



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Московский государственный технический университет имени
Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчет по лабораторной работе №3 по курсу "Архитектура ЭВМ"

Тема Организация памяти суперскалярных ЭВМ

Студент Шавиш тарек

Группа ИУ7и-54Б

Преподаватель Попов Ю.А.

Москва — 2024 г.

Оглавление

Введение	2
1 Основные теоретические сведения	3
2 Эксперименты	4
2.1 Задание 1.	4
2.2 Задание 2.	5
2.3 Задание 3.	5
2.4 Задание 4.	6
2.5 Задание 5.	7
2.6 Задание 6.	8
Заключение	10

Введение

Целью данной работы является освоение принципов эффективного использования подсистемы памяти современных универсальных ЭВМ, обеспечивающей хранение и своевременную выдачу команд и данных в центральное процессорное устройство. Работа проводится с использованием программы для сбора и анализа производительности PCLAB.

В ходе работы необходимо ознакомиться с теоретическим материалом, касающимся особенностей функционирования подсистемы памяти современных конвейерных суперскалярных ЭВМ, программы PCLAB, изучить средства идентификации микропроцессоров, провести исследования времени выполнения тестовых программ, выводы о архитектурных особенностях используемых ЭВМ.

1 Основные теоретические сведения

Программа PCLAB предназначена для исследования производительности x86 совместимых ЭВМ с IA32 архитектурой, работающих под управлением операционной системы Windows (версий 95 и старше). Исследование организации ЭВМ заключается в проведении ряда экспериментов, направленных на построение зависимостей времени обработки критических участков кода от изменяемых параметров. Набор реализуемых программой экспериментов позволяет исследовать особенности построения современных подсистем памяти ЭВМ и процессорных устройств, выявить конструктивные параметры конкретных моделей ЭВМ.

Процесс сбора и анализа экспериментальных данных в PCLAB основан на процедуре профилировки критического кода и измерении времени его обработки центральным процессорным устройством. При исследовании конвейерных суперскалярных процессорных устройств, таких как 32-х разрядные процессоры фирмы Intel или AMD, способных выполнять переупорядоченную обработку последовательности команд программы, требуется использовать специальные средства измерения временных интервалов и за-преращения переупорядочивания микрокоманд.

2 Эксперименты

Для исследования производительности был проведен ряд экспериментов, которые представлены ниже.

2.1 Задание 1

Цель эксперимента – определение способа трансляции физического адреса, используемого при обращении к динамической памяти.

Суть эксперимента – для определения способа трансляции физического адреса при формировании сигналов выборки банка, выборки строки и столбца запоминающего массива применяется процедура замера времени обращения к динамической памяти по последовательным адресам с изменяющимся шагом чтения. Для сравнения времен используется обращение к одинаковому количеству различных ячеек, лежащих друг от друга на определенный шаг. Результат эксперимента представляется зависимостью времени (или количества тактов процессора), потраченного на чтение ячеек, от шага чтения.

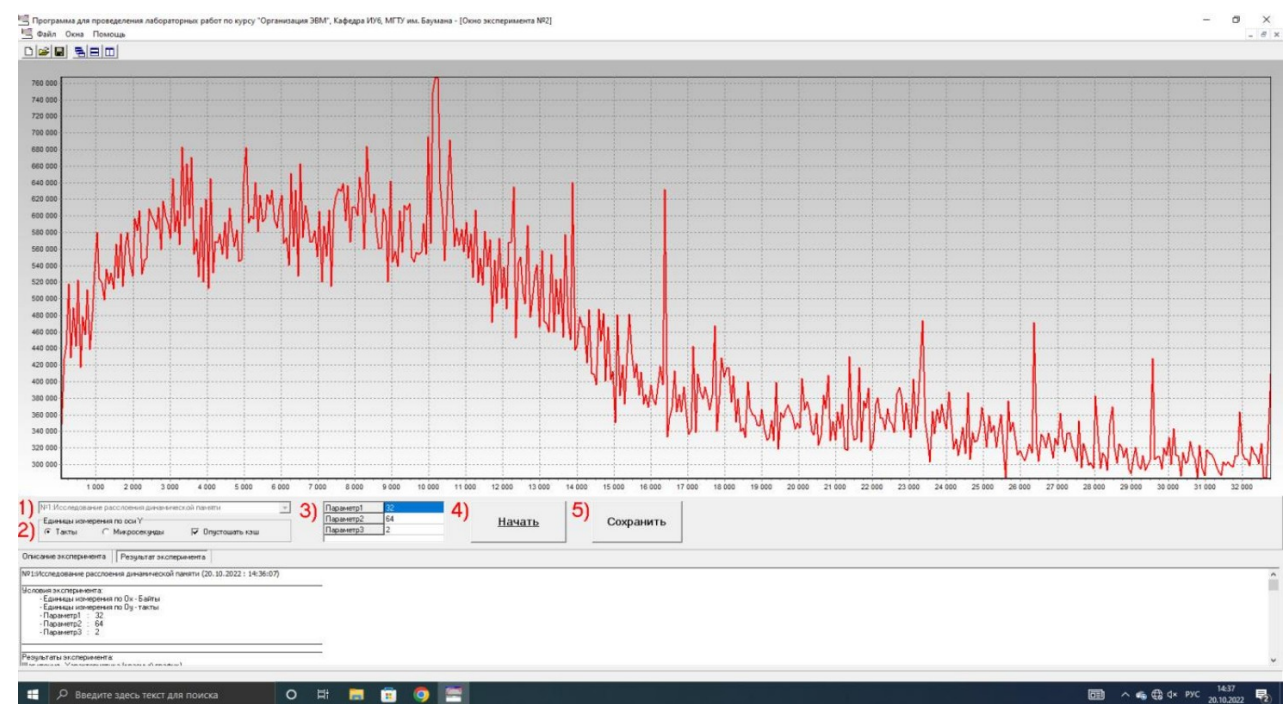


Рис. 2.1: Исследования расслоения динамической памяти

2.2 Задание 2

Цель эксперимента: оценка влияния зависимости команд по данным на эффективность вычислений.

Суть эксперимента: для сравнения эффективности векторных и списковых структур в эксперименте применяется профилировка кода двух алгоритмов поиска минимального значения. Первый алгоритм использует для хранения данных список, в то время как во втором применяется массив. Очевидно, что время работы алгоритма поиска минимального значения в списке зависит от его фрагментации, от среднего расстояния между элементами списка.

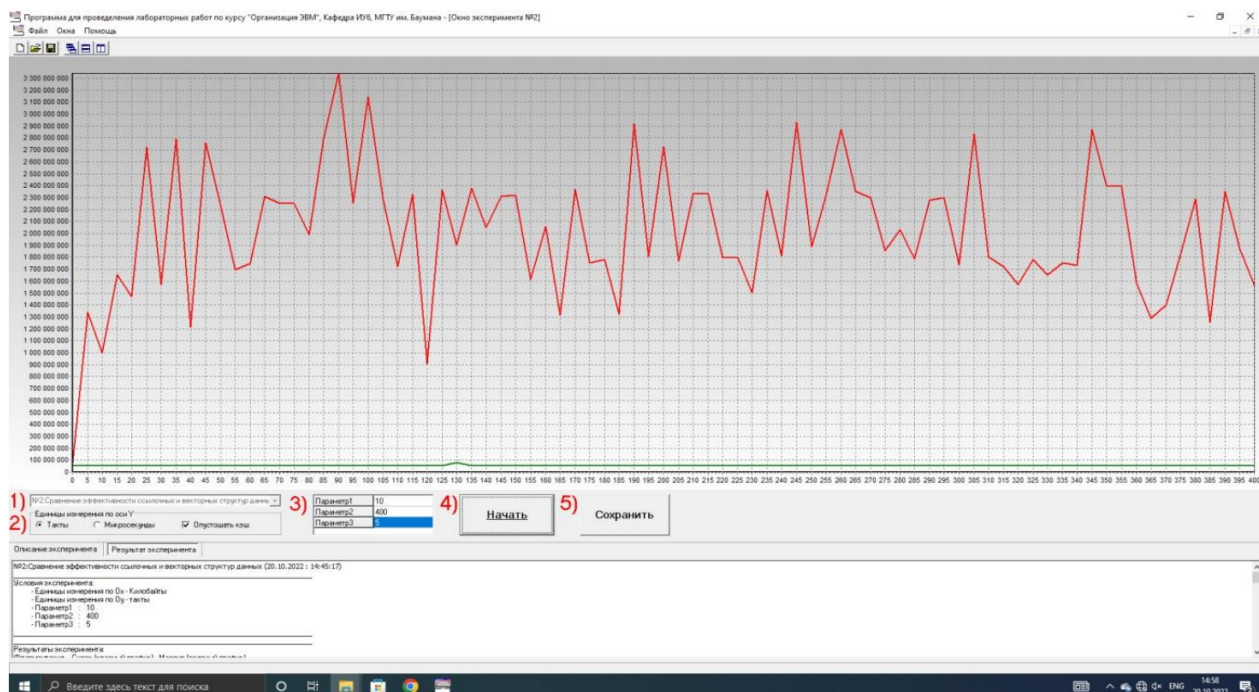


Рис. 2.2: Сравнение эффективности ссылочных и векторных структур

2.3 Задание 3

Цель эксперимента: выявление способов ускорения вычислений благодаря применению предвыборки данных.

Суть эксперимента: эксперимент основан на замере времени двух вариантов подпрограмм последовательного чтения страниц оперативной

памяти. В первом варианте выполняется последовательное чтение без дополнительной оптимизации, что приводит к дополнительным двойным обращениям. Во втором варианте перед циклом чтения страниц используется дополнительный цикл предвыборки, обеспечивающий своевременную загрузку информации в TLB данных.

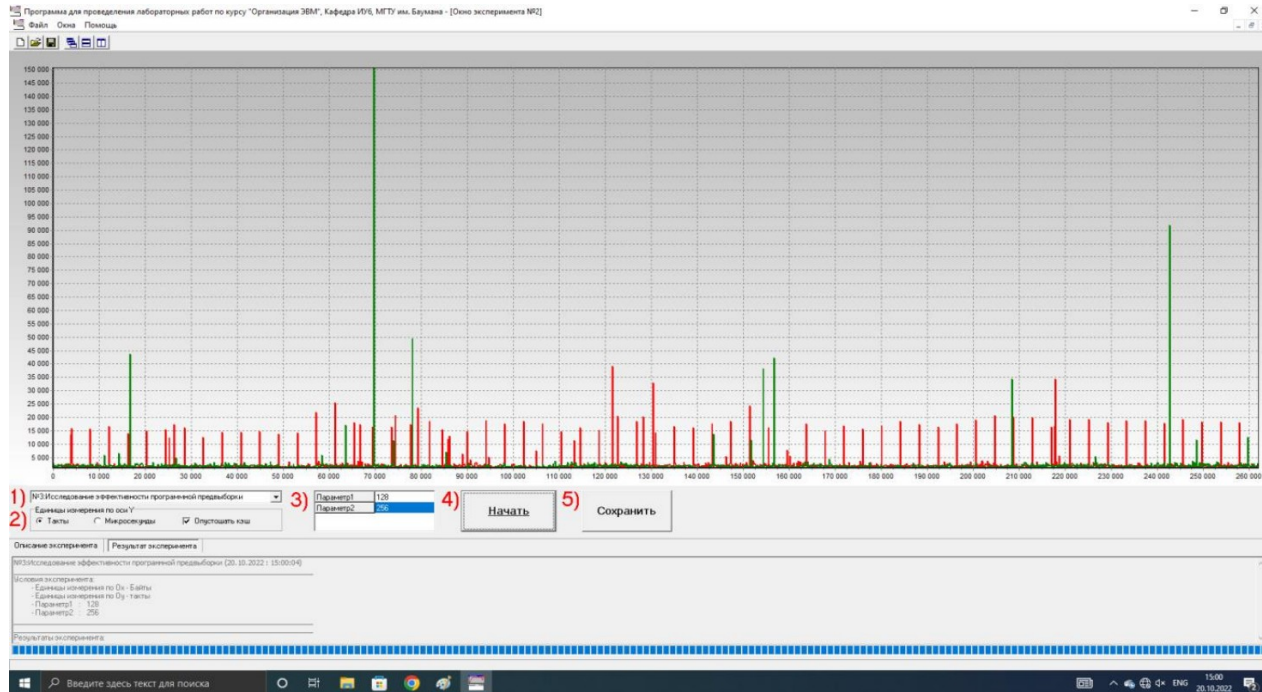


Рис. 2.3: Исследование эффективности программной предвыборки

2.4 Задание 4

Цель эксперимента: исследование возможности ускорения вычислений благодаря использованию структур данных, оптимизирующих механизм чтения оперативной памяти.

Суть эксперимента: для сравнения производительности алгоритмов, использующих оптимизированные и неоптимизированные структуры данных используется профилировка кода двух подпрограмм, каждая из которых должна выполнить обработку нескольких блоков оперативной памяти. В алгоритмах обрабатываются двойные слова данных (4 байта), существенно меньше размера пакета (32-128 байт). Неоптимизированный вариант структуры данных представляет собой несколько массивов в оперативной памяти, в то время как оптимизированная структура состоит из

чередующихся данных каждого массива.

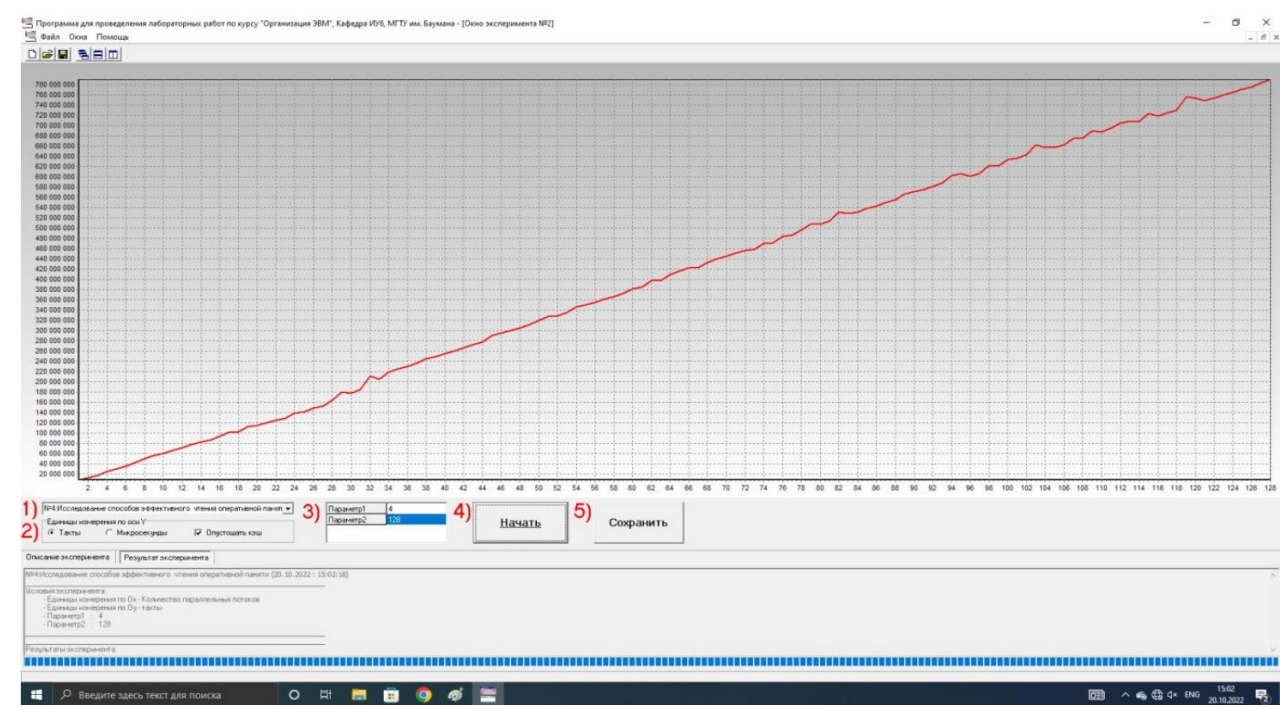


Рис. 2.4: Исследование способов эффективного чтения оперативной памяти

2.5 Задание 5

Цель эксперимента: исследование влияния конфликтов кэш-памяти на эффективность вычислений.

Суть эксперимента: для определения степени влияния конфликтов в кэш-памяти на эффективность вычислений используется профилировка двух процедур чтения и обработки данных. Первая процедура построена таким образом, что чтение данных выполняется с шагом, кратным размеру банка. Это порождает постоянные конфликты в кэш-памяти. Вторая процедура оптимизирует размещение данных в кэш с помощью задания смещения востребованных данных на некоторый шаг, достаточный для выбора другого набора. Этот шаг соответствует размеру линейки.

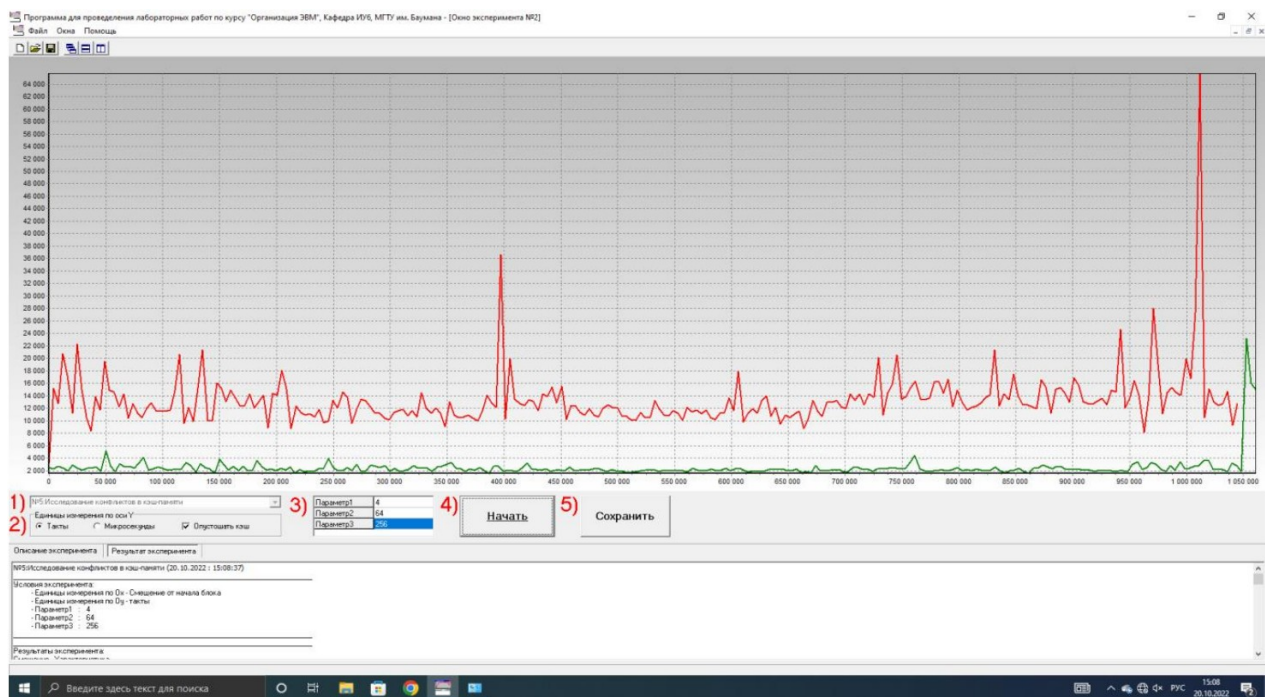


Рис. 2.5: Исследование конфликтов в кэш-памяти

2.6 Задание 6

Цель эксперимента: исследование способов эффективного использования памяти и выявление наиболее эффективных алгоритмов сортировки, применимых в вычислительных системах.

Суть эксперимента: эксперимент основан на замере времени трех вариантов алгоритмов сортировки (Quick Sort, Radix-Counting Sort, оптимизированный Radix-Counting Sort).



Рис. 2.6: Сравнение алгоритмов сортировки

Заключение

В ходе выполнения данной лабораторной работы были освоены принципы эффективного использования подсистемы памяти современных универсальных ЭВМ, обеспечивающей хранение и своевременную выдачу команд и данных в центральное процессорное устройство.