МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ

ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Вятский государственный университет»

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

УСТРОЙСТВО УПРАВЛЕНИЯ ПАМЯТЬЮ

Отчет по лабораторной работе №3 дисциплины

«Организация памяти ЭВМ»

Вариант 4

Выполнил студент группы ИВТ-32\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Рзаев А. Э./

Проверил ассистент кафедры ЭВМ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Куваев А.С./

Киров 2018

# Задание

Таблица 1 – Варианты заданий Таблица 2 – Варианты заданий базовых адресов и атрибутов защиты базовых адресов и атрибутов защиты системных сегментов пользовательских сегментов

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Вариант 4 | | | | |  | Вариант 4 | | | | |
| № сегмента | Тип сегмента | Базовый | Размер | Атрибуты  защиты |
| № сегмента | Тип сегмента | Базовый  адрес | Размер | Атрибуты |
| 0  1  2  3  4  5  6  7 | С  К  Д  Д  С  К  Д  Д | 4016  5245  6432  0430  6330  4770  6140  7600 | 14\*  47  57  72  11\* 43  111  177 | В, Ч  З  З    В  З, Ч |
| 0  1  2  3  4  5  6  7 | Д  К  Д  К  С  Д  С  Д | 1210  0245  0432  1140  4000  2770  3140  7600 | 14  55  67  72  11\* 73  55\*  77 | В  З, Ч  В, Ч    З |

защиты

Таблица 3 – Варианты заданий номеров используемых ячеек памяти

|  |  |  |
| --- | --- | --- |
| Вариант | N | M |
| 4 | 2200 | 750 |

Таблица 4 – Варианты заданий мнемоник и адресаций для двухадресной команды

|  |  |  |  |
| --- | --- | --- | --- |
| Вариант | Команда | Адресация первого операнда | Адресация второго операнда |
| 4 | SUB | Автоинкрементная | Индексная |

Таблица 5 – Варианты заданий мнемоник и адресации для одноадресных команд

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вариант | Команда | Адресация | Команда | Адресация |
| 4 | TST | Автодекрементная | DECB | Автоинкрементная |

Таблица 6 – Варианты заданий для изменения пользовательских дескрипторов для свопинга сегментов

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № сегмента | Тип | Базовый  адрес | Размер | Атрибуты  защиты |  |
| Вариант 4 | | | |  |  |
| 4  0 | К Д | 4750  6310 | 47  51 | В |  |

# Ход работы

Содержимое регистров-дескрипторов

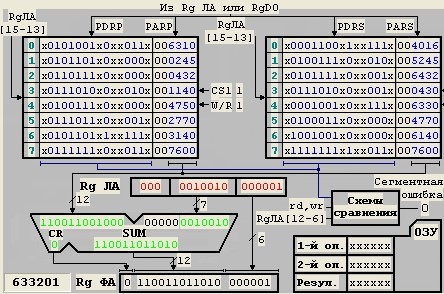


Рисунок 1 – Содержимое регистров-дескрипторов

# Содержимое таблицы IDT

Таблица 7 – Содержимое таблицы IDT

┌─────────┬─────────────────────────────────────────────────┐

│ Vector │ Number of command │

│ number │ System mode │ User mode │

├─────────┼────────────────────────┼────────────────────────┤

│ 000 │ 055 │ 132 │

│ 002 │ 174 │ 000 │

│ 004 │ 170 │ 000 │

│ 006 │ 214 │ 000 │

│ 010 │ 170 │ 000 │

│ 012 │ 220 │ 000 │

│ 014 │ 170 │ 000 │

│ 120 │ 170 │ 000 │

│ 160 │ 170 │ 000 │

│ 250 │ 170 │ 000 │

└─────────┴────────────────────────┴────────────────────────┘ Листинг входного окна

Таблица 8 – Листинг программы

┌─────┬────────────────────────────────┬────────────────────────────────┐

│ N │ Command │ Comments │

├─────┼────────────────────────────────┼────────────────────────────────┤

│ 000 │mov #4016, @#172340 │load base │

│ 001 │mov #5245, @#172342 │ │

│ 002 │mov #6432, @#172344 │ │

│ 003 │mov #0430, @#172346 │ │

│ 004 │mov #6330, @#172350 │ │

│ 005 │mov #4770, @#172352 │ │

│ 006 │mov #6140, @#172354 │ │

│ 007 │mov #7600, @#172356 │ │

│ 010 │mov #06036, @#172300 │load base rights │

│ 011 │mov #23404, @#172302 │ │

│ 012 │mov #27422, @#172304 │ │

│ 013 │mov #35022, @#172306 │ │

│ 014 │mov #04436, @#172310 │ │

│ 015 │mov #21400, @#172312 │ │

│ 016 │mov #44420, @#172314 │ │

│ 017 │mov #77426, @#172316 │ │

│ 020 │mov #1, @#177572 │enable mem manager │

│ 021 │mov #1210, @#177640 │load user base │

│ 022 │mov #0245, @#177642 │ │

│ 023 │mov #0432, @#177644 │ │

│ 024 │mov #1140, @#177646 │ │

│ 025 │mov #4000, @#177650 │ │

│ 026 │mov #2770, @#177652 │ │

│ 027 │mov #3140, @#177654 │ │

│ 030 │mov #7600, @#177656 │ │

│ 031 │mov #06026, @#177600 │load base user rights │

│ 032 │mov #26400, @#177602 │ │

│ 033 │mov #33420, @#177604 │ │

│ 034 │mov #35004, @#177606 │ │

│ 035 │mov #04436, @#177610 │ │

│ 036 │mov #35422, @#177612 │ │

│ 037 │mov #26436, @#177614 │ │

│ 040 │mov #37426, @#177616 │ │

│ 041 │mov #140000, @#177776 │user mode │

│ 042 │emt #0 │n=2200,m=750 │

│ 043 │jsr @#070 │ │

│ 044 │jsr @#106 │ │

│ 045 │trap #0 │ │

│ 046 │emt #2 │change base user mode │

│ 047 │trap #0 │again │

│ 050 │jsr @#202 │part 2.1 │

│ 051 │jsr @#227 │part 2.2 │

│ ... │ │ │

│ 055 │clr @#750 │sum │ │ 056 │add @#002200, @#750 │ │

│ 057 │add @#022200, @#750 │ │

│ 060 │add @#042200, @#750 │ │

│ 061 │add @#062200, @#750 │ │

│ 062 │add @#102200, @#750 │ │

│ 063 │add @#122200, @#750 │ │

│ 064 │add @#142200, @#750 │ │

│ 065 │add @#162200, @#750 │ │

│ 066 │rti │ │

│ 067 │ │ │

│ 070 │mov @#002200, r0 │sub, 0-1 │

│ 071 │mov @#020700, r1 │ │

│ 072 │sub (r0)+, 50(r1) │ │

│ 073 │mov @#042200, r0 │2-3 │

│ 074 │mov @#060700, r1 │ │

│ 075 │sub (r0)+, 50(r1) │ │

│ 076 │mov @#102200, r0 │4-5 │

│ 077 │mov @#120700, r1 │ │

│ 100 │sub (r0)+, 50(r1) │ │

│ 101 │mov @#142200, r0 │6-7 │

│ 102 │mov @#160700, r1 │ │

│ 103 │sub (r0)+, 50(r1) │ │

│ 104 │rts │ │

│ 105 │ │ │

│ 106 │mov @#002200, r0 │tst, 0 │

│ 107 │tst (r0)- │ │

│ 110 │mov @#022200, r0 │ │

│ 111 │tst (r0)- │ │

│ 112 │mov @#042200, r0 │ │

│ 113 │tst (r0)- │ │

│ 114 │mov @#062200, r0 │ │

│ 115 │tst (r0)- │ │

│ 116 │mov @#102200, r0 │ │

│ 117 │tst (r0)- │ │

│ 120 │mov @#122200, r0 │ │

│ 121 │tst (r0)- │ │

│ 122 │mov @#142200, r0 │ │

│ 123 │tst (r0)- │ │

│ 124 │mov @#162200, r0 │ │

│ 125 │tst (r0)- │ │

│ 126 │rts │ │

│ ... │ │ │

│ 132 │mov @#002200, r0 │decb, 0 │

│ 133 │decb (r0)+ │ │

│ 134 │mov @#022200, r0 │ │

│ 135 │decb (r0)+ │ │

│ 136 │mov @#042200, r0 │ │

│ 137 │decb (r0)+ │ │

│ 140 │mov @#062200, r0 │ │

│ 141 │decb (r0)+ │ │

│ 142 │mov @#102200, r0 │ │ │ 143 │decb (r0)+ │ │

│ 144 │mov @#122200, r0 │ │

│ 145 │decb (r0)+ │ │

│ 146 │mov @#142200, r0 │ │

│ 147 │decb (r0)+ │ │

│ 150 │mov @#162200, r0 │ │

│ 151 │decb (r0)+ │ │

│ 152 │rtt │ │

│ ... │ │ │

│ 170 │rti │system int handler │

│ ... │ │ │

│ 174 │mov #6310, @#177640 │change user base mode │

│ 175 │mov #4750, @#177650 │ │

│ 176 │mov #24426, @#177600 │change user rights │

│ 177 │mov #23400, @#177610 │ │

│ 200 │rti │ │

│ 201 │ │ │

│ 202 │mov @#002201, r1 │part 2.1, odd address │

│ 203 │;mo @#102700, r1 │illegal instr │

│ 204 │emt #6 │trace bit │

│ 205 │emt #12 │PhA gt 777776 │

│ 206 │clr @#160400 │no such device │

│ 207 │add @#142700, @#160450 │unknown mode │

│ 210 │rts │ │

│ ... │ │ │

│ 214 │mov #140020, @#177776 │trace bit, emt │

│ 215 │rti │ │

│ ... │ │ │

│ 220 │mov #7776, @#172356 │gt 777776 │

│ 221 │inc @#172356 │gt 777776 │

│ 222 │rti │ │

│ ... │ │part 2.2 │

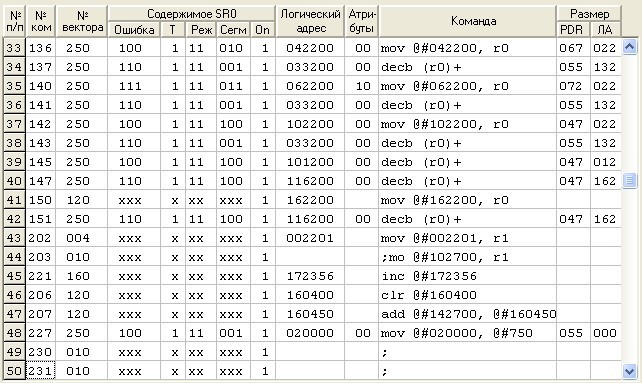
│ 227 │mov @#020000, @#750 │100 │

│ 230 │; │001 │

│ 231 │; │011 │

│ 232 │rts │ │ └─────┴────────────────────────────────┴────────────────────────────────┘ Окно результатов

Таблица 9 – Окно результатов



# Вывод

В ходе выполнения лабораторной работы была разработана программа для УУП, выполняющая загрузку регистров базовых адресов и регистров прав доступа, так же в программе были реализованы подпрограммы: нахождение суммы N-ых элементов сегментов, выполнение двухадресной команды (ADD), а также выполнение одноадресных команд (TST и DECB). Обращение к данным подпрограммам выполнялось при помощи команд EMT, JSR и TRAP. EMT – это командное прерывание для системных программ, поэтому работа происходит с системными сегментами, в отличие от команды TRAP – командное прерывание для пользовательских программ – где работа осуществляется с пользовательскими сегментами. JSR – переход к подпрограмме по абсолютному адресу.

При реализации подпрограмм необходимо было учитывать атрибуты защиты для каждого из сегментов (З – защита по записи; З, Ч – защита по записи и чтению; В – чтение из кодового сегмента запрещено; В, Ч – чтение из кодового сегмента разрешено), т.к. возможно вырабатывание вектора прерывания 250 – ошибка диспетчера памяти. У этого вектора есть несколько кодов ошибки, каждый из которых отвечает за свою исключительную ситуацию.

Кроме данного прерывания, возможны следующие: 004 – нечетный адрес (напр. @#111111), 010 – нелегальные или резервные инструкции процессора (напр. MOVE, вместо правильной MOV), 014 – внутреннее прерывание по биту трассировки T регистра PSW, 120 – обращение к неподключенному внешнему устройству, 160 – физический адрес ВУ больше 777776. В реализованной программе есть подпрограмма, в которой происходит выработка данных векторов прерываний. Для выработки прерывания 014 необходимо перейти в системный режим, а затем установить бит трассировки в единицу.

Также, есть подпрограмма, выполняющая перезагрузку некоторых регистров базовых адресов и прав доступа для пользовательского режима. Данное действие возможно только из системного режима. Если после перезагрузки этих регистров попытаться выполнить команду, где выполняется обращение (чтение или запись) к сегментным данным, то такое обращение может закончиться неудачей, т.к. теперь могут присутствовать атрибуты защиты у текущих сегментов.

Освоение данной информации и применение её на практике, при реализации программы для УУП, позволило приобрести навыки в программировании системы защиты памяти, использовании команд программных прерываний для системного и пользовательского режимов работы.