

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

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Операционные системы»
ТЕМА: Исследование структур загрузочного модулей

Студентка гр. 9383

Лихашва А.Д.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

Постановка задачи

Цель работы.

Изучить основные принципы трансляции, отладки и выполнения программ на языке Ассемблера. Исследование различий в структурах исходных текстов модулей типов .COM и .EXE, структур файлов загрузочных модулей и способов их загрузки в основную память.

Сведения о функциях и структурах данных.

В данной программе используются следующие функции и структуры данных:

Процедура	Описание
TETR_TO_HEX	Перевод десятичной цифры в код символа, который записывается в AL
BYTE_TO_HEX	Перевод значений байта в число 16-ой СС и его представление в виде двух символов
WRD_TO_HEX	Перевод слова в число 16-ой СС и представление его в виде четырех символов
BYTE_TO_DEC	Перевод значения байта в число 10-ой СС и представляет его в виду символов
PRINT_STRING	Вывод строки на экран
PRINT_PC_TYPE	Печать на экран тип ПК
PRINT_OS_VERSION	Печать на экран версии ОС, серийного номера OEM и серийного номера пользователя

Последовательность действий

В ходе работы программа выполняет следующие действия:

1. Процедура PRINT_PC_TYPE, которая выводит на экран тип ПК пользователя. Информация о типе ПК находится в предпоследнем байте ROM BIOS по адресу 0F000:0FFFEh. Значение этого байта определяет

2. Процедура PRINT_OS_VERSION, которая выводит на экран версию ОС, серийный номер OEM и серийный номер пользователя. В данной процедуре используется функция 30h прерывания 21h.
3. Завершение работы программы.

1 шаг:

```
C:\>lab1_com.exe

5 0
0x0Type0of my PC: PC
PC: PC
```

Рис. 1. - Пример работы "плохого" модуля .EXE
lab1 com.exe

```
C:\>lab1_com.com
Type of my PC: AT
Version MS DOS: 5.0
Serial number OEM: 0
User serial number: 000000
```

Рис. 2. - Пример работы "хорошего" .COM модуля
lab1 com.com

Был написан исходный текст .EXE модуля lab1_exe.asm, который выполняет те же функции, что и модуль в Шаге 1. Далее был получен «хороший» .EXE модуль lab1_exe.exe.

```
C:\>lab1_exe.exe
Type of my PC: AT
Version MS DOS: 5.0
Serial number OEM: 0
User serial number: 000000
```

Рис. 3. - Пример работы хорошего .EXE модуля lab1_exe.exe

3 шаг:

«Отличия исходных текстов .COM и .EXE программ»

1) Сколько сегментов должна содержать COM-программа?

COM-программа должна содержать только один сегмент, потому что данные программы и сам хранятся в одном сегменте, а стек автоматически устанавливается на последнюю ячейку сегмента.

2) EXE-программа?

EXE-программа должна содержать один или более сегментов. Количество сегментов зависит от выбранной модели памяти.

3) Какие директивы должны обязательно быть в тексте COM-программы?

В COM-программе обязательно должна быть директива ORG 100h. Данная директива устанавливает CS:IP на конец PSP, так как после загрузки все сегментные регистры (как и CS) указывают на начало PSP, а IP = 0, а это значит, что программа не будет выполняться, начиная с этого адреса. Именно эта директива смещает все относительные адреса на 100h байт.

В COM-программе обязательно должна быть директива ASSUME. Данная директива указывает ассемблеру с каким сегментом или группой сегментов связаны регистры.

В COM-программе обязательно должна быть директива END. Данная директива завершает работу программы на ассемблере.

4) Все ли форматы команд можно использовать в COM-программе?

Нельзя использовать команды вида: seg NAME, где NAME – название сегмента, так как в COM-программе отсутствует таблица настройки.

Шаг 4:

Шестнадцатеричное представление модуля .COM:

00000000	E9	78 02 54 79 70 65 20	6F 66 20 6D 79 20 50 43	ex.Type of my PC
00000010	3A 20 50 43 0D 0A 24 54	79 70 65 20 6F 66 20 6D	: PC..\$Type of m	
00000020	79 20 50 43 3A 20 50 43	2F 58 54 0D 0A 24 54 79	y PC: PC/XT..\$Ty	
00000030	70 65 20 6F 66 20 6D 79	20 50 43 3A 20 41 54 0D	pe of my PC: AT.	
00000040	0A 24 54 79 70 65 20 6F	66 20 6D 79 20 50 43 3A	..\$Type of my PC:	
00000050	20 50 53 32 20 6D 6F 64	65 6C 20 33 30 0D 0A 24	PS2 model 30..\$	
00000060	54 79 70 65 20 6F 66 20	6D 79 20 50 43 3A 20 50	Type of my PC: P	
00000070	53 32 20 6D 6F 64 65 6C	20 35 30 20 6F 72 20 36	S2 model 50 or 6	
00000080	30 0D 0A 24 54 79 70 65	20 6F 66 20 6D 79 20 50	0..\$Type of my P	
00000090	43 3A 20 50 53 32 20 6D	6F 64 65 6C 20 38 30 3A	C: PS2 model 80:	
000000A0	20 0D 0A 24 54 79 70 65	20 6F 66 20 6D 79 20 50	..\$Type of my P	
000000B0	43 3A 20 50 D0 A1 6A 72	0D 0A 24 54 79 70 65 20	C: Pllijr..\$Type	
000000C0	6F 66 20 6D 79 20 50 43	3A 20 50 43 20 43 6F 6E	of my PC: PC Con	
000000D0	76 65 72 74 69 62 6C 65	0D 0A 24 56 65 72 73 69	vertible..\$Versi	
000000E0	6F 6E 20 4D 53 20 44 4F	53 3A 20 20 2E 20 20 0D	on MS DOS: . .	
000000F0	0A 24 53 65 72 69 61 6C	20 6E 75 6D 62 65 72 20	..\$Serial number	
00000100	4F 45 4D 3A 20 20 20 20	20 20 20 0D 0A 24 55 73	OEM: ..\$Us	
00000110	65 72 20 73 65 72 69 61	6C 20 6E 75 6D 62 65 72	er serial number	
00000120	3A 20 20 20 20 20 20 20	0D 0A 24 45 72 72 6F 72	: ..\$Error	
00000130	21 20 54 68 65 20 62 79	74 65 20 76 61 6C 75 65	! The byte value	
00000140	20 64 6F 65 73 20 6E 6F	74 20 6D 61 74 63 68 20	does not match	
00000150	74 68 65 20 50 43 20 74	79 70 65 20 76 61 6C 75	the PC type valu	
00000160	65 73 24 0F 3C 09 76 02	04 07 04 30 C3 51 8A E0	es\$.<.v....0 Qèα	
00000170	E8 EF FF 86 C4 B1 04 D2	E8 E8 E6 FF 59 C3 53 8A	φn â-∑.тφμ Y Sè	
00000180	FC E8 E9 FF 88 25 4F 88	05 4F 8A C7 E8 DE FF 88	нφê ê%0ê.0è φ ê	
00000190	25 4F 88 05 5B C3 51 52	32 E4 33 D2 B9 0A 00 F7	%0ê.[QR2Σ3т ..≈	
000001A0	F1 80 CA 30 88 14 4E 33	D2 3D 0A 00 73 F1 3C 00	±Ç 0ê.N3т=..s±<.	
000001B0	74 04 0C 30 88 04 5A 59	C3 50 B4 09 CD 21 58 C3	t..0ê.ZY P .!= X	
000001C0	B8 00 F0 8E C0 26 A0 FE	FF 3C FF 74 26 3C FE 74	т.≡ÄL&á· < t&<·t	
000001D0	28 3C FB 74 24 3C FC 74	26 3C FA 74 28 3C FC 74	(<√t\$<^nt&<·t(<^nt	
000001E0	2A 3C F8 74 2C 3C FD 74	2E 3C F9 74 30 BA 2B 02	*<^°t,<^2t.<·t0 +.	
000001F0	EB 31 90 BA 03 01 EB 2B	90 BA 17 01 EB 25 90 BA	δ1É ..δ+É ..δ%É	
00000200	2E 01 EB 1F 90 BA 42 01	EB 19 90 BA 60 01 EB 13	..δ.É B.δ.É `.δ.	
00000210	90 BA 84 01 EB 0D 90 BA	A4 01 EB 07 90 BA BB 01	É ä.δ.É ñ.δ.É т.	
00000220	EB 01 90 E8 93 FF C3 50	53 51 52 56 57 B4 30 CD	δ.Éφô PSQRVW 0=	
00000230	21 BE DB 01 83 C6 10 E8	5C FF 8A C4 83 C6 03 E8	!д .â φ\ è-â φ.φ	
00000240	54 FF BA DB 01 E8 71 FF	BE F2 01 83 C6 15 8A C7	T .φq д≥.â φ.è	
00000250	E8 43 FF BA F2 01 E8 60	FF BF 0E 02 83 C7 19 8B	φC ≥.φ` т..â φ.ĩ	
00000260	C1 E8 1A FF 8A C3 E8 04	FF 83 EF 02 89 05 BA 0E	└φ. è φ. ân.ë. .	
00000270	02 E8 45 FF 5A 59 5B 58	5E 5F C3 E8 42 FF E8 A6	.φE ZY[X^_ φB φª	
00000280	FF 32 C0 B4 4C CD 21	+	2└└L=!	

Рис. 4. - Шестнадцатеричное представление модуля .COM:

Шестнадцатеричное представление плохого модуля .EXE:

00000000	4D 5A 87 01 03 00 00 00	20 00 00 00 FF FF 00 00	MZç..... ..
00000010	00 00 FC 24 00 01 00 00	1E 00 00 00 01 00 00 00	..n\$......
00000020	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000030	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000040	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000050	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

Рис. 5. - Шестнадцатеричное представление плохого модуля .EXE

000002F0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000300	E9 78 02 54 79 70 65 20	6F 66 20 6D 79 20 50 43	ex.Type of my PC
00000310	3A 20 50 43 0D 0A 24 54	79 70 65 20 6F 66 20 6D	: PC..\$Type of m
00000320	79 20 50 43 3A 20 50 43	2F 58 54 0D 0A 24 54 79	y PC: PC/XT..\$Ty
00000330	70 65 20 6F 66 20 6D 79	20 50 43 3A 20 41 54 0D	pe of my PC: AT.
00000340	0A 24 54 79 70 65 20 6F	66 20 6D 79 20 50 43 3A	.\$Type of my PC:
00000350	20 50 53 32 20 6D 6F 64	65 6C 20 33 30 0D 0A 24	PS2 model 30..\$
00000360	54 79 70 65 20 6F 66 20	6D 79 20 50 43 3A 20 50	Type of my PC: P
00000370	53 32 20 6D 6F 64 65 6C	20 35 30 20 6F 72 20 36	S2 model 50 or 6
00000380	30 0D 0A 24 54 79 70 65	20 6F 66 20 6D 79 20 50	0..\$Type of my P
00000390	43 3A 20 50 53 32 20 6D	6F 64 65 6C 20 38 30 3A	C: PS2 model 80:
000003A0	20 0D 0A 24 54 79 70 65	20 6F 66 20 6D 79 20 50	..\$Type of my P
000003B0	43 3A 20 50 D0 A1 6A 72	0D 0A 24 54 79 70 65 20	C: Pllijr..\$Type
000003C0	6F 66 20 6D 79 20 50 43	3A 20 50 43 20 43 6F 6E	of my PC: PC Con
000003D0	76 65 72 74 69 62 6C 65	0D 0A 24 56 65 72 73 69	vertible..\$Versi
000003E0	6F 6E 20 4D 53 20 44 4F	53 3A 20 20 2E 20 20 0D	on MS DOS: . .
000003F0	0A 24 53 65 72 69 61 6C	20 6E 75 6D 62 65 72 20	.\$Serial number
00000400	4F 45 4D 3A 20 20 20 20	20 20 20 0D 0A 24 55 73	OEM: ..\$Us
00000410	65 72 20 73 65 72 69 61	6C 20 6E 75 6D 62 65 72	er serial number
00000420	3A 20 20 20 20 20 20 20	0D 0A 24 45 72 72 6F 72	: ..\$Error
00000430	21 20 54 68 65 20 62 79	74 65 20 76 61 6C 75 65	! The byte value
00000440	20 64 6F 65 73 20 6E 6F	74 20 6D 61 74 63 68 20	does not match
00000450	74 68 65 20 50 43 20 74	79 70 65 20 76 61 6C 75	the PC type valu
00000460	65 73 24 0F 3C 09 76 02	04 07 04 30 C3 51 8A E0	es\$.<.v....0 Qèα
00000470	E8 EF FF 86 C4 B1 04 D2	E8 E8 E6 FF 59 C3 53 8A	φη à-тφφμ Y Sè
00000480	FC E8 E9 FF 88 25 4F 88	05 4F 8A C7 E8 DE FF 88	ηφθ è%Oè.0è φ è
00000490	25 4F 88 05 5B C3 51 52	32 E4 33 D2 B9 0A 00 F7	%Oè. [QR2Σ3T ..≈
000004A0	F1 80 CA 30 88 14 4E 33	D2 3D 0A 00 73 F1 3C 00	±çLlOè.N3T-.s±<.
000004B0	74 04 0C 30 88 04 5A 59	C3 50 B4 09 CD 21 58 C3	t..0è.ZY P .!= X
000004C0	B8 00 F0 8E C0 26 A0 FE	FF 3C FF 74 26 3C FE 74	η.≡ÄL&â· < t&<·t
000004D0	28 3C FB 74 24 3C FC 74	26 3C FA 74 28 3C FC 74	(<√t\$<^t&<·t(<^t
000004E0	2A 3C F8 74 2C 3C FD 74	2E 3C F9 74 30 BA 2B 02	*<^t,<^t.<·t0 +.
000004F0	EB 31 90 BA 03 01 EB 2B	90 BA 17 01 EB 25 90 BA	δ1É ..δ+É ..δ%É
00000500	2E 01 EB 1F 90 BA 42 01	EB 19 90 BA 60 01 EB 13	..δ.É B.δ.É `δ.
00000510	90 BA 84 01 EB 0D 90 BA	A4 01 EB 07 90 BA BB 01	É ä.δ.É ñ.δ.É η.
00000520	EB 01 90 E8 93 FF C3 50	53 51 52 56 57 B4 30 CD	δ.Éφδ PSQRVW 0=
00000530	21 BE DB 01 83 C6 10 E8	5C FF 8A C4 83 C6 03 E8	!= ■.â φ. è-â φ.φ
00000540	54 FF BA DB 01 E8 71 FF	BE F2 01 83 C6 15 8A C7	T ■.φq ≥.â φ.è
00000550	E8 43 FF BA F2 01 E8 60	FF BF 0E 02 83 C7 19 8B	φC ≥.φ· η..â φ.î
00000560	C1 E8 1A FF 8A C3 E8 04	FF 83 EF 02 89 05 BA 0E	└φ. è φ. ân.è. .
00000570	02 E8 45 FF 5A 59 5B 58	5E 5F C3 E8 42 FF E8 A6	·φE ZY[X^_ φB φ^
00000580	FF 32 C0 B4 4C CD 21	+	2└└L=!

Рис. 6. - Шестнадцатеричное представление плохого модуля .EXE

Шестнадцатеричное представление хорошего модуля .EXE:

00000000	4D 5A 8A 01 03 00 01 00	20 00 00 00 FF FF 00 00	MZè..... ..
00000010	00 01 D1 43 19 01 26 00	1E 00 00 00 01 00 1A 01	..TC..&.....
00000020	26 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	&.....

Рис. 7. - Шестнадцатеричное представление хорошего модуля .EXE

000002F0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000300	54 79 70 65 20 6F 66 20	6D 79 20 50 43 3A 20 50	Type of my PC: P
00000310	43 0D 0A 24 54 79 70 65	20 6F 66 20 6D 79 20 50	C..\$Type of my P
00000320	43 3A 20 50 43 2F 58 54	0D 0A 24 54 79 70 65 20	C: PC/XT..\$Type
00000330	6F 66 20 6D 79 20 50 43	3A 20 41 54 0D 0A 24 54	of my PC: AT..\$T
00000340	79 70 65 20 6F 66 20 6D	79 20 50 43 3A 20 50 53	ype of my PC: PS
00000350	32 20 6D 6F 64 65 6C 20	33 30 0D 0A 24 54 79 70	2 model 30..\$Typ
00000360	65 20 6F 66 20 6D 79 20	50 43 3A 20 50 53 32 20	e of my PC: PS2
00000370	6D 6F 64 65 6C 20 35 30	20 6F 72 20 36 30 0D 0A	model 50 or 60..
00000380	24 54 79 70 65 20 6F 66	20 6D 79 20 50 43 3A 20	\$Type of my PC:
00000390	50 53 32 20 6D 6F 64 65	6C 20 38 30 3A 20 0D 0A	PS2 model 80: ..
000003A0	24 54 79 70 65 20 6F 66	20 6D 79 20 50 43 3A 20	\$Type of my PC:
000003B0	50 D0 A1 6A 72 0D 0A 24	54 79 70 65 20 6F 66 20	Pl-ijr..\$Type of
000003C0	6D 79 20 50 43 3A 20 50	43 20 43 6F 6E 76 65 72	my PC: PC Conver
000003D0	74 69 62 6C 65 0D 0A 24	56 65 72 73 69 6F 6E 20	tible..\$Version
000003E0	4D 53 20 44 4F 53 3A 20	20 2E 20 20 0D 0A 24 53	MS DOS: . . . \$S
000003F0	65 72 69 61 6C 20 6E 75	6D 62 65 72 20 4F 45 4D	erial number OEM
00000400	3A 20 20 20 20 20 20 20	0D 0A 24 55 73 65 72 20	: ..\$User
00000410	73 65 72 69 61 6C 20 6E	75 6D 62 65 72 3A 20 20	serial number:
00000420	20 20 20 20 20 0D 0A 24	45 72 72 6F 72 21 20 54	..\$Error! T
00000430	68 65 20 62 79 74 65 20	76 61 6C 75 65 20 64 6F	he byte value do
00000440	65 73 20 6E 6F 74 20 6D	61 74 63 68 20 74 68 65	es not match the
00000450	20 50 43 20 74 79 70 65	20 76 61 6C 75 65 73 00	PC type values.
00000460	24 0F 3C 09 76 02 04 07	04 30 C3 51 8A E0 E8 EF	\$.<.v....0 Qèαφη
00000470	FF 86 C4 B1 04 D2 E8 E8	E6 FF 59 C3 53 8A FC E8	â-µ.τφμ Y Sè^nφ
00000480	E9 FF 88 25 4F 88 05 4F	8A C7 E8 DE FF 88 25 4F	ø è%0è.0è φ è%0
00000490	88 05 5B C3 51 52 32 E4	33 D2 B9 0A 00 F7 F1 80	è. [f QR2Σ3T ..≈±Ç
000004A0	CA 30 88 14 4E 33 D2 3D	0A 00 73 F1 3C 00 74 04	±0è.N3T=..s±<.t.
000004B0	0C 30 88 04 5A 59 C3 50	B4 09 CD 21 58 C3 B8 00	.0è.ZY P .!= X η.
000004C0	F0 8E C0 26 A0 FE FF 3C	FF 74 26 3C FE 74 28 3C	≡Äl&a. < t&<•t(<
000004D0	FB 74 24 3C FC 74 26 3C	FA 74 28 3C FC 74 2A 3C	√t\$<^nt&<•t(<^nt*<
000004E0	F8 74 2C 3C FD 74 2E 3C	F9 74 30 BA 28 01 EB 31	°t,<^2t.<•t0 (.δ1
000004F0	90 BA 00 00 EB 2B 90 BA	14 00 EB 25 90 BA 2B 00	É ..δ+É ..δ%É +.
00000500	EB 1F 90 BA 3F 00 EB 19	90 BA 5D 00 EB 13 90 BA	δ.É ?.δ.É].δ.É
00000510	81 00 EB 0D 90 BA A1 00	EB 07 90 BA B8 00 EB 01	ü.δ.É í.δ.É η.δ.
00000520	90 E8 93 FF C3 50 53 51	52 56 57 B4 30 CD 21 BE	Éφô PSQRVW 0= J
00000530	D8 00 83 C6 10 E8 5C FF	8A C4 83 C6 03 E8 54 FF	±.â φ.φ\ è-â φ.φT
00000540	BA D8 00 E8 71 FF BE EF	00 83 C6 15 8A C7 E8 43	±.φq η.â φ.è φC
00000550	FF BA EF 00 E8 60 FF BF	0B 01 83 C7 19 8B C1 E8	η.φ` η..â φ.î±φ
00000560	1A FF 8A C3 E8 04 FF 83	EF 02 89 05 BA 0B 01 E8	. è φ. ân.è. ..φ
00000570	45 FF 5A 59 5B 58 5E 5F	C3 B8 10 00 8E D8 E8 3D	E ZY[X^_ η..Ä±φ=
00000580	FF E8 A1 FF 32 C0 B4 4C	CD 21	φí 2 L=!

Рис. 8. - Шестнадцатеричное представление хорошего модуля .EXE

«Отличие форматов файлов COM и EXE модулей»

1) Какова структура файла COM? С какого адреса располагается код?

В данном файле код, данные и стек находятся в одном сегменте. Код и данные начинаются с адреса 0h (См. Рис. 4).

2) Какова структура файла «плохого» EXE? С какого адреса располагается код? Что располагается с адреса 0?

В «плохом» EXE файле код, данные и стек находятся в одном сегменте. Код и данные начинаются с адреса 300h. С адреса 0h находится управляющая информация загрузчика, которая содержит заголовок и таблицу настроек. (См. Рис. 5-6)

3) Какова структура «хорошего» EXE? Чем он отличается от файла «плохого» EXE?

У «хорошего» EXE код, данные и стек находятся в разных сегментах, а в «плохом» - в одном сегменте. С адреса 0 в «хорошем» EXE располагается валидная таблица настроек, в отличие от «плохого» EXE. У «хорошего» EXE выделяется память под стек между PSP и кодом.

Шаг 5:

«Загрузка COM модуля в основную память»

1) Какой формат загрузки модуля COM? С какого адреса располагается код?

В начале определяется сегментный адрес участка ОП, способного вместить загрузку программы, затем создается блок памяти для PSP и программы. После считывания COM-файл помещается в память с 100h. После сегментные регистры устанавливаются в начало PSP. SP устанавливается в конец PSP, 0000h помещается в стек, а в IP записывается 100h.

Код располагается с адреса 100h.

AX 0000	SI 0000	CS 19F5	IP 0100	Stack +0 0000	Flags 7202
BX 0000	DI 0000	DS 19F5		+2 20CD	
CX 0287	BP 0000	ES 19F5	HS 19F5	+4 9FFF	OF DF IF SF ZF AF PF CF
DX 0000	SP FFFE	SS 19F5	FS 19F5	+6 EA00	0 0 1 0 0 0 0 0

CMD >				<div>101234567</div> <div>DS:0000CD 20 FF 9F 00 EA F0 FE</div> <div>DS:0008AD DE 1B 05 C5 06 00 00</div> <div>DS:001018 01 10 01 18 01 92 01</div> <div>DS:001801 01 01 00 02 FF FF FF</div> <div>DS:0020FF FF FF FF FF FF FF FF</div> <div>DS:0028FF FF FF FF EB 19 C0 11</div> <div>DS:0030A2 01 14 00 18 00 F5 19</div> <div>DS:0038FF FF FF FF 00 00 00 00</div> <div>DS:004005 00 00 00 00 00 00 00</div> <div>DS:004800 00 00 00 00 00 00 00</div>															
-------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

2	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
DS:0000	CD	20	FF	9F	00	EA	F0	FE	AD	DE	1B	05	C5	06	00	00	= f.Ω≡ i ..†...
DS:0010	18	01	10	01	18	01	92	01	01	01	01	00	02	FF	FF	FFff.
DS:0020	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	EB	19	C0	11δ.L.
DS:0030	A2	01	14	00	18	00	F5	19	FF	FF	FF	FF	00	00	00	00	ó.....J.
DS:0040	05	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Рис. 9. - .COM в отладчике

2) Что располагается с адреса 0?

С адреса 0 располагается PSP размером в 100h байт.

3) Какие значения имеют сегментные регистры? На какие области памяти они указывают?

Регистры DS, ES, CS, SS указывают на начало блока PSP.

4) Как определяется стек? Какую область он занимает? Какие адреса?

Стек генерируется автоматически. Регистр SS указывает на начало блока PSP, а SP на конец стека. Стек расположен между адресами SS:0000h – SS:FFFFh и заполняется с конца модуля в сторону уменьшения адресов.

Шаг 6:

«Загрузка «хорошего» EXE модуля в основную память»

1) Как загружается «хороший» EXE? Какие значения имеют сегментные регистры?

Данный EXE загружается со считыванием информации заголовка EXE, выполняется перемещение адресов сегментов, ES и DS устанавливаются в на-

чало PSP, SS – на начало сегмента стека, а CS – на начало сегмента команд. В IP загружается смещение точки входа в программу.

AX 0000	SI 0000	CS 1A2B	IP 0119	Stack +0 7954	Flags 7202
BX 0000	DI 0000	DS 19F5		+2 6570	
CX 038A	BP 0000	ES 19F5	HS 19F5	+4 6F20	OF DF IF SF ZF AF PF CF
DX 0000	SP 0100	SS 1A05	FS 19F5	+6 2066	0 0 1 0 0 0 0 0

CMD >				<table><tr><td>1</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr><tr><td>DS:0000</td><td>CD</td><td>20</td><td>FF</td><td>9F</td><td>00</td><td>EA</td><td>F0</td><td>FE</td></tr><tr><td>DS:0008</td><td>AD</td><td>DE</td><td>1B</td><td>05</td><td>C5</td><td>06</td><td>00</td><td>00</td></tr><tr><td>DS:0010</td><td>18</td><td>01</td><td>10</td><td>01</td><td>18</td><td>01</td><td>92</td><td>01</td></tr><tr><td>DS:0018</td><td>01</td><td>01</td><td>01</td><td>00</td><td>02</td><td>FF</td><td>FF</td><td>FF</td></tr><tr><td>DS:0020</td><td>FF</td><td>FF</td><td>FF</td><td>FF</td><td>FF</td><td>FF</td><td>FF</td><td>FF</td></tr><tr><td>DS:0028</td><td>FF</td><td>FF</td><td>FF</td><td>FF</td><td>EB</td><td>19</td><td>C0</td><td>11</td></tr><tr><td>DS:0030</td><td>A2</td><td>01</td><td>14</td><td>00</td><td>18</td><td>00</td><td>F5</td><td>19</td></tr><tr><td>DS:0038</td><td>FF</td><td>FF</td><td>FF</td><td>FF</td><td>00</td><td>00</td><td>00</td><td>00</td></tr><tr><td>DS:0040</td><td>05</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td></tr><tr><td>DS:0048</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td></tr></table>																1	0	1	2	3	4	5	6	7	DS:0000	CD	20	FF	9F	00	EA	F0	FE	DS:0008	AD	DE	1B	05	C5	06	00	00	DS:0010	18	01	10	01	18	01	92	01	DS:0018	01	01	01	00	02	FF	FF	FF	DS:0020	FF	FF	FF	FF	FF	FF	FF	FF	DS:0028	FF	FF	FF	FF	EB	19	C0	11	DS:0030	A2	01	14	00	18	00	F5	19	DS:0038	FF	FF	FF	FF	00	00	00	00	DS:0040	05	00	00	00	00	00	00	00	DS:0048	00	00	00	00	00	00	00	00
1	0	1	2	3	4	5	6	7																																																																																																														
DS:0000	CD	20	FF	9F	00	EA	F0	FE																																																																																																														
DS:0008	AD	DE	1B	05	C5	06	00	00																																																																																																														
DS:0010	18	01	10	01	18	01	92	01																																																																																																														
DS:0018	01	01	01	00	02	FF	FF	FF																																																																																																														
DS:0020	FF	FF	FF	FF	FF	FF	FF	FF																																																																																																														
DS:0028	FF	FF	FF	FF	EB	19	C0	11																																																																																																														
DS:0030	A2	01	14	00	18	00	F5	19																																																																																																														
DS:0038	FF	FF	FF	FF	00	00	00	00																																																																																																														
DS:0040	05	00	00	00	00	00	00	00																																																																																																														
DS:0048	00	00	00	00	00	00	00	00																																																																																																														

2	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
DS:0000	CD	20	FF	9F	00	EA	F0	FE	AD	DE	1B	05	C5	06	00	00
DS:0010	18	01	10	01	18	01	92	01	01	01	01	00	02	FF	FF	FF
DS:0020	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	EB	19	C0	11
DS:0030	A2	01	14	00	18	00	F5	19	FF	FF	FF	FF	00	00	00	00
DS:0040	05	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

= f.Ω≡	↓ .†...
.....ff.
	δ.L.
ó.....J.
.....

Рис. 10. - .EXE в отладчике

2) На что указывают регистры DS и ES?

ES и DS указывают на начало сегмента PSP.

3) Как определяется стек?

Стек определяется на основе директивы .stack с указанием размера стека. SS указывает на начало сегмента стека, а SP указывает на конец.

4) Как определяется точка входа?

Точка входа определяется параметром после директивы END.

Заключение.

В результате выполнения лабораторной работы были изучены структурные отличия .COM и .EXE модулей и получены навыки работы с отладчиком TD.EXE.