



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ «РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

**ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
ТЕХНОЛОГИЧЕСКОГО ОБРАЗОВАНИЯ**
Кафедра информационных технологий и электронного обучения

Основная профессиональная образовательная программа
Направление подготовки 09.03.01 Информатика и вычислительная техника
Направленность (профиль) «Технологии разработки программного
обеспечения»
форма обучения – очная

АНАЛИТИЧЕСКИЙ ОБЗОР

по теме: Архитектура ЭВМ

Обучающегося 4 курса
Киняева Ильи Константиновича

Санкт-Петербург
2025

ВВЕДЕНИЕ

Архитектура ЭВМ представляет собой динамично развивающуюся дисциплину, охватывающую как фундаментальные принципы организации вычислительных систем, так и современные инновационные подходы к проектированию аппаратно-программных комплексов. Изучение архитектуры ЭВМ сегодня — это исследование не только классических моделей фон Неймана и базовых логических элементов, но и анализ сложных многоядерных, параллельных, энергоэффективных и специализированных систем, определяющих технологический прогресс XXI века.

Современная архитектура ЭВМ сохраняет свою связь с историческими основами — двоичной системой счисления, иерархией памяти, принципами программного управления — но при этом радикально трансформируется под влиянием новых вызовов: необходимости обработки невообразимо большого количество данных в реальном времени; требований к энергоэффективности и мобильности; развития искусственного интеллекта и квантовых вычислений; потребностей в аппаратной безопасности и надёжности.

В данном обзоре рассматривается эволюция архитектурных подходов — от механических вычислительных устройств до современных гетерогенных систем, облачных платформ и нейроморфных процессоров. Особое внимание уделяется преемственности принципов и их адаптации к новым технологическим реалиям, что позволяет понять не только как устроены современные вычислительные системы, но и текущее положение дел в мире тотальной цифровизации и интеллектуализации вычислений.

В конце 40-х и начале 50-х годов прошлого века во многих научных центрах велись разработки электронных вычислительных машин. Можно отметить первую ламповую ЭВМ ENIAC, построенную в 1945 году сотрудниками Пенсильванского университета США Дж. Эккертом (Eckert John Presper, Jr.) и Дж. Моучли (John Mauchly), МЭСМ (Малую Электронную Счетную Машину), созданную в 1951 году в Киеве под руководством академика С.А. Лебедева и другие проекты.

Большой вклад в дело развития вычислительной техники внёс талантливый математик и физик венгерского происхождения Джон фон Нейман(н) (John von Neumann). Во время мировой войны он участвовал в знаменитом Манхэттенском проекте по созданию атомной бомбы и хорошо понимал важность машин для проведения больших объёмов математических вычислений. По мнению многих современников, фон Нейман был в числе самых выдающихся учёных прошлого века. Приехав в Америку, в 1933 году он начал работать в Пристонском Институте Перспективных Исследований США. Достаточно упомянуть, что там в это время работали А.Эйнштейн и К.Гёдель. Ещё в 1945 году, ознакомившись с ЭВМ ENIAC, фон Нейман принял участие в проектировании нового компьютера с названием EDVAC (Electronic Discrete Variable Automatic Computer). Вскоре в соавторстве с Артуром Берксом (Arthur Burks) и Германом Голдстейном (Herman Goldstine) он описал в техническом докладе эту ЭВМ, обладающую рядом новых особенностей. Со временем стало ясно, что эти особенности желательно включать в архитектуру всех разрабатываемых в то время компьютеров. А вскоре пришло и понимание того, что эти новые свойства вычислительных машин, по сути, описывают архитектуру некоторого абстрактного универсального вычислителя, который сейчас принято называть **машиной фон Неймана**.

В некотором смысле машина фон Неймана подобна абстрактным структурам данных. У таких структур данных, например, двоичных деревьев, для их использования необходимо предварительно выполнить конкретную реализацию: произвести отображение на структуры данных в некотором языке

программирования, а также реализовать соответствующие операции над этими данными.

Основополагающие свойства архитектуры машины фон Неймана сформулированы в виде **принципов фон Неймана**, которые лежат в основе большинства современных компьютеров, однако существует и другой подход: **гарвардская архитектура**, в которой память для команд и память для данных физически разделены и имеют независимые шины. Это позволяет одновременно считывать команду и обращаться к данным, повышая скорость выполнения. Данный подход широко используется в микроконтроллерах и сигнальных процессорах (DSP), а также в виде модифицированной гарвардской архитектуры внутри современных CPU (кэш-память команд и данных), однако этот подход не стал массовым из-за сложности с динамическим распределением ресурсов, сложности проектирования, затруднений с реализацией самомодифицирующего кода и др., поэтому рассмотрим архитектуру фон Неймана.

Её ключевые принципы:

Принцип однородности памяти.

Команды и данные хранятся в одной и той же памяти и внешне в памяти неразличимы. Распознать их можно только по способу использования: одно и то же значение в ячейке памяти может использоваться и как данные, и как команда, и как адрес в зависимости лишь от способа обращения к нему.

Это позволяет производить над командами те же операции, что и над числами, и открывает ряд возможностей, например, циклически изменяя адресную часть команды, можно обеспечить обращение к последовательным элементам массива данных.

Принцип программного управления.

Все вычисления, предусмотренные алгоритмом решения задачи, представлены в виде программы, состоящей из последовательности управляющих слов — команд (инструкций). Каждая команда предписывает

некоторую операцию из фиксированного набора операций, реализуемых вычислительной машиной.

Команды программы хранятся в последовательных ячейках памяти и выполняются в естественной последовательности — в порядке их положения в программе. При необходимости, с помощью специальных команд, эта последовательность может быть изменена. Решение об изменении порядка выполнения команд программы принимается либо на основании анализа результатов предшествующих вычислений, либо безусловно.

Принцип адресуемости памяти.

Структурно основная память состоит из пронумерованных ячеек, причём процессору в произвольный момент доступна любая ячейка.

Двоичные коды команд и данных разделяются на единицы информации, называемые словами, и хранятся в ячейках памяти, а для доступа к ним используются номера соответствующих ячеек — адреса.

Отсюда следует возможность давать имена областям памяти, так чтобы к хранящимся в них значениям можно было впоследствии обращаться или менять их в процессе выполнения программы с использованием присвоенных имён.

Принцип последовательного выполнения.

Компьютер выполняет команды последовательно — от первой к последней, согласно порядку следования в программе. Этот порядок может быть изменён с помощью специальных команд в зависимости от результатов вычислений. К сожалению, архитектура фон Неймана не поддерживает эффективное выполнение нескольких инструкций одновременно, что ограничивает производительность в многозадачных средах.

Основными **компонентами** архитектуры фон Неймана являются:

Оперативное записывающее устройство (ОЗУ).

Память машины фон Неймана — это **линейная** (упорядоченная) и **однородная** последовательность некоторых элементов, называемых ячейками (memory cells). В любую ячейку памяти другие устройства машины могут

записывать и считывать информацию, причем время чтения из любой ячейки **одинаково** для всех ячеек памяти. Время записи в любую ячейку тоже одинаково (это и есть принцип однородности памяти). Время чтения из ячейки памяти, однако, может не совпадать со временем записи в неё. Такая память в современных компьютерах называется памятью с произвольным доступом **RAM** (Random Access Memory).

Современные ЭВМ, однако, могут иметь участки памяти разных видов. Например, некоторые области памяти поддерживают только чтение (по-английски эта память называется **ROM** (Read Only Memory), данные в такую память записываются один раз при изготовлении этой памяти, они сохраняются и при отключении электрического напряжения. Другие виды памяти могут допускать запись, но за большее время, чем в остальную память (это так называемая полупостоянная память, на которой обычно реализованы схемы для хранения служебных программ, например, BIOS), флэш-память и др.

Ячейки памяти в машине фон Неймана нумеруются от нуля до некоторого положительного числа N , это и означает, что память линейная, т.е. упорядоченная, за каждой ячейкой (кроме последней) есть следующая, и перед каждой ячейкой (кроме первой) есть предыдущая. Число N в "настоящих" ЭВМ часто равно степени двойки, минус единица, т.е.

$$N = 2^q - 1$$

Адресом ячейки называется её номер. Каждая ячейка состоит из более мелких частей, именуемых разрядами, они тоже нумеруются от нуля и до определенного числа K , причём обычно нумерация идёт справа-налево. Количество разрядов в ячейке обозначает разрядность памяти. Каждый разряд может хранить одну цифру в некоторой системе счисления.

В большинстве ЭВМ используется **двоичная система**, т.к. это более выгодно с инженерной точки зрения — двоичная система хорошо описывает архитектуру узлов ЭВМ с помощью **логических** (булевских) выражений.

Содержимое ячейки называется **машинным словом**. С точки зрения архитектуры машинное слово — минимальный объем данных, которым могут обмениваться между собой различные узлы машины (кроме передачи сигналов).

Ячейки памяти располагаются не только в основной памяти, но и в других записывающих устройствах, они называются **регистрами** — они имеют гораздо меньший объем по сравнению с ОЗУ, но отличаются очень быстрой скоростью из-за реализации на основе триггеров.

Типовые характеристики памяти современных ЭВМ:

- Объем памяти — от нескольких до десятков миллиардов ячеек (обычно восьмиразрядных);
- Скорость работы — **время доступа** (минимальная задержка на чтение слова из памяти в некоторый регистр) и **время цикла** (минимальная задержка на чтение из той же самой ячейки памяти).

Устройство управления (УУ).

Как ясно из названия, устройство управления (УУ) **управляет** всеми остальными устройствами ЭВМ. Оно осуществляет это путем послыки управляющих сигналов, подчиняясь которым остальные устройства производят определенные действия, предписанные этими сигналами. Это устройство реализует принцип программного управления.

В современных ЭВМ количество команд варьируется от 50-100 команд. Например, в Intel 8086 использовалось 98 команд.

На данный момент существует несколько **архитектур инструкций** в зависимости от количества команд: **CISC** (Complex Instruction Set Computer), **RISC** (Reduced Instruction Set Computer), **MISC** (Minimal Instruction Set Computer), **VLIW** (Very Long Instruction Word).

Наиболее распространённым является CISC, который применяется в архитектуре x86.

Арифметико-логическое устройство.

Арифметико-логическое устройство может **выполнять** следующие команды:

- Чтение содержимого ячейки памяти (помещение копии машинного слова из памяти в регистр);
- Записать копию содержимого одного из своих регистров в ячейку памяти;
- Выполнять математические и логические операции над данными в своих регистрах.

В современных процессорах используется не одно, а **несколько АЛУ** (часто 4 или более), которые работают параллельно, а сами АЛУ могут быть **целочисленные** — для выполнения базовых операций, и **адресные** — для быстрого вычисления.

Кроме того, современные процессоры обладают высокой степенью **конвейеризации** за счет нескольких АЛУ — каждая команда разбивается на несколько мелких этапов, и каждый АЛУ выполняет свою часть работы.

Современные процессоры характеризуются следующими параметрами:

- Тактовая частота — частота синхронизации вентилей (логических элементов, процессора. Чем она выше — тем выше производительность процесса (тем больше инструкций он может выполнять за одинаковый промежуток времени), однако с повышением частоты повышается тепловыделение и энергопотребление процессора. Современные ядро процессора имеет частоту от 2 до 5 ГГц (Гига Гц — 10^9 операций в секунду);
- Количество ядер/потоков — количество АЛУ, которые могут выполнять логические и арифметические функции. Рост производительности не растет в геометрической прогрессии с ростом количества процессоров — некоторые задачи могут плохо распараллеливаться и некоторые приложения могут быть плохо оптимизированы под работу с несколькими потоками). Типовое количество ядер/потоков: 2/2 до 8/16, в профессиональных задачах: 12/24 — 16/32;

- Размер кэш-памяти — по скорости эта память находится между регистрами и ОЗУ, но физически ограничена по объему из-за ограниченности места на кристалле процессора.

Устройства ввода и вывода

Устройства ввода и вывода — периферийные (внешние) устройства, которые предназначены для ввода и вывода информации в компьютере. Большинство устройств относятся к одной из этих категорий, хотя некоторые могут выполнять и то, и другое.

Устройства ввода — это оборудование, с помощью которого пользователи взаимодействуют с компьютерами и другими электронными устройствами, передавая данные и команды. Некоторые виды устройств ввода:

- Клавиатура — для ввода текстовой и числовой информации;
- Сканер — для ввода информации с бумажных носителей;
- Микрофон — для ввода звуковой информации;
- Компьютерная мышь — для управления курсором и выполнения команд с помощью кликов;
- Сенсорные экраны — дают пользователям возможность управлять устройствами с помощью прикосновений;
- Графические планшеты и стилусы — используются для создания цифровых рисунков и иллюстраций;
- Сканеры — переводят физические изображения и документы в цифровую форму;
- Контроллеры и игровые устройства — устройства для взаимодействия с играми и симуляциями (геймпады, джойстики).

Устройства вывода — это компоненты компьютерной системы, которые предоставляют пользователю информацию, обработанную процессором. Эта информация может быть представлена в различных формах: текстом, изображениями, звуками и др. Некоторые виды устройств вывода:

- Мониторы (дисплеи) — показывают текст, изображения и видео на экране компьютера;
- Принтеры — используются для печати документов, фотографий и другой информации на бумаге;
- Колонки, наушники — позволяют пользователю слушать музыку, звуки;
- Проектор — позволяет выводить изображение или видео на большой экран или стену, часто используется для презентаций, просмотра фильмов;
- Электронные доски — используются в учебных заведениях и на деловых встречах для интерактивной презентации материалов.

Устройства ввода и вывода:

- Сенсорный экран — позволяет выводить на экран графическую информацию и использовать его чувствительную к касаниям поверхность для ввода информации;
- Гарнитура — сочетает наушники (выход) и микрофон (вход), что удобно для видеозвонков, онлайн-игр;
- USB-накопитель — может как считывать (выводить), так и записывать (вводить) данные.

ЗАКЛЮЧЕНИЕ

Рассмотрение исторического пути и фундаментальных принципов архитектуры электронно-вычислительных машин позволяет сделать ряд важных выводов.

Во-первых, предложенная Джоном фон Нейманом в середине 1940-х годов архитектурная концепция, несмотря на свою простоту и возраст, оказалась исключительно жизнеспособной и плодотворной. Её ключевые принципы — однородность памяти, программное управление, адресуемость и последовательное выполнение команд — легли в основу подавляющего

большинства вычислительных систем второй половины XX и начала XXI века. Даже в условиях современных вызовов эти принципы не отброшены, а эволюционно развиваются и адаптируются, о чём свидетельствует широкое использование модифицированной гарвардской архитектуры на уровне кэш-памяти процессоров.

Во-вторых, анализ основных компонентов архитектуры фон Неймана — оперативного запоминающего устройства, устройства управления, арифметико-логического устройства и устройств ввода-вывода — демонстрирует их радикальную внутреннюю трансформацию при сохранении функционального назначения. Современные процессоры представляют собой сложнейшие комплексы из множества параллельных и специализированных АЛУ, глубоко конвейеризированных и управляемых сложными алгоритмами. Память организована в виде многоуровневой иерархии, а устройство управления поддерживает десятки и сотни команд в рамках различных наборов инструкций (CISC, RISC). Это наглядный пример того, как фундаментальная абстракция реализуется через стремительно усложняющуюся физическую и микроархитектурную основу.

В-третьих, эволюция вычислительной техники привела к осознанию и активному преодолению изначальных ограничений последовательной модели. Возникновение и доминирование многоядерных и многопоточных процессоров, появление векторных расширений команд (SIMD) и гетерогенных вычислений — это прямой ответ на «узкие места» классической архитектуры, в частности, на её неспособность к эффективному параллельному выполнению множества инструкций. Современная вычислительная система — это симбиоз проверенных временем принципов фон Неймана и передовых технологий параллелизма, направленных на удовлетворение требований производительности, энергоэффективности и специализации.

Таким образом, архитектура ЭВМ, берущая своё начало от теоретических работ фон Неймана и первых практических реализаций, таких

как ENIAC и МЭСМ, продолжает оставаться динамично развивающейся областью. Её история — это история блестящего компромисса между элегантной абстракцией и инженерными реалиями, между универсальностью и специализацией, между последовательностью и параллелизмом. Понимание этих основ и их современной интерпретации является ключом к грамотному использованию и проектированию вычислительных систем будущего.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Архитектура ЭВМ и вычислительные системы : учебник для студ. учреждений сред. проф. образования / А. В. Сенкевич. — М. : Издательский центр «Академия», 2014. — 240 с.
2. Arch2030: A Vision of Computer Architecture Research over the Next 15 Years / Luis Ceze, Mark D. Hill, Thomas F. Wenisch — : «CCC», 2016 —12 с.
3. Архитектура ЭВМ и язык ассемблера: учебник для студентов / Падарян Вартан Андроникович — ВМК МГУ, — 291 с.