**ВОПРОС 12**

Способы адресации.

**Способ адресации**

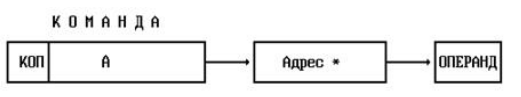
В адресном поле команды содержится командный (исполнительный) адрес (т.е. адресное поле 1, адресное поле 2 и т.д. т.е. все адреса, которые входят в команду.)

Команда выбирается по физическому адресу (в том числе ячейка, связанная с адресом самой команды, ячейка связанная с операндами и ячейка записи результата, они также выбираются процессором по физическому адресу, в большинстве случаев физический адрес не совпадает с командным.)

**Способ адресации**—алгоритм получения физического адреса по командному

То есть, то что содержится в команде преобразуется, и получается физический адрес, по коему и изымается операнд, либо определяется приемник результата (именна та ячейка памяти, куда происходит запись результата), либо адрес следующей команды.

Способ определяется кодом операции. То есть процессор считывает код операции и в нём также определяется порядок работы. (Кодом операции задается способ адресации.)



На рисунке видно, что у нас есть команды, в ней есть некий адрес. Этот адрес является командным, он определенным образом преобразуется. Формируется физический адрес и вынимается операнд из памяти уже по физическому адресу.

**Классификация способов адресации**

**Явные** - в адресном поле есть командный адрес (вот если поле присутствует и там есть какое то значение)

**Неявные** - нет адреса в адресном поле (дескать, второй операнд во внутреннем регистре хранится или сам внутренний регистр процессора подразумевается под приемником результата) (т.е. подразумевается какое то значение)

**Способы адресации. Явные**:

**Непосредственная адресация**: операнд (байт или слово) непосредственно в самом коде команды. (Например если складываем число с константой (скажем 5). Мы указываем код операции, затем адрес операнда и константу)

[ (код операции - адрес операнда - число или слово (скажем, 5) ]

**Прямая адресация**: физический адрес совпадает с командным (единственный в своём виде, редко используется) !!! (используется в курсаче, физический адрес совпадает с командным, чтобы упростить задание)

**Косвенная адресация**: командный адрес содержит адрес ячейки памяти или регистра, где находится операнд. (пример: я не знаю, где этот, но я знаю где тот, кто знает) - удобно, не меняя код, можно менять содержимое ячейки памяти и адресовать разные ячейки памяти, обращаясь к ним одним и тем же кодом

**Автоинкрементная**: после выполнения операции с данными адрес операнда увеличивается на 1, если формат БАЙТ, или на два, если операнд формата СЛОВО или иную константу. Используется для вычисления следующей команды (иногда). Автодекрементная - то же самое, только не +, а - (тот же тип)

**Базовая адресация**: физический адрес является суммой базового адреса и содержимого адресного поля (командного адреса). Базовый адрес обычно хранится в внутреннем регистре процессора, а командный адрес в команде. (Удобно при работе с одномерными массивами, когда адрес начала массива задаётся базовым адресом, а конкретный элемент при помощи командного адреса. При этом массив должен располагаться в памяти одним куском)

**Базово-индексная адресация**: физический адрес является суммой базового, индексного и командного адресов. Базовый и индексный обычно находятся во внутреннем регистре процессора, а командный в команде. Сложение трёх величин этих = физическому адресу. (удобно при работе с двумерными массивами, когда базовый начало расположения массива в памяти, индексный - адрес строки; и командный адрес - конкретный элемент в строке. Меняя содержимое индексного регистра шагаем по строкам, задавая базовый адрес - начало расположения массива в памяти, а командный - обратиться к конкретному элементу массива)

**Укороченная адресация**: адресный код содержит часть физ.адреса, другая лишь подразумевается (сокращает память под адрес и программы в целом). Вычисление физ.адреса по командному заключается в добавлении с одной из сторон, чаще всего слева, к тому адресу, который в команде, некоего числа.

**Регистровая адресация**: адресный код содержит номер регистра. Дескать, надо сделать над регистром операцию. Задается либо номер либо другой признак регистра.

**Стековая адресация**: указывается вершина стека. Процессор изымает из стека один операнд, далее второй, сделает операцию и помещает результат обратно в вершину стека. (часто используется, популярна) [можно для разнообразия ещё рассказать кратко, что такое стек]

**Неявные**:

1. Подразумевается операнд (типа, а++; - подразумевается а = а+1)

Подразумевается адрес операнда или приемника результата (в курсаче машинная команда имеет два поля: код операции и адрес первого операнда. Если взять сложение, например, то первый операнд сложат с тем, что в аккумуляторе, а записываться это всё будет в аккумулятор). Следующая команда - сдвигается на единицу от текущей (текущая + 1).