

# SIMATIC

## S7-200 Примеры

|               |
|---------------|
| <b>Группа</b> |
|---------------|

|   |
|---|
| 2 |
|---|

|                      |
|----------------------|
| <b>Пример к теме</b> |
|----------------------|

|   |
|---|
| Различные возможности, установки битов или байтов |
|---|

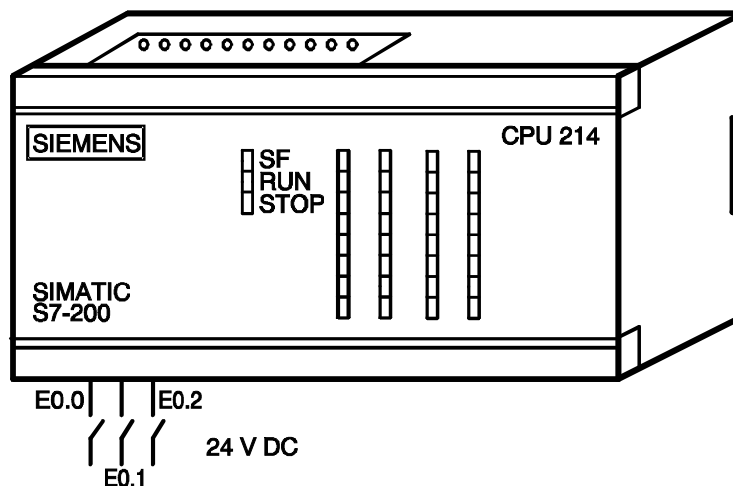
### Краткое описание

Данная программа описывает различные возможности занесения в области памяти определенных значений или очистки определенных областей памяти.

В этом примере применения рассмотрены:

- Команда FILL,
- цикл FOR-NEXT с применением указателя и
- команда RESET.


### Схема включения



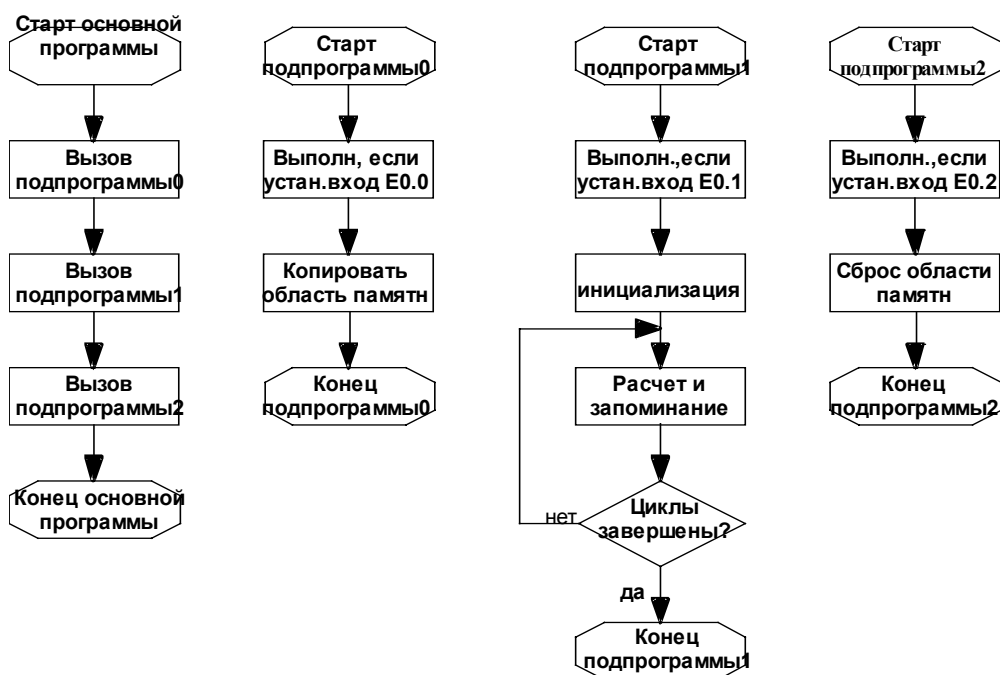
Industrial automation

**Elincom Group**

 European Union: [www.elinco.eu](http://www.elinco.eu)

 Russia: [www.elinc.ru](http://www.elinc.ru)

## Структура программы



## Описание программы вкл. листинг

Этот пример программы описывает различные возможности занесения определенных значений в заданные биты и байты или очистки определенных областей памяти.

Используются команды:

|             |  |
|-------------|--|
| FILL        | Устанавливает один или несколько битов |
| FOR... NEXT | Цикл FOR...NEXT                        |
| R           | Сбрасывает один или несколько битов    |

Размер программы составляет 55 слов.

Более подробную информацию об установке и сбросе битов и байтов Вы найдете в Руководстве по программированию SIMATIC S7-200 в Главе 4.7 "Операции с областями данных" (FILL), 5.6 "Операции FOR и NEXT" и 3.5 "Логические операции с выходами" (RESET).

|                   |                |
|-------------------|----------------|
| KOP (S7-MicroDOS) | AWL (TOOLITE2) |
|-------------------|----------------|

### Основная программа

```
// TITEL=FORNEXT
```

```
// Основная программа содержит вызовы подпрограмм 0, 1 и 2.
```

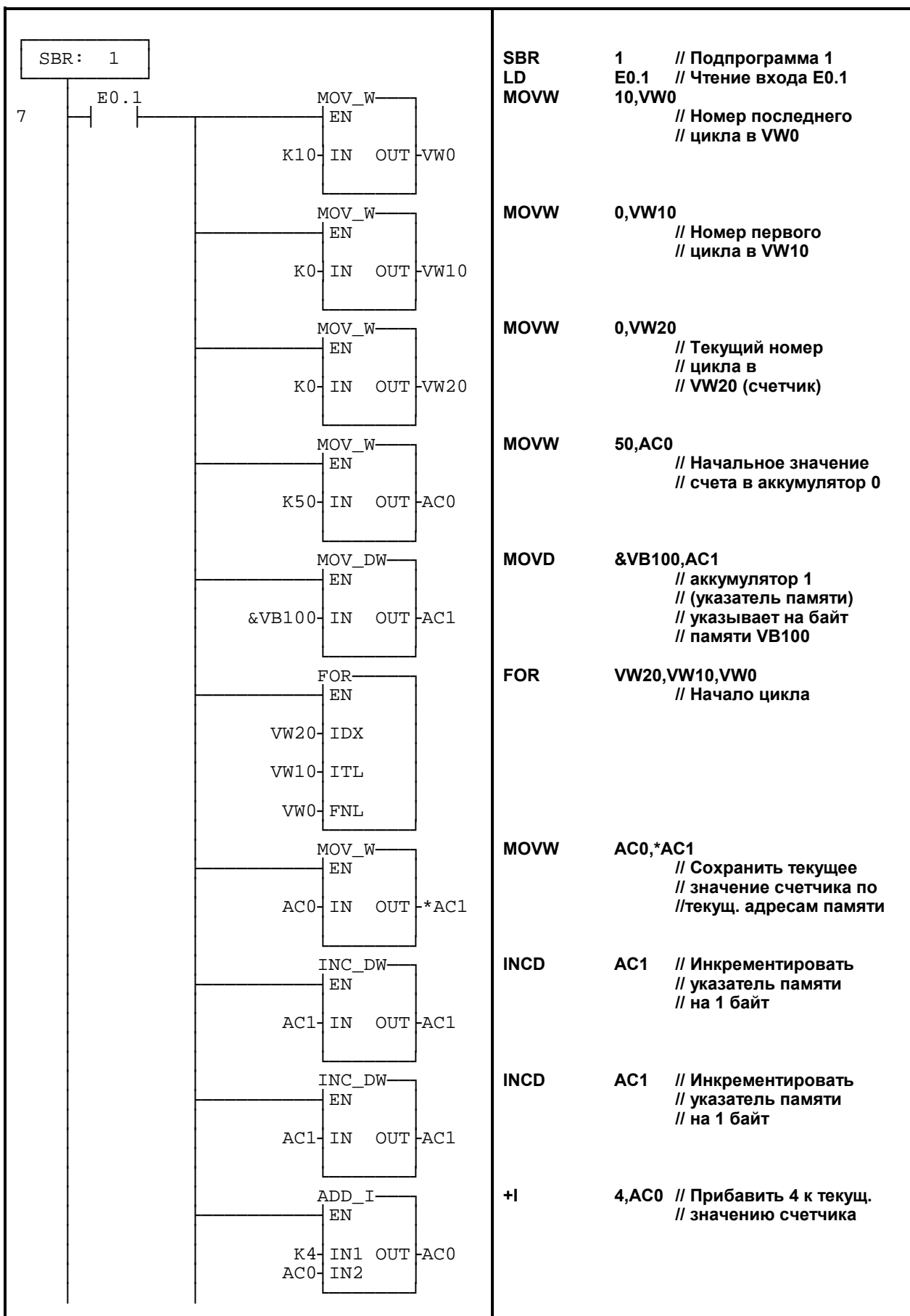
|  |  |
|--|--|
|  | <pre>LD      SM0.0 // Всегда в единице CALL    0     // Вызов подпрограммы 0            // (FILL) CALL    1     // Вызов подпрограммы 1            // (FOR...NEXT) CALL    2     // Вызов подпрограммы 2            // (Reset) MEND      // Конец основной            // программы</pre> |
|--|--|

### Подпрограммы

```
// Подпрограмма 0 копирует значение слова VW200 в слово VW204
// и в 6 последующих слов памяти (до VW216), если установлен
// вход E0.0.
```

|  |   |
|--|---|
|  | <pre>SBR      0     // Подпрограмма 0  LD      E0.0  // Чтение входа E0.0 MOVW    16#ABC3,VW200            // Запись 16-рич.значения            // ABC3 в VW200  FILL    VW200,VW204,7            // Копирование значения            // слова VW200 в            // слово VW204 и шесть            // последующих            // слов памяти (до            // VW216) RET      // Конец подпрограммы 0</pre> |
|--|---|

```
// Подпрограмма 1 копирует следующие друг за другом числа в переменную
// область памяти, если установлен вход E0.1. Число выполняемых
// циклов определяется номерами первого (в VW10) и последнего
// цикла (в VW0). Текущее значение выполненных циклов находится в
// слове памяти VW20. Первое число, с которого надо начинать счет,
// загружается в аккумулятор AC0. Первый адрес, под которым надо
// запоминать значение, заносится в аккумулятор AC1, действующий
// в качестве указателя. Затем начинается выполнение циклов. К
// начальному значению AC0 за один цикл прибавляется 4, указатель
// AC1 увеличивается на следующее слово памяти, пока не будет
// обработан последний цикл.
```



|   |   |  |
|---|---|--|
| 8   |   | <b>NEXT</b> // Конец цикла<br><br><b>RET</b> // Конец подпрограммы 1   |
| // Подпрограмма 2 сбрасывает биты памяти с V100.0 до V121.7 и<br>// с V204.0 до 217.7, если установлен вход E0.2. |   |  |
| 10  | <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">SBR : 2</div> | <b>SBR</b> 2     // Подпрограмма 2<br><br><b>LD</b> E0.2 // Установка входа E0.2<br><b>R</b> V100.0,176<br>// Сброс битов с V100.0<br>// до 121.7<br><br><b>R</b> V204.0,112<br>// Сброс битов с V204.0<br>// до 217.7<br><br><b>RET</b> // Конец подпрограммы 2 |
| 11  |   | <b>RET</b> // Конец подпрограммы 2   |

### Указания по преобразованию

Для того чтобы преобразовать TOOLITE2 AWL в S7-Micro/DOS AWL

- Установите 'K' перед каждым числом, не являющимся 16-ричной константой (напр. 4 → K4)
- Замените '16#' → 'KH' для всех 16-ричных констант (напр. 16#FF → KHFF)
- Поставьте запятые для смены полей. Используйте клавиши перемещения или клавишу TAB для перехода от поля к полю.
- Для преобразования программы S7-Micro/DOS AWL в KOP-форму нужно начинать каждый сегмент словом 'NETWORK' и номером. Каждый сегмент в этом примере имеет свой номер на диаграмме KOP. Используйте NWENFG в меню редактора для ввода нового сегмента. Команды MEND, RET, RETI, LBL, SBR и INT требуют отдельных сегментов.
- Комментарии к строкам, начинающиеся с "//" в S7-Micro/DOS невозможны, зато возможны комментарии к сегментам.

### Общие указания

Примеры SIMATIC S7-200 предоставляются заказчику бесплатно. Данные примеры не привязаны к конкретной задаче и являются общей информацией о возможностях применения S7-200. Решение заказчика может отличаться от приведенного здесь.

За правильную работу системы заказчик несет ответственность сам. Мы обращаем Ваше внимание на действующие нормы Вашей страны и предписания по установке соответствующей системы. Ошибки и изменения возможны.