Федеральное агентство связи (Россвязь)

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Кафедра вычислительных систем

**РАСЧЁТНО-ГРАФИЧЕСКОЕ ЗАДАНИЕ**

по дисциплине «Архитектура вычислительных систем»

Вариант № 10

Выполнил:

студент гр. ИС-441 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Е.В. Маклашкина /

подпись

Проверил:

доцент кафедры ВС

к.т.н. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /А.В. Ефимов /

ОЦЕНКА, подпись

Новосибирск 2016

ОГЛАВНЕНИЕ

[1.1 Задание №1 3](#_Toc470370597)

[1.2 Ответ на задание 1 3](#_Toc470370598)

[2.1 Задание №2 9](#_Toc470370599)

[2.2 Ответ на задание 2 9](#_Toc470370600)

# **1.1 Задание №1**

**Оценить архитектурные возможности модели вычислителя. Привести пример суперВС, в которой используется модель вычислителя.**

# **1.2 Ответ на задание 1**

В технике «вычислитель» - это средство обработки информации (например, арифмометр, электронный калькулятор, процессор, ЭВМ), работа которого основывается на примитивной имитации деятельности человека, занятого расчетами. Процесс обработки информации на вычислителе требует участия человека - оператора. Чем шире функциональные возможности вычислителя, чем выше уровень или механизации, или автоматизации вычислений, тем ниже частота взаимодействий между ним и оператором. Среди вычислителей как технических средств наивысшей степенью автоматизации счета обладает ЭВМ.

Итак, состав устройств и структура ЭВМ есть результат технической интерпретации функциональной организации человека вычислителя. Процесс обработки информации (решения задач) на ЭВМ, по сути, сводится к имитации вычислительной деятельности человека. Следовательно, концептуальную основу конструкции ЭВМ и ее функционирования (или же основу автоматизации вычислений на ЭВМ) должна составить модель вычислителя.

Модель вычислителя есть пара:

*c = <h, a>,*

где h и a — описания конструкции вычислителя и алгоритма его функционирования при обработке информации (или коротко: *h* — конструкция вычислителя, *a* — алгоритм его работы). Конструкция вычислителя допускает следующее представление:

*h = <U, g>,*

где U = {} – множество устройств , i ϵ {1,2, ..., k} (k = 5 для концептуальной машины Дж. фон Неймана); *g* – описание структуры (или просто структура) сети связей между устройствами . Иначе говоря, структура вычислителя это граф, вершинам которого сопоставлены устройства , a ребрам линии связей между ними. В основе конструкции вычислителя лежат следующие три принципа:

1. последовательная обработка информации, т. e. последовательное выполнение:

— операций на множестве U устройств , взаимодействующих через связи структуры *g*;

— микроопераций в пределах устройств , i =);

2) фиксированность (автоматическая неизменяемость) структуры (и g, и микроструктуры устройств ϵ U);

3) неоднородность составляющих устройств ( ϵ U, i =) и связей между ними (структуры *g*). Алгоритм функционирования вычислителя обеспечивает согласованную работу всех устройств (множества U) и связей между ними (структуры g) процессе обработки информации или, говоря иначе, при решении задач. Для решения любой задачи вычислитель должен иметь исходные данные *D* и программу *p* или запись алгоритма вычислений (на одном из возможных языков). Поэтому алгоритм *a* допускает представление в виде суперпозиции:

*a(P(D))*

Для заданных *D* и *p* алгоритм должен приводить к однозначному результату. Степень универсальности алгоритма работы вычислителя определяется разнообразием классов решаемых задач.

Итак, на основании *c = <h, a>* и *h = <U, g>* модель вычислителя

*с= <U, g, a(p(D))>,*

где U - множество устройств, обеспечивающих ввод, обработку, хранение и вывод информации; *g* структура связей между устройствами; *а* алгоритм работы вычислителя или алгоритм управления вычислительными процессами при реализации программы *p* обработки данных *D*. B модель вычислителя вкладывается каноническая ЭВМ Дж. фон Неймана.

Следует отметить, что описанные три принципа конструирования ЭВМ соответствовали уровню развития ВТ лишь в 50-x годах ХХ столетия, они позволили создать первые технико-экономически эффектные электронные машины. В последующих ЭВМ, основанных на новой элементной базе, технико-экономическая эффективность машин была достигнута уже за счет совмещения операций во времени их выполнения, ручной реконфигурируемости структур, возможности изменения (upgrade) составов машин. Каждых новый проект ЭВМ характеризовался очередной модификацией принципов построения, смена поколений ЭВМ сопровождалась все большим отходом от трех первоначальных принципов. B конце концов создатели средств обработки информации пришли к необходимости применения диалектических противоположностей названных здесь принципов. Таким образом, мы можем дать еще одно определение: *средство обработки информации, основанное на модели вычислителя, называется ЭВМ.* Процесс проектирования ЭВМ включает в себя выбор системы счисления и формы представления данных *D*; определение средств для написания программ *p* вычислений; подбор состава U вычислительных устройств и системы операций, реализуемых ими; формирование структуры *g* и разработку микроструктуры («логический» синтез) устройств ϵ U; выбор элементной базы и конструирование устройств ϵ U; построение такого алгоритма *a* функционирования вычислителя *c*, который обеспечивал бы реализацию и программ *p*, и, в частности, операции как последовательности микрооперации.

Допустимы аппаратурные, аппаратурно-программные и программные реализации модели вычислителя *с= <U, g, a(p(D))>*. Аппаратурное исполнение вычислителя *c* предопределяет каноническая ЭВМ Дж. фон Неймана; такое исполнение соотносится c первыми ЭВМ (первым поколением ЭВМ). Однако здесь уместно заметить, что даже в этих машинах имели место эволюционные модификации. Так, в машине JONIAC в отличие от EDVAC уже осуществлялась параллельная обработка всех разрядов слова (что может рассматриваться как «параллельное выполнение микроопераций»). B последующих разработках ЭВМ закладывалась и возможность совмещения операции.

Аппаратyрно-прогpаммнaя реализация *c*, включая конструкцию *h* и алгоритм *a* (если учесть микропрограммное управление), сопоставляется c современными ЭВМ. Имеет место тенденция к вложению функций системного программного обеспечения в аппаратуру. Последнее поддерживается непрерывным совершенствованием технологии БИС, удешевлением элементной базы (в современных условиях - микропроцессоров).

Программное исполнение *c* следует воспринимать как машинный имитатор средства обработки информации, основанного на модели вычислителя. Говоря иначе, при программном исполнении модели вычислителя порождается виртуальная ЭВМ (или машинная модель ЭВМ).

Развитие ВТ по пути создания ЭВМ (как аппаратурных или аппаратурно-программных реализаций модели вычислителя или функциональной структуры машины Дж. фон Неймана) может осуществляться в ограниченных пределах, обусловленных, в частности, конечностью скорости распространения сигналов в физических средах (конечностью скорости света, которая в вакууме равна (299 792 ± 0,4) км/с).

Использование модели вычислителя для построения высокопроизводительных вычислительных средств ограничено теоретическими и техническими пределами скорости выполнения операций (возможностями элементной базы и фундаментальными физическими законами).

Отсутствие возможности автоматического изменения структуры не позволяет в полной мере адаптировать ЭВМ к области применения (подобрать адекватную структуру и режим обработки), учесть особенности и характеристики задач при их программировании и т. п. Жесткий алгоритм управления вычислительными процессами вынуждает создавать программы решения задач, в которых фиксированы последовательность выполнения операции и порядок использования данных. Такая фиксированность сохраняется при повторных решениях задач даже c другими массивами данных (независимо от их объемов и структур, связанных c физической сущностью задач). Жесткость структуры ЭВМ в ряде случаев приводит к значительным трудностям программирования задач и не позволяет использовать эффективные методы их решения. Говоря иначе, фиксированность структуры ЭВМ однозначно приводит к процедурному обработки информации, вариации в ходе реализации программ ограничены возможностями команд условных переходов.

Существовавшее в 40-x годах ХХ в. представление o методах обработки информации и об организации вычислительных процессов, острая потребность в средствах автоматизации вычислителей, уровни развития электроники и техники для вычислений, a также экономические ограничения однозначно определили структуру и состав приемлемого средства автоматической обработки данных ЭВМ дж. фон Неймана. Конструктивный принцип неоднородности в машине дж. фон Неймана реализован на нескольких уровнях: структура ЭВМ в целом нерегулярна, a состав гетерогенный: каждое из пяти устройств (см. рис. 2.1) имеет свое функциональное назначение и свою логическую организацию, основывается на специфических принципах, обладает своими особенностями технической реализации. B пределах всей конфигурации ЭВМ однородность достигается лишь фрагментарно (например, оперативная память допускает однородную реализацию). Построение неоднородных ЭВМ находится в резком противоречии c тенденцией развития микро- и наноэлектроники. Таким образом, путь эволюционного совершенствования средств ВТ на основе модели вычислителя (принципов последовательной обработки информации, фиксированности структуры и конструктивной неоднородности) не даст кардинального улучшения их технических характеристик.

**Пример суперВС:**

Cray-1 — один из первых суперкомпьютеров. Пиковая производительность — 133 Мфлопса. Cray-1 — первый суперкомпьютер компании Cray Research, созданной "отцом суперкомпьютеров" — Сеймуром Крэем — после ухода из компании CDC.

В данной ВС, С.Крэй отказался от транзисторов в пользу больших интегральных микросхем, которые давали такую плотность упаковки логических элементов при высокой надежности, которую невозможно было достичь с помощью транзисторов. Это позволило снизить тактовую частоту до 12.5 нс (80 МГц), вместо абмициозных 8 нс (125 МГц) в CDC 8600 без потери производительности. Во-вторых, он отказался от многопроцессорной системы в пользу векторного процессора, как у проекта-конкурента CDC STAR-100.

Далее Крэй учел недостатки STAR-100. Компьютеру во время исполнения программы требуется выполнять как векторные, так и скалярные вычисления. STAR-100 показывал высокую скорость на векторных вычислениях, но был медленным в скалярных. Из-за этого мощь STAR-100 проявлялась только на специальных задачах, где требовалась обработка больших массивов данных. Для Cray-1 Сеймур Крэй построил процессор, который быстро выполнял и скалярные и векторные вычисления. Этого удалось добиться через создание так называемых "векторынх регистров" - модулей памяти небольшого объема, которые располагались близко к процессору и работали очень быстро (но стоили очень дорого). Таким образом центральный процессор брал данные из регистров и записывал данные тоже в регистры, реализуя новый принцип работы с памятью "регистр-регистр", в то время как CDC STAR-100 использовал прежний способ работы с памятью — "load-store", т.е. чтение и запись в память (которая была медленной) напрямую. В CDC STAR-100 основная память была на ферромагнитных сердечниках, а в Cray-1 для памяти использовались полупроводники. Кроме того CDC STAR-100 строился совместимым с предыдущими моделями компании CDC 6600 и CDC 7600, а Cray-1 начинался с нового листа, и совместимости с предыдущими моделями не требовалось, что значительно облегчало задачу Крэю. В 1974 году первые тесты машины показали производительность 80 MFLOPS.

ОП (от 1 до 4 мегаслов), большой набор процессорных регистров, состоящих из группы векторных регистров по 64 элемента, блок скалярных регистров, блок адресных регистров. Каждая группа регистров связана со своим конвейерным процессором.

Данная система могла выполнять скалярные операции над векторными данными, над адресами, числами с плавающей запятой (порядок — 15, мантисса — 49). Быстродействие 180 млн операций в секунду с плавающей запятой. В данной ВС используются команды длиной 16 или 32 разряда. В коротких командах 7 разрядов выделяется под код операции, 3 адресных поля по 3 разряда, определяли номер регистра для хранения операндов. В длинных — 22 разряда для того, чтобы можно было найти операнд в общем поле ОП. Один из регистров определяет длину вектора, второй — регистр маски.

Центральный процессор Cray-1 состоял из 500 печатных плат, на каждой их которых с обеих строн располагалось по 144 микросхемы. Всего получалось 144.000 микросхем, которые охлаждались фреоном. Для лучшего охлаждения и циркуляции фреона в охладительной системе центральный процессор был выполнен в стиле "башни" c 12 колоннами, составенными в форме дуги длиной 270 градусов (в виде буквы "C" - от "Cray", если смотреть сверху), а охладительная система была расположена в основании этой башни. Так был создан характерный, оригинальный и узнаваемый вид компьютера, напоминающий диван.

# **2.1 Задание №2**

Построить блок-схему p -алгоритма умножения матриц:

L [1: X; 1: Y], U [1: Y; 1: Z],

обеспечивающего распределение элементов результирующей матрицы (C) по горизонтальным полосам в элементарных машинах ВС.

Отыскать максимум коэффициента ε накладных расходов при реализации p -алгоритма на вычислительной системе, имеющей следующие параметры:

– разрядность l = 32;

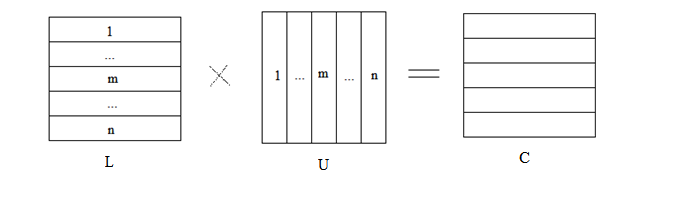
– полосу пропускания канала между машинами ν = 1 Мегабод;

– время выполнения операции сложения = 1 мкс;

– время выполнения операции умножения = 10 мкс.

# **2.2 Ответ на задание 2**

При параллельной обработке необходимо, чтобы каждый вычислитель рассчитывал n-ую часть матрицы C, при этом следует заметить, что размещение матриц L и U требует большой суммарной емкости памяти, и необходимо каждому вычислителю выделить n-ую часть матрицы L и n-ую часть матрицы U, т.е. необходимо произвести однородное распределение элементов этих матриц по вычислителям. Это можно сделать, разрезав матрицу L на n горизонтальных частей, а U на n вертикальных. Каждую из таких полос будем размещать в одном вычислителе.



Вычислительный процесс можно организовать следующим образом: сначала 1-й вычислитель высылает свою 1-ю строку из полосы матрицы L всем остальным вычислителям. После этого все вычислители параллельно рассчитывают все свои элементы матрицы C (первой строки). На каждом вычислителе хранятся строки L и столбцы U. Затем первый вычислитель высылает всем вторую строку из своей полосы матрицы L, после чего все вычислители занимаются расчетом своих элементов второй строки матрицы С. Рассылка строк из первого вычислителя заканчивается после передачи остальными вычислителями n. Затем рассылкой начинает заниматься второй вычислитель и так далее, пока процесс не завершится после рассылки обработки последней строки из полосы матрицы L n-ого вычислителя. Применяя такое однородное распределение исходных данных и процедуру обработки получили параллельный алгоритм, состоящий из идентичных ветвей.

?

Нет

Нет

Нет

,?

*լ := α*

*?*

*i = i + 1*

*i := 1*

*α := 0*

Нет

аа

Прием

Передача

Да

аа

Коэффициент накладных расходов:

1 Мегабод = бод

1 мкс = с

10 мкс = с