

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: Трансляции, отладка и выполнение программ на языке Ассемблера.

Студент гр. 9382

Дерюгин Д.А.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

Цель работы.

Научится понимать программы на языке Ассамблера.

Задание.

Часть 1

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

3. Загрузил файл `hello1.asm` из каталога Задания в каталог Masm.

4. Протранслировал 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. Скомпоновал загрузочный модуль с помощью строки

```
> link hello1.obj
```

с созданием карты памяти и исполняемого файла `hello1.exe`.

6. Выполнил программу в автоматическом режиме путем набора строки

```
> hello1.exe
```

убедился в корректности ее работы и зафиксировать результат выполнения в протоколе.

7. Запустил выполнение программы под управлением отладчика с помощью команды

> afd hello1.exe

Записал начальное содержимое сегментных регистров CS, DS, ES и SS (Табл. 1 и табл. 4). Выполнил 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.

Таблицы результатов.

Табл. 1. (начальное содержание регистров)

CS	DS	ES	SS
1A05	19F5	19F5	19F5

Табл. 2. (HELLO1.EXE)

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0013	Mov DS, AX	8ED8	(DS) = 1A07 (AX) = 1A07 (IP) = 0013	(DS) = 1A07 (AX) = 1A07 (IP) = 0015
0015	MOV DX, 0000	BA0000	(DX) = 0000 (IP) = 0015	(DX) = 0000 (IP) = 0018
0018	MOV AH, 09	B409	(AX) = 1A07 (IP) = 18	(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) = 0020

Часть 2

Выполнил пункты 1 - 7 части 1 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.

Табл. 3 (Начальное положение регистров)

CS	DS	ES	SS
1A0A	19F5	19F5	1A05

Табл. 4 (HELLO2.EXE)

Адрес команды	Символически й код команд	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0005	PUSH DS	1E	(SP) = 0018 (IP) = 0005 (DS) = 19F5	(SP) = 0016 (IP) = 0006 (DS) = 19F5
0006	SUB AX, AX	2BC0	(AX) = 0000 (IP) = 0006	(AX) = 0000 (IP) = 0008
0008	PUSH AX	50	(AX) = 0000 (IP) = 0008 (SP) = 0016	(AX) = 0000 (IP) = 0009 (SP) = 0014
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 (SP) = 0014	(IP) = 0000 (SP) = 0012
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	(SP) = 0014 (IP) = 0014
0014	MOV DX, 0010	BA1000	(DX) = 0000 (IP) = 0014	(DX) = 0010 (IP) = 0017
0017	CALL 0000	E8E6FF	(IP) = 0017 (SP) = 0014	(IP) = 0000 (SP) = 0012
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	(SP) = 0014 (IP) = 001A
001A	RET Far	CB	(SP) = 0014 (IP) = 001A	(SP) = 0018 (IP) = 0000

Выводы.

В данной лабораторной работы мы научились работать с окладчиком AFDPRO, читать код на языке Ассамблера и работать с регистрами.

Тексты исходных файлов.

Файл 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 'Greetings from st. gr. 9389 Deruygin D.A.',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

```

Файл 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

```

```

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 4350 - $'
DATA ENDS

```

; Код программы

CODE SEGMENT

 ASSUME CS:Code DS:DATA SS:AStack

; 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

Файл HELLO1.LST

Microsoft (R) Macro Assembler Version 5.10

9/22/20 12:53:02

Page 1-1

; 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
С "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 регистры ah 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
0000 Greeting LABEL BYTE
; Текст приветствия
0000 82 A0 E1 20 AF E0 DB 'Вас приветствует ст.гр.7303 - Иванов И.И
.;13,10','$'

A8 A2 A5 E2 E1 E2
A2 E3 A5 E2 20 E1
E2 2E A3 E0 2E 37
33 30 33 20 2D 20
88 A2 A0 AD AE A2
20 88 2E 88 2E 0D
0A 24

.CODE ; Начал
о сегмента кода
0000 B8 ---- R mov ax, @data ; Загр
узка в DS адреса начала
0003 8E D8 mov ds, ax ; сегм
ента данных
0005 BA 0000 R mov dx, OFFSET Greeting ; Загр
узка в dx смещения

```

Microsoft (R) Macro Assembler Version 5.10

9/22/20 12:53:02

Page 1-2

```

; адрес
а текста приветствия
0008      DisplayGreeting:
0008 B4 09      mov ah, 9      ; # фу
          нкции ДОС печати строки
000A CD 21      int 21h      ; вывод
          на экран приветствия
000C B4 4C      mov ah, 4ch      ; # фу
          нкции ДОС завершения программы
000E CD 21      int 21h      ; завер
          шение программы и выход в ДОС
          END

```

Microsoft (R) Macro Assembler Version 5.10 9/22/20 12:53:02

Symbols-1

Segments and Groups:

N a m e	Length	Align	Combine	Class
DGROUP	GROUP			
_DATA	002C	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 + 461313 Bytes symbol space free

0 Warning Errors

0 Severe Errors

Файл HELLO2.LST

Microsoft (R) Macro Assembler Version 5.10

9/22/20 12:37:40

Page 1-1

; HELLO2 - Учебная программа N2 лаб.раб.#1 по
дисциплине "Архитектура компьютера"

; Программа использует процедуру для п
ечати строки

;

; ТЕКСТ ПРОГРАММЫ

```

= 0024          EOFLine EQU '$'      ; Определение символъ
              ной константы
              ; "Конец строки"

```

```

; стек программы

```

```

0000          AStack SEGMENT STACK
0000 000C[      DW 12 DUP(?) ; Отводится 12 слов п
              амяти
              ???
              ]

```

```

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    EGREETING DB 'Student from 4350 - $'
    74 20 66 72 6F 6D
    20 34 33 35 30 20
    2D 20 24
0025          DATA ENDS

```

; Код программы

0000 CODE SEGMENT

ASSUME CS:Code DS:DATA SS:AStack

HELLO2.ASM(28): warning A4001: Extra characters on line

; 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

0005 WriteMsg ENDP

; Головная процедура

0005 Main PROC FAR

0005 1E push DS ;\ Сохранение адреса начала PSP в стеке

0006 2B C0 sub AX,AX ; > для последующего в
остановления по

Microsoft (R) Macro Assembler Version 5.10

9/22/20 12:37:40

Page 1-2

0008 50 push AX ;/ команде ret, завершающей процедуру.

0009 B8 ---- R mov AX,DATA ; Загрузка
сегментного

000C 8E D8 mov DS,AX ; регистра
данных.

000E BA 0000 R mov DX, OFFSET HELLO ; Вывод на

```

                                экран первой
0011 E8 0000 R                call WriteMsg          ; строки пр
                                иветствия.
0014 BA 0010 R                mov  DX, OFFSET GREETING ; Вывод на
                                экран второй
0017 E8 0000 R                call WriteMsg          ; строки пр
                                иветствия.
001A CB                      ret                    ; Выход в D
                                OS по команде,
                                ; находящей
                                ся в 1-ом слове PSP.
001B                          Main    ENDP
001B                          CODE    ENDS
                                END Main

```

Microsoft (R) Macro Assembler Version 5.10 9/22/20 12:37:40
 Symbols-1

Segments and Groups:

N a m e	Length	Align	Combine	Class
ASTACK	0018	PARA		STACK
CODE	001B	PARA		NONE
DATA	0025	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	

51 Source Lines

51 Total Lines

13 Symbols

48002 + 461305 Bytes symbol space free

1 Warning Errors

0 Severe Errors