

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МО ЭВМ**

**ОТЧЕТ**  
**по лабораторной работе №1**  
**по дисциплине «Организация ЭВМ и систем»**  
**Тема: «Трансляция, отладка и выполнение программ на языке**  
**Ассемблера»**

Студентка гр. 1384

Усачева Д.В.

Преподаватель

Кирияничков В.А.

Санкт-Петербург 2022

## **Цель работы.**

Изучить процесс трансляции, отладки и выполнения программ на языке Ассемблер. Ознакомиться со структурой и реализацией каждого сегмента программ.

## **Задание.**

Лабораторная работа 1 использует 2 готовых программы на ассемблере: hello1 – составлена с использованием сокращенного описания сегментов и hello2 – составлена с описанием сегментов и выводом строки, оформленным как процедура.

## **Часть 1**

1. Просмотреть программу hello1.asm, которая формирует и выводит на экран приветствие пользователя с помощью функции OCMSDOS, вызываемой через прерывание с номером 21h (команда Int 21h). Выполняемые функцией действия и задаваемые ей параметры – следующие: - обеспечивается вывод на экран строки символов, заканчивающейся знаком "\$"; - требуется задание в регистре ah номера функции, равного 09h, а в регистре dx

- смещения адреса выводимой строки; - используется регистр и не сохраняется его содержимое.

2. Разобраться в структуре и реализации каждого сегмента программы. Непонятные фрагменты прояснить у преподавателя. Строку-приветствие преобразовать в соответствии со своими личными данными.

3. Загрузить файл hello1.asm из каталога Задания в каталог Masm.

4. Протранслировать программу с помощью строки > masm hello1.asm с созданием объектного файла и файла диагностических сообщений (файла листинга). Объяснить и исправить синтаксические ошибки, если они будут обнаружены транслятором. Повторить трансляцию программы до получения объектного модуля.

5. Скомпоновать загрузочный модуль с помощью строки > link hello1.obj с созданием карты памяти и исполняемого файла hello1.exe.

6. Выполнить программу в автоматическом режиме путем набора строки `> hello1.exe` убедиться в корректности ее работы и зафиксировать результат выполнения в протоколе.

7. Запустить выполнение программы под управлением отладчика с помощью команды `> afd hello1.exe`. Записать начальное содержимое сегментных регистров CS, DS, ES и SS. Результаты прогона программы под управлением отладчика должны быть представлены в таблицы.

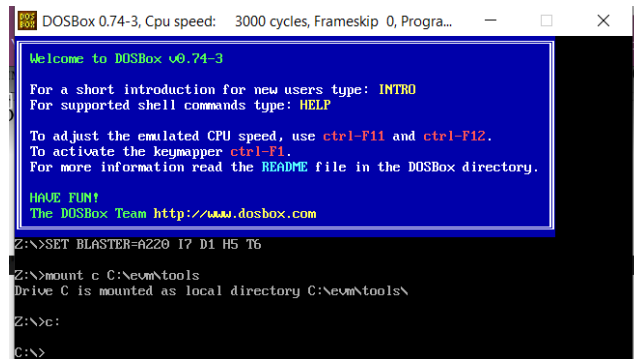
### **Часть 2.**

Выполнить пункты 1 - 7 части 1 настоящего задания применительно к программе `hello2.asm`.

## Ход работы.

### Часть 1.

Перед выполнением работы в эмуляторе DOSBox был смонтирован каталог с папкой MASM, как диск с с помощью команды «mounted c:\evm\tools» и осуществлен переход с помощью команды «с:»

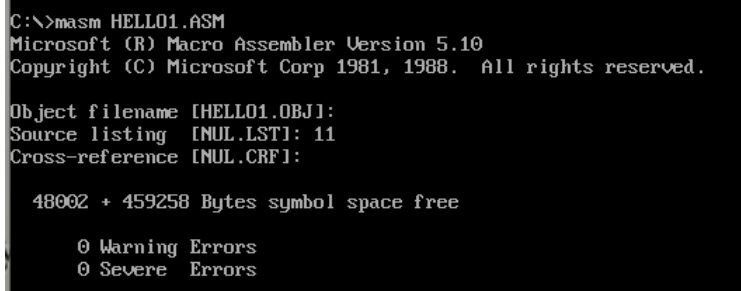


```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...
Welcome to DOSBox v0.74-3
For a short introduction for new users type: INTRO
For supported shell commands type: HELP
To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.
HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6
Z:\>mount c C:\evm\tools
Drive C is mounted as local directory C:\evm\tools\
Z:\>c:
C:\>
```

1. Для начала была просмотрена программа hello1.asm и изучена ее структура. Приветствие изменено на ‘Student Usacheva Daria of 1384 group is greeting you’.

2. Затем транслируется программа, в ходе чего создается объектный файл hello1.obj и файл диагностических сообщений list1.LST.



```
C:\>masm HELLO1.ASM
Microsoft (R) Macro Assembler Version 5.10
Copyright (C) Microsoft Corp 1981, 1988. All rights reserved.

Object filename [HELLO1.OBJ]:
Source listing [NUL.LST]: 11
Cross-reference [NUL.CRF]:

48002 + 459258 Bytes symbol space free

0 Warning Errors
0 Severe Errors
```

3. Линковка файла и создание файла с расширением .exe

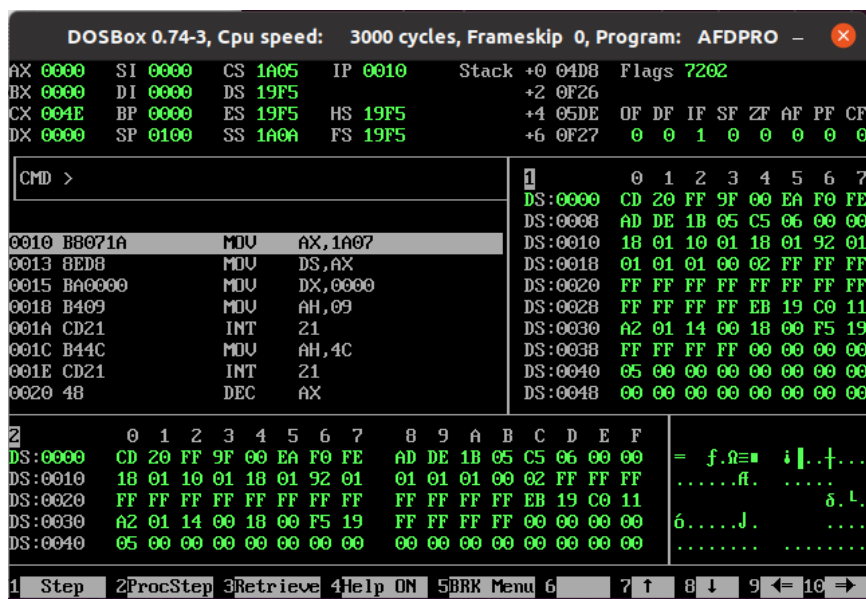
```
C:\>LINK HELLO1.OBJ

Microsoft (R) Overlay Linker Version 3.64
Copyright (C) Microsoft Corp 1983-1988. All rights reserved.

Run File [HELLO1.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:

C:\>HELLO1.EXE
Student Usacheva Daria of 1384 group 1384 is greeting you.
```

4. Запуск программы hello1.exe под отладчиком командой afdpro  
hello1.exe



Начальное содержимое сегментных регистров: (CS) = 1A05, (DS) = 19F5, (ES) = 19F5, (SS) = 1A0B.

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0010	MOV AX, 1A07	B8071A	(AX) = 0000 (IP) = 0010	(AX) = 1A07 (IP) = 0013
0013	MOV DS, AX	8ED8	(DS) = 19F5 (IP) = 0013	(DS) = 1A07 (IP) = 0015
0015	MOV DX, 0000	BA0000	(DX) = 0000 (IP) = 0015	(DX) = 0000 (IP) = 0018
0018	MOV AH, 09	B409	(AX) = 1A07 (IP) = 0018	(AX) = 0907 (IP) = 001A
001A	INT 21	CD21	(IP) = 001A	(IP) = 001C
001C	MOV AH, 4C	B44C	(AX) = 0907 (IP) = 001C	(AX) = 4C07 (IP) = 001E
001E	INT 21	CD21	(AX) = 4C07 (DS) = 1A07 (IP) = 001E	(AX) = 0000 (DS) = 19F5 (IP) = 0010

## Часть 2

1. Была протранслирована программа командой `masm hello2.asm`. В ходе трансляции создается объектный файл `hello2.obj` и файл диагностических сообщений `22.lst`. трансляция произошла без предупреждений и ошибок.

```
C:\>masm HELLO2.ASM
Microsoft (R) Macro Assembler Version 5.10
Copyright (C) Microsoft Corp 1981, 1988. All rights reserved.

Object filename [HELLO2.OBJ]:
Source listing [NUL.LST]: ls2
Cross-reference [NUL.CRF]:

47994 + 459263 Bytes symbol space free

0 Warning Errors
0 Severe Errors
```

1. Запуск hello2.exe в автоматическом режиме. Убеждаемся в корректной работе программы.

```
C:\>LINK HELLO2.OBJ

Microsoft (R) Overlay Linker Version 3.64
Copyright (C) Microsoft Corp 1983-1988. All rights reserved.

Run File [HELLO2.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:

C:\>HELLO2.EXE
Hello Worlds!
Student from 1384 - Usacheva Daria
```

2. Запуск программы hello1.exe под отладчиком командой afdpro hello2.exe

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: AFDPRO

AX 0000 SI 0000 CS 1A0B IP 0005 Stack +0 0000 Flags 7202  
BX 0000 DI 0000 DS 19F5 +2 0000  
CX 007B BP 0000 ES 19F5 HS 19F5 +4 0000 OF DF IF SF ZF AF PF CF  
DX 0000 SP 0018 SS 1A05 FS 19F5 +6 0000 0 0 1 0 0 0 0 0

CMD >

0005 1E PUSH DS  
0006 2BC0 SUB AX,AX  
0008 50 PUSH AX  
0009 BB071A MOV AX,1A07  
000C 8ED8 MOV DS,AX  
000E BA0000 MOV DX,0000  
0011 EBECFF CALL 0000  
0014 BA1000 MOV DX,0010

DS:0000 CD 20 FF 9F 00 EA F0 FE  
DS:0008 AD DE 1B 05 C5 06 00 00  
DS:0010 18 01 10 01 18 01 92 01  
DS:0018 01 01 01 00 02 FF FF FF  
DS:0020 FF FF FF FF FF FF FF FF  
DS:0028 FF FF FF FF EB 19 C0 11  
DS:0030 A2 01 14 00 18 00 F5 19  
DS:0038 FF FF FF FF 00 00 00 00  
DS:0040 05 00 00 00 00 00 00 00  
DS:0048 00 00 00 00 00 00 00 00

2 0 1 2 3 4 5 6 7 8 9 A B C D E F  
DS:0000 CD 20 FF 9F 00 EA F0 FE AD DE 1B 05 C5 06 00 00 = f.0= i.+. ...  
DS:0010 18 01 10 01 18 01 92 01 01 01 01 00 02 FF FF FF .....f. ....  
DS:0020 FF FF FF FF FF FF FF FF FF FF FF FF EB 19 C0 11 .....δ.L.  
DS:0030 A2 01 14 00 18 00 F5 19 FF FF FF FF 00 00 00 00 6.....J. ....  
DS:0040 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... .....

1 Step 2ProcStep 3Retrieve 4Help ON 5BRK Menu 6 7↑ 8↓ 9← 10→

Начальное содержимое сегментных регистров: (CS) = 1A0B, (DS) = 19F5, (ES) = 19F5, (SS) = 1A05.

Адрес команды	Символический код	16-ричный код команды	Содержимое ячеек и регистров памяти	
			До выполнения	После выполнения
0005	PUSH DS	1E	(DS) = 19F5 (SP) = 0018 (IP) = 0005 Stack: +0 0000 +2 0000 +4 0000 +6 0000	(DS) = 19F5 (SP) = 0016 (IP) = 0006 Stack: +0 19F5 +2 0000 +4 0000 +6 0000



0006	SUB AX, AX	2BC0	(AX) = 0000 (IP) = 0006	(AX) = 0000 (IP) = 0008
0008	PUSH AX	50	(AX) = 0000 (SP) = 0016 (IP) = 0008 Stack: +0 19F5 +2 0000 +4 0000 +6 0000	(AX) = 0000 (SP) = 0014 (IP) = 0009 Stack: +0 0000 +2 19F5 +4 0000 +6 0000
0009	MOV AX, 1A07	B8071A	(AX) = 0000 (IP) = 0009	(AX) = 1A07 (IP) = 000C
000C	MOV DS, AX	8ED8	(DS) = 19F5 (IP) = 000C	(DS) = 1A07 (IP) = 000E
000E	MOV DX, 0000	BA0000	(DX) = 0000 (IP) = 000E	(DX) = 0000 (IP) = 0011
0011	CALL 0000	E8ECFF	(SP) = 0014 (IP) = 0011 Stack: +0 0000 +2 19F5 +4 0000 +6 0000	(SP) = 0012 (IP) = 0000 Stack: +0 0014 +2 0000 +4 19F5 +6 0000
0000	MOV AH, 09	B409	(AX) = 1A07 (IP) = 0000	(AX) = 0907 (IP) = 0002
0002	INT 21	CD21	(IP) = 0002	(IP) = 0004
0004	RET	C3	(SP) = 0012 (IP) = 0004 Stack: +0 0014 +2 0000 +4 19F5 +6 0000	(SP) = 0014 (IP) = 0014 Stack: +0 0000 +2 19F5 +4 0000 +6 0000
0014	MOV DX, 0010	BA1000	(DX) = 0000	(DX) = 0010

			(IP) = 0014	(IP) = 0017
0017	CALL 0000	E8E6FF	(SP) = 0014 (IP) = 0017 Stack: +0 0000 +2 19F5 +4 0000 +6 0000	(SP) = 0012 (IP) = 0000 Stack: +0 001A +2 0000 +4 19F5 +6 0000
0000	MOV AH, 09	B409	(AX) = 0907 (IP) = 0000	(AX) = 0907 (IP) = 0002
0002	INT 21	CD21	(IP) = 0002	(IP) = 0004
0004	RET	C3	(SP) = 0012 (IP) = 0004 Stack: +0 001A +2 0000 +4 19F5 +6 0000	(SP) = 0014 (IP) = 001A Stack: +0 0000 +2 19F5 +4 0000 +6 0000
001A	RET Far	CB	(SP) = 0014 (CS) = 1A0B (DS) = 1A07 (IP) = 001A Stack: +0 0000 +2 19F5 +4 0000 +6 0000	(SP) = 0018 (CS) = 19F5 (DS) = 1A07 (IP) = 0000 Stack: +0 0000 +2 0000 +4 0000 +6 0000
0000	INT 20	CD20	(IP) = 0000	(IP) = 0005

### **Выводы.**

В ходе данной работы была освоена трансляция, изучено, как происходит выполнение и отладка программ на языке Ассемблер, а также разобрана структура и реализация приведенных в работе программ.

# ПРИЛОЖЕНИЕ А

## ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: hello1.asm

```
; HELLO1.ASM - упрощенная версия учебной программы лаб.раб. N1
;               по дисциплине "Архитектура компьютера"
; *****
; Назначение: Программа формирует и выводит на экран приветствие
;               пользователя с помощью функции ДОС "Вывод строки"
;               (номер 09 прерывание 21h), которая:
;               - обеспечивает вывод на экран строки символов,
;               заканчивающейся знаком "$";
;               - требует задания в регистре ah номера функции=09h,
;               а в регистре dx - смещения адреса выводимой
;               строки;
;               - использует регистр ax и не сохраняет его
;               содержимое.
; *****

DOSSEG                      ; Задание сегментов под ДОС
.MODEL SMALL                ; Модель памяти-SMALL (Малая)
.STACK 100h                 ; Отвести под Стек 256 байт
.DATA                       ; Начало сегмента данных
Greeting LABEL BYTE        ; Текст приветствия
DB 'Student Usacheva Daria of 1384 group 1384 is greeting you.',13,10,'$'
.CODE                       ; Начало сегмента кода
mov ax, @data               ; Загрузка в DS адреса начала
mov ds, ax                  ; сегмента данных
mov dx, OFFSET Greeting     ; Загрузка в dx смещения
                             ; адреса текста приветствия

DisplayGreeting:
mov ah, 9                   ; # функции ДОС печати строки
int 21h                     ; вывод на экран приветствия
mov ah, 4ch                 ; # функции ДОС завершения программы
int 21h                     ; завершение программы и выход в ДОС
END
```

## Название файла: hello2.asm

```
; HELLO2 - Учебная программа N2 лаб.раб.#1 по дисциплине "Архитектура
компьютера"
; Программа использует процедуру для печати строки
;
; ТЕКСТ ПРОГРАММЫ

EOFLine EQU '$' ; Определение символьной константы
; "Конец строки"

; Стек программы

ASSUME CS:CODE, SS:AStack

AStack SEGMENT STACK
DW 12 DUP('!') ; Отводится 12 слов памяти
AStack ENDS

; Данные программы

DATA SEGMENT

; Директивы описания данных

HELLO DB 'Hello Worlds!', 0AH, 0DH, EOFLine
GREETING DB 'Student from 1384 - Usacheva Daria $'
DATA ENDS

; Код программы

CODE SEGMENT
; Процедура печати строки
WriteMsg PROC NEAR
mov AH, 9
int 21h ; Вызов функции DOS по прерыванию
ret
WriteMsg ENDP

; Головная процедура
Main PROC FAR
push DS ;\ Сохранение адреса начала PSP в стеке
sub AX, AX ; > для последующего восстановления по
push AX ;/ команде ret, завершающей процедуру.
mov AX, DATA ; Загрузка сегментного
mov DS, AX ; регистра данных.
mov DX, OFFSET HELLO ; Вывод на экран первой
call WriteMsg ; строки приветствия.
mov DX, OFFSET GREETING ; Вывод на экран второй
call WriteMsg ; строки приветствия.
ret ; Выход в DOS по команде,
; находящейся в 1-ом слове PSP.

Main ENDP
CODE ENDS
END Main
```

## ПРИЛОЖЕНИЕ В ЛИСТИНГИ

Название файла: 11.lst

Microsoft (R) Macro Assembler Version 5.10

9/28/22 02:42:45

Page 1-1

```

; HELLO1.ASM - упрощенная версия учебной программы лаб.рб. N1
;               по дисциплине "Архитектура компьютера"
; *****
; Назначение: Программа формирует и выводит на экран приветствие
;               пользователя с помощью функции DOS
"Вывод строки"
;               (номер 09 прерывания 21h), которая:
;               - обеспечивает вывод на экран строки
СИМВОЛ
ОВ,
;               заканчивающейся знаком "$";
;               - требует задания в регистре ah номера функции=09h,
;               а в регистре dx - значения адреса
ВЫВОДИМО
й
;               строки;
;               - использует регистры ah и не
сохраняет его
;               содержимое.
; *****

DOSSEG
; Задание сегментов под DOS
C
.MODEL SMALL
; Модель памяти-SMALL (Малая)
.STACK 100h
; Отвести под стек 256 байт
.DATA
; Начало сегмента данных
0000 Greeting LABEL BYTE
; Текст приветствия
0000 53 74 75 64 65 6E DB 'Student Usacheva Daria of 1384 group is greeting
you',13,10,'$'
74 20 52 79 6E 64
79 63 68 20 41 2E
45 2E 20 6F 66 20
31 33 38 34 20 67
72 6F 75 70 20 69
73 20 67 72 65 65
```

6F 75 0D 0A 24

```

        .CODE                                ; На
        ; начало сегмента кода
0000    B8 ---- R    mov ax, @data           ; За
        ; грузка в DS адреса начала
0003    8E D8        mov ds, ax             ; се
        ; гмента данных
0005    BA 0000 R    mov dx, OFFSET Greeting ; За
        ; грузка в dx смещения
                                ; ад
0008    ; аса текста приветствия
        DisplayGreeting:
0008    B4 09        mov ah, 9              ; # ф
        ; ункции ДОС печати строки
000A    CD 21        int 21h                ; ВЫ
000C    B4 4C        mov ah, 4ch            ; # ф
        ; ункции ДОС завершения про
        ; граммы
000E    CD 21        int 21h                ; за
        ; ершение программы и выход
        ; в ДОС
        END

```

Microsoft (R) Macro Assembler Version 5.10

9/28/22 02:42:45  
Symbols-1

# Segments and Groups:

Name	Length	Align	Combine	Class
DGROUP . . . . .	GROUP			
_DATA . . . . .	0035	WORD	PUBLIC	'DATA'
_STACK . . . . .	0100	PARA	STACK	'STACK'
_TEXT . . . . .	0010	WORD	PUBLIC	'CODE'

## Symbols:

Name	Type	Value	Attr
DISPLAYGREETING . . . . .	L NEAR	0008	_TEXT
GREETING . . . . .	L BYTE	0000	_DATA
@CODE . . . . .	TEXT	_TEXT	
@CODESIZE . . . . .	TEXT	0	
@CPU . . . . .	TEXT	0101h	
@DATASIZE . . . . .	TEXT	0	
@FILENAME . . . . .	TEXT	HELLO1	
@VERSION . . . . .	TEXT	510	

33 Total Lines  
19 Symbols

47998 + 459262 Bytes symbol space free

0 Warning Errors  
0 Severe Errors

## Название файла: Hello2.lst

Microsoft (R) Macro Assembler Version 5.10

9/28/22 03:21:13

Page 1-1

```

; HELLO2 - Учебная программа N2
; лаб.раб.#1 по дисциплине "Архитектура компьютера"
; Программа использует процедуру для печати строки
;
; ТЕКСТ ПРОГРАММЫ

= 0024 EOFLine EQU '$' ; Определена символьная константа
; "Конец строки"
;
; Стек программы

ASSUME CS:CODE, SS:AStack

0000 AStack SEGMENT STACK
0000 000C[ DW 12 DUP('!') ; Отводится 12 слов памяти
0021 ]

0018 AStack ENDS

; Данные программы

0000 DATA SEGMENT

; Директивы описания данных

0000 48 65 6C 6C 6F 20 HELLO DB 'Hello Worlds!', 0AH, 0DH, EOFLine
57 6F 72 6C 64 73
21 0A 0D 24
0010 53 74 75 64 65 6E GREETING DB 'Student from 1384 Usacheva
$'
74 20 66 72 6F 6D
20 31 33 38 34 20
52 79 6E 64 79 63
68 20 41 2E 45 2E
```

```

20 2D 20 24
0032          DATA          ENDS

          ; Код программы

0000          CODE          SEGMENT
          ; Процедура печати строки
0000 WriteMsg PROC NEAR
0000 B4 09          mov     AH,9
0002 CD 21          int     21h ; Вызов функции
          ; DOS по прерыванию
0004 C3          ret
Microsoft (R) Macro Assembler Version 5.10
9/28/22 03:21:13
Page 1-2

0005 WriteMsg ENDP

          ; Головная процедура
0005 Main PROC FAR
0005 1E          push     DS ;\ Сохранени
          е адреса начала PSP в стеке
0006 2B C0          sub     AX,AX ; > для после
          ;ующего восстановления по
0008 50          push     AX ;/ команде ret
          , завершающей процедуру.
0009 B8 ---- R          mov     AX,DATA ; Загр
          ;зка сегментного
000C 8E D8          mov     DS,AX ; реги
          ;тра данных.
000E BA 0000 R          mov     DX, OFFSET HELLO ; Выво
          ; на экран первой
0011 E8 0000 R          call    WriteMsg ; стро
          ;и приветствия.
0014 BA 0010 R          mov     DX, OFFSET GREETING ; Выво
          ; на экран второй
0017 E8 0000 R          call    WriteMsg ; стро
          ;и приветствия.
001A CB          ret ; Выхо
          ; в DOS по команде,
          ; нахо
          ;ящейся в 1-ом слове PSP.
001B Main ENDP
001B CODE ENDS
          END Main

```



## Segments and Groups:

N a m e	Length	Align	Combine Class
ASTACK . . . . .	0018	PARA	STACK
CODE . . . . .	001B	PARA	NONE
DATA . . . . .	0032	PARA	NONE

## Symbols:

N a m e	Type	Value	Attr
EOFLINE . . . . .	NUMBER		0024
GREETING . . . . .	L BYTE		0010 DATA
HELLO . . . . .	L BYTE		0000 DATA
MAIN . . . . .	F PROC		0005 CODE Length = 0016
WRITEMSG . . . . .	N PROC		0000 CODE Length = 0005
@CPU . . . . .	TEXT	0101h	
@FILENAME . . . . .	TEXT	HELLO2	
@VERSION . . . . .	TEXT	510	

52 Source Lines

52 Total Lines

13 Symbols

47990 + 459267 Bytes symbol space free

0 Warning Errors

0 Severe Errors