

**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»**  
**Тема: Трансляция, отладка и выполнение программ на языке**  
**Ассемблера**

Студент(ка) гр. 1381

Биктагирова Д.С.

Преподаватель

Ефремов М. А.

Санкт-Петербург

2022

## **Цель работы.**

Научится транслировать программу с помощью компилятора MASM (с получением файла листинга), компоновать загрузочный модуль и выполнять программу с помощью отладчика AFDPRO. Разобраться в особенностях хранения сегментов программы в памяти. Изучить, как меняется содержимое регистров в ходе выполнения программы.

## **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. Результаты прогона программы под управлением отладчика должны быть представлены в виде, показанном на примере одной команды в табл.1, и подписаны преподавателем.

Табл. 1

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0003	Mov DS, AX	8E D8	(AX) = 2D87 (DS) = 2D75 (IP) = 0003	(AX) = 2D75 (DS) = 2D87 (IP) = 0005

## 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.

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

```
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 d:/tools
Drive C is mounted as local directory d:/tools\

Z:\>c:

C:\>_
```

1. Была просмотрена программа hello1.asm, изучена ее структура и реализация каждого сегмента программы. Приветствие изменено на следующее: «Вас приветствует ст.гр.1381 – Биктагирова Д. С.»

```
change.log x HELLO1.LST x HELLO2.LST x hello1.asm x hello2.asm x
4 ; *****
5 ; Назначение: Программа формирует и выводит на экран приветствие
6 ; пользователя с помощью функции ДОС "Вывод строки"
7 ; (номер 09 прерывание 21h), которая:
8 ; - обеспечивает вывод на экран строки символов,
9 ; заканчивающейся знаком "$";
10 ; - требует задания в регистре ah номера функции=09h,
11 ; а в регистре dx - смещения адреса выводимой
12 ; строки;
13 ; - использует регистр ax и не сохраняет его
14 ; содержимое.
15 ; *****
16
17 DOSSEG ; Задание сегментов под ДОС
18 .MODEL SMALL ; Модель памяти-SMALL (Малая)
19 .STACK 100h ; Отвести под Стек 256 байт
20 .DATA ; Начало сегмента данных
21 Greeting LABEL BYTE ; Текст приветствия
22 DB 'Вас приветствует ст.гр.1381 - Биктагирова Д.С.',13,10,'$'
23 .CODE ; Начало сегмента кода
24 mov ax, @data ; Загрузка в DS адреса начала
25 mov ds, ax ; сегмента данных
26 mov dx, OFFSET Greeting ; Загрузка в dx смещения
```

2. Протранслирована программа командой «`masm hello1.asm`». В ходе трансляции создается объектный файл `hello1.obj` и файл диагностических сообщений `1hello1.lst`. Во время выполнения ошибки не были обнаружены

```
C:\>masm hello1.asm
Microsoft (R) Macro Assembler Version 5.10
Copyright (C) Microsoft Corp 1981, 1988. All rights reserved.

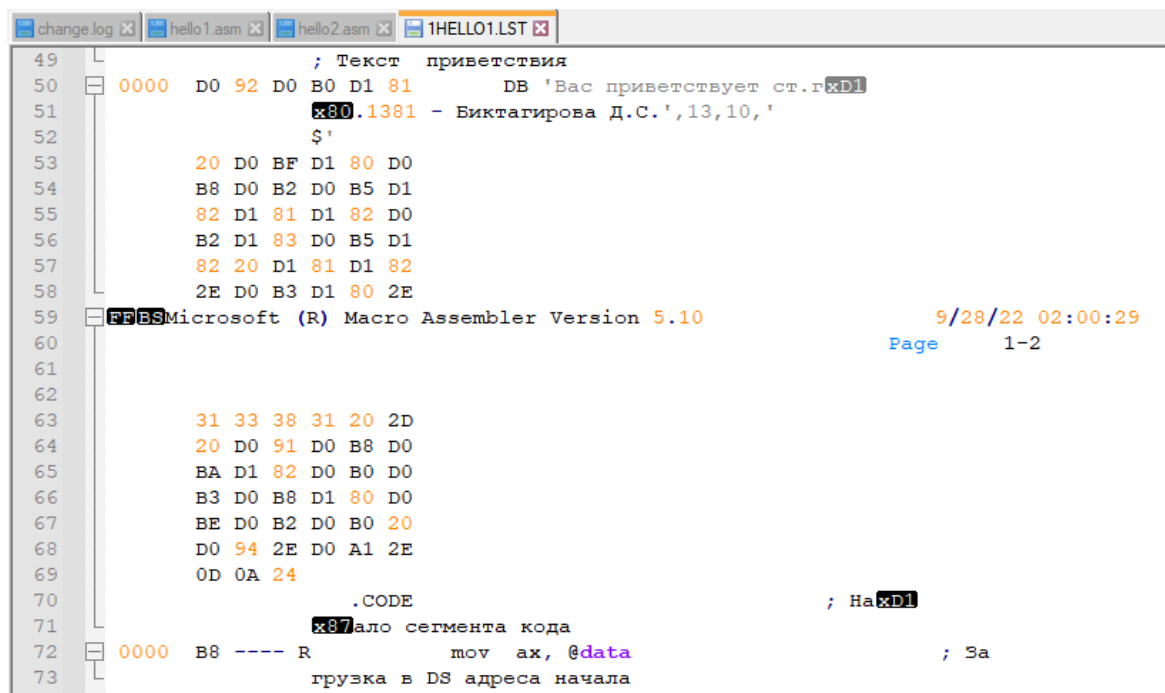
Object filename [hello1.OBJ]:
Source listing [NUL.LST]: 1hello1.lst
Cross-reference [NUL.CRF]:

47994 + 459266 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>
```

3. Просмотрен файл листинга `1hello1.lst`



```
49 ; Текст приветствия
50 0000 D0 92 D0 B0 D1 81 DB 'Вас приветствует ст.г'
51 x80.1381 - Биктагирова Д.С.',13,10,'
52 $'
53 20 D0 BF D1 80 D0
54 B8 D0 B2 D0 B5 D1
55 82 D1 81 D1 82 D0
56 B2 D1 83 D0 B5 D1
57 82 20 D1 81 D1 82
58 2E D0 B3 D1 80 2E
59 Microsoft (R) Macro Assembler Version 5.10
60 Page 1-2
61
62
63 31 33 38 31 20 2D
64 20 D0 91 D0 B8 D0
65 BA D1 82 D0 B0 D0
66 B3 D0 B8 D1 80 D0
67 BE D0 B2 D0 B0 20
68 D0 94 2E D0 A1 2E
69 0D 0A 24
70 .CODE ; Начало сегмента кода
71 x87ало сегмента кода
72 0000 B8 --- R mov ax, @data ; Загрузка в DS адреса начала
73 грузка в DS адреса начала
```

4. В коде файла `hello1.asm` была намеренно допущена ошибка

```
23 .CODE ; Начало сегмента кода
24 mov ux, @data ; Загрузка в DS адреса начала
25 mov ds, ax ; сегмента данных
26 mov dx, OFFSET Greeting ; Загрузка в dx смещения
27 ; адреса текста при
```

## 5. Снова протранслирована программа

```
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]: 2hello1
Cross-reference [NUL.CRF]:
hello1.asm(24): error A2009: Symbol not defined: UX

47994 + 459266 Bytes symbol space free

0 Warning Errors
1 Severe Errors

C:\>
```

## 6. Был просмотрен файл 2hello1.lst

```
72  0000 C7 06 0000 U ---- R      mov  ux, @data
73                                     грузка в DS адреса начала
74  hello1.asm(24): error A2009: Symbol not defined: UX
75  0006 8E D8                                     mov  ds, ax
76                                     гмента данных
77  0008 BA 0000 R                                     mov  dx, OFFSET Greeting
```

7. После в файле hello1.asm была исправлена ошибка и вновь протранслирована программа командой. В течение трансляции создается объектный файл hello1.obj и файл диагностических сообщений hello1.lst. Во время выполнения ошибки не были обнаружены.

```
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]: hello1
Cross-reference [NUL.CRF]:

47994 + 459266 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>
```

8. Была выполнена линковка программы командой link hello1.obj. В результате был создан загрузочный модуль hello1.exe и файл карты памяти hello1.map

```

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]: hello1
Libraries [.LIB]:

C:\>

```

9. Запущена программа командой hello1.exe

```

C:\>hello1
Das privetstvet student gr. 1381 тАУ Biktagirova D. S.

C:\>

```

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

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

AX 0000 SI 0000 CS 1A05 IP 0010 Stack +0 E4D9 Flags 7202  
 BX 0000 DI 0000 DS 19F5 +2 0F25  
 CX 005A BP 0000 ES 19F5 HS 19F5 +4 9BD8 OF DF IF SF ZF AF PF CF  
 DX 0000 SP 0100 SS 1A0B FS 19F5 +6 1F30 0 0 1 0 0 0 0 0

CMD >

Address	Instruction	Disassembly
0010	B8071A	MOV AX, 1A07
0013	BED8	MOV DS, AX
0015	BA0000	MOV DX, 0000
0018	B409	MOV AH, 09
001A	CD21	INT 21
001C	B44C	MOV AH, 4C
001E	CD21	INT 21
0020	56	PUSH SI

Memory dump (DS:0000):

Address	Hex	ASCII
DS:0000	CD 20 FF 9F 00 EA F0 FE	= Я.ЪЁ■ н ..†...
DS:0008	AD DE 1B 05 C5 06 00 00	.....Т. ....
DS:0010	18 01 10 01 18 01 92 01	.....ы.Л.
DS:0018	01 01 01 00 02 FF FF FF	В.....і. ....
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	

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

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

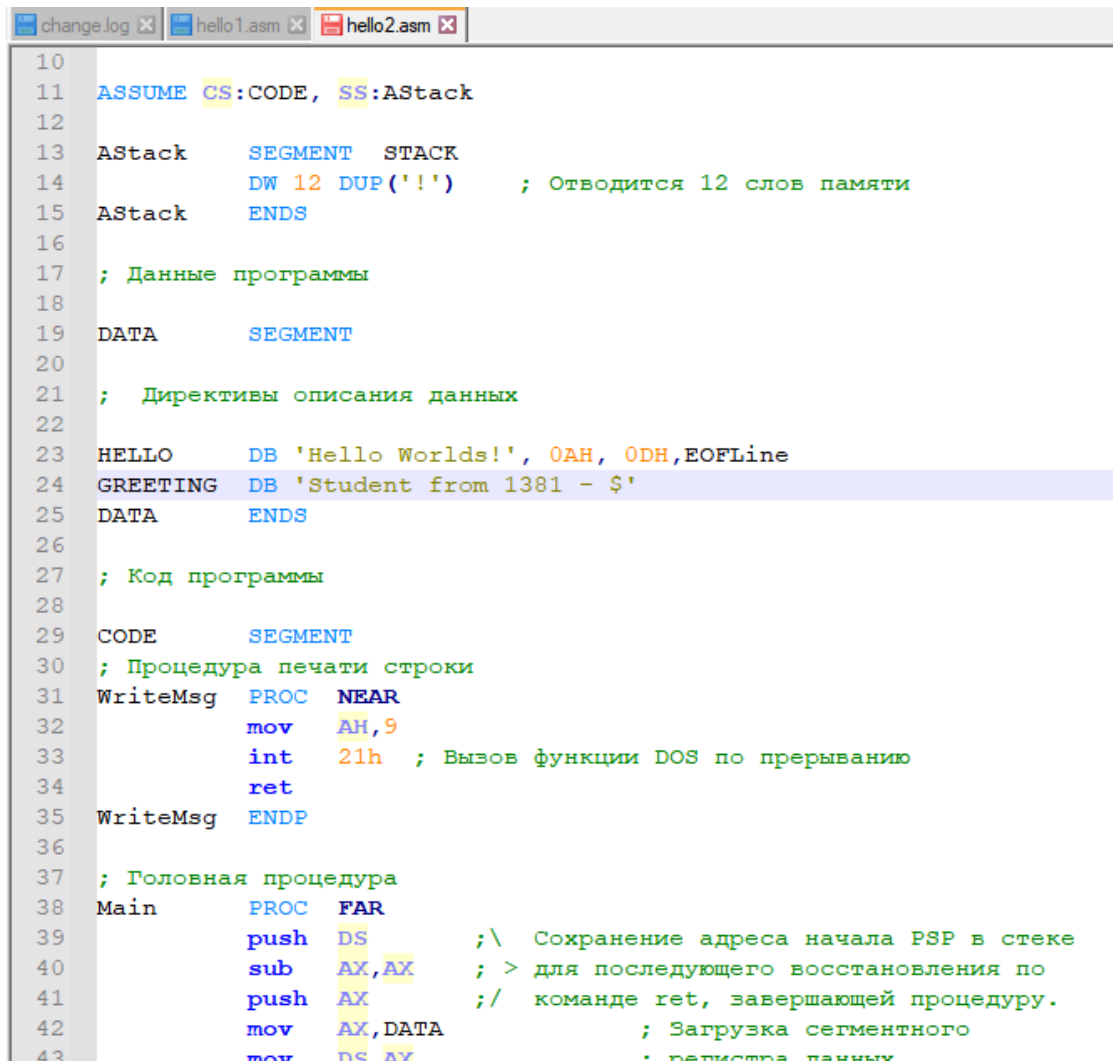
Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0010	MOV AX, 1A07	B8071A	(AX) = 0000 (IP) = 0010	(AX) = 1A07 (IP) = 0013

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



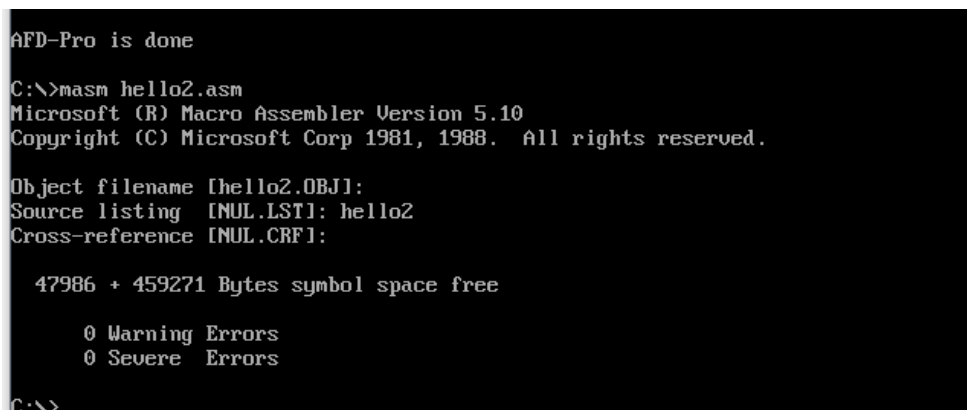
## Часть 2.

1. Была просмотрена программа hello2.asm в режиме редактирования, изучена ее структура и реализация каждого сегмента программы



```
10
11 ASSUME CS:CODE, SS:Astack
12
13 Astack    SEGMENT    STACK
14           DW 12 DUP('!!')    ; Отводится 12 слов памяти
15 Astack    ENDS
16
17 ; Данные программы
18
19 DATA     SEGMENT
20
21 ; Директивы описания данных
22
23 HELLO     DB 'Hello Worlds!', 0AH, 0DH, EOFLine
24 GREETING  DB 'Student from 1381 - $'
25 DATA     ENDS
26
27 ; Код программы
28
29 CODE      SEGMENT
30 ; Процедура печати строки
31 WriteMsg  PROC NEAR
32           mov AH, 9
33           int 21h ; Вызов функции DOS по прерыванию
34           ret
35 WriteMsg  ENDP
36
37 ; Головная процедура
38 Main      PROC FAR
39           push DS           ;\ Сохранение адреса начала PSP в стеке
40           sub AX, AX        ; > для последующего восстановления по
41           push AX           ;/ команде ret, завершающей процедуру.
42           mov AX, DATA     ; Загрузка сегментного
43           mov DS, AX        ; регистра данных
```

2. Была протранслирована программа командой `masm hello2.asm`. В ходе трансляции создается объектный файл `hello2.obj` и файл диагностических сообщений `hello2.lst`. Во время выполнения ошибки не были обнаружены.



```
AFD-Pro is done

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]: hello2
Cross-reference [NUL.CRF]:

47986 + 459271 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>
```

3. Линковка программы командой `link hello2.obj`. В результате был создан загрузочный модуль `hello2.exe` и файл карты памяти `hello2.map`

```
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]: hello2
Libraries [.LIB]:

C:\>
```

4. Запуск `hello2.exe` в автоматическом режиме. Убеждаемся, что все работает корректно

```
C:\>hello2
Hello Worlds!
Student from 1381 - Biktagirova D. S.
C:\>
```

5. Запуск программы `hello2.exe` под отладчиком с помощью команды `afipro 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
DX	0000	SP	0018	SS	1A05	FS	19F5		+6	0000	0	0	1	0	0
CMD >								1 0 1 2 3 4 5 6 7							
								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								= я.ъЁ■ н ..†...							
DS:0000 CD 20 FF 9F 00 EA F0 FE AD DE 1B 05 C5 06 00 00								.....T. ....							
DS:0010 18 01 10 01 18 01 92 01 01 01 01 00 02 FF FF FF								.....							
DS:0020 FF FF FF FF FF FF FF FF FF FF FF FF FF EB 19 C0 11								.....							
DS:0030 A2 01 14 00 18 00 F5 19 FF FF FF FF 00 00 00 00								B.....і. ....							
DS:0040 05 00 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 команды	16-ричный код команды	Содержимое регистров и ячеек 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 (IP) = 0014	(DX) = 0010 (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
0000	INT 20	CD20	(IP) = 0000	(IP) = 0005

### **Выводы.**

В ходе данной работы была разобрана структура и реализация каждого сегмента программ. Были изучены процессы трансляции, отладки и выполнения программ на языке Ассемблера.

## 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 'Vas privetstvuet student gr. 1381 - Biktagirova D.
S.',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 1381 - Biktagirova D. S. $'
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 Б

### ТЕСТИРОВАНИЕ

Если результаты тестирования велики (больше 1 страницы), то их выносят в приложение.

Процесс тестирования можно представить в виде таблицы, например:

Таблица Б.2 - Примеры тестовых случаев

№ п/п	Входные данные	Выходные данные	Комментарии
1.			
2.			
3.			
...			

Обратите внимание, что в нумерации таблицы в приложении обязательно должен быть в качестве префикса номер самого приложения: Б.