



Как стать программистом

Бесплатная книга о программировании
для начинающих и бывалых.

[Получить >>>](#)

Подписаться:



[Главная](#) [Ассемблер](#) [Микроконтроллеры](#) [Инструкции Intel](#) [Дневник](#)



Микроконтроллеры для ЧАЙНИКОВ

[Изучать БЕСПЛАТНО](#)

14.09.2020 г.

Добавлена статья [Уменьшение энергопотребления.](#)

05.09.2020 г.

Добавлены видео и статья [Самое простое устройство на микроконтроллере](#).

[21.08.2020 г.](#)

Добавлены видео и статья [Инструкция CLI](#).

[19.06.2020 г.](#)

Добавлена статья [Выводы ATtiny13A](#).

[19.05.2020 г.](#)

Добавлена статья [Регистр PRR](#).

Инструкция SBB



Что такое JavaScript

Если вы интересуетесь программированием вообще, и сайтостроением в частности, то вы наверняка слышали слово JavaScript. И, если вы до сих пор не узнали толком, что же это такое, то пришло время сделать это.

[Подробнее...](#)

Инструкция SBB



Команда SBB в Ассемблере выполняет вычитание с заёмом. Синтаксис:

SBB РАЗНОСТЬ, ЧИСЛО

Состояние [флагов](#) зависит от результата.

ЧИСЛОМ может быть одно из следующих:

- Область памяти (MEM)
- [Регистр](#) общего назначения (REG)
- Непосредственное значение (например, число) (IMM)

РАЗНОСТЬЮ может быть одно из следующих:

- Область памяти (MEM)
- Регистр общего назначения (REG)

Эта команда, также как и [команда SUB](#), не работает с сегментными регистрами. Комбинации РАЗНОСТЬ-ЧИСЛО могут быть следующими:

REG,	MEM
MEM,	REG
REG,	REG
MEM,	IMM
REG,	IMM

Команда SBB работает также, как и команда SUB, но кроме вычитания ЧИСЛА из РАЗНОСТИ она ещё вычитает из результата значение флага переноса CF. То есть алгоритм работы команды SBB такой:

РАЗНОСТЬ = РАЗНОСТЬ - ЧИСЛО - CF

Вычитание с заёмом

Эту команду можно использовать для вычитания 64-разрядных чисел на 32-разрядных процессорах, или 32-разрядных чисел на 16-разрядных процессорах.

В первом случае можно использовать пары регистров EDX:EAX и EBX:ECX, а во втором DX:AX и BX:CX. Если при вычитании значений в младших регистрах произошёл заём, то он будет учтён при вычитании старших слов.

Пример:

```
.model tiny
.code
ORG 100h

start:

MOV DX, 1 ;В паре DX:AX (0001:0000)
MOV AX, 0 ;32-разрядное число 65536
MOV BX, 0 ;В паре BX:CX
MOV CX, 1 ;32-разрядное число 1
;65536 - 1 = 65535, то есть после вычитания
;DX:AX = 65535 (0000:FFFF)
SUB AX, CX ;AX = FFFF
SBB DX, BX ;DX = 0000

RET

END start
```

В этом примере мы используем 16-разрядные регистры для выполнения операции вычитания 32-разрядных чисел.

В паре регистров DX:AX у нас будет число 65536 (00010000h), которое не помещается в один 16-разрядный регистр.

От этого числа мы отнимаем число 1. Это число можно отнять как непосредственное значение, но для понимания того, что можно отнять любое другое 32-разрядное число, мы поместим единицу в пару регистров BX:CX (число будет 00000001h).

Затем сначала мы из младшего регистра первого числа вычитаем младший регистр второго числа с помощью команды SUB. Результат будет равен FFFF, поскольку мы от нуля отнимаем единицу, а флаг переноса CF будет установлен.

Затем с помощью команды SBB мы от старшего регистра первого числа отнимаем старший регистр второго числа (от единицы мы отнимаем ноль). Если бы мы использовали команду SUB, то в регистре DX мы бы получили 1. То есть пара DX:AX содержала бы 32-разрядное число 1FFFFh (131071 в десятичной):

$65536 - 1 = 131071$

что, разумеется, неправильно.

Но мы используем команду SBB, которая кроме того, что вычитает в нашем примере из DX значение BX, ещё отнимает от результата значение флага переноса (которое в нашем примере равно 1). В итоге получаем правильное значение DX:AX = 0000:FFFF = 65535.

Ну и напоследок, как всегда, о происхождении аббревиатуры SBB.

SBB - это **SuB**tract with **B**orrow - вычитание с заимствованием.

[Подписаться на канал в YouTube](#)

[Вступить в группу "Основы программирования"](#)

[Подписаться на рассылки по программированию](#)



Первые шаги в программирование

Главный вопрос начинающего программиста – с чего начать? Вроде бы есть желание, но иногда «не знаешь, как начать думать, чтобы до такого додуматься». У человека, который никогда не имел дело с информационными технологиями, даже простые вопросы могут вызвать большие трудности и отнять много времени на решение. [Подробнее...](#)

Инфо-МАСТЕР®

Все права защищены ©

e-mail: mail@info-master.su

[Главная](#)

[Карта](#)

[Контакты](#)

