***Задание 1.*** По выбранной элементной базе и адресам 8-разрядных регистров ввода и вывода и 2-разрядного регистра ввода-вывода представить принципиальную схему подключения портов к системной шине ISA.

Магистраль ISA была разработана специально для персональных компьют6еров типа IBM PC AT и является фактическим стандартом для всех изготовителей этих компьютеров.

Магистраль ISA относится к демультиплексированным (то есть имеющим раздельные шины адреса и данных) 16-разрядными системными магистралями среднего быстродействия. Обмен осуществляется 8- или 16-разрядными данными. Максимальный объем адресуемой памяти составляет 16Мбайт (24 адресные линии). Максимальной адресное пространство для устройств ввода-вывода – 64Кбайта (16 адресных линий), хотя практически все выпускаемые платы расширения используют только 10 адресных линий (1Кбайт). Магистраль поддерживает регенерацию динамической памяти, радиальные прерывания и прямой доступ к памяти.

Структурная схема подключения устройств показана на рисунке 1.1. (стр.2).

Основными сигналами управления в схеме являются: IOR, IOW, AEN, IRQ N.

IOR - строб чтения данных из устройств ввода-вывода.

IOW – строб записи данных в устройства ввода-вывода.

AEN (разрешение адреса) – используется в ПДП для сообщения всем платам расширения, что производится цикл ПДП.

IRQ N – сигналы запроса радиальных прерываний.

В магистрали ISA для каждого подключаемого устройства забронированы конкретные адреса, наше устройство не является стандартным, поэтому для его адресации используем резервные адреса: 360h – регистр ввода, 361h – регистр вывода, 362h – регистр ввода-вывода.

Для адресации конкретного устройства используем дешифратор адреса (ДшА). К нему подведены старшие биты адреса (SA2-SA9). И управляющий сигнал AEN. Младшие биты адреса (SA0-SA1) включены в дешифратор управляющих сигналов (ДшУС). Для дешифрации управляющих сигналов используются сигналы IOR и IOW.

На выходе ДшУС образуются сигналы чтение регистра ввода (ЧтРВв), запись в регистр вывода (ЗпРВыв), чтение регистра ввода-вывода (ЧтРВ/В) и запись в регистр ввода-вывода (ЗпРВ/В). В таблице 1.1. показан принцип образования сигналов управления.

Таблица 1.1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Управляющие сигналы/ Сигналы ISA** | **ЧтРВв** | **ЗпРв** | **ЧтРВ/В** | **ЗпРВ/В** |
| **SA0** | **0** | **1** | **0** | **0** |
| **SA1** | **0** | **0** | **1** | **1** |
| **IOR** | **0** | **1** | **0** | **1** |
| **IOW** | **1** | **0** | **1** | **0** |

В приемопередатчик (ПП) поступают восьмиразрядные данные (SD0-SD7) и сигнал разрешения чтения (IOR).

В регистры ввода (РгВв) и вывода (РгВыв) восьмиразрядные, управляются сигналами ЧтРВв и ЗпРВыв соответственно. Регистр ввода/вывода (РВ/В) двухразрядный, управляется сигналами ЗпРВ/В и ЧтРВ/В.

Кроме того, регистры ввода и ввода/вывода управляются сигналом запись в регистр ввода из периферийного устройства.

Принципиальная схема устройства показана на рисунке 1.2.(стр.4).

Дешифратор адреса реализован микросхемами К1533ЛН1, К1533ЛА2 и К555ЛЛ1. В дешифраторе используется простая логика, поэтому описывать подробно принцип образования сигналов не имеет смысла.

Дешифратор управляющих сигналов реализован на микросхеме К1533ИД3, которая представляет собой дешифратор - демультиплексор с 4 на 16.

Согласно таблице истинности этой микросхемы и таблице 1.1. были определены выводы, с которых снимаем управляющие сигналы.

Приемопередатчик реализован на микросхеме К1533АП6, представляющей из себя двунаправленный восьмиразрядный шинный усилитель с тремя состояниями выхода.

Регистры ввода и вывода собраны на микросхемах К1533ИР22, а регистр ввода/вывода – на ИМС К1533ИР34.

***Задание 2.*** Клавиатуру, содержащую 16х6 клавиш, подключить к интерфейсу Multibus с помощью соответствующих портов, необходимых для сканирования клавиш контактного типа. Представить подробную структурную схему с указанием всех управляющих сигналов.

Схема подключения устройства показана на рисунке 2.1 (стр.6). Устройство работает по прерыванию, т.е. если не нажата ни одна клавиша, то устройство в работу не включается.

Несколько узлов структурной схемы уже были использованы и описаны в задании 1. В данном случае используется интерфейс MULTIBUS, который имеет следующие сигналы управления: IORC –чтение порта; IOWC- запись в порт. Обмен осуществляется в режиме квитирования с формированием исполнителем ответного сигнала XACK.

Управляющими сигналами для дешифратора адреса (ДшА) служат адреса А2-А9, формирующие сигнал SEL. Младшие биты адреса (A0-A1) включены в дешифратор управляющих сигналов (ДшУС). Для дешифрации управляющих сигналов используются сигналы IORС и IOWС.

На выходе ДшУС образуются сигналы ВклБуф, ВклР12, ЗпР1, ЗпР2, ВклАП5, которые будут описаны ниже.

На приемопередатчик (ПП) поступают восемь разрядов данных (Д0-Д7), управляется он сигналами SEL и IORC.

В структурной схеме ФСК – формирователь сигнала квитирования.

Регистры RG1 и RG2 – регистры сканирования.

В исходном состоянии регистры сканирования отключены от клавиатуры сигналом ВклР12. К клавиатуре подключены буферные элементы (ИМС К555ЛП10), через которые выходы К0-К15 клавиатуры включены на землю. Микросхема КР1533АП5 отключена от шины данных ВД0-ВД5. Т.к. ни одна клавиша не нажата, то на выходе логического элемента (ИМС К1533ЛА2) уровень логического нуля.

Если на клавиатуре нажата клавиша, то через диод выход логического элемента становится подключенным на землю, на выходе элемента устанавливается высокий уровень, т.е. формируется сигнал запроса прерывания INT N. После этого запускается управляющая программа по вектору прерывания.

От шины К0-К15 отключаются буферные элементы сигналом ВклБуф, и подключаются регистры сканирования. Программно реализуется режим «бегущего нуля» на регистрах сканирования и дальше через КР1533АП5 смотрим состояние шины КР0-КР5. Таким образом вычисляется код нажатой клавиши.

***Задание 3.*** Разработать принципиальную схему адаптера принтера, содержащего 8-разрядный регистр данных (записи) с адресом 378h, 5-разрядный регистр управления (записи) с адресом 37Ah и 5-разрядный порт состояния (чтения с линий Д3-Д7), имеющий адрес 379h. Подключить порты к системной шине расширения ISA.

Принципиальная схема показана на рис. 3.1.(стр.8).

Дешифратор адреса реализован по тому же принципу, что и в задании 1, но регистры имеют адреса: 378h- регистр данных; 379h – регистр состояния; 37Аh- регистр управления.

Дешифратор управляющих сигналов выполнен так же, но с учетом таблицы 3.1.

Таблица 3.1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Управляющие сигналы/ сигналы ISA** | ЗпРД | **ЗпРУ** | **ЧтРС** |
| **SA0** | **0** | **0** | **1** |
| **SA1** | **0** | **1** | **0** |
| **IOR** | **1** | **1** | **0** |
| **IOW** | **0** | **0** | **1** |

Восьмиразрядный приемопередатчик выполнен на микросхеме К1533АП6.

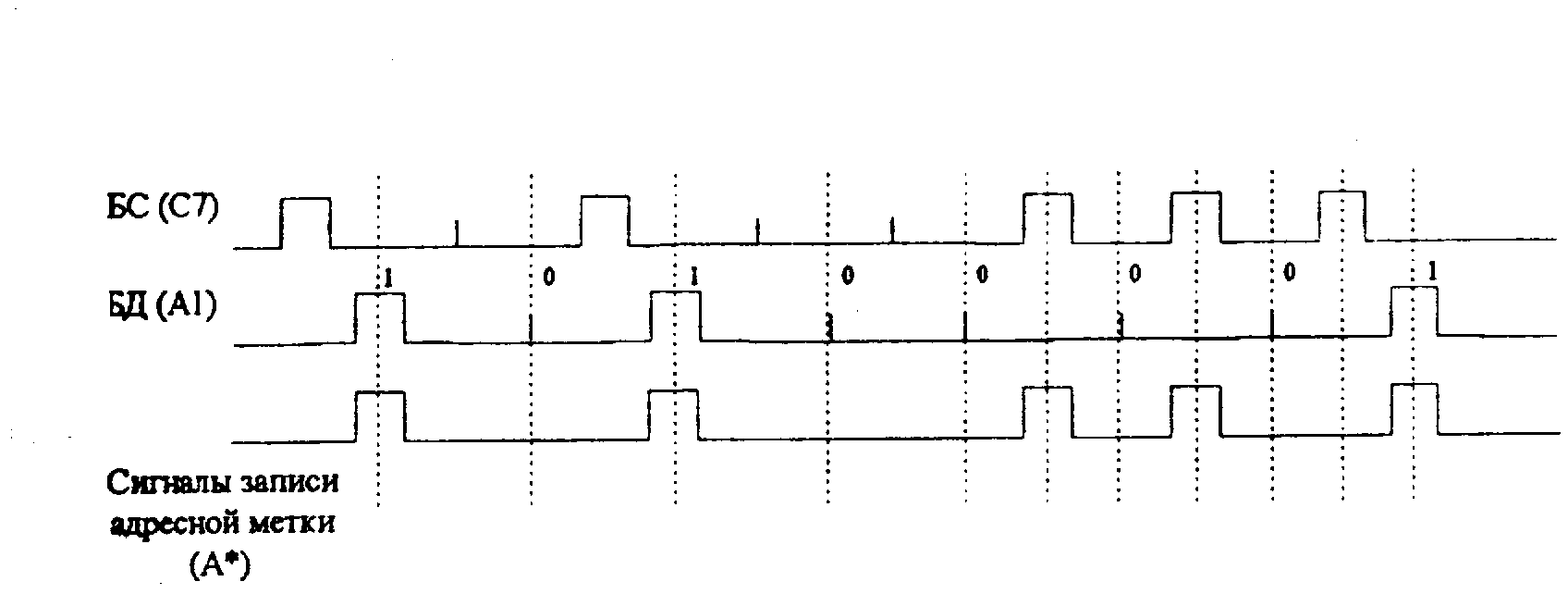
Регистры данных и управления выполнены на микросхемах К1533ИР22, в качестве регистра состояний используются буферные элементы – ИМС К555ЛП10.

В исходном состоянии на выходе дешифратора адреса постоянно присутствует уровень логической единицы, вследствие чего ДшУС и ПП не участвуют в работе.

Перед началом цикла передачи данных компьютер должен убедиться, что сняты сигналы BUSY и ACK. Для этого мы должны считать состояние этих сигналов из регистра состояния (РС). Это сделаем с помощью команды IN, где укажем адрес РС –379h. ДшУС выдаст сигнал чтения РС (ЧтРС), и сигналы состояния принтера по шине ВД0-ВД4 отобразятся на шине данных SD0-SD4 интерфейса ISA. После этого выставляем данные путем выполнения команды OUT с указанием адреса регистра данных (378h). В результате чего ДшУС сформирует сигнал записи в регистр данных (ЗпРД). Затем формируется строб. В команде OUT указываем адрес регистра управления (РУ) 37Аh, после чего формируется сигнал записи в регистр управления (ЗпРУ) ДшУСом.

После этого строб снимается. При получении строба принтер формирует сигнал BUSY, а после окончания обработки данных выставляет сигнал ACK, снимает BUSY, снимает ACK, затем может начаться новый цикл.

***Задание 4.*** Показать временную диаграмму формирования адресной метки, у которой биты синхронизации равны C7hЮ, а биты данных – А1, метод кодирования MFM.



Метод MFM является методом с удвоенной плотностью записи за счет исключения части битов синхронизации и увеличения тактовой частоты записи.

**Рис. 4.1**. Запись адресной метки методом MFM

Если бит данных равен единице, то стоящий перед ним бит синхронизации не записывается. Если бит данных равен нулю, но предыдущий бит данных равен единице, то бит синхронизации также не записывается.

Рассмотрим последовательность битов, изображенных на рисунке 4.1. начиная со старшего. Перед первым битом данных стоит бит синхронизации, поэтому в сигнале записи будет присутствовать только бит данных. Перед вторым битом данных нет бита синхронизации, но предыдущий бит данных был равен единице, поэтому бит синхронизации тоже не пишется, бит данных равен нулю, таким образом, сигнал записи равен нулю.

Третий бит записывается аналогично первому, четвёртый – аналогично второму. Пятый бит равен нулю, предыдущий бит данных также равен нулю, поэтому бит синхронизации и бит данных должны записываться в сигнал записи, но они равны нулям, т.е. записывается нуль.

Шестой бит равен нулю, предыдущий бит данных тоже равен нулю, т.е. записываются и бит данных и бит синхронизации. Бит синхронизации равен единице, бит данных равен нулю т.е. сигнал записи равен единице. Седьмой бит записывается аналогично шестому, а восьмой – первому.

**Задание 5.** Разработать подробную структурную схему сопряжения ПК с шиной ISA (приемник) и шиной Multibus (передатчик) с изображением всех управляющих сигналов. Обмен информацией осуществляется в режиме прерывания.

Структурная схема показана на рисунке 5.1. (стр.11).

В предыдущих заданиях были разработаны схемы подключения и к шине Isa и к шине Multibus, сигналы управления и принцип их формирования, поэтому описывать это снова нет смысла. Скажем только, что адреса А0-А9 интерфейса Multibus и SA0-SA9 интерфейса ISA для ДшА берем из области резервных адресов (360…36F).

Для передачи данных из ПК с интерфейсом Multibus используем регистр вывода данных (РВывД). Второй ПК с интерфейсом ISA включится в работу по прерываниюIRQ N в момент передачи данных по сигналу ЗпРВвД.

С помощью управляющей программы он считает данные из регистра ввода (РВвД) по сигналу чтения из регистра ввода данных (ЧтВвД), который в свою очередь сбросит сигнал IRQ N и подготовит ПК с шиной ISA для дальнейшего приема данных.

***Задание 6.*** Изучить принцип работы печатающего устройства. Написать программу на ассемблере, осуществляющую вывод на печать с помощью портов адаптера и опросом флага готовности принтера.

.model tiny ;модель памяти, исп. для СОМ

.code ;начало сегмента кода

org 100h ;нач. знач. счетчика - 100Н

**;Начало основной программы---------------------------------------------------**

start: call clear ;ярко-белые симв. на синем

;

mov cx,len1 ;

mov dh,10 ;

mov dl,27 ;

mov bp,offset mes1 ;

call symv ;вывод на экран строки симв.

;

call nklav ;идентиф. нажатой клавиши

call clsym ;стирание из буфера символа

;

call clear ;ярко-белые симв. на синем

;

mov cx,len2 ;

mov dh,10 ;

mov dl,23 ;

mov bp,offset mes2 ;

call symv ;вывод на экран строки симв.

;

mov cx,len3 ;

mov dh,12 ;

mov dl,23 ;

mov bp,offset mes3 ;

call symv ;вывод на экран строки симв.

;

nrv2: call nklav ;идентиф. нажатой клавиши

;

cmp ah,02h ;

je nrv3 ;

cmp ah,03h ;

je nrv10 ;

call clsym ;

jmp nrv2 ;

nrv10: jmp nrv6 ;

;

nrv3: call clear ;ярко-белые симв. на синем

call clsym ;стирание из буфера символа

;

mov cx,len4 ;

mov dh,0 ;

mov dl,10 ;

mov bp,offset mes4 ;

call symv ;вывод на экран строки симв.

;

mov ah,06h ;функция задания окна

mov al,0 ;режим создания

mov bh,2Fh ;атрибут символов - з/ярко-б

mov ch,1 ;верхняя Y-координата

mov cl,0 ;левая Х-координата

mov dh,24 ;нижняя Y-координата

mov dl,79 ;правая Х-координата

int 10h ;прерывание BIOS

;

mov ah,02 ;

mov bh,0 ;

mov dh,1 ;

mov dl,0 ;

int 10h ;

**;Запись символов в буфер(buf1)**

mov ah,3Fh ;

mov bx,0 ;

mov cx,100 ;

mov dx,offset buf1 ;

int 21h ;

;

mov lensym,ax ;

inc lensym ;

call nrv11 ;Состояние ACK,BUSY

mov si,0 ;

dlsh: dec lensym ;

mov al,offset buf1[si] ;

mov dx,378h ;

out dx,al ;Выставление данных

mov dx,1000 ;

call delay

mov dx,37Ah

mov al,00001001b

out dx,al ;Выставляем строб

mov dx,1000

call delay

mov dx,37Ah

mov al,00001000b

out dx,al ;Снимаем строб

cmp lensym,0

jz kon

inc si

call nrv11

jmp dlsh

**;Передача в принтер символа конца строки**

kon: call nrv11

mov al,10

mov dx,1000

call delay

mov dx,378h

out dx,al

mov dx,1000

call delay

mov dx,37Ah

mov al,00001000b

out dx,al

mov dx,1000

call delay

mov dx,37Ah

mov al,00001001b

out dx,al

mov dx,10000

call delay

mov al,13

mov dx,378h

out dx,al

mov dx,10000

call delay

mov dx,37Ah

mov al,00001000b

out dx,al

mov dx,10000

call delay

mov dx,37Ah

mov al,00001001b

out dx,al

jmp nrv6

;

nrv6: mov ax,0003h

int 10h

;

call clsym

ret

;

;**ПП Экран: ярко-белые символы на синем фоне.---------------------------------**

clear: mov ah,06h ;функция задания окна

mov al,0 ;режим создания

mov bh,1Fh ;атрибут символов - c/ярко-б

mov ch,0 ;верхняя Y-координата

mov cl,0 ;левая Х-координата

mov dh,24 ;нижняя Y-координата

mov dl,79 ;правая Х-координата

int 10h ;прерывание BIOS

ret

**;ПП. Вывод на экран строки символов.-----------------------------------------**

symv: mov ah,13h

mov al,0

mov bh,0

mov bl,1Fh

push ds

pop es

int 10h

ret

**;ПП. Идентификация нажатой клавиши.------------------------------------------**

nklav: lahf

or ah,01000000b

sahf

nrv1: mov ah,11h

int 16h

jz nrv1

ret

**;ПП. Стирание из буфера символа.---------------------------------------------**

clsym: mov ax,0C00h

int 21h

ret

**;ПП. Задержка----------------------------------------------------------------**

delay: mov ah,86h

mov cx,0

int 15h

ret

**;ПП. Проверка регистра состояния---------------------------------------------**

nrv11: mov cx,1000

nrv112: push cx

mov dx,379h

in al,dx

mov ack,al

mov busy,al

and ack,40h

cmp ack,40h

jne nrv111

and busy,80h

cmp busy,80h

jne nrv111

nrv111: pop cx

loop nrv112

ret

**;ДАННЫЕ.---------------------------------------------------------------------**

mes1 db 'ПРОГРАММА - "PRINT"'

len1=$-mes1

mes2 db '1. Вывод на печать данных из буфера.'

len2=$-mes2

mes3 db '2. Выход из программы.'

len3=$-mes3

mes4 db 'Наберите выводимый на печать текст с помощью клавиатуры.'

len4=$-mes4

buf1 db 100 dup('\*')

lensym dw 0

ack db ?

busy db ?

;----------------------------------------------------------------------------

end start

**Задание 7.** Исследовать режим тестирования СОМ. Написать программу на ассемблере.

.model tiny ;модель памяти, исп. для СОМ

.code ;начало сегмента кода

org 100h ;нач. знач. счетчика - 100Н

**;Начало основной программы---------------------------------------------------**

start: call clear ;ярко-белые симв. на синем

;

mov cx,len1 ;

mov dh,10 ;

mov dl,27 ;

mov bp,offset mes1 ;

call symv ;вывод на экран строки симв.

;

call nklav ;идентиф. нажатой клавиши

call clsym ;стирание из буфера символа

;

call clear ;ярко-белые симв. на синем

;

mov cx,len2 ;

mov dh,10 ;

mov dl,23 ;

mov bp,offset mes2 ;

call symv ;вывод на экран строки симв.

;

mov cx,len3 ;

mov dh,12 ;

mov dl,23 ;

mov bp,offset mes3 ;

call symv ;вывод на экран строки симв.

;

nrv2: call nklav ;идентиф. нажатой клавиши

;

cmp ah,02h ;

je nrv3 ;

cmp ah,03h ;

je nrv10 ;

call clsym ;

jmp nrv2 ;

nrv10: jmp nrv6 ;

;

nrv3: call clear ;ярко-белые симв. на синем

call clsym ;стирание из буфера символа

;

mov cx,len4 ;

mov dh,0 ;

mov dl,10 ;

mov bp,offset mes4 ;

call symv ;вывод на экран строки симв.

;

mov ah,06h ;функция задания окна

mov al,0 ;режим создания

mov bh,2Fh ;атрибут символов - з/ярко-б

mov ch,1 ;верхняя Y-координата

mov cl,0 ;левая Х-координата

mov dh,24 ;нижняя Y-координата

mov dl,79 ;правая Х-координата

int 10h ;прерывание BIOS

;

mov ah,02

mov bh,0

mov dh,1

mov dl,0

int 10h

;

mov ah,3Fh

mov bx,0

mov cx,100

mov dx,offset buf1

int 21h

**;Инициалицация СОМ1**

initcom1 proc

mov ax,40h

mov es,ax

mov dx,es:[0]

add dx,3

mov al,10000000b

out dx,al

dec dx

dec dx

mov al,0

out dx,al

dec dx

mov al,0C0h

out dx,al

ret

initcom1 endp

**;Инициализация регистра контроля линии**

initrg proc

add dx,3

mov al,00000000b

or al,00000011b

or al,00000000b

or al,00000000b

out dx,al

inc dx

mov al,10h

out dx,al

sub dx,3

mov al,0

out dx,al

ret

initrg endp

**;Готовность СОМ1**

check proc

try: mov dx,es:[0]

add dx,5

in al,dx

test al,1Eh

jz m0

lea dx,error

mov cx,len5

mov dh,10

mov dl,23

mov bp,offset mes5

call symv ;вывод на экран строки симв.

jmp exit

m03: test al,01h

jnz m4

test al,20h

jz try

ret

check endp

**;Работа с портом**

work proc

m4: mov ah,0

int 16h

push ax

push ax

mov dx,es:[0]

add dx,5

mov cx,10

m1: in al,dx

test al,20h

jz m2

loop m1

m2: sub dx,5

pop ax

out dx,al

add dx,5

m3: in al,dx

test al,1

jz m3

sub dx,5

in al,dx

mov ah,0Eh

int 10h

pop ax

cmp al,1Bh

jne m4

ret

work endp

**;Выход из программы**

exit: mov ax,4C00h

int 21h

nrv6: call clsym

ret

**;ПП Экран: ярко-белые символы на синем фоне.---------------------------------**

clear: mov ah,06h ;функция задания окна

mov al,0 ;режим создания

mov bh,1Fh ;атрибут символов - c/ярко-б

mov ch,0 ;верхняя Y-координата

mov cl,0 ;левая Х-координата

mov dh,24 ;нижняя Y-координата

mov dl,79 ;правая Х-координата

int 10h ;прерывание BIOS

ret

**;ПП. Вывод на экран строки символов.-----------------------------------------**

symv: mov ah,13h

mov al,0

mov bh,0

mov bl,1Fh

push ds

pop es

int 10h

ret

**;ПП. Идентификация нажатой клавиши.------------------------------------------**

nklav: lahf

or ah,01000000b

sahf

nrv1: mov ah,11h

int 16h

jz nrv1

ret

**;ПП. Стирание из буфера символа.---------------------------------------------**

clsym: mov ax,0C00h

int 21h

ret

**;ПП. Задержка----------------------------------------------------------------**

delay: mov ah,86h

mov cx,0

int 15h

ret

;

**;ДАННЫЕ.---------------------------------------------------------------------**

mes1 db 'ПРОГРАММА - "COM"'

len1=$-mes1

mes2 db '1. Передача данных в СОМ-порт.'

len2=$-mes2

mes3 db '2. Выход из программы.'

len3=$-mes3

mes4 db 'Наберите выводимый в СОМ-порт текст с помощью клавиатуры.'

len4=$-mes4

buf1 db 100 dup('\*')

mes5 db 'Ошибка порта СОМ 1',7,0Dh,0Ah,S

len5=$-mes5

;----------------------------------------------------------------------------

end start