

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: ТРАНСЛЯЦИИ, ОТЛАДКА И ВЫПОЛНЕНИЕ ПРОГРАММ НА ЯЗЫКЕ
АССЕМБЛЕРА

Студент гр. 1383

Преподаватель

Харитонов Н.М.

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Просмотреть и понять, как работает простая программа на Ассемблере.

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.

Таблица 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.

Был выполнен просмотр программы, были изучены команды, был отредактирован код в соответствии со своими данными

Затем, была выполнено транслирование программы, где выполнено создание объектного файла и файла листинга.

Была произведена линковка и получен файл .exe

Произведен запуск программы.

Произведена отладка программы.

Начальное положение регистров:

CS = 1A05; DS = 19F5; ES = 19F5; SS = 1A0C

Таблицы с пошаговым исполнением программы

Таблица 2 – исполнение файла hello1.asm

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	(DS) = 19F5 (IP) = 0013	(DS) = 1A07 (IP) = 0015
0015	MOV DX,0000	BA0000	(IP) = 0015	(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	(AX) = 4C07 (IP) = 001E	(AX) = 0000 (IP) = 0010

Таблица 3 – исполнение файла hello2.asm

Адрес Команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			до выполнения	после выполнения

0005	PUSH DS	1E	STACK +0 0000 +2 0000 +4 0000 +6 0000 (SP) = 0018 (IP) = 0005	STACK +0 19F5 +2 0000 +4 0000 +6 0000 (SP) = 0016 (IP) = 0006
0006	SUB AX,AX	2BC0	(IP) = 0006	(IP) = 0008
0008	PUSH AX	50	STACK +0 19F5 +2 0000 +4 0000 +6 0000 (IP) = 0008 (SP) = 0016	STACK +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 0009 (SP) = 0014
0009	MOV AX,1A07	B8071A	(IP) = 0009 (AX) = 0000	(IP) = 000C (AX) = 1A07
000C	MOV DS,AX	8ED8	(IP) = 000C (DS) = 19F5	(IP) = 000E (DS) = 1A07
000E	MOV DX,0000	BA0000	(IP) = 000E	(IP) = 0011
0011	CALL 0000	E8ECFF	STACK +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 0011 (SP) = 0014	STACK +0 0000 +2 0000 +4 19F5 +6 0000 (IP) = 0000 (SP) = 0012
0000	MOV AH,09	B409	(IP) = 0000 (AX) = 1A07	(IP) = 0002 (AX) = 0907
0002	INT 21	CD21	(IP) = 0002	(IP) = 0004
0004	RET	C3	STACK +0 0000	STACK +0 0000

			+2 0000 +4 19F5 +6 0000 (IP) = 0004 (SP) = 0012	+2 19F5 +4 0000 +6 0000 (IP) = 0014 (SP) = 0014
0014	MOV DX,0010	BA0100	(IP) = 0014 (DX) = 0000	(IP) = 0017 (DX) = 0010
0017	CALL 0000	E8E6FF	STACK +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 0014 (SP) = 0017	STACK +0 001A +2 19F5 +4 0000 +6 0000 (IP) = 0000 (SP) = 0012
0000	MOV AH, 09	B409	(IP) = 0000	(IP) = 0002
0002	INT 21	CD21	(IP) = 0002	(IP) = 0004
0004	RET	C3	STACK +0 001A +2 19F5 +4 0000 +6 0000 (IP) = 0004 (SP) = 0012	STACK +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 001A (SP) = 0014
001A	RET FAR	CB	STACK +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 001A (SP) = 0014 (CS) = 1A0A	STACK +0 0000 +2 0000 +4 0000 +6 0000 (IP) = 0000 (SP) = 0018 (CS) = 19F5
0000	INT 20	CD20	(AX) = 0907	(AX) = 0000

			(DX) = 0010 (CS) = 19F5 (DS) = 1A07 (IP) = 0000	(DX) = 0000 (CS) = 1A0A (DS) = 19F5 (IP) = 0005
--	--	--	--	--

Выводы.

Были изучены такие вещи, как запуск программ, их сборка и отладка.

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 'Вас приветствует ст.гр.1383 - Харитонов
Н.М.',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 1383 - Харитонов Н.М.$'
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

```

ПРИЛОЖЕНИЕ В

ЛИСТИНГИ

Название файла: hello1.lst

```

__Microsoft      (R)      Macro      Assembler      Version      5.10
9/28/22 01:27:50

```

Page 1-1

```

; HELLO1.ASM - CíPìCṪPsC%PμPSPSP°CṪ
PIPμCṪCÍPë

CṪ CíC†PμP±PSPsPN° PìCṪPsPìCṪP°PjPjC<
P»P°P±.CṪP

°P±. N1

; PìPs PṛPëCÍC†PëPìP»PëPSPμ

"PḥCṪ

C...PëC,PμPëC,CíCṪP° PëPsPjPìCḂCṪC,PμCṪP°"

;

*****

*****

; PḰP°P·PSP°C†PμPSPëPμ: PμCṪPsPìCṪP°PjPjP°

C,,Ps

CṪPjPëCṪCíPμC, Pë PIC<PIPsPṛPëC, PSP°

CḰPëCṪP°P

S PìCṪPëPIPμC,CÍC,PIPëPμ

; PìPsP»CḂP·PsPIP°C,PμP»CṪ CÍ

PìPsP

jPsC%CḂCṪ C,,CíPSPëC†PëPë P"PḥPŸ

"P' C<PIPsPṛ CÍC

,CṪPsPëPë"

; (PSPsPjPμCṪ 09

PìCṪPμCṪC<PIP°PSPë

Pμ 21h), PëPsC,PsCṪP°CṪ:

```

```

PIC< P
; - PsP±PμCÍPìPμC±PëPIP°PμC,
IPsPr PSP° CÍPεCßP°PS CÍC,CßPsPePë
CÍPëPjPIPsP»
PsPI,
;
P·P°PeP°PSC±PëPIP°CßC%PμPN°CÍCΠ
P·PSP°PePsPj "$";
; - C,CßPμP±CíPμC,
P·P°PrP°PSPëCΠ
PI CßPμPìPëCÍC,CßPμ ah PSPsPjPμCßP°
C„CíPSPεC†P
ëPë=09h,
; P° PI CßPμPìPëCÍC,CßPμ dx
- C
ÍPjPμC%PμPSPëCΠ P°PrCßPμCÍP°
PIC< PIPsPrPëPjPs
PN°
; CÍC,CßPsPePë;
; - PëCÍPìPsP»CßP·CíPμC,
CßPμPìPëC
ÍC,Cß ax Pë PSPμ CÍPSC...CßP°PSCΠPμC,
PμPìPs
; CÍPsPrPμCßP¶PëPjPsPμ.
;
*****
*****

DOSSEG
; P-P°PrP°PSPëPμ CÍPμPìPjPμPSC,PsPI PìPsPr
P"Ph
PŸ
.MODEL SMALL
; PßPsPrPμP»Cß PìP°PjCΠC,Pë-
SMALL (PßP°P»P°CΠ)

```

```

        .STACK 100h
; PhC,PIPµCfC,Pë PîPsPr PŸC,PµPe 256
P±P°PNcC,

        .DATA
; PkP°C±P°P»Ps CfPµPiPjPµPSC,P°
PrP°PSPSC<C...
0000 Greeting LABEL BYTE
; PŸPµPeCfC, PîCßPëPIPµC,CfC,PIPëCŲ
0000 D0 92 D0 B0 D1 81 DB 'P'P°Cf
PîCßPëPIPµC,CfC,PICíPµC, CfC,.PiC
ß.1383 - PŦP°CßPëC,PSPSPsPI
Pk.Pß.'',13,10,'$'
20 D0 BF D1 80 D0
B8 D0 B2 D0 B5 D1
82 D1 81 D1 82 D0
B2 D1 83 D0 B5 D1
82 20 D1 81 D1 82
2E D0 B3 D1 80 2E
31 33 38 33 20 2D
__Microsoft (R) Macro Assembler Version 5.10
9/28/22 01:27:50

Page 1-2

20 D0 A5 D0 B0 D1
80 D0 B8 D1 82 D0
BE D0 BD D0 BE D0
B2 20 D0 9D 2E D0
9C 2E 0D 0A 24

        .CODE
;
PkP°C
±P°P»Ps CfPµPiPjPµPSC,P° PePsPrP°
0000 B8 ---- R mov ax,
@data ; P-P°

```

```

                                PiCbCfP·PeP°      PI      DS      P°PrCbPµCfP°
PSP°C†P°P»P°
                                0003      8E D8                                mov                                ds,
ax                                ; CfPµ
                                PiPjPµPSC,P° PrP°PSPSC<C...
                                0005      BA 0000 R                                mov                                dx,      OFFSET
Greeting                        ; P-P°
                                PiCbCfP·PeP° PI dx CfPjPµC%PµPSPëCµ
                                                                ;
P°PrC
                                BµPµCfP°                                C,PµPeCfC,P°
PiCbPëPIPµC,CfC,PIPëCµ
                                0008                                DisplayGreeting:
                                0008      B4 09                                mov                                ah,
9                                ; # C,,
                                CfPSPeC†PëPë      P"PhPŸ      PiPµC†P°C,Pë
CfC,CBPsPePë
                                000A      CD 21                                int
21h                                ; PIC<P
                                IPSPr      PSP°                                CKPeCbP°PS
PiCbPëPIPµC,CfC,PIPëCµ
                                000C      B4 4C                                mov                                ah,
4ch                                ; # C,,
                                CfPSPeC†PëPë      P"PhPŸ      P·P°PIPµCbCëPµPSPëCµ
PiCbPs
                                PiCbP°PjPjC<
                                000E      CD 21                                int
21h                                ; P·P°P
                                IPµCbCëPµPSPëPµ      PiCbPsPiCbP°PjPjC<      Pë
PIC<C...PsP
                                r PI P"PhPŸ
                                END
__Microsoft      (R)      Macro      Assembler      Version      5.10
9/28/22 01:27:50

```

Symbols-1

Segments and Groups:

Class	N a m e	Length	Align	Combine
DGROUP	GROUP		
	_DATA	004D 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 Warning Errors

0 Severe Errors

Название файла: hello2.lst

Microsoft (R) Macro Assembler Version 5.10

9/28/22 01:28:02

Page

1-1

```
; HELLO2 - PJC+PμP±PSP°CЦ PιCБPsPιCБP°PjPjP° N2
P»P°P±.CБP°P±.#1 PιPs PrPëCГC†PëPιP»PëPSPμ "P
hCБC...PëC,PμPëC,CГCБP° PePsPjPιCБCБC,PμCБP°"
; PμCБPsPιCБP°PjPjP° PëCГPιPsP»CБP·CГP
μC, PιCБPsC†PμPrCГCБCГ PrP»CЦ PιPμC†P°C,Pë CГC,
CБPsPePë
;
; PŸP•PљPŸPŸ PμP PћP"P PђPњPњP«

= 0024 EOFLine EQU '$' ;
PћPιCБPμPrPμP»PμPSP
ëPμ CГPëPjPιPsP»CБPSPsPN PePsPSCГC,P°PSC,C<
; "PљPsPSPμC† CГC
,CБPsPePë"

; PŸC,PμPë PιCБPsPιCБP°PjPjC<

ASSUME CS:CODE, SS:AStack

0000 AStack SEGMENT STACK
0000 000C[ DW 12 DUP('!') ;
PћC,PιPsPrPëC,CГC
Ц 12 CГP»PsPI PιP°PjCЦC,Pë
0021 ]

0018 AStack ENDS

; P"P°PSPSC<Pμ PιCБPsPιCБP°PjPjC<

0000 DATA SEGMENT

; P"PëCБPμPëC,PëPIC< PsPιPëCГP°PSPëCЦ PrP°PSPS
C<C...

0000 48 65 6C 6C 6F 20 HELLO DB 'Hello Worlds!', 0AH,
ODH,EOFLine
57 6F 72 6C 64 73
21 0A 0D 24
0010 53 74 75 64 65 6E GREETING DB 'Student from 1383 -
PГP°CБPëC,PsP
```

```

SPsPI Pk.Pb.$'
74 20 66 72 6F 6D
20 31 33 38 33 20
2D 20 D0 A5 D0 B0
D1 80 D0 B8 D1 82
D0 BE D0 BD D0 BE
D0 B2 20 D0 9D 2E
D0 9C 2E 24

```

```

003E                                DATA        ENDS

```

```

; PpPsPr PíCpPsPíCpP°PjPjC<

```

```

0000                                CODE        SEGMENT
; PpCpPsC†PpPrCíCpP° PíPpC†P°C,Pë CíC,CpPsPePë
0000                                WriteMsg    PROC    NEAR
0000    B4 09                                mov     AH,9
0002    CD 21                                int      21h    ; P'C<P·PsPI

```

C„CíPSPeC†PëP

Microsoft (R) Macro Assembler Version 5.10

9/28/22 01:28:02

Page

1-2

```

; DOS PíPs PíCpPpCpC<PIP°PSPëCp
0004    C3                                ret
0005                                WriteMsg    ENDP

; P`PsP»PsPIPS°Cp PíCpPsC†PpPrCíCpP°
0005                                Main        PROC    FAR
0005    1E                                push    DS        ;\ PŸPsC...CpP°PSPpPSPë
Pp P°PrCpPpCíP° PSP°C†P°P»P° PSP PI CíC,PpPePp
0006    2B C0                                sub     AX,AX      ; > PrP»Cp
PíPsCíP»PpP
rCíCpC%PpPíPs PíPsCíCíC,P°PSPsPIP»PpPSPëCp PíPs
0008    50                                push    AX        ;/ PePsPjP°PSPPrPp ret
, P·P°PIPpCpCëP°CpC%PpPp PíCpPsC†PpPrCíCpCí.
0009    B8 ---- R                                mov     AX,DATA    ; P-
P°PíCpC
íP·PeP° CíPpPíPjPpPSC,PSPsPíPs
000C    8E D8                                mov     DS,AX      ;
CpPpPíPëC
íC,CpP° PrP°PSPSC<C...
000E    BA 0000 R                                mov     DX, OFFSET HELLO ;
P'C<PIPsP
r PSP° CíPeCpP°PS PíPpCpPIPsPp
0011    E8 0000 R                                call    WriteMsg   ;
CíC,CpPsP
ePë PíCpPëPIPpC,CíC,PIPëCp.
0014    BA 0010 R                                mov     DX, OFFSET GREETING ;
P'C<PIPsP
r PSP° CíPeCpP°PS PIC,PsCpPsPp

```

```

0017 E8 0000 R          call WriteMsg          ;
CFC,CBPSP
      ePë PïCBPëPIPµC,CFC,PIPëCµ.
001A CB                ret                      ; P'C<C...PsP
      r' PI DOS PïPs PePsPjP°PSPrPµ,
                                ; PSP°C...PsP
      rCµC%PµPN°CFCµ PI 1-PsPj CFP»PsPIPµ PSP.
001B                    Main          ENDP
001B                    CODE          ENDS
                                END Main

```

Microsoft (R) Macro Assembler Version 5.10
 9/28/22 01:28:02

Sym-

bols-1

Segments and Groups:

Class	N a m e	Length	Align	Combine
ASTACK		0018	PARA	STACK
CODE		001B	PARA	NONE
DATA		003E	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 + 459258 Bytes symbol space free

0 Warning Errors

0 Severe Errors