

# Организационные блоки

# 1

## Обзор главы

В раз-деле	Вы найдете	на стр.
1.1	Обзор ОВ	1–2
1.2	Циклическая программа (ОВ 1)	1–4
1.3	ОВ прерываний по времени (ОВ 10 - ОВ 17)	1–6
1.4	ОВ прерываний с задержкой (ОВ 20 - ОВ 23)	1–9
1.5	ОВ циклических прерываний (ОВ 30 - ОВ 38)	1–11
1.6	ОВ прерываний от процесса (ОВ 40 - ОВ 47)	1–13
1.7	ОВ мультипроцессорных прерываний (ОВ 60)	1–15
1.8	ОВ ошибок времени (ОВ 80)	1–17
1.9	ОВ неисправностей источника питания (ОВ 81)	1–20
1.10	ОВ диагностических прерываний (ОВ 82)	1–22
1.11	ОВ установки/снятия модулей (ОВ 83)	1–24
1.12	ОВ аппаратных ошибок CPU (ОВ 84)	1–26
1.13	ОВ ошибок исполнения программы (ОВ 85)	1–27
1.14	ОВ неисправностей носителей модулей (ОВ 86)	1–30
1.15	ОВ коммуникационных ошибок (ОВ 87)	1–34
1.16	ОВ фонового режима (ОВ 90)	1–36
1.17	ОВ запуска (ОВ 100 и ОВ 101)	1–38
1.18	ОВ ошибок программирования (ОВ 121)	1–41
1.19	ОВ ошибок доступа к периферии (ОВ 122)	1–44

## 1.1. Обзор ОВ

### Организационные блоки (ОВ)

Организационные блоки образуют интерфейс между операционной системой CPU и прикладной программой. С помощью ОВ можно обеспечивать исполнение программ при:

- при старте CPU
- при циклическом и тактированном по времени исполнении
- в определенное время или день
- по истечении заданного интервала времени
- при возникновении ошибок
- при возникновении прерываний по сигналам от процесса.

Организационные блоки обрабатываются в соответствии с присвоенными им приоритетами.

### Применимость ОВ

Не все CPU могут обрабатывать полный комплект ОВ, имеющийся в S7. Необходимо иметь перечень ОВ, которые могут использоваться конкретным типом CPU.

### Дополнительные сведения

Дополнительная информация может быть получена как из online-помощи, так и из следующих руководств:

- **/70/**: Это руководство содержит справочные данные о функциональных возможностях различных CPU S7–300. Там же описаны возможные стартовые события для каждого ОВ.
- **/101/**: Это руководство содержит справочные данные о функциональных возможностях различных CPU S7–400. Там же описаны возможные стартовые события для каждого ОВ.

Таблица 1–1 содержит соответствующее стартовое событие и предустановленный класс приоритета для каждого ОВ.

Таблица 1-1 Обзор ОВ

ОВ	Стартовое событие	Предустановленный приоритет	Примечания
ОВ 1	Конец запуска или конец ОВ 1	1	Свободный цикл
ОВ 10	Прерывание по времени 0	2	
ОВ 11	Прерывание по времени 1	2	
ОВ 12	Прерывание по времени 2	2	
ОВ 13	Прерывание по времени 3	2	
ОВ 14	Прерывание по времени 4	2	
ОВ 15	Прерывание по времени 5	2	
ОВ 16	Прерывание по времени 6	2	
ОВ 17	Прерывание по времени 7	2	
ОВ 20	Прерывание с задержкой 0	3	
ОВ 21	Прерывание с задержкой 1	4	
ОВ 22	Прерывание с задержкой 2	5	
ОВ 23	Прерывание с задержкой 3	6	
ОВ 30	Циклическое прерывание 0 (по умолчанию : такт 5с)	7	
ОВ 31	Циклическое прерывание 1 (по умолчанию : такт 2с)	8	
ОВ 32	Циклическое прерывание 2	9	

ОВ 33	(по умолчанию : такт 1с) Циклическое прерывание 3	10	
ОВ 34	(по умолчанию : такт 500 мс) Циклическое прерывание 4	11	
ОВ 35	(по умолчанию : такт 200 мс) Циклическое прерывание 5	12	
ОВ 36	(по умолчанию : такт 100 мс) Циклическое прерывание 6	13	
ОВ 37	(по умолчанию : такт 50 мс) Циклическое прерывание 7	14	
ОВ 38	(по умолчанию : такт 20 мс) Циклическое прерывание 8	15	
ОВ 38	(по умолчанию : такт 10 мс)		
ОВ 40		16	
ОВ 41		17	
ОВ 42		18	
ОВ 43		19	
ОВ 44		20	
ОВ 45		21	
ОВ 46		22	
ОВ 47		23	
ОВ 60	Вызов SFC 35 "MP_ALM"	25	Прерывание мультимикропроцессор- ной обработки
ОВ 80	Ошибка времени		
ОВ 81	Неисправность источника питания		
ОВ 82	Диагностическое прерывание		
ОВ 83	Прерывание установки/снятия модуля		
ОВ 84	Аппаратная ошибка CPU		
ОВ 85	Ошибка исполнения программы		
ОВ 86	Выход из строя носителя модулей, подсети или станции у децентрализованной периферии		
ОВ 87	Коммуникационные ошибки		
ОВ 90	Новый пуск или стирание блока, обрабатываемого ОВ 90, или загрузка ОВ 90 в CPU или окончание ОВ 90	29 2)	Фоновый цикл
ОВ 100	Новый пуск	27 1)	
ОВ 101	Повторный пуск	27 1)	
ОВ 121	Ошибка программирования	Приоритет ОВ, вызвавшего ошибку	
ОВ 122	Ошибка доступа к периферии	Приоритет ОВ, вызвавшего ошибку	

1) Классы приоритетов 27 и 28 действительны в модели приоритетов запуска.

2) Класс приоритета 29 соответствует приоритету 0.29. Фоновый цикл, таким образом, имеет более низкий приоритет, чем свободный цикл.

## 1.2. Циклическая программа (OB 1)

### Описание

Операционная система S7 CPU обрабатывает OB 1 циклически: как только заканчивается обработка OB 1, операционная система начинает его обработку вновь. Циклическая обработка OB начинается по окончании стадии запуска. Существует возможность вызывать в OB 1 функциональные блоки (FB, SFB) или функции (FC, SFC).

### Принцип действия OB 1

OB 1 имеет самый низкий приоритет всех среди контролируемых по времени исполнения OB. За исключением OB 90, все остальные OB могут прерывать выполнение OB 1. Операционная система вызывает OB 1 при следующих событиях:

- Конец обработки запуска
- Конец обработки OB 1 (предыдущего цикла).

Если обработка OB 1 закончена, то операционная система записывает отображение процесса на выходах в модули вывода и посылает глобальные данные. Прежде чем OB 1 будет вновь запущен, операционная система обновляет отображение процесса на входах и получает глобальные данные для CPU.

S7 осуществляет контроль максимальной длительности цикла, чем гарантируется максимальное время реакции. Значение максимальной длительности предусмотрено в 150 мс; Вы можете его перепараметризовать или же при помощи SFC 43 "RE\_TRIGR" в любой точке Вашей программы запустить заново контроль времени. Если время выполнения программы превышает максимальное время цикла OB 1, то операционная система вызывает OB 80 (OB ошибок времени). Если OB 80 не запрограммирован, то CPU переводится в состояние STOP.

Одновременно с контролем максимального времени цикла может быть гарантировано минимальное время цикла. Операционная система задерживает следующий старт OB 1 так долго, пока не будет обеспечено минимальное время цикла.

Диапазоны параметров задания минимальной и максимальной длительности цикла приведены в руководствах /70/ и /101/. Изменение параметров осуществляется средствами STEP 7.

### Локальные данные OB1

Таблица 1–2 содержит временные переменные (TEMP) OB 1. В качестве имен переменных выбраны имена OB 1 по умолчанию.

Таблица 1-2 Локальные переменные (TEMP) OB 1

Переменные	Тип данных	Описание
OB1_EV_CLASS	BYTE	Класс событий и идентификаторы: V#16#11: активен
OB1_SCAN_1	BYTE	V#16#01: Окончание нового пуска V#16#02: Окончание повторного пуска V#16#03: Окончание свободного цикла
OB1_PRIORITY	BYTE	Класс приоритета: 1
OB1_OB_NUMBR	BYTE	№ OB (01)
OB1_RESERVED_1	BYTE	Резерв
OB1_RESERVED_2	BYTE	Резерв

Таблица 1-2 Локальные переменные (TEMP) OB 1 (продолжение)

Переменные	Тип данных	Описание
OB1_PREV_CYCLE	INT	Длительность предыдущего цикла (мс)
OB1_MIN_CYCLE	INT	Минимальная длительность цикла (мс) со времени последнего запуска
OB1_MAX_CYCLE	INT	Максимальная длительность цикла (мс) со времени последнего запуска
OB1_DATE_TIME	DATE_AND_TIME	Дата и время, когда был запрошен OB

### 1.3. ОВ прерываний по времени (ОВ 10 - ОВ 17)

- Описание** S7 предоставляет в распоряжение до восьми прерываний по времени (ОВ 10 - ОВ 17), которые могут запускаться однократно или периодически. Вы можете так параметрировать Ваше CPU при помощи SFC или STEP 7, что эти ОВ будут обрабатываться со следующими интервалами:
- Однократно
  - Ежеминутно
  - Ежечасно
  - Ежедневно
  - Еженедельно
  - Ежемесячно
  - Ежегодно
- Принцип действия прерываний по времени** Чтобы запустить прерывание по времени, его необходимо вначале **ОВ** установить, а потом активизировать. Существует три следующих способа **по** старта:
- Автоматический старт прерывания по времени. Он происходит, если Вы как устанавливаете, так и активизируете прерывания по времени при помощи STEP 7. Таблица 1–3 показывает принципиальные возможности при активизации прерываний по времени посредством STEP 7.
  - Вы устанавливаете прерывания по времени при помощи STEP 7 и активизируете их вызовом SFC 30 "ACT\_TINT" из Вашей программы.
  - Вы устанавливаете прерывание по времени вызовом SFC 28 "SET\_TINT" и активизируете его вызовом SFC 30 "ACT\_TINT".

Таблица 1-3. Принципиальные возможности активизации прерываний по времени при помощи STEP 7

Интервал	Описание
Не активизируется	ОВ прерывания по времени не обрабатывается, даже если он загружен в CPU. Вы можете активизировать прерывание по времени вызовом SFC 30.
Однократно активизируется	ОВ прерывания по времени автоматически отменяется после того, как он был однократно обработан. Ваша программа может с помощью SFC 28 заново установить прерывание по времени и вновь его активизировать через SFC 30.
Периодически активизируется	CPU определяет для момента прерывания по времени момент следующего старта прерывания по времени исходя из текущего времени и периода .

Поведение прерываний по времени при переводе часов вперед или назад описаны в /234/.

#### Замечание

Если Вы конфигурируете прерывание по времени на однократную обработку ОВ, то заданные значения дат и времени не могут находиться в прошлом по отношению к показанию реального времени CPU.  
Если Вы прерывание по времени конфигурируете так, что соответствующий ОВ будет обрабатываться периодически, а дата и время лежат в прошлом, то обработка осуществится в следующем моменте, соответствующем текущему времени. Пример такого варианта приведен на рисунке 1-1.

С помощью SFC 39 - 42 Вы можете прерывание по времени запретить или отложить и вновь разрешить. Более подробная информация содержится в главе 11.

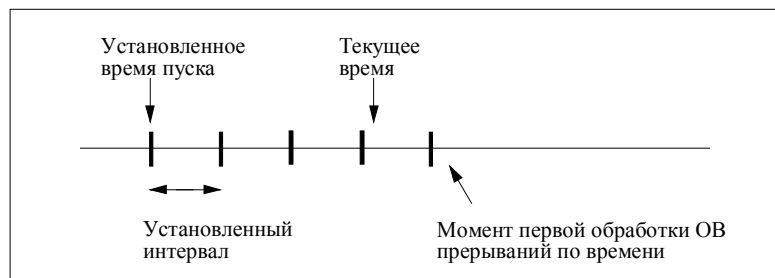


Рис. 1-1. Первая обработка ОВ прерываний по времени для случая, когда время старта находится в прошлом и была установлена периодическая активизация

### Условия, нарушающие работу ОВ прерываний по времени

Т.к. прерывание по времени наступает через определенные интервалы времени, то определенные условия могут нарушать работу соответствующего ОВ во время обработки Вашей программы. Таблица 1–4 представляет некоторые из таких условий и описывает их воздействие на обработку ОВ прерываний по времени.

Таблица 1-4. Условия, которые нарушают функционирование ОВ прерываний по времени

Условие	Результат
Ваша программа вызывает SFC29 "CAN_TINT" и отменяет прерывание.	Операционная система стирает стартовое событие (дату и время) прерывания по времени. Вам необходимо вновь установить стартовое событие и произвести активацию, если ОВ должен быть вновь вызван.
Ваша программа пыталась активизировать ОВ прерывания по времени, который ко времени активации не был загружен в CPU.	Операционная система вызывает ОВ 85. Если ОВ 85 не был запрограммирован (загружен в CPU), то CPU переводится в состояние STOP.
При синхронизации или коррекции системных часов CPU Вы переставили время вперед и перескочили стартовое событие, т.е. дату или время для ОВ прерывания по времени.	Операционная система вызывает ОВ 80 и кодирует номер ОВ прерывания по времени и информацию стартового события в ОВ 80. Операционная система обрабатывает затем один раз ОВ прерывания по времени, независимо от того, как часто этот ОВ собственно должен был бы обрабатываться. Информация о стартовом событии ОВ 80 показывает дату и время, когда ОВ прерывания по времени в первый раз был "перепрыгнут".
При синхронизации или коррекции системных часов CPU Вы перевели время назад так, что стартовое событие, т.е. дата или время для ОВ прерывания по времени повторяется.	Если ОВ прерывания по времени был активизирован до того, как часы были переведены назад, то он не будет вновь вызываться для уже прошедших интервалов времени.
CPU выполняет новый пуск.	Каждый ОВ прерывания по времени, который был сконфигурирован через SFC, вновь воспринимает конфигурацию, которая была запрограммирована при помощи STEP 7.
ОВ прерывания по времени еще обрабатывается, когда происходит стартовое событие для следующего интервала.	Операционная система вызывает ОВ 80. Если ОВ 80 не был запрограммирован, то CPU переходит в состояние STOP. В противном случае после обработки ОВ 80 и ОВ прерывания по времени осуществляется обработка затребованного ОВ.

**Локальные данные  
ОВ прерываний  
по времени**

Таблица 1–5 содержит временные переменные (TEMP) одного ОВ прерывания по времени. В качестве имен переменных выбраны имена по умолчанию ОВ 10.

Таблица 1-5. Локальные переменные (TEMP) ОВ прерывания по времени на примере ОВ 10

Переменная	Тип данных	Описание
OB10_EV_CLASS	BYTE	Класс события и идентификаторы: B#16#11: прерывание активно
OB10_STRT_INFO	BYTE	B#16#11: Запрос на запуск для ОВ 10 (B#16#12: Запрос на запуск для ОВ 11) : : (B#16#18: Запрос на запуск для ОВ 17)
OB10_PRIORITY	BYTE	Класс приоритета: 2
OB10_OB_NUMBR	BYTE	Номер ОВ (10 - 17)
OB10_RESERVED_1	BYTE	Резерв
OB10_RESERVED_2	BYTE	Резерв
OB10_PERIOD_EXE	WORD	ОВ обрабатывается с заданными интервалами: W#16#0000: Однократно W#16#0201: Ежеминутно W#16#0401: Ежечасно W#16#1001: Ежедневно W#16#1201: Еженедельно W#16#1401: Ежемесячно W#16#1801: Ежегодно
OB10_RESERVED_3	INT	Резерв
OB10_RESERVED_4	INT	Резерв
OB10_DATE_TIME	DATE_AND_TIME	Дата и время, когда был затребован ОВ.



## 1.4. ОВ прерываний с задержкой (ОВ 20 - ОВ 23)

### Описание

S7 предоставляет в распоряжение до четырех ОВ (ОВ 20 - ОВ 23), которые по мере надобности обрабатываются через параметрируемое время задержки. Каждый ОВ прерывания с задержкой запускается посредством вызова SFC32 "SRT\_DINT". Время задержки является входным параметром SFC.

Если Ваша программа вызывает функцию SFC32 "SRT\_DINT", то ей передается номер ОВ, время задержки и индивидуальный код пользователя. Через заданное время задержки стартует соответствующий ОВ. Имеется возможность отменить обработку прерывания с задержкой, которое еще не стартовало.

### Принцип действия ОВ прерываний с задержкой

По истечении времени, значение которого в мс Вы вместе с номером ОВ передали в SFC32, операционная система запускает соответствующий ОВ. Если Вы желаете использовать ОВ прерываний с задержкой, то Вы должны выполнить следующие требования:

- Вы должны вызвать SFC32 "SRT\_DINT".
- Вы должны загрузить ОВ прерываний с задержкой в CPU как часть Вашей программы.

Прерывания с задержкой обрабатываются только в том случае, когда CPU находится в режиме RUN. Новый запуск CPU стирает любое стартовое событие ОВ прерывания с задержкой. Прерывание с задержкой, которое еще не активизировано, можно отменить при помощи SFC33 "CAN\_DINT".

Время задержки измеряется с точностью 1 мс. Время задержки может быть повторно запущено непосредственно после его окончания. С помощью SFC34 "QRY\_DINT" Вы можете определить состояние прерывания с задержкой.

Операционная система вызывает ОВ для асинхронных ошибок в случаях, когда наступает одно из следующих событий:

- Когда операционная система пытается запустить ОВ, который не загружен, а его номер Вы задали при вызове SFC 32 "SRT\_DINT".
- Когда наступает следующее стартовое событие для запуска прерывания с задержкой прежде, чем закончилась обработка текущего ОВ прерывания с задержкой.

С помощью SFC 39 - 42 Вы можете прерывания с задержкой запретить или отложить и вновь разрешить. Более подробная информация содержится в главе 11.

**Локальные данные  
ОВ прерываний  
с задержкой**

Таблица 1–6 содержит временные переменные (TEMP) ОВ прерывания с задержкой. В качестве имен переменных взяты по умолчанию имена ОВ 20.

Таблица 1-6. Локальные переменные (TEMP) ОВ прерывания с задержкой на примере ОВ 20

Переменная	Тип данных	Объявление	Описание
OB20_EV_CLASS	BYTE	TEMP	Класс события и идентификаторы: В#16#11: Прерывание активно
OB20_STRT_INF	BYTE	TEMP	В#16#21: Требование запуска для ОВ 20 (В#16#22: Требование запуска для ОВ 21) (В#16#23: Требование запуска для ОВ 22) (В#16#24: Требование запуска для ОВ 23)
OB20_PRIORITY	BYTE	TEMP	Класс приоритета: 3 (ОВ 20) до 6 (ОВ 23) (значения по умолчанию)
OB20_OB_NUMBR	BYTE	TEMP	Номер ОВ (20 - 23)
OB20_RESERVED_1	BYTE	TEMP	Резерв
OB20_RESERVED_2	BYTE	TEMP	Резерв
OB20_SIGN	WORD	TEMP	Код пользователя: входной параметр SIGN из вызова SFC32 "SRT_DINT"
OB20_DTIME	TIME	TEMP	Истекшая часть времени задержки в мс
OB20_DATE_TIME	DATE_AND_TIME	TEMP	Дата и время, когда был затребован ОВ

## 1.5. ОВ циклических прерываний (ОВ 30 - ОВ 38)

### Описание

S7 представляет в распоряжение до девяти ОВ циклических прерываний (ОВ 30 - ОВ 38). С их помощью можно запускать программу через равные промежутки времени. Таблица 1-7 приводит сведения о предустановленных значениях временного раstra и классов приоритета ОВ циклических прерываний.

Таблица 1-7. Предустановленные значения для временного раstra и классов приоритета каждого ОВ циклических прерываний.

ОВ циклических прерываний	Значение по умолчанию для временного раstra	Значение по умолчанию для класса приоритетов
ОВ 30	5 с	7
ОВ 31	2 с	8
ОВ 32	1 с	9
ОВ 33	500 мс	10
ОВ 34	200 мс	11
ОВ 35	100 мс	12
ОВ 36	50 мс	13
ОВ 37	20 мс	14
ОВ 38	10 мс	15

### Принцип действия ОВ циклических прерываний

Эквидистантные моменты запуска ОВ циклических прерываний получаются из соответствующей периодичности и фазового сдвига. Как взаимозависят момент запуска, периодичность и фазовый сдвиг, описано в /234/.

---

#### Замечание

Вы должны позаботиться о том, чтобы время работы ОВ циклических прерываний было заметно меньше интервала времени, через который он вызывается. В случае, когда ОВ циклических прерываний еще не закончен, а в следствие окончания интервала времени должен обрабатываться вновь, запускается ОВ ошибок времени (ОВ 80). Затем вызвавшее ошибку циклическое прерывание будет отработано.

---

С помощью SFC 39 - 42 Вы можете циклическое прерывание запретить или отложить и вновь разрешить. Более подробная информация содержится в главе 13.

Области значений для периодичности (Zeittakt), класса приоритетов и фазовых сдвигов приводятся в технических данных CPU. Изменения параметров производится параметризацией с помощью STEP 7.

**Локальные данные  
ОВ циклических  
прерываний**

Таблица 1–8 содержит временные (TEMP) переменные одного из ОВ циклических прерываний. В качестве имен переменных по умолчанию взяты имена ОВ 35.

Таблица 1-8. Локальные переменные (TEMP) ОВ циклических прерываний на примере ОВ 35

Переменные	Тип данных	Описание
OB35_EV_CLASS	BYTE	Класс события и идентификаторы: B#16#11: Прерывание активно
OB35_STRT_INF	BYTE	(B#16#31 : Требование запуска для ОВ 30) : B#16#36 : Требование запуска для ОВ 35 : (B#16#39 : Требование запуска для ОВ 38)
OB35_PRIORITY	BYTE	Класс приоритета: от 7 (ОВ 30) до 15 (ОВ 38) (значения по умолчанию)
OB35_OB_NUMBR	BYTE	Номер ОВ (30 - 38)
OB35_RESERVED_1	BYTE	Резерв
OB35_RESERVED_2	BYTE	Резерв
OB35_PHASE_OFFSET	WORD	Фазовый сдвиг в мс
OB35_RESERVED_3	INT	Резерв
OB35_EXC_FREQ	INT	Периодичность в мс
OB35_DATE_TIME	DATE_AND_TIME	Дата и время, когда был затребован ОВ.

## 1.6 ОВ прерываний по сигналам процесса (ОВ 40 - ОВ 47)

### Описание

S7 представляет в распоряжение до восьми независимых друг от друга прерываний по сигналам процесса по одному ОВ на каждое прерывание.

При помощи STEP 7 Вы задаете для каждого сигнального модуля, который должен вызывать прерывание по сигналам процесса:

- какие каналы при каких граничных условиях должны вызывать прерывание по сигналам процесса
- какой ОВ прерываний по сигналам процесса ставится в соответствие отдельным группам каналов (по умолчанию все прерывания по сигналам процесса обрабатываются через ОВ 40).

У СР и FM Вы должны использовать для этого соответствующее программное обеспечение.

Классы приоритета для отдельных ОВ прерываний по сигналам процесса устанавливаются посредством STEP 7.

### Принцип действия ОВ прерываний по сигналам процесса

После вызова модулем прерывания по сигналам процесса операционная система идентифицирует слот и выясняет соответствующий ОВ прерывания. Если он имеет более высокий приоритет, чем активный в данный момент класс приоритета, то он запускается. Специфическое для канала квитирование осуществляется по окончании этого ОВ прерывания по сигналам процесса.

Если в промежуток времени между идентификацией и квитированием прерывания по сигналам процесса на том же самом модуле возникает событие, которое вызывает прерывание по сигналам процесса, то:

- Если наступит событие на том же самом канале, который до этого вызвал прерывание по сигналам процесса, то соответствующее прерывание теряется. Такая ситуация изображена на рисунке 1-2 на примере канала модуля дискретного ввода. Пусть вызывающим событием является нарастающий фронт. И пусть соответствующим ОВ прерываний по сигналам процесса является ОВ 40.



Рис. 1-2 Взаимосвязь между сигналом от процесса и обработкой соответствующих прерываний

- Если событие происходит на другом канале того же самого модуля, то оно не может вызвать немедленного прерывания, однако это прерывание не теряется, а будет выполняться после квитирования текущего активного прерывания (только для S7-400). В S7-300 прерывание по сигналам процесса теряется, если вызывающее его событие закончилось раньше квитирования.

Если затребовано прерывание по сигналам процесса, ОВ которого в настоящий момент активизирован из другого модуля, то новый вызов прерывания регистрируется и ОВ будет обработан в заданное время (только для S7–400). Для S7–300 прерывание по сигналу процесса окажется потерянным, если событие, вызывающее прерывание, не сохранилось после квитирования.

С помощью SFC 39 - 42 Вы можете прерывания по сигналам процесса запретить или отложить и вновь разрешить. Более подробная по этому вопросу информация содержится в главе 11.

Вы можете осуществлять параметрирование прерываний по сигналам процесса не только при помощи STEP 7, но и при помощи SFC 55 - 57. Более подробная информация по этому вопросу содержится в главе 11.

**Локальные данные** Таблица 1–9 описывает временные (TEMP) переменные одного ОВ  
**ОВ прерываний** прерывания по сигналам процесса. В качестве имен переменных  
**по сигналам** выбраны имена по умолчанию ОВ 40.  
**процесса**

Таблица 1-9. Локальные переменные (TEMP) прерывания по сигналам процесса на примере ОВ 40

Переменная	Тип данных	Описание
OB40_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11: Прерывание активно
OB40_STRT_INF	BYTE	В#16#41: прерывание через управление прерываниями 1 В#16#42: прерывание через управление прерываниями 2 (только для S7–400) В#16#43: прерывание через управление прерываниями 3 (только для S7–400) В#16#44: прерывание через управление прерываниями 4 (только для S7–400)
OB40_PRIORITY	BYTE	Класс приоритета: 16 (ОВ 40) до 23 (ОВ 47) (по умолчанию)
OB40_OB_NUMBR	BYTE	Номер ОВ (40 - 47)
OB40_RESERVED_1	BYTE	Резерв
OB40_IO_FLAG	BYTE	Модуль ввода: В#16#54 Модуль вывода: В#16#55
OB40_MDL_ADDR	WORD	Логический базовый адрес модуля, который вызвал прерывание
OB40_POINT_ADDR	DWORD	Для цифровых модулей: битовый массив с состоянием входов на модуле Для аналоговых модулей, CP или IM: состояние прерываний модуля
OB40_DATE_TIME	DATE_AND_TIME	Дата и время, когда был ОВ затребован

## 1.7. ОВ мультипроцессорных прерываний (ОВ 60)

### Описание

С помощью мультипроцессорного прерывания Вы можете при мультипроцессорной обработке на соответствующих CPU обеспечить синхронную реакцию соответствующих CPU на событие. В противоположность прерываниям от сигналов процесса, которые вызываются сигнальными модулями, мультипроцессорное прерывание может выдаваться исключительно CPU.

### Принцип действия ОВ мультипроцессорных прерываний

Мультипроцессорное прерывание инициируется вызовом SFC 35 "MP\_ALM". При мультипроцессорной обработке это приводит к синхронизированному старту ОВ 60 на всех установленных CPU данного шинного сегмента, если Вы не заблокировали ОВ 60 (посредством SFC 39 "DIS\_IRT") или не задержали (посредством SFC 41 "DIS\_AIRT"). В случае, если Вы не загрузили ОВ 60 в CPU, то соответствующий CPU немедленно возвращается в последний обработанный класс приоритетов и продолжает там обработку программы. При однопроцессорной работе и при работе с сегментированным носителем модулей ОВ 60 запускается только на том CPU, на котором Вы вызвали SFC 35 "MP\_ALM".

Если Ваша программа вызывает SFC 35 "MP\_ALM", то Вы передаете ей выбранный Вами идентификатор задания. Этот идентификатор передается на все CPU. Таким образом, у Вас появляется возможность реагировать в зависимости от имеющегося события. Если ОВ 60 на отдельных CPU запрограммирован по-разному, то время его обработки может оказаться различным. В этом случае прерванный при необходимости класс приоритета продолжает обрабатываться для различных моментов времени. Если следующее мультипроцессорное прерывание выдается одним CPU в то время, как другой еще занят обработкой ОВ 60 предыдущего мультипроцессорного прерывания, то запуска ОВ 60 не происходит ни на затребованном, ни на каком другом установленном CPU шинного сегмента. Такое положение вещей, изображенное в качестве примера на рис. 1-3 для двух CPU, сообщается Вам через соответствующее значение вызванной SFC 35.

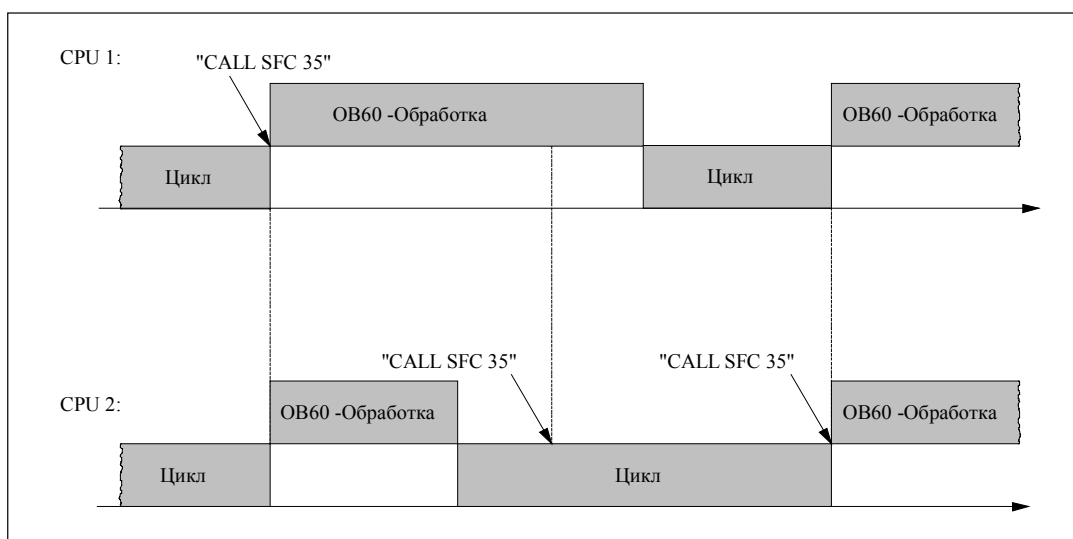


Рис. 1-3. Пример потери мультипроцессорного прерывания

**Локальные данные  
ОВ мульти-  
процессорных  
прерываний**

Таблица 1-10 содержит временные (TEMP) переменные ОВ мультипроцессорных прерываний. В качестве имен переменных взяты имена по умолчанию ОВ 60.

Таблица 1-10. Локальные переменные (TEMP) ОВ мультипроцессорных прерываний

Переменные	Тип данных	Описание
ОВ60_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11: Прерывание активно
ОВ60_STRT_INF	BYTE	В#16#61: Мультипроцессорное прерывание вызвано собственным CPU В#16#62: Мультипроцессорное прерывание вызвано другим CPU
ОВ60_PRIORITY	BYTE	Класс приоритета: 25
ОВ60_OB_NUMBR	BYTE	Номер ОВ: 60
ОВ60_RESERVED_1	BYTE	Резерв
ОВ60_RESERVED_2	BYTE	Резерв
ОВ60_JOB	INT	Идентификатор задания: Входная переменная JOB SFC 35 "MP_ALM"
ОВ60_RESERVED_3	INT	Резерв
ОВ60_RESERVED_4	INT	Резерв
ОВ60_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван ОВ.



## 1.8. ОВ ошибок времени (ОВ 80)

### Описание

Операционная система CPU вызывает ОВ 80 в тех случаях, когда при обработке какого-либо ОВ возникает одна из следующих ошибок: превышение времени цикла, ошибка квитирования при обработке ОВ, перевод часов вперед (скачок времени) по отношению к запуску какого-либо ОВ. Если, например, стартовое событие для ОВ циклических прерываний возникает до того, как была закончена обработка предыдущего ОВ, то операционная система вызывает ОВ 80.

Если ОВ 80 не был запрограммирован, то CPU переходит в рабочий режим STOP.

С помощью SFC 39 - 42 Вы можете ОВ циклических прерываний запретить или отложить и вновь разрешить. Более полная информация содержится в главе 11.

---

### Замечание

Если ОВ 80 в одном и том же цикле будет вызван два раза из-за превышения времени цикла, то CPU переходит в режим STOP. Вы можете этому препятствовать вызовом SFC43 "RE\_TRIGR" в соответствующем месте.

---

### Локальные данные

Таблица 1-11 содержит временные (TEMP) переменные ОВ ошибок  
**ОВ ошибок времени** времени. В качестве имен переменных взяты имена по умолчанию ОВ 80.

Таблица 1-11 . Локальные переменные (TEMP) OB 80

Переменные	Тип данных	Описание
OB80_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#35
OB80_FLT_ID	BYTE	Коды ошибок (возможные значения: В#16#01, В#16#02, В#16#05, В#16#06, В#16#07)
OB80_PRIORITY	BYTE	Класс приоритета: 26
OB80_OB_NUMBR	BYTE	Номер OB (80)
OB80_RESERVED_1	BYTE	Резерв
OB80_RESERVED_2	BYTE	Резерв
OB80_ERROR_INFO	WORD	Информация об ошибках: зависит от кода ошибки
OB80_ERR_EV_CLASS	BYTE	Класс события, которое эту ошибку вызвало
OB80_ERR_EV_NUM	BYTE	Номер события, которое эту ошибку вызвало
OB80_OB_PRIORITY	BYTE	Класс приоритета OB, который обрабатывался, когда произошла ошибка
OB80_OB_NUM	BYTE	Номер OB, который обрабатывался , когда произошла ошибка
OB80_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB

Переменные, зависящие от кода ошибок, имеют следующие значения:

- Код ошибки B#16#01 Превышено время цикла
  - OB80\_ERROR\_INFO: Длительность последнего цикла (мс)
  - OB80\_ERR\_EV\_CLASS: Класс события, которое вызвало прерывание
  - OB80\_ERR\_EV\_NUM: Номер события, которое вызвало прерывание
  - OB80\_OB\_PRIORITY: Класс приоритета, активный на момент прерывания
  - OB80\_OB\_NUM: ОВ, активный на момент прерывания
- Коды ошибок B#16#02 Затребованный ОВ еще обрабатывается.  
B#16#07 Переполнение буфера вызовов ОВ для текущего класса приоритета (каждое требование старта ОВ для одного класса приоритета заносится в соответствующий буфер вызовов ОВ; по завершении ОВ запись стирается. Если же для одного класса приоритета количество вызовов ОВ превысило максимально возможное число записей соответствующего буфера, то вызывается ОВ 80 с кодом ошибки B#16#07.)
  - OB80\_ERROR\_INFO: Как соответствующее значение в прерванном программном уровне
  - OB80\_ERR\_EV\_CLASS: Класс события, которое вызвало прерывание
  - OB80\_ERR\_EV\_NUM: Номер события, которое вызвало прерывание
  - OB80\_OB\_PRIORITY: Класс приоритета, активный на момент прерывания
  - OB80\_OB\_NUM: ОВ, активный на момент прерывания
- Коды ошибок B#16#05 Прошедшее прерывание по времени вследствие скачка времени  
B#16#06 Прошедшее прерывание по времени после возвращения в RUN из HALT
  - OB80\_ERROR\_INFO: Бит 0 установлен: для прерывания по времени №0 стартовая точка лежит в прошлом  
:  
:  
Бит 7 установлен: для прерывания по времени №7 стартовая точка лежит в прошлом  
Биты 8 - 15: не используются
  - OB80\_ERR\_EV\_CLASS: не используется
  - OB80\_ERR\_EV\_NUM: не используется
  - OB80\_OB\_PRIORITY: не используется
  - OB80\_OB\_NUM: не используется

## 1.9. ОВ неисправностей источника питания (ОВ 81)

**Описание**                      Операционная система CPU вызывает ОВ 81 тогда, когда происходит событие, вызванное неисправностью источника питания (только для S7-400) или буферизации (как при наступающем, и при текущем событии).  
В отличие от ОВ для других асинхронных ошибок CPU в данном случае не переходит в режим STOP, если ОВ 81 не был запрограммирован.  
С помощью SFC 39 - 42 Вы можете ОВ неисправностей источника питания запретить или отложить и вновь разрешить. Более подробная информация содержится в главе 11.

**Локальные данные**        Таблица 1-12 содержит временные (TEMP) переменные ОВ неисправностей

**ОВ неисправностей**        источника питания. В качестве имен переменных взяты имена по

**источника питания**        умолчанию ОВ 81.

Таблица 1-12 . Локальные переменные (TEMP) ОВ 81

Переменная	Тип данных	Описание
OB81_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#38: текущее событие В#16#39: наступающее событие
OB81_FLT_ID	BYTE	Коды ошибок (возможные значения: В#16#21, В#16#22, В#16#23, В#16#31, В#16#32, В#16#33)
OB81_PRIORITY	BYTE	Класс приоритета: 26 (по умолчанию для рабочего состояния RUN) или 28 (рабочее состояние ANLAUF)
OB81_OB_NUMBR	BYTE	Номер ОВ (81)
OB81_RESERVED_1	BYTE	Резерв
OB81_RESERVED_2	BYTE	Резерв
OB81_MDL_ADDR	INT	Резерв
OB81_RESERVED_3	BYTE	Относится только к кодам ошибок: В#16#31, В#16#32, В#16#33.
OB81_RESERVED_4	BYTE	
OB81_RESERVED_5	BYTE	
OB81_RESERVED_6	BYTE	
OB81_DATE_TIME	DATE_AND_TIME	Дата и время, когда ОВ был затребован

Переменные OB81\_RESERVED\_i,  $3 \leq i \leq 6$ , содержат те носители модулей, у которых батарейная буферизация (для кода ошибки В#16#31), буферное напряжение (для кода ошибки В#16#32) или питание 24В (для кода ошибки В#16#33) вышли из строя или восстановились. Следующая таблица показывает, какой бит в переменных OB81\_RESERVED\_i,  $3 \leq i \leq 6$ , к какому носителю модулей относится.

Таблица 1-13

	OB81_RESERVED_6	OB81_RESERVED_5	OB81_RESERVED_4	OB81_RESERVED_3
Бит 0	Резерв	8-й носитель расшир.	16-й носитель расшир.	Резерв
Бит 1	1-й носитель расшир.	9-й носитель расшир.	17-й носитель расшир.	Резерв
Бит 2	2-й носитель расшир.	10-й носитель расшир.	18-й носитель расшир.	Резерв
Бит 3	3-й носитель расшир.	11-й носитель расшир.	19-й носитель расшир.	Резерв
Бит 4	4-й носитель расшир.	12-й носитель расшир.	20-й носитель расшир.	Резерв
Бит 5	5-й носитель расшир.	13-й носитель расшир.	21-й носитель расшир.	Резерв
Бит 6	6-й носитель расшир.	14-й носитель расшир.	Резерв	Резерв
Бит 7	7-й носитель расшир.	15-й носитель расшир.	Резерв	Резерв

Биты в переменной OB81\_RESERVED\_i имеют следующее значение (для соответствующих носителей модулей):

При наступающем событии отмечаются носители модулей (устанавливаются соответствующие биты), у которых вышли из строя хотя бы одна батарея, буферное питание или питание 24В. Носители модулей, в которых до этого уже хотя бы одно питание вышло из строя, повторно не индицируются. При текущем событии индицируется восстановленная буферизация хотя бы на одном носителе модулей (устанавливаются соответствующие биты).

Переменная OB81\_FLT\_ID имеет следующее значение:

- В#16#21: Хотя бы одна буферная батарея центрального устройства не установлена / устранено (BATTF)
- В#16#22: Буферное напряжение в центральном устройстве отсутствует / устранено (BAF).
- В#16#23: Отказ питания 24В в центральном модуле / устранено.
- В#16#31: Хотя бы одна буферная батарея хотя бы в одном устройстве расширения отсутствует / устранено (BATTF)
- В#16#32: Буферное напряжение хотя бы в одном устройстве расширения отсутствует / устранено (BAF).
- В#16#33: Отказ питания 24В хотя бы в одном устройстве расширения / устранено.

## 1.10. ОВ диагностических прерываний (ОВ 82)

### Описание

Когда самодиагностируемый модуль, которому Вы разрешили диагностические прерывания, распознает ошибку, он формирует на CPU требование диагностического прерывания (как при наступающем событии, так и при текущем). При этом операционная система вызывает ОВ 82.

ОВ 82 содержит в своих локальных переменных как логические базовые адреса, так и четырехбайтовую диагностическую информацию неисправного модуля (см. таблицу 1–14).

Если ОВ 82 не был запрограммирован, то CPU переходит в рабочий режим STOP.

С помощью SFC 39 - 42 Вы можете ОВ диагностического прерывания запретить или отложить и вновь разрешить. Более подробная информация находится в главе 13.

### Локальные данные

#### ОВ диагностических прерываний

Таблица 1-14 содержит временные (TEMP) переменные ОВ диагностических прерываний. В качестве имен переменных взяты имена по умолчанию ОВ 82.

Таблица 1-14. Локальные переменные (TEMP) ОВ 82

Переменная	Тип данных	Описание
OB82_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#38: текущее событие В#16#39: наступающее событие
OB82_FLT_ID	BYTE	Код ошибки (В#16#42)
OB82_PRIORITY	BYTE	Класс приоритета: 26 (значение по умолчанию для рабочего режима RUN) или 28 (рабочий режим ANLAUF)
OB82_OB_NUMBR	BYTE	Номер ОВ (82)
OB82_RESERVED_1	BYTE	Резерв
OB82_IO_FLAG	BYTE	Модуль ввода: В#16#54 Модуль вывода: В#16#55
OB82_MDL_ADDR	INT	Базовый логический адрес группы, в которой произошла ошибка
OB82_MDL_DEFECT	BOOL	Дефект модуля
OB82_INT_FAULT	BOOL	Внутренняя ошибка
OB82_EXT_FAULT	BOOL	Внешняя ошибка
OB82_PNT_INFO	BOOL	Ошибка канала
OB82_EXT_VOLTAGE	BOOL	Отсутствует внешнее вспомогательное напряжение
OB82_FLD_CONNCTR	BOOL	Отсутствует фронтальный штекер
OB82_NO_CONFIG	BOOL	Модуль не параметрирован
OB82_CONFIG_ERR	BOOL	Неверные параметры в модуле

Таблица 1-14 Локальные переменные (TEMP) OB 82 (продолжение)

Переменная	Тип данных	Описание
OB82_MDL_TYPE	BYTE	Биты 0 - 3: Класс модуля Бит 4: Имеется информация канала Бит 5: Имеется информация пользователя Бит 6: Диагностическое прерывание от заместителя Бит 7: Резерв
OB82_SUB_MDL_ERR	BOOL	Модуль пользователя не тот / отсутствует
OB82_COMM_FAULT	BOOL	Неисправность связи
OB82_MDL_STOP	BOOL	Рабочий режим (0: RUN, 1: STOP)
OB82_WTCH_DOG_FLT	BOOL	Сработал контроль времени
OB82_INT_PS_FLT	BOOL	Отказ внутреннего питания модулей
OB82_PRIM_BATT_FLT	BOOL	Отсутствует батарея
OB82_BCKUP_BATT_FLT	BOOL	Выход из строя общей буферизации
OB82_RESERVED_2	BOOL	Резерв
OB82_RACK_FLT	BOOL	Выход из строя носителя модулей
OB82_PROC_FLT	BOOL	Выход из строя процессора
OB82_EPROM_FLT	BOOL	Сбой СППЗУ
OB82_RAM_FLT	BOOL	Сбой ОЗУ
OB82_ADU_FLT	BOOL	Сбой АЦП/ЦАП
OB82_FUSE_FLT	BOOL	Выход из строя защиты
OB82_HW_INTR_FLT	BOOL	Потеряно прерывание по сигналу от процесса
OB82_RESERVED_3	BOOL	Резерв
OB82_DATE_TIME	DATE_AND_TIME	Дата и время, когда был затребован OB

## 1.11. ОВ снятия/установки модулей (ОВ 83)

### Описание

Установка или снятие модулей контролируется внутри системы каждую секунду. Чтобы установка или снятие модуля распознавались CPU, необходимо чтобы между установкой и снятием прошло как минимум две секунды.

Каждая установка или снятие запроецированного модуля (не разрешено для блоков питания, CPU, адаптерных корпусов и IM) в режимах RUN, STOP и ANLAUF приводит к прерыванию снятия/установки. Это прерывание вызывает у соответствующего CPU запись в диагностический буфер и в список состояний системы. Кроме того в режиме RUN осуществляется запуск ОВ снятия/установки. Если этот ОВ не был запрограммирован, то CPU переходит в состояние STOP.

С помощью SFC 39 - 42 Вы можете ОВ снятия/установки запретить или отложить и вновь разрешить. Более подробная информация содержится в главе 11.

### Принцип действия ОВ снятия/ установки модулей

При снятии запроецированного модуля в рабочем режиме RUN запускается ОВ 83. Так как снятие контролируется с интервалом в 1 секунду, то до этого может быть распознана ошибка доступа при прямом обращении или актуализации отображения процесса.

При установке модуля в запроецированный слот в режиме RUN операционная система перепроверяет соответствие установленного модуля с проектом. Затем запускается ОВ 83 и при соответствии типов модулей осуществляется параметрирование.

### Локальные данные ОВ снятия/ установки модулей

Таблица 1-15 содержит временные (TEMP) переменные ОВ снятия/установки. В качестве имен переменных взяты имена по умолчанию ОВ 83.

Таблица 1-15. Локальные переменные (TEMP) ОВ 83

Переменные	Тип данных	Описание
OB83_EV_CLASS	BYTE	Класс события идентификаторы: В#16#38: модуль установлен В#16#39: модуль снят или не реагирует
OB83_FLT_ID	BYTE	Коды ошибок (возможные значения: В#16#61, В#16#63, В#16#64, В#16#65)
OB83_PRIORITY	BYTE	Класс приоритета: 26 (по умолчанию рабочее состояние RUN) или 28 (рабочее состояние ANLAUF)
OB83_OB_NUMBR	BYTE	Номер ОВ (83)
OB83_RESERVED_1	BYTE	Резерв
OB83_MDL_TD	BYTE	Область: В#16#54: периферийная область входов (PE) В#16#55: периферийная область выходов (PA)
OB83_MDL_ADDR	WORD	Логический базовый адрес соответствующего модуля
OB83_RACK_NUM	WORD	Номер носителя модулей или № DP-станции (младший байт) и идентификатор мастер-системы DP (старший байт)
OB83_MDL_TYPE	WORD	Тип соответствующего модуля
OB83_DATE_TIME	DATE_AND_TIME	Дата и время, когда был запрошен ОВ



Переменная OB83\_MDL\_TYPE имеет следующее значение в зависимости от кода ошибки:

- Код ошибки V#16#61: Модуль установлен, тип модуля верен (для класса события V#16#38)  
Модуль снят или не срабатывает (класс события V#16#39)  
OB83\_MDL\_TYPE: Фактический тип модуля
- Код ошибки V#16#63: Модуль установлен, однако не верен тип модуля  
OB83\_MDL\_TYPE: Фактический тип модуля
- Код ошибки V#16#64: Модуль установлен, однако неисправен (не читается идентификатор типа)  
OB83\_MDL\_TYPE: Заданный в конфигурации тип модуля
- Код ошибки V#16#65: Модуль установлен, однако ошибка при параметрировании модуля  
OB83\_MDL\_TYPE: Фактический тип модуля

## 1.12. ОВ аппаратных ошибок CPU (ОВ 84)

<b>Описание</b>	<p>Операционная система CPU вызывает ОВ 84, если распознается ошибка в интерфейсе с сетью MPI, K-шиной или с подключением децентрализованной периферии.</p> <p>Если возникла такая ошибка, а ОВ 84 не был запрограммирован, то CPU переходит в состояние STOP.</p> <p>С помощью SFC 39 - 42 Вы можете ОВ аппаратных ошибок запретить или отложить и вновь разрешить. Более подробная информация находится в главе 11.</p>
<b>Локальные данные ОВ аппаратных ошибок</b>	<p>Таблица 1-16 содержит временные (TEMP) переменные ОВ аппаратных ошибок CPU. В качестве имен переменных взяты имена по умолчанию ОВ 84.</p>

Таблица 1-16. Локальные переменные (TEMP) ОВ 84

Переменная	Тип данных	Описание
OB84_EV_CLASS	BYTE	Класс события и идентификатор: B#16#38: текущее событие B#16#39: наступающее событие
OB84_FLT_ID	BYTE	Код ошибок (B#16#81)
OB84_PRIORITY	BYTE	Класс приоритета: 26 (по умолчанию для рабочего режима RUN или 28 (режим ANLAUF)
OB84_OB_NUMBR	BYTE	Номер ОВ (84)
OB84_RESERVED_1	BYTE	Резерв
OB84_RESERVED_2	BYTE	Резерв
OB84_RESERVED_3	WORD	Резерв
OB84_RESERVED_4	DWORD	Резерв
OB84_DATE_TIME	DATE_AND_TIME	Дата и время, когда был затребован ОВ

### 1.13. ОВ ошибок исполнения программы (ОВ 85)

#### Описание

Операционная система CPU вызывает ОВ 85 в случаях, когда возникает одно из следующих событий:

- Стартовое событие для незагруженного ОВ
- Ошибка при обращении операционной системы к блоку
- Ошибка доступа к периферии при обновлении общего отображения процесса.

---

#### Замечание

Если ОВ 85 не был запрограммирован, а хотя бы одно из этих событий наступило, то CPU переводится в режим STOP.

---

С помощью SFC 39 - 42 Вы можете ОВ ошибок исполнения программы запретить или отложить и вновь разрешить. Более подробная информация находится в главе 11.

#### Локальные данные ОВ ошибок исполнения программы

Таблица 1-17 содержит временные (TEMP) переменные ОВ ошибок исполнения программы. В качестве имен переменных взяты имена по умолчанию ОВ 85.

Таблица 1-17 Локальные переменные (TEMP) ОВ 85

Переменная	Тип данных	Описание
OB85_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#35, В#16#39 (только при кодах ошибок В#16#В1 и В#16#В2)
OB85_FLT_ID	BYTE	Коды ошибок (возможные значения: В#16#А1, В#16#А2, В#16#А3, В#16#В1, В#16#В2)
OB85_PRIORITY	BYTE	Класс приоритета: 26 (значение по умолчанию для рабочего режима RUN) или 28(рабочий режим ANLAUF)
OB85_OB_NUMBR	BYTE	Номер ОВ (85)
OB85_RESERVED_1	BYTE	Резерв
OB85_RESERVED_2	BYTE	Резерв
OB85_RESERVED_3	INT	Резерв
OB85_ERR_EV_CLASS	BYTE	Класс события, которое вызвало ошибку.
OB85_ERR_EV_NUM	BYTE	Номер события, которое вызвало ошибку
OB85_OB_PRIOR	BYTE	Класс приоритета ОВ, который обрабатывался в момент возникновения ошибки
OB85_OB_NUM	BYTE	Номер ОВ, который обрабатывался во время возникновения ошибки
OB85_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван ОВ

Если желательно запрограммировать OB 85 в зависимости от возможных кодов ошибок, рекомендуется локальные переменные организовывать следующим образом:

Таблица 1-18. Локальные переменные (TEMP) OB 85

Переменные	Тип данных
OB85_EV_CLASS	BYTE
OB85_FLT_ID	BYTE
OB85_PRIORITY	BYTE
OB85_OB_NUMBR	BYTE
OB85_DKZ23	BYTE
OB85_RESERVED_2	BYTE
OB85_Z1	WORD
OB85_Z23	DWORD
OB85_DATE_TIME	DATE_AND_TIME

Измененные по сравнению с умолчанием переменные в зависимости от кодов ошибок имеют следующие значения:

- Код ошибки B#16#A1      Ваша программа или операционная система (на основе Вашего проектирования с помощью STEP 7) генерирует стартовое событие для OB, который не загружен в CPU.

B#16#A2      Ваша программа или операционная система (на основе Вашего проектирования с помощью STEP 7) генерирует стартовое событие для OB, который не загружен в CPU.

OB85\_Z1:      Как соответствующее значение в прерванной программе.

OB85\_Z23:      старшее слово: класс и номер события, инициирующего прерывание  
младшее слово: на момент появления ошибки активный уровень программы и активный OB
- Код ошибки B#16#A3      Ошибка при обращении операционной системы к блоку

OB85\_Z1:      идентификатор ошибки операционной системы  
старшие биты:    1: встроенная функция  
                      2: IEC-таймер  
младшие биты:    0: ошибки не обнаружены  
                      1: блок не загружен  
                      2: ошибка длины области  
                      3: ошибка защиты записи

OB85\_Z23      старшее слово:    номер блока  
младшее слово:    относительный адрес команды MC7, вызвавшей ошибку. Тип блока следует    взять из OB

85\_DKZ23      (B#16#88: OB, B#16#8C: FC, B#16#8E: FB, B#16#8A: DB).

- Код ошибки B#16#B1:Ошибка доступа к периферии при обновлении  
общего отображения процесса на входах  
B#16#B2:Ошибка доступа к периферии при обновлении  
общего отображения процесса на выходах  
OB85\_Z1: Зарезервирована для внутреннего использования CPU  
OB85\_Z23: Номер периферийного байта, вызвавшего ошибку доступа  
к периферии

## 1.14. ОВ неисправностей носителей модулей (ОВ 86)

### Описание

Операционная система CPU вызывает ОВ 86 в случаях, когда распознается неисправность носителя модулей, подсети или станции у децентрализованной периферии (как при текущем, так и при наступающем событии).

Если ОВ 86 не был Вами запрограммирован, а происходит такая ошибка, то CPU переходит в состояние STOP.

С помощью SFC 39 - 42 Вы можете ОВ 86 запретить или отложить и вновь разрешить. Более подробная информация находится в главе 11.

### Локальные данные

### ОВ неисправностей носителей модулей

Таблица 1-19 содержит временные (TEMP) переменные ОВ неисправностей носителей модулей. В качестве имен переменных взяты имена по умолчанию ОВ 86.

Таблица 1-19. Локальные переменные (TEMP) ОВ 86

Переменная	Тип данных	Описание
OB86_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#38: текущее событие В#16#39: наступающее событие
OB86_FLT_ID	BYTE	Код ошибки (возможные значения: В#16#C1, В#16#C3, В#16#C4, В#16#C5, В#16#C6, В#16#C7)
OB86_PRIORITY	BYTE	Класс приоритета: 26 (значение по умолчанию для рабочего режима RUN) или 28 (рабочий режим ANLAUF)
OB86_OB_NUMBR	BYTE	Номер ОВ (86)
OB86_RESERVED_1	BYTE	Резерв
OB86_RESERVED_2	BYTE	Резерв
OB86_MDL_ADDR	WORD	В зависимости от кода ошибки
OB86_RACKS_FLTD	ARRAY [0 ..31] OF BOOL	В зависимости от кода ошибки
OB86_DATE_TIME	DATE_AND_TIME	Дата и время, когда был затребован ОВ

Если Вы желаете программировать OB 86 в зависимости от кодов ошибок, то рекомендуется локальные переменные организовывать следующим образом:

Таблица 1-20. Локальные переменные (TEMP) OB 86

Переменная	Тип данных
OB86_EV_CLASS	BYTE
OB86_FLT_ID	BYTE
OB86_PRIORITY	BYTE
OB86_OB_NUMBR	BYTE
OB86_RESERVED_1	BYTE
OB86_RESERVED_2	BYTE
OB86_MDL_ADDR	WORD
OB86_Z23	DWORD
OB86_DATE_TIME	DATE_AND_TIME

Переменные, содержание которых зависит от кодов ошибок, имеют следующее значение:

- Код ошибки B#16#C1:      неисправность носителя модулей
- OB86\_MDL\_ADDR:      логический базовый адрес IM
- OB86\_Z23:      содержит для каждого возможного носителя модулей один бит:
  - Бит 0:      центральное устройство (всегда 0)
  - Бит 1:      1-е устройство расширения
  - :      :
  - :      :
  - Бит 21:      21-е устройство расширения
  - Биты 22 - 29:      всегда 0
  - Бит 30:      неисправность носителя модулей в области SIMATIC-S5
  - Бит 31:      всегда 0

Значение: при наступающем событии индицируются как неисправные те носители модулей (соответствующие биты устанавливаются), которые обусловили вызов OB 86. При этом носители модулей, вышедшие из строя ранее, не индицируются. При текущем событии индицируются восстановленные носители модулей (соответствующие биты устанавливаются).

- Код ошибки B#16#C2: Восстановление носителя модулей (выход из строя носителя модулей с отклонением истинной конфигурации от заданной)

OB86\_MDL\_ADDR: логический базовый адрес IM

OB86\_Z23: Содержит для каждого возможного носителя модулей один бит, см. код ошибки B#16#C1. Значение установленного бита: в соответствующем носителе модулей

  - имеются модули с неправильным идентификатором типа.
  - отсутствуют запроектированные модули.
  - хотя бы один модуль неисправен.
- Код ошибки B#16#C3: Децентрализованная периферия: выход из строя подсети PROFIBUS (Только наступающее событие ведет к старту OB 86 с кодом ошибки B#16#C3. Текущее событие ведет к старту OB 86 с кодом ошибки B#16#C4 и классом события B#16#38: следствием восстановления любой нижерасположенной DP-станции является старт OB 86).

OB86\_MDL\_ADDR: логический базовый адрес ведущего DP

OB86\_Z23: DP-Mastersystem-ID (идентификатор мастер-системы DP)

  - Биты 0 - 7: зарезервированы
  - Биты 8 - 15: DP-Mastersystem-ID (идентификатор мастер-системы DP)
  - Биты 16 - 31: зарезервированы
- Код ошибки B#16#C4: Выход из строя DP-станции

Код ошибки B#16#C5: Сбой DP-станции

OB86\_MDL\_ADDR: логический базовый адрес ведущего DP

OB86\_Z23: адрес соответствующих ведомых DP (DP-Slave)

  - Бит 0 - 7: номер DP-станции
  - Бит 8 - 15: DP-Master-ID (идентификатор мастер-системы DP)
  - Бит 16 - 30: логический базовый адрес у S7-Slave или диагностический адрес DP-Normslave
  - Бит 31: Идентификатор ввода/вывода



- ⑩ Код ошибки B#16#C6: Восстановление носителя модулей, однако  
ошибка при параметрировании модуля
- OB86\_MDL\_ADDR: логический базовый адрес IM
- OB86\_Z23: содержит один бит для каждого возможного  
носителя модулей  
Бит 0: центральное устройство  
(всегда 0)  
Бит 1:1-е устройство расширения  
:  
:  
Бит 21: 21-е устройство расширения  
Биты 22 - 30: зарезервированы  
Бит 31: всегда 0  
Значение установленного бита в  
соответствующем носителе модулей:  
- существуют модули с неправильным  
идентификатором типа.  
- существуют модули с неправильными или  
не заданными параметрами.
- Код ошибки B#16#C7: Восстановление DP-станции, однако  
ошибка параметризации модуля
- OB86\_MDL\_ADDR: логический базовый адрес ведущего DP
- OB86\_Z23: адрес соответствующего DP-Slave  
Бит 0 - 7: номер DP-станции  
Бит 8 - 15: DP-Mastersystem-ID  
(идентификатор мастер-системы DP)  
Бит 16 - 30: логический базовый адрес  
DP-Slave  
Бит 31: идентификатор ввода/вывода

## 1.15. ОВ коммуникационных ошибок (ОВ 87)

<b>Описание</b>	Операционная система CPU вызывает ОВ 87 в случаях, когда наступает событие, которое было вызвано коммуникационной ошибкой. Если ОВ 87 не был запрограммирован, то CPU при наступлении такого события меняет рабочий режим на STOP. С помощью SFC 39 - 42 Вы можете ОВ коммуникационных ошибок запретить или отложить и вновь разрешить. Более подробная информация находится в главе 11.
<b>Локальные данные ОВ коммуника- ционных ошибок</b>	Таблица 1-21 содержит временные (TEMP) переменные ОВ коммуникационных ошибок. В качестве имен переменных взяты имена по умолчанию ОВ 87.

Таблица 1-21. Локальные переменные (TEMP) ОВ 87

Переменная	Тип данных	Описание
OB87_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#35
OB87_FLT_ID	BYTE	Коды ошибок (возможные значения: В#16#D2, В#16#D3, В#16#D4, В#16#D5, В#16#E1, В#16#E2, В#16#E3, В#16#E4, В#16#E5, В#16#E6)
OB87_PRIORITY	BYTE	Класс приоритета: 26 (по умолчанию для рабочего режима RUN) или 28 (рабочий режим ANLAUF)
OB87_OB_NUMBR	BYTE	Номер ОВ (87)
OB87_RESERVED_1	BYTE	Резерв
OB87_RESERVED_2	BYTE	Резерв
OB87_RESERVED_3	WORD	В зависимости от кодов ошибок
OB87_RESERVED_4	DWORD	В зависимости от кодов ошибок
OB87_DATE_TIME	DATE_AND_TIME	Дата и время, когда был затребован ОВ

Переменные, зависящие от кода ошибок, имеют следующие значения:

- Код ошибки В#16#D2: передача диагностических записей в настоящее время невозможна.  
В#16#D3: не могут быть посланы синхронизационные кодовые послышки (Master).  
В#16#D4: недопустимый скачок времени при синхронизации времени  
В#16#D5: ошибка при приеме времени синхронизации (Slave)  
OB87\_RESERVED\_3: Не содержит дальнейшей информации  
OB87\_RESERVED\_4: Не содержит дальнейшей информации
- Код ошибки В#16#E1: Неверный идентификатор кодовой послышки при связи с помощью глобальных данных.  
В#16#E3: Ошибка длины кодовой послышки при связи с помощью глобальных данных  
В#16#E4: Принят недопустимый номер пакета GD  
OB87\_RESERVED\_3: старший байт: идентификатор интерфейса (0: К-шина, 1: MPI)  
младший байт: номер GD - контура  
OB87\_RESERVED\_4: Не содержит дальнейшей информации
- Код ошибки В#16#E2: Статус пакета GD не может быть внесен в DB

- |                  |   |
|------------------|---|
| OB87_RESERVED_3: | Номер DB  |
| OB87_RESERVED_4: | старшее слово: не содержит дальнейшей информации<br>младшее слово: номер GD-контура (старший байт),<br>номер пакета GD (младший байт) |
- Код ошибки B#16#E5: Ошибка доступа к DB при обмене данными через коммуникационные функциональные блоки
- |                  |  |
|------------------|--|
| OB87_RESERVED_3: | Зарезервирована для внутреннего применения CPU   |
| OB87_RESERVED_4: | старшее слово: номер блока с вызвавшей ошибку командой MC7<br>младшее слово: относительный адрес команды MC7, вызвавшей ошибку |
- Тип блока следует взять из OB87\_RESERVED\_1 (B#16#88: OB, B#16#8A: DB, B#16#8C: FC, B#16#8E: FB).
- Код ошибки B#16#E6: Общий статус GD не заносится в DB.
- |                  |                                     |
|------------------|-------------------------------------|
| OB87_RESERVED_3: | Номер DB                            |
| OB87_RESERVED_4: | Не содержится дальнейшей информации |

## 1.16. ОВ фонового режима (ОВ 90)

### Описание

S7 представляет возможность контролировать максимальное время цикла и гарантирует обеспечение минимального времени цикла. Если время обработки ОВ 1, включая все вложенные обработки прерываний и системные операции, оказывается меньше заданной Вами минимальной длительности цикла, то:

- операционная система вызывает ОВ фонового режима (если он имеется в CPU)
- операционная система задерживает следующий старт ОВ1 (в случае, если ОВ 90 отсутствует в CPU).

### Принцип действия ОВ 90

ОВ 90 имеет самый низкий приоритет среди всех остальных ОВ. Он прерывается любой системной операцией и обработкой любого прерывания и продолжается только в том случае, когда заданное минимальное время цикла еще не достигнуто. Исключение составляет обработка SFC, которые были запущены из ОВ 90. Они обрабатываются с приоритетом ОВ 1 и поэтому не прерываются обработкой ОВ 1. Контроля длительности ОВ 90 не ведется.

Операционная система CPU вызывает ОВ 90 в следующих случаях:

- после нового пуска или повторного пуска
- после стирания одного из блоков, находящихся в обработке в ОВ 90 (посредством STEP 7 или SFC 23 "DEL\_DB")
- после загрузки ОВ 90 в CPU в рабочем режиме RUN
- по окончании фонового цикла.

В отличие от других ОВ при повторном пуске системы ОВ 90 продолжает не с места, на котором он прервался, а сбрасывается и только в режиме RUN стартует вновь. Кроме того, ОВ 90 сбрасывается в следующих случаях:

- при новом пуске
- если один из содержащихся в ОВ 90 блоков прикладной программы стерт или заменен.

### Локальные данные ОВ 90

Таблица 1-22 содержит временные (TEMP) ОВ 90. В качестве имен переменных взяты имена по умолчанию ОВ 90.

Таблица 1-22 Локальные переменные (TEMP) ОВ 90

Переменная	Тип данных	Описание
ОВ90_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11: активен
ОВ90_STRT_INF	BYTE	В#16#91: Новый пуск / повторный пуск В#16#92: стирание блока В#16#93: загрузка ОВ 90 в CPU в режиме RUN В#16#95: окончание фонового цикла
ОВ90_PRIORITY	BYTE	Класс приоритета: 29
ОВ90_OB_NUMBR	BYTE	Номер ОВ (90)
ОВ90_RESERVED_1	BYTE	Резерв

Таблица 1-22. Локальные переменные (TEMP) OB 90 (продолжение)

Переменная	Тип данных	Описание
OB90_RESERVED_2	BYTE	Резерв
OB90_RESERVED_3	INT	Резерв
OB90_RESERVED_4	INT	Резерв
OB90_RESERVED_5	INT	Резерв
OB90_DATE_TIME	DATE_AND_TIME	Дата и время, когда был затребован OB

## 1.17. ОВ запуска (ОВ 100 и ОВ 101)

### Виды запуска

- В S7–300 существует только один вид пуска - новый пуск.
- В S7–400 возможны новый пуск и повторный пуск.

Во время пуска операционная система вызывает соответствующий ОВ запуска: при новом пуске - ОВ нового пуска (ОВ 100), при повторном пуске - ОВ повторного пуска (ОВ 101). Более подробную информацию по другим операциям в режиме запуска Вы найдете в /234/.

### Описание

CPU производит запуск:

- при включении сетевого напряжения
- при переводе переключателя режимов работы из положения STOP в RUN
- по запросу через коммуникационную функцию (по команде меню из PG или вызовом коммуникационных функциональных блоков “START” или “RESUME” на другом CPU).

В зависимости от стартового события, от имеющегося CPU и установленных на нем параметров вызывается ОВ нового пуска или ОВ повторного пуска. В нем Вы соответствующим программированием можете сделать определенные предварительные установки для Вашей циклической программы.

### Локальные данные ОВ запуска

Таблица 1-23 содержит временные (TEMP) переменные ОВ запуска. В качестве имен переменных взяты имена по умолчанию ОВ 100

Таблица 1-23. Локальные переменные (TEMP) ОВ 100

Переменная	Тип данных	Описание
OB100_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#13: активен
OB100_STRTUP	BYTE	Требование запуска: V#16#81: ручной новый пуск V#16#82: автоматический новый пуск V#16#83: ручной повторный пуск V#16#84: автоматический повторный пуск
OB100_PRIORITY	BYTE	Класс приоритета: 27
OB100_OB_NUMBR	BYTE	Номер ОВ (100)
OB100_RESERVED_1	BYTE	Резерв
OB100_RESERVED_2	BYTE	Резерв
OB100_STOP	WORD	Номер события, которое перевело CPU в STOP
OB100_STRT_INFO	DWORD	Дополнительная информация для текущего пуска (смотри таблицу)
OB100_DATE_TIME	DATE_AND_TIME	Дата и время, когда был затребован ОВ

Таблица 1-24. Распределение битов переменных OB100 STRT INFO и OB101 STRT INFO

№ бита	Назначение	Возможные значения (двоичные)	Пояснение
31 – 28	Многопроцессорный режим	0000 0001 0010	Однопроцессорный режим Мультипроцессорный режим (только для S7–400) Работа нескольких центральных модулей на сегментированном носителе модулей (только для S7–400)
27 – 24	Дополнительный идентификатор запуска	...0 ...1 ..0. ..1. 0... 1...	Нет различия между фактической и заданной конфигурацией (только S7–300) Есть отличия между фактической и заданной конфигурацией (только S7–300) Нет различий между фактической и заданной конфигурацией Есть различия между фактической и заданной конфигурацией Часы для метки времени при последнем включении питания не буферизовались Часы для метки времени при последнем включении питания буферизовались
23 – 16	Вид только что проведенного пуска	00000001  00000011 00000100 00001010  00001011 00001100 00010000 00010011  00010100  00100000 00100011 00100100  1010 0000	Ручной новый пуск при мультипроцессорной обработке без управления на CPU в соответствии с параметризацией (только S7–400) Ручной новый пуск от переключателя режимов работы Ручной новый пуск через коммуникационную функцию Ручной повторный пуск при мультипроцессорной обработке без управления на CPU в соответствии с параметризацией (только S7–400) Ручной повторный пуск от переключателя режимов работы (только S7–400) Ручной повторный пуск через коммуникационную функцию (только S7–400) Автоматический новый пуск после буферизованного включения питания Ручной новый пуск от переключателя режимов работы, последнее включение питания буферизовано Ручной новый пуск через коммуникационную функцию; последнее включение питания буферизовано Автоматический новый пуск после небуферизованного включения питания (с общим стиранием со стороны системы) Ручной новый пуск от переключателя режимов работы; последнее включение питания не буферизовано Ручной новый пуск через коммуникационную функцию; последнее включение питания не буферизовано  Автоматический повторный пуск после буферизованного включения питания в соответствии с параметризацией (только S7–400)

15 – 12	Допустимость видов автоматического пуска	0000	Автоматический пуск недопустим, затребовано общее стирание
		0001	Автоматический пуск недопустим, требуется изменение параметров и т.д.
		0111	Автоматический новый пуск допустим
		1111	Автоматический новый пуск / повторный пуск допустимы (только S7–400)
11 – 8	Допустимость видов ручного пуска	0000	Ручной пуск недопустим, затребовано общее стирание
		0001	Ручной пуск недопустим, требуется изменение параметров и т.д.
		0111	Ручной новый пуск допустим
		1111	Ручной новый пуск / повторный пуск допустимы (только S7–400)
7 – 0	Последнее действительное управление или установка автоматического пуска при включении питания	00000000	Нет пуска
		00000001	Ручной новый пуск при мультипроцессорной обработке без управления на CPU в соответствии с параметрированием (только S7–400)
		00000011	Ручной новый пуск от переключателя режимов работы
			Ручной новый пуск через коммуникационную функцию
		00000100	Ручной повторный пуск при мультипроцессорной обработке без управления на CPU в соответствии с параметрированием (только S7–400)
		00001010	Ручной повторный пуск от переключателя режимов работы (только S7–400)
		00001011	Ручной новый пуск через коммуникационную функцию (только S7–400)
		00001100	Автоматический новый пуск после буферизованного включения питания
		00010000	Ручной новый пуск через переключатель режимов работы; последнее включения питания
		00010011	буферизовано
		00010100	Ручной новый пуск через коммуникационную функцию; последнее включение питания
			буферизовано
		00100000	Автоматический новый пуск после небуферизованного включения питания (с общим стиранием со стороны системы)
		00100011	Ручной новый пуск через переключатель режимов работы; последнее включение питания не буферизовано
		00100100	Ручной новый пуск через коммуникационную функцию; последнее включение питания не буферизовано
			Автоматический повторный пуск после буферизованного включения питания в соответствии с параметрированием (только S7–400)
		1010 0000	



## 1.18. ОВ ошибок программирования (ОВ 121)

**Описание** Операционная система CPU вызывает ОВ 121 в тех случаях, когда наступает событие, вызванное ошибкой при обработке Вашей программы. Если Вы, например, вызвали в Вашей программе блок, который не был загружен в CPU, то вызывается ОВ 121.

**Принцип работы ОВ ошибок программирования** ОВ 121 работает с тем же приоритетом, что и прерванный блок. Если ОВ 121 не был запрограммирован, то CPU меняет рабочее состояние с RUN на STOP.

S7 располагает следующими SFC, с помощью которых Вы можете маскировать и демаскировать стартовые события ОВ 121 в то время, когда обрабатывается Ваша программа:

- SFC36 "MSK\_FLT" маскирует определенные коды ошибок.
- SFC37 "DMSK\_FLT" демаскирует коды ошибок, которые были замаскированы с помощью SFC36.
- SFC38 "READ\_ERR" читает регистр состояния событий.

### Локальные данные ОВ ошибок программирования

Таблица 1-25. Локальные переменные (TEMP) ОВ 121

Переменная	Тип данных	Описание
OB121_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#25
OB121_SW_FLT	BYTE	Коды ошибок (возможные значения: В#16#21, В#16#22, В#16#23, В#16#24, В#16#25, В#16#26, В#16#27, В#16#28, В#16#29, В#16#30, В#16#31, В#16#32, В#16#33, В#16#34, В#16#35, В#16#3A, В#16#3C, В#16#3D, В#16#3E, В#16#3F)
OB121_PRIORITY	BYTE	Класс приоритета: класс приоритета ОВ, в котором произошла ошибка
OB121_OB_NUMBR	BYTE	Номер ОВ (121)
OB121_BLK_TYPE	BYTE	Вид блока, в котором произошла ошибка (для S7-300 не заносится никакого действующего значения): В#16#88: ОВ, В#16#8A: DB, В#16#8C: FC, В#16#8E: FB
OB121_RESERVED_1	BYTE	Резерв
OB121_FLT_REG	WORD	Источник ошибки (зависит от кода ошибки), например: <ul style="list-style-type: none"><li>• регистр, в котором возникла ошибка конвертирования</li><li>• ошибочный адрес (ошибка записи / чтения)</li><li>• ошибочный номер таймера, счетчика или блока</li><li>• ошибочная область памяти</li></ul>
OB121_BLK_NUM	WORD	Номер блока с командой MC7, вызвавшей ошибку
OB121_PRG_ADDR	WORD	Относительный адрес команды MC7, вызвавшей ошибку (у S7-300 не устанавливается никакого действующего значения)
OB121_DATE_TIME	DATE_AND_TIME	Дата и время, когда был затребован ОВ

Переменные, зависящие от кодов ошибок, имеют следующие значения:

- Код ошибки В#16#21: Ошибка преобразования в BCD-код

- OB121\_FLT\_REG: Идентификатор соответствующего регистра (W#16#0000: аккумулятор 1)
- Код ошибки V#16#22: Ошибка длины области при чтении  
V#16#23: Ошибка длины области при записи  
V#16#28: Обращение для чтения к байту, слову или двойному слову с указателем, битовый адрес которого не равен нулю  
V#16#29: Обращение для записи к байту, слову или двойному слову с указателем, битовый адрес которого не равен нулю
- OB121\_FLT\_REG: Ошибочный байтовый адрес. Область данных и способ доступа следует получить из
- OB121\_RESERVED\_1.
- OB121\_RESERVED\_1: Биты 7 - 4 способ доступа:  
0: битовый доступ, 1: байтовый доступ,  
2: доступ к слову, 3: доступ к двойному слову
- Биты 3 - 0 область памяти:  
0: область периферии,  
1: отображение процесса на входах,  
2: отображение процесса на выходах,  
3: меркеры, 4: глобальный DB,  
5: экземпляр DB,  
6: собственные локальные данные,  
7: локальные данные вызывающего блока
- Код ошибки V#16#24: Ошибка области при чтении  
V#16#25: Ошибка области при записи
- OB121\_FLT\_REG: Содержит в младшем байте идентификатор недоступной области (V#16#86 область собственных локальных данных).
- Код ошибки V#16#26: Ошибка номера таймера  
V#16#27: Ошибка номера счетчика
- OB121\_FLT\_REG: Недопустимый номер
- Код ошибки V#16#30: Попытка записи в защищенный от записи глобальный DB  
V#16#31: Попытка записи в защищенный от записи экземпляр DB  
V#16#32: Ошибка номера DB при обращении к глобальному DB  
V#16#33: Ошибка номера DB при обращении к экземпляру DB
- OB121\_FLT\_REG: Недопустимый номер DB
- Код ошибки V#16#34: Ошибка номера FC при вызове FC  
V#16#35: Ошибка номера FB при вызове FB
- V#16#3A: Обращение к незагруженному DB; номер DB находится в допустимой области.
- V#16#3C: Обращение к незагруженной FC; номер FC находится в допустимой области.

B#16#3D:	Обращение к незагруженной SFC; номер SFC лежит в допустимой области.
B#16#3E:	Обращение к незагруженному FB; номер FB лежит в допустимой области.
B#16#3F:	Обращение к незагруженному SFB; номер SFB лежит в допустимой области.
OB121_FLT_REG:	Недопустимый номер

## 1.19. ОВ ошибок доступа к периферии (ОВ 122)

### Описание

Операционная система CPU вызывает ОВ 122 в случае, если при обращении к данным какого-либо модуля происходит ошибка. Если, например, CPU распознает ошибку чтения при обращении к данным сигнального модуля, то операционная система вызывает ОВ 122.

### Принцип работы ОВ ошибок доступа к периферии

ОВ 122 работает в том же классе приоритета, что и прерванный блок. Если ОВ 122 не запрограммирован, то CPU переходит из RUN в STOP. **к** S7 располагает следующими SFC, с помощью которых можно маскировать и демаскировать стартовые события ОВ 122 во время обработки Вашей программы:

- SFC36 "MSK\_FLT" маскирует определенные коды ошибок
- SFC37 "DMSK\_FLT" демаскирует коды ошибок, замаскированные с помощью SFC 36.
- SFC38 "READ\_ERR" читает регистр состояния событий.

### Локальные данные ОВ ошибок доступа к периферии

Таблица 1-26 содержит временные (TEMP) переменные ОВ ошибок доступа к периферии. В качестве имен переменных взяты имена по умолчанию ОВ 122.

Таблица 1-26 Локальные переменные (TEMP) ОВ 122

Переменная	Тип данных	Описание
OB122_EV_CLASS	BYTE	Класс события и идентификаторы: B#16#29
OB122_SW_FLT	BYTE	Код ошибки B#16#42 Для S7-300: ошибка доступа к периферии, для чтения Для S7-400: ошибка при первом обращении для чтения после появления ошибки B#16#43 Для S7-300: ошибка доступа к периферии, для записи Для S7-400: ошибка при первом обращении для записи после появления ошибки B#16#44 (только у S7-400): ошибка при n-ом обращении для чтения ( $n > 1$ ) после появления ошибки B#16#45 (только у S7-400) ошибка при n-ом обращении записи ( $n > 1$ ) после появления ошибки
OB122_PRIORITY	BYTE	Класс приоритета: Класс приоритета ОВ, в котором произошла ошибка
OB122_OB_NUMBR	BYTE	Номер ОВ (122)
OB122_BLK_TYPE	BYTE	Тип блока, в котором произошла ошибка (B#16#88: OB, B#16#8A: DB, B#16#8C: FC, B#16#8E: FB)
OB122_MEM_AREA	BYTE	Область памяти и вид доступа: Биты 7 - 4 способ доступа: 0: битовый доступ, 1: байтовый доступ, 2: доступ к слову, 3: доступ к двойному слову Биты 3 - 0 область памяти: 0: область периферии, 1: отображение процесса на входах, 2: отображение процесса на выходах
OB122_MEM_ADDR	WORD	Адрес в памяти, на котором произошла ошибка
OB122_BLK_NUM	WORD	Номер блока с командой MC7, вызвавшей ошибку
OB122_PRG_ADDR	WORD	Относительный адрес команды MC7, вызвавшей ошибку
OB122_DATE_TIME	DATE_AND_TIME	Дата и время, когда был запрошен ОВ