

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

Студентка гр. 0383

Петровская Е.С.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

Цель работы.

Изучение трансляции, отладки и выполнения программ на языке Ассемблера.

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. 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.

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.

Были просмотрены и протранслированы программы hello1.asm и hello2.asm, скомпонованы их загрузочные модули, произведен запуск программ в автоматическом режиме. Работа обеих программ под управлением отладчика расписана в таблицах 1 и в соответственно.

```
C:\>hello1.exe
You're welcomed by the student of the 0383 group - Petrovskaya E.S.
```

Рис.1 — результат работы программы hello1.exe в автоматическом режиме

```
C:\>hello2.exe
Hello Worlds!
Student from 0383 - Petrovskaya Evgeniya
```

Рис.2 — результат работы программы hello2.exe в автоматическом режиме

Таблица 1 – Результаты прогона программы hello1 под управлением отладчика

Начальное содержимое сегментных регистров		(CS) = 1A05	(DS) = 19F5	(ES) = 19F5	(SS) = 1A0C
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) = 0807 (IP) = 001A	
001A	INT 21	CD21	(IP) = 001A	(IP) = 001C	
001C	MOV AH, 4C	B44C	(AX) = 0907	(AX) = 4C07	

			(IP) = 001C	(IP) = 001E
001E	INT 21	CD21	(IP) = 001E	

Таблица 2 – Результаты прогона программы hello2 под управлением отладчика

Начальное содержимое сегментных регистров		(CS) = 1A0B	(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	(SP) = 0018 (IP) = 0005	(SP) = 0016 (IP) = 0006	
0006	SUB AX, AX	2BC0	(AX) = 0000 (IP) = 0006	(AX) = 0000 (IP) = 0008	
0008	PUSH AX	50	(AX) = 0000 (SP) = 0016 (IP) = 0008	(AX) = 0000 (SP) = 0014 (IP) = 0009	
0009	MOV AX, 1A07	B8071A	(AX) = 0000 (IP) = 0009	(AX) = 1A07 (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	(AX) = 1A07 (IP) = 0011	(AX) = 0907 (IP) = 0014	
0014	MOV DX, 0010	BA1000	(DX) = 0000 (IP) = 0014	(DX) = 0010 (IP) = 0017	
0017	CALL 0000	E8E6FF	(IP) = 0017	(IP) = 001A	
001A	RET Far	CB	(SP) = 0014	(SP) = 0018	

			(CS) = 1A0B (IP) = 001A	(CS) = 19F5 (IP) = 0000
0000	INT 20	CD20	(IP) = 0000	

Выводы.

В ходе выполнения лабораторной работы произошло ознакомление с базовыми функциями эмулятора DOSBOX, особенностями трансляции, компоновки и запуска программ на языке Ассемблера, а также их отладки с помощью утилиты ADF PRO.

ПРИЛОЖЕНИЕ А

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`re welcomed by the student of the 0383 group -
Petrovskaya E.S.',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"

; 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 0383 - Petrovskaya Evgeniya$'
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 PSP.

Main      ENDP
CODE      ENDS
END Main

```

Название файла: hello1.lst

```

#Microsoft      (R)      Macro      Assembler      Version      5.10
9/15/21 07:51:36

```

Page

1-1

;
 ԹՀԾՅ ԹՃԾՃՆՆՁԾՀԾ»ԾՃԾԾԾ "ԾՁ(
 ՆՆԾՃՆ(ԾԾԾ°Ն(Ն)Ն(Ծ° Ծ°ԾՅԾԾԾՀՆՆՁՆ(ԾԾԾ°"
 ;

* * * * *

; ĐꞵĐỖĐ»ÑỈĐ·ĐỖĐ²Đ°Ñ(ĐµĐ»Ñ□ Ñ) ĐꞵĐỖĐ
 CEDỖÑ)ÑỈÑ□ Ñ(Ñ)ĐæĐ°Ñ<ĐŽĐŽ Đ□Đ□Đ; "Đ□Ñ)Đ²ĐỖĐŽ Ñ)Ñ
 (Ñ(ĐỖĐ°ĐŽ"

$$\begin{aligned} & - \partial \ddot{Y} \partial \pm \partial \mu \ddot{N} \rangle \partial \zeta \partial \mu \ddot{N} \rangle \partial \ddot{Z} \partial^2 \partial^\circ \partial \mu \ddot{N} (\partial^2 \ddot{N} \rangle \\ & {}^2 \partial \ddot{Y} \partial \ddot{Z} \partial \partial \partial^\circ \ddot{N} \rangle \partial^\circ \ddot{N} (\partial^\circ \partial \partial \ddot{N}) \ddot{N} (\ddot{N} \langle \partial \ddot{Y} \partial^\circ \partial \ddot{Z} \ddot{N} \rangle \partial \ddot{Z} \partial \partial \partial^2 \partial \ddot{Y} \partial \rangle \rangle \\ & \partial \ddot{Y} \partial^2, \end{aligned}$$

$$\begin{aligned} & - \tilde{N}(\tilde{N}(\Theta\mu\Theta\pm\tilde{N})\Theta\mu\tilde{N}(\Theta\cdot\Theta^\circ\Theta\check{\Theta}\Theta^\circ\Theta\Theta\check{\Theta}\tilde{N}\Theta \\ & \Theta^2\tilde{N}(\Theta\mu\Theta^3\Theta\check{\Theta}\tilde{N})\tilde{N}(\tilde{N}(\Theta\mu\text{ ah }\Theta\Theta\check{\Theta}\check{\Theta}\Theta\Theta\mu\tilde{N}(\Theta^\circ\tilde{N}(\tilde{N})\Theta\Theta\Theta^\circ\tilde{N}\langle\Theta \\ & \check{\Theta}\check{\Theta}=\Theta9h, \end{aligned}$$

$$; \quad \tilde{N})\tilde{N}(\tilde{N}(\mathfrak{D}\ddot{Y}\mathfrak{D}^{\circ}\mathfrak{D}\check{Z};$$

;

* * * * *

; $\mathbb{D}_\square \mathbb{D}^\circ \mathbb{D} \check{\mathbb{Z}} \mathbb{D}^\circ \mathbb{D} \mathfrak{d} \mathbb{Z} \mathbb{D} \mu \quad \tilde{\mathbb{N}}) \mathbb{D} \mu \mathbb{D}^3 \mathbb{D} \mathbb{E} \mathbb{D} \mu \mathbb{D} \mathfrak{d} \tilde{\mathbb{N}} (\mathbb{D} \check{\mathbb{Y}} \mathbb{D}^2 \quad \mathbb{D} \zeta \mathbb{D} \check{\mathbb{Y}} \mathbb{D} \check{\mathbb{Z}} \quad \mathbb{D}_\square \mathbb{D}_\square$

```

        D;
        .MODEL    SMALL
        ;  D D Y D Z D μ D » Ñ (  D Ć D ° D Ć Ñ D Ñ ( D Ž - SMALL ( D D D ° D » D ° Ñ D )
        .STACK    100h
        ;  D Ñ ( D² D μ Ñ ) Ñ ( D Ž  D Ć D Y D Ž  D ; Ñ ( D μ D °  256  D ± D ° D¹ Ñ (
        .DATA
        ;  D D D ° Ñ » D ° D » D Y  Ñ ) D μ D³ D Ć D μ D æ Ñ ( D °  D Ž D ° D æ D æ Ñ ) Ñ )
0000      Greeting    LABEL    BYTE
        ;  D Ć D μ D ° Ñ ) Ñ (  D Ć Ñ ( D Ž D² D μ Ñ ( Ñ ) Ñ ( D² D Ž Ñ D
0000      59 6F 75 60 72 65          DB 'You`re welcomed by the student of
the 03
                                83 group - Petrovskaya E.S.',13,10,'$'
20 77 65 6C 63 6F
6D 65 64 20 62 79
20 74 68 65 20 73
74 75 64 65 6E 74
20 6F 66 20 74 68
65 20 30 33 38 33
20 67 72 6F 75 70

```

1-2

20 2D 20 50 65 74
72 6F 76 73 6B 61
79 61 20 45 2E 53
2E 0D 0A 24

.CODE ;

□□°Ñ

»°«»Ÿ Ñ)µ³«µ«Ñ(° °ŸŽ°

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

□□°

³Ñ(Ñ)·°° °² DS °ŽÑ(µÑ)° «°Ñ»°«»°

0003 8E D8 mov ds, ax

; Ñ)µ

³«µ«Ñ(° Ž°««Ñ(Ñ)

0005 BA 0000 R mov dx, OFFSET Greeting ;

□□°

³Ñ(Ñ)·°° °² dx Ñ)«µÑ)µ«ŽÑ□

;

°ŽÑ

(µÑ)° Ñ(µ°Ñ)Ñ(° žÑ(Ž²µÑ(Ñ)Ñ(²ŽÑ□

0008 DisplayGreeting:

0008 B4 09 mov ah, 9

; # Ñ(

Ñ)«°Ñ«ŽŽ □□□; žµÑ»°Ñ(Ž Ñ)Ñ(Ÿ°Ž

000A CD 21 int

21h

; °²Ñ)°

²ŸŽ «° Ñ)°Ñ(°« žÑ(Ž²µÑ(Ñ)Ñ(²ŽÑ□

000C B4 4C mov ah, 4ch

; # Ñ(

Ñ)«°Ñ«ŽŽ □□□; ·°²µÑ(Ñ)µ«ŽÑ□ žÑ(Ÿ
³Ñ(°««Ñ)

000E CD 21

int

21h

; D·D°D

²DμÑ(Ñ(μDæDŽDμ DžÑ(DŸD³Ñ(D°DCEÑ) DŽ D²Ñ)Ñ)DŸD

Ž D² D□D□Dj

END

```
#Microsoft      (R)      Macro      Assembler      Version      5.10
9/15/21 07:51:36
```

Symbol

s-1

Segments and Groups:

	N a m e	Length	Align	Combine	Class
DGROUP	GROUP			
_DATA	0046	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 + 459266 Bytes symbol space free

0 Severe Errors

Название файла: hello2.lst

```
#Microsoft (R) Macro Assembler Version 5.10
9/15/21 07:47:40
```

Page

1-1

```
; HELLO2 - ÐŁŃ»ÐµÐ±ÐæÐ°Ń ÐŁŃ(ÐŸÐ³Ń(Ð°ÐŒÐŒÐ° N2
  Ð»Ð°Ð±.Ń(Ð°Ð±.#1 ÐŁŸ ÐŹŹŹŃ)Ń(ŒŹŒŁÐ»ÐŹÐæÐµ "Ð
  Ń(Ń)ÐŹŃ(ÐµÐ°Ń(Ń)Ń(Ð° Ð°ŸÐŒÐŁŃŃŃŃ(ÐµŃ(Ð°"
; ÐŃŃ(ÐŸÐ³Ń(Ð°ÐŒÐŒÐ° ÐŹŃ)ÐŁŸÐ»ŃŃ(Œ·Ń)Ð
µŃ( ÐŁŃ(ÐŸŃ(ÐµÐŹŃ)Ń(Ń) ÐŹŹ»Ń ÐŁÐµŃ)Ð°Ń(ÐŹ Ń)Ń(
Ń(ÐŸÐ°ÐŹ
;
; ÐŒÐŃÐŃÐŃÐŒ ÐŃÐ ÐŃÐŃÐ ÐŃÐŃÐŃÐŒ«
```

```

= 0024          EOFLine    EQU    '$'          ;

```

[illegible] $\tilde{N} \triangleright \tilde{N}$ $(\tilde{N}(\mathfrak{D}^{\vee}\mathfrak{D}^{\circ}\mathfrak{D}^{\vee}))$
$$; \mathfrak{D} \vdash \tilde{N}(\mathfrak{D} \mu \mathfrak{D}^{\circ} \quad \mathfrak{D} \zeta \tilde{N}(\mathfrak{D} \ddot{Y} \mathfrak{D}^3 \tilde{N}(\mathfrak{D}^{\circ} \mathfrak{D} \mathfrak{E} \mathfrak{D} \mathfrak{E} \tilde{N}))$$

```
ASSUME CS:CODE, SS:AStack
```

```

0000          AStack      SEGMENT  STACK
0000  0000C[                                DW  12  DUP('!')          ;

```

$$\mathfrak{D}_{\square}\tilde{N}(\mathfrak{D}^2\mathfrak{D}\ddot{Y}\mathfrak{D}\check{Z}\mathfrak{D}\check{Z}\tilde{N}(\tilde{N})\tilde{N}$$

□ 12 Ñ)Ð»ÐŸÐ² Ð¿Ð°ÐŸÑ□Ñ(Ðž

0021

1

```
0018             AStack     ENDS
```

$$; \mathfrak{D}_{\square} \mathfrak{D}^{\circ} \mathfrak{D}_{\mathfrak{ae}} \mathfrak{D}_{\mathfrak{ae}} \tilde{\mathfrak{N}}) \mathfrak{D}_{\mu} \mathfrak{D}_{\zeta} \tilde{\mathfrak{N}} (\mathfrak{D} \ddot{\mathfrak{Y}} \mathfrak{D}^3 \tilde{\mathfrak{N}} (\mathfrak{D}^{\circ} \mathfrak{D}_{\mathfrak{CE}} \mathfrak{D}_{\mathfrak{CE}} \tilde{\mathfrak{N}}))$$

0000	DATA	SEGMENT
------	------	---------

$$; \quad \mathfrak{D} \square \mathfrak{D} \check{Z} \check{N} (\mathfrak{D} \mu \mathfrak{D}^{\circ} \check{N} (\mathfrak{D} \check{Z} \mathfrak{D}^2 \check{N}) \quad \mathfrak{D} \check{Y} \mathfrak{D} \check{Z} \mathfrak{D} \check{N}) \mathfrak{D}^{\circ} \mathfrak{D} \mathfrak{a} \mathfrak{D} \check{Z} \check{N} \square \quad \mathfrak{D} \check{Z} \mathfrak{D}^{\circ} \mathfrak{D} \mathfrak{a} \mathfrak{D} \mathfrak{a} \check{N} \check{N})$$

```

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 0383 -
Petrovskaya Evgeniya$'
```

74 20 66 72 6F 6D

```

20 30 33 38 35 20
2D 20 50 65 74 72
6F 76 73 6B 61 79
61 20 45 76 67 65
6E 69 79 61 24

0039                                DATA            ENDS

                                ; 0x00000000 0x00000000 0x00000000 0x00000000

0000                                CODE            SEGMENT
                                ; 0x00000000 0x00000000 0x00000000 0x00000000
0000                                WriteMsg    PROC    NEAR
0000    B4 09                                mov     AH,9
0002    CD 21                                int      21h    ; 0x00000000 0x00000000

0x00000000 0x00000000 0x00000000 0x00000000
                                ; DOS 0x00000000 0x00000000 0x00000000 0x00000000

```



```

#Microsoft      (R)      Macro      Assembler      Version      5.10
9/15/21 07:47:40
s-1
Symbol

```

Segments and Groups:

N a m e	Length	Align	Combine Class
ASTACK	0018	PARA	STACK
CODE	001B	PARA	NONE
DATA	0039	PARA	NONE

Symbols:

N a m e	Type	Value	Attr
EOFLINE	NUMBER	0024	
GREETING	L BYTE	0010	DATA
HELLO	L BYTE	0000	DATA
0016 MAIN	F PROC	0005	CODE Length =
0005 WRITEMSG	N PROC	0000	CODE Length =
@CPU	TEXT	0101h	
@FILENAME	TEXT	hello2	
@VERSION	TEXT	510	

```

52 Source Lines
52 Total Lines
13 Symbols

```

47986 + 459271 Bytes symbol space free

```

0 Warning Errors
0 Severe Errors

```