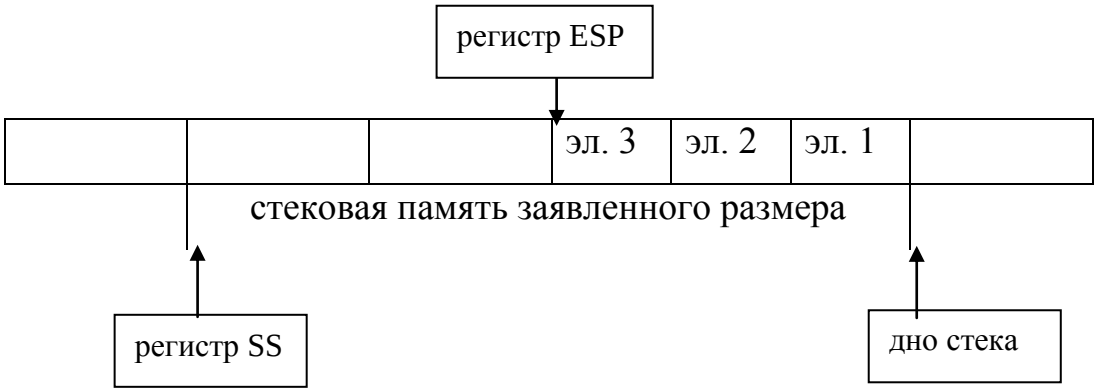
10. Работа со стеком.

Стек является важнейшим механизмом вычислительной системы, и поэтому работа со стеком изначально заложена в систему команд процессоров. Фактически, стек – это область оперативной памяти со специальным использованием. Начальный адрес стековой памяти сохраняется в специальном регистре SS (от Stack Segment). Для каждой программы под стек отводится некоторая область памяти с заявленным размером. Размер этой области определяется интенсивностью использования стека в программе. В процессорах семейства Intel стек растет от конца заявленной области к началу, т.е. от старших адресов к младшим. Адрес вершинного элемента сохраняется в регистре ESP (от Stack Pointer).



Основными командами работы со стеком являются две команды – занесение в стек (PUSH) и выталкивание с вершины стека (POP). Особенностью их использования является то, что запись и чтение выполняется двойными словами, т.е.по 4 байта. Обе команды – однооперандные, причем операндом может быть либо РОН, либо именованная область памяти, а у команды PUSH еще и непосредственная константа. В силу этого формат команд очень прост:

```
PUSH    операнд     ; занести операнд в стек
POP     операнд     ; извлечь значение с вершины стека в операнд
```

Команда PUSH сначала уменьшает значение в регистре ESP на 4, а потом заносит в память по полученному адресу значение операнда. Команда POP,

наоборот, сначала извлекает из памяти значение и потом уже увеличивает регистр ESP на 4.

Необходимо помнить, что команды PUSH и POP не проверяют особые ситуации пустоты и заполненности стека и поэтому их приходится проверять отдельно. Условие отсутствия в стеке свободных мест определяется соотношением ESP = 0, а условие пустоты – соотношением ESP = размер стековой памяти.

Основное назначение стека – передача параметров между подпрограммами, но иногда бывают полезными следующие приемы:

- временное сохранение в стеке значений одного или нескольких регистров
- пересылка данных между двумя областями памяти без использования промежуточных регистров:

```
PUSH    From    ; из области памяти с именем From в стек
POP     To      ; из стека в область памяти с именем To
```

Кроме двух основных команд PUSH и POP есть еще 4 дополнительные команды работы со стеком:

- запись регистра EFlags в стек (PUSHF)
- чтение двойного слова с вершины стека с занесением в регистр EFlags (POPF)
- запись в стек всех РОНов сразу (PUSHA) в порядке: EAX, ECX, EDX, EBX, ESP, EBP, ESI, EDI
- чтение из стека восьми двойных слов с записью их в соответствующие РОНы в обратном порядке

Для полной или частичной очистки стека можно не удалять из него элементы, достаточно правильно изменить регистр ESP. Например, если надо удалить N элементов, то достаточно прибавить к ESP величину 4N:

```
ADD   ESP,   4*N
```

Наконец, если нужен доступ к одной или нескольким ячейкам стека БЕЗ их удаления и стека, можно использовать индексный регистр BP (только этот!).