

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МОЭВМ

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Организация ЭВМ и систем»
Тема: Трансляция, отладка и выполнение программ на языке
Ассемблера

Студент гр. 9382

Субботин М.О.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

Цель работы.

Произвести трансляцию, отладку и выполнение программ на языке Ассемблер. Сравнить промежуточные и конечные результаты выполнения двух программ.

Основные теоретические положения.

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`).

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.

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.9382 – Субботин М.О.',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
```

Код программы hello2.asm :

```
; HELLO2 - "çŸi i "àøÈà ~ N2 « i.à i.#1 "ø n"áæ"«"Ÿ "Èàá"áŸ*ááà àø~iáŸà '
;
; àøÈà ~ "á"ø«iŸáŸà "àøŸŸ=ááà n«i "Ÿç á" ááàø"
;
; '...Ÿ' ' Žf €œœ»

EOFLine EQU '$' ; Ž"àŸŸ=Ÿ«Ÿ"Ÿ á"~çø«iøø àøáá áè
; "ŸøŸæ ááàø"

; áŸà "àøÈà ~è

AStack SEGMENT STACK
DW 12 DUP(?) ; Žáçø= ááí 12 á«øç " ~iá"
AStack ENDS

; „ èŸ "àøÈà ~è

DATA SEGMENT

; „ "àŸá"çè ø"~á "i n èá

HELLO DB 'Hello Worlds!', 0AH, 0DH, EOFLine
GREETING DB 'Student from 9382 – Subbotin M.O.$'
DATA ENDS

; Ÿø« "àøÈà ~è

CODE SEGMENT
ASSUME CS:Code, DS:DATA, SS:AStack
; àøŸŸ=áá "Ÿç á" ááàø"
WriteMsg PROC NEAR
mov AH,9
int 21h ; ,èŸøç ááæ" DOS "ø "àŸàèç "i
ret
WriteMsg ENDP

; Ÿø«øç i "àøŸŸ=áá
Main PROC FAR
push DS ;\ 'øáà Ÿ"Ÿ nàŸá ç « PSP ç ááŸáŸ
sub AX,AX ; > n«i "øá«Ÿ=náíéŸø çøááá øç«Ÿ"i "ø
push AX ;/ àø~ nŸ ret, Ÿ çŸàè íéŸø "àøŸŸ=ááà.
mov AX,DATA ; † ÈáŸŸ áŸŸ-ŸàøÈø
mov DS,AX ; àŸÈ"ááà n èá.
mov DX, OFFSET HELLO ; ,èçø= íàà "Ÿàçøø
call WriteMsg ; ááàø" "à"çŸáááç"i.
mov DX, OFFSET GREETING ; ,èçø= íàà çàøàøø
call WriteMsg ; ááàø" "à"çŸáááç"i.
ret ; ,èáø= ç DOS "ø àø~ nŸ,
; áø=íéŸøáí ç 1-ø~ á«øçŸ PSP.
Main ENDP
CODE ENDS
END Main
```

Экспериментальные результаты.

Программа hello1 выводила сообщение на экран:

“Вас приветствует ст.гр.9382 – Субботин М.О.”

Программа hello2 выводила на экран сообщение:

“Hello Worlds!”

“Student from 9382 – Subbotin M.O.”

Обработка результатов эксперимента.

В коде программы hello2 изначально была строка:

ASSUME CS:Code DS:DATA SS:AStack

Правильно же следовало бы написать:

ASSUME CS:Code, DS:DATA, SS:Astack

Эта ошибка сильно ничего не решает.

Также стоит заметить, что в программе hello1 сегмент CS имеет в памяти место 1A05 и SS: 1A0A. В то же время в программе hello2 CS:1A0A и SS:1A05 они поменяны местами.

Выводы.

Произвел трансляцию, отладку и выполнил программу. Сравнил промежуточные и конечные результаты выполнения программы.

ПРОТОКОЛ

Строку приветствие преобразовал в соответствии со своими личными данными.

Протранслировал программу с помощью строки

>masm hello1.asm

С созданием объектного файла(hello1.obj) и файла диагностических сообщений(hello1.lst). Синтаксических ошибок обнаружено не было.

Текст файла диагностических сообщений hello1.lst:

```
Microsoft (R) Macro Assembler Version 5.10          9/14/20 24:41:43
                                                    Page      1-1

; HELLO1.ASM — упрощенная версия учебной прог
; амы 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 "$";
; — требует задания в регистре ah
; номера функции=09h,
; а в регистре dx — смещения а
; дреса выводимой
; строки;
; — использует регистр ax и не
; сохраняет его
; содержимое.
; *****
; *****
; DOSSEG
; Задание сегментов под ДОС
; .MODEL SMALL
; Модель памяти—SMALL(Малая)
; .STACK 100h
; Отвести под стек 256 байт
; .DATA
; Начало сегмента данных
; Greeting LABEL BYTE
; Текст приветствия
0000 82 A0 E1 20 AF E0 DB 'Вас приветствует ст.гр.9382 — Субботин М
; ".','13,10','$'
A8 A2 A5 E2 E1 E2
A2 E3 A5 E2 20 E1
E2 2E A3 E0 2E 39
33 38 32 20 2D 20
91 E3 A1 A1 AE E2
A8 AD 20 8C 2E 8E
2E 0D 0A 24
; .CODE
; Начало
0000 B8 — R о сегмента кода 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          9/14/20 24:41:43
                                                    Page      1-2
; адрес
0008 а текста приветствия
0008 B4 09 DisplayGreeting: 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          9/14/20 24:41:43
                                                    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
```

Скомпилировал загрузочный модуль с помощью строки
>link hello1.obj

С созданием карты памяти(hello1.map) и исполняемого файла hello1.exe

Выполнил программу в автоматическом режиме путем набора строки
>hello1.exe

Убедился в корректности.

Результат работы :

“Вас приветствует ст.гр.9382 – Субботин М.О.”

Начальное содержимое сегментных регистров CS, DS, ES, SS:

CS: 1A05, DS: 19F5, ES: 19F5, SS: 1A0A

Протокол пошагового исполнения программы hello1 под управлением
отладчика:

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	AH=001A IP=0018	AH=0009 IP=001A
001A	INT 21	CD21	IP=001A	IP=001C
001C	MOV AH,4C	B44C	AH=0009 IP=001C	AH=004C IP=001E
001E	INT 21	CD21	IP=001E	IP=0010

Текст файла диагностических сообщений hello2.lst:

8

Скомпилировал загрузочный модуль с помощью строки

>link hello2.obj

С созданием карты памяти(hello2.map) и исполняемого файла hello2.exe

Выполнил программу в автоматическом режиме путем набора строки

>hello2.exe

Проверил корректность работы программы.

Результат работы :

“Hello worlds!”

“Student from 9382 – Subbotin M.O.”

Начальное содержимое сегментных регистров CS, DS, ES, SS:

CS: 1A0A

DS: 19F5

ES: 19F5

SS: 1A05

Протокол пошагового исполнения программы hello2 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	DS=19F5 IP=0005	AX=19F5 IP=0006
0006	SUB AX,AX	2BC0	AX=0000 IP=0006	AX=0000 IP=0008
0008	PUSH AX	50	AX=0000 IP=0008	AX=0000 IP=0009
0009	MOV DS,AX	8ED8	DS=19F5 AX=1A07	DS=1A07 AX=1A07

			IP=000C	IP=000E
000E	MOV DX,0000	BA0000	DX=0000 IP=000E	DX=0000 IP=0011
0011	CALL 0000	E8ECFF	IP=0011	IP=0000
0000	MOV AH,09	B409	AH=001A IP=0000	AH=0009 IP=0002
0002	INT 21	CD21	IP=0002	IP=0004
0004	RET	C3	IP=0004	IP=0014
0014	MOV DX,0010	BA1000	DX=0000 IP=0014	DX=0010 IP=0017
0017	CALL 0000	E8E6FF	IP=0017	IP=0000
0000	MOV AH,09	B409	AH=0009 IP=0000	AH=0009 IP=0002
0002	INT 21	CD21	IP=0002	IP=0004
0004	RET	C3	IP=0004	IP=001A
001A	RET FAR	CB	IP=001A	IP=0000
0000	INT 20	CD20	IP=0000	IP=0005