Лекция №4
Тема "COM, EXE-программы"

Загрузка и выполнение программ в DOS

При загрузке программ в оперативную память DOS (дисковая операционная система) инициализирует как минимум три сегментных регистра: CS, DS и SS.

Кодовый сегмент (CS) должен обязательно описываться в программе, все остальные сегменты могут отсутствовать. В этом случае DOS при загрузке программы в оперативную память инициирует регистры DS и ES значением адреса префикса программного сегмента PSP (Program Segment Prefics) – специальной области оперативной памяти размером 256 (100h) байт.

PSP может использоваться в программе для определения имен файлов и параметров из командной строки, введенной при запуске программы на выполнение, объема доступной памяти, переменных окружения системы и т.д. Регистр SS при этом инициализируется значением сегмента, находящегося сразу за PSP, т.е. первого сегмента программы.

Распределение памяти при загрузке программы на исполнение показано на рис. 2.2.

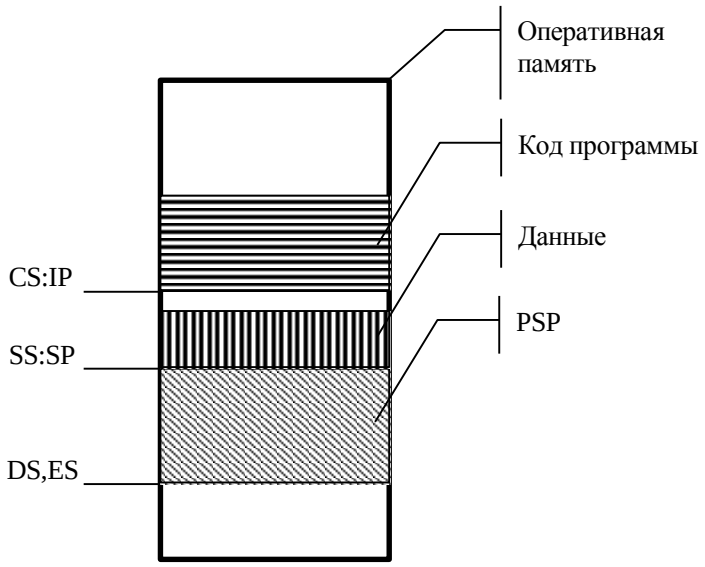

Рис. 2.2. Распределение памяти простейшей программы

После инициализации в регистре IP находится смещение первой команды программы относительно начала кодового сегмента, адрес которого помещен в регистр CS. Процессор, считывая эту команду, начинает выполнение программы, постоянно изменяя содержимое регистра IP и при необходимости CS для получения кодов очередных команд до тех пор, пока не встретит команду завершения программы. DS после загрузки программы установлен на начало PSP, поэтому для его использования в первых двух командах программы выполняется загрузка DS значением сегмента данных.

EXE- и COM-программы

Файл COM-формата – это двоичный образ кода и данных программы. Такой файл должен занимать менее 64K и не содержать перемещаемых адресов сегментов.

Файл EXE-формата содержит специальный заголовок, при помощи которого загрузчик выполняет настройку ссылок на сегменты в загруженном модуле.

Перед загрузкой COM- или EXE-программы DOS определяет сегментный адрес, называемый префиксом программного сегмента (PSP), как базовый для программы. Затем DOS выполняет следующие шаги:

– создает копию текущего окружения DOS (область памяти, содержащая ряд строк в формате ASCIIZ, которые могут использоваться приложениями для получения некоторой системной информации и для передачи данных между программами) для программы;

– помещает путь, откуда загружена программа, в конец окружения;

– заполняет поля PSP информацией, полезной для загружаемой программы (количество памяти, доступное программе; сегментный адрес окружения DOS; текущие векторы прерываний INT 22H INT 23H и INT 24H и т.д).

Программы никогда не пишутся в предположении, что они будут загружаться с определенного адреса (за исключением некоторых самозагружающихся, защищенных от копирования программ).

COM-программы. COM-программы предпочтительнее EXE-программ, когда дело касается небольших ассемблерных утилит. Они быстрее загружаются, ибо не требуется перемещения сегментов, и занимают меньше места на диске, поскольку заголовок EXE и сегмент стека отсутствуют в загрузочном модуле.

Написание EXE- программ

Программы в формате EXE могут иметь любое количество сегментов команд и сегментов данных.
Файл EXE-формата содержит специальный заголовок, при помощи которого операционная система выполняет настройку ссылок на сегменты во время загрузки программы.

Количество допустимых сегментов определяется используемой моделью памяти:

small	один сегмент кода, один сегмент данных
compact	один сегмент кода, несколько сегментов данных
medium	несколько сегментов кода, один сегмент данных
large	несколько сегментов кода, несколько сегментов данных
huge	много сегментов кода, много сегментов данных

Задание модели памяти необходимо для указания компилятору на необходимость генерации дальних ссылок и переопределения сегментов.

Во время загрузки EXE - программы:

1. Выделяется сегмент для PSP (префикс программного сегмента). Значение этого сегмента записывается в регистры ES (дополнительный сегмент) и DS (сегмент данных).
2. Затем следует сегмент программы.
3. На основании информации в заголовке EXE-программы загрузчик пересчитывает дальние ссылки с учетом реального расположения сегментов.

Таким образом, программа заранее не знает, в каких сегментах она будет выполняться. Для правильного обращения к сегменту данных используется служебный указатель @data, который содержит реальное значение сегмента данных. Поэтому для обращения к данным в начале EXE-программы необходимо загрузить в регистр DS значение этого указателя:

```
mov   ax, @data
mov   ds, ax
```

Шаблон для EXE- программы

```
%Title " Оболочка для EXE-файла "

Model Small
Stack 256

DATASEG

;----Если произойдет ошибка и программа вынуждена будет прерваться,
;---- запишите соответствующий код ошибки в exCode и выполните команду
;---- JMP Exit

exCode DB 0

;---- Здесь опишите другие переменные с помощью DB, DW и т.д..

CODESEG

Start:
mov ax, @data       ; Установка в DS адреса
mov ds, ax          ;сегмента данных, т.к. в EXE- программах
                    ; переменные хранятся отдельно от кода программы

;---- Здесь располагается программа, вызов подпрограмм и т.д.

Exit:
mov ah, 04Ch        ; Функция DOS: выход из программы
mov al, [exCode]    ; Возврат значения кода выхода
int 21h             ; Вызов DOS. Остановка программы

END Start           ; Конец программы
```

Написание COM- программ

Являются наиболее быстрыми и компактными. В этих программах и *код,* и *данные* находятся в одном сегменте, поэтому отсутствуют *межсегментные* переходы и *межсегментные* вызовы.
COM- программы быстрее загружаются в память.

Недостатком COM- программы является то, что общий размер не может превышать 64 Кбайт.

Правила создания COM- программы:

1. Указать модель памяти *tiny.*
2. Не задавать сегмент стека, так как для COM- программы стек выделяется операционной системой в конце сегмента программы.
3. Установить в начале программы значение счетчика адресов равное 100h, т.к. операционная система при загрузке программы размещает в ее первые 100h байт префикс программного сегмента (PSP - информация о количестве памяти, доступной программе, параметры командной строки и другая системная информация). Счетчик устанавливается командой ORG 100h.

- Во время загрузки COM- программы выделяется первый свободный сегмент памяти и в его начале размещается PSP.
- Все сегментные регистры устанавливаются на этот сегмент.
  CS Рег. сегмента кода
  DS Рег. сегмента данных
  SS Рег. сегмента стека
  ES Доп. сегментный
- Регистр SP (указатель стека) устанавливается на конец сегмента программы.
- В стек задается число 00h.
- Вся остальная память выделена программе. Это означает, что размер динамической памяти, с которой может работать COM- программа, может значительно превышать 64 Кбайт.

Шаблон для COM- программы

```
%Title "Оболочка для COM-файла"

Model Tiny

DATASEG

;---- Если программа будет прервана по ошибке, запишите
;---- соответствующий код ошибки в exCode и выполните команду
;---- JMP Exit

exCode DB 0
;---- Здесь опишите другие переменные с помощью DB, DW и т.д.

CODESEG

ORG 100h            ; Стандартный адрес относительно начала кодового
                    ; сегмента COM-программы

Start:
;----  Здесь располагается программа, вызов подпрограмм и т.д.

Exit:
mov ah, 04Ch        ; Функция DOS: выход из программы
mov al, [exCode]    ; Возврат значения кода выхода
int 21h             ; Вызов DOS. Остановка программы

END Start           ; Конец программы
```

Пример

Программа выводит на экран сообщение «Привет!» (файл Z2.ASM).

```
DOSSEG
MODEL SMALL
STACK 100h

DATASEG
Message  DB 'Привет!',13,10,'$'
; 13 – Enter, 10 – переход на новую строку;
; '$' – признак окончания вводимых данных

CODESEG

mov    ax,@Data
mov    ds,ax                  ; установить регистр DS таким
                              ; образом, чтобы он указывал
                              ; на сегмент данных

mov    ah,9                   ; функция DOS вывода строки
mov    dx,OFFSET Message      ; ссылка на сообщение "Привет!"
int    21h                    ; вывести "Привет!" на экран

mov    ah,4ch                 ; функция DOS завершения
                              ; программы
int    21h                    ; завершить программу
END
```

Окончание программы на ассемблере

Все программы на языке ассемблера должны передавать управление либо другим программам, либо DOS, используя для этих целей специально предназначенные команды.

Завершить программу можно следующими способами:

- через функцию 4CH (EXIT) прерывания 21H в любой момент, независимо от значений регистров;
- через функцию 00H прерывания 21H или прерывание INT 20H, когда CS указывает на PSP.

Функция DOS 4CH позволяет возвращать родительскому процессу код выхода, который может быть проверен вызывающей программой или командой COMMAND.COM "IF ERRORLEVEL".
Можно также завершить программу и оставить ее постоянно резидентной (TSR), используя либо INT 27H , либо функцию 31H (KEEP) прерывания 21H. Последний способ имеет те преимущества, что резидентный код может быть

длиннее 64K, и что в этом случае можно сформировать код выхода для родительского процесса.

Поскольку процессор работает непрерывно, программа не может просто закончиться, она должна передать управление другой программе.
Игнорирование этого действия почти всегда приводит к разрушительным результатам.
Если вы не передали управление другой программе, процессор продолжает обрабатывать содержимое памяти за физическим концом вашей программы, а это могут быть фрагменты других программ, данных или просто случайные значения, появляющиеся там после включения компьютера.
Обработка этой неизвестной информации обычно приводит к эффектному краху системы, появлению «мусор» на экране или, в наихудшем варианте, к частичному разрушению данных на диске.

Ассемблирование COM- программы

tasm Z2_COM

tlink **/t** Z2_COM Необходимо указать ключ **/t**

Ассемблирование EXE- программы

tasm Z2_EXE

tlink Z2_EXE

Контрольные вопросы

1. Основные различия COM и EXE – форматов.
2. Из каких основных этапов состоит процесс создания программы на языке ассемблера? Какие программы используются на каждом из этих этапов,
3. Что является результатом работы программы TASM?