

## 13. Команды побитовых операций.

### Логические команды.

---

- **AND** <приёмник - регистр, переменная>, <источник - число, регистр, переменная> - побитовое "И". Источник и приемник не могут быть переменными одновременно Пример: AND al, 00001111b
- **OR** <приёмник - регистр, переменная>, <источник- число, регистр, переменная> - побитовое "ИЛИ". Источник и приемник не могут быть переменными одновременно Пример: OR al, 00001111b
- **XOR** <приёмник- регистр, переменная>, <источник- число, регистр, переменная> - побитовое исключающее "ИЛИ". Любой бит результата равен 1, если соответствующие биты операндов различны, и нулю - в противном случае. Источник и приемник не могут быть переменными одновременно Пример: XOR AX, AX
- **NOT** <приёмник> - инверсия, флаги не затрагиваются

Все эти команды меняют регистр FLAGS.

### Логический, арифметический, циклический сдвиг

---

- SAR приёмник, счетчик - Арифметический сдвиг вправо
- SAL приёмник, счетчик - Арифметический сдвиг влево
- SHR приёмник, счетчик - Логический сдвиг вправо
- SHL приёмник, счетчик - Логический сдвиг влево

Эти четыре команды выполняют двоичный сдвиг приемника (регистр или переменная) вправо (в сторону младшего бита) или влево (в сторону старшего бита) на значение счетчика (число или регистр CL, из которого учитываются только младшие 5 бит, принимающие значения от 0 до 31). Операция сдвига на 1 эквивалентна умножению (сдвиг влево) или делению (сдвиг вправо) на 2.

Команды SAL и SHL выполняют одну и ту же операцию (на самом деле это одна и та же команда) - на каждый шаг сдвига старший бит заносится в CF, все биты сдвигаются влево на одну позицию, младший бит обнуляется.

Команда SHR осуществляет прямо противоположную операцию: младший бит заносится в CF, все биты сдвигаются на 1 вправо, старший бит обнуляется. Эта команда эквивалентна беззнаковому целочисленному делению на 2. Команда SAR действует по аналогии с SHR, только старший бит не обнуляется, а сохраняет предыдущее значение. SAR, таким образом, эквивалентна знаковому делению на 2, но, в отличие от IDIV, округление происходит не в сторону нуля, а в сторону отрицательной бесконечности.

Для сдвигов на несколько битов значение OF не определено. Флаги SF, ZF, PF назначаются всеми сдвигами в соответствии с результатом, параметр AF не определен (кроме случая, когда счетчик сдвига равен нулю: ничего не происходит и флаги не изменяются).

- ROR приёмник, счетчик - Циклический сдвиг вправо

- `ROL` приемник, счетчик - Циклический сдвиг влево
- `RCR` приемник, счетчик - Циклический сдвиг вправо через флаг переноса CF
- `RCL` приемник, счетчик - Циклический сдвиг влево через флаг переноса

Эти команды осуществляют циклический сдвиг приемника (регистр или переменная) на число битов, указанное в счетчике (число или регистр CL, из которого учитываются только младшие 5 бит, принимающие значения от 0 до 31). При выполнении циклического сдвига на 1 команды ROR (ROL) перемещают каждый бит приемника вправо (влево) на одну позицию, за исключением самого младшего (старшего), который записывается в позицию самого старшего (младшего) бита. Команды RCR и RCL выполняют аналогичное действие, но включают флаг CF в цикл, как если бы он был дополнительным битом в приемнике.

После выполнения команд циклического сдвига флаг CF всегда равен последнему вышедшему за пределы приемника биту, флаг OF определен только для сдвигов на 1 - он устанавливается, если изменилось значение самого старшего бита, и сбрасывается в противном случае. Флаги SF, ZF, AF и PF не изменяются

## Операции над битами и байтами

---

База - регистр или переменная.

Смещение - от 0 до 15/31 в зависимости от размера регистра(базы), иначе берётся остаток от деления на 16/32

- `BT <база>, <смещение>` - считать в CF значение бита из битовой строки
- `BTS <база>, <смещение>` - установить бит в 1
- `BTR <база>, <смещение>` - сбросить бит в 0
- `BTC <база>, <смещение>` - инвертировать бит
- `BSF <приёмник>, <источник>` - прямой поиск бита (от младшего разряда)
- `BSR <приёмник>, <источник>` - обратный поиск бита (от старшего разряда)

Сканирование битов выполняется командами BSF и BSR. Эти команды очень похожи. У них 2 операнда. Первый операнд должен быть 16-битным регистром, в него записывается результат. Второй операнд может быть 16-битным регистром или словом в памяти — это обрабатываемое значение.

Команда BSF просматривает биты второго операнда от младшего к старшему и помещает индекс первого единичного бита в регистр. Биты нумеруются, начиная с нуля. Если единичный бит найден, то флаг нуля сбрасывается (ZF=0). Если все биты нулевые, то флаг нуля устанавливается (ZF=1), а значение первого операнда будет неопределённым (на разных процессорах может быть по-разному).