**ВОПРОС 11**

Процессор. Типы архитектур.

**Микропроцессор**

В общем случае процессор — это мозг компьютера. Он читает инструкции из памяти, которые указывают, что делать компьютеру. Инструкции — это просто числа, которые интерпретируются специальным образом. (Инструкция для процессора представлена в числовом виде и Интерпретируется им специальным образом. Из него процессор понимает, что делать, с чем делать, куда помещать и какая команда следующая.)

**Микропроцессоры (CPU) выполняют простые операции:**   
load r1, 150 - загрузка в регистр r1 из ячейки памяти 150

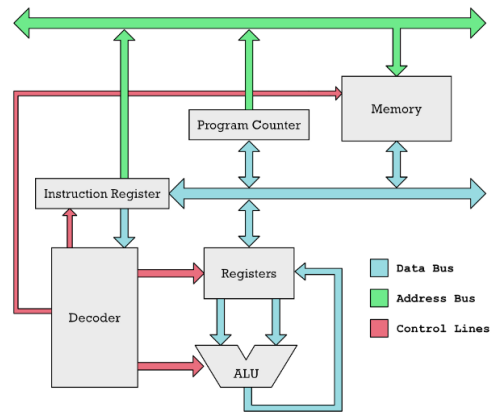
load r2, 200 - загрузка в r2 из ячейки 200

add r1, r2 - сложение содержимого регистров r1 и r2, результат помещается в r1

store r1, 310 - выгрузка r1 в ячейку 310

**Диаграмма операций в процессоре:**

*Program counter, Memory, Instruction Register, Decoder, Registers, ALU*

****

У нас имеется шина данных, шина адреса и шина управления. Они подключены к памяти. Все операции выполняется в АЛУ (арифметико-логическое устройство), который оперирует содержимым регистра. Команда после считывания попадает в инстракшн регистр, после декодируется в декодере и попадает в алу, в котором выполняется. Также показан программный счетчик, который используется для вычисления адреса следующей команды. В случае фиксированной длины команды, когда известна длина команды, счётик просто прибавляет к адресу текущей команды фиксированную величину, по которой можно вычислить адрес следующий команды. Адрес следующей команды выставляется на шину адреса. Оттуда адрес попадает в память, из памяти считывается команда по шине данных. Поступает в регистр команд instruction register. Опять же декодируется, подается на АЛУ и на регистр, процессор выполняет операцию. Ну и процесс продолжает далее.

**Архитектура набора команд (ISA)**

Важной отличительной особенностью процессоров является архитектура набора команд.

У каждого процессора существует фиксированное количество команд, которые он понимает. Каждый тип архитектур имеет свой набор команд. В мире представлено множество различных микропроцессоров, и они не используют одинаковый набор команд. Иными словами, они интерпретируют числа в инструкции по-разному.

Микропроцессоры, например, Intel и AMD, используют архитектуру набора команд x86. А микропроцессоры, например, A12, A13, A14 от Apple, понимают набор команд ARM. Теперь в список ARM-процессоров можно включить M1.

Набор команд x86 и ARM не является взаимозаменяемым. (x86 является запатентованной архитектурой, а АRM является открытой. Компания ARM не продает как таковые процессоры, а разрабатывает концепцию и предоставляет экземпляры процессоров для демонстрации.) Архитектура набора команд сильно влияет на архитектуру процессора. Использование определенной архитектуры набора команд может усложнить или упростить задачу по созданию высокопроизводительного или энергоэффективного процессора.

Набор команд определяет свойства этого процессора.

Изначально производители процессоров стремились увеличить количество и сложность добавляемых команд (даже разложение в ряд фурье добавили). Затем было замечено (примерно в 1985), что компьютер 90% времени выполнял простые команды, поэтому производители процессоров начали стремиться минимизировать время работы простых команд и избавления от сложных команд. Вся нагрузка начала ложиться на компиляторы, которым необходимо было преобразовать сложные команды в простые, поддерживаемые процессором.

**Архитектура CISC**

**CISC — COMPLEX INSTRUCTION SET COMPUTER (РАСШИРЕННЫЙ НАБОР КОМАНД)**

**Отличительные особенности:**

• большой набор команд (поддерживались достаточно сложные высокоуровневые команды.)

• использует переменные (плавающие) диапазоны форматов от 16-64 бит на инструкцию (позволяет экономить память при больших сложных командах)

• использует много способов адресации от 12 до 24 (проще реализовывать компиляцию программы)

• архитектура использует 8-24 регистров общего назначения (мало)

• использует механизм памяти к памяти для выполнения операций (большая часть операций производилась над ячейками памяти и помещает результат в эту же ячейку)

• использует унифицированный кеш для данных и инструкций

• имеет микрокодирование и использует управляющую память (ПЗУ)

**Архитектура RISC**

**RISC - REDUCED INSTRUCTION SET COMPUTER (СОКРАЩЕННЫЙ НАБОР КОМАНД)**

**Отличительные особенности:**

• малый набор команд

• фиксированный формат машинной команды (32 бита) (позволяет очень быстро переходить к следующей команде позволяет, легче декодировать команду)

• использует один такт на команду и режим ограниченной адресации (т.е. способов адресации 3-5)

• число регистров общего назначения колеблется от 32 до 192 (большое)

• основные операции в процессоре выполняются над содержимым регистров - данные загружаются в регистры, затем внутри регистров проводится некая операция (повышается эффективности и скорости вычислений)

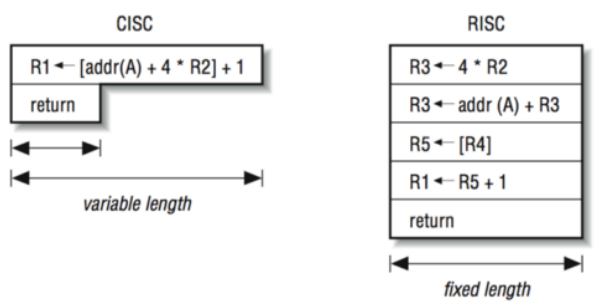
• имеет разделенный дизайн данных и кеш инструкций

• большая часть управления процессором является аппаратной

Хоть RISC процессоры и производительнее (быстрее) CISC при одинаковой тактовой частоте и прочих параметрах, но большинство современных компьютеров используют х86 архитектуру (т.е. это CISС процессоры). Потому что большая часть софта писалось под циск и неэффективно его переписывать под риск.

Такая проблема было уже в 80-х годах. RISC пользователей очень мало, поэтому софт продолжался писаться под CISC. Также на производстве уже имелся налаженный рынок сбыта CISC и себестоимость была меньше, чем у RISC. Поэтому RISC используется в новых устройствах. В планшетах там в телефонах и т.п.

**Отличие форматов команд CISC и RISC**



**Гибридные архитектуры**

Идея — обеспечить программную совместимость с CISC процессорами, но добиться эффективности RISC процессоров.

Микропроцессор имеет интерфейс CISC, а команды выполняет как RISC. Поддерживается конвейеризация.

Процессор имеет циск. Внутри ставится декодер с циск на риск. И внутри вычисления происходят на риск. Конвертация подобного рода занимает время (снижает производительность), но позволяет создать совместимость циск и риск архитектур.