**Федеральное государственное образовательное учреждение   
высшего образования**

**«ОРЛОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
ИМЕНИ И.С. ТУРГЕНЕВА»**

**Институт приборостроения, автоматизации и информационных технологий**

**Кафедра информационных систем и цифровых технологий**

Отчет к лабораторной работе № 3

«Команды условного и безусловного переходов. Организация ветвлений и циклов в программе»

Выполнила:

Студентка группы 21ПГ

Банных Мария Алексеевна

Приняла:

доцент кафедры   
информационных систем   
и цифровых технологий  
Конюхова Оксана Владимировна  
  
доцент кафедры   
информационных систем   
и цифровых технологий  
Амелина Ольга Викторовна

Орёл 2022 год

Ответы на контрольные вопросы:

1. Ветвления в алгоритмах. Реализация ветвлений на языке ассемблера.

Ветвление (условный переход) происходит только при соблюдении определённого условия, в противном случае выполняется следующая по порядку команда программы. Условием является признак результата выполнения предшествующей арифметической или логической команды. Каждый из признаков фиксируется в своём разряде регистра флагов PSW. Возможен и такой подход, когда решение о переходе принимается в зависимости от состояния одного из регистров общего назначения.

1. Команды условного и безусловного переходов. Каким образом вычисляются адреса переходов?

Для условного и безусловного перехода. Внутрисегментный прямой. Эффективный адрес перехода вычисляется как сумма текущего содержимого указателя команд IP и 8- или 16- битного относительного смещения.

Для безусловного перехода.

Внутрисегментный косвенный. В этом случае двухбайтовый адрес перехода размещается в ячейках памяти по некоторому адресу в сегменте данных.

Межсегментный прямой. Адрес перехода четырёхбайтовый. В адресном поле команды содержится адрес перехода – адрес, по которому размещается следующая выполняемая команда.

Межсегментный косвенный. Четырёхбайтовый адрес перехода размещается в смежных ячейках памяти по некоторому адресу в сегменте данных.

1. Циклы в алгоритмах. Организация циклов на языке ассемблера. Особенности и ограничения цикла LOOP.

Цикл можно организовать, с помощью команд условного и безусловного переходов или с помощью специальных команд. Команда LOOP и её расширения позволяет организовывать циклы, подобные циклам for в языках высокого уровня с автоматическим уменьшением счётчика цикла. Количество повторений содержится в регистре CX, который в командах управления циклами выполняет функции счётчика цикла.



1. В каком регистре находится во время выполнения программы смещение кода? Каким образом вычисляется адрес команды?

Регистр указателя команды IP – имеет размер два байта и содержит смещение следующей подлежащей выполнению команды относительно содержимого сегментного регистра CS в текущем сегменте команд. В регистр CS записывается начальный адрес сегмента, в IP – адрес команды в сегменте.

1. Какую принципиальную роль играет оператор безусловного перехода JMP при организации ветвлений?

JMP – команда условного и безусловного переходов.

1. Что означает корректное завершение программы?

Корректное завершение работы и передача управления ОС.

1. Реальный и защищённый режимы работы процессора. Вычисление физических адресов ячеек памяти (Real-мод и Рrotected-мод)

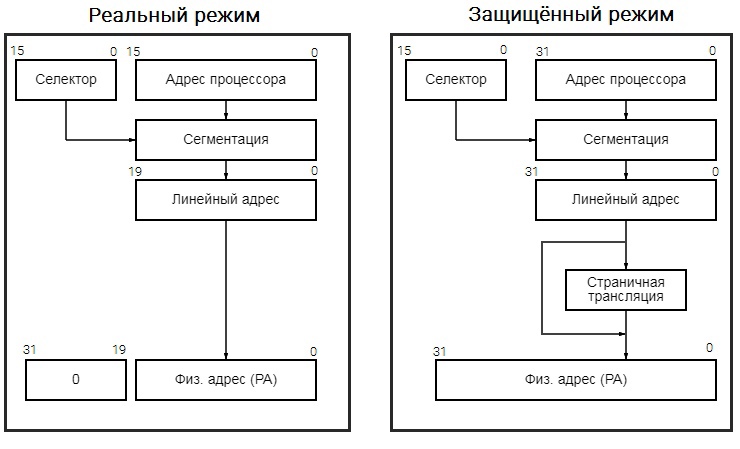
Real-мод.

В Real-моде программист имеет дело только с физической памятью. Ему доступны все инструкции процессора, прямой доступ к портам оборудования, и др.

Под смещение выделяется 16-бит, т.е. 64 Кб внутри выбранного сегмента памяти. Тогда размер сегмента в реальном режиме равен 64К, всего сегментов 16. В сумме они дают: 64Кх16=1М памяти. Но можно сделать наоборот. Тогда получим 65.536 сегментов, по 16-байт в каждом.

Рrotected-мод.

Процессор переводится в Рrotected-моду всего одним\нулевым битом в регистре CR0. Сразу же о сегментной памяти реального режима можно забыть. Хотя сегменты по-прежнему и присутствуют, они несут в себе уже другую нагрузку. Адрес теперь не состоит из двух составляющих Seg:Offs, а вся память приобретает плоский FLAT вид. Так же присутствует множество защитных механизмов.

Таким образом, память в защищённом режиме работы процессора – вся линейна. Сегментные регистры служат только для защиты доступа к памяти.

Чтобы вычислить физический (абсолютный) адрес ячейки в памяти, необходимо сложить начальный адрес сегмента и смещение.

Практическая часть:

.model small

data segment

a db 70h ;112(d)

b db 2Ah ;42(d)

nod db 0

data ends

kod segment

assume DS: data, CS: kod

begin:

mov ax, data

mov ds, ax

xor ax, ax

mov dl, a ;dl = 70(h) = 112(d)

mov ah, b ;ah = 2A(h) = 42(d)

mov cx, 3

delitel:

mov bl, dl

mov dl, ah

mov ax, bx

div dl

LOOP delitel

mov cl, dl ;cl = 0E(h) = 14(d)

xor ax, ax

xor bx, bx

xor dx, dx

mov bl, cl ;bl = 0E(h) = 14(d)

mov bh, 2 ;bh = 02(h) = 2(d)

mov cx, 4

start:

mov al, bl

div bh

add dl, ah

SHR bl, 1

LOOP start

mov ah, 4ch

int 21h

kod ends

end begin

end