Команда XLATB в Ассемблере выполняет замену (подмену) байта в [**регистре**](http://av-assembler.ru/asm/afd/asm-cpu-registers.htm) AL байтом из таблицы - последовательности байтов в памяти.

Синтаксис: XLATB

Команда XLATB не имеет операндов. Никакие [**флаги**](http://av-assembler.ru/asm/afd/asm-flags-register.htm) при выполнении команды не изменяются.

С помощью этой команды можно, например, выполнять перекодировку символов, подменяя код в регистре AL кодом из таблицы в памяти.

Адрес таблицы должен находиться здесь:

DS:[BX + беззнаковый AL]

Алгоритм работы команды XLATB:

* вычислить адрес, равный DS:BX+(AL)
* выполнить замену байта в регистре AL байтом из памяти по вычисленному адресу.

Пример на ассемблере:

.model tiny

.code

ORG 100h

[**LEA**](http://av-assembler.ru/instructions/lea.php) BX, TABLE ;ВХ - адрес TABLE

[**MOV**](http://av-assembler.ru/instructions/mov.php) AL, 2 ;DS:BX = DS:[адрес TABLE + 2]

**XLATB** ;AL = 43h (код исмвола C)

[**RET**](http://av-assembler.ru/instructions/ret.php) ;Выйти из программы

TABLE DB 'AB**C**DEFGHIJKLMNOPQRSTUVWXYZ'

Здесь мы сначала записываем в ВХ адрес последовательности байтов - таблицы TABLE. Затем в регистр AL записываем адрес символа из этой таблицы. Поскольку адресация начинается с нуля, то символ с адресом 2 - это 3-й символ в этой таблице, то есть символ **С**.

Затем выполняем команду XLATB, которая в регистр AL записывает ASCII-код третьего символа из таблице - число 43h.

Таким образом код 2h мы преобразовали в код 43h. Ну а что со всем этим делать - это уже вам подскажет ваше воображение.

Напоследок, как всегда, о происхождении аббревиатуры XLATB.

**XLATB** - это trans**LAT**e **B**yte from table (преобразовать байт из таблицы).