

**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: Трансляции, отладка и выполнение программ на языке**  
**Ассемблера**

Студентка гр. 0382

Михайлова О.Д.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

## **Цель работы.**

Освоение трансляции, отладки и выполнение программ на языке Ассемблера.

## **Задание.**

### Часть 1

1. Просмотреть программу hello1.asm, которая формирует и выводит на экран приветствие пользователя с помощью функции ОС MSDOS, 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 ax 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. Выполнить программу в пошаговом режиме с фиксацией используемых регистров и ячеек памяти до и после выполнения каждой команды.

## Часть 2

Выполнить пункты 1 - 7 части 1 настоящего задания применительно к программе hello2.asm, приведенной в каталоге Задания, которая выводит на экран приветствие пользователя с помощью процедуры WriteMsg, а также использует полное определение сегментов. Сравнить результаты прогона под управлением отладчика программ hello1 и hello2 и объяснить различия в размещении сегментов.

## **Выполнение работы.**

### **Часть 1:**

- 1) Просмотрен и разобран код программы hello1.asm
- 2) Строка-приветствие преобразована в соответствии со своими личными данными.
- 3) Выполнена трансляция файла hello1.asm в hello1.obj и создан файл листинг hello1.lst. Синтаксических ошибок не обнаружено.
- 4) Скомпонован загрузочный модуль, создан файл hello1.map и исполняемый файл hello1.exe
- 5) Выполнен запуск файла в автоматическом режиме. На экран выведено сообщение: «Вас приветствует ст.гр.0382 - Михайлова О.Д.»
- 6) Произведен запуск выполнения программы под управлением отладчика.

Результаты выполнения программы hello1.exe под управлением отладчика представлены в таблице 1.

Начальное содержимое сегментных регистров: CS = 1A05, DS = 19F5, ES = 19F5, SS = 1A0C

Таблица 1 - результаты выполнения программы hello1.exe

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	IP = 001E	IP = 0010

## Часть 2:

- 1) Просмотрен и разобран код программы hello2.asm
- 2) Строка-приветствие преобразована в соответствии со своими личными данными.
- 3) Выполнена трансляция файла hello2.asm в hello2.obj и создан файл листинг hello2.lst. Синтаксических ошибок не обнаружено.
- 4) Скомпонован загрузочный модуль, создан файл hello2.map и исполняемый файл hello2.exe

5) Выполнен запуск файла в автоматическом режиме. На экран выведено сообщение: «Hello Worlds! \n Student from 0382 - Mikhailova O.D.»

6) Произведен запуск выполнения программы под управлением отладчика.

Результаты выполнения программы hello2.exe под управлением отладчика представлены в таблице 2.

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

Таблица 2 - результаты выполнения программы hello2.exe

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0005	PUSH DS	1E	IP = 0005 DS = 19F5 SP = 0018 STACK +0 0000	IP = 0006 DS = 19F5 SP = 0016 STACK +0 19F5
0006	SUB AX, AX	2BC0	AX = 0000 IP = 0006	AX = 0000 IP = 0008
0008	PUSH AX	50	IP = 0008 AX = 0000 STACK +0 19F5 +2 0000	IP = 0009 AX = 0000 STACK +0 0000 +2 19F5
0009	MOV AX, 1A07	B8071A	AX = 0000 IP = 0009	AX = 1A07 IP = 000C
000C	MOV DS, AX	8ED8	DS = 19F5 AX = 1A07 IP = 000C	DS = 1A07 AX = 1A07 IP = 000E
000E	MOV DX, 0000	BA0000	DX = 0000 IP = 000E	DX = 0000 IP = 0011
0011	CALL 0000	E8ECFF	IP = 0011	IP = 0000

			SP = 0014 STACK +0 0000 +2 19F5 +4 0000	SP = 0012 STACK +0 0014 +2 0000 +4 19F5
0000	MOV AH, 09	B409	AX = 1A07 IP = 0000	AX = 0907 IP = 0002
0002	INT 21	CD21	IP = 0002	IP = 0004
0004	RET	C3	IP = 0004 SP = 0012 STACK +0 0014 +2 0000 +4 19F5	IP = 0014 SP = 0014 STACK +0 0000 +2 19F5 +4 0000
0014	MOV DX, 0010	BA1000	DX = 0000 IP = 0014	DX = 0010 IP = 0017
0017	CALL 0000	E8E6FF	IP = 0017 SP = 0014 STACK +0 0000 +2 19F5 +4 0000	IP = 0000 SP = 0012 STACK +0 001A +2 0000 +4 19F5
0000	MOV AH, 09	B409	AX = 0907 IP = 0000	AX = 0907 IP = 0002
0002	INT 21	CD21	IP = 0002	IP = 0004
0004	RET	C3	IP = 0004 SP = 0012 STACK +0 001A +2 0000 +4 19F5	IP = 001A SP = 0014 STACK +0 0000 +2 19F5 +4 0000
001A	RET Far	CB	IP = 001A SP = 0014 CS = 1A0B	IP = 0000 SP = 0018 CS = 19F5
0000	INT 20	CD20	IP = 0000	IP = 0005

--	--	--	--	--

### **Выводы.**

Был изучен синтаксис языка Ассемблера и освоены трансляция, выполнение и отладка программ.

# ПРИЛОЖЕНИЕ А

## ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: hello1.asm

```
DOSSEG
.MODEL    SMALL
.STACK    100h
.DATA
Greeting LABEL BYTE
    DB 'Вас приветствует ст.гр.0382 - Михайлова О.Д.',13,10,'$'
.CODE
    mov ax, @data
    mov ds, ax
    mov dx, OFFSET Greeting

DisplayGreeting:
    mov ah, 9
    int 21h
    mov ah, 4ch
    int 21h
    END
```

Название файла: hello2.asm

```
EOFLine EQU '$'

ASSUME CS:CODE, SS:AStack

AStack    SEGMENT STACK
    DW 12 DUP('!')
AStack    ENDS

DATA      SEGMENT

HELLO      DB 'Hello Worlds!', 0AH, 0DH,EOFLine
GREETING   DB 'Student from 0382 - Mikhailova O.D. $'
DATA       ENDS

CODE       SEGMENT
WriteMsg   PROC NEAR
    mov AH,9
    int 21h
    ret
WriteMsg   ENDP

Main       PROC FAR
    push DS
    sub AX,AX
    push AX
    mov AX,DATA
    mov DS,AX
    mov DX, OFFSET HELLO
    call WriteMsg
    mov DX, OFFSET GREETING
```



```
        call WriteMsg
        ret

Main    ENDP
CODE    ENDS
        END Main
```

## ПРИЛОЖЕНИЕ Б

### ЛИСТИНГИ

Название файла: hello1.lst

#Microsoft (R) Macro Assembler Version 5.10  
9/15/21 16:13:42  
Page

1-1

```
DOSSEG
.MODEL SMALL
.STACK 100h
.DATA
0000 Greeting LABEL BYTE
0000 DB 'Вас приветствует ст.гр.0382 -
Михайлова О.Д.',13,10,'$'
20 DB BF D1 80 D0
B8 DB B2 D0 B5 D1
82 DB 81 D1 82 D0
B2 DB 83 D0 B5 D1
82 DB 20 D1 81 D1 82
2E DB B3 D1 80 2E
30 DB 33 38 32 20 2D
```

#Microsoft (R) Macro Assembler Version 5.10 9/15/21  
16:13:42  
Page

1-2

```
20 DB 9C D0 B8 D1
85 DB B0 D0 B9 D0
BB DB BE D0 B2 D0
B0 DB 20 D0 9E 2E D0
94 DB 2E 0D 0A 24

.CODE
0000 B8 ---- R mov ax, @data
0003 8E D8 mov ds, ax
0005 BA 0000 R mov dx, OFFSET Greeting
0008 DisplayGreeting:
0008 B4 09 mov ah, 9
000A CD 21 int 21h
000C B4 4C mov ah, 4ch
000E CD 21 int 21h

END
```

#Microsoft (R) Macro Assembler Version 5.10 9/15/21  
16:13:42  
Symbol

s-1

Segments and Groups:

N a m e	Length	Align	Combine Class
DGROUP . . . . .	GROUP		
_DATA . . . . .	004D	WORD PUBLIC	'DATA'
STACK . . . . .	0100	PARA STACK	'STACK'
_TEXT . . . . .	0010	WORD PUBLIC	'CODE'

Symbols:

	N a m e	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 Source Lines  
33 Total Lines  
19 Symbols

47994 + 459266 Bytes symbol space free

0 Warning Errors  
0 Severe Errors

Название файла: hello2.lst

#Microsoft (R) Macro Assembler Version 5.10  
9/15/21 11:51:58  
Page

1-1

```

= 0024                                EOFLine EQU '$'
                                ASSUME CS:CODE, SS:AStack
0000                                AStack SEGMENT STACK
0000 000C[DW 12 DUP('!')0021]

0018                                AStack ENDS
0000                                DATA SEGMENT

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 0382 -
Mikhailova O.D. $'
                                74 20 66 72 6F 6D
                                20 30 33 38 32 20
                                2D 20 4D 69 6B 68
                                61 69 6C 6F 76 61
                                20 4F 2E 44 2E 20
                                24

0035                                DATA ENDS
0000                                CODE SEGMENT
0000                                WriteMsg PROC NEAR
0000 B4 09                                mov AH,9
0002 CD 21                                int 21h

```

1-2

```
0004 C3                ret
0005                WriteMsg ENDP
0005                Main      PROC FAR
0005 1E                push    DS
0006 2B C0            sub     AX,AX
0008 50                push    AX
0009 B8 ---- R        mov     AX,DATA
000C 8E D8            mov     DS,AX
000E BA 0000 R        mov     DX, OFFSET HELLO
0011 E8 0000 R        call    WriteMsg
0014 BA 0010 R        mov     DX, OFFSET GREETING
0017 E8 0000 R        call    WriteMsg
001A CB                ret
001B                Main      ENDP
001B                CODE     ENDS
                                END Main
```

s-1

Segments and Groups:

	N a m e	Length	Align	Combine Class
ASTACK	. . . . .	0018	PARA	STACK
CODE	. . . . .	001B	PARA	NONE
DATA	. . . . .	0035	PARA	NONE

Symbols:

	N a m e	Type	Value	Attr
EOFLINE	. . . . .	NUMBER		0024
GREETING	. . . . .	L BYTE		0010 DATA
HELLO	. . . . .	L BYTE		0000 DATA
0016 MAIN	. . . . .	F PROC		0005 CODE Length =
0005 WRITEMSG	. . . . .	N PROC		0000 CODE Length =
@CPU	. . . . .	TEXT	0101h	
@FILENAME	. . . . .	TEXT	hello2	
@VERSION	. . . . .	TEXT	510	

52 Source Lines  
52 Total Lines  
13 Symbols

47986 + 459271 Bytes symbol space free

0 Warning Errors

0 Severe Errors