



Архитектура фон Неймана

Архитектура **фон Неймана** — также известная как **модель фон Неймана** или **архитектура Принстона** — это компьютерная архитектура, основанная на *Первом черновике отчета по EDVAC* ^[1], написанном Джоном фон Нейманом в 1945 году, описывающем проекты, обсуждавшиеся с Джоном Мочли и Дж. Преспером Экертом в Школе электротехники Мура Пенсильванского университета. В документе описывается архитектура проекта электронного цифрового компьютера, состоящего из «органов», которые, как позже поняли, имели следующие компоненты:

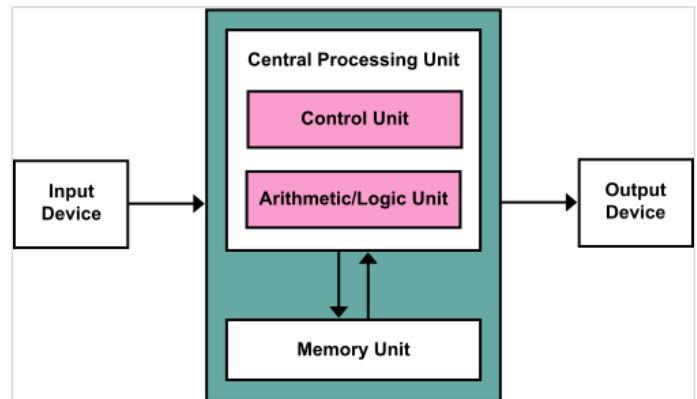


Схема архитектуры фон Неймана

- Процессорное устройство с арифметико-логическим устройством и регистрами процессора
- Блок управления, включающий регистр команд и счетчик программ.
- Память, в которой хранятся данные и инструкции
- Внешнее запоминающее устройство
- Механизмы ввода и вывода ^{[1][2]}

Приписывание изобретения архитектуры фон Нейману является спорным, не в последнюю очередь потому, что Эккерт и Мокли проделали большую часть необходимой проектной работы и утверждают, что у них возникла идея хранить программы задолго до обсуждения этих идей с фон Нейманом и Германом Голдстейном. ^[3]

Термин «архитектура фон Неймана» развился для обозначения любого компьютера с хранимой программой, в котором выборка инструкций и операция с данными не могут происходить одновременно (поскольку они используют общую шину). Это называется узким местом фон Неймана, которое часто ограничивает производительность соответствующей системы. ^[4]

Архитектура фон Неймана проще, чем архитектура Гарварда (которая имеет один выделенный набор шин адреса и данных для чтения и записи в память, а также другой набор шин адреса и данных для выборки инструкций).

Компьютер с хранимой программой использует тот же базовый механизм для кодирования как программных инструкций, так и данных, в отличие от конструкций, которые используют такой механизм, как дискретная коммутационная панель или фиксированная схема управления для реализации инструкций. Компьютеры с хранимой программой были шагом вперед по сравнению с вручную перенастраиваемыми или фиксированными по функциям компьютерами 1940-х годов, такими как Colossus и ENIAC. Они программировались путем установки переключателей и вставки соединительных кабелей для маршрутизации данных и сигналов управления между различными функциональными блоками.

Подавляющее большинство современных компьютеров используют один и тот же аппаратный механизм для кодирования и хранения как данных, так и программных инструкций, но имеют кэши между ЦП и памятью, а для кэшей, расположенных ближе всего к ЦП, имеют отдельные кэши для инструкций и данных, так что большинство выборок инструкций и данных используют отдельные шины (архитектура с разделенным кэшем).

История

Самые ранние вычислительные машины имели фиксированные программы. Некоторые очень простые компьютеры все еще используют эту конструкцию, либо для простоты, либо в целях обучения. Например, настольный калькулятор (в принципе) является компьютером с фиксированной программой. Он может выполнять базовые математические операции , но не может запускать текстовый процессор или игры. Изменение программы машины с фиксированной программой требует перемонтажа, реструктуризации или перепроектирования машины. Самые ранние компьютеры были не столько «запрограммированы», сколько «разработаны» для определенной задачи. «Перепрограммирование» — когда это вообще было возможно — было трудоемким процессом, который начинался с блок-схем и бумажных заметок, за которыми следовали подробные инженерные проекты, а затем часто трудоемкий процесс физического перемонтажа и перестройки машины. Настройка и отладка программы на ENIAC могла занять три недели . ^[5]

С появлением компьютера с хранимой программой ситуация изменилась. Компьютер с хранимой программой по своей конструкции включает набор инструкций и может хранить в памяти набор инструкций (программу), которые детализируют вычисления .

Хранимая программа также допускает самомодифицирующийся код . Одной из ранних мотиваций для такой возможности была потребность в программе для увеличения или иного изменения адресной части инструкций, что операторы должны были делать вручную в ранних разработках. Это стало менее важным, когда индексные регистры и косвенная адресация стали обычными функциями архитектуры машины. Другим применением было встраивание часто используемых данных в поток инструкций с использованием немедленной адресации .

Когда фон Нейман описывал автоматические вычислительные системы, используя терминологию, отличную от той, которая обычно описывается с помощью модели. В первом *черновике отчета по EDVAC* ^[1] архитектура состояла из «высокоскоростной памяти М, центрального арифметического устройства СА, внешнего носителя записи R, входного органа I, выходного органа О и центрального управления СС» ^[6]

Возможности

В больших масштабах способность рассматривать инструкции как данные — это то, что делает возможными ассемблеры , компиляторы , компоновщики , загрузчики и другие автоматизированные инструменты программирования. Это делает возможными «программы, которые пишут программы». ^[7] Это привело к расцвету сложной самохостинговой вычислительной экосистемы вокруг машин с архитектурой фон Неймана.

Некоторые языки высокого уровня используют архитектуру фон Неймана, предоставляя абстрактный, независимый от машины способ манипулирования исполняемым кодом во время выполнения (например, LISP) или используя информацию во время выполнения для настройки оперативной

компиляции (например, языки, размещенные на виртуальной машине Java , или языки, встроенные в веб-браузеры).

В меньшем масштабе некоторые повторяющиеся операции, такие как BITBLT или пиксельные и вершинные шейдеры, могут быть ускорены на процессорах общего назначения с помощью методов компиляции JIT. Это одно из применений самомодифицирующегося кода, которое осталось популярным.

Разработка концепции хранимой программы

Математик Алан Тьюринг , которого натолкнули на проблему математической логики лекции Макса Ньюмена в Кембриджском университете , написал в 1936 году статью под названием « *О вычислимых числах с приложением к проблеме Entscheidungsproblem* » , которая была опубликована в *Трудах Лондонского математического общества* .^[8] В ней он описал гипотетическую машину, которую он назвал *универсальной вычислительной машиной* , теперь известную как « Универсальная машина Тьюринга ». Гипотетическая машина имела бесконечный магазин (память в современной терминологии), который содержал как инструкции, так и данные. Джон фон Нейман познакомился с Тьюрингом, когда он был приглашенным профессором в Кембридже в 1935 году, а также во время года обучения Тьюринга в докторантуре Института перспективных исследований в Принстоне, штат Нью-Джерси, в 1936–1937 годах. Неясно, знал ли он в то время о статье Тьюринга 1936 года.

В 1936 году Конрад Цузе в двух патентных заявках также предположил, что машинные инструкции могут храниться в том же хранилище, что и данные.^[9]

Независимо друг от друга, Дж. Преспер Эккерт и Джон Мокли , которые разрабатывали ENIAC в Школе электротехники Мура Пенсильванского университета , написали о концепции хранимой программы в декабре 1943 года.^{[10] [11]} Планируя новую машину, EDVAC , Эккерт в январе 1944 года писал, что они будут хранить данные и программы в новом адресуемом запоминающем устройстве, памяти с линией задержки на ртутном металле . Это был первый случай, когда было предложено создание практической машины с хранимой программой. В то время он и Мокли не знали о работе Тьюринга.

Фон Нейман был вовлечен в Манхэттенский проект в Лос-Аламосской национальной лаборатории . Он требовал огромного количества вычислений, и поэтому летом 1944 года привлек его к проекту ENIAC. Там он присоединился к продолжающимся обсуждениям по проекту этого компьютера с хранимой программой, EDVAC. В составе этой группы он написал описание под названием « *Первый черновик отчета об EDVAC* »^[1] , основанное на работе Эккерта и Мокли. Оно было незаконченным, когда его коллега Герман Голдстейн распространил его, и в нем было указано только имя фон Неймана (к ужасу Эккерта и Мокли).^[12] Доклад прочитали десятки коллег фон Неймана в Америке и Европе, и он повлиял на следующий этап компьютерных разработок.

Джек Коупленд считает, что «исторически некорректно называть электронные цифровые компьютеры с хранимой программой «машинами фон Неймана»» .^[13] Его коллега из Лос-Аламоса Стэн Франкель сказал об отношении фон Неймана к идеям Тьюринга^[14]

Я знаю, что в 1943 или 1944 году или около того фон Нейман был хорошо осведомлен о фундаментальной важности статьи Тьюринга 1936 года... Фон Нейман познакомил меня с этой статьей, и по его настоянию я внимательно ее изучил. Многие люди провозгласили фон Неймана «отцом компьютера» (в современном смысле этого слова), но я уверен, что он сам

никогда бы не совершил такой ошибки. Его, возможно, можно было бы назвать повивальной бабкой, но он твердо подчеркивал мне и другим, я уверен, что фундаментальная концепция обязана Тьюрингу — в той мере, в какой ее не предвосхитил Бэббидж... И Тьюринг, и фон Нейман, конечно, также внесли существенный вклад в « приведение к практике » этих концепций, но я бы не считал их сопоставимыми по важности с введением и объяснением концепции компьютера, способного хранить в своей памяти программу своих действий и изменять эту программу в ходе этих действий.

В то время, когда был распространен отчет «Первый черновик», Тьюринг работал над отчетом под названием «*Предлагаемый электронный калькулятор*» . В нем подробно описывалась его идея машины, которую он назвал *Автоматической вычислительной машиной (ACE)* . ^[15] Он представил ее исполнительному комитету Британской национальной физической лаборатории 19 февраля 1946 года. Хотя Тьюринг знал по своему военному опыту в Блетчли-парке, что то, что он предложил, осуществимо, секретность, окружающая «Колосс» , которая впоследствии поддерживалась в течение нескольких десятилетий, помешала ему сказать об этом. Были созданы различные успешные реализации проекта ACE.

В статьях фон Неймана и Тьюринга описывались компьютеры с хранимой программой, но более ранняя статья фон Неймана получила большее распространение, а описанная в ней архитектура компьютера стала известна как «архитектура фон Неймана». В публикации 1953 года « *Быстрее мысли: симпозиум по цифровым вычислительным машинам*» (под редакцией Б. В. Боудена) раздел в главе « *Компьютеры в Америке*» звучит следующим образом: ^[16]

Машина Института перспективных исследований, Принстон

В 1945 году профессор Дж. фон Нейман, работавший тогда в Школе инженерии Мура в Филадельфии, где был построен ENIAC, от имени группы своих коллег опубликовал отчет о логическом проектировании цифровых компьютеров. Отчет содержал подробное предложение по проектированию машины, которая с тех пор стала известна как EDVAC (электронный дискретно-переменный автоматический компьютер). Эта машина была только недавно завершена в Америке, но отчет фон Неймана вдохновил на создание EDSAC (электронный запаздывающий автоматический калькулятор) в Кембридже (см. стр. 130).

В 1947 году Беркс, Голдстайн и фон Нейман опубликовали еще один отчет, в котором описывалась конструкция другого типа машины (на этот раз параллельной машины), которая была бы чрезвычайно быстрой, способной, возможно, на 20 000 операций в секунду. Они указали, что нерешенной проблемой при создании такой машины была разработка подходящей памяти с мгновенно доступным содержимым. Сначала они предложили использовать специальную вакуумную трубку — называемую « Селектор », — которую изобрели Принстонские лаборатории RCA. Эти трубки были дорогими и сложными в изготовлении, поэтому фон Нейман впоследствии решил построить машину на основе памяти Уильямса . Эта машина, завершенная в июне 1952 года в Принстоне, стала широко известна как «Маньяк». Конструкция этой машины вдохновила по крайней мере на полдюжины машин, которые сейчас строятся в Америке, все они известны как «Джониаки».

В той же книге первые два абзаца главы об ACE звучат следующим образом: ^[17]

Автоматические вычисления в Национальной физической лаборатории

Один из самых современных цифровых компьютеров, воплощающий разработки и усовершенствования в технике автоматических электронных вычислений, был недавно продемонстрирован в Национальной физической лаборатории в Теддингтоне, где он был спроектирован и построен небольшой группой математиков и инженеров-исследователей электроники из штата Лаборатории, которым помогали несколько инженеров-производственников из English Electric Company, Limited. Оборудование, установленное на сегодняшний день в Лаборатории, является лишь пилотной моделью гораздо более крупной установки, которая будет известна как Автоматическая вычислительная машина, но хотя она сравнительно мала по размеру и содержит всего около 800 термоионных вентилях, как можно судить по таблицам XII, XIII и XIV, это чрезвычайно быстрая и универсальная вычислительная машина.

Основные понятия и абстрактные принципы вычислений с помощью машины были сформулированы доктором А. М. Тьюрингом, членом Королевского общества, в докладе¹, прочитанном перед Лондонским математическим обществом в 1936 году, но работа над такими машинами в Британии была задержана войной. Однако в 1945 году исследование проблем было проведено в Национальной физической лаборатории г-ном Дж. Р. Уомерсли, тогдашним суперинтендантом математического отдела лаборатории. К нему присоединились доктор Тьюринг и небольшой штат специалистов, и к 1947 году предварительное планирование было достаточно продвинуто, чтобы оправдать создание уже упомянутой специальной группы. В апреле 1948 года последняя стала Электронным отделом лаборатории под руководством г-на Ф. М. Колбука.

Ранние компьютеры с архитектурой фон Неймана

В *первом черновике* описывалась конструкция, которая использовалась многими университетами и корпорациями для создания своих компьютеров.^[18] Среди этих различных компьютеров только ILLIAC и ORDVAC имели совместимые наборы инструкций.

- ARC2 (Биркбек, Лондонский университет) официально вступил в строй 12 мая 1948 года.^[19]
- Manchester Baby (Университет Виктории в Манчестере , Англия) осуществил первый успешный запуск сохраненной программы 21 июня 1948 года.
- EDSAC (Кембриджский университет , Англия) был первым практическим электронным компьютером с хранимой программой (май 1949 г.)
- Manchester Mark 1 (Манчестерский университет , Англия), разработанный на основе Baby (июнь 1949 г.)
- CSIRAC (Совет по научным и промышленным исследованиям) Австралия (ноябрь 1949 г.)
- МЭСМ в Киевском электротехническом институте в Киеве , Украинская ССР (ноябрь 1950 г.)
- EDVAC (Лаборатория баллистических исследований , вычислительная лаборатория на Абердинском испытательном полигоне , 1951 г.)
- Машина IAS в Институте перспективных исследований (1951)
- ORDVAC (Университет Иллинойса) на испытательном полигоне Абердин , Мэриленд (завершен в ноябре 1951 г.)^[20]

- MANIAC I в Лос-Аламосской научной лаборатории (март 1952 г.)
- ILLIAC в Университете Иллинойса (сентябрь 1952 г.)
- БЭСМ-1 в Москве (1952)
- AVIDAC в Аргоннской национальной лаборатории (1953)
- ORACLE в Окриджской национальной лаборатории (июнь 1953 г.)
- BESK в Стокгольме (1953)
- ДЖОННИАК в корпорации RAND (январь 1954 г.)
- DASK в Дании (1955)
- WEIZAC в Институте науки Вейцмана в Реховоте , Израиль (1955)
- ПЕРМ в Мюнхене (1956)
- SILLIAC в Сиднее (1956)

Ранние компьютеры с хранимой программой

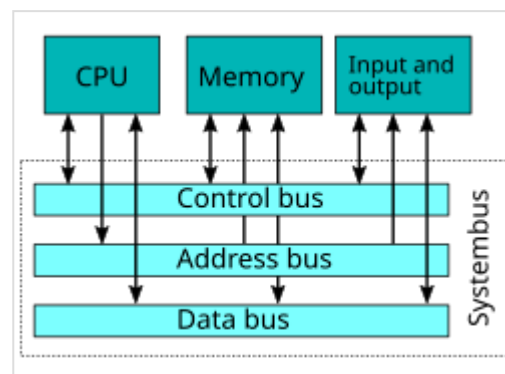
Информацию о датах в следующей хронологии трудно расположить в правильном порядке. Некоторые даты относятся к первому запуску тестовой программы, некоторые даты относятся к первой демонстрации или завершению работы компьютера, а некоторые даты относятся к первой поставке или установке.

- IBM SSEC обладал способностью обрабатывать инструкции как данные и был публично продемонстрирован 27 января 1948 года. Эта способность была заявлена в патенте США . ^{[21] [22]} Однако он был частично электромеханическим , а не полностью электронным. На практике инструкции считывались с бумажной ленты из-за ее ограниченной памяти. ^[23]
- ARC2 , разработанный Эндрю Бутом и Кэтлин Бут в Биркбеке, Лондонском университете, официально вошел в эксплуатацию 12 мая 1948 года. ^[19] Он был оснащен первым вращающимся барабанным устройством хранения данных . ^{[24] [25]}
- Manchester Baby был первым полностью электронным компьютером, который запускал сохраненную программу. Он запускал программу факторизации в течение 52 минут 21 июня 1948 года, после запуска простой программы деления и программы, показывающей, что два числа являются взаимно простыми .
- ENIAC был модифицирован для работы в качестве примитивного компьютера с хранимой программой, доступной только для чтения (с использованием таблиц функций для ПЗУ программ) , и был продемонстрирован в таком качестве 16 сентября 1948 года, выполняя программу Адель Голдстейн для фон Неймана.
- BINAC провел несколько испытательных программ в феврале, марте и апреле 1949 года, однако завершил их только в сентябре 1949 года .
- Manchester Mark 1 был разработан на основе проекта Baby. Промежуточная версия Mark 1 была доступна для запуска программ в апреле 1949 года, но была завершена только в октябре 1949 года.
- Первая программа EDSAC была реализована 6 мая 1949 года .
- EDVAC был доставлен в августе 1949 года, но у него возникли проблемы , из-за которых его не вводили в регулярную эксплуатацию до 1951 года.
- Первая программа CSIR Mk I была запущена в ноябре 1949 года .
- SEAC был продемонстрирован в апреле 1950 года .
- Pilot ACE провел свою первую программу 10 мая 1950 года и был продемонстрирован в декабре 1950 года.
- SWAC был завершён в июле 1950 года .
- Строительство «Вихря» было завершено в декабре 1950 года, а в апреле 1951 года он был введен в эксплуатацию.

- Первый ERA Atlas (позднее коммерческий ERA 1101/UNIVAC 1101) был установлен в декабре 1950 года.

Эволюция

В течение десятилетий 1960-х и 1970-х годов компьютеры в целом стали и меньше, и быстрее, что привело к эволюции их архитектуры. Например, отображаемый в память ввод/вывод позволяет обрабатывать устройства ввода и вывода так же, как память. [26] Единая системная шина может использоваться для обеспечения модульной системы с более низкой стоимостью . Иногда это называют «упорядочиванием» архитектуры. [27] В последующие десятилетия простые микроконтроллеры иногда исключали функции модели для снижения стоимости и размера. Более крупные компьютеры добавляли функции для более высокой производительности.



Эволюция архитектуры единой системной шины

Ограничения конструкции

узкое место фон Неймана

Использование одной и той же шины для извлечения инструкций и данных приводит к *узкому месту фон Неймана* , ограниченной пропускной способности (скорости передачи данных) между центральный процессором (ЦП) и памятью по сравнению с объемом памяти. Поскольку одна шина может получить доступ только к одному из двух классов памяти одновременно, пропускная способность ниже скорости, с которой может работать ЦП. Это серьезно ограничивает эффективную скорость обработки, когда ЦП требуется выполнять минимальную обработку больших объемов данных. ЦП постоянно вынужден ждать, пока необходимые данные будут перемещены в память или из нее. Поскольку скорость ЦП и объем памяти увеличились намного быстрее, чем пропускная способность между ними, узкое место стало большей проблемой, проблемой, серьезность которой возрастает с каждым новым поколением ЦП.

Узкое место фон Неймана было описано Джоном Бэкусом в его лекции на вручении премии Тьюринга АСМ в 1977 году . По словам Бэкуса:

Конечно, должен быть менее примитивный способ внесения больших изменений в хранилище, чем проталкивание огромного количества слов вперед и назад через узкое место фон Неймана. Эта трубка не только является буквальным узким местом для трафика данных проблемы, но, что более важно, это интеллектуальное узкое место, которое удерживает нас привязанными к мышлению пословно, вместо того чтобы побуждать нас думать в терминах более крупных концептуальных единиц текущей задачи. Таким образом, программирование в основном планирует и детализирует огромный трафик слов через узкое место фон Неймана, и большая часть этого трафика касается не самих значимых данных, а того, где их найти. [28] [29]

Смягчение последствий

Существует несколько известных методов смягчения узкого места производительности фон Неймана. Например, все нижеперечисленное может улучшить производительность:

- Предоставление кэша между ЦП и основной памятью .
- Предоставление отдельных кэшей или отдельных путей доступа для данных и инструкций (так называемая модифицированная гарвардская архитектура).
- Использование алгоритмов и логики прогнозирования ветвлений .
- Предоставление ограниченного стека ЦП или другой встроенной оперативной памяти для сокращения доступа к памяти.
- Реализация иерархии ЦП и памяти в виде системы на кристалле , обеспечивающая большую локальность ссылок и, таким образом, уменьшающая задержку и увеличивающая пропускную способность между регистрами процессора и основной памятью.

Проблему также можно обойти, используя параллельные вычисления , например, архитектуру с неравномерным доступом к памяти (NUMA) — этот подход обычно используется суперкомпьютерами . Менее ясно, сильно ли изменилось *интеллектуальное узкое место* , которое *критиковал Бэкус, с 1977 года*. Предложенное Бэкусом решение не оказало большого влияния. Современное функциональное программирование и объектно-ориентированное программирование гораздо меньше ориентированы на «передачу огромного количества слов туда и обратно» , чем более ранние языки, такие как FORTRAN , но внутренне это все еще то, на что компьютеры тратят большую часть своего времени, даже высокопараллельные суперкомпьютеры.

Самоизменяющийся код

Помимо узкого места фон Неймана, изменения программ могут быть весьма вредными, как случайно, так и намеренно. В некоторых простых конструкциях компьютеров с хранимой программой неисправная программа может повредить себя, другие программы или операционную систему, что может привести к сбою компьютера. Однако эта проблема применима и к обычным программам, в которых отсутствует проверка границ . Защита памяти и различные элементы управления доступом обычно защищают как от случайных, так и от вредоносных изменений программ.

Смотрите также

- Наглядное пособие по вычислениям CARDboard
- Узкое место в межсоединении
- Маленький компьютер для человека
- Машина с произвольным доступом
- Гарвардская архитектура
- машина Тьюринга

Ссылки

1. фон Нейман, Джон(1945), *Первый черновик отчета по EDVAC* (<https://web.archive.org/web/20130314123032/http://qss.stanford.edu/~godfrey/vonNeumann/vnedvac.pdf>) (PDF), заархивировано из оригинала (<https://sites.google.com/site/michaeldgodfrey/vonneumann/vnedvac.pdf>) 14 марта 2013 г., извлечено 24 августа 2011 г..
2. Ганесан 2009 .

3. Бергин, Томас Дж. (2000), *Пятьдесят лет армейских вычислений: от ENIAC до MSRC, Исследовательская лаборатория армии США* (<https://www.govinfo.gov/app/details/GOVPU B-D105-PURL-LPS58495>) , стр. 34 , получено 5 ноября 2024 г. (<https://www.govinfo.gov/app/details/GOVPU B-D105-PURL-LPS58495>)
4. Маркграф, Джои Д. (2007), *Узкое место фон Неймана* (<https://web.archive.org/web/20131212205159/http://aws.linnbenton.edu/cs271c/markgrj/>) , архивировано из оригинала (<http://aws.linnbenton.edu/cs271c/markgrj/>) 12 декабря 2013 г. (<https://web.archive.org/web/20131212205159/http://aws.linnbenton.edu/cs271c/markgrj/>) (<http://aws.linnbenton.edu/cs271c/markgrj/>).
5. Коупленд 2006 , стр. 104.
6. фон Нейман, Джон (1966). *Теория самовоспроизводящихся автоматов* . Издательство Иллинойского университета. стр. 10. ISBN 978-0252727337.
7. MFTL (*My Favorite Toy Language*) запись Жаргонный файл 4.4.7 (<http://catb.org/~esr/jargon/html/M/MFTL.html>) , получено 11 июля 2008 г..
8. Тьюринг, Алан М. (1936), «О вычислимых числах с приложением к проблеме Entscheidungsproblem», *Труды Лондонского математического общества* , 2, т. 42 (опубликовано в 1937 г.), стр. 230–265 , doi : 10.1112/plms/s2-42.1.230 (<https://doi.org/10.1112%2Fplms%2Fs2-42.1.230>) , S2CID 73712 (<https://api.semanticscholar.org/CorpusID:73712>) (<https://doi.org/10.1112%2Fplms%2Fs2-42.1.230>) (<https://api.semanticscholar.org/CorpusID:73712>)и Тьюринг, Алан М. (1938), «О вычислимых числах с приложением к проблеме Entscheidungsproblem. Исправление», *Труды Лондонского математического общества* , 2, т. 43, № 6 (опубликовано в 1937 г.), стр. 544–546 , doi : 10.1112/plms/s2-43.6.544 (<http://doi.org/10.1112%2Fplms%2Fs2-43.6.544>) (<https://doi.org/10.1112%2Fplms%2Fs2-43.6.544>)
9. Уильямс, ФК; Килберн, Т. (25 сентября 1948 г.), «Электронные цифровые компьютеры», *Nature* , **162** (4117): 487, Bibcode : 1948Natur.162..487W (<https://ui.adsabs.harvard.edu/abs/1948Natur.162..487W>) , doi : 10.1038/162487a0 (<https://doi.org/10.1038%2F162487a0>) , S2CID 4110351 (<https://api.semanticscholar.org/CorpusID:4110351>) (<https://ui.adsabs.harvard.edu/abs/1948Natur.162..487W>) (<https://doi.org/10.1038%2F162487a0>) (<https://api.semanticscholar.org/CorpusID:4110351>).
10. Лукофф, Герман (1979). *Om Dits to Bits: личная история электронного компьютера* . Портленд, Орегон: Robotics Press. ISBN 0-89661-002-0. LCCN 79-90567 (<https://lccn.loc.gov/79-90567>) .
11. В отчете администратора проекта ENIAC Гриста Брейнерда от декабря 1943 года о ходе работ по первому периоду разработки ENIAC неявно предлагалась концепция хранимой программы (одновременно отвергалась ее реализация в ENIAC), в котором говорилось, что «чтобы иметь максимально простой проект и не усложнять ситуацию», ENIAC будет построен без какого-либо «автоматического регулирования».
12. Коупленд 2006 , стр. 113.
13. Коупленд, Джек (2000), *Краткая история вычислений: ENIAC и EDVAC* (http://www.alanturing.net/turing_archive/pages/Reference%20Articles/BriefHistofComp.html#ACE) , получено 27 января 2010 г. (http://www.alanturing.net/turing_archive/pages/Reference%20Articles/BriefHistofComp.html#ACE).
14. Коупленд, Джек (2000), *Краткая история вычислений: ENIAC и EDVAC* (http://www.alanturing.net/turing_archive/pages/Reference%20Articles/BriefHistofComp.html#ACE) , получено 27 января 2010 г. (http://www.alanturing.net/turing_archive/pages/Reference%20Articles/BriefHistofComp.html#ACE)(работа, в которой цитируются Рэнделл, Брайан (1972), Мельцер, Б.; Мичи, Д. (ред.), «Об Алане Тьюринге и происхождении цифровых компьютеров», *Machine Intelligence* , 7 , Эдинбург: Edinburgh University Press : 10, ISBN 0-902383-26-4.
15. Коупленд 2006 , стр. 108–111.
16. Боуден 1953 , стр. 176, 177.
17. Боуден 1953 , стр. 135.
18. "Electronic Computer Project" (<http://www.ias.edu/people/vonneumann/ecp/>) . Institute for Advanced Study . 11 сентября 2009 г. . Получено 26 мая 2011 г. . (<http://www.ias.edu/people/vonneumann/ecp/>)

19. Кэмпбелл-Келли, Мартин (апрель 1982 г.). «Развитие компьютерного программирования в Великобритании (1945–1955 гг.)». *IEEE Annals of the History of Computing*. **4**(2):121–139. doi:10.1109/MAHC.1982.10016 (<https://doi.org/10.1109%2FMAHC.1982.10016>) (<https://api.semanticscholar.org/CorpusID:14861159>). (<https://doi.org/10.1109%2FMAHC.1982.10016>) (<https://api.semanticscholar.org/CorpusID:14861159>)
20. Робертсон, Джеймс Э. (1955), *Illiac Design Techniques*, номер отчета UIUCDCS-R-1955–146, Лаборатория цифровых компьютеров, Иллинойский университет в Урбане-Шампейне.
21. Электронный калькулятор селективной последовательности (веб-сайт USPTO) (<http://patft.uspto.gov/netacgi/nph-Parser?patentnumber=2636672>) .
22. Электронный калькулятор с селективной последовательностью (патенты Google) (<https://patents.google.com/patent/US2636672>) .
23. Грош, Герберт Р. Дж. (1991), *Компьютер: кусочки жизни* (<http://www.columbia.edu/acis/history/computer.html>) , книги третьего тысячелетия, ISBN (<http://www.columbia.edu/acis/history/computer.html>) 0-88733-085-1.
24. Лавингтон, Саймон, ред. (2012). *Алан Тьюринг и его современники: создание первых компьютеров в мире* . Лондон: Британское компьютерное общество . стр. 61. ISBN 978-1906124908.
25. Джонсон, Роджер (апрель 2008 г.). «Школа компьютерных наук и информационных систем: краткая история» (<http://www.dcs.bbk.ac.uk/site/assets/files/1029/50yearsofcomputing.pdf>)(PDF) . Колледж Биркбек . Лондонский университет . Получено 23 июля 2017 г. . (<http://www.dcs.bbk.ac.uk/site/assets/files/1029/50yearsofcomputing.pdf>)
26. Белл, К. Гордон ; Кэди, Р.; Макфарланд, Х.; О'Лафлин, Дж.; Нунан, Р.; Вульф, В. (1970), «Новая архитектура для мини-компьютеров – DEC PDP-11» (<http://research.microsoft.com/en-us/um/people/gbell/CGB%20Files/New%20Architecture%20PDP11%20SJCC%201970%20c.pdf>)(PDF) , Весенняя совместная компьютерная конференция , стр. 657–675 (<http://research.microsoft.com/en-us/um/people/gbell/CGB%20Files/New%20Architecture%20PDP11%20SJCC%201970%20c.pdf>) .
27. Null, Линда; Lobur, Джулия (2010), *Основы организации и архитектуры компьютера* (https://books.google.com/books?id=f83XxoBC_8MC&pg=PA36) (3-е изд.), Jones & Bartlett Learning, стр. 36, 199–203 , ISBN (https://books.google.com/books?id=f83XxoBC_8MC&pg=PA36) 978-1-4496-0006-8.
28. Бэкус, Джон В. (август 1978 г.). «Можно ли освободить программирование от стиля фон Неймана? Функциональный стиль и его алгебра программ» (<https://doi.org/10.1145%2F359576.359579>) . *Сообщения ACM* . **21** (8): 613– 641. doi : 10.1145/359576.359579 (<https://doi.org/10.1145%2F359576.359579>) . S2CID 16367522 (<https://api.semanticscholar.org/CorpusID:16367522>) . (<https://doi.org/10.1145%2F359576.359579>) (<https://doi.org/10.1145%2F359576.359579>) (<https://api.semanticscholar.org/CorpusID:16367522>)
29. Дейкстра, Эдсгер В. "Архив Э. В. Дейкстры: обзор лекции о вручении премии Тьюринга 1977 года" (<http://www.cs.utexas.edu/~EWD/transcriptions/EWD06xx/EWD692.html>). Получено 11 июля 2008 г. (<http://www.cs.utexas.edu/~EWD/transcriptions/EWD06xx/EWD692.html>)

Дальнейшее чтение

- Боуден, Б.В., ред. (1953), *Быстрее, чем мысль: симпозиум по цифровым вычислительным машинам* , Лондон: Sir Isaac Pitman and Sons Ltd.
- Рокас, Рауль; Хашаген, Ульф, ред. (2000), *Первые компьютеры: история и архитектура* , MIT Press , ISBN 0-262-18197-5
- Дэвис, Мартин (2000), *Универсальный компьютер: путь от Лейбница к Тьюрингу* , Нью-Йорк: WW Norton & Company Inc. , ISBN 0-393-04785-7перездано как: Дэвис, Мартин (2001), *Двигатели логики: Математики и происхождение компьютеров* , Нью-Йорк: WW Norton & Company , ISBN 978-0-393-32229-3

- *Можно ли освободить программирование от стиля фон Неймана?*. Бэкус , Джон . Лекция на церемонии вручения премии Тьюринга ACM 1977 года. Communications of the ACM, август 1978 г., том 21, номер 8. Онлайн-формат PDF. (<http://www.stanford.edu/class/cs242/readings/backus.pdf>) Архивировано (<https://web.archive.org/web/20070621162552/http://www.stanford.edu/class/cs242/readings/backus.pdf>) 21 июня 2007 г. на Wayback Machine . Подробности см. на сайте <https://www.cs.tufts.edu/~nr/backus-lecture.html> .
- Белл, К. Гордон; Ньюэлл, Аллен (1971), *Компьютерные структуры: материалы и примеры* , McGraw-Hill Book Company , Нью-Йорк. Массивный (668 страниц)
- Коупленд, Джек (2006), «Колосс и расцвет современного компьютера», в Коупленд, Б. Джек (ред.), *Колосс: Секреты взлома кодов компьютеров Блетчли-Парка* , Оксфорд: Oxford University Press , ISBN 978-0-19-284055-4
- Ганесан, Дипак (2009), *Модель фон Неймана* (<https://web.archive.org/web/20120425083227/http://none.cs.umass.edu/~dganesan/courses/fall09/handouts/Chapter4.pdf>) (PDF) , заархивировано из оригинала (<http://none.cs.umass.edu/~dganesan/courses/fall09/handouts/Chapter4.pdf>) (PDF) 25 апреля 2012 г. , извлечено 22 октября 2011 г.
- Маккартни, Скотт (1999). *ENIAC: Триумфы и трагедии первого в мире компьютера* (<http://archive.org/details/eniac00scot>) . Walker & Co. ISBN 0-8027-1348-3.
- Голдштейн, Герман Х. (1972). *Компьютер от Паскаля до фон Неймана* (<https://archive.org/details/computerfrompasc00herm>) . Princeton University Press . ISBN 0-691-08104-2.
- Шуркин, Джоэл (1984). *Двигатели разума: история компьютера* . Нью-Йорк, Лондон: WW Norton & Company . ISBN 0-393-01804-0.

Внешние ссылки

- Гарвард против фон Неймана (<http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.faqs/ka11516.html>)
 - Инструмент, имитирующий поведение машины фон Неймана. (<https://web.archive.org/web/20080219131555/http://home.gna.org/vov/>)
 - ДЖОННИ: Простой симулятор машины фон Неймана с открытым исходным кодом для образовательных целей (<http://sourceforge.net/projects/johnnysimulator/>)
-

Retrieved from "https://en.wikipedia.org/w/index.php?title=Von_Neumann_architecture&oldid=1291491503"