**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: Освоение трансляции, выполнения и отладки программ на

языке Ассемблера процессора Intel X86

|  |  |  |
| --- | --- | --- |
| Студент гр. 7381 |  | Вологдин М.Д. |
| Преподаватель |  | Ефремов М.А. |

Санкт-Петербург

2018

**Цель работы**

Освоение трансляции, выполнения и отладки программ на языке Ассемблера процессора Intel X86

**Задание**

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.

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.

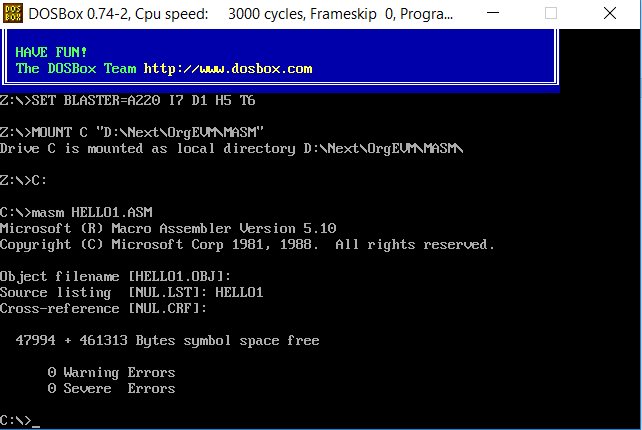
1. Разобраться в структуре и реализации каждого сегмента программы. Непонятные фрагменты прояснить у преподавателя. Строку-приветствие преобразовать в соответствии со своими личными данными.
2. Загрузить файл hello1.asm из каталога Tasks в каталог Masm.
3. Протранслировать программу с помощью строки> 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.
4. Скомпоновать загрузочный модуль с помощью строки> link hello1.obj с созданием карты памяти и исполняемого файла hello1.exe.
5. Выполнить программу в автоматическом режиме путем набора строки> hello1.exe убедиться в корректности ее работы и зафиксировать результат выполнения в протоколе.
6. 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

Для части 2 выполнить пункты 1 - 7 части 1 настоящего задания применительно к программе hello2.asm, приведенной в каталоге Tasks, 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.

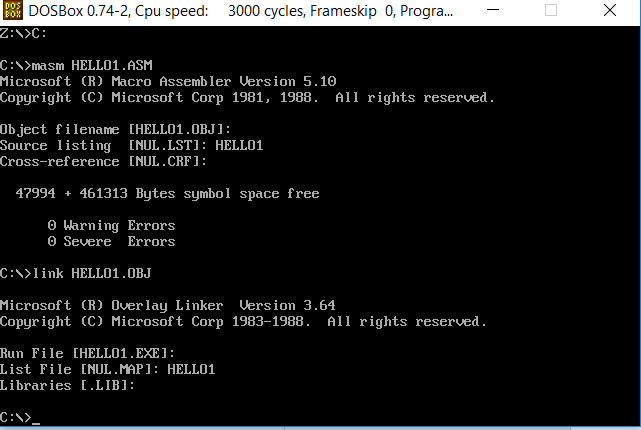
**Ход работы**

Часть 1

1. Смог загрузить файл hello1.asm из каталога Tasks в каталог Masm.
2. Протранслировал программу c созданием объектного файла hello1.obj и файла диагностических сообщений (файла листинга) hello1.lst.

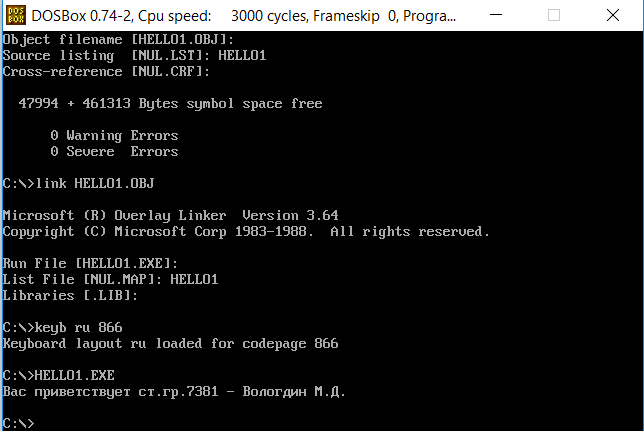


1. Слинковал загрузочный модуль (*hello1.exe*) c созданием загрузочного модуля (*hello1.exe*) и файла карты памяти (*hello1.map*). (рис. 3)



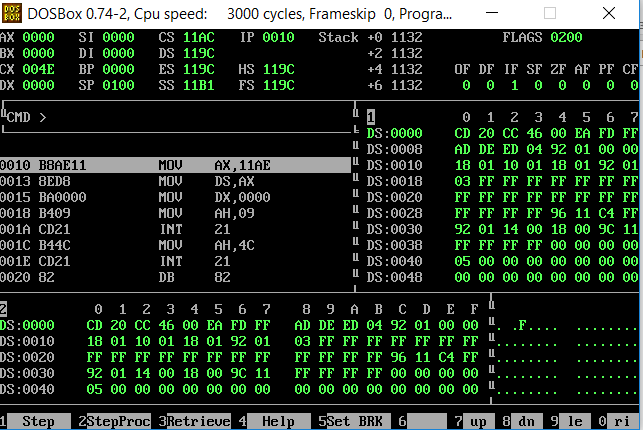
1. Выполнение программы

В режиме редактирования произведена замена строки, выводимой на экран на следующий текст «Вас приветствует ст. гр. 7381 – Вологдин М.Д.»



1. Запустил выполнение программы под управлением отладчика

Записал начальное содержимое сегментных регистров CS, DS, ES и SS. Выполнил программу в пошаговом режиме с фиксацией используемых регистров и ячеек памяти до и после выполнения каждой команды.

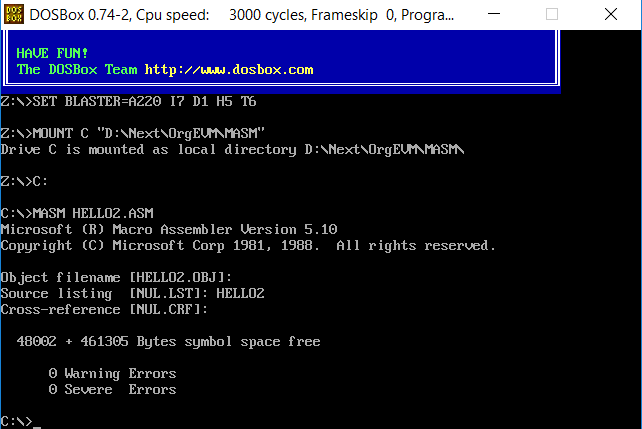


Результаты прогона программы под управлением отладчика.

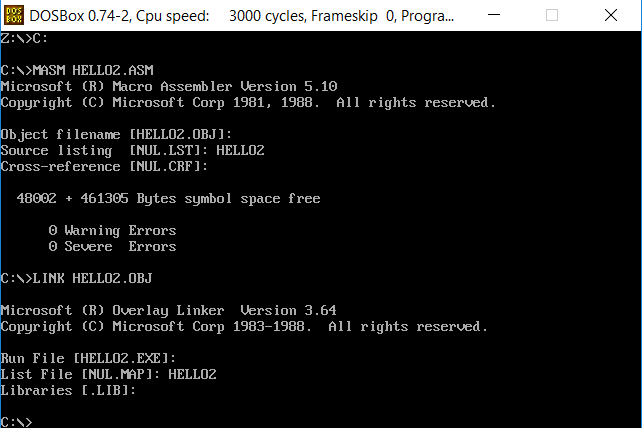
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес  Команды | Символический  код команды | 16-ричный  код команды | Содержимое регистров и ячеек памяти | |
| до выполнения . | После выполнения |
| 0010 | Mov AX, 11AE | B8AE11 | (AX) = 0000  (DS) = 119C  (IP) = 0010 | (AX) = 11AE  (DS) = 119C  (IP) = 0013 |
| 0013 | MOV DS, AX | 8ED8 | (AX) = 11AE  (DS) = 119C  (IP) = 0013 | (AX) = 11AE  (DS) = 11AE  (IP) = 0015 |
| 0015 | MOV DX, 0000 | BA0000 | (AX) = 11AE  (DS) = 11AE  (IP) = 0015 | (AX) = 11AE  (DS) = 11AE  (IP) = 0018 |
| 0018 | MOV AH, 09 | B409 | (AX) = 11AE  (DS) = 11AE  (IP) = 0018 | (AX) = 09AE  (DS) = 11AE  (IP) = 001A |
| 001A | INT 21 | CD21 | (AX) = 09AE  (DS) = 11AE  (IP) = 001A | (AX) = 09AE  (DS) = 11AE  (IP) = 001C |
| 001C | MOV AH, 4C | B44C | (AX) = 09AE  (DS) = 11AE  (IP) = 001C | (AX) = 4CAE  (DS) = 11AE  (IP) = 001E |
| 001E | INT 21 | CD21 | (AX) = 4CAE  (DS) = 11AE  (IP) = 001E | (AX) = 0000  (DS) = 119С  (IP) = 14A0 |

Часть 2

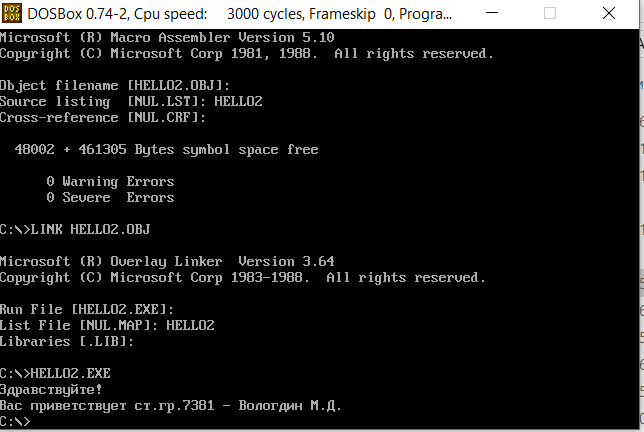
1. Просмотрел программу hello2.asm в режиме редактирования, разобрался в структуре и реализации каждого сегмента программы.
2. Протранслировал программу hello2.asm с помощью транслятора masm c созданием объектного файла и файла диагностических сообщений (файла листинга).



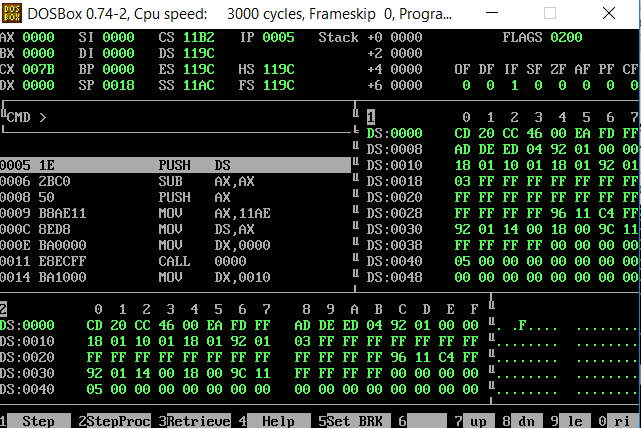
1. Слинковал загрузочный модуль hello2.exe с помощью компоновщика Link, используя команду:



1. Выполнил программу в автоматическом режиме и убедился в ее работоспособности



1. Выполнил программу hello2.exe в пошаговом режиме под управлением отладчика afd с фиксацией содержимого используемых регистров и ячеек памяти до и после выполнения команд.



1. Результаты прогона программы под управлением отладчика.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес  Команды | Символический  код команды | 16-ричный  код команды | Содержимое регистров и ячеек памяти | |
| до выполнения . | После выполнения |
| 0005 | PUSH DS | 1E | (AX) = 0000  (DS) = 119C  (IP) = 0005  (SP) = 0018  Stack+0 0000  +2 0000  +4 0000  +6 0000 | (AX) = 0000  (DS) = 119C  (IP) = 0006  (SP) = 0016  Stack+0 119C  +2 0000  +4 0000  +6 0000 |
| 0006 | SUB AX, AX | 2BCO | (AX) = 0000  (IP) = 0006 | (AX) = 0000  (IP) = 0008 |
| 0008 | PUSH AX | 50 | (IP) = 0008  (SP)=0016  Stack +0 119C  +2 0000  +4 0000  +6 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  (IP) = 000C | (DS) = 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) = 0014  (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)=0012  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 | CD 21 | (IP) = 0002 | (IP) = 0004 |
| 0004 | RET | C3 | (IP) = 0004  (SP)=0012  Stack +0 001A  +2 0000  +4 119C  +6 0000 | (IP) = 001A  (SP)=0014  Stack +0 0000  +2 119C  +4 0000  +6 0000 |
| 001A | RET Far | CB | (SP)=0014  (CS) = 11B2  (IP) = 001A  Stack +0 0000  +2 119C  +4 0000  +6 0000 | (SP)=0018  (CS) = 119C  (IP) = 0000  Stack +0 0000  +2 0000  +4 0000  +6 0000 |
| 0000 | INT 20 | CD20 | (AX) = 09AE  (DS) = 11AE  (IP) = 0000 | (AX) = 0000  (DS) = 119C  (IP) = 0005 |

**Вывод**

В ходе данной лабораторной работы мною была освоена трансляция, выполнение и отладка программ на языке Ассемблера процессора Intel x86. Результатом выполнения лабораторной работы являются программы hello1.exe и hello2.exe.

**Приложение**

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 'Вас приветствует ст.гр.7381 - Вологдин М.Д.',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 hello1.lst

Microsoft (R) Macro Assembler Version 5.10 9/12/18 09:59:02

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.7381 - Вологдин М

.Д.',13,10,'$'

A8 A2 A5 E2 E1 E2

A2 E3 A5 E2 20 E1

E2 2E A3 E0 2E 37

33 38 31 20 2D 20

82 AE AB AE A3 A4

A8 AD 20 8C 2E 84

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 9/12/18 09:59:02

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 9/12/18 09:59:02

Symbols-1

Segments and Groups:

N a m e Length Align Combine Class

DGROUP . . . . . . . . . . . . . GROUP

\_DATA . . . . . . . . . . . . 002E 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.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.7381 - Вологдин М.Д.$'

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 hello2.lst

Microsoft (R) Macro Assembler Version 5.10 9/12/18 10:15:49

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.7381 - Вол

огдин М.Д.$'

A8 A2 A5 E2 E1 E2

A2 E3 A5 E2 20 E1

E2 2E A3 E0 2E 37

33 38 31 20 2D 20

82 AE AB AE A3 A4

A8 AD 20 8C 2E 84

2E 24

003C 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 9/12/18 10:15:49

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 9/12/18 10:15:49

Symbols-1

Segments and Groups:

N a m e Length Align Combine Class

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

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

DATA . . . . . . . . . . . . . . 003C 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

48002 + 461305 Bytes symbol space free

0 Warning Errors

0 Severe Errors