**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра МОЭВМ**

**ОТЧЕТ**

**по лабораторной работе №1**

**по дисциплине «Организация ЭВМ и систем»**

**Тема: «Трансляции, отладка и выполнение программ на языке Ассемблера».**

|  |  |  |
| --- | --- | --- |
| Студент гр. 1303 |  | Иевлев Е. А. |
| Преподаватель |  | Ефремов М. А. |

Санкт-Петербург

2022

**Цель работы.**

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

**Задание**

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.

1. Разобраться в структуре и реализации каждого сегмента программы. Непонятные фрагменты прояснить у преподавателя. Строку-приветствие преобразовать в соответствии со своими личными данными.
2. Загрузить файл hello1.asm из каталога Задания в каталог Masm.
3. Протранслировать программу с помощью строки

> masm hello1.asm

c 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.

1. Скомпоновать загрузочный модуль с помощью строки

> link hello1.obj

с созданием карты памяти и исполняемого файла hello1.exe.

1. Выполнить программу в автоматическом режиме путем набора строки

> hello1.exe

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

1. 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.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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) = 2D87  (DS) = 2D87  (IP) = 0005 |

**Ход работы.**

**Часть 1.**

## Часть 1.

1. В текстовом редакторе была просмотрен код программы hello1.asm

2. Строка — приветствие была преобразована в соответствии с моими личными данными, каждый сегмент программы был разобран.

3. Файл hello1.asm был загружен в папку DOSBOX в эмуляторе MS-DOS

4. Программа была протранслированна с помощью команды:

> masm hello1.asm

5. Скомпонован загрузочный модуль с созданием карты памяти и исполняемого файла hello1.exe. с помощью строки:

> link hello1.obj

6. Программа выполнена в автоматическом режиме путем набора строки:

> hello1.exe

Программа выполнена корректно

7. Программа была запущена под управлением отладчика AFDPRO:

> afdpro hello1.exe

Содержимое сегментных регистров (CS) = 1A05, (DS) = 19F5, (ES) = 19F5, (SS) = 1A0A.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес Команды | Символический код команды | 16-ричный код команды | Содержимое регистров и ячеек памяти | |
| до выполнения | после выполнения |
| 0010 | MOV AX, 1A07 | B8071A | (IP) = 0010 (AX) = 0000 | (IP) = 0013 (AX) = 1A07 |
| 0013 | MOV DS, AX | 8ED8 | (IP) = 0013 (DS) = 19F5 | (IP) = 0015 (DS) = 1A07 |
| 0015 | MOV DX, 0000 | BA0000 | (IP) = 0015  (DX) = 0000 | (IP) = 0018 (DX) = 0000 |
| 0018 | MOV AH, 09 | B409 | (IP) = 0018 (AX) = 1A07 | (IP) = 001A (AX) = 0907 |
| 001A | INT 21 | CD21 | (IP) = 001A | (IP) = 001C |
| 001C | MOV AH, 4C | B44C | (IP) = 001C (AX) = 0907 | (IP) = 001E (AX) = 4C07 |
| 001E | INT 21 | CD21 | (IP) = 001E  (AX) = 4C07  (DS) = 1A07 | (IP) = 0010 (AX) = 0000 (DS) = 19F5 |

**Часть 2.**

## Часть 2.

1. В текстовом редакторе была просмотрен код программы hello2.asm2

2. Строка — приветствие была преобразована в соответствии с моими личными данными, каждый сегмент программы был разобран.

3. Файл hello2.asm был загружен в папку DOSBOX в эмуляторе MS-DOS

4. Программа была протранслированна с помощью команды:

> masm hello2.asm

5. Скомпонован загрузочный модуль с созданием карты памяти и исполняемого файла hello2.exe. с помощью строки:

> link hello2.obj

6. Программа выполнена в автоматическом режиме путем набора строки:

> hello2.exe

Программа выполнена корректно

7. Программа была запущена под управлением отладчика AFDPRO:

> afdpro hello2.exe

Содержимое сегментных регистров (CS) = 1A09, (DS) = 19F5, (ES) = 19F5, (SS) = 1A05.

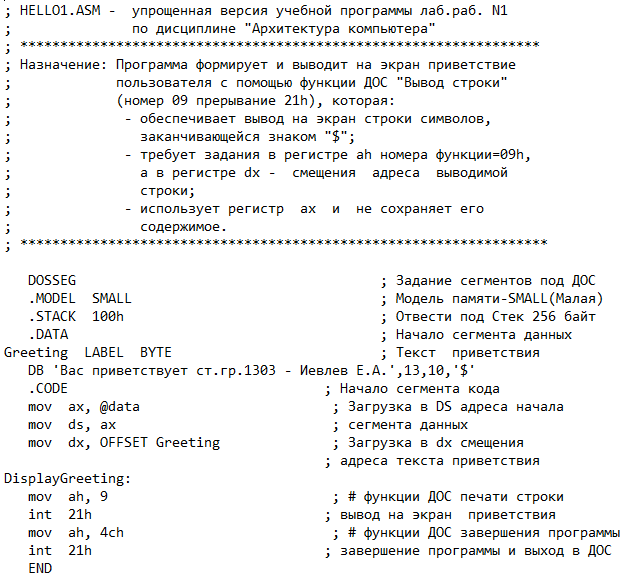
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес Команды | Символический код команды | 16-ричный код команды | Содержимое регистров и ячеек памяти | |
| до выполнения | после выполнения |
| 0005 | PUSH DS | 1E | (IP) = 0005  (DS) = 19F5  (SP) = 0018  Stack  +0 0000  +2 0000  +4 0000  +6 0000 | (IP) = 0006  (DS) = 19F5  (SP) = 0016  Stack  +0 19F5  +2 0000  +4 0000  +6 0000 |
| 0006 | SUB AX, AX | 2BC0 | (IP) = 0006  (AX) = 0000 | (IP) = 0008  (AX) = 0000 |
| 0008 | PUSH AX | 50 | (IP) = 0008  (AX) = 0000  (SP) = 0016  Stack  +0 19F5  +2 0000  +4 0000  +6 0000 | (IP) = 0009  (AX) = 0000  (SP) = 0014  Stack  +0 0000  +2 19F5  +4 0000  +6 0000 |
| 0009 | MOV AX, 1A07 | B8071A | (IP) = 0009  (AX) = 0000 | (IP) = 000C  (AX) = 1A07 |
| 000C | MOV DS, AX | 8ED8 | (IP) = 000C  (DS) = 19F5  (AX) = 1A07 | (IP) = 000E  (DS) = 1A07  (AX) = 1A07 |
| 000E | MOV DX, 0000 | BA0000 | (IP) = 000E  (DX) = 0000 | (IP) = 0011  (DX) = 0000 |
| 0011 | CALL 0000 | E8ECFF | (IP) = 0011  (SP) = 0014  Stack  +0 0000  +2 19F5  +4 0000  +6 0000 | (IP) = 0000  (SP) = 0012  Stack  +0 0014  +2 0000  +4 19F5  +6 0000 |
| 0000 | MOV AH, 09 | B409 | (IP) = 0000  (AX) = 1A07 | (IP) = 0002  (AX) = 0907 |
| 0002 | INT 21 | CD21 | (IP) = 0002 | (IP) = 0004 |
| 0004 | RET | C3 | (IP) = 0004  (SP) = 0012  Stack  +0 0014  +2 0000  +4 19F5  +6 0000 | (IP) = 0014  (SP) = 0014  Stack  +0 0000  +2 19F5  +4 0000  +6 0000 |
| 0014 | MOV DX, 0010 | BA1000 | (IP) = 0014  (DX) = 0000 | (IP) = 0017  (DX) = 0010 |
| 0017 | CALL 0000 | E8E6FF | (IP) = 0017  (SP) = 0014  Stack  +0 0000  +2 19F5  +4 0000  +6 0000 | (IP) = 001A  (SP) = 0012  Stack  +0 0000  +2 19F5  +4 0000  +6 0000 |
| 0000 | MOV AH, 09 | B409 | (IP) = 0000  (AX) = 0907 | (IP) = 0002  (AX) = 0907 |
| 0002 | INT 21 | CD21 | (IP) = 0002 | (IP) = 0004 |
| 0004 | RET | C3 | (IP) = 0004  (SP) = 0012  Stack  +0 001A  +2 0000  +4 19F5  +6 0000 | (IP) = 001A  (SP) = 0014  Stack  +0 0000  +2 19F5  +4 0000  +6 0000 |
| 001A | RET Far | CB | (IP) = 001A  (SP) = 0014  (CS) = 1A0B  Stack  +0 0000  +2 19F5  +4 0000  +6 0000 | (IP) = 0000  (SP) = 0018  (CS) = 19F5  Stack  +0 0000  +2 0000  +4 0000  +6 0000 |
| 0000 | INT 20 | CD20 | (IP) = 0000  (AX) = 0907  (DS) = 1A07 | (IP) = 0002  (AX) = 0000  (DS) = 19F5 |

**Выводы.**

В ходе выполнения лабораторной работы была освоена трансляция, линковка и режим отладчика. Разобраны структуры в работе программы. Изучены сегменты, регистры, логические и арифметические операции.

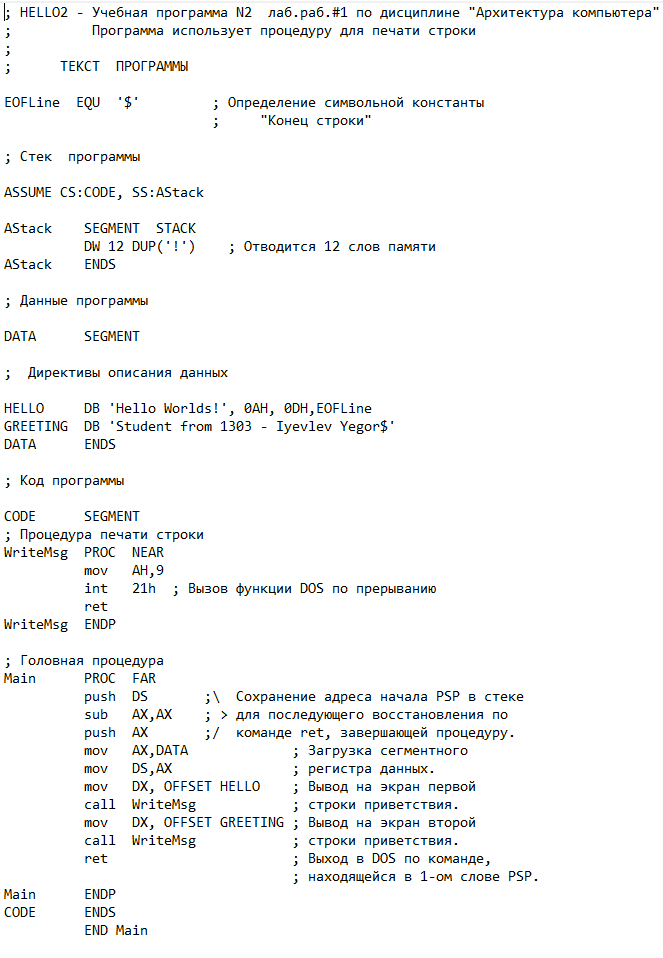
**Приложение 1**

**HELLO1.ASM**

****

**Приложение 2**

**HELLO2.ASM**

****