

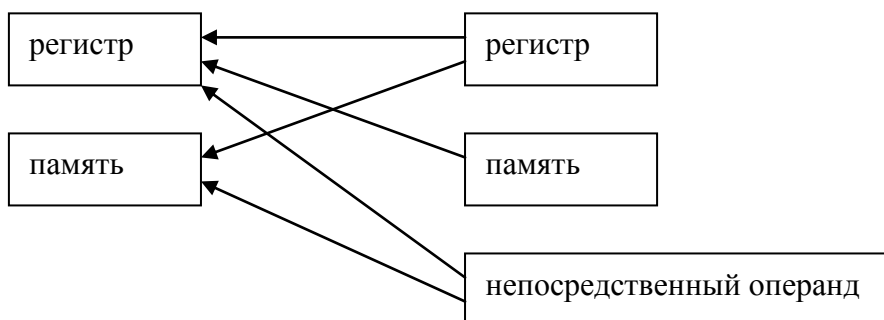
### 3. Команды перемещения данных

Основная команда пересылки имеет мнемонику MOV и является одной из наиболее часто используемых. Это объясняется постоянной необходимостью загрузки регистров данными из оперативной памяти для обработки и сохранением результатов в памяти. Общий формат команды:

MOV приемник, источник

Здесь в качестве источника данных может выступать регистр, именованная область памяти или непосредственный операнд (НО), а в качестве приемника - регистр или область памяти. Пересылать можно байт, слово или двойное слово. Никаких специальных указаний по размеру пересылаемого объекта нет, ассемблер проверяет операнды по их описанию, с учетом непротиворечивости. Если размеры операндов НЕ согласованы, фиксируется ошибка. Никаких преобразований типа «байт  $\Leftrightarrow$  слово» не выполняется.

Нельзя пересылать данные сразу из памяти в память, для этого приходится использовать две команды и промежуточный РОН. Схематично возможные пересылки представлены на следующем рисунке.



Отсюда видно, что существует пять разновидностей передачи данных:

- регистр  $\rightarrow$  регистр
- регистр  $\rightarrow$  память
- память  $\rightarrow$  регистр
- НО  $\rightarrow$  регистр
- НО  $\rightarrow$  память.

Примеры использования команды MOV.

Пусть объявлены две следующие именованные области памяти

Byte1 DB ? ; резервирование 1 байта  
Word1 DB ? ; резервирование 1 слова

Тогда можно записать следующие команды:

MOV AL, 10 ; число 10 в младший байт регистра AX  
MOV AX, 1234 ; число 1234 в регистр AX  
MOV BH, Byte1 ; байт из области памяти с именем Byte1 в старший  
; байт регистра BX  
MOV Word1, BX ; содержимое регистра BX в память с именем Word1  
MOV Byte1, 0 ; число 0 в область памяти с именем Byte1  
MOV BX, CX ; регистр CX в регистр BX

Примеры неправильных команд:

MOV AX, Byte1 ; не согласованы размеры операндов  
MOV BH, CX ; аналогично  
MOV CL, 1000 ; число 1000 требует 2 байтов  
MOV Byte1, Byte2 ; нельзя из памяти в память

Приведем пример программы, в которой используется команда MOV.  
Пусть требуется определить в памяти короткую (3-4 байта) строку и  
«перевернуть» ее в другую область, т.е. скопировать байты в обратном  
порядке, например: abc -> cba

Объявим две области памяти для исходной (S1) и «перевернутой»  
строки (S2):

S1 DB 'abc' ; исходная строка  
S2 DB 3 DUP(?) ; резервируем память для результата

Тогда схематично размещение символов строки в памяти с  
соответствующими именами областей памяти можно представить так:

	a	b	c	.....	c	b	a	.....
имя области памяти	S1	S1+1	S1+2		S2	S2+1	S2+2	

Основной набор команд по перемещению символов будет выглядеть так:

MOV AL, S1 ; символ "а" из памяти с именем S1 в регистр AL  
MOV S2+2, AL ; символ "а" из регистра AL в память с именем S2+2  
MOV AL, S1+1 ; символ "б" из памяти с именем S1+1 в регистр AL  
MOV S2+1, AL ; символ "б" из регистра AL в память с именем S2+1  
MOV AL, S1+2 ; символ "с" из памяти с именем S1+2 в регистр AL  
MOV S2, AL ; символ "с" из регистра AL в память с именем S2