

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

Деткова А.С.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2018

Цель работы.

Познакомиться с основами языка Ассемблера, его трансляцией, отладкой, выполнением 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.

Результаты прогона программы под управлением отладчика должны быть представлены в виде таблицы.

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 текст программы hello1.asm. Программа разобрана построчно с объяснением каждого фрагмента кода. Строка приветствия изменена в соответствии с личными данными.
2. Выполнена загрузка hello1.asm в каталог Masm.
3. Программа протранслирована с помощью строки: > masm hello1.asm с созданием объектного файла и файла листинга.
4. Загрузочный модуль скомпонован с помощью строки: > link hello1.obj с созданием карты памяти и исполняемого файла hello1.exe.
5. Программа выполнена в автоматическом режиме путем набора строки: > hello1.exe, результатом получена строка: Hello, it is Detkova A.S. from 0382
6. Запущено выполнение программы под управлением отладчика: > afdpro hello1.exe

Начальные значения сегментных регистров:

(CS) = 1A05; (DS) = 19F5; (ES) = 19F5; (SS) = 1A0A

Результаты пошагового выполнения представлены в табл. 1.

Таблица 1 – Результаты пошагового выполнения

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	(AX) = 1A07 (DS) = 19F5 (IP) = 0013	(AX) = 1A07 (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:

Для файла hello2.asm были проделаны те же шаги, что и для hello1.asm

При запуске hello2.exe было выведено сообщение:

Hello worlds!

Student from 0382 — Detkova Anna.

Запущено выполнение программы под управлением отладчика:

> afdpro hello1.exe

Начальные значения сегментных регистров:

(CS) = 1A0B; (DS) = 19F5; (ES) = 19F5; (SS) = 1A05

Результаты пошагового выполнения представлены в табл. 2.

Таблица 2 – Результаты пошагового выполнения

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

			(IP) = 0009	(IP) = 000C
000C	MOV DS,AX	8ED8	(DS) = 19F5 (AX) = 1A07 (IP) = 001A	(DS) = 1A07 (AX) = 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	(SP) = 0012 (IP) = 0000 Stack: +0 0000 +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 0000 +2 0000 +4 19F5	(IP) = 0014 (SP) = 0014 Stack: +0 0000 +2 19F5
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	(SP) = 0012 (IP) = 0000 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

001A	RET Far	CB	(IP) = 001A (SP) = 0014 (CS) = 1A0B	(IP) = 0000 (SP) = 0018 (CS) = 19F5 Stack: +0 0000 +2 0000
0000	INT 20	CD20	(IP) = 0000	(IP) = 0005

Разработанный программный код см. в приложении А.

Выводы.

Был изучен основной синтаксис языка Ассемблера и правила написания программа на нём. Были разобраны процессы трансляции, компоновки, отладки и выполнения 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 'Hello, it is Detkova A.S. from 0382',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,EOLine
GREETING  DB 'Student from 0382 - Detkova Anna.$'
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

; Главная 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 А ЛИСТИНГИ

Название файла: hello1.lst

#Microsoft (R) Macro Assembler Version 5.10

9/15/21 21:48:15

Page 1-1

```
DOSSEG
.MODEL    SMALL
.STACK    100h
.DATA

0000          Greeting    LABEL    BYTE
0000    48 65 6C 6C 6F 2C          DB 'Hello, it is Detkova A.S. from
0382',13,10,'$'

        20 69 74 20 69 73
        20 44 65 74 6B 6F
        76 61 20 41 2E 53
        2E 20 66 72 6F 6D
        20 30 33 38 32 0D
        0A 24

.CODE
```

#Microsoft (R) Macro Assembler Version 5.10

9/15/21 21:48:15

Page 1-2

```
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
```

Segments and Groups:

N a m e	Length	Align	Combine	Class
DGROUP	GROUP			
_DATA	0026	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

47992 + 459268 Bytes symbol space free

0 Warning Errors

0 Severe Errors

Название файла: hello2.lst

#Microsoft (R) Macro Assembler Version 5.10

9/15/21

22:00:05

Page 1-1

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 - Detkova
Anna.\$'

74 20 66 72 6F 6D

20 30 33 38 32 20

2D 20 44 65 74 6B

6F 76 61 20 41 6E

6E 61 2E 24

0032 DATA ENDS

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

#Microsoft (R) Macro Assembler Version 5.10

9/15/21 22:00:05

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

47984 + 459273 Bytes symbol space free

0 Warning Errors

0 Severe Errors