

**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**

Студентка гр. 0381

Сарычева А.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.

### **Задание.**

Напишите текст исходного .COM модуля, который определяет тип PC и версию системы. Ассемблерная программа должна читать содержимое предпоследнего байта ROM BIOS, по таблице, сравнивая коды, определять тип PC 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 и серийным номером пользователя. Полученные строки выводятся на экран.

Таблица 1 – Необходимые данные: соотношение типа IBM PS с кодом

Тип IBM PS	Код
PS	FF
PC/XT	FE,FB
AT	FC
PS2 модель 30	FA
PS2 модель 50 или 60	FC
PS2 модель 80	F8
PCjr	FD
PC Convertible	F9

### **Выполнение работы.**

В файле lb1\_com.asm написан код исходного .COM модуля. В начале данного модуля прописаны строки для вывода запрашиваемой информации, для

вывода которых была создана процедура OUTPUT. Кроме того, были использованы процедуры TETR\_TO\_HEX, BYTE\_TO\_HEX, WRD\_TO\_HEX, BYTE TO DEC из предоставленного шаблона.

Написана процедура TYPE\_PC, которая определяет тип PC. В регистр AL сохраняется значение байта, который хранит в себе код типа PC. Данный код используется для сравнения с кодами, указанными в табл.1, при наличии совпадения с одним из указанных кодов происходит переход по соответствующей метке, где происходит запись в регистр DX смещение необходимой строки сообщения, после чего происходит вызов процедуры печати сообщения – OUTPUT.

Далее написана процедура SYSTEM\_VERSION, которая определяет версию MS-DOS, серийный номер OEM и серийный номер пользователя. С помощью функции 30H прерывания 21H определяются необходимые данные, которые выводятся в соответствующем формате с помощью процедуры OUTPUT.

Командой `MASM lb1_com.asm` был получен объектный файл `lb1_com.obj`, из которого командой `LINK lb1_com.obj` собирается «плохой» `.EXE`-модуль. При его запуске выводятся строки, представленные на рис.1.

```
F:\>lb1_com.exe

                                     щ(ⓈТип PC: PC
                                     щ(ⓈТип PC: PC
5 0
                                     щ(ⓈТип PC: PC
                                     щ(ⓈТип PC: PC
00
щ(ⓈТип PC: PC
0000000C: PC\XT
```

Рисунок 1 – вывод модуля `lb1 com.exe`

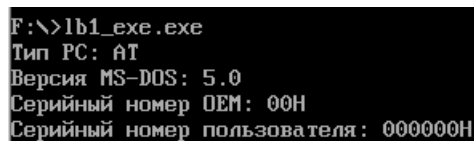
Командой EXE2BIN lb1\_com.exe lb1\_com.com был получен «хороший» .COM-модуль. При его запуске выводятся строки, представленные на рис.2.

```
F:\>lb1_com.com
Тип PC: AT
Версия MS-DOS: 5.0
Серийный номер OEM: 00H
Серийный номер пользователя: 000000H
```

Рисунок 2 – вывод модуля lb1 com.com

Далее в файле lb1\_exe.asm был написан код «хорошего» .EXE-модуля, на основе кода из файла lb1\_com.asm, в который были добавлены определения сегментов данных и стека, необходимые данные были перенесены в соответствующие им сегмент, кроме того, была создана далекая головная процедура MAIN, в которой инициализируется регистр DS адресом начала сегмента данных и вызываются процедуры TYPE\_PC и SYSTEM\_VERSION.

Командой MASM lb1\_exe.asm был получен объектный файл lb1\_exe.obj, из которого командой LINK lb1\_exe.obj собирается «хороший» .EXE-модуль. При его запуске выводятся строки, представленные на рис.3.



```
F:\>lb1_exe.exe
Тип PC: AT
Версия MS-DOS: 5.0
Серийный номер OEM: 00H
Серийный номер пользователя: 000000H
```

Рисунок 3 – вывод модуля lb1\_exe.com

## **Выводы.**

В ходе лабораторной работы были исследованы различия в структурах исходных текстов модулей типов .COM и .EXE, структуры файлов загрузочных модулей и способы их загрузки в основную память.

## **Ответы на контрольные вопросы.**

Отличия исходных текстов COM и EXE программ

1) Сколько сегментов должна содержать COM-программа?

COM-программа должна содержать 1 сегмент, являющийся сегментом кода. В нем же определяются данные, а стек генерируется автоматически, в следствие чего он опускается в COM-программе.

2) Сколько сегментов должна содержать EXE-программа?

EXE-программа должна содержать сегмент кода, сегмент данных, сегмент стека. Сегмент стека может быть опущен, в таком случае будет использоваться стек DOS. В общем случае, EXE-программа должна содержать не менее одного сегмента.

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

В тексте COM-программы должны обязательно присутствовать:

-директива ASSUME, которая связывает сегментные регистры с именем единственного сегмента;

-директива ORG 100h, которая обеспечивает смещение на 256 байт от нулевого адреса для устранения попадания в PSP.

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

В COM-программе не поддерживаются команды с указанием сегментов в виде операндов, т.к. в COM-программах отсутствует таблица настройки, которая используется для определения абсолютных адресов сегментов.

Отличия форматов файлов COM и EXE модулей

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

Файл состоит из одного сегмента, в котором находятся код и данные. Код располагается с адреса 0, так как в COM-файлы не содержат заголовка и таблицы настройки адресов. Размер файла не может превышать 65280 байт.

Address	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	Dump
00000000	e9	28	02	92	a8	af	20	50	43	3a	20	50	43	0d	0a	24	й(.'ЁЇ PC: PC..\$
00000010	92	a8	af	20	50	43	3a	20	50	43	2f	58	54	0d	0a	24	'ЁЇ PC: PC/XT..\$
00000020	92	a8	af	20	50	43	3a	20	41	54	0d	0a	24	92	a8	af	'ЁЇ PC: AT..'ЁЇ
00000030	20	50	43	3a	20	50	53	32	20	ac	ae	a4	a5	ab	ec	20	PC: PS2 -ЁЇ«м
00000040	33	30	0d	0a	24	92	a8	af	20	50	43	3a	20	50	53	32	30..\$'ЁЇ PC: PS2
00000050	20	ac	ae	a4	a5	ab	ec	20	35	30	20	a8	ab	a8	20	36	-ЁЇ«м 50 Ё«Ё 6
00000060	30	0d	0a	24	92	a8	af	20	50	43	3a	20	50	53	32	20	0..\$'ЁЇ PC: PS2
00000070	ac	ae	a4	a5	ab	ec	20	38	30	0d	0a	24	92	a8	af	20	-ЁЇ«м 80..\$'ЁЇ
00000080	50	43	3a	20	50	91	6a	72	0d	0a	24	92	a8	af	20	50	PC: P'jr..\$'ЁЇ P
00000090	43	3a	20	50	43	20	43	6f	6e	76	65	72	74	69	62	6c	C: PC Convertibl
000000a0	65	0d	0a	24	92	a8	af	20	50	43	3a	20	ad	a5	ae	af	e..\$'ЁЇ PC: -ГЇЇ
000000b0	e0	a5	a4	a5	ab	a5	ad	2c	20	aa	ae	a4	3a	20	20	0d	aГ«ГГ-, ё«м: .
000000c0	0a	24	82	a5	e0	e1	a8	ef	20	4d	53	2d	44	4f	53	3a	.\$,ГабЁп MS-DOS:
000000d0	20	20	2e	20	20	0d	0a	24	91	a5	e0	a8	a9	ad	eb	a9	. ..\$'ГaЁЁ-л«
000000e0	20	ad	ae	ac	a5	e0	20	4f	45	4d	3a	20	20	20	48	0d	-«-Гa OEM: Н.
000000f0	0a	24	91	a5	e0	a8	a9	ad	eb	a9	20	ad	ae	ac	a5	e0	.\$'ГaЁЁ-л« -«-Гa
00000100	20	af	ae	ab	ec	a7	ae	a2	a0	e2	a5	ab	ef	3a	20	20	Ї«м\$«Ї.вГ«п:
00000110	20	20	20	20	20	48	24	24	0f	3c	09	76	02	04	07	04	Н\$\$.<.v....
00000120	30	c3	51	8a	e0	e8	ef	ff	86	c4	b1	04	d2	e8	e8	e6	ОГQЇаипя†Д†.Гииж
00000130	ff	59	c3	53	8a	fc	e8	e9	ff	88	25	4f	88	05	4f	8a	яYГSЇьийя€%О€.ОЇ
00000140	c7	e8	de	ff	88	25	4f	88	05	5b	c3	51	52	32	e4	33	ЗиЮя€%О€. [ГQR2л3
00000150	d2	b9	0a	00	f7	f1	80	ca	30	88	14	4e	33	d2	3d	0a	ТЇ..чсЪK0€.N3T=.
00000160	00	73	f1	3c	00	74	04	0c	30	88	04	5a	59	c3	b4	09	.sc<.t...0€.ZYГг.
00000170	cd	21	c3	b8	00	f0	8e	c0	26	a0	fe	ff	3c	ff	74	1e	Н!Гё.рЇA€.юя<ят.
00000180	3c	fe	74	20	3c	fb	74	1c	3c	fc	74	1e	3c	fa	74	20	<ют <ыт.<ьт.<ьт
00000190	3c	f8	74	22	3c	fd	74	24	3c	f9	74	26	74	2a	ba	03	<шт"<эт\$<шт&t*е.
000001a0	01	eb	38	90	ba	10	01	eb	32	90	ba	20	01	eb	2c	90	.л8.е..л2.е .л,. .
000001b0	ba	2d	01	eb	26	90	ba	64	01	eb	20	90	ba	7c	01	eb	е-.л&.ед.л .е .л
000001c0	1a	90	ba	8b	01	eb	14	90	bf	a4	01	83	c7	19	8a	c7	..е<.л..їя.їЗ.ЇЗ
000001d0	e8	4f	ff	89	05	ba	a4	01	eb	01	90	e8	90	ff	c3	b4	иОя%.ен.л..и.яГг
000001e0	30	cd	21	50	be	c2	01	83	c6	0f	e8	5e	ff	58	8a	c4	ОН!РзВ.їЖ.и^яХЇД
000001f0	83	c6	03	e8	55	ff	ba	c2	01	e8	72	ff	bf	d8	01	83	їЖ.иУяеВ.игяїШ.ї
00000200	c7	14	8a	c7	e8	1b	ff	89	05	ba	d8	01	e8	5f	ff	bf	З.ЇЗи.я%.еШ.и_яї
00000210	f2	01	83	c7	22	8b	c1	e8	19	ff	8a	c3	e8	03	ff	83	т.їЗ"<Би.яЇГи.яї
00000220	ef	02	89	05	ba	f2	01	e8	44	ff	c3	e8	45	ff	e8	ae	п.%.ет.иДяГиЕяи«
00000230	ff	32	c0	b4	4c	cd	21										я2ArLH!

Рисунок 4 – содержимое lb1\_com.com

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

«Плохой» EXE файл содержит в себе заголовок, таблицу настройки адресов, 1 сегмент, в котором находятся код и данные. Код располагается с адреса 300h, а с адреса 0 располагается заголовок EXE файла, в котором содержится сигнатура EXE файла, размер файла, размер заголовка, смещение таблицы настроек адресов и др. технические параметры.

Address	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	Dump
00000000	4d	5a	37	01	03	00	00	00	20	00	00	00	ff	ff	00	00	M27..... ..ЯЯ..
00000010	00	00	c4	bf	00	01	00	00	1e	00	00	00	01	00	00	00	..Ді.....
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	.....
00000060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
Address	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	Dump
000001b0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000001c0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000001d0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000001e0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000001f0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000200	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000210	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000220	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000230	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000240	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000250	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000260	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000270	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000280	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000290	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002a0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002b0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002c0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002d0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002e0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002f0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000300	e9	28	02	92	a8	af	20	50	43	3a	20	50	43	0d	0a	24	И('ЕІ PC: PC..\$
00000310	92	a8	af	20	50	43	3a	20	50	43	2f	58	54	0d	0a	24	'ЕІ PC: PC/XT..\$
00000320	92	a8	af	20	50	43	3a	20	41	54	0d	0a	24	92	a8	af	'ЕІ PC: AT..\$'ЕІ
00000330	20	50	43	3a	20	50	53	32	20	ac	ae	a4	a5	ab	ec	20	PC: PS2 -ЭнГем
00000340	33	30	0d	0a	24	92	a8	af	20	50	43	3a	20	50	53	32	30..\$'ЕІ PC: PS2
00000350	20	ac	ae	a4	a5	ab	ec	20	35	30	20	a8	ab	a8	20	36	-ЭнГем 50 ЭаЭ 6
00000360	30	0d	0a	24	92	a8	af	20	50	43	3a	20	50	53	32	20	0..\$'ЕІ PC: PS2
00000370	ac	ae	a4	a5	ab	ec	20	38	30	0d	0a	24	92	a8	af	20	-ЭнГем 80..\$'ЕІ
00000380	50	43	3a	20	50	91	6a	72	0d	0a	24	92	a8	af	20	50	PC: P'jr..\$'ЕІ P
00000390	43	3a	20	50	43	20	43	6f	6e	76	65	72	74	69	62	6c	C: PC Convertibl
000003a0	65	0d	0a	24	92	a8	af	20	50	43	3a	20	ad	a5	ae	af	e..\$'ЕІ PC: -ГәІ
000003b0	e0	a5	a4	a5	ab	a5	ad	2c	20	aa	ae	a4	3a	20	20	0d	aГмГәІ-, 6Эм: .
000003c0	0a	24	82	a5	e0	e1	a8	ef	20	4d	53	2d	44	4f	53	3a	.\$,ГәБЭн MS-DOS:
000003d0	20	20	2e	20	20	0d	0a	24	91	a5	e0	a8	a5	ad	eb	a9	...\$'ГәБЭ-мЭ
000003e0	20	ad	ae	ac	a5	e0	20	4f	45	4d	3a	20	20	20	48	0d	-Э-Гә OEM: H.
000003f0	0a	24	91	a5	e0	a8	a5	ad	eb	a5	20	ad	ae	ac	a5	e0	.\$'ГәБЭ-мЭ -Э-Гә
00000400	20	af	ae	ab	ec	a7	ae	a2	a0	e2	a5	ab	ef	3a	20	20	ІЭм\$ЭЭ.аГем:
00000410	20	20	20	20	20	20	48	24	24	0f	3c	09	76	02	04	07	04 Н\$\$.<.v....
00000420	30	c3	51	8a	e0	e8	ef	ff	86	c4	b1	04	d2	e8	e8	e6	OTQ\$амл+Д+.Тмк
00000430	ff	59	c3	53	8a	fc	e8	e9	ff	88	25	4f	88	05	4f	8a	лYTS\$ыл\$%OC.OБ
00000440	c7	e8	de	ff	88	25	4f	88	05	5b	c3	51	52	32	e4	33	Эд\$м\$%OC.[TQR2л3
00000450	d2	b9	0a	00	f7	f1	80	ca	30	88	14	4e	33	d2	3d	0a	ТБ..чс\$KOC.N3T=.
00000460	00	73	f1	3c	00	74	04	0c	30	88	04	5a	59	c3	b4	09	..sc<.t...OC.2YTr.
00000470	cd	21	c3	b8	00	f0	8e	c0	26	a0	fe	ff	3c	ff	74	1e	НІГә.p\$А\$.мл<ат.
00000480	3c	fe	74	20	3c	fb	74	1c	3c	fc	74	1e	3c	fa	74	20	<мт <мт.<ат.<ат
00000490	3c	f8	74	22	3c	fd	74	24	3c	f9	74	26	74	2a	ba	03	<мт"<ат\$<мт\$т*е.
000004a0	01	eb	22	80	ba	10	01	eb	22	80	ba	20	01	eb	2c	80	м\$ а а а а а

Рисунок 5 – содержимое lb1\_com.exe

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

«Хороший» EXE файл содержит заголовок, таблицу настройки адресов, затем сегмент стека, сегмент данных и сегмент кода. «Хороший» EXE файл от «плохого» отличается наличием деления на сегменты (данные находятся в своем личном сегменте).

Address	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	Dump
00000000	4d	5a	48	01	03	00	01	00	20	00	00	00	ff	ff	00	00	MZH.....яя..
00000010	00	01	8c	5d	14	01	22	00	1e	00	00	00	01	00	18	01	..Б]...".....
00000020	22	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	.....
00000060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....

Address	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	Dump
00000200	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000210	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000220	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000230	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000240	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000250	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000260	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000270	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000280	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000290	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002a0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002b0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002c0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002d0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002e0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
000002f0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000300	92	a8	af	20	50	43	3a	20	50	43	0d	0a	24	92	a8	af	'EИ PC: PC..\$'EИ
00000310	20	50	43	3a	20	50	43	2f	58	54	0d	0a	24	92	a8	af	PC: PC/XT..\$'EИ
00000320	20	50	43	3a	20	41	54	0d	0a	24	92	a8	af	20	50	43	PC: AT..\$'EИ PC
00000330	3a	20	50	53	32	20	ac	ae	a4	a5	ab	ec	20	33	30	0d	: PS2 -ОмГем 30.
00000340	0a	24	92	a8	af	20	50	43	3a	20	50	53	32	20	ac	ae	.\$'EИ PC: PS2 -О
00000350	a4	a5	ab	ec	20	35	30	20	a8	ab	a8	20	36	30	0d	0a	мГем 50 ЕкЕ 60..
00000360	24	92	a8	af	20	50	43	3a	20	50	53	32	20	ac	ae	a4	\$'EИ PC: PS2 -Ом
00000370	a5	ab	ec	20	38	30	0d	0a	24	92	a8	af	20	50	43	3a	Гем 80..\$'EИ PC:
00000380	20	50	91	6a	72	0d	0a	24	92	a8	af	20	50	43	3a	20	P'jr..\$'EИ PC:
00000390	50	43	20	43	6f	6e	76	65	72	74	69	62	6c	65	0d	0a	PC Convertible..
000003a0	24	92	a8	af	20	50	43	3a	20	ad	a5	ae	af	e0	a5	a4	\$'EИ PC: -Г0IaГм
000003b0	a5	ab	a5	ad	2c	20	aa	ae	a4	3a	20	20	0d	0a	24	82	Г«Г-, 60м: ..\$,
000003c0	a5	e0	e1	a8	ef	20	4d	53	2d	44	4f	53	3a	20	20	2e	ГaБЕн MS-DOS: ..
000003d0	20	20	0d	0a	24	91	a5	e0	a8	a9	ad	eb	a9	20	ad	ae	...\$'ГaЕ0-л0 -О
000003e0	ac	a5	e0	20	4f	45	4d	3a	20	20	20	48	0d	0a	24	91	-Гa OEM: Н..\$'
000003f0	a5	e0	a8	a9	ad	eb	a9	20	ad	ae	ac	a5	e0	20	af	ae	ГaЕ0-л0 -О-Гa I0
00000400	ab	ec	a7	ae	a2	a0	e2	a5	ab	ef	3a	20	20	20	20	20	«м\$0у.мГ«п:
00000410	20	20	48	24	00	00	00	00	00	00	00	00	00	00	00	00	Н\$......
00000420	24	0f	3c	09	76	02	04	07	04	30	c3	51	8a	e0	e8	ef	\$.<.v....0ГQБаип
00000430	ff	86	c4	b1	04	d2	e8	e8	e6	ff	59	c3	53	8a	fc	e8	я+Д±.ТижкЯТСЪм
00000440	e9	ff	88	25	4f	88	05	4f	8a	c7	e8	de	ff	88	25	4f	Ия\$%0\$.0ЛЗм0я\$%0
00000450	88	05	5b	c3	51	52	32	e4	33	d2	b9	0a	00	f7	f1	80	€.IQR2л3ТМ..чсБ
00000460	ca	30	88	14	4e	33	d2	3d	0a	00	73	f1	3c	00	74	04	K0\$.N3T=..sc<.t.
00000470	0c	30	88	04	5a	59	c3	b4	09	cd	21	c3	b8	00	f0	8e	.0\$.ZYTr.H!Гё.pн
00000480	c0	26	a0	fe	ff	3c	ff	74	1e	3c	fe	74	20	3c	fb	74	A\$.мя<ят.<ят <ят
00000490	1c	3c	fc	74	1e	3c	fa	74	20	3c	f8	74	22	3c	fd	74	..<ят.<ят <ят"<ят
000004a0	24	3c	f9	74	26	74	2a	ba	00	00	eb	38	90	ba	0d	00	\$<ят&т*е..л8.е..
000004b0	eb	32	90	ba	1d	00	eb	2c	90	ba	2a	00	eb	26	90	ba	л2.е..л.,е*.л&.е
000004c0	61	00	eb	20	90	ba	79	00	eb	1a	90	ba	88	00	eb	14	а.л .еу.л..еБ.л.
000004d0	90	bf	a1	00	83	c7	19	8a	c7	e8	4f	ff	89	05	ba	a1	.iУ.fЗ.БзиОяк.еУ
000004e0	00	eb	01	90	e8	90	ff	c3	b4	30	cd	b1	50	be	bf	00	..л..и.яГrOH!Psi..
000004f0	82	c5	0f	c8	5c	ff	58	8a	c4	82	c5	02	c8	55	ff	ba	фW м«сУБлфW мI«с

Рисунок 6 – содержимое lb1\_exe.exe

Загрузка COM модуля в основную память



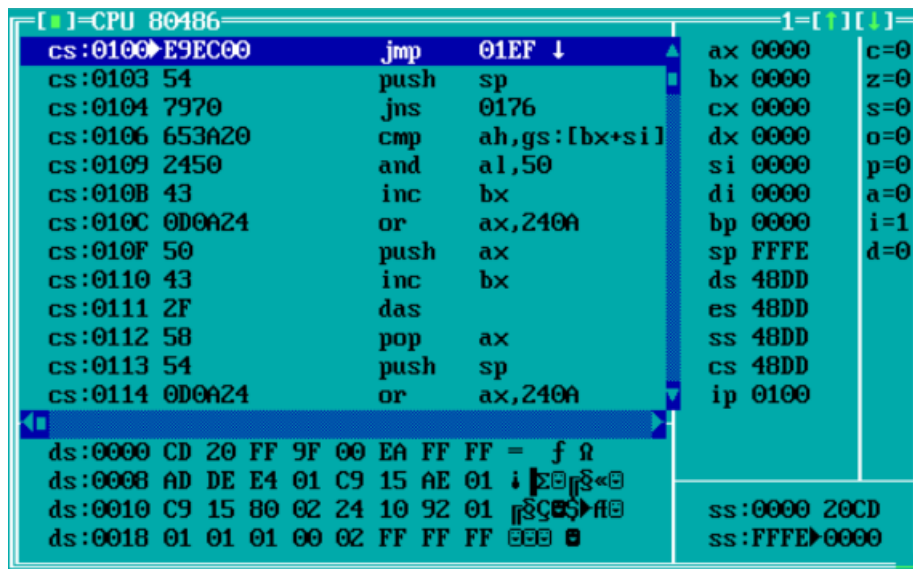


Рисунок 7 – отладчик TD.EXE с открытым COM-файлом

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

Выделяется свободный сегмент памяти достаточного размера. Затем в первые 256 байт записывается PSP, после которого следует содержание COM-файла. В стек записывается адрес начала PSP, сегментные регистры также указывают на начало PSP. Указатель стека SP принимает значение конца сегмента. IP на начало программы принимает значение 0100h. Код располагается с адреса 0100h.

2) Что располагается с адреса 0?

С адреса 0 располагается PSP ( префикс программного сегмента).

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

Сегментные регистры CS, DS, SS, ES указывают на начало PSP и имеют значение 48BD.

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

Стек в COM программе генерируется автоматически и находится в сегменте кода. В него автоматически записывается значение 0000. Указатель стека на начало программа принимает значение конца сегмента – FFFEh. Таким образом, по мере заполнения стека значение указателя будет уменьшаться и так таковой конечной границей является адрес 0000.

## Загрузка «хорошего» EXE модуля в основную память

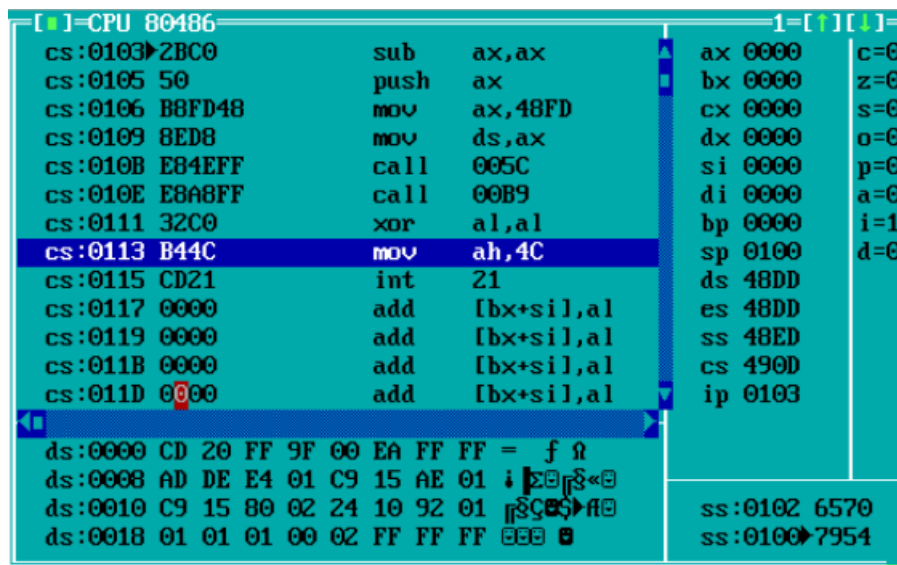


Рисунок 8 – отладчик TD.EXE с открытым EXE-файлом

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

Определяется сегментный адрес свободного участка памяти, затем создается два блока: блок памяти для переменных среды и блок памяти для PSP и программы. В блок памяти переменных среды помещается путь к файлу программы, затем записывается PSP, с помощью которого определяется сегментный адрес для загрузки программы. Далее после обработки таблицы настройки адресов, определяется абсолютные адреса сегментов. К началу выполнения программы DS и ES устанавливаются на начало сегмента PSP(48DDh), SS – на начало сегмента стека(48EDh), CS – на начало сегмента команд(490Dh). Управление передается по адресу точки входа, указанного в заголовке файла(0103h).

2) На что указывают регистры DS и ES?

В начале выполнения программы регистры DS и ES указывают на начало сегмента PSP.

3) Как определяется стек?

Стек определяется либо упрощенной директивой .STACK, которой обозначается начало сегмента стека, либо стандартной директивой SEGMENT.

Формат: ИмяСегмента SEGMENT STACK

...

ИмяСегмента ENDS

4) Как определяется точка входа?

Точка входа определяется в сегменте кода директивой END (start\_label).