МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ЛИПЕЦКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет автоматизации и информатики

Кафедра автоматизированных систем управления

ЛАБОРАТОРНАЯ РАБОТА №4

по дисциплине “Архитектура вычислительных систем”

Студент АС-21-1 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Станиславчук С. М.

(подпись, дата)

Руководитель

Уч. степень, уч. звание \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Болдырихин О. В.

(подпись, дата)

Липецк 2023

**Цель работы** – изучение сегментирования и обработки прерываний

в защищенном режиме процессоров IA-32

# Листинг программы

# 1 ; sample.asm

# 2 0000 indat segment

# 3 0000 0004 len dw 4 ;количество элементов

# 4 0002 04\*(00) mas db 4 dup(0) ;инициализируем массив входных данных

# 5 0006 08\*(00) outmas db 8 dup(0) ;массив выходных данных (для вывода в +

# 6 ASCII)

# 7 000E indat ends

# 8

# 9 0000 code segment

# 10 assume cs:code, ds:indat

# 11

# 12 0000 begin:

# 13 0000 BA 0000s mov dx, indat

# 14 0003 8E DA mov ds, dx

# 15 0005 BA B800 mov dx, 0B800h ;база видеопамяти

# 16 0008 8E C2 mov es, dx ;дополнительный сегмент - видеопамять

# 17

# 18 000A 33 FF xor di, di ;смещение в видеопамяти - левая верхняя +

# 19 позиция

# 20 000C 33 D2 xor dx, dx ;в DX будет формироваться вводимое число

# 21

# 22 ;очистка экрана

# 23 000E B8 0720 mov ax, 0720h ;07 - серый символ на чёрном фоне, 20 - ASCII-код +

# 24 пробела

# 25 0011 B9 07D0 mov cx, 80\*25 ;размер видеопамяти одного текстового +

# 26 экрана в словах

# 27 0014 F3> AB rep stosw ;цикл заполнения серым пробелом на чёрном +

# 28 фоне одного экрана

# 29

# 30 ;инициализация регистров для ввода числа

# 31 0016 33 FF xor di, di ;снова на начало видеопамяти, в левый +

# 32 верхний угол экрана

# 33 0018 33 F6 xor si, si ;счетчик массива mas, в который будет вестись +

# 34 запись

# 35 001A 8B 0E 0000r mov cx, len ;счетчик введенных чисел

# 36

# 37 ;цикл ввода

# 38 001E inloop:

# 39 001E E4 60 in al, 60h ;чтение скан-кода клавиши из порта 60h +

# 40 контроллера клавиатуры

# 41 0020 3A C3 cmp al, bl ;сравнение с предыдущим скан-кодом, если не +

# 42 равно, то клавиша отжата

# 43 0022 74 FA je inloop ;цикл повторяется, пока клавиша не будет +

# 44 отжата

# 45 0024 8A D8 mov bl, al ;скан-код в BL для последующего сравнения

# 46

# 47 0026 3C 01 cmp al, 01h ;скан-код клавиши Esc

# 48 0028 75 03 jne NotEsc

# 49 002A EB 40 90 jmp quit

# 50

# 51 002D NotEsc:

# 52 002D 3C 0B cmp al, 0Bh ;скан-код клавиши 0

# 53 002F 74 0D je num0

# 54

# 55 0031 3C 02 cmp al, 02h ;скан-код клавиш >=1

# 56 0033 7C E9 jl inloop

# 57 0035 3C 10 cmp al, 10h ;скан-код клавиш <=9

# 58 0037 7F E5 jg inloop

# 59

# 60 ; получаем число из сканкода

# 61 0039 FE C8 dec al ; клавиша 1 имеет сканкод 2, поэтому уменьшаем +

# 62 на 1

# 63 003B EB 03 90 jmp action

# 64 003E num0:

# 65 003E B0 00 mov al, 0

# 66 0040 action:

# 67 0040 04 30 add al, '0' ; преобразуем числовое значение в ASCII-код +

# 68 цифры

# 69 0042 26: 89 05 mov es:[di], ax ;видео-буфер кладем цифру на черном фоне

# 70 0045 88 84 0002r mov mas[si], al ;сохранение считанного числа в массив

# 71

# 72 0049 47 inc di ;смещение в видеопамяти на следующий символ

# 73 004A 47 inc di

# 74 004B 46 inc si ;увеличиваем счетчик массивов

# 75 004C 49 dec cx ;декремент счетчика

# 76 004D 75 CF jnz inloop ;если еще не все цифры введены, ввод +

# 77 следующей цифры

# 78

# 79 ; вывод введенного числа

# 80 004F 33 FF xor di, di ;сброс смещения

# 81 0051 BE 0000 mov si, 0 ;сброс счетчика массива

# 82 0054 8B 0E 0000r mov cx, len ;восстановление счетчика

# 83

# 84 0058 Loopit:

# 85 0058 8A 84 0002r mov al, mas[si] ;в регистр al кладем si-тый элемент из +

# 86 входного массива

# 87 005C 26: 88 05 mov es:[di], al ;выводим цифру на экран

# 88 005F 47 inc di ;смещаемся в видеопамяти

# 89 0060 46 inc si ;увеличиваем счетчик массивов

# 90 0061 E2 F5 loop Loopit ;повторяем цикл

# 91

# 92 ; выход по нажатию ESC

# 93 0063 wait\_l:

# 94 0063 E4 60 in al, 60h

# 95 0065 3C 01 cmp al, 01h

# 96 0067 75 FA jne wait\_l

# 97 0069 EB 01 90 jmp quit

# 98

# 99 006C quit:

# 100 006C B8 4C00 mov ax, 4c00h

# 101 006F CD 21 int 21h

# 102 0071 code ends

# 103 end begin

**Исследование процесса выполнения команд**

Результаты исследования работы программы представлены в таблице 1. Таблица 1 — результаты исследования.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес команды | Команда на машинном языке | Команда на языке ассемблера | Регистр команд | Изменившиеся регистры и переменные(hex) |
| 0000 | BAB348 | mov dx, indat |  | dx 48B3 |
| 0003 | 8EDA | mov ds,dx |  | ds 48B3 |
| 0005 | BA00B8 | mov dx,B800h |  | dx B800 |
| 0008 | 8EC2 | mov es,dx |  | es B800 |
| 000А | 33FF | xor di,di |  | di 0000 |
| 000C | 33D2 | xor dx,dx |  | dx 0000 |
| 000E | B82007 | mov ax,0720h |  | ax 0720 |
| 0011 | B9D007 | mov cx,80\*25 |  | cx 07D0 |
| 0014 | F3AB | rep stosw |  | cx 0000 di 0FA0 |
| 0016 | 33FF | xor di,di |  | di 0000 |
| 0018 | 33F6 | xor si,si |  | si 0000 |
| 001A | 8B0E0000 | mov cx, len |  | cx 0004 |
| #inloop | | | | |
| 001E | E460 | in al,60h |  | ax 0742 |
| 0020 | 3AC3 | cmp al,bl |  |  |
| 0022 | 74FA | je inloop |  |  |
| 0024 | 8AD8 | mov bl,al |  | bx 7642 |
| 0026 | 3C01 | cmp al, 01h |  |  |
| 0028 | 7503 | jne NotEsc |  |  |
| #NotEsc | | | | |
| 002D | 3C0B | cmp al, 0Bh |  |  |
| 002F | 740D | je num0 |  |  |
| 0031 | 3C02 | cmp al, 02h |  |  |
| 0033 | 7CE9 | jl inloop |  |  |
| 0035 | 3C10 | cmp al, 10h |  |  |
| 0037 | 7FE5 | jg inloop |  |  |
| 0020 | 3AC3 | cmp al, bl |  |  |
| 0022 | 74FA | je inloop |  |  |
| \*INPUT\* | | | | |
| 0040 | 0430 | add al, ‘0’ |  | ax 0705  bx F606 |
| 0042 | 268805 | moc byte ptr [es:di], al | 260005 | ax 0735 |
| 0045 | 88840200 | mov mas[si], al | 8884 |  |
| 0049 | 47 | inc di |  | di 0001 |

Продолжнение таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 004A | 47 | inc di |  | di 0002 |
| 004B | 46 | inc si |  | si 0001 |
| 004C | 49 | dec cx |  | cx 0003 |
| 004D | 75CF | jnz inloop |  |  |
| **001E** | **E460** | **in al, 60h** |  | **ax 0742** |
| **0020** | **33F6** | **cmp al, bl** |  |  |
| **0022** | **74FA** | **je inloop** |  |  |
| 0024 | 8AD8 | mov bl, al |  | bx F642 |
| 0026 | 3C01 | cmp al, 01h |  |  |
| 0028 | 7503 | jne NotEsc |  |  |
| 002D | 3C0B | cmp al, 0Bh |  |  |
| 002F | 740D | je num0 |  |  |
| 0031 | 3C02 | cmp al, 02h |  |  |
| 0033 | 7CE9 | jl inloop |  |  |
| 0035 | 3C10 | cmp al, 10h |  |  |
| 0037 | 7FE5 | jg inloop |  |  |
| 0058 | 8A840200 | mov al, mas[si] |  |  |
| 005C | 268805 | mov byte ptr es:[di], al |  |  |
| 005F | 47 | inc di | 26C705 |  |
| 0060 | 46 | inc si |  |  |
| 0061 | E2F5 | loop Loopit |  | cx 0000 si 0004 di 00004 |
| 0063 | E460 | in al, 60h |  |  |
| 0065 | 3C01 | cmp al, 01h |  |  |
| 0067 | 75FA | jne wait\_l |  |  |
| (вывод чисел на экран) | | | | |  |
| 006C | B8004C | mov ax, 4C00h |  | ax 4C00 |
| 006F | CD21 | int 21h |  |  |

Сравнение команд по расположению и передаче операндов, направлению обмена информацией и количеству циклов шин представлены в таблице 2.

Таблица 2 - Сравнение команд по расположению и передаче операндов, направлению обмена информацией и количеству циклов шины

|  |  |  |
| --- | --- | --- |
| Расположение операндов | Направление — только приема | Направление — приема и передачи |
| Внутри процессора | Mov ds, dx:  извлечение кода операции |  |
| Процессор - память | Mov dx,B800 :  извлечение кода операции, чтение | Mov mas[si], al: извлечение кода операции, чтение,  запись |
| Процессор - УВВ | In al, 60h:  извлечение кода операции, чтение,  ввод | Mov es:word ptr [di],0730: извлечение кода операции, чтение,  запись |

Пример работы программы приведен на рисунке 1.

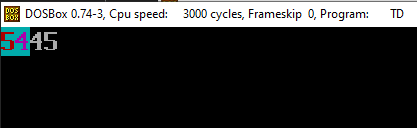
 

Рисунок 1 — Результаты выполнения программы

При вводе с клавиатуры последовательности «5445» программа «эхом» выводит введенные символы на экран.

При введении 4 символов, программа обрабатывает эту последовательность и на экран выводит «5445», что является верным результатом.

# Выводы

В ходе лабораторной работы мы исследовали взаимодействие процессора с памятью и устройствами ввода-вывода, а также изучили порядок функционирования ЭВМ.