

*Арифметические операции
(1-ая часть)*

- Обработка двоичных данных
- Беззнаковые и знаковые данные
- Умножение
- Сдвиг регистровой пары DX: AX

План:

- Сложение и вычитание
- Переполнение

Обработка двоичных данных

- Команды ADD и SUB выполняют сложение и вычитание байтов или слов, содержащих двоичные данные.
- Пример:
 - MOV AX,WORDA
 - ADD AX,WORDB
 - MOV WORDB,AX

Сложение и вычитание

- ADD AL,20H
- CBW ;Расширение AL до AX
- ADD AX,20H ;Прибавить к AX

Переполнения

- Команды ADD и SUB не делают разницы между знаковыми и беззнаковыми величинами, они просто складывают и вычитают биты.

Беззнаковые и знаковые данные

Беззнаковое Знаковое

1111 1001 249 -7

+ + + +

0000 0010 2 +2

1111 1011 251 -5

- Байт на байт
- Слово на слово
- Беззнаковое умножение
- Знаковое умножение
- Повышение эффективности умножения
- Многословное умножение
- Умножение двойного слова на слово
- Умножение двойного слова на двойное слово

Умножение

- Операция умножения для беззнаковых данных выполняется командой MUL, а для знаковых - IMUL
- *Байт на байт* Множимое находится в регистре AL, а множитель в байте памяти или в однобайтовом регистре. После умножения произведение находится в регистре AX. Операция игнорирует и стирает любые данные, которые находились в регистре AH.
- *Слово на слово* Множимое находится в регистре AX, а множитель - в слове памяти или в регистре. После умножения произведение находится в двойном слове, для которого требуется два регистра: старшая (левая) часть произведения находится в регистре DX, а младшая (правая) часть в регистре AX. Операция игнорирует и стирает любые данные, которые находились в регистре DX.

- MUL MULTR

MUL CL ;Байт-множитель: множимое в AL,
произвед. в AX

MUL BX ;Слово-множитель:множимое в AX,
произвед. в DX:AX

- Команда MUL (MULtiplication - умножение) умножает беззнаковые числа.

Беззнаковое умножение: Команда MUL

- Команда IMUL (Integer MULtiplication - умножение целых чисел) умножает знаковые числа.

Знаковое умножение: Команда IMUL

- Умножение на 2:

```
SHL AL,1
```

- Умножение на 8:

```
MOV CL,3
```

```
SHL AX,CL
```

Повышение эффективности умножения

1365

x

12

2730

1365

16380

Многословное умножение

$$\begin{array}{r} 13 \quad 65 \\ \times \quad \times \\ \hline 12 \quad 12 \\ \hline 26 \quad 130 \\ 13 \quad 65 \\ \hline 156 \quad 780 \end{array}$$

15600

+

780

16380

- Процедура E10XMUL умножает двойное слово на слово. Множимое, MULTCND, состоит из двух слов, содержащих соответственно шест.3206 и шест.2521. Определение данных в виде двух слов (DW) вместо двойного слова (DD) обусловлено необходимостью правильной адресации для команд MOV, пересылающих слова в регистр AX. Множитель MULTPLR содержит шест.6400.

Умножение двойного слова на слово

- Область для записи произведения, PRODUCT, состоит из трех слов.
- Первая команда MUL перемножает MULTPLR и правое слово поля MULTCND; произведение - шест. 0E80 E400 записывается в PRODUCT+2 и PRODUCT+4.
- Вторая команда MUL перемножает MULTPLR и левое слово поля MULTCND, получая в результате шест. 138A 5800.

Произведение 1: 00000E80E400

Произведение 2: 138A5800

—

Результат: 138A6680E400

- слово 2 слово 2
- слово 2 слово 1
- слово 1 слово 2
- слово 1 слово 1
- Каждое произведение в регистрах DX и AX складывается с соответствующим словом в окончательном результате.

Умножение двойного слова на двойное слово

Сдвиг влево на 4 бита

MOV CX,04 ;Инициализация на 4 цикла C20:

SHL DX,1 ;Сдвинуть DX на 1 бит влево

SHL AX,1 ;Сдвинуть AX на 1 бит влево

ADC DX,00 ;Прибавить значение переноса

LOOP C20 ;Повторить Сдвиг вправо на 4 бита

MOV CX,04 ;Инициализация на 4 цикла D20:

SHR AX,1 ;Сдвинуть AX на 1 бит вправо

SHR DX,1 ;Сдвинуть DX на 1 бит вправо

JNC D30 ;В случае, если есть перенос,

OR AH,10000000B ; то вставить 1 в AH D30:

LOOP D20 ;Повторить

Сдвиг регистровой пары DX:AX

MOV CL,04 ;Установить фактор сдвига
SHL DX,CL ;Сдвинуть DX влево на 4 бита
MOV BL,AH ;Сохранить AH в BL
SHL AX,CL ;Сдвинуть AX влево на 4 бита
SHR BL,CL ;Сдвинуть BL вправо на 4 бита
OR DL,BL ;Записать 4 бита из BL в DL