# 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: Трансляция, отладка и выполнение программ на языке**

# Ассемблера

|  |  |  |
| --- | --- | --- |
| Студент гр. 1303 |  | Токун Г.С. |
| Преподаватель |  | Ефремов М.А. |

Санкт-Петербург 2022

# Цель работы.

Изучить программу на Ассемблере, выводящую строку на экран, её трансляцию, компоновку, отладку.

# 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.

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.

Просмотрена и изучена программа hello1.asm, изменена строка Greeting «You are greeted by Tokun G.S. st. from 1303».

Выполнена трансляция программы c cозданием объектного файла и файла листинга:

>masm hello1.asm Выполнена компоновка:

>link hello1.obj

Запущен исполняемый файл hello1.exe. Была выведена строка «You are greeted by Tokun G.S. st. from 1303».

Далее с помощью отладчика AFDPRO была осуществлена отладка программы и просмотр регистров процессора.

>afdpro hello1.exe

Была проведена аналогичная работа с программой hello2.asm. Строка Greeting была изменена на «'Student from 1303 – Tokun G.S.$»

Результаты отладки обеих программ представлены в табл. 1 и табл. 2. Файлы листинга см. в приложении А.

Таблица 1 – Результаты отладки программы hello1.exe

Начальные значения 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 | (IP) = 0013 |
|  |  |  | (AX) = 0000 | (AX) = 1A07 |
| 0013 | MOV DS, AX | 8ED8 | (IP) = 0013 | (IP) = 0015 |
|  |  |  | (AX) = 1A07 | (AX) = 1A07 |
|  |  |  | (DS) = 19F5 | (DS) = 1A07 |
| 0015 | MOV DX, 0000 | BA0000 | (IP) = 0015 | (IP) = 0018 |
|  |  |  | (DX) = 0000 | (DX) = 0000 |
| 0018 | MOV AH, 09 | B409 | (IP) = 0018 | (IP) = 001A |
|  |  |  | (AX) = 1A07 | (AX) = 0907 |
| 001A | INT 21 | CD21 | (IP) = 001A | (IP) = 001C |
| 001C | MOV AH, 4C | B44C | (IP) = 001C | (IP) = 001E |
|  |  |  | (AX) = 0907 | (AX) = 4C07 |
| 001E | INT 21 | CD21 | (IP) = 001E | (IP) = 0010 |

Таблица 2 – Результаты отладки программы hello2.exe

Начальные значения сегментных регистров: CS — 1A0A, 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 | (IP) = 0006 |
|  |  |  | (DS) = 19F5 | (DS) = 19F5 |
|  |  |  | (SP) = 0018 | (SP) = 0016 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | Stack | Stack |
| +0 0000 | +0 19F5 |
| +2 0000 | +2 0000 |
| +4 0000 | +4 0000 |
| +6 0000 | +6 0000 |
| 0006 | SUB AX, AX | 2BC0 | (IP) = 0000 | (IP) = 0008 |
|  |  |  | (AX) = 0000 | (AX) = 0000 |
| 0008 | PUSH AX | 50 | (IP) = 0008 | (IP) = 0009 |
|  |  |  | (AX) = 0000 | (AX) = 0000 |
|  |  |  | (SP) = 0016 | (SP) = 0014 |
|  |  |  | Stack | Stack |
|  |  |  | +0 19F5 | +0 0000 |
|  |  |  | +2 0000 | +2 19F5 |
|  |  |  | +4 0000 | +4 0000 |
|  |  |  | +6 0000 | +6 0000 |
| 0009 | MOV AX, 1A07 | B8071A | (IP) = 0009 | (IP) = 000C |
|  |  |  | (AX) = 0000 | (AX) = 1A07 |
| 000C | MOV DS, AX | 8ED8 | (IP) = 000C | (IP) = 000E |
|  |  |  | (AX) = 1A07 | (AX) = 1A07 |
|  |  |  | (DS) = 19F5 | (DS) = 1A07 |
| 000E | MOV DX, 0000 | BA0000 | (IP) = 000E | (IP) = 0011 |
|  |  |  | (DX) = 0000 | (DX) = 0000 |
| 0011 | CALL 0000 | E8ECFF | (IP) = 0011 | (IP) = 0000 |
|  |  |  | (SP) = 0014 | (SP) = 0012 |
|  |  |  | Stack | Stack |
|  |  |  | +0 0000 | +0 0014 |
|  |  |  | +2 19F5 | +2 0000 |
|  |  |  | +4 0000 | +4 19F5 |
|  |  |  | +6 0000 | +6 0000 |
| 0000 | MOV AH, 09 | B4 09 | (IP) = 0000 | (IP) = 0002 |
|  |  |  | (AX) = 1A07 | (AX) = 0907 |
| 0002 | INT 21 | CD21 | (IP) = 0002 | (IP) = 0004 |
| 0004 | RET | C3 | (IP) = 0004 | (IP) = 0014 |
|  |  |  | (SP) = 0012 | (SP) = 0014 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | Stack | Stack |
| +0 0014 | +0 0000 |
| +2 0000 | +2 19F5 |
| +4 19F5 | +4 0000 |
| +6 0000 | +6 0000 |
| 0014 | MOV DX, 0010 | BA 1000 | (IP) = 0014 | (IP) = 0017 |
|  |  |  | (DX) = 0000 | (DX) = 0010 |
| 0017 | CALL 0000 | E8E6FF | (IP) = 0017 | (IP) = 0000 |
|  |  |  | (SP) = 0014 | (SP) = 0012 |
|  |  |  | Stack | Stack |
|  |  |  | +0 0000 | +0 001A |
|  |  |  | +2 19F5 | +2 0000 |
|  |  |  | +4 0000 | +4 19F5 |
|  |  |  | +6 0000 | +6 0000 |
| 0000 | MOV AH, 09 | B409 | (IP) = 0000 | (IP) = 0002 |
|  |  |  | (AX) = 0907 | (AX) = 0907 |
| 0002 | INT 21 | CD21 | (IP) = 0002 | (IP) = 0004 |
| 0004 | RET | C3 | (IP) = 0004 | (IP) = 001A |
|  |  |  | (SP) = 0012 | (SP) = 0014 |
|  |  |  | Stack | Stack |
|  |  |  | +0 001A | +0 0000 |
|  |  |  | +2 0000 | +2 19F5 |
|  |  |  | +4 19F5 | +4 0000 |
|  |  |  | +6 0000 | +6 0000 |
| 001A | RET Far | CB | (IP) = 001A | (IP) = 0000 |
|  |  |  | (SP) = 0014 | (SP) = 0018 |
|  |  |  | (CS) = 1A0A | (CS) = 19F5 |
|  |  |  | Stack | Stack |
|  |  |  | +0 0000 | +0 0000 |
|  |  |  | +2 19F5 | +2 0000 |
|  |  |  | +4 0000 | +4 0000 |
|  |  |  | +6 0000 | +6 0000 |
| 0000 | INT 20 | CD20 | (IP) = 0000 | (IP) = 0005 |
|  |  |  | (AX) = 0907 | (AX) = 0000 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | (DX) = 0010 | (DX) = 0000 |
| (CS) = 19F5 | (CS) = 1A0A |

# Выводы.

Были изучены два варианта программы на Ассемблере, осуществляющей вывод строки с помощью прерывания 21h, их отладка.

# ПРИЛОЖЕНИЕ А

Название файла: HELLO1.LST

#Microsoft (R) Macro Assembler Version 5.10

9/14/22 19:32:26

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Page | | | | | | | | 1-1 |
| ; | 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 "$"; | | | | | | | |
| ; | - требует задания в регистре ah номера функции=09h, | | | | | | | |
| ; | а в регистре dx - смещения адреса выводимой | | | | | | | |
| ; | строки; | | | | | | | |
| ; | - использует регистр ax и не сохраняет его | | | | | | | |
| ; | содержимое. | | | | | | | |
| ; | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* | | | | | | | |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  DOSSEG | | | | |  |  | ; Задание сегментов под ДОС | | |
| .MODEL SMALL  .STACK 100h  .DATA  0000 Greeting | | | | | LABEL | BYTE | ; Модель памяти-SMALL(Малая)  ;Отвести под Стек 256 байт  ; Начало сегмента данных  ; Текст приветствия | | |
| 0000 | | 59 6F 75 | 20 61 72 | DB | 'You are greeted by Tokun G.S. st. from 1303',13,10,'$' | | | | |

|  |  |
| --- | --- |
| 65 20 67 72 65 | 65 |
| 74 65 64 20 62 | 79 |
| 20 4D 61 6E 74 | 73 |
| 65 76 61 20 54 | 2E |
| 4B 2E 20 73 74 | 2E |
| 20 66 72 6F 6D | 20 |
| 31 33 38 31 0D | 0A |
| 24 |  |

.CODE ; Начало сегмента кода

0000 B8 ---- R mov ax, @data ; Загрузка в DS адреса начала

0003 8E D8 mov ds, ax ; сегмента данных

0005 BA 0000 R mov dx, OFFSET Greeting ; Загр

#Microsoft (R) Macro Assembler Version 5.10

9/14/22 19:32:26

узка в dx смещения текста приветствия

0008 DisplayGreeting:

Page 1-2

; адреса

0008 B4 09 mov ah, 9 ; # функции ДОС печати строки

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 000A | CD 21 | int | 21h |  | ; вывод на экран приветствия |
| 000C | B4 4C | mov | ah, 4ch | ;# | функции ДОС завершения программы |
| 000E | CD 21 | int | 21h | ;завершение программы и выход в ДОС | |

END

#Microsoft (R) Macro Assembler Version 5.10

9/14/22 19:32:26

Symbols-1

Segments and Groups:

N a m e Length Align Combine Class DGROUP . . . . . . . . . . . . . GROUP

\_DATA . . . . . . . . . . . . 0031 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 . . . . . . . . . . . . .  @CODESIZE . . . . . . . . . . . | | TEXT  TEXT | \_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 |  | |

Название файла: HELLO2.LST

#Microsoft (R) Macro Assembler Version 5.10

9/14/22 19:33:47

Page 1-1

; HELLO2 - Учебная программа N2 лаб.раб.#1 по дисциплине "Архитектура компьютера"

; Программа использует процедуру для печати строки

;

; ТЕКСТ ПРОГРАММЫ

= 0024 EOFLine EQU '$' ; Определение символьной константы

; "Конец строки"

; Стек программы

ASSUME CS:CODE, SS:AStack

0000 AStack SEGMENT STACK

0000 000C[ DW 12 DUP('!') ; Отводится 12 слов памяти 0021

]

0018 AStack ENDS

; Данные программы

0000 DATA SEGMENT

; Директивы описания данных

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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 1303 – Tokun G.S.$'

74 20 66 72 6F 6D

20 31 33 38 31 20

2D 20 4D 61 6E 74

73 65 76 61 20 54

2E 4B 2E 24

0032 DATA ENDS

; Код программы

0000 CODE SEGMENT

; Процедура печати строки

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0000 | WriteMsg | | | PROC | NEAR | | |
| 0000 | B4 | 09 | mov AH,9 | | |  |  |
| 0002 | CD | 21 | int 21h ; Вызов функции DOS | | | по | прерыванию |
| 0004 | C3 |  | ret | | |  |  |
| 0005 |  |  | WriteMsg ENDP | | |  |  |

; Головная процедура

0005 Main PROC FAR

#Microsoft (R) Macro Assembler Version 5.10

9/14/22 19:33:47

Page 1-2

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0005 | 1E | push | | | | DS | | ;\ | Сохранение адреса  начала PSP в стеке |
| 0006 | 2B | C0 |  | sub AX,AX ;> для | | | | последующего восстановления по | |
| 0008  процедуру.  0009  сегментного | 50  B8 | ---- | R | push AX  mov AX,DATA | | | | ;/ команде ret, завершающей  ; Загрузка | |
| 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 приветствия. |
| 0014 | BA | 0010 | R | mov | DX, OFFSET | | GREETING | | ; Вывод на экран второй |
| 0017  001A | E8  CB | 0000 | R | call  ret | WriteMsg | | ; | | ; строки приветствия.  Выход в DOS по команде, |

; находящейся в 1-ом слове PSP.

001B Main ENDP

001B CODE ENDS

END Main

#Microsoft (R) Macro Assembler Version 5.10

9/14/22 19:33:47

Symbo

ls-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

47986 + 459274 Bytes symbol space free

0 Warning Errors

0 Severe Errors

# ПРИЛОЖЕНИЕ В 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 'You are greeted by Mantseva T.K. st. from 1381',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"

PSP.

; 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,EOFLine GREETING DB 'Student from 1381 - Mantseva T.K.$' 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

; 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

Main ENDP

CODE ENDS

END Main