

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

ОТЧЕТ
по практической работе №1
по дисциплине «Операционные системы»
Тема: Исследование структур загрузочных модулей

Студент гр. 7383

Зуев Д.В.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2019

Постановка задачи.

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

Реализация задачи:

В данной работе были написаны и использовались готовые следующие функции:

TETR_TO_HEX – вспомогательная функция для функции BYTE_TO_HEX

BYTE_TO_HEX – переводит байт AL в два символа шестнадцатеричного числа в AX.

WRD_TO_HEX – переводит шестнадцатиразрядное число, расположенное в AX в шестнадцатеричную систему счисления.

BYTE_TO_DEC – переводит число в десятичную систему счисления.

PRINT – вызывает прерывание программы для печати строки.

FOUND_PC_TYPE – используя исходные данные, определяет и выводит в консоль тип PC.

FOUND_VERSION_OS – с помощью функции 30H определяет и выводит в консоль номер основной версии ОС, номер модификации ОС, серийный номер OEM и серийный номер пользователя

Данная утилита действует следующим образом:

1. Берет код по адресу 0F000:0FFFEh и сравнивает его с соответствующим ему по табл. 1 типом IBM PC.

Таблица 1 – Соответствие кода и типа PC.

Тип PC	Код
PC	FF
PC/XT	FE, FB
AT	FC
PS2 model 30	FA
PS2 model 50 or 60	FC
PS2 model 80	F8
PCjr	FD

PC Convertible	F9
----------------	----

2. Выводит тип PC в консоль.
3. Выполняет функцию 30h с помощью прерывания int 21h.
Выходными параметрами данной функции являются:
AL – номер основной версии.
AH – номер модификации.
BH – серийный номер OEM.
BL:CH – 24-битовый серийный номер пользователя.
4. Полученные на предыдущем шаге данные переводятся в шестнадцатеричную систему счисления и выводятся соответствующими сообщениями в консоль.

В ходе данной работы были написаны программы на языке Ассемблера для .COM и .EXE загрузочных модулей.

Результатом работы «плохого» и «хорошего» .EXE модулей, а также .COM модуля представлены на рисунках 1 – 3.

```

C:\>bad_exe.exe

Type of PC is 50
Type of PC is 50
Type of PC is 25
Type of PC is 000000
Type of PC is 000000

```

Рисунок 1 - Выполнение "плохого" .EXE модуля

```

C:\>GOOD_EXE.EXE
Type of PC is AT
OS Version is 5.0
OEM number is 255
User number is 000000

```

Рисунок 2 - Выполнение "хорошего" .EXE модуля

```

C:\>BAD_EXE.COM
Type of PC is AT
OS Version is 5.0
OEM number is 255
User number is 000000

```

Рисунок 3 - Выполнение .COM модуля

В данной работе также были просмотрены шестнадцатеричные коды исполняемых файлов. Скриншоты представлены на рисунках 4 – 8.

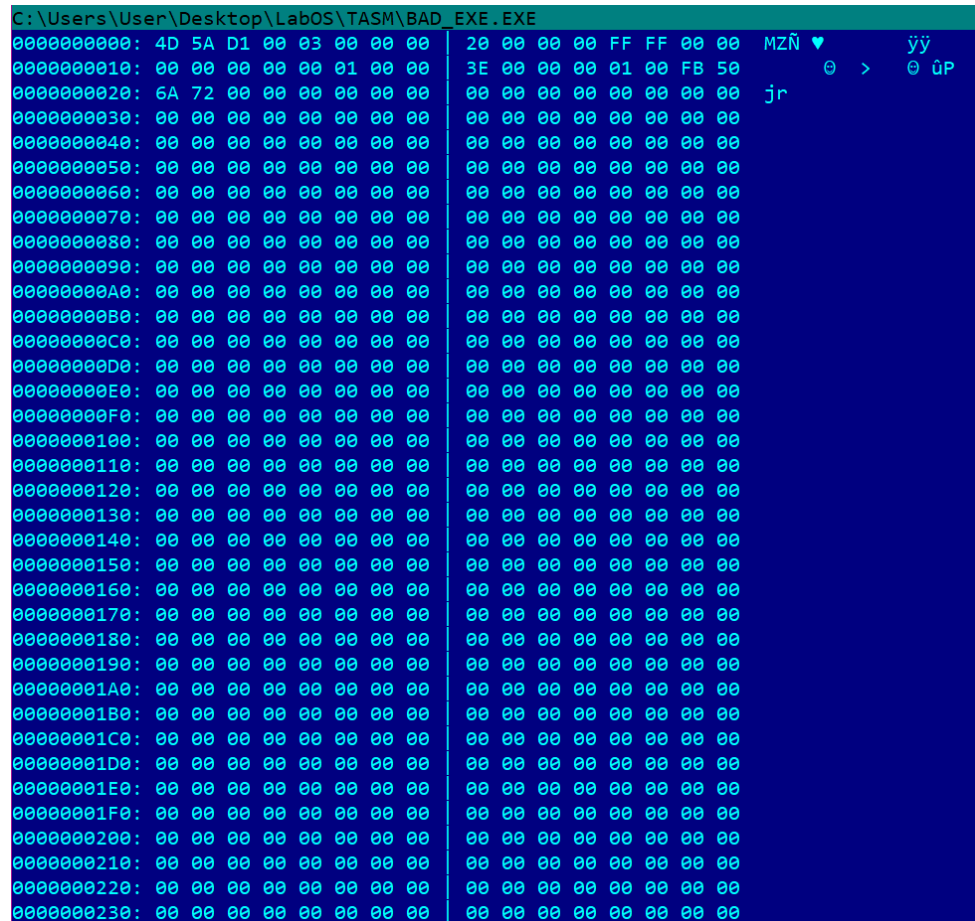


Рисунок 4 - Начало структуры "плохого" .EXE файла

00000002A0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
00000002B0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
00000002C0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
00000002D0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
00000002E0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
00000002F0:	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
0000000300:	E9 C1 01 54 79 70 65 20	6F 66 20 50 43 20 69 73	éÁ@Type of PC is
0000000310:	20 24 4F 53 20 56 65 72	73 69 6F 6E 20 69 73 20	\$OS Version is
0000000320:	20 20 2E 20 20 0D 0A 24	4F 45 4D 20 6E 75 6D 62	. ¼\$OEM numb
0000000330:	65 72 20 69 73 20 20 20	20 0D 0A 24 55 73 65 72	er is ¼\$User
0000000340:	20 6E 75 6D 62 65 72 20	69 73 20 24 09 24 0D 0A	number is \$o\$¼
0000000350:	24 50 43 0D 0A 24 50 43	2F 58 54 0D 0A 24 41 54	\$PC¼\$PC/XT¼\$AT
0000000360:	0D 0A 24 50 53 32 20 6D	6F 64 65 6C 20 33 30 0D	¼\$PS2 model 30¼
0000000370:	0A 24 50 53 32 20 6D 6F	64 65 6C 20 35 30 20 6F	¼\$PS2 model 50 o
0000000380:	72 20 36 30 0D 0A 24 50	53 32 20 6D 6F 64 65 6C	r 60¼\$PS2 model
0000000390:	20 38 30 0D 0A 24 50 43	6A 72 0D 0A 24 50 43 20	80¼\$PCjr¼\$PC
00000003A0:	43 6F 6E 76 65 72 74 69	62 6C 65 0D 0A 24 24 0F	Convertible¼\$ \$o
00000003B0:	3C 09 76 02 04 07 04 30	C3 51 8A E0 E8 EF FF 86	<ov0♦♦0AQŠàèiÿ†
00000003C0:	C4 B1 04 D2 E8 E8 E6 FF	59 C3 53 8A FC E8 E9 FF	Ä±♦0èèæÿYÄSSüèÿ
00000003D0:	88 25 4F 88 05 4F 8A C7	E8 DE FF 88 25 4F 88 05	^%O^+OŠÇèÿ^%O^+
00000003E0:	5B C3 51 52 32 E4 33 D2	B9 0A 00 F7 F1 80 CA 30	[ÄQR2ä3Ö± ¼ ÷ñèÈ0
00000003F0:	88 14 4E 33 D2 3D 0A 00	73 F1 3C 00 74 04 0C 30	^¼N3Ö=¼ sñ< t♦90
0000000400:	88 04 5A 59 C3 B4 09 CD	21 C3 BA 03 01 E8 F5 FF	^♦ZYÄ´oi¼Ä±♥0èöÿ
0000000410:	BB 00 F0 8E C3 26 A1 FE	FF BA 51 01 3C FF 74 47	» öŽÄ&¼þÿ°Q0<ÿtG
0000000420:	BA 56 01 3C FE 74 40 3C	FB 74 3C BA 5E 01 3C FC	°V0<þt@<üt<°^0<ü
0000000430:	74 35 BA 63 01 3C FA 74	2E BA 72 01 3C FC 74 27	t5°c0<üt.°r0<üt'
0000000440:	BA 87 01 3C F8 74 20 BA	96 01 3C FD 74 19 BA 9D	°±0<øt °-0<ÿt¼°¼
0000000450:	01 3C F9 74 12 E8 61 FF	8B D8 8A D3 B4 02 CD 21	0<üt†èäÿ<0ŠÖ´0í!
0000000460:	8A D7 CD 21 BA 4E 01 E8	9B FF C3 33 C0 B4 30 CD	Š×í!°N0è>ÿÄ3Ä´0í
0000000470:	21 BE 12 01 83 C6 0F 50	E8 67 FF 83 C6 03 58 8A	!%†0fÄ±PèÿÿfÄ♥XŠ
0000000480:	C4 E8 5E FF BA 12 01 E8	7B FF BE 28 01 83 C6 10	Äè^ÿ°†0è{ÿ%(0fÄ±
0000000490:	8A C7 E8 4D FF BA 28 01	E8 6A FF BA 3C 01 E8 64	ŠÇæMÿ°(0èÿÿ°<0èd
00000004A0:	FF 8A C3 E8 13 FF 8B D8	8A D3 B4 02 CD 21 8A D7	ÿŠÄè!!ÿ<0ŠÖ´0í!Š×
00000004B0:	CD 21 BF 4C 01 83 C7 03	8B C1 E8 0D FF BA 4C 01	í!¼L0fÇ♥<Äè¼ÿ°L0
00000004C0:	E8 42 FF CB E8 43 FF 0E	E8 A0 FF 32 C0 B4 4C CD	èVÿÈèCÿ¼è ÿ2Ä´Lí
00000004D0:	21		!

Рисунок 5 - Конец структуры "плохого" .EXE файла

C:\Users\User\Desktop\LabOS\TASM\BAD_EXE.COM			
0000000000:	E9 C1 01 54 79 70 65 20	6F 66 20 50 43 20 69 73	éÁ@Type of PC is
0000000010:	20 24 4F 53 20 56 65 72	73 69 6F 6E 20 69 73 20	\$OS Version is
0000000020:	20 20 2E 20 20 0D 0A 24	4F 45 4D 20 6E 75 6D 62	. ¼\$OEM numb
0000000030:	65 72 20 69 73 20 20 20	20 0D 0A 24 55 73 65 72	er is ¼\$User
0000000040:	20 6E 75 6D 62 65 72 20	69 73 20 24 09 24 0D 0A	number is \$o\$¼
0000000050:	24 50 43 0D 0A 24 50 43	2F 58 54 0D 0A 24 41 54	\$PC¼\$PC/XT¼\$AT
0000000060:	0D 0A 24 50 53 32 20 6D	6F 64 65 6C 20 33 30 0D	¼\$PS2 model 30¼
0000000070:	0A 24 50 53 32 20 6D 6F	64 65 6C 20 35 30 20 6F	¼\$PS2 model 50 o
0000000080:	72 20 36 30 0D 0A 24 50	53 32 20 6D 6F 64 65 6C	r 60¼\$PS2 model
0000000090:	20 38 30 0D 0A 24 50 43	6A 72 0D 0A 24 50 43 20	80¼\$PCjr¼\$PC
00000000A0:	43 6F 6E 76 65 72 74 69	62 6C 65 0D 0A 24 24 0F	Convertible¼\$ \$o
00000000B0:	3C 09 76 02 04 07 04 30	C3 51 8A E0 E8 EF FF 86	<ov0♦♦0AQŠàèiÿ†
00000000C0:	C4 B1 04 D2 E8 E8 E6 FF	59 C3 53 8A FC E8 E9 FF	Ä±♦0èèæÿYÄSSüèÿ
00000000D0:	88 25 4F 88 05 4F 8A C7	E8 DE FF 88 25 4F 88 05	^%O^+OŠÇèÿ^%O^+
00000000E0:	5B C3 51 52 32 E4 33 D2	B9 0A 00 F7 F1 80 CA 30	[ÄQR2ä3Ö± ¼ ÷ñèÈ0
00000000F0:	88 14 4E 33 D2 3D 0A 00	73 F1 3C 00 74 04 0C 30	^¼N3Ö=¼ sñ< t♦90
0000000100:	88 04 5A 59 C3 B4 09 CD	21 C3 BA 03 01 E8 F5 FF	^♦ZYÄ´oi¼Ä±♥0èöÿ
0000000110:	BB 00 F0 8E C3 26 A1 FE	FF BA 51 01 3C FF 74 47	» öŽÄ&¼þÿ°Q0<ÿtG
0000000120:	BA 56 01 3C FE 74 40 3C	FB 74 3C BA 5E 01 3C FC	°V0<þt@<üt<°^0<ü
0000000130:	74 35 BA 63 01 3C FA 74	2E BA 72 01 3C FC 74 27	t5°c0<üt.°r0<üt'
0000000140:	BA 87 01 3C F8 74 20 BA	96 01 3C FD 74 19 BA 9D	°±0<øt °-0<ÿt¼°¼
0000000150:	01 3C F9 74 12 E8 61 FF	8B D8 8A D3 B4 02 CD 21	0<üt†èäÿ<0ŠÖ´0í!
0000000160:	8A D7 CD 21 BA 4E 01 E8	9B FF C3 33 C0 B4 30 CD	Š×í!°N0è>ÿÄ3Ä´0í
0000000170:	21 BE 12 01 83 C6 0F 50	E8 67 FF 83 C6 03 58 8A	!%†0fÄ±PèÿÿfÄ♥XŠ
0000000180:	C4 E8 5E FF BA 12 01 E8	7B FF BE 28 01 83 C6 10	Äè^ÿ°†0è{ÿ%(0fÄ±
0000000190:	8A C7 E8 4D FF BA 28 01	E8 6A FF BA 3C 01 E8 64	ŠÇæMÿ°(0èÿÿ°<0èd
00000001A0:	FF 8A C3 E8 13 FF 8B D8	8A D3 B4 02 CD 21 8A D7	ÿŠÄè!!ÿ<0ŠÖ´0í!Š×
00000001B0:	CD 21 BF 4C 01 83 C7 03	8B C1 E8 0D FF BA 4C 01	í!¼L0fÇ♥<Äè¼ÿ°L0
00000001C0:	E8 42 FF CB E8 43 FF 0E	E8 A0 FF 32 C0 B4 4C CD	èVÿÈèCÿ¼è ÿ2Ä´Lí
00000001D0:	21		!

Рисунок 6 - Структура .COM файла

C:\Users\User\Desktop\LabOS\TASM\GOOD_EXE.EXE																		
0000000000:	4D	5A	D9	01	03	00	01	00	20	00	00	00	FF	FF	00	00	MZÜ0♥ 0	ÿÿ
0000000010:	00	02	00	00	16	01	2B	00	3E	00	00	00	01	00	FB	50	0 0+ >	0 ÛP
0000000020:	6A	72	00	00	00	00	00	00	00	00	00	00	00	00	00	00	j r	
0000000030:	00	00	00	00	00	00	00	00	00	00	00	00	00	17	01			ı0
0000000040:	2B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	+	
0000000050:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000060:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000070:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000080:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000090:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000000A0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000000B0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000000C0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000000D0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000000E0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000000F0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000100:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000110:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000120:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000130:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000140:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000150:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000160:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000170:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000180:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000190:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000001A0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000001B0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000001C0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000001D0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000001E0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000001F0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000200:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000210:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000220:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000230:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000240:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
0000000250:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		

Рисунок 7 - Начало структуры "хорошего" .EXE файла

C:\Users\User\Desktop\LabOS\TASM\GOOD_EXE.EXE																
000000380:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000390:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000003A0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000003B0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000003C0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000003D0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000003E0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000003F0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000400:	54	79	70	65	20	6F	66	20	50	43	20	69	73	20	24	4F
000000410:	53	20	56	65	72	73	69	6F	6E	20	69	73	20	20	20	2E
000000420:	20	20	0D	0A	24	4F	45	4D	20	6E	75	6D	62	65	72	20
000000430:	69	73	20	20	20	20	0D	0A	24	55	73	65	72	20	6E	75
000000440:	6D	62	65	72	20	69	73	20	24	0D	0A	24	09	24	50	43
000000450:	0D	0A	24	50	43	2F	58	54	0D	0A	24	41	54	0D	0A	24
000000460:	50	53	32	20	6D	6F	64	65	6C	20	33	30	0D	0A	24	50
000000470:	53	32	20	6D	6F	64	65	6C	20	35	30	20	6F	72	20	36
000000480:	30	0D	0A	24	50	53	32	20	6D	6F	64	65	6C	20	38	30
000000490:	0D	0A	24	50	43	6A	72	0D	0A	24	50	43	20	43	6F	6E
0000004A0:	76	65	72	74	69	62	6C	65	0D	0A	24	00	00	00	00	00
0000004B0:	24	0F	3C	09	76	02	04	07	04	30	C3	51	8A	E0	E8	EF
0000004C0:	FF	86	C4	B1	04	D2	E8	E8	E6	FF	59	C3	53	8A	FC	E8
0000004D0:	E9	FF	88	25	4F	88	05	4F	8A	C7	E8	DE	FF	88	25	4F
0000004E0:	88	05	5B	C3	51	52	32	E4	33	D2	B9	0A	00	F7	F1	80
0000004F0:	CA	30	88	14	4E	33	D2	3D	0A	00	73	F1	3C	00	74	04
000000500:	0C	30	88	04	5A	59	C3	B4	09	CD	21	C3	BA	00	00	E8
000000510:	F5	FF	BB	00	F0	8E	C3	26	A1	FE	FF	BA	4E	00	3C	FF
000000520:	74	47	BA	53	00	3C	FE	74	40	3C	FB	74	3C	BA	5B	00
000000530:	3C	FC	74	35	BA	60	00	3C	FA	74	2E	BA	6F	00	3C	FC
000000540:	74	27	BA	84	00	3C	F8	74	20	BA	93	00	3C	FD	74	19
000000550:	BA	9A	00	3C	F9	74	12	E8	61	FF	8B	D8	8A	D3	B4	02
000000560:	CD	21	8A	D7	CD	21	BA	49	00	E8	9B	FF	CB	33	C0	B4
000000570:	30	CD	21	BE	0F	00	83	C6	0F	50	E8	67	FF	83	C6	03
000000580:	58	8A	C4	E8	5E	FF	BA	0F	00	E8	7B	FF	BE	25	00	83
000000590:	C6	10	8A	C7	E8	4D	FF	BA	25	00	E8	6A	FF	BA	39	00
0000005A0:	E8	64	FF	8A	C3	E8	13	FF	8B	D8	8A	D3	B4	02	CD	21
0000005B0:	8A	D7	CD	21	BF	4C	00	83	C7	03	8B	C1	E8	0D	FF	BA
0000005C0:	4C	00	E8	42	FF	CB	B8	20	00	8E	D8	0E	E8	3D	FF	0E
0000005D0:	E8	9A	FF	32	C0	B4	4C	CD	21							

Рисунок 8 - Конец структуры "хорошего" .EXE файла

Скриншоты отладчика для исполняемых файлов представлены на рисунках 9 – 11.

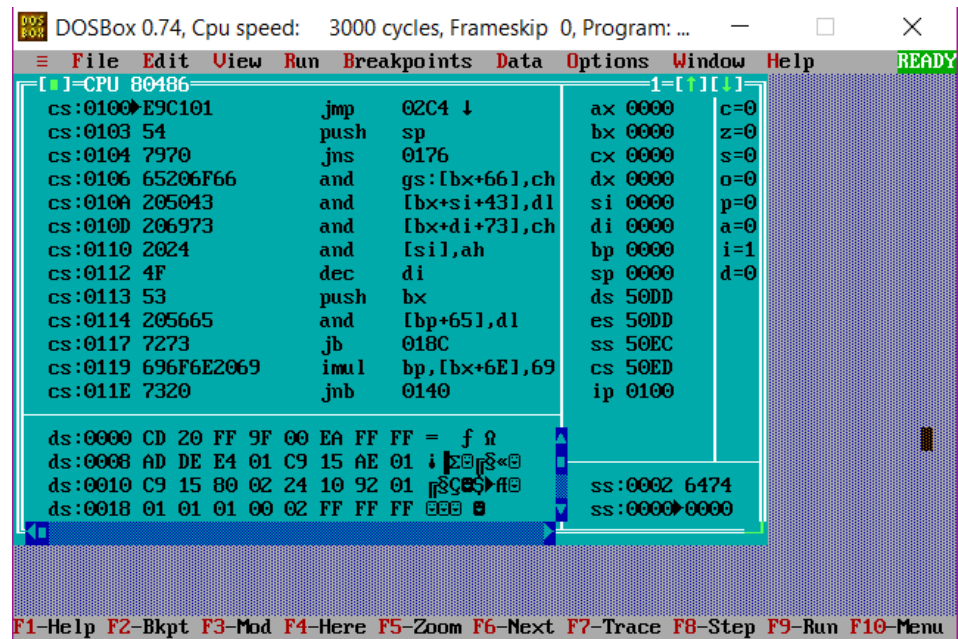


Рисунок 9 - Отладка "плохого" .EXE файла

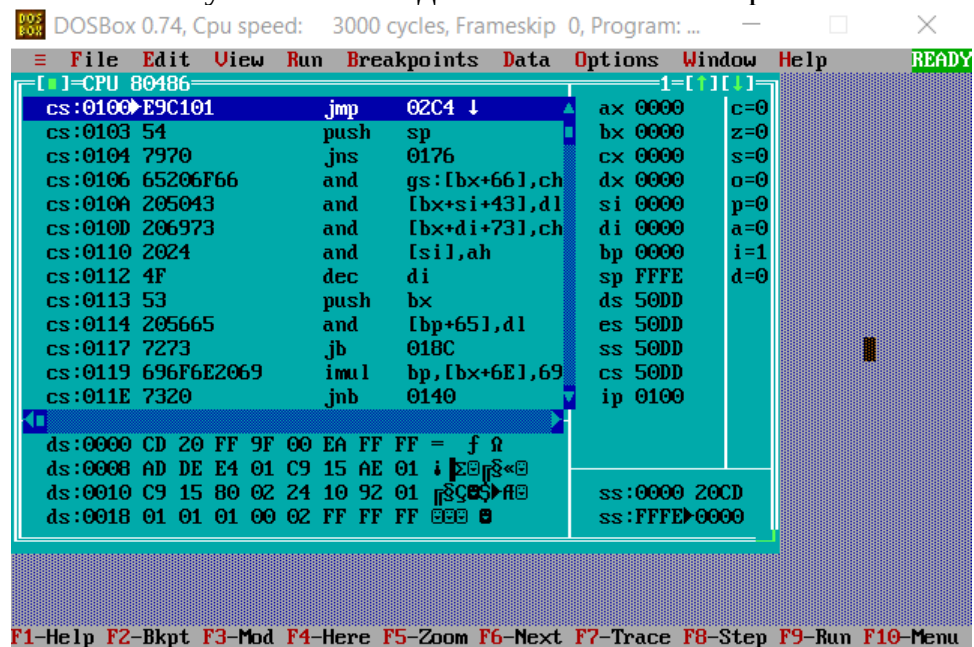


Рисунок 10 - Отладка .COM файла

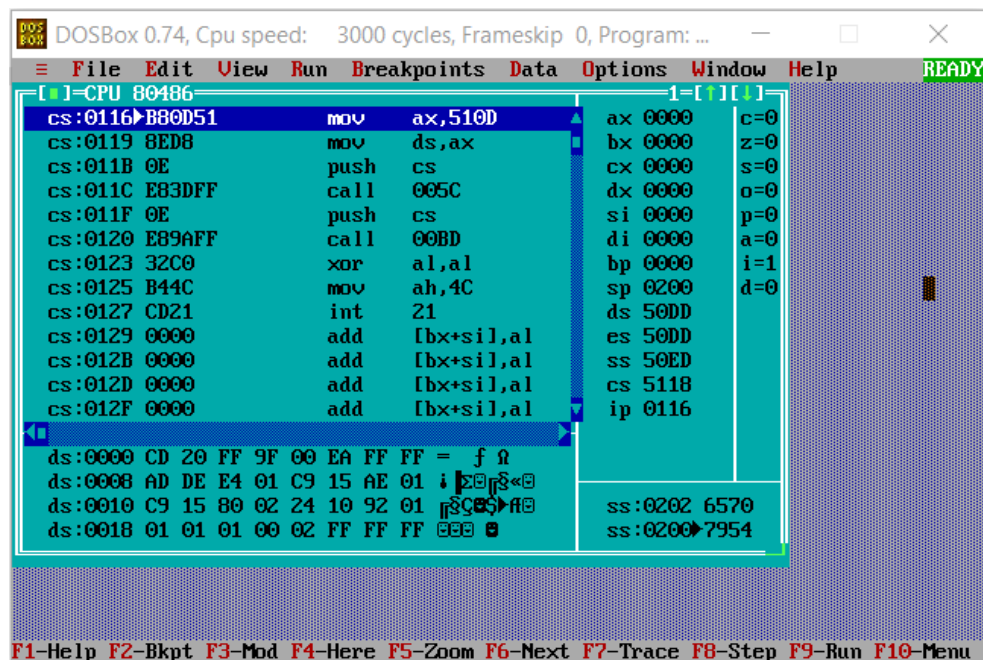


Рисунок 11 - Отладка "хорошего" .EXE файла

Выводы.

В ходе выполнения данной лабораторной работы была написана программа в двух версиях: для .COM и для .EXE модулей, определяющая тип IBM PC, версию операционной системы, номер OEM и серийный номер пользователя. Исследованы различия в структурах исходных текстов модулей типов .COM и .EXE, структур файлов загрузочных модулей и способов их загрузки в основную память.

Контрольные вопросы.

Отличия исходных текстов COM и EXE программ:

- 1) Сколько сегментов должна содержать COM-программа?
COM-программа содержит 1 сегмент.
- 2) EXE-программа?
EXE-программа содержит любое количество сегментов.
- 3) Какие директивы должны обязательно быть в тексте COM-программы?
В COM-программе обязательно должна быть директива ORG 100h, резервирующая 256 байт для PSP.
- 4) Все ли форматы команд можно использовать в COM-программе?
В COM-программе нельзя использовать команды, использующие адреса сегментов.

Отличия форматов файлов COM и EXE модулей:

- 1) Какова структура файла COM? С какого адреса располагается код?
Выполнимый файл .COM включает в себя программные строки вместе со строками данных. Код располагается с адреса 0h.
- 2) Какова структура «плохого» EXE? С какого адреса располагается код? Что располагается с адреса 0?

Так же как и в .COM файле в «плохом» .EXE данные и код располагаются в одном сегменте. Код располагается с адреса 300h. С адреса 0h располагается заголовок, состоящий из 1 блока размером 200h, хранящий в себе информацию, необходимую для правильной настройки регистров процессора и самой программы при ее загрузке в память. Так же 100h зарезервировано директивой ORG 100h.

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

В отличие от «плохого» EXE в «хорошем» EXE данные, стек и код расположены в разных сегментах.

Загрузка COM модуля в основную память:

- 1) Какой формат загрузки модуля COM? С какого адреса располагается код?
Система выделяет для программы 64 Кбайт адресного пространства. Первые 256 байт отводятся под PSP, на который указывают регистры CS, DS, ES, SS. Далее располагается код программы и данных. По рис. 10 видно, что код располагается с адреса CS+IP=51DD. Оставшуюся память занимает стек. Регистр SP хранит значение вершины стека, равное FFFEh.
- 2) Что располагается с адреса 0?
С адреса 0 располагается PSP.
- 3) Какие значения имеют сегментные регистры? На какие области памяти они указывают?
Значения сегментных регистров одинаковы и равны 50DD (см. рис. 10). Сегментные регистры указывают на PSP.
- 4) Как определяется стек? Какую область памяти он занимает? Какие адреса?
Стек определяется сегментным регистром стека SS и адресом вершины стека SP. Занимает область памяти расположенную между SS и SP, то есть 0h – FFFEh.

Загрузка «хорошего» EXE модуля в основную память:

- 1) Как загружается «хороший» EXE? Какие значения имеют сегментные регистры?
Регистры DS и ES равны 50DD и указывают на PSP. Регистр CS равен 5118 и указывает на сегмент кода (см. рис. 11). Регистр SS равен 50ED и указывает на сегмент стека.
- 2) На что указывают регистры DS и ES?
Регистры DS и ES указывают на PSP.
- 3) Как определяется стек?
Стек определяется с помощью директивы сегментации SEGMENT и атрибута комбинирования сегментов определяющего сегмент стека STACK.
- 4) Как определяется точка входа?
Точка входа определяется директивой END.