

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. 1383

Ермакова В.М.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Получение навыков трансляции, выполнения и отладки 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. Обычные команды выполняются по F1 (Step), а вызовы обработчиков прерываний (Int) - по F2 (StepProc), чтобы не входить внутрь обработчика прерываний. Продвижение по сегментам экранной формы отладчика выполняется с помощью клавиш F7 – F10 (up, down, left, right). Перезапуск программы в отладчике выполняется клавишей F3 (Retrieve). Выход из отладчика - по команде Quit.

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.

Работа с файлом hello1.asm

1.Просмотрен файл hello1.asm и изменены личные данные в соответствии с условием задания (Фамилия, Имя, группа)

2. Протранслирована программа

hello1.asm, с помощью команды:

>masm hello1.asm

После чего были созданы объектный файл, листинг файл и кросс файл HELLO1.OBJ, HELLO1.LST, HELLO1.CRF

3. Скомпанован объектный файл HELLO1.OBJ, после чего создан исполняемый файл HELLO1.EXE и файл HELLO1.MAP

4. Выполнена программа HELLO1.EXE, без смены кодировки на нормальную, вывод программы изображен на Рис. 1. После изменен на кириллицу на Рис. 2.

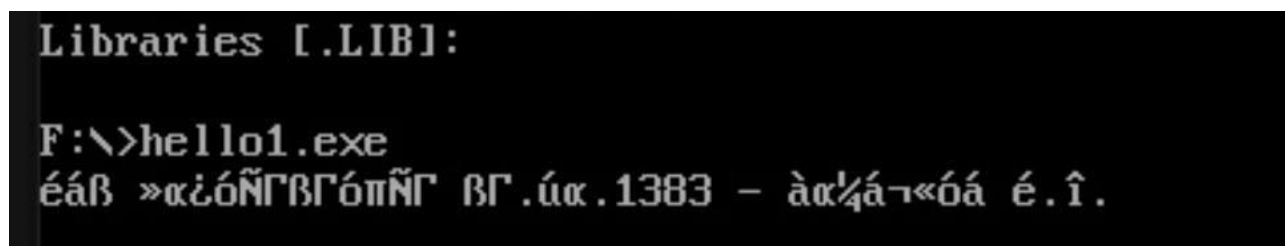


Рис. 1 Вывод программы hello1.exe

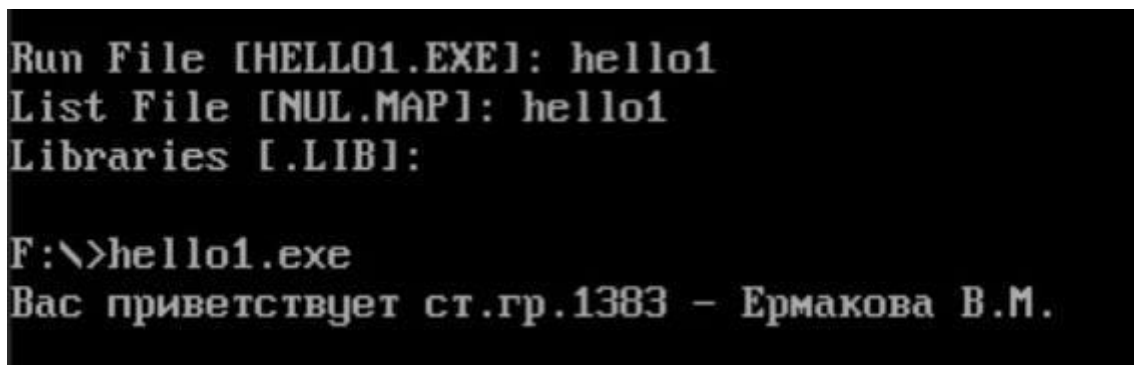


Рис. 2. Вывод программы hello2.exe на кириллице

6. С помощью команды

>afdpro hello1.exe

был запущен отладчик.

7. Начальное содержимое сегментных регистров

CS = 1A05, DS = 19F5, ES = 19F5, SS = 1A0D.

Таблица 1- исполнение файла hello1.asm

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, OFFSET Greeting	BA0000	(IP) = 0015	(IP) = 0018
0018	Mov ah, 9	B409	(AX) = 1A07 (IP) = 0018	(AX) = 0907 (IP) = 001A
001A	int 21h	CD21	(IP) = 001A	(IP) = 001C
001C	Mov ah, 4ch	B44C	(AX) = 0907 (IP) = 001C	(AX) = 4C07 (IP) = 001E
001E	int 21h	CD21	(AX) = 4C07 (CX) = 0071 (DS) = 1A07 (IP) = 001E	(AX) = 0000 (CX) = 0000 (DS) = 19F5 (IP) = 0010

Работа с файлом hello2.asm

1. Просмотрен файл hello2.asm и изменены личные данные в соответствии с условием задания (Фамилия, Имя, группа)

2. Протранслирована программа

hello2.asm, с помощью команды:

>masm hello2.asm

После чего были созданы объектный файл, листинг файл и кросс файл
HELLO2.OBJ, HELLO2.LST, HELLO2.CRF

3. Скомпилирован объектный файл HELLO2.OBJ, после чего создан исполняемый файл HELLO2.EXE и файл HELLO2.MAP

4. Выполнена программа HELLO2.EXE, вывод программы изображен на Рис. 2.

```
F:\>hello2.exe
Hello Worlds!
Student from 1383 - Ермакова В.Н.
F:\>keyb ru 866
Keyboard layout ru loaded for codepage 866
```

Рис. 2 Вывод программы hello2.exe

6. С помощью команды

>afdpro hello2.exe

был запущен отладчик.

7. Начальное содержимое сегментных регистров

CS = 1A0B, DS = 19F5, ES = 19F5, SS = 1A05.

Таблица 2- исполнение файла hello2.asm

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0005	PUSH DS	1E	Stack: +0 0000 (SP) = 0018 (IP) = 0005	Stack: +0 19F5 (SP) = 0016 (IP) = 0006
0006	SUB AX,AX	2BC0	(IP) = 0006	(IP) = 0008
0008	PUSH AX	50	Stack: +0 19F5 +2 0000 (IP) = 0008 (SP) = 0016	Stack: +0 0000 +2 19F5 (IP) = 0009 (SP) = 0014
0009	MOV AX,DATA	B8071A	(IP) = 0009 (AX) = 0000	(IP) = 000C (AX) = 1A07

000C	MOV DS,AX	8ED8	(IP) = 000C (DS) = 19F5	(IP) = 000E (DS) = 1A07
000E	MOV DX, 0000	BA0000	(IP) = 000E	(IP) = 0011
0011	CALL 0000	E8ECFF	Stack: +0 0000 +2 19F5 +4 0000 (IP) = 0011 (SP) = 0014	Stack: +0 0014 +2 0000 +4 19F5 (IP) = 0000 (SP) = 0012
0000	MOV AH, 09	B409	(IP) = 0000 (AX) = 1A07	(IP) = 0002 (AX) = 0907
0002	INT 21	CD21	(IP) = 0002	(IP) = 0004
0004	RET	C3	Stack: +0 0014 +2 0000 +4 19F5 (IP) = 0004 (SP) = 0012	Stack: +0 0000 +2 19F5 +4 0000 (IP) = 0014 (SP) = 0014
0014	MOV DX, 0010	BA1000	(IP) = 0014 (DX) = 0000	(IP) = 0017 (DX) = 0010
0017	CALL 0000	E8E6FF	Stack: +0 0000 +2 19F5 +4 0000 (IP) = 0017 (SP) = 0014	Stack: +0 001A +2 0000 +4 19F5 (IP) = 0000 (SP) = 0012
0000	MOV AH,9	B409	(IP) = 0000	(IP) = 0002
0002	INT 21	CD21	(IP) = 0001	(IP) = 0004

0004	RET	C3	Stack: +0 001A +2 0000 +4 19F5 (IP) = 0004 (SP) = 0012	Stack: +0 0000 +2 19F5 +4 0000 (IP) = 001A (SP) = 0014
001A	RET Far	CB	Stack: +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 001A (SP) = 0014 (CS) = 1A0A	Stack: +0 0000 +2 0000 +4 0000 +6 0000 (IP) = 0000 (SP) = 0018 (CS) = 19F5
0000	INT 20	CD20	(AX) = 0907 (DX) = 0010 (IP) = 0000 (CS) = 19F5 (DS) = 1A07	(AX) = 0000 (DX) = 0000 (IP) = 0005 (CS) = 1A0B (DS) = 19F5

Программный код см. в приложении А.

Файлы диагностических сообщений см. в приложении Б.

Выводы.

В ходе выполнения работы был получен навыки трансляции отладки и запуска программ на языке Ассемблера.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: hello1.asm

```
DOSSEG ; Задание сегмента под DOS
.MODEL SMALL ; Модель памяти - SMALL
.STACK 100h ; Отвести под Стек 256 байт
.DATA ; начало сегмента данных
Greeting LABEL BYTE ; Текст приветствия
DB Вас приветствует ст.гр.1383 - Ермакова В.М.',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 ; # функция DOS печати строки
int 21h ; вывод на экран приветствия
mov ah, 4ch ; # функции DOS завершения программы
int 21h ; завершение программы и выход в DOS
END
```

Название файла: hello2.asm

```
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 1383 - Ермакова В.М.$'
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              ; \ Созрание 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

ФАЙЛЫ ДИАГНОСТИЧЕСКИХ СООБЩЕНИЙ

Название файла: hello1.lst

Microsoft (R) Macro Assembler Version 5.10 10/8/22 24:56:17
Page 1-1

```

1
2      ; HELLO1.ASM - гІа®ЙГ п ўІабЁп гзІЎ
3      ;      ®© Іа®Ja ґґл « Ў.а Ў. N1
4      ;      І® ґЁбжЁІ«ЁІ "ґаеЁвІЄв
5      ;      га Є®ґІмовІа "
6      ;      *****
7      ;      *****
8      ; К § зІЁІ: Ца®Ja ґґ дґаґЁагІв Ё ўлў
9      ;      ®ґЁв нСа ІаЁўІвбвўЁІ
10     ;      І®«м§®ў вІ«п б І®ґЙмо дг
11     ;      ЄжЁЁ „ґ“ ",лў®ґ бва®ЄЁ"
12     ;      (®ґІа 09 ІаІалў ЁІ 21h)
13     ;      , Є®в®а п:
14     ;      - ®ЎІбІзЁў Ів ўлў®ґ
15     ;      нСа бва®ЄЁ бЁґў®«®ў,
16     ;      § Є зЁў ойІ©бп § Є®ґ
17     ;      "$";
18     ;      - ваІЎІгІв § ґ Ёп ў аІЈЁ
19     ;      бваІ аh ®ґІа дгЄжЁЁ=09h,
20     ;      ў аІЈЁбваІ dx - бґІ
21     ;      йІЁп ґаІб ўлў®ґЁґ®©
22     ;      бва®ЄЁ;
23     ;      - ЁбІ®«м§гІв аІЈЁбва ах
24     ;      Ё І б®еа пІв ІЈ®
25     ;      б®ґІаЈЁґІ.
26     ;      *****
27     ;      *****
28
29     DOSSEG
30     ; ґ ґ ЁІ бІЈґІв®ў І®ґ „ґ“

```

```

18      .MODEL SMALL
        ; Ъ®«Г«М Ї ¬пвЁ-SMALL(Ъ « п)
19      .STACK 100h
        ; ЪвЇГбвЁ Ї®« ‘вГ€ 256 Ы ©в
20      .DATA
        ; К з «® ЪГЈ¬Гв « ле
21 0000      Greeting LABEL BYTE
        ; ‘Г€бв ЇаЁЇГвбвЇЁп
22 0000 82 A0 E1 20 AF E0      DB ' , Ъ ЇаЁЇГвбвЇГв бв.Їа.1383 - ...а
        ¬ €®Ї ‘.Ъ.',13,10,'$'

23      A8 A2 A5 E2 E1 E2
24      A2 E3 A5 E2 20 E1
25      E2 2E A3 E0 2E 31
26      33 38 33 20 2D 20
27      85 E0 AC A0 AA AE
28      A2 A0 20 82 2E 8C
29      2E 0D 0A 24
30      .CODE
        ; К з «® ЪГЈ¬Гв €®«
31 0000 B8 ---- R      mov ax, @data
        ; ‡ Јаг$€ Ы DS «аГб з «
32 0003 8E D8      mov ds, ax
        ; ЪГЈ¬Гв « ле
33 0005 BA 0000 R      mov dx, OFFSET Greeting
Microsoft (R) Macro Assembler Version 5.10      10/8/22 24:56:17
Page      1-2

        ; ‡ Јаг$€ Ы dx б¬ГйГЁп
34
        ; «аГб вГ€бв ЇаЁЇГвбвЇЁп
35 0008      DisplayGreeting:
36 0008 B4 09      mov ah, 9
        ; # дГ€жЁЁ ‚Ѓ‘ Їз вЁ бва®€Ё
37 000A CD 21      int 21h
        ; ЫлЇ®« нСа ЇаЁЇГвбвЇЁп
38 000C B4 4C      mov ah, 4ch
        ; # дГ€жЁЁ ‚Ѓ‘ § ЫГайГЁп Їа®Ја ¬¬л
39 000E CD 21      int 21h
        ; § ЫГайГЁп Їа®Ја ¬¬л Ё Ыле®« Ы ‚Ѓ‘
40      END
Microsoft (R) Macro Assembler Version 5.10      10/8/22 24:56:17
Symbols-1

```

Segments and Groups:

N a m e	Length	Align	Combine Class
DGROUP	GROUP		
_DATA	002E	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

47458 + 461849 Bytes symbol space free

0 Warning Errors
0 Severe Errors

Название файла: hello2.lst

Microsoft (R) Macro Assembler Version 5.10 10/9/22 16:46:23
Page 1-1

```
1          ; HELLO2 - "зІŸ п Іа®Ja ¬¬ N2 « Ÿ.a
           Ÿ.#1 İ® ¢ĖбжĖİ«Ėİ "ҺаеĖВІЄвга Є®¬İмов
           Іа "
2          ; Ца®Ja ¬¬ Ėōİ®«м§гІв Іа®жІ¬г
           аг ¢«п İгз вĖ бва®ЄĖ
3          ;
4          ; '...Љ' Цђћгђћђћђћ
5
6 = 0024          EOFLine EQU '$' ; һІаІ¬І«ІĖİ
           бĖ¬Ÿ®«м®© Є®бв вл
7          ; "ЉІж
           бва®ЄĖ"
8
9          ; 'вІЄ Іа®Ja ¬¬л
10
11          ASSUME CS:CODE, SS:AStack
12
13 0000          AStack SEGMENT STACK
14 0000 000C[          DW 12 DUP(") ; һвŸ®Ėвбп
           12 б«®Ÿ İ ¬пвĖ
15      0021
16          ]
17
18 0018          AStack ENDS
19
20          ; ,, лІ Іа®Ja ¬¬л
21
22 0000          DATA SEGMENT
23
24          ; ,,ĖаІЄвĖŸл ®İĖб Ėп ¢ ле
25
26 0000 48 65 6C 6C 6F 20 HELLO DB 'Hello Worlds!', 0AH, 0DH,
           EOFLine
27      57 6F 72 6C 64 73
28      21 0A 0D 24
29 0010 53 74 75 64 65 6E GREETING DB 'Student from 1383 - ...а¬ Є
           ®Ÿ ,.Һ.$'
30      74 20 66 72 6F 6D
31      20 31 33 38 33 20
32      2D 20 85 E0 AC A0
33      AA AE A2 A0 20 82
34      2E 8C 2E 24
35 0032          DATA ENDS
```

```

36
37 ; Лб®¤ ĩa®Ja ¬¬л
38
39 0000 CODE SEGMENT
40 ; Цa®жГ¤ra Ĩз вĒ бba®ĒĒ
41 0000 WriteMsg PROC NEAR
42 0000 B4 09 mov AH,9
43 0002 CD 21 int 21h ; ,лš®Ÿ дгЄжĒĒ DO
S Ĩ® ĩaГалŸ Ēo
44 0004 C3 ret
45 0005 WriteMsg ENDP
Microsoft (R) Macro Assembler Version 5.10 10/9/22 16:46:23
Page 1-2

```

```

46
47 ; ĩ®«®Ÿ п ĩa®жГ¤ra
48 0005 Main PROC FAR
49 0005 1E push DS ; \ '®ea ГĒГ
¤aГб з « PSP Ÿ бвГĒГ
50 0006 2B C0 sub AX,AX ; > ¤«п Ĩ®б«Г¤
roЙГЈ® Ÿ®ббв ®Ÿ«ГĒп Ĩ®
51 0008 50 push AX ; / €®¬ ¤Г re
t, § ŸГan ойГ© ĩa®жГ¤ra.
52 0009 B8 ---- R mov AX,DATA ; ‡
JaršЄ бГЈ¬Гв®Ј®
53 000C 8E D8 mov DS,AX ; a
ГЈĒбba ¤ ле.
54 000E BA 0000 R mov DX, OFFSET HELLO ; ,
лŸ®¤ нCa ĨГaŸ®©
55 0011 E8 0000 R call WriteMsg ; б
ba®ĒĒ ĩaĒŸГвбŸĒп.
56 0014 BA 0010 R mov DX, OFFSET GREETING ; ,
лŸ®¤ нCa Ÿв®a®©
57 0017 E8 0000 R call WriteMsg ; б
ba®ĒĒ ĩaĒŸГвбŸĒп.
58 001A CB ret ; ,
ле®¤ Ÿ DOS Ĩ® €®¬ ¤Г,
59 ;
e®¤пЙГ©бп Ÿ 1-®¬ б«®ŸГ PSP.
60 001B Main ENDP
61 001B CODE ENDS
62 END Main
Microsoft (R) Macro Assembler Version 5.10 10/9/22 16:46:23
Symbols-1

```

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

47450 + 459810 Bytes symbol space free

0 Warning Errors

0 Severe Errors