



Процессоры

Работа ученика 8-В класса

Ястребова Никита

История развития производства процессоров полностью соответствует истории развития технологии производства прочих электронных компонентов и схем.

Первым этапом, затронувшим период с 1940-х по конец 1950-х годов, было создание процессоров с использованием электромеханических реле, ферритовых сердечников (устройств памяти) и вакуумных ламп. Они устанавливались в специальные разъёмы на модулях, собранных в стойки. Большое количество таких стоек, соединённых проводниками, в сумме представляли процессор. Отличительной особенностью была низкая надёжность, низкое быстродействие и большое тепловыделение.

Вторым этапом, с середины 1950-х до середины 1960-х, стало внедрение транзисторов. Транзисторы монтировались уже на близкие к современным по виду платам, устанавливаемым в стойки. Как и ранее, в среднем процессор состоял из нескольких таких стоек. Возросло быстродействие, повысилась надёжность, уменьшилось энергопотребление.

Третьим этапом, наступившим в середине 1960-х годов, стало использование микросхем. Первоначально использовались микросхемы низкой степени интеграции, содержащие простые транзисторные и резисторные сборки, затем, по мере развития технологии, стали использоваться микросхемы, реализующие отдельные элементы цифровой схмотехники (сначала элементарные ключи и логические элементы, затем более сложные элементы — элементарные регистры, счётчики, сумматоры), позднее появились микросхемы, содержащие функциональные блоки процессора — микропрограммное устройство, арифметическо-логическое устройство, регистры, устройства работы с шинами данных и команд.

Четвёртым этапом, в начале 1970-х годов, стало создание, благодаря прорыву в технологии создания БИС и СБИС (больших и сверхбольших интегральных схем, соответственно), микропроцессора — микросхемы, на кристалле которой физически были расположены все основные элементы и блоки процессора. Фирма Intel в 1971 году создала первый в мире 4-разрядный микропроцессор 4004, предназначенный для использования в микрокалькуляторах. Постепенно практически все процессоры стали выпускаться в формате микропроцессоров. Исключением долгое время оставались только малосерийные процессоры, аппаратно оптимизированные для решения специальных задач (например, суперкомпьютеры или процессоры для решения ряда военных задач), либо процессоры, к которым предъявлялись особые требования по надёжности, быстродействию или защите от электромагнитных импульсов и ионизирующей радиации. Постепенно, с удешевлением и распространением современных технологий, эти процессоры также начинают изготавливаться в формате микропроцессора.

История



Большинство современных процессоров для персональных компьютеров в общем основаны на той или иной версии циклического процесса последовательной обработки данных, изобретённого Джоном фон Нейманом.

Дж. фон Нейман придумал схему постройки компьютера в 1946 году.

Отличительной особенностью архитектуры фон Неймана является то, что инструкции и данные хранятся в одной и той же памяти.

В различных архитектурах и для различных команд могут потребоваться дополнительные этапы. Например, для арифметических команд могут потребоваться дополнительные обращения к памяти, во время которых производится считывание операндов и запись результатов.

Этапы цикла выполнения:

- Процессор выставляет число, хранящееся в регистре счётчика команд, на шину адреса и отдаёт памяти команду чтения.
- Если последняя команда не является командой перехода, процессор увеличивает на единицу (в предположении, что длина каждой команды равна единице) число, хранящееся в счётчике команд; в результате там образуется адрес следующей команды.
- Выставленное число является для памяти адресом; память, получив адрес и команду чтения, выставляет содержимое, хранящееся по этому адресу, на шину данных и сообщает о готовности.
- Процессор получает число с шины данных, интерпретирует его как команду (машинную инструкцию) из своей системы команд и исполняет её.

Данный цикл выполняется неизменно, и именно он называется процессом (откуда и произошло название устройства).

Во время процесса процессор считывает последовательность команд, содержащихся в памяти, и исполняет их. Такая последовательность команд называется программой и представляет алгоритм работы процессора. Очередность считывания команд изменяется в случае, если процессор считывает команду перехода, — тогда адрес следующей команды может оказаться другим. Другим примером изменения процесса может служить случай получения команды останова или переключение в режим обработки прерывания.

Команды центрального процессора являются самым нижним уровнем управления компьютером, поэтому выполнение каждой команды неизбежно и безусловно. Не производится никакой проверки на допустимость выполняемых действий, в частности, не проверяется возможная потеря ценных данных. Чтобы компьютер выполнял только допустимые действия, команды должны быть соответствующим образом организованы в виде необходимой программы.

Скорость перехода от одного этапа цикла к другому определяется тактовым генератором. Тактовый генератор вырабатывает импульсы, служащие ритмом для центрального процессора. Частота тактовых импульсов называется тактовой частотой.

Архитектура фон Неймана

Архитектура фон Неймана



Гарвардская архитектура отличается от архитектуры фон Неймана тем, что программный код и данные хранятся в разной памяти. В такой архитектуре невозможны многие методы программирования (например, программа не может во время выполнения менять свой код; невозможно динамически перераспределять память между программным кодом и данными); зато гарвардская архитектура позволяет более эффективно выполнять работу в случае ограниченных ресурсов, поэтому она часто применяется во встраиваемых системах.

Гарвардская архитектура



Первоначально перед разработчиками ставится техническое задание, исходя из которого принимается решение о том, какова будет архитектура будущего процессора, его внутреннее устройство, технология изготовления. Перед различными группами ставится задача разработки соответствующих функциональных блоков процессора, обеспечения их взаимодействия, электромагнитной совместимости. В связи с тем, что процессор фактически является цифровым автоматом, полностью отвечающим принципам булевой алгебры, с помощью специализированного программного обеспечения, работающего на другом компьютере, строится виртуальная модель будущего процессора. На ней проводится тестирование процессора, исполнение элементарных команд, значительных объёмов кода, отрабатывается взаимодействие различных блоков устройства, ведётся оптимизация, ищутся неизбежные при проекте такого уровня ошибки.

После этого из цифровых базовых матричных кристаллов и микросхем, содержащих элементарные функциональные блоки цифровой электроники, строится физическая модель процессора, на которой проверяются электрические и временные характеристики процессора, тестируется архитектура процессора, продолжается исправление найденных ошибок, уточняются вопросы электромагнитной совместимости (например, при практически рядовой тактовой частоте в 1 ГГц отрезки проводника длиной в 7 мм уже работают как излучающие или принимающие антенны).

Затем начинается этап совместной работы инженеров-схемотехников и инженеров-технологов, которые с помощью специализированного программного обеспечения преобразуют электрическую схему, содержащую архитектуру процессора, в топологию кристалла. Современные системы автоматического проектирования позволяют, в общем случае, из электрической схемы напрямую получить пакет трафаретов для создания масок. На этом этапе технологи пытаются реализовать технические решения, заложенные схемотехниками, с учётом имеющейся технологии. Этот этап является одним из самых долгих и сложных в разработке и иногда требует компромиссов со стороны схемотехников по отказу от некоторых архитектурных решений. Следует отметить, что ряд производителей заказных микросхем (foundry) предлагает разработчикам (дизайн-центру или fabless) компромиссное решение, при котором на этапе конструирования процессора используются представленные ими стандартизованные в соответствии с имеющейся технологией библиотеки элементов и блоков (Standard cell). Это вводит ряд ограничений на архитектурные решения, зато этап технологической подгонки фактически сводится к игре в конструктор «Лего». В общем случае, изготовленные по индивидуальным проектам микропроцессоры являются более быстрыми по сравнению с процессорами, созданными на основании имеющихся библиотек.

Процесс изготовления

Следующим, после этапа проектирования, является создание прототипа кристалла микропроцессора. При изготовлении современных сверхбольших интегральных схем используется метод литографии. При этом на подложку будущего микропроцессора (тонкий круг из монокристаллического кремния, либо сапфира) через специальные маски, содержащие прорезы, поочерёдно наносятся слои проводников, изоляторов и полупроводников. Соответствующие вещества испаряются в вакууме и осаждаются сквозь отверстия маски на кристалле процессора. Иногда используется травление, когда агрессивная жидкость разъедает не защищённые маской участки кристалла. Одновременно на подложке формируется порядка сотни процессорных кристаллов. В результате появляется сложная многослойная структура, содержащая от сотен тысяч до миллиардов транзисторов. В зависимости от подключения транзистор работает в микросхеме как транзистор, резистор, диод или конденсатор. Создание этих элементов на микросхеме отдельно, в общем случае, невыгодно. После окончания процедуры литографии подложка распиливается на элементарные кристаллы. К сформированным на них контактными площадкам (из золота) припаиваются тонкие золотые проводники, являющиеся переходниками к контактными площадкам корпуса микросхемы. Далее, в общем случае, крепится теплоотвод кристалла и крышка микросхемы.

Затем начинается этап тестирования прототипа процессора, когда проверяется его соответствие заданным характеристикам, ищутся оставшиеся незамеченными ошибки. Только после этого микропроцессор запускается в производство. Но даже во время производства идёт постоянная оптимизация процессора, связанная с совершенствованием технологии, новыми конструкторскими решениями, обнаружением ошибок.

