

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. 9382

Кодуков А.В.

Преподаватель

Ефремов М.А .

Санкт-Петербург

2020

Задание.

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
```

с 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.

Выполнение работы:

Hello1

Трансляция (hello1.lst в приложении В)

```
C:\>masm hello1.asm
Microsoft (R) Macro Assembler Version 5.10
Copyright (C) Microsoft Corp 1981, 1988. All rights reserved.
```

```
Object filename [hello1.OBJ]:
Source listing [NUL.LST]: hello1.lst
Cross-reference [NUL.CRF]:
```

```
47994 + 461313 Bytes symbol space free
```

```
0 Warning Errors
0 Severe Errors
```

Компоновка

```
C:\>link hello1.obj
```

```
Microsoft (R) Overlay Linker Version 3.64
Copyright (C) Microsoft Corp 1983-1988. All rights reserved.
```

```
Run File [HELLO1.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
```

Запуск

```
C:\>hello1.exe
Вас приветствует ст.гр.9382 - Кодуков А.В.
```

Отладка

```
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	(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		

Hello2

Трансляция (hello2.lst в приложении В)

```
C:\>masm hello2.asm
Microsoft (R) Macro Assembler Version 5.10
Copyright (C) Microsoft Corp 1981, 1988. All rights reserved.
```

```
Object filename [hello2.OBJ]:
Source listing [NUL.LST]: hello2.lst
Cross-reference [NUL.CRF]:
hello2.asm(28): warning A4001: Extra characters on line
```

```
48002 + 461305 Bytes symbol space free
```

```
1 Warning Errors
0 Severe Errors
```

Компоновка

```
C:\>link hello2.obj
```

```
Microsoft (R) Overlay Linker Version 3.64
Copyright (C) Microsoft Corp 1983-1988. All rights reserved.
```

```
Run File [HELLO2.EXE]:
List File [NUL.MAP]:
Libraries [LIB]:
```

Запуск

```
C:\>hello2.exe
Hello Worlds!
Student from 9382 - Kodukov A.V.
```

Отладка

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	(DS) = 19F5 (SP) = 0016 (IP) = 0006 Stack: +0 19F5 +2 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	(AX) = 0000 (SP) = 0014 (IP) = 0009 Stack: +0 0000 +2 19F5
0009	MOV AX, 1A07	B8071A	(AX) = 0000	(AX) = 1A07

			(IP) = 0009	(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	INT21	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 19F5 +4 0000
0014	MOV DX, 0010	BA1000	(DX) = 0000 (IP) = 0014	(DX) = 0010 (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 19F5
0000	MOV AH, 09	B409	(AX) = 1A07 (IP) = 0000	(AX) = 0907 (IP) = 0002
0002	INT21	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	(CS) = 1A0B (IP) = 001A	(CS) = 19F5 (IP) = 0000
0000	INT 20	CD20	(IP) = 0000	

Вывод:

В ходе выполнения работы были скомпилированы и запущены программы по выводу текста на экран, а также получены результаты их отладки. Во второй программе в отличие от первой полностью определены сегменты данных, кода и стека, а так же вывод реализован с помощью собственной процедуры, что потребовало сохранения адреса начала PSP для возможности восстановления после завершения процедур. Были изучены вывод текста на экран в DOS (номер 09 прерывание 21h), определение сегментов и процедур.

ПРИЛОЖЕНИЕ А ИСХОДНЫЙ КОД ПРОГРАММЫ

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.7303 - Иванов И.И.',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

```

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 'Hello Worlds!', 0AH, 0DH, EOFLine
GREETING  DB 'Student from 4350 - $'
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

```

ПРИЛОЖЕНИЕ В

Файлы диагностических сообщений

Hello1.lst

Microsoft (R) Macro Assembler Version 5.10

9/16/20 21:28:48

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.9382 - Кодуков А.
В.',13,10,'$'
A8 A2 A5 E2 E1 E2
A2 E3 A5 E2 20 E1
E2 2E A3 E0 2E 39
33 38 32 20 2D 20
8A AE A4 E3 AA AE
A2 20 80 2E 82 2E
0D 0A 24

.CODE ; Начал
о сегмента кода
0000 B8 ---- R mov ax, @data ; Загр
узка в DS адреса начала
0003 8E D8 mov ds, ax ; сегм
ента данных
0005 BA 0000 R mov dx, OFFSET Greeting ; Загр
узка в dx смещения
Microsoft (R) Macro Assembler Version 5.10 9/16/20 21:28:48
Page 1-2

; адрес
а текста приветствия
0008 DisplayGreeting:
0008 B4 09 mov ah, 9 ; # фу
нкции ДОС печати строки
000A CD 21 int 21h ; вывод
на экран приветствия
000C B4 4C mov ah, 4ch ; # фу
нкции ДОС завершения программы
000E CD 21 int 21h ; завер
шение программы и выход в ДОС
END
Microsoft (R) Macro Assembler Version 5.10 9/16/20 21:28:48
Symbols-1

```

Segments and Groups:

	N a m e	Length	Align	Combine Class
DGROUP	GROUP		
_DATA	002D WORD PUBLIC		'DATA'
_STACK	0100 PARA STACK PUBLIC		'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

Hello2.lst

Microsoft (R) Macro Assembler Version 5.10

9/16/20 21:52:28
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 символъ
;          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  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 9382 - Kodukov A.V.$
;          ,
```

```

74 20 66 72 6F 6D
20 39 33 38 32 20
2D 20 4B 6F 64 75
6B 6F 76 20 41 2E
56 2E 24
0031          DATA          ENDS

; Код программы

0000          CODE          SEGMENT
                ASSUME CS:Code DS:DATA SS:AStack
hello2.asm(28): warning A4001: Extra characters on line
                ; 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

                ; Головная процедура
0005          Main      PROC  FAR
0005  1E                      push DS ;\ Сохранение адреса
Microsoft (R) Macro Assembler Version 5.10          9/16/20 21:52:28
                                                    Page      1-2

                                начала PSP в стеке
0006  2B C0                      sub  AX,AX ; > для последующего в
                                останова 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          9/16/20 21:52:28
                                                    Symbols-1

```

Segments and Groups:

N a m e	Length	Align	Combine Class
ASTACK	0018	PARA	STACK
CODE	001B	PARA	NONE
DATA	0031	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	

51 Source Lines
51 Total Lines
13 Symbols

48002 + 461305 Bytes symbol space free

1 Warning Errors
0 Severe Errors