

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.

Студент гр. 9382

Докукин В.М.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

Цель работы.

Изучить основные принципы трансляции, отладки и выполнения программ на языке Ассемблера.

Ход работы.

Часть 1.

1. Были загружены файлы AFDPRO.EXE, HELLO1.ASM, HELLO2.ASM, LINK.EXE, MASM.EXE в каталог C:\comp_arch\labs\tools.
2. Была запущена программа DOSBox, смонтирован виртуальный диск C: в каталоге MASM при помощи mount d c:\comp_arch\labs\tools.
3. Программа была разобрана в текстовом редакторе, были изучены структура и реализация каждого сегмента программы. Преобразована строку-приветствие.
4. В DOS осуществлен переход на виртуальный диск при помощи команды C:
5. Выполнена трансляция программы с помощью команды:
 > MASM HELLO1.ASM
В результате трансляции был создан объектный файл HELLO1.OBJ. Трансляция прошла без ошибок и предупреждений.
6. Слинкован загрузочный модуль HELLO1.exe с помощью строки:
 > LINK HELLO1.OBJ
В результате работы линковщика создан загрузочный модуль HELLO1.exe.
7. Загружена русскую кодовую таблицу символов вводом команды:
 > keyb ru 866
8. Запущена программа в автоматическом режиме путем набора строки:
 > HELLO1.EXE

9. Вывод программы:

> Greetings from 9382 - Dokukin

10. Используя отладчик, выполнен запуск программы HELLO1.EXE. В ходе выполнения программы записывались используемые регистры и ячейки памяти до и после выполнения команд. Отладчик был запущен при помощи команды:

> AFDPRO.EXE HELLO1.EXE.

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.

Содержимое сегментных регистров до старта программы: CS:1A05, DS:19F5, ES:19F5, SS:1A0B, HS:19F5, FS:19F5

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	(AX) = 4C07 (DS) = 1A07 (IP) = 001E	(AX) = 0000 (DS) = 19F5 (IP) = 0010

Часть 2.

1. Просмотрена программа HELLO2.ASM в текстовом редакторе, изучены структура и реализация каждого сегмента программы. Преобразована строку-приветствие.

2. Выполнена трансляция программы HELLO2.ASM с помощью транслятора MASM и команды:

>MASM HELLO2.ASM

В результате трансляции был создан объектный файл HELLO2.OBJ. Трансляция прошла без ошибок, но с одним предупреждением.

3. Используя линковщик LINK, создал загрузочный модуль HELLO2.EXE. Используемая команда:

>LINK HELLO2.EXE

4. Запустил программу HELLO2.EXE и проверил корректность ее работы:

> Hello Worlds!

> Student from 9382 – Dokukin V.M.

5. Запущена программа с помощью отладчика AFDPRO.EXE и пошагово исполнена с записью используемых регистров и ячеек команд.

Содержимое сегментных регистров до старта программы: CS:1A0B, DS:19F5, ES:19F5, SS:1A05, HS:19F5, FS:19F5

Табл.2.

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	(DS) = 19F5 (SP)=0016 (IP) = 0006 Stack +0 19F5 +2 0000 +4 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	(AX) = 0000 (SP) = 0014 (IP) = 0009 Stack +0 0000 +2 19F5 +4 0000
0009	MOV AX, 1A07	B8071A	(AX) = 0000 (IP) = 0009	(AX) = 1A07 (IP) = 000C
000C	MOV DS, AX	8ED8	(AX) = 1A07 (DS) = 19F5 (IP) = 000C	(AX) = 1A07 (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	(SP) = 0012 (IP) = 0000 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	(SP) = 0012 (IP) = 0004 Stack +0 0014 +2 0000 +4 19F5	(SP) = 0014 (IP) = 0014 Stack +0 0000 +2 119C +4 0000
0014	MOV DX,000A	BA0A00	(DX) = 0000 (IP) = 0014	(DX) = 000A (IP) = 0017
0017	CALL 0000	E8E6FF	(SP) = 0014 (IP) = 0017 Stack +0 0000 +2 19F5 +4 0000	(SP) = 0012 (IP) = 0000 Stack +0 001A +2 0000 +4 119C
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	(SP) = 0014 (IP) = 001A Stack +0 0000 +2 19F5 +4 0000
001A	RET Far	CB	(SP) = 0014 (IP) = 001A (CS) = 1A0B Stack +0 0000 +2 19F5 +4 0000	(SP) = 0018 (IP) = 0000 (CS) = 19F5 Stack +0 0000 +2 0000 +4 0000
0000	INT 20	CD20	(AX) = 0907 (DS) = 1A07 (IP) = 0000	(AX) = 0000 (DS) = 19F5 (IP) = 0005

Выводы.

В результате выполнения лабораторной работы был освоен этап трансляции, а также выполнение и пошаговое исполнение программы при помощи отладчика на языке Ассемблера.