ввод с клавиатуры	Ввод символа с клавиатуры. Комбинации клавиш <Ctrl+C> и <Ctrl+Break> не проверяются
08h	Ввод с клавиатуры	Аналогично функции 07h, но проверяются комбинации клавиш <Ctrl+C> и <Ctrl+Break>
02h	Отобразить символ	Отображаемый символ посылается на стандартное устройство вывода
09h	Отобразить строку	На стандартное устройство вывода символов посылается строка, в конце которой находится символ "\$"
03h	Ввод из последовательного порта	Вводится символ из последовательного порта
04h	Вывод в последовательный порт	Выводится символ на последовательный порт
05h	Вывод на принтер	Выводится символ на принтер

Пример 1 Вывод строк на экран (работа с прерыванием 21h)

```

model small
.stack 100h
.data
message1 db 'Это первая строка', '$'
message2 db 'Это вторая строка $'
; здесь показаны два различных варианта задания конца строки
.code

```

```

start:
    mov ax,@data
    mov ds,ax
    mov ah,09h
    mov dx,offset message1
    int 21h
    mov ah,09h
    mov dx,offset message2
    int 21h
    mov ax,4c00h    ; стандартный выход - ah=00h
    int 21h
end start

```

Пример 2: Вывести на экран символ '\$'

```

Mov ah, 02 ;номер функции
mov dl, '$' ;входные данные
int 21h    ;вызов прерывания

```

Функции BIOS для работы с экраном. Возможности DOS для вывода на экран символьной информации весьма ограничены: она не имеет функций для изменения цвета выводимых символов и позиционирования курсора. Кроме того, в DOS отсутствуют средства формирования графических изображений.

Все возможности видеосистемы компьютера можно реализовать с помощью видеофункций BIOS прерывания *int 10h*. Прерывание *int 10h* обеспечивает: смену видеорежима (текстовый или графический); вывод символьной и текстовой информации; смену шрифтов, настройку цветовой палитры, работу с графическим изображением. Программирование видеосистемы с помощью средств BIOS более громоздко, однако большие возможности и высокая скорость вывода обуславливают широкое использование этого метода в прикладных программах.

Прерывание BIOS 10H (обмен данными с дисплеем).

Это прерывание обеспечивает выполнение 16-и операций с дисплеем. Выбор операции производится в зависимости от значения регистра *АH*. Операции с дисплеем можно разделить на 5 групп:

1. Операции интерфейса

- *АH = 00h* установка видеорежима
- *АH = 01h* установка конфигурации курсора
- *АH = 02h* установка позиции курсора
- *АH = 03h* получение положения курсора
- *АH = 04h* чтение положения светового пера
- *АH = 05h* выбор активной страницы
- *АH = 06h* прокрутка активной страницы (экрана) вверх
- *АH = 07h* прокрутка активной страницы (экрана) вниз

2. Операции обработки символов.

- AH = 08h чтение символа и его атрибута в текущей позиции курсора
- AH = 09h запись символа и атрибута в текущую позицию курсора
- AH = 0Ah запись символа в текущую позицию курсора

3. Операции графического интерфейса.

- AH = 0Bh задание палитры цветов
- AH = 0Ch изображение точки
- AH = 0Dh чтение точки (позиция, атрибут)

4. Операция вывода в режиме телетайпа - позволяет использовать монитор, как простой терминал (вывод на экран с перемещением курсора)

AH = 0EH

5. Операция чтения в видеорежиме

AH = 15 0FH получение видеорежима

Пример 3: Позиционирование курсора

```
mov ah, 2 ;запрос на установку курсора
mov dh, 8 ;номер строки - 8
mov dl, 12 ;номер столбца - 12
mov bh, 0 ;номер страницы
int 10h
```

Пример 4: Запись символа в позицию курсора

Записывает символ ASCII кода в текущую позицию курсора. Символ принимает атрибут, установленный ранее для этой позиции. После вывода курсор следует сместить к следующей позиции функцией 02h.

```
mov AL, 'a' ;выводимый символ (или его порядковый номер в таблице.)
mov bl, 10001100b ;атрибут – ярко-красный мигающий
mov CX, 1 ;коэффициент повторения
int 10h
```

Пример 5: Очистка экрана

Для очистки экрана используется прокрутка, т.к. при этой операции появляющиеся на экране строки заполняются пробелами.

```
mov ax, 0600h |mov ah, 6 ;прокрутка вверх
                |mov al, 0 ;строки, появляющиеся внизу, заполняются
                |пробелами. 0 - все окно заполняется пробелами
mov bh, 3fh ;байт атрибута
mov cx, 0000 |mov ch, 0 ;номер строки верхнего левого угла окна
                |mov cl, 0 ;номер столбца верхнего левого угла окна

mov dx, 184FH |mov dh, 18h ;номер строки нижнего правого угла окна
                |mov dl, 4FH;номер столбца нижнего правого угла |окна
int 10h
```

Режим цветного текста

Из рис. 1 следует, что каждый символ может принимать любой из 16 возможных цветов, определяемых сочетанием младших 4-х битов. Биты 4-6 байта атрибутов задают цвет фона под данным символом. Последний бит 7, в зависимости от режима видеоадаптера, определяет либо яркость фона под данным символом (тогда фон также может принимать 16 разных цветов), либо мерцание символа (устанавливается DOS по умолчанию).

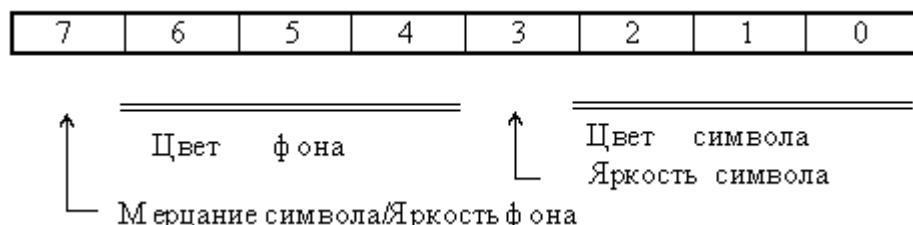


Рис.1 Структура байта атрибутов

При загрузке машины устанавливается стандартная палитра, коды цветов которой приведены в табл. 1. Рассмотрим некоторые примеры. Так, в режиме мерцания значение старшего полубайта атрибута *8h* обозначает не серый фон, а чёрный при мерцающем символе, цвет которого по-прежнему определяется младшим полубайтом; значение старшего полубайта *0Ch* – красный фон при мерцающем символе. Переключение назначения бита 7 осуществляется подфункцией *03h* функции *10h* прерывания *int 10h*.

Т а б л и ц а 1

Коды цветов стандартной палитры

Двоичное число	Шестнадца- теричное число	Цвет	Двоичное число	Шестнадца- теричное число	Цвет
0000	0h	Чёрный	1000	8h	Серый
0001	1h	Синий	1001	9h	Голубой
0010	2h	Зелёный	1010	0Ah	Салатовый
0011	3h	Бирюзовый	1011	0Bh	Светло-бирюзовый
0100	4h	Красный	1100	0Ch	Розовый
0101	5h	Фиолетовый	1101	0Dh	Светло-фиолетовый
0110	6h	Коричневый	1110	0Eh	Жёлтый
0111	7h	Белый	1111	0Fh	Ярко- белый

Контрольные вопросы

1. Что такое прерывание?
2. Назначение прерываний.
3. Формат команды прерывания?
4. Каково назначение команд INT 21H, INT 10H?
5. Прерывания операционной системы DOS. Типы прерываний DOS. Привести примеры.
6. Прерывания BIOS. Типы прерываний BIOS. Привести примеры.
7. Приведите общий алгоритм применения функций любого прерывания?
8. Как задается функция для выполнения прерывания?
9. Откорректируйте логические ошибки в коде, в котором предполагается отобразить строку на экране с помощью функции 09 прерывания INT 21H.

```
st1 db "TTTTTTTTTT",0dh,0ah, '$'
.code
mov ah,09h           ; вывод строки
mov dx, st1
int 21h
```

10. Какое прерывание очищает экран, и какие значения входных величин должны быть помещены в регистры?
11. Закончите следующий фрагмент для установки размещения курсора в строке 5, столбце 10 на видеостранице 0?

```
mov ah,
mov dh,
mov dl,
mov bh,
int 10h
```

12. Приведите структуру байта атрибутов для режима цветного текста.
13. Расшифруйте приведённый байт атрибута 10011111b, запишите его в шестнадцатеричной системе счисления?