

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 загрузочных модулей

Студент гр. 0381

Просекин Т.А.

Преподаватель

Ефремов М. А.

Санкт-Петербург

2022

Цель работы.

Исследование различий в структурах исходных текстов модулей типов .COM и .EXE, структур файлов загрузочных модулей и способов их загрузки в основную память.

Постановка задачи.

Требуется написать текст исходного .COM модуля, который определяет тип РС и версию системы. Ассемблерная программа должна читать содержимое предпоследнего байта ROM BIOS, по таблице, сравнивая коды, определять тип РС и выводить строку с названием модели. Если код не совпадает ни с одним значением, то двоичный код переводиться в символьную строку, содержащую запись шестнадцатеричного числа и выводиться на экран в виде соответствующего сообщения. Затем определяется версия системы. Ассемблерная программа должна по значениям регистров AL и AH формировать текстовую строку в формате xx.yy, где xx – номер основной версии, а yy - номер модификации в десятичной системе счисления, формировать строки с серийным номером OEM (Original Equipment Manufacturer) и серийным номером пользователя. Полученные строки выводятся на экран. Далее необходимо отладить полученный исходный модуль и получить «хороший» .COM модуль, а также необходимо построить «плохой» .EXE, полученный из исходного текста для .COM модуля. Затем нужно написать текст «хорошего» .EXE модуля, который выполняет те же функции, что и модуль .COM, далее его построить, отладить и сравнить исходные тексты для .COM и .EXE модулей.

Таблица 1 — Процедуры в программе.

Процедура	Описание
TETR_TO_HEX	Перевод десятичной цифры в код символа
BYTE_TO_HEX	Перевод байта в 16-ной с/с в символьный код
WRD_TO_HEX	Перевод слова в 16-ной с/с в символьный код
BYTE_TO_DEC	Перевод байта в 16-ной с/с в символьный код в 10-ной с/с

Выполнение работы.

Данные объявленные в программе:

```
typePC db 'IBM PC type: PC', 0Dh, 0Ah, '$'
```

```
typePC_xt db 'IBM PC type: PC/XT', 0Dh, 0Ah, '$'
```

```
typeAt db 'IBM PC type: AT or PS2 (50 or 60)', 0Dh, 0Ah, '$'
```

```
typePC30 db 'IBM PC type: PS2 30', 0Dh, 0Ah, '$'
```

```
typePC80 db 'IBM PC type: PS2 80', 0Dh, 0Ah, '$'
```

```
typePCjr db 'IBM PC type: PCjr', 0Dh, 0Ah, '$'
```

```

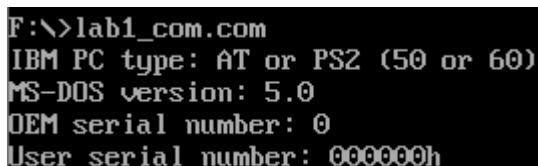
typePC_convert db 'IBM PC type: PC Convertible', 0Dh, 0Ah, '$'
type_undefined db 'Undefined IBM PC type code:  h', 0Dh, 0Ah, '$'

ver db 'MS-DOS version:  . ', 0Dh, 0Ah, '$'
oemNum db 'OEM serial number:  ', 0Dh, 0Ah, '$'
userNum db 'User serial number:      h$'

```

Программа последовательно выводит тип ПК, версию ОС, OEM и номер пользователя.

Далее предоставлены скриншоты полученных модулей.

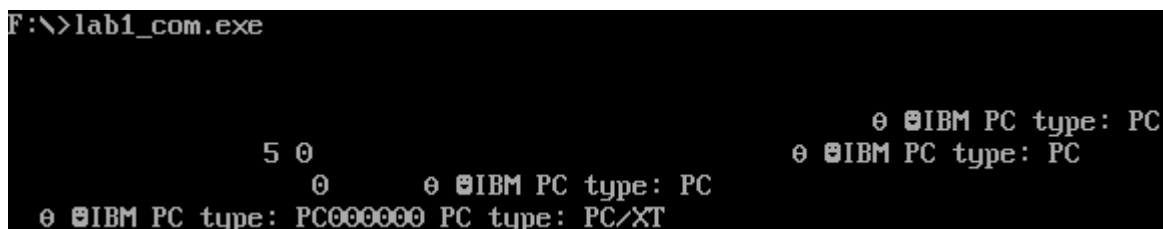


```

F:\>lab1_com.com
IBM PC type: AT or PS2 (50 or 60)
MS-DOS version: 5.0
OEM serial number: 0
User serial number: 000000h

```

Рис. 1 — Хороший com модуль



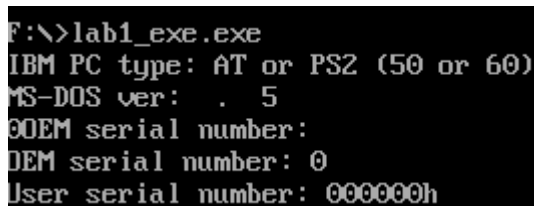
```

F:\>lab1_com.exe

5 0
0
0 IBM PC type: PC
0 IBM PC type: PC
0 IBM PC type: PC000000 PC type: PC/XT

```

Рис. 2 — Плохой exe модуль



```

F:\>lab1_exe.exe
IBM PC type: AT or PS2 (50 or 60)
MS-DOS ver:  . 5
OEM serial number:
OEM serial number: 0
User serial number: 000000h

```

Рис. 3 — Хороший exe модуль

Отличия исходных текстов COM и EXE программ

1. Сколько сегментов должна содержать COM-программа?

COM-программа должна содержать ровно один сегмент. Код и данные находятся в одном сегменте, а стек генерируется автоматически.

2. EXE-программа?

EXE-программа должна содержать не менее одного сегмента. Сегменты кода, данных и стека описываются отдельно друг от друга, но есть возможность не описывать сегмент стека, в таком случае будет использоваться стек DOS.

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

Должна быть обязательна директива `ORG 100h`, потому что при загрузке модуля все сегментные регистры содержат адрес префикса программного сегмента (PSP), который является 256-байтовым(100H) блоком. `ASSUME` - для того, чтобы сегмент данных и сегмент кода указывали на один общий сегмент.

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

Не все форматы поддерживаются. Нельзя использовать команды вида `mov <регистр>, seg <имя сегмента>`, так как в .com-программе отсутствует таблица настроек (содержит описание адресов, которые зависят от размещения загрузочного модуля в ОП).

Отличия форматов файлов .COM и .EXE программ

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

Address	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	Dump
00000000	e9	20	02	49	42	4d	20	50	43	20	74	79	70	65	3a	20	й .IBM PC type:
00000010	50	43	0d	0a	24	49	42	4d	20	50	43	20	74	79	70	65	PC..\$IBM PC type
00000020	3a	20	50	43	2f	58	54	0d	0a	24	49	42	4d	20	50	43	: PC/XT..\$IBM PC
00000030	20	74	79	70	65	3a	20	41	54	20	6f	72	20	50	53	32	type: AT or PS2
00000040	20	28	35	30	20	6f	72	20	36	30	29	0d	0a	24	49	42	(50 or 60)..\$IB
00000050	4d	20	50	43	20	74	79	70	65	3a	20	50	53	32	20	33	M PC type: PS2 3
00000060	30	0d	0a	24	49	42	4d	20	50	43	20	74	79	70	65	3a	0..\$IBM PC type:
00000070	20	50	53	32	20	38	30	0d	0a	24	49	42	4d	20	50	43	PS2 80..\$IBM PC
00000080	20	74	79	70	65	3a	20	50	d0	a1	6a	72	0d	0a	24	49	type: PPÿjr..\$I
00000090	42	4d	20	50	43	20	74	79	70	65	3a	20	50	43	20	43	BM PC type: PC C
000000a0	6f	6e	76	65	72	74	69	62	6c	65	0d	0a	24	55	6e	64	onvertible..\$Und
000000b0	65	66	69	6e	65	64	20	49	42	4d	20	50	43	20	74	79	efined IBM PC ty
000000c0	70	65	20	63	6f	64	65	3a	20	20	20	68	0d	0a	24	4d	pe code: h..\$M
000000d0	53	2d	44	4f	53	20	76	65	72	73	69	6f	6e	3a	20	20	S-DOS version:
000000e0	2e	20	20	0d	0a	24	4f	45	4d	20	73	65	72	69	61	6c	. ..\$OEM serial
000000f0	20	6e	75	6d	62	65	72	3a	20	20	20	0d	0a	24	55	73	number: ..\$Us
00000100	65	72	20	73	65	72	69	61	6c	20	6e	75	6d	62	65	72	er serial number
00000110	3a	20	20	20	20	20	20	20	68	24	24	0f	3c	09	76	02	: h\$\$.<.v.
00000120	04	07	04	30	c3	51	8a	e0	e8	ef	ff	86	c4	b1	04	d2	...0ГQЉаипя†Д±.Т
00000130	e8	e8	e6	ff	59	c3	53	8a	fc	e8	e9	ff	88	25	4f	88	иижяУТСЉыйяё%Оё
00000140	05	4f	8a	c7	e8	de	ff	88	25	4f	88	05	5b	c3	51	52	.ОЉзиЮяё%Оё. [ГQR
00000150	32	e4	33	d2	b9	0a	00	f7	f1	80	ca	30	88	14	4e	33	2дЗТЉ...чсЪК0ё. NЗ
00000160	d2	3d	0a	00	73	f1	3c	00	74	04	0c	30	88	04	5a	59	T=...sc<.t...0ё. ZY
00000170	c3	b4	09	cd	21	c3	b8	00	f0	8e	c0	26	a0	fe	ff	3c	Гг.Н!Гё.рѢ&.юя<
00000180	ff	74	2b	3c	fe	74	2d	3c	fb	74	29	3c	fc	74	2b	3c	ят+<ѳт-<ѳт) <ѳт+<
00000190	fa	74	2d	3c	f8	74	2f	3c	fd	74	31	3c	f9	74	33	e8	ѳт-<ѳт/<ѳт1<ѳт3и
000001a0	83	ff	bb	ad	01	89	47	1c	ba	ad	01	eb	28	90	ba	03	ђя»-.%G.с-..л(.е.
000001b0	01	eb	22	90	ba	15	01	eb	1c	90	ba	2a	01	eb	16	90	.л".е...л...е*.л..
000001c0	ba	4e	01	eb	10	90	ba	64	01	eb	0a	90	ba	7a	01	eb	еN.л...ед.л...ез.л
000001d0	04	90	ba	8f	01	e8	99	ff	c3	b4	30	cd	21	50	be	cf	..е...иѳяГг0Н!РсП
000001e0	01	83	c6	10	e8	67	ff	58	8a	c4	83	c6	03	e8	5e	ff	.ђЖ.игяХЉдђЖ.и^я
000001f0	ba	cf	01	e8	7b	ff	be	e6	01	83	c6	13	8a	c7	e8	4d	еП.и{ясж.ђЖ.ЉЗим
00000200	ff	ba	e6	01	e8	6a	ff	bf	fe	01	83	c7	19	8b	c1	e8	яеж.ијяію.ђЗ.<Ви
00000210	24	ff	8a	c3	e8	0e	ff	83	ef	02	89	05	ba	fe	01	e8	\$яЉГи.яѓп.%..ею.и
00000220	4f	ff	c3	e8	50	ff	e8	b0	ff	32	c0	b4	4c	cd	21		ОяГиРяи°я2АгЛН!

Рис. 4 — хороший com файл

COM-файл состоит из одного сегмента, состоящего из сегмент кода и сегмент данных, сегмент стека генерируется автоматически при создании

COM модуля. COM-файл ограничен размером одного сегмента и не превышает 64 Кб

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

```
00000300 e9 20 02 49 42 4d 20 50 43 20 74 79 70 65 3a 20 й .IBM PC type:
00000310 50 43 0d 0a 24 49 42 4d 20 50 43 20 74 79 70 65 PC..$IBM PC type
00000320 3a 20 50 43 2f 58 54 0d 0a 24 49 42 4d 20 50 43 : PC/XT..$IBM PC
00000330 20 74 79 70 65 3a 20 41 54 20 6f 72 20 50 53 32 type: AT or PS2
00000340 20 28 35 30 20 6f 72 20 36 30 29 0d 0a 24 49 42 (50 or 60)..$IB
00000350 4d 20 50 43 20 74 79 70 65 3a 20 50 53 32 20 33 M PC type: PS2 3
00000360 30 0d 0a 24 49 42 4d 20 50 43 20 74 79 70 65 3a 0..$IBM PC type:
00000370 20 50 53 32 20 38 30 0d 0a 24 49 42 4d 20 50 43 PS2 80..$IBM PC
00000380 20 74 79 70 65 3a 20 50 d0 a1 6a 72 0d 0a 24 49 type: PPŸjr..$I
00000390 42 4d 20 50 43 20 74 79 70 65 3a 20 50 43 20 43 BM PC type: PC C
000003a0 6f 6e 76 65 72 74 69 62 6c 65 0d 0a 24 55 6e 64 onvertible..$Und
000003b0 65 66 69 6e 65 64 20 49 42 4d 20 50 43 20 74 79 efined IBM PC ty
000003c0 70 65 20 63 6f 64 65 3a 20 20 20 68 0d 0a 24 4d pe code: h..$M
000003d0 53 2d 44 4f 53 20 76 65 72 73 69 6f 6e 3a 20 20 S-DOS version:
000003e0 2e 20 20 0d 0a 24 4f 45 4d 20 73 65 72 69 61 6c . ..$OEM serial
000003f0 20 6e 75 6d 62 65 72 3a 20 20 20 0d 0a 24 55 73 number: ..$Us
00000400 65 72 20 73 65 72 69 61 6c 20 6e 75 6d 62 65 72 er serial number
00000410 3a 20 20 20 20 20 20 20 68 24 24 0f 3c 09 76 02 : h$$.<.v.
00000420 04 07 04 30 c3 51 8a e0 e8 ef ff 86 c4 b1 04 d2 ...0ГQЉиипя†Д±.Т
00000430 e8 e8 e6 ff 59 c3 53 8a fc e8 e9 ff 88 25 4f 88 иижяYГSЉийяє%Oє
00000440 05 4f 8a c7 e8 de ff 88 25 4f 88 05 5b c3 51 52 .OЉзиюяє%Oє.[ГQR
00000450 32 e4 33 d2 b9 0a 00 f7 f1 80 ca 30 88 14 4e 33 2дЗТЉ..чсЪK0є.NЗ
00000460 d2 3d 0a 00 73 f1 3c 00 74 04 0c 30 88 04 5a 59 Т=..sc<.t..0є.ZY
00000470 c3 b4 09 cd 21 c3 b8 00 f0 8e c0 26 a0 fe ff 3c Гг.Н!Гё.рЪА&.юя<
00000480 ff 74 2b 3c fe 74 2d 3c fb 74 29 3c fc 74 2b 3c ят+<ют-<ът)<ът+<
00000490 fa 74 2d 3c f8 74 2f 3c fd 74 31 3c f9 74 33 e8 ът-<шт/<эт1<шт3и
000004a0 83 ff bb ad 01 89 47 1c ba ad 01 eb 28 90 ba 03 фя»-.%G.е-.л(.е.
000004b0 01 eb 22 90 ba 15 01 eb 1c 90 ba 2a 01 eb 16 90 .л".е..л..е*.л..
000004c0 ba 4e 01 eb 10 90 ba 64 01 eb 0a 90 ba 7a 01 eb eN.л..ed.л..ez.л
000004d0 04 90 ba 8f 01 e8 99 ff c3 b4 30 cd 21 50 be cf ..е..и™яГг0Н!PsП
000004e0 01 83 c6 10 e8 67 ff 58 8a c4 83 c6 03 e8 5e ff .фЖ.игряХЉдфЖ.и^я
000004f0 ba cf 01 e8 7b ff be e6 01 83 c6 13 8a c7 e8 4d eП.и{ясж.фЖ.ЉЗим
00000500 ff ba e6 01 e8 6a ff bf fe 01 83 c7 19 8b c1 e8 яеж.ијяію.фЗ.<Би
00000510 24 ff 8a c3 e8 0e ff 83 ef 02 89 05 ba fe 01 e8 $яЉГи.яфп.%ею.и
00000520 4f ff c3 e8 50 ff e8 b0 ff 32 c0 b4 4c cd 21 ОяГиРяи°я2АгЛН!
```

Рис. 5 — плохой exe файл

У «плохого» EXE данные и код располагаются в одном сегменте, что для EXE файла некорректно, так как код и данные должны быть разделены на отдельные сегменты. Код располагается с адреса 300h, а с адреса 0h идёт таблица настроек.

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

```

00000300 49 42 4d 20 50 43 20 74 79 70 65 3a 20 50 43 0d IBM PC type: PC.
00000310 0a 24 49 42 4d 20 50 43 20 74 79 70 65 3a 20 50 .IBM PC type: P
00000320 43 2f 58 54 0d 0a 24 49 42 4d 20 50 43 20 74 79 C/XT..$IBM PC ty
00000330 70 65 3a 20 41 54 20 6f 72 20 50 53 32 20 28 35 pe: AT or PS2 (5
00000340 30 20 6f 72 20 36 30 29 0d 0a 24 49 42 4d 20 50 0 or 60)..$IBM P
00000350 43 20 74 79 70 65 3a 20 50 53 32 20 33 30 0d 0a C type: PS2 30..
00000360 24 49 42 4d 20 50 43 20 74 79 70 65 3a 20 50 53 $IBM PC type: PS
00000370 32 20 38 30 0d 0a 24 49 42 4d 20 50 43 20 74 79 2 80..$IBM PC ty
00000380 70 65 3a 20 50 d0 a1 6a 72 0d 0a 24 49 42 4d 20 pe: PPŸjr..$IBM
00000390 50 43 20 74 79 70 65 3a 20 50 43 20 43 6f 6e 76 PC type: PC Conv
000003a0 65 72 74 69 62 6c 65 0d 0a 24 55 6e 64 65 66 69 ertible..$Undefi
000003b0 6e 65 64 20 49 42 4d 20 50 43 20 74 79 70 65 20 ned IBM PC type
000003c0 63 6f 64 65 3a 20 20 20 68 0d 0a 24 4d 53 2d 44 code: h..$MS-D
000003d0 4f 53 20 76 65 72 3a 20 20 2e 20 20 0d 0a 24 4f OS ver: . .$.O
000003e0 45 4d 20 73 65 72 69 61 6c 20 6e 75 6d 62 65 72 EM serial number
000003f0 3a 20 20 20 0d 0a 24 55 73 65 72 20 73 65 72 69 : ..$User seri
00000400 61 6c 20 6e 75 6d 62 65 72 3a 20 20 20 20 20 al number:
00000410 20 68 24 00 00 00 00 00 00 00 00 00 00 00 00 h$.
00000420 24 0f 3c 09 76 02 04 07 04 30 c3 51 8a e0 e8 ef $.<.v....0ГQЪаип
00000430 ff 86 c4 b1 04 d2 e8 e8 e6 ff 59 c3 53 8a fc e8 я†Д†.ТиижяҮТЅЉи
00000440 e9 ff 88 25 4f 88 05 4f 8a c7 e8 de ff 88 25 4f йяё%Оё.ОљзиЮяё%О
00000450 88 05 5b c3 51 52 32 e4 33 d2 b9 0a 00 f7 f1 80 €. [ГQР2д3ТЎ..чсЪ
00000460 ca 30 88 14 4e 33 d2 3d 0a 00 73 f1 3c 00 74 04 KОё.N3Т=..sc<.t.
00000470 0c 30 88 04 5a 59 c3 b4 09 cd 21 c3 b8 00 f0 8e .0ё.ZҮҮҮҮ.Н!Гё.p҃҃
00000480 c0 26 a0 fe ff 3c ff 74 2b 3c fe 74 2d 3c fb 74 A&.юя<ят+<ют-<ыт
00000490 29 3c fc 74 2b 3c fa 74 2d 3c f8 74 2f 3c fd 74 )<ьт+<ьт-<шт/<ст
000004a0 31 3c f9 74 33 e8 83 ff bb aa 00 89 47 1c ba aa 1<шт3и҃҃я»Е.%G.её
000004b0 00 eb 28 90 ba 00 00 eb 22 90 ba 12 00 eb 1c 90 .л(.е..л".е..л..
000004c0 ba 27 00 eb 16 90 ba 4b 00 eb 10 90 ba 61 00 eb е'.л..еК.л..еа.л
000004d0 0a 90 ba 77 00 eb 04 90 ba 8c 00 e8 99 ff c3 b4 ..ew.л..еѢ.и™яГ҃
000004e0 30 cd 21 50 be cc 00 83 c6 10 e8 67 ff 58 8a c4 0Н!РsМ.ѓЖ.иг҃яХЉД
000004f0 83 c6 03 e8 5e ff ba cc 00 e8 7b ff be df 00 83 ѓЖ.и^яеМ.и{ясЯ.ѓ
00000500 c6 13 8a c7 e8 4d ff ba df 00 e8 6a ff bf f7 00 Ж.ЉзиМяеЯ.и҃҃яіч.
00000510 83 c7 19 8b c1 e8 24 ff 8a c3 e8 0e ff 83 ef 02 ѓЗ.<Ви$яЉ҃҃и.я҃҃п.
00000520 89 05 ba f7 00 e8 4f ff c3 2b c0 50 b8 10 00 8e %.еч.иОяГ+АРё...҃҃
00000530 d8 e8 48 ff e8 a8 ff 32 c0 b4 4c cd 21 ШиНияиЕя2ArLH!

```

Рис. 6 — хороший exe файл

В EXE-модуле код и данные являются отдельными сегментами, также присутствует таблица связей, заголовков, отвечающий за настройку адресов.

В «хорошем» EXE-модуле происходит разделение сегментов (кода и данных), необходимое для правильного форматирования, а в «плохом» содержится лишь один сегмент, объединяющий код и данные. «Плохой» EXE начинает код с 300h, так как он получается из COM модуля, в котором изначально сегмент кода смещён на 100h. Но, так как, происходит создание EXE-модуля, добавляется еще и сдвиг PSP (200h). В «хорошем» EXE присутствует только смещение для PSP модуля, поэтому код начинается с 200h.

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

Стек находится между PSP и данными и занимает с 100h до 300h

Загрузка COM модуля в основную память

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

Определяется сегментный адрес участка Основной Памяти, у которого достаточно места для загрузки программы, образ COM-файла считывается с диска и помещается в память, начиная с адреса PSP 100h. После загрузки двоичного образа COM-модуля сегментные регистры CS, DS, ES и SS указывают на PSP(в данном случае сегментные регистры указывают на 48DD), SP указывает на конец сегмента PSP (FFFE), слово 00H помещено в стек, IP содержит 100H. (Это можно увидеть на Рис. 7)

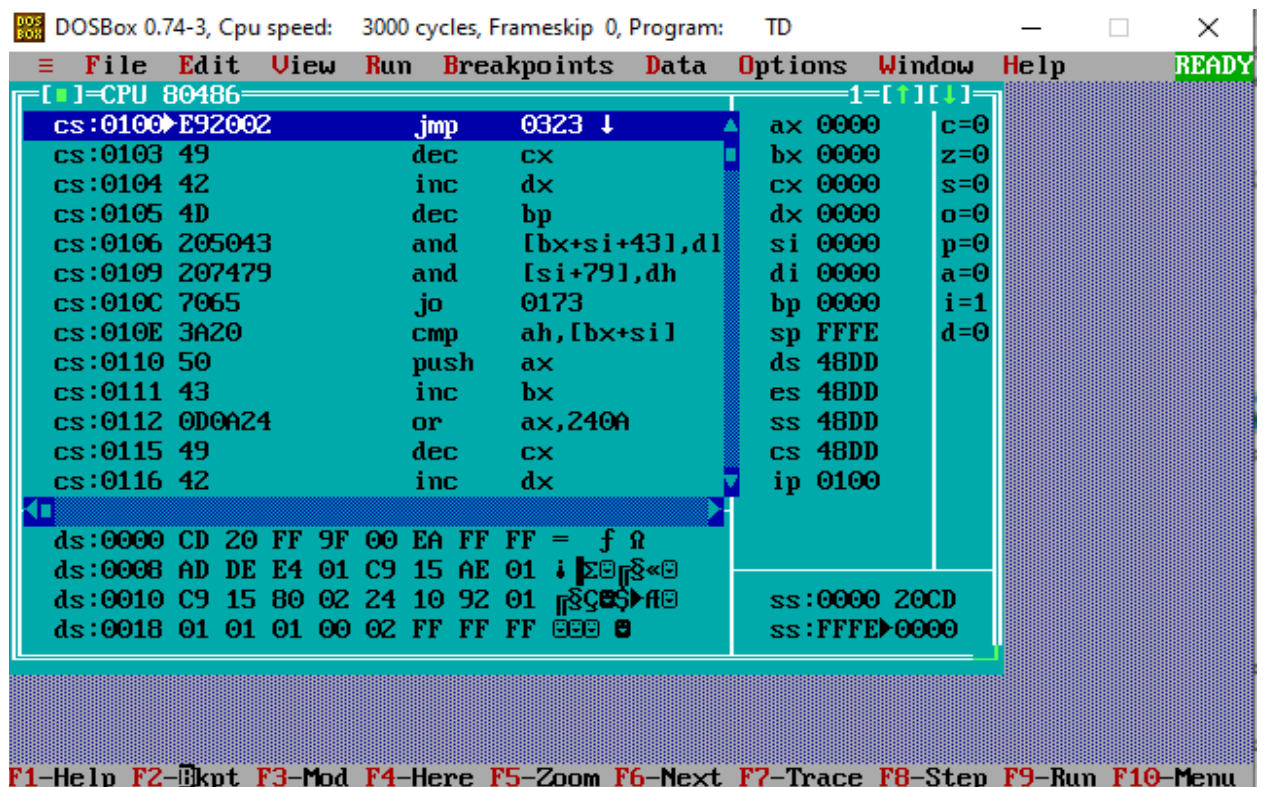


Рис. 7 - Отладчик для "Хорошего" COM-модуля

2. Что располагается с адреса 0?

Программный сегмент PSP, размером 256 байт (100h), зарезервируемый операционной системой.

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

Сегментные регистры CS, DS, ES и SS указывают на PSP и имеют значения 48DD. (Это можно увидеть на Рис. 7)

4. Как определяется стек? Какую область памяти он занимает? Какие адреса?

Стек генерируется автоматически при создании COM-программы. SS – на начало (0h), регистр SP указывает на конец стека (FFFEh), Адреса стека расположены в диапазоне 0h – FFFEh (FFFEh, – последний адрес, кратный двум). (Это можно увидеть на Рис. 7)

Загрузка «хорошего» EXE модуля в основную память

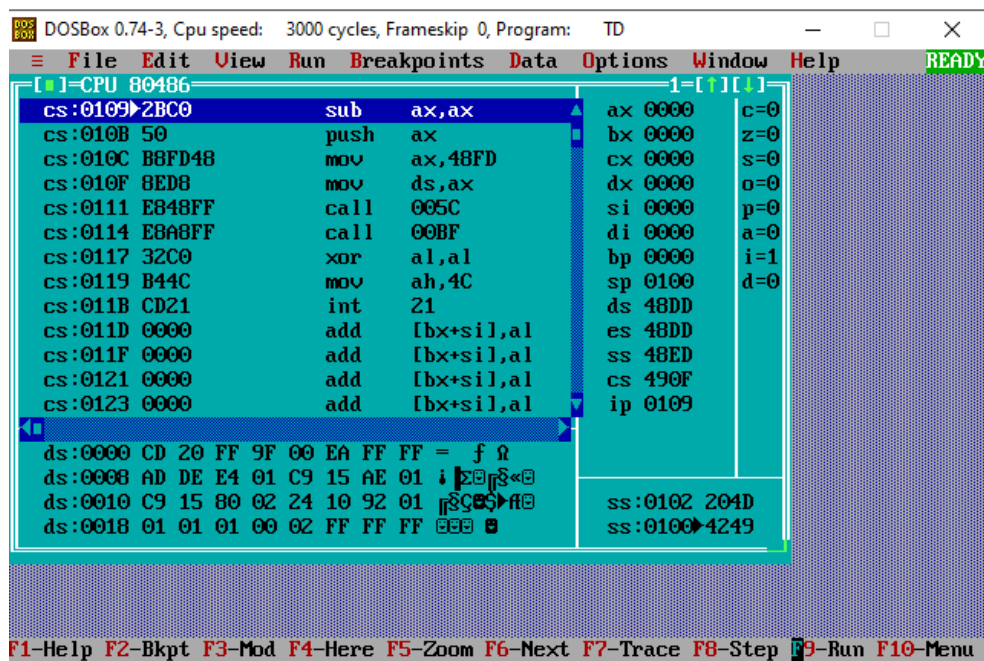


Рис. 8 - Отладчик "хорошего" EXE-модуля

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

EXE-файл загружается, начиная с адреса PSP:0100h. В процессе загрузки считывается информация заголовка (PSP) EXE в начале файла и выполняется перемещение адресов сегментов, то есть DS и ES устанавливаются на начало сегмента PSP (DS=ES=48DD), SS (SS=48ED) – на начало сегмента стека, CS (CS=4905) – на начало сегмента команд. В IP загружается смещение точки входа в программу, которая берётся из метки после директивы END. (Это иллюстрирует рисунок 8)

2. На что указывают регистры DS и ES?

Регистры DS и ES указывают на начало PSP.

3. Как определяется стек?

Стек определяется с помощью Stack Segment, после которой задаётся размер стека. При исполнении регистр SS указывает на начало сегмента стека, а SP на конца стека(его смещение).

4. Как определяется точка входа?

Точка входа определяется при помощи директивы END.

Выводы.

Были написаны COM и EXE модули, на основе которых производилось сравнение данных форматов. Также были выявлены недостатки и преимущества каждого из них.