Числа в ассемблере могут представляться в двоичной, десятеричной или шестнадцатеричной системе.

* Бинарной (двоичной) системы пишется буква b (пример: 0000010b, 001011010b),
* Десятеричной системы можно ничего не указывать после числа или добавить в конце букву d (примеры: 4589, 2356d),
* Шестнадцатеричной системы надо добавить в конце букву h, запись шестнадцатеричного числа начинается с нуля, за которым следуют буквы A-F.

Также шестнадцатеричные числа можно записывать в форматах языков высокого уровня: с начальным знаком $, как в pascal, или сочетанием 0х, как в языке С. Символы значения могут быть заключены как в кавычки, так и в апострофы.

Идентификаторы в программе не могут содержать символов +-\*/=<>() []{}:, | &-#'. Чаще всего используется шестнадцатеричные числа, причём отсчёт чисел будет всегда начинаться с нуля (например, если речь идёт про пятый бит, значит, это шестой бит в привычном понимании).

Для того чтобы указать программе о результате выполнения очередной команды, процессор должен сохранить результат в некотором регистре. Этот регистр называется регистром флагов ЕFLAGS (FLAGS, RFLAGS). Регистр флагов является ключевым в работе процессора: он управляет основополагающими аспектами его работы.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Команда MOV

*Синтаксис команды MOV: mov <назначекие>, <источник>.*

Фактически команда присваивает регистру или памяти некоторое значение, которое можно взять из памяти, регистра или непосредственного значения (просто цифра).

* Если назначением является регистр, то источником может быть память, регистр или непосредственно значение;
* Если назначение - память, то источником может быть только регистр или непосредственное значение (перемещение из памяти в память недопустимо).

Размерность источника и назначения должны быть равными.

**Листинг 1.1. Работа с регистрами при помощи команды MOV**

mov edx, ebx

mov rax, rl0

mov eax, rl2d

mov ah, dh

mov cx, rl4w

mov ecx, 789056h

mov ax, 12h

mov al, 89 Oh ; неправильно

mov dx, rbx ; неправильно

Для работы с памятью надо указать адрес и заключить его в квадратные скобки. Адрес должен быть совместим с текущим режимом работы процессора: например, в режиме реальных адресов необходимо указывать 16-битный адрес, а в 64- битном режиме - 64-разрядный адрес.

**Листинг 1.2. Работа с памятью при помощи команды MOV**

mov еах, dword [0000509Ah]

mov rbx, qword [00A056Flh]

mov dword [0F81562Dh], r9d

mov qword [0F8156F6h], 0DF899564FFh

|  |  |  |
| --- | --- | --- |
|  | операторы размера | Размер в байтах |
|  | Byte | 1 |
|  | word | 2 |
|  | dword | 4 |
|  | fword | 6 |
|  | pword | 6 |
|  | qword | 8 |
|  | tbyte | 10 |
|  | tword | 10 |
|  | dqword | 16 |

64-битном режиме. Вместо квадратных скобок можно указать директиву PTR перед адресом.

**Листинг 1.3. Использование директивы PTR**

mov еах, dword ptr 0000509Ah

mov rbx, qword ptr 00A056Flh

mov qword ptr 0F81562Dh/ r9d

mov qword ptr 0F8156F6h, 09564FFh

Перемещение из памяти в память недопустимо. Иногда размер можно не указывать: например, когда мы переносим значение из памяти в регистр, размер регистра уже показывает размерность операнда. Также можно использовать регистр для адресации памяти.

**Листинг 1.4. Использование регистра для адресации памяти**

mov eax, [eax]

mov rbx, ptr ebx

mov [rax], r9d

mov ptr edx, 09564FFh

Свободно работать можно только с регистрами общего назначения, а работа с остальными регистрами возможна только вкупе с регистрами общего назначения (отсюда и их название). Т. е. перемещать, присваивать и получать значения можно только через регистры общего назначения. Все остальные регистры (не считая регистров математического сопроцессора) будем называть привилегированными регистрами, потому в защищённом режиме доступ к ним можно получить только из привилегированного участка кода. Фактически с привилегированными регистрами может работать только эта команда. Есть также набор специфических для каждого типа машины регистров - регистров MSR. К ним нельзя обратиться с помощью команды MOV, для этого предназначены команды RDMSR (чтение), WRMSR (запись).