**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по практической работе №1**

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

**Тема: Трансляции, отладка и выполнение программ на языке Ассемблер.**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 1304 |  | Чернякова В.А. |
| Преподаватель |  | Кирьянчиков В.А. |

Санкт-Петербург

2022

## Цель работы.

Ознакомиться с трансляцией, выполнением и отладкой программы на языке Ассемблер.

## Задание.

Лабораторная работа 1 использует 2 готовых программы на ассемблере: hello1 – составлена с использованием сокращенного описания сегментов и hello2 – составлена с полным описанием сегментов и выводом строки, оформленным как процедура. Выполнение работы состоит из двух частей, по каждой из которых необходимо представить протокол с фиксацией всех выполняемых действий и полученных результатов, и подписать его у преподавателя.

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

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.

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, показанном на примере одной команды в табл.1, и подписаны преподавателем.

Табл.1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |

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.

## Ход работы.

**Часть 1.**

1. Установить программу DOSBox – эмулятор MS-DOS – для осуществления запуска транслятора Masm.exe. Скачать с официального сайта http://www.dosbox.com/, выбрав нужную операционную систему – Windows 10 x64.
2. Запустить DOSBox.
3. Смонтировать каталог с MASM в эмулятор DOSBox с помощью команды mount <диск в эмуляторе> <реальный путь к каталогу с MASM>. В данной лабораторной работе смонтируем mount каталог C:\lr1\MASM как диск d. 
4. Перейти на только что смонтированный диск, подав команду в эмуляторе d:. Содержимым диска d: в эмуляторе будет содержимое каталога C:\MASM\_EXE реального компьютера, то есть можно непосредственно запускать ассемблер, компоновщик и отладчик для работы с заданной программой.



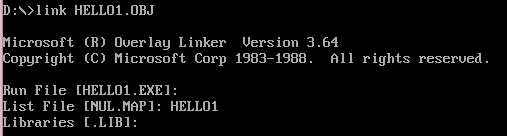
1. Осуществить переключение на русскую раскладку в DOSBox, так как в предложенной для лабораторной работы программе используются русские буквы. В DOSBox необходимо загрузить кодовую страницу с буквами кириллицы, подав команду keyb ru 866(переключение на кодировку CP866, использовавшуюся в MS-DOS для работы с буквами русского алфавита).



1. Просмотреть программу hello1.asm в режиме редактирования, используется редактор notepad++. Данная программа формирует и выводит на экран приветствие пользователя с помощью функции ОС MSDOS. Разобраться в структуре и реализации каждого сегмента программы. Строку, в которой содержится приветствие, преобразовать в соответствии со своими личными данными.
2. Протранслировать программу с помощью строки masm hello1.asm c созданием объектного файла, HELLO1.OBJ, и файла диагностических сообщений – файла листинга, LIST1.lst.



1. Скомпоновать загрузочный модуль, HELLO1.EXE, с помощью строки link HELLO1.OBJ с созданием карты памяти HELLO1.MAP 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 = 11AC, DS = 119C, ES = 119C и SS = 11B1.

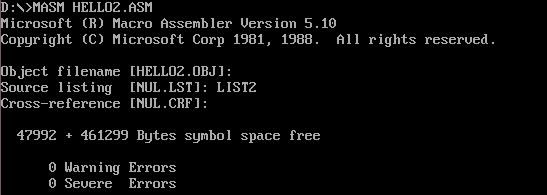
Выполнить программу в пошаговом режиме с фиксацией используемых регистров и ячеек памяти до и после выполнения каждой команды. Результаты представлены в табл.1.

Таблица 1.

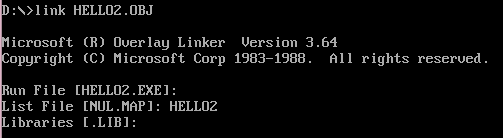
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес  Команды | Символический  код команды | 16-ричный  код команды | Содержимое регистров и ячеек памяти | |
| до выполнения . | После выполнения |
| 0010 | MOV AX,11AE | B8AE11 | (AX) = 0000  (IP) = 0010 | (AX) = 11AE  (IP) = 0013 |
| 0013 | MOV DS,AX | 8ED8 | (DS) = 119C  (IP) = 0013 | (DS) = 11AE  (IP) = 0015 |
| 0015 | MOV DX,0000 | BA0000 | (IP) = 0015 | (IP) = 0018 |
| 0018 | MOV AH,09 | B409 | (AX) = 11AE  (IP) = 0018 | (AX) = 09AE  (IP) = 001A |
| 001A | INT 21 | CD21 | (IP) = 001A | (IP) = 001C |
| 001C | MOV AH,4C | B44C | (AX) = 09AE  (IP) = 001C | (AX) = 4CAE  (IP) = 001E |
| 001E | INT 21 | CD21 | (AX) = 4CAE  (DS) = 119C  (IP) = 001E | (AX) = 0000  (DS) = 11AE  (IP) = 0010 |

**Часть 2.**

1. Просмотреть программу hello2.asm в режиме редактирования, используется редактор notepad++. Она формирует и выводит на экран приветствие пользователя с помощью процедуры WriteMsg. Разобраться в структуре и реализации каждого сегмента программы. Строку, в которой содержится приветствие, преобразовать в соответствии со своими личными данными.
2. Протранслировать программу с помощью строки masm hello2.asm c созданием объектного файла, HELLO2.OBJ, и файла диагностических сообщений – файла листинга, LIST2.lst.



1. Скомпоновать загрузочный модуль, HELLO2.EXE, с помощью строки link HELLO2.OBJ с созданием карты памяти HELLO2.MAP и исполняемого файла HELLO2.EXE.



1. Выполнить программу в автоматическом режиме путем набора строки HELLO2.EXE убедиться в корректности ее работы и зафиксировать результат выполнения в протоколе.



1. Запустить выполнение программы под управлением отладчика с помощью команды afd HELLO2.EXE.

Записать начальное содержимое сегментных регистров CS = 11B2,

DS = 119C, ES = 119C и SS = 11AC.

Выполнить программу в пошаговом режиме с фиксацией используемых регистров и ячеек памяти до и после выполнения каждой команды. Результаты представлены в табл.2.

Таблица 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес  Команды | Символический  код команды | 16-ричный  код команды | Содержимое регистров и ячеек памяти | |
| до выполнения | после выполнения |
| 0005 | push DS | 1E | (DS) = 119C  (IP) = 0005  (sp) = 0018  sTack +0 0000  +2 0000  +4 0000  +6 0000 | (DS) = 119C  (IP) = 0006  (sp) = 0016  sTack +0 119c  +2 0000  +4 0000  +6 0000 |
| 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 119c  +2 0000  +4 0000  +6 0000 | (AX) = 0000  (ip) = 0009  (SP) = 0014  sTack  +0 0000  +2 119C  +4 0000  +6 0000 |
| 0009 | mov AX,11AE | B8AE11 | (AX) = 0000  (ip) = 0009 | (AX) = 11AE  (ip) = 000C |
| 000C | mov DS,AX | 8ED8 | (DS) = 119C  (AX) = 11AE  (ip) = 000C | (DS) = 11AE  (AX) = 11AE  (ip) = 000E |
| 000E | mov DX, 0000 | BA0000 | (DX) = 0000  (IP) = 000E | (DX) = 0000  (IP) = 0011 |
| 0011 | call 0000 | E8ECFF | (ip) = 0011  (SP) = 0014  sTack +0 0000  +2 119C  +4 0000  +6 0000 | (ip) = 0000  (SP) = 0012  sTack +0 0014  +2 0000  +4 119C  +6 0000 |
| 0000 | mov AH,09 | B409 | (ax) = 11ae  (ip) = 0000 | (ax) = 09ae  (ip) = 0002 |
| 0002 | int 21 | CD21 | (ip) = 0002 | (ip) = 0004 |
| 0004 | ret | C3 | (ip) = 0004  (SP) = 0012  sTack +0 0014  +2 0000  +4 119C  +6 0000 | (ip) = 00014  (SP) = 0014  sTack +0 0000  +2 119C  +4 0000  +6 0000 |
| 0014 | mov DX, 0010 | BA1000 | (DX) = 0000  (IP) = 0014 | (DX) = 0010  (IP) = 0017 |
| 0017 | call 0000 | E8e6ff | (ip) = 0017  (SP) = 0014  sTack +0 0000  +2 119C  +4 0000  +6 0000 | (ip) = 0000  (SP) = 0012  sTack +0 001A  +2 0000  +4 119C  +6 0000 |
| 0000 | mov AH,09 | B409 | (aX) = 09AE  (ip) = 0000 | (aX) = 09AE  (ip) = 0002 |
| 0002 | int 21 | CD21 | (ip) = 0002 | (ip) = 0004 |
| 0004 | ret | C3 | (ip) = 001a  (SP) = 0012  sTack +0 001A  +2 0000  +4 119C  +6 0000 | (ip) = 0001A  (SP) = 0014  sTack  +0 0000  +2 119C  +4 0000  +6 0000 |
| 001A | ret Far | CB | (ip) = 001a  (SP) = 0014  sTack  +0 0000  +2 119C  +4 0000  +6 0000 | (ip) = 0000  (SP) = 0019  sTack  +0 0000  +2 0000  +4 0000  +6 0000 |
| 0000 | int 20 | CD20 | (IP) = 0000  (ax) = 09AE  (DS) = 11AE | (IP) = 0005  (ax) = 0000  (DS) = 119C |

## Вывод.

В ходе лабораторной работы были разобраны некоторые базовые концепции языка Ассемблера. Была произведена трансляция, линковка, выполнение и отладка программ на языке Ассемблер. Результатом выполнения лабораторной работы являются рабочие программы HELLO1.EXE и HELLO2.EXE.

# ПРИЛОЖЕНИЕ.

**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 'Вас приветствует ст.гр.1304 - Чернякова В.А.'**,**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

**LIST1.lst**

Microsoft **(**R**)** Macro Assembler Version 5**.**10 9**/**21**/**22 09**:**34**:**31

Page 1-1

**;** HELLO1**.**ASM **-** упрощенная версия учебной прогр

аммы лаб**.**раб**.** N1

**;** по дисциплине "Архитектура комп

ьютера"

**;** \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**\*\***\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**;** Назначение**:** Программа формирует и выводит на

экран приветствие

**;** пользователя с помощью функции ДО

С "Вывод строки"

**;** **(**номер 09 прерывание 21h**),** котора

я**:**

**;** **-** обеспечивает вывод на экран ст

роки символов**,**

**;** заканчивающейся знаком "$"**;**

**;** **-** требует задания в регистре ah

номера функции**=**09h**,**

**;** а в регистре dx **-** смещения а

дреса 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

0000 Greeting LABEL BYTE

**;** Текст приветствия

0000 82 A0 E1 20 AF E0 DB 'Вас приветствует ст.гр.1304 - Чернякова

В**.**А**.**',13,10,'$'

A8 A2 A5 E2 E1 E2

A2 E3 A5 E2 20 E1

E2 2E A3 E0 2E 31

33 30 34 20 2D 20

97 A5 E0 AD EF AA

AE A2 A0 20 82 2E

80 2E 0D 0A 24

**.**CODE **;** Начал

о сегмента кода

0000 B8 **----** R mov ax**,**@**data** **;** Загру

зка в DS адреса начала

0003 8E D8 mov ds**,**ax **;** сегме

нта данных

0005 BA 0000 R mov dx**,**OFFSET Greeting **;** Загру

зка в dx смещения

Microsoft **(**R**)** Macro Assembler Version 5**.**10 9**/**21**/**22 09**:**34**:**31

Page 1-2

**;** адрес

а текста приветствия

0008 DisplayGreeting**:**

0008 B4 09 mov ah**,**9 **;** # фун

кции ДОС печати строки

000A CD 21 int 21h **;** вывод

на экран приветствия

000C B4 4C mov ah**,**4ch **;** # фун

кции ДОС завершения программы

000E CD 21 int 21h **;** завер

шение программы и выход в ДОС

END

Microsoft **(**R**)** Macro Assembler Version 5**.**10 9**/**21**/**22 09**:**34**:**31

Symbols-1

Segments and Groups**:**

N a m **e** **Length** Align Combine Class

DGROUP **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** GROUP

\_**DATA** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** 002F 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

47984 **+** 461307 Bytes symbol space **free**

0 Warning Errors

0 Severe Errors

## 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 'Здравствуйте!'**,** 0AH**,** 0DH**,**EOFLine

GREETING DB 'Вас приветствует ст.гр.1304 - Чернякова В.А.$'

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

## LIST2.lst

Microsoft **(**R**)** Macro Assembler Version 5**.**10 9**/**21**/**22 21**:**27**:**21

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 символь

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 87 A4 E0 A0 A2 E1 HELLO DB 'Здравствуйте!'**,** 0AH**,** 0DH**,**EOFLine

E2 A2 E3 A9 E2 A5

21 0A 0D 24

0010 82 A0 E1 20 AF E0 GREETING DB 'Вас приветствует ст.гр.1304 - Чер

някова В**.**А**.**$'

A8 A2 A5 E2 E1 E2

A2 E3 A5 E2 20 E1

E2 2E A3 E0 2E 31

33 30 34 20 2D 20

97 A5 E0 AD EF AA

AE A2 A0 20 82 2E

80 2E 24

003D **DATA** ENDS

**;** Код программы

0000 CODE SEGMENT

ASSUME CS**:**CODE**,** DS**:DATA,** SS**:**AStack

**;** Процедура печати строки

0000 WriteMsg PROC NEAR

0000 B4 09 mov AH**,**9

0002 CD 21 int 21h **;** Вызов функции DOS по пре

рыванию

0004 C3 ret

0005 WriteMsg ENDP

Microsoft **(**R**)** Macro Assembler Version 5**.**10 9**/**21**/**22 21**:**27**:**21

Page 1-2

**;** 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

осстановления по

0008 50 push AX **;/** команде ret**,** завер

шающей 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**/**21**/**22 21**:**27**:**21

Symbols-1

Segments and Groups**:**

N a m **e** **Length** Align Combine Class

ASTACK **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** 0018 PARA STACK

CODE **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** 001B PARA NONE

**DATA** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** **.** 003D 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

47992 **+** 461299 Bytes symbol space **free**

0 Warning Errors

0 Severe Errors