

# SIMATIC

## S7-200 Примеры

**Группа**

2

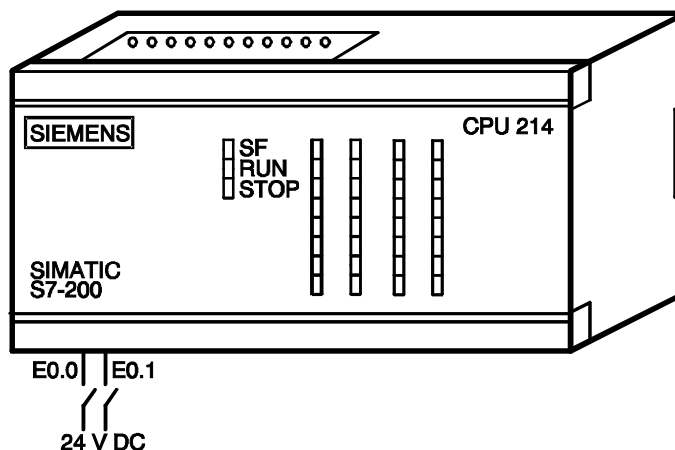
**Пример к теме**

Управление прерываниями по времени

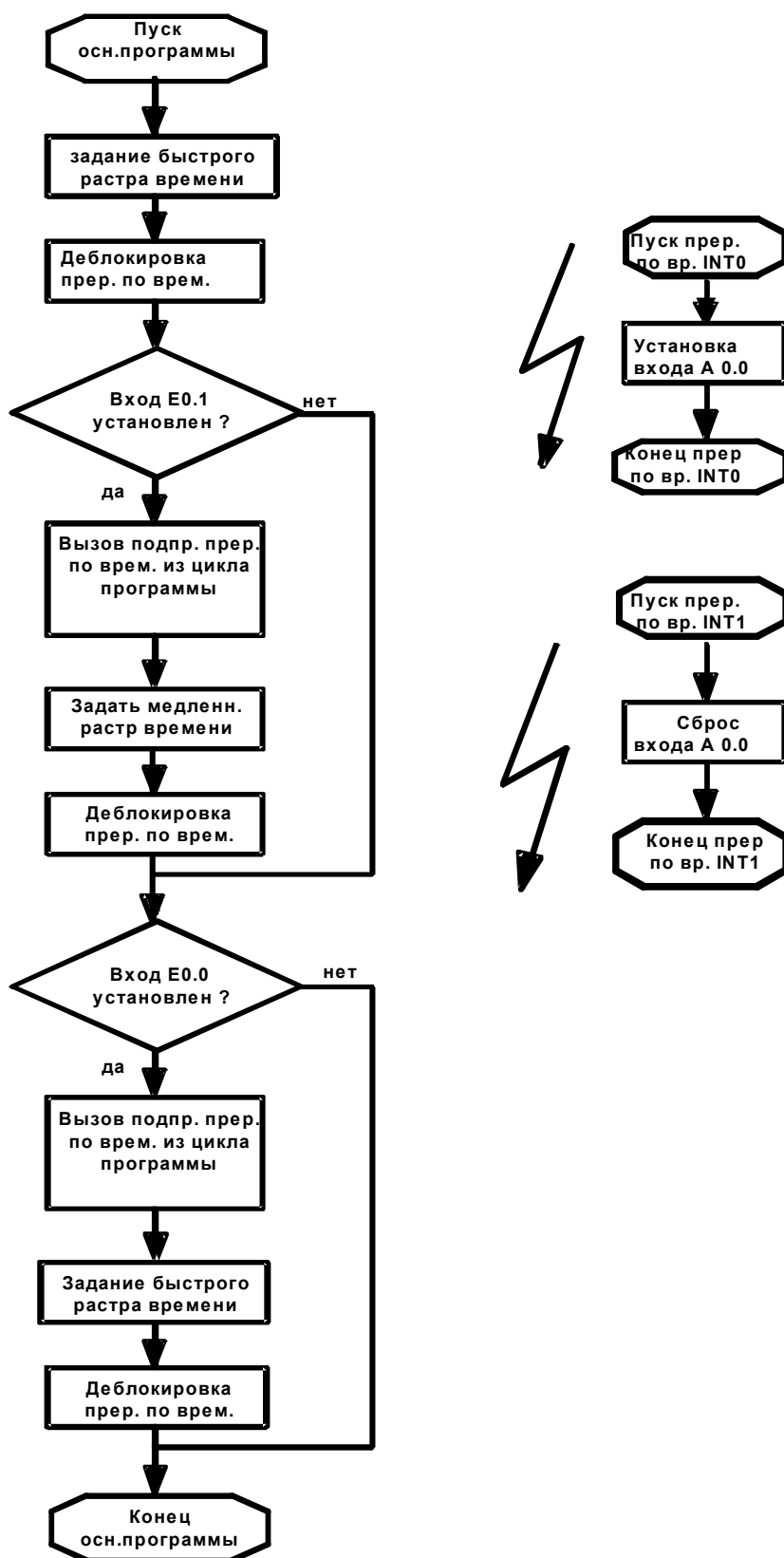
**Краткое описание**

С помощью прерываний по времени создается тактовое мигание, причем частота мигания уполовинивается нажатием выключателя E0.1. Подача сигнала на вход 0.0 восстанавливает первоначальную частоту.

Этот пример призван пояснить общую работу с прерываниями по времени и изменением временного раstra.

**Схема включения**

## Структура программы



**Описание программы вкл. листинг**

В байт специальных меркеров SMB34 заносится растр времени, с которым вызывается прерывание по времени с номером 10 (первое прерывание по времени). В байт специальных меркеров SMB 35 растр времени, с которым вызывается прерывание по времени с номером 11 (второе прерывание по времени - поддерживается только CPU 214). Растр времени в обоих случаях будет определен с инкрементом 1мс. Минимальное допустимое значение растра времени составляет 5 мс, максимальное - 255 мс.

Программа состоит из следующих подпрограмм:

Main	Инициализация и задание времени
INT 0	Установка выхода A0.0
INT 1	Сброс выхода A0.0

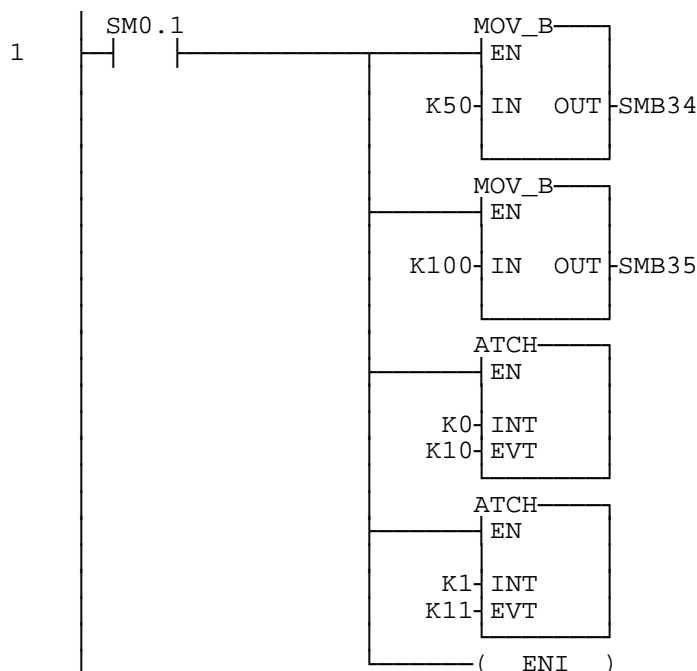
Размер программы составляет 51 слово.

Подробнее о прерываниях управляемых по времени Вы найдете в главе 6.2 "Операции с прерываниями" в руководстве по программированию SIMATIC S7-200.

**KOP (S7-MicroDOS)****AWL (TOOLITE2)****Основная программа**

// TITEL = TIME-INT

// В этой первой части основной программы задается начальный растр времени  
// и связываются два прерывания управляемых по времени.



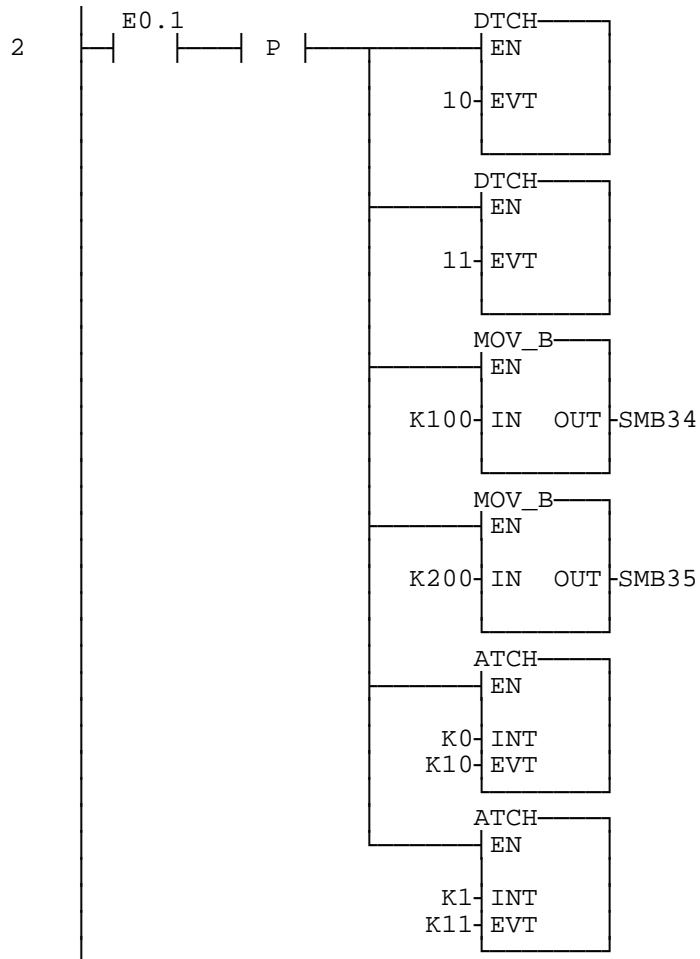
```
LD      SM0.1 // Обращается только
           // в первом цикле
MOV      50,SMB34
           // Задание растра времени
           // для прерывания по
           // времени 0: 50 мс
MOV      100,SMB35
           // Задание растра времени
           // для прерывания по
           // времени 1: 100 мс

ATCH     0,10 // Связь прерывания 10 с
           // подпрограммой
           // прерывания 0

ATCH     1,11 // Связь прерывания 10 с
           // подпрограммой
           // прерывания 1

ENI      // Деблокировка всех
           // прерываний
```

// По положительному фронту входа E0.1 растр времени прерывания по времени  
 // удваивается. Для выполнения этого нового задания необходимо  
 // разорвать связи между прерываниями и подпрограммами прерывания,  
 // так как в противном случае новые значения не будут восприняты. Разрыв  
 // связи производится командой DTCH.  
 // После задания нового растра времени, связи должны быть заново созданы  
 // командой ATCH.



LD E0.1 // Загрузка входа E0.1  
 EU // и разрыв связи с  
 DTCH 10 // прерыванием 0 по  
 // положительному фронту  
 // E0.1

DTCH 11 // Разрыв связи с  
 // прерыванием 1

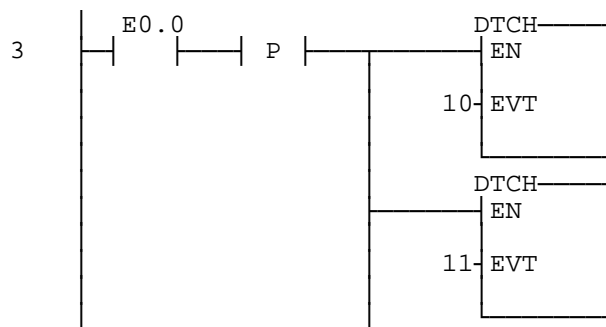
MOVB 100,SMB34  
 // Задание нового растра  
 // времени для  
 прерывания // по времени: 100 мс

MOVB 200,SMB35  
 // Задание нового растра  
 // времени для  
 прерывания // по времени: 200 мс

ATCH 0,10 // Восстановление  
 // связи

ATCH 1,11 // Восстановление  
 // связи

// По положительному фронту входа E0.0 будет восстановлена старая тактовая  
 // частота.



LD E0.0 // Положительный фронт  
 EU // ... фронта E0.0  
 DTCH 10 // разрывает связь с  
 // прерыванием по  
 времени // 10

DTCH 11 // Прерывание связи с  
 // прерыванием по  
 времени // 11

	<pre> MOV_B  50,SMB34       // Время интервала       // прерывания 0: 50 мс  MOV_B  100,SMB35       // Время интервала       // прерывания 1: 100 мс  ATCH   0,10  // Деблокировка 0      // прерывания по времени  ATCH   1,11  // Деблокировка 1      // прерывания по времени  MEND    // Конец         // основной программы </pre>
--	---

## Подпрограммы прерываний

// \*\*\*\*\*

// При вызове подпрограммы прерывания 0 устанавливается выход A0.0.

	<pre> INT    0      // Подпрограмма            // прерывания 0 LD     SM0.0 // устанавливает S      A0.0,1 // выход A0.0 в 1  RETI      // Конец подпрограммы            // прерывания 0 </pre>
--	---

// \*\*\*\*\*

// При вызове подпрограммы прерывания 1 выход A0.0 сбрасывается.

// Так как растр времени для вызова подпрограммы прерывания 1 в два раза длиннее, чем

// растр времени для вызова подпрограммы прерывания 0, возникает такт мигания.

	<pre> INT    1      // Подпрограмма            // прерывания 1 LD     SM0.0 // сбрасывает R      A0.0,1 // выход A0.0  RETI      // Конец подпрограммы            // прерывания 0 </pre>
--	--

**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 станет K4)
- Замените '16#' на '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.

**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.