МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования **«Вятский государственный университет»**

**(ФГБОУ ВПО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

УСТРОЙСТВО УПРАВЛЕНИЯ ПАМЯТЬЮ

Отчет по лабораторной работе №5

по дисциплине

«Организация памяти ЭВМ»

Выполнила студентка группы ИВТ-31\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Кудяшев Я.Ю./

Проверил преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Мельцов В.Ю./

Киров 2022

1. **Задание**

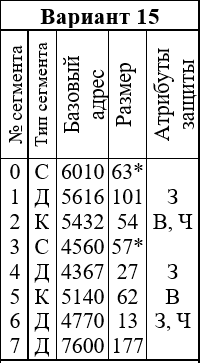
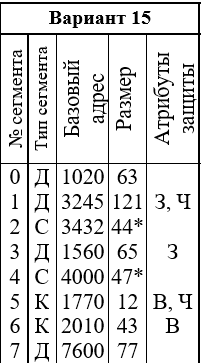


Рисунок 1 – Варианты заданий базовых адресов и атрибутов защиты системного и пользовательского режимов

|  |  |  |
| --- | --- | --- |
| Вариант | N | M |
| 15 | 3400 | 450 |

Рисунок 2 – Варианты заданий номеров используемых ячеек памяти

|  |  |  |  |
| --- | --- | --- | --- |
| Вариант | Команда | Адресация первого операнда | Адресация второго операнда |
| 15 | SUB | Автоинкрементная | Индексная |

Рисунок 3 – Варианты заданий мнемоник и адресаций для двухадресной команды

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вариант | Команда | Адресация | Команда | Адресация |
| 15 | TST | Автодекрементная | DECB | Автоинкрементная |

Рисунок 4 – Варианты заданий мнемоник и адресаций для одноадресных команд

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № сегмента | Тип сегмента | Базовый  адрес | Размер | Атрибуты защиты |
| **Вариант 15** | | | | |
| 4  0 | К  Д | 4750  6310 | 47  51 | В |

Рисунок 5 – Варианты заданий для изменения пользовательских дескрипторов для свопинга сегментов

1. **Ход работы**

Содержимое регистров-дескрипторов представлено на рисунке 6.

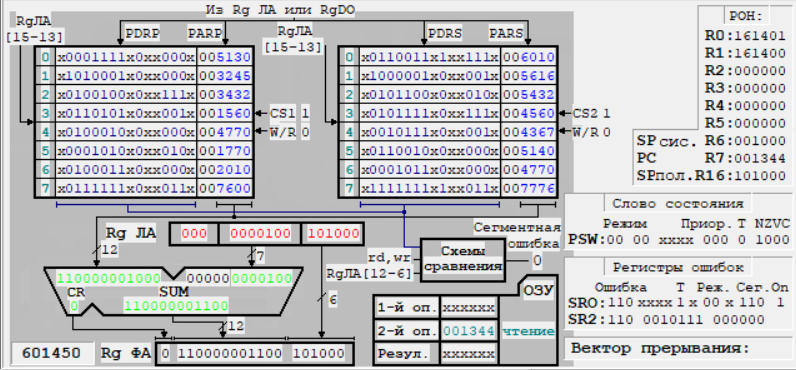


Рисунок 6 – Содержимое регистров дескрипторов

1. **Содержимое таблицы IDT**

Содержимое таблицы IDT представлено представлено на рисунке 7.

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

│ N │ Номер команды │

│ вектора │ Системный режим │ Пользовательский режим │

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

│ 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 │

└─────────┴────────────────────────┴────────────────────────┘

Рисунок 7 – Содержимое таблицы IDT

1. **Листинг входного окна**

Таблица 1 – Листинг программы

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

│ N │ Команда │ Комментарии │

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

│ 000 │mov #6010, @#172340 │Загрузка б.адреса системный │

│ 001 │mov #5616, @#172342 │Загрузка б.адреса системный │

│ 002 │mov #5432, @#172344 │Загрузка б.адреса системный │

│ 003 │mov #4560, @#172346 │Загрузка б.адреса системный │

│ 004 │mov #4367, @#172350 │Загрузка б.адреса системный │

│ 005 │mov #5140, @#172352 │Загрузка б.адреса системный │

│ 006 │mov #4770, @#172354 │Загрузка б.адреса системный │

│ 007 │mov #7600, @#172356 │Загрузка б.адреса системный │

│ 010 │mov #31436, @#172300 │Загрузка дескриптора системный │

│ 011 │mov #40422, @#172302 │Загрузка дескриптора системный │

│ 012 │mov #26004, @#172304 │Загрузка дескриптора системный │

│ 013 │mov #27436, @#172306 │Загрузка дескриптора системный │

│ 014 │mov #13422, @#172310 │Загрузка дескриптора системный │

│ 015 │mov #31000, @#172312 │Загрузка дескриптора системный │

│ 016 │mov #5420, @#172314 │Загрузка дескриптора системный │

│ 017 │mov #77426, @#172316 │Загрузка дескриптора системный │

│ 020 │mov #1, @#177572 │Включение УУП │

│ 021 │mov #1020, @#177640 │Загрузка б.адреса пользователя │

│ 022 │mov #3245, @#177642 │Загрузка дескриптора системный │

│ 023 │mov #3432, @#177644 │Загрузка дескриптора системный │

│ 024 │mov #1560, @#177646 │Загрузка дескриптора системный │

│ 025 │mov #4000, @#177650 │Загрузка дескриптора системный │

│ 026 │mov #1770, @#177652 │Загрузка дескриптора системный │

│ 027 │mov #2010, @#177654 │Загрузка дескриптора системный │

│ 030 │mov #7600, @#177656 │Загрузка дескриптора системный │

│ 031 │mov #31426, @#177600 │Зашрузка п.дескриптора │

│ 032 │mov #50420, @#177602 │Зашрузка п.дескриптора │

│ 033 │mov #22036, @#177604 │Зашрузка п.дескриптора │

│ 034 │mov #32422, @#177606 │Зашрузка п.дескриптора │

│ 035 │mov #23436, @#177610 │Зашрузка п.дескриптора │

│ 036 │mov #5004, @#177612 │Зашрузка п.дескриптора │

│ 037 │mov #21400, @#177614 │Зашрузка п.дескриптора │

│ 040 │mov #37426, @#177616 │Зашрузка п.дескриптора │

│ 041 │mov #14000, @#177776 │Установка п.режима работы ЦП │

│ 042 │emt #0 │Прерывание для сис. программ │

│ 043 │jsr @#070 │Переход к подпрограмме │

│ 044 │jsr @#106 │Переход к подпрограмме │

│ 045 │trap #0 │Прерывание для п.режима │

│ 046 │emt #2 │Меняем на польз.режим │

│ 047 │trap #0 │Прерывание для п.режима │

│ 050 │jsr @#202 │Переход к подпрограмме │

│ 051 │jsr @#227 │Переход к подпрограмме │

│ 052 │ │ │

│ 053 │ │ │

│ 054 │ │ │

│ 055 │clr @#450 │сумма sum, очистка ячейки │

│ 056 │add @#003400, @#450 │Сложение │

│ 057 │add @#023400, @#450 │Сложение │

│ 060 │add @#043400, @#450 │Сложение │

│ 061 │add @#063400, @#450 │Сложение │

│ 062 │add @#103400, @#450 │Сложение │

│ 063 │add @#123400, @#450 │Сложение │

│ 064 │add @#143400, @#450 │Сложение │

│ 065 │add @#163400, @#450 │Сложение │

│ 066 │rti │Возврат из сис.прерывания │

│ 067 │ │ │

│ 070 │mov @#003400,r0 │Вычитание sub, запись операндо │

│ 071 │mov @#020700,r1 │Запись в регистр │

│ 072 │sub (r0)+,50(r1) │Вычитание │

│ 073 │mov @#043400,r0 │Запись в регистр │

│ 074 │mov @#060700,r1 │Запись в регистр │

│ 075 │sub (r0)+,50(r1) │Вычитание │

│ 076 │mov @#103400, r0 │Запись в регистр │

│ 077 │mov @#120700, r1 │Запись в регистр │

│ 100 │sub (r0)+, 50(r1) │Вычитание │

│ 101 │mov @#143400, r0 │Запись в регистр │

│ 102 │mov @#160700, r1 │Запись в регистр │

│ 103 │sub (r0)+, 50(r1) │Вычитание │

│ 104 │rts │Возврат из подпрограммы │

│ 105 │ │ │

│ 106 │mov @#003400,r0 │tst, запись в регистр │

│ 107 │tst (r0)- │Проверка на 0 или минус │

│ 110 │mov @#023400,r0 │Запись в регистр │

│ 111 │tst (r0)- │Проверка на 0 или минус │

│ 112 │mov @#043400,r0 │Запись в регистр │

│ 113 │tst (r0)- │Проверка на 0 или минус │

│ 114 │mov @#063400,r0 │Запись в регистр │

│ 115 │tst (r0)- │Проверка на 0 или минус │

│ 116 │mov @#103400,r0 │Запись в регистр │

│ 117 │tst (r0)- │Проверка на 0 или минус │

│ 120 │mov @#123400,r0 │Запись в регистр │

│ 121 │tst (r0)- │Проверка на 0 или минус │

│ 122 │mov @#143400,r0 │Запись в регистр │

│ 123 │tst (r0)- │Проверка на 0 или минус │

│ 124 │mov @#163400,r0 │Запись в регистр │

│ 125 │tst (r0)- │Проверка на 0 или минус │

│ 126 │rts │Возврат из подпрограммы │

│ 127 │ │ │

│ 130 │ │ │

│ 131 │ │ │

│ 132 │mov @#003400, r0 │decb, запись в регистр │

│ 133 │decb (r0)+ │Вычитание 1 из регистра │

│ 134 │mov @#023400, r0 │Запись в регистр │

│ 135 │decb (r0)+ │Вычитание 1 из регистра │

│ 136 │mov @#043400, r0 │Запись в регистр │

│ 137 │decb (r0)+ │Вычитание 1 из регистра │

│ 140 │mov @#063400, r0 │Запись в регистр │

│ 141 │decb (r0)+ │Вычитание 1 из регистра │

│ 142 │mov @#103400, r0 │Запись в регистр │

│ 143 │decb (r0)+ │Вычитание 1 из регистра │

│ 144 │mov @#123400, r0 │Запись в регистр │

│ 145 │decb (r0)+ │Вычитание 1 из регистра │

│ 146 │mov @#143400, r0 │Запись в регистр │

│ 147 │decb (r0)+ │Вычитание 1 из регистра │

│ 150 │mov @#163400, r0 │Запись в регистр │

│ 151 │decb (r0)+ │Вычитание 1 из регистра │

│ 152 │rtt │Возврат из п.прерывания │

│ 153 │ │ │

│ 154 │ │ │

│ 155 │ │ │

│ 165 │ │ │

│ 166 │ │ │

│ 167 │ │ │

│ 170 │rti │Возврат из сис.прерывания │

│ 171 │ │ │

│ 172 │ │ │

│ 173 │ │ │

│ 174 │mov #5130, @#177640 │Изменение п.адресов │

│ 175 │mov #4770, @#177650 │Изменение п.адресов │

│ 176 │mov #7420, @#177600 │Изменение п.дескрипторов │

│ 177 │mov #21000, @#177610 │Изменение п.дескрипторов │

│ 200 │rti │Возврат из сис.прерывания │

│ 201 │ │ │

│ 202 │mov @#003401, r1 │Вторая часть, нечетный адрес │

│ 203 │;mo @#102700,r1 │Нелегальная инструкция ЦП │

│ 204 │emt #6 │Прерывание по биту трассивроки │

│ 205 │emt #12 │Адрес больше 777776 │

│ 206 │clr @#160400 │Отсутствие девайса │

│ 207 │add @#142700, @#160450 │Неизвестный режим │

│ 210 │rts │Возврат к подпрограмме │

│ 211 │ │ │

│ 212 │ │ │

│ 213 │ │ │

│ 214 │mov @140020, @#177776 │Запись данных │

│ 215 │rti │Возврат из сис.прерывания │

│ 216 │ │ │

│ 217 │ │ │

│ 220 │mov #7776, @#172356 │Запись данных │

│ 221 │inc @#172356 │Увеличение значения на 1 │

│ 222 │rti │Возврат из сис.прерывания │

│ 223 │ │ ││

│ 227 │mov @#020000, @#450 │Запись данных в память │

│ 230 │; │Нелегальная команда │

│ 231 │; │Нелегальная команда │

│ 232 │rts │Возврат из подпрограммы │

1. **Окно результатов**

На рисунке 8 переставлено окно результатов.

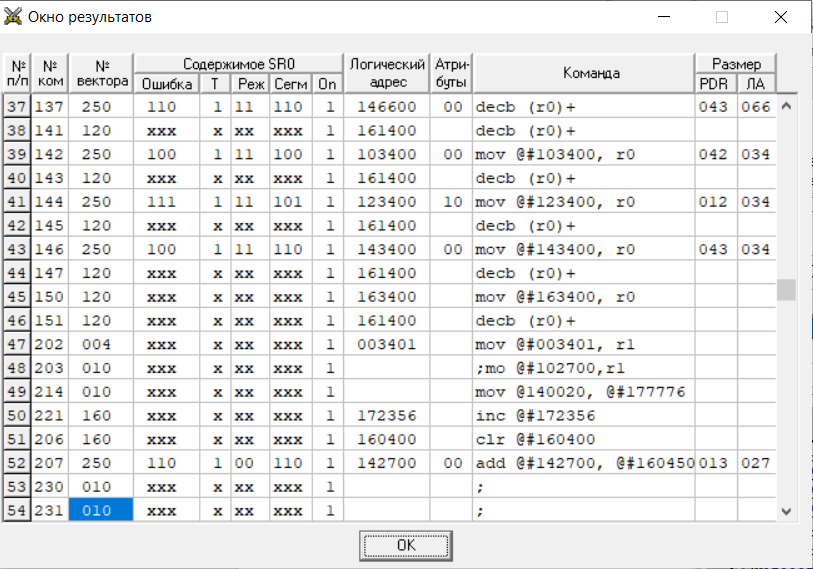


Рисунок 8 – Окно результатов

1. **Вызов прерываний**

На рисунках 9, 10, 11, 12 и 13 представлены всевозможные ситуации возникновения векторов прерывания.

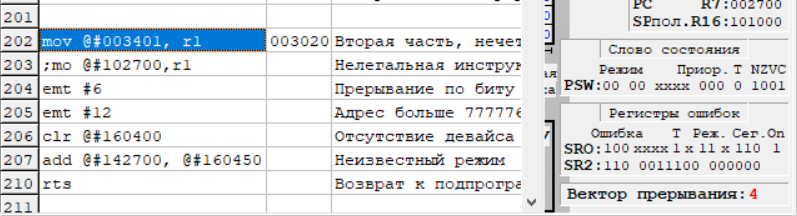


Рисунок 9 – Вызов прерывания #4 при обращении к данным с нечетным адресом

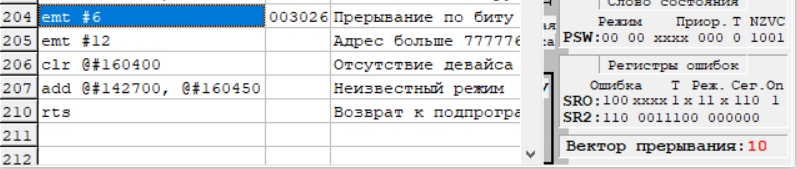


Рисунок 10 – Вызов прерывания #10 при попытке выполнения нелегальной или резервной инструкции процессора

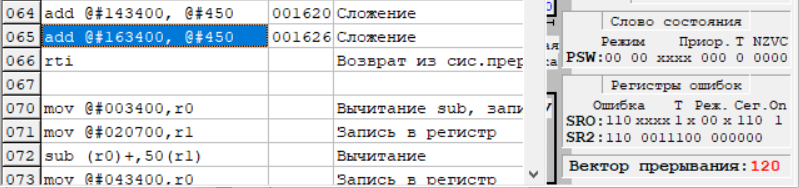


Рисунок 11 – Вызов прерывания #120 при попытке обращения к адресу неподключенного внешнего устройства

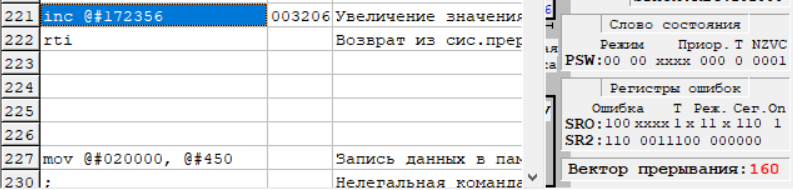


Рисунок 12 – Вызов прерывания #160 при обращении к физическому адресу внешнего устройства больше 777776

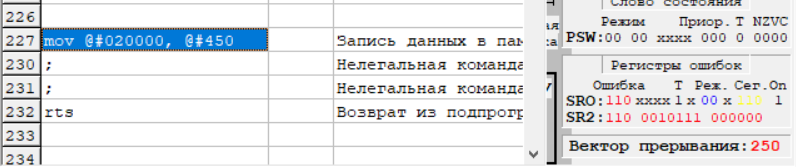


Рисунок 13 – Вызов прерывания #250 при возникновении ошибки диспетчера памяти

1. **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.