



Основы C++

Неважно, на каком языке вы программируете. Если вы не знаете C++, вы не можете считать себя программистом.

[Подробнее >>>](#)

Подписаться:



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



Микроконтроллеры для ЧАЙНИКОВ
[Изучать БЕСПЛАТНО](#)

http://av-assembler.ru запрашивает разрешение на:

Показывать оповещения

[14.09.2020](#)

Добавлена

Предоставлено SendPulse

[Блокировать](#)

[Разрешить](#)

[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 AAD



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...](#)

http://av-assembler.ru запрашивает разрешение на:

Показывать оповещения

Предоставлено SendPulse

[Блокировать](#) [Разрешить](#)

Команда AAD



Инструкция AAD в Ассемблере подготавливает два [BCD-значения](#) для деления. У этой команды нет операндов.

Команда AAD работает согласно следующему алгоритму:

```
AL = AH * 10 + AL
AH = 0
```

[Флаги](#) ZF, SF и PF изменяются в зависимости от результата выполнения команды. Остальные флаги имеют неопределенное состояние.

Команда AAD преобразует неупакованное [двоично-десятичное число](#), которое находится в [регистре AX](#). Делается это таким образом, чтобы последующее деление привело к десятичному результату.

Команда AAD может использоваться с любой [системой счисления](#). [Машинный код](#) этой команды D5h **0Ah**. В этом коде второй байт можно заменить на другое число.

Действие инструкции AAD заключается в том (см. алгоритм выше), что содержимое регистра AH умножается на второй байт команды (по умолчанию это **0Ah**) и складывается с содержимым регистра AL. После этого регистр AH обнуляется. Команду AAD можно также использовать для быстрого умножения на число с записью результата в AL.

Пример программы:

```
.model tiny
.code
ORG 100h

start:
    MOV AX, 1000h
    AAD
    MOV BL, 5
    DIV BL
    ;37 / 5
```

http://av-assembler.ru запрашивает разрешение на:

Показывать оповещения

Предоставлено SendPulse

[Блокировать](#) [Разрешить](#)

[RET](#)

END start

Здесь мы помещаем в регистр AX число 37 в BCD-формате. Затем с помощью команды AAD преобразуем его в десятичное число. Далее делим его на 5 и получаем 7 в регистре AL и 2 в регистре AH (остаток от деления).

А теперь об умножении с помощью команды AAD. Для того, чтобы умножить содержимое регистра AH не на 10, как это делается по умолчанию, а на другое число, нам надо перед этим записать в то место памяти, где расположен второй байт команды AAD, другое значение.

Для этого нам надо знать адрес этого байта. Наша программа начинается с адреса DS:100h (значение 100h определено в начале программы директивой ORG). Зная количество команд в программе перед командой AAD и сколько какая команда занимает места в памяти, мы можем вычислить этот адрес. Ну а если говорить об эмуляторе Emu8086, то мы просто можем посмотреть этот адрес в окне отображения памяти (см. видео выше).

А код будет примерно таким:

```
MOV [BYTE PTR DS:0113h], 2 ;Множитель
MOV AX, 0300h           ;AH - умножаемое число
AAD                   ;Теперь AL = 6
```

Здесь красным выделен адрес второго байта команды AAD. В вашем случае этот адрес, разумеется, может быть другим.

Ну и напоследок о том, как образовалась мнемоника AAD.

AAD - это **ASCII Adjust before Division** - ASCII-коррекция перед делением.

[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 дела 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@av-assembler.ru

[Главная](#)

[Карта](#)

[Контакты](#)

<http://av-assembler.ru> запрашивает разрешение на:

Показывать оповещения

Предоставлено SendPulse

[Блокировать](#)

[Разрешить](#)

