

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

**ЛАБОРАТОРНАЯ РАБОТА № 7**  
**по дисциплине «Операционные системы»**  
**ТЕМА: Построение модуля оверлейной структуры.**

Студентка гр. 7381

Алясова А.Н.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2019

### **Цель работы.**

Исследование возможности построения загрузочного модуля оверлейной структуры. Исследуется структура оверлейного сегмента и способ загрузки и выполнения оверлейных сегментов. Для запуска вызываемого оверлейного модуля используется функция 4B03h прерывания int 21h. Все загрузочные и оверлейные модули находятся в одном каталоге.

В этой работе также рассматривается приложение, состоящее из нескольких модулей, поэтому все модули помещаются в один каталог и вызываются использованием полного пути.

### **Ход работы.**

1) Написала и отладила программный модуль типа .EXE, который выполняет функции:

- Проверяет, установлено ли пользовательское прерывание с вектором 09h.
- Если прерывание не установлено, то устанавливает резидентную функцию для обработки прерывания и настраивает вектор прерываний. Адрес точки входа в стандартный обработчик прерывания находится в теле пользовательского обработчика. Осуществляется выход по функции 4Ch прерывания int21h.
- Если прерывание установлено, то выводится соответствующее сообщение и осуществляется выход по функции 4Ch прерывания int 21h.

2) Написала и отладила оверлейные сегменты, которые выводят адрес сегмента, в который они загружены.

3) Написала и отладила оверлейные сегменты, которые загружаются с одного и того же адреса, перекрывая друг друга.

4) Запустила приложение из другого каталога.

5) Запустила приложение в том случае, когда оверлея нет в каталоге.

### Функции программы, переменных и структур данных.

Названия функций	Описание
Byte_To_Hex	Функция перевода байта из AL в два символа HEX
Tetr_To_Hex	Функция перевода десятичной цифры в код символа
Название переменных (тип)	Описание
_ss (dw)	Для хранения значения регистра ss
_sp (dw)	Для хранения значения регистра sp
PSP (dw)	Для хранения PSP
path (db)	Для хранения пути
dta(db)	Буфер dta
Count (db)	Счетчик
Way (db)	Хранит текст: Way:
Adress (db)	Хранит текст: Overlay segment address:
Error1 (db)	Хранит текст: Error! File not found.
Error2 (db)	Хранит текст: Error! Path not found.
file_ov1 (db)	Хранит текст: ovr1.ovl
file_ov2 (db)	Хранит текст: ovr2.ovl
endl (db)	Конец строки

### Результат работы программы.

Запуск программы.

```
A:\>?
Way: A:\ovr1.ovl
Overlay segment address: 01DE
Way: A:\ovr2.ovl
Overlay segment address: 01DE
A:\>
```

Запустила программы из другого каталога.

```
A:\>cat\7.exe  
Way: A:\CAT\ovr1.ovl  
Overlay segment address: 01DE  
Way: A:\CAT\ovr2.ovl  
Overlay segment address: 01DE  
A:\>_
```

Убрала из каталога 2 оверлей и запустила программу.

```
A:\>7  
Way: A:\ovr1.ovl  
Overlay segment address: 01DE  
Way: A:\ovr2.ovl  
Error! File not found.  
A:\>_
```

## Выводы.

В ходе данной работы были исследованы возможности построения загрузочного модуля оверлейной структуры. Рассмотрено приложение, состоящее из нескольких модулей. Реализовано приложение в возможностью запуска модуля оверлейной структуры из любого каталога.

### **Ответы на контрольные вопросы:**

1. Как должна быть устроена программа, если в качестве оверлейного сегмента использовать .COM модули?

Для выполнения такого сегмента необходимо в начале выделенной памяти сформировать блок PSP размером в 100h. При обращении к оверлейному сегменту необходимо обращаться к сегменту, смещённому на 100h, так как COM-сегмент COM-модуля-оверлея загружается без этого смещения(без смещения 100h).