Институт информационных технологий

Кафедра «Информационные системы»

ОТЧЕТ

по лабораторной работе № 3

«Исследование архитектуры и системы команд 16-разрядного процессора»

по дисциплине «Технические средства информационных систем»

Выполнил студент группы ИС/б-22-1-о

Крюкова К.М.

Проверил доцент

Чернега В.С.

Севастополь

2024

3.1 Цель работы

Исследовать систему команд, архитектуру и основные блоки процессора Intel 8086 и взаимодействие этих блоков процессора при выполнении команд разных типов. Приобрести практические навыки написания ассемблерных программ и отладки их в эмуляторе микропроцессора — экранным отладчиком типа emu8086.

3.2 Ход выполнения работы

3.3.1 После изучения методических указаний к лабораторной работе была рассмотрена программа из приложения А. Программа была откомментирована для разъяснения выполняемых действий. Код программы продемонстрирован в листинге 3.1.

Листинг 3.1 – Программа сортировки массива

mov ax, 0255; Загрузить в ax число FF(255)

inc ax; Инкрементировать ax

add ax, alpha; Добавить в ax значение alpha

nop; Осуществление задержки

mov bx, ax; Загрузка в bx значения из ax

dec bx; Декрементируем значение в регистре bx

sub bx, beta; Вычитаем beta из значения регистра bx

mov dx, bx; Загрузка в dx значения из bx

sub dx,10; Вычитание из dx числа A(10)

xchg ax, dx; Поменять местами значения ax и dx

push bx; Поместить значение bx в стек

push ax; Поместить значение ax в стек

pop cx; Загрузить в cx значение из стека(ax)

mov si, cx; Загрузить в si значение cx

mov di, dx; Загрузить в di значение dx

mov 0150h, cx; Загрузить в адрес 0150h значение cx

shl ax, 2; Логический сдвиг влево значения в ax

mov dx, offset hello; Запись в dx переменной hello

mov ax,0900h; Загрузка в ax 0900h (запись номера функции вывода

; строк)

int 21h; Вызов прерывания 21

mov ax,4c00h; Загрузка в ax 4c00h (запись номера функции

; завершения программы)

int 21h; Вызов прерывания 21

ret; Возврат к началу

alpha dw 25; Запись целого числа 25 в переменную

beta dw 32; Запись целого числа 32 в переменную

hello db 'Privet kafedra IS! '; Запись адреса начала строки в

; переменную

3.3.2 Далее была рассмотрена программа из приложения Б. Были даны пояснения в изменениях регистров для каждой команды программы:

org 100h

jmp start

v dw 12345

pak db 13,10,'Press any key...$'

start:

mov bx,[v]; В регистр bx загружается значение переменной v (3039h)

mov ah, 2; В регистр ax загружается значение 0200h

mov cx,16; В регистр cx загружается значение 0010h

lp:

shl bx,1; Сдвиг значения в регистре bx на 1 бит влево

mov dl,'0'; В регистр dx загружается значение 0030h

jnc print ;Переход, если выдвинутый бит равен 0

inc dl; В регистр dx загружается значение 0031h

print:

int 21h ;Обращение к функции DOS 02h

loop lp ;Команда цикла

mov ah,9; Запись в регистр ax значения 0900h

mov dx, offset pak; Вывод строки

int 21h;

mov ah, 8; Запись в регистр ax значения 0800h

int 21h ; Ввод символа без эха

mov ax, 4C00h; Запись в регистр ax значения 4C00h

int 21h ; Завершение программы

3.3.3 Для программ были подсчитаны их длительности выполнения. Было выделено число тактов для используемых в первой программе команд: inc ax: 3 такта, add ax, alpha: 4 такта, nop: 3 такта, mov bx, ax: 2 такта, dec bx: 3 такта, sub bx, beta: 4 такта, mov dx, bx: 2 такта, sub dx, 10: 4 такта, xchg ax, dx: 3 такта, push bx: 11 тактов, push ax: 11 тактов, pop cx: 8 тактов, mov si, cx: 2 такта, mov di, dx: 2 такта, mov 0150h, cx: 16 тактов, shl ax, 2: 2 такта, mov dx, offset hello: 4 такта, mov ax, 0900h: 4 такта, int 21h: 51 такт, mov ax, 4c00h: 4 такта, int 21h: 51 такт, ret: 10 тактов. Со знанием этой информации был произведен подсчет:

Общее кол-во тактов: 4 + 3 + 4 + 3 + 2 + 3 + 4 + 2 + 4 + 3 + 11 + 11 + 8 + 2 + 2 + 16 + 2 + 4 + 4 + 51 + 4 + 51 + 10 = 200 тактов;

При тактовой частоте работы процессора, составляющей 5МГц (5,000,000 тактов в секунду), время, потраченное на выполнение программы будет составлять: 200/5000000 ≈ 0,04 миллисекунды.

Для второй программы так же было рассчитано количество тактов для каждой операции: jmp start: 15 тактов, mov bx, [v]: 12 тактов, mov ah, 2: 2 такта, mov cx, 16: 3 такта, shl bx, 1: 2 такта (в цикле 16 раз), mov dl, '0': 3 такта (в цикле 16 раз), jnc print: 4 такта (в цикле 16 раз), inc dl: 3 такта (в цикле 6 раз), int 21h: 51 такт (в цикле 16 раз), loop lp: 17 тактов (в цикле 15 раз), mov ah, 9: 2 такта, mov dx, offset pak: 3 такта, int 21h: 51 такт, mov ah, 8: 2 такта, int 21h: 51 такт, mov ax, 4C00h: 3 такта, int 21h: 51 такт.

Со знанием этой информации был произведен подсчет:

Общее кол-во тактов: 15 + 12 + 2 + 3 + (2 \* 16) + (3 \* 16) + (4 \* 16) + (3 \* 6) + (51 \* 16) + (17 \* 15) + 2 + 3 + 51 + 2 + 51 + 3 + 51 = 1446 такта;

Время, потраченное на выполнение программы будет составлять: 1446/5000000 ≈ 0,3 миллисекунд.

**Выводы**

В ходе работы была исследована архитектура и основные блоки процессора Intel 8086 и взаимодействие этих блоков процессора при выполнении команд разных типов. Были приобретены практические навыки написания ассемблерных программ и отладки их в эмуляторе микропроцессора — экранным отладчиком типа emu8086.