

SFC для манипулирования событиями, вызывающими синхронные ошибки

10

Обзор главы

В раз-деле	Вы найдете	на стр.
10.1	Маскирование событий, вызывающих синхронные ошибки	10–2
10.2	Маскирование событий, вызывающих синхронные ошибки, с помощью SFC 36 "MSK_FLT"	10–9
10.3	Демаскирование событий, вызывающих синхронные ошибки, с помощью SFC 37 "DMSK_FLT"	10–10
10.4	Чтение регистра состояний события с помощью SFC 38 "READ_ERR"	10–11

10.1. Маскирование событий, вызывающих синхронные ошибки

Введение

Событиями, вызывающими синхронные ошибки, являются ошибки программирования и доступа. Такие события появляются вследствие программирования с использованием неправильных областей операндов, номеров операндов или неправильных адресов. Маскирование таких событий означает, что

- маскированные события не запускают ОВ ошибки и не приводят к запрограммированной реакции компенсации,
- при маскированных ошибках CPU “запоминает” появившиеся ошибки в регистре состояний события.

Демаскирование событий, вызывающих ошибки, означает отмену ранее выполненного маскирования. Маскирование отменяется

- путем вызова SFC37 ”DMSK_FLT”
- когда заканчивается обработка действующего ОВ.

Если событие, вызывающее ошибку, появляется после его демаскирования, то операционной системой запускается принадлежащий ему ОВ. Для реагирования на ошибку программирования программируйте ОВ 121, а для реагирования на ошибку доступа программируйте ОВ 122.

Вы можете считывать маскированные и появившиеся события, вызывающие ошибки, с помощью SFC 38 ”READ_ERR”.

Указание: Независимо от маскирования или демаскирования событий, вызывающих ошибки, в S7–300 происходит запись этих событий в диагностический буфер, и загорается светодиод общей ошибки CPU.

Общие сведения об обработке ошибок

Если в прикладной программе появляются ошибки программирования или ошибки доступа, то Вы можете реагировать на них по-разному:

- Вы можете запрограммировать ОВ ошибки, и операционная система вызовет этот ОВ ошибки, когда появится соответствующее событие, вызвавшее ошибку.
- Вы можете заблокировать вызов ОВ ошибки для каждого приоритетного класса отдельно. В этом случае CPU не переходит в STOP, когда в приоритетном классе появляется соответствующая ошибка. CPU записывает соответствующую ошибку в регистр состояний события. Однако по этой записи Вы не можете узнать, когда и как часто эта ошибка появлялась.

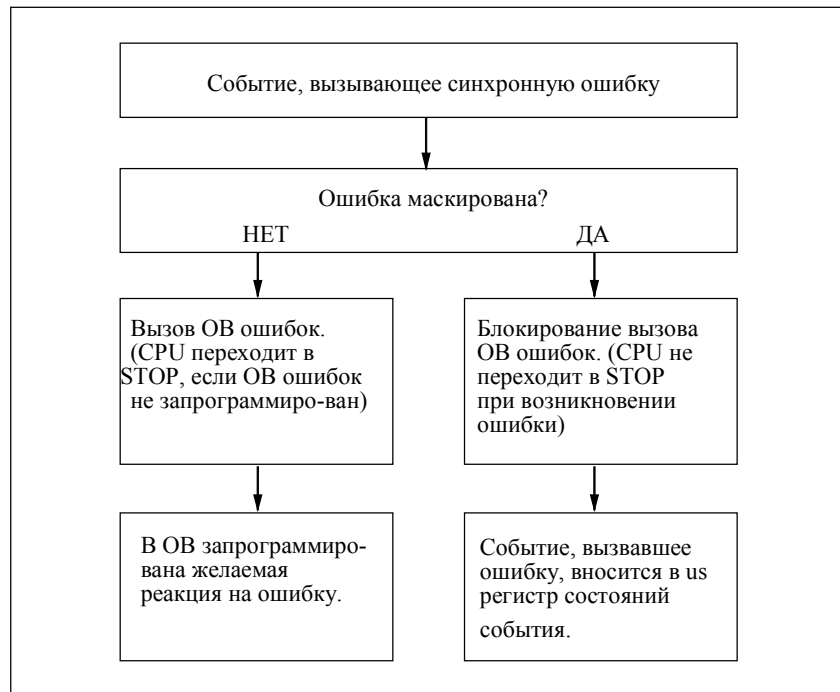


Рис. 10-1. Возможные реакции на события, вызывающие синхронные ошибки

Маска ошибок

События, вызывающие синхронные ошибки, сопоставлены определенному битовому шаблону, маске ошибок. Эту маску ошибок Вы опять найдете во входных и выходных параметрах SFC 36, 37 и 38.

В случае событий, вызывающих синхронные ошибки, мы различаем ошибки программирования и ошибки доступа, которые Вы можете маскировать в двух масках. Соответствующие маски ошибок Вы найдете на рисунках 10-2 и 10-3.

Маска ошибок программирования

На рисунке 10–2 мы показываем Вам битовый шаблон маски ошибок для ошибок программирования. Маска ошибок для ошибок программирования стоит в параметрах PRGFLT_... Объяснение ошибок программирования Вы найдете в таблицах 10–1 и 10–2.

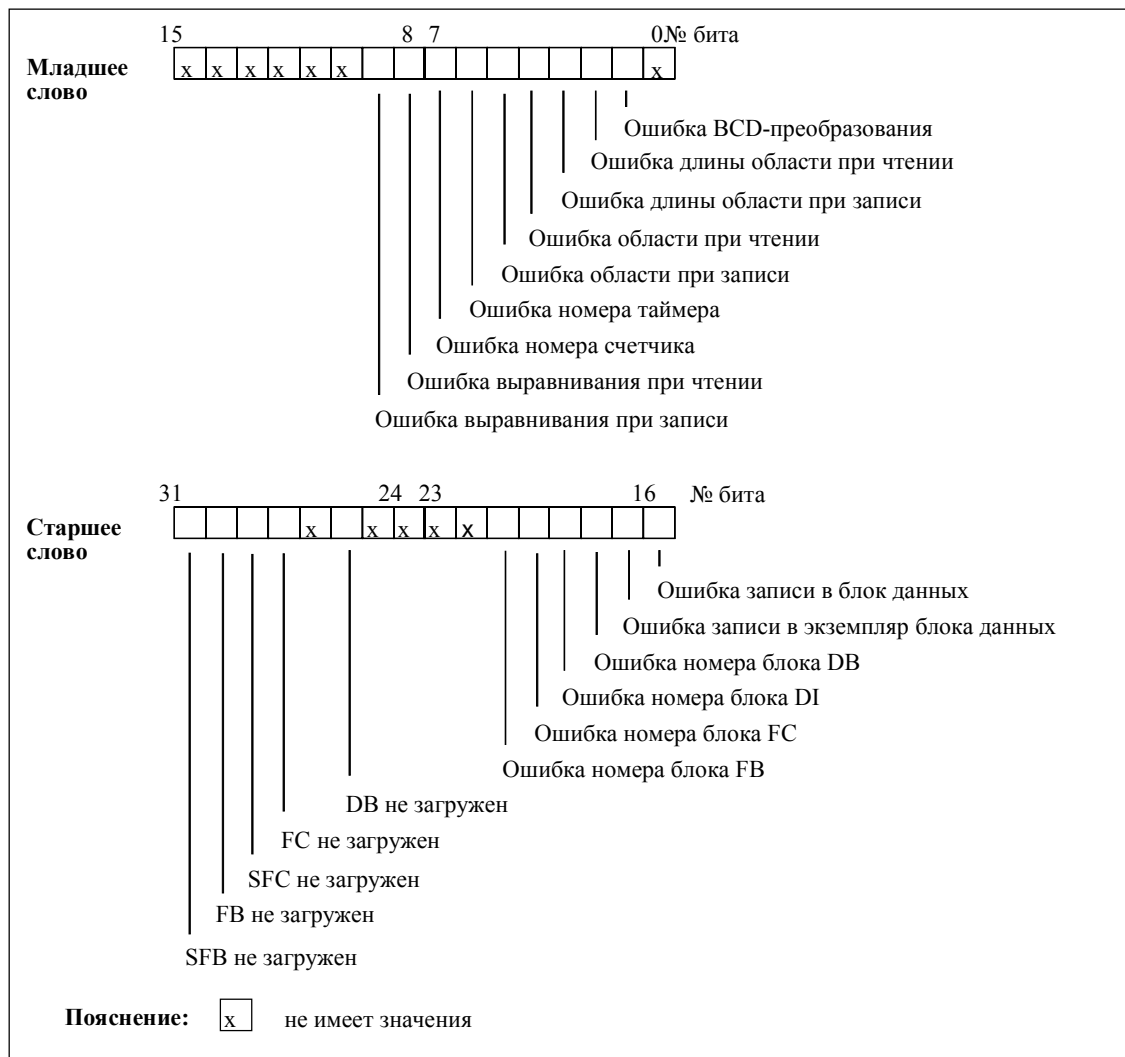


Рис. 10-2. Маска ошибок программирования

Несущественные биты

На рисунке 10–2 знак x обозначает для ...

- ... входных параметров SFC 36, 37, 38 = "0"
- ... выходных параметров SFC 36, 37 = "1" для S7–300
= "0" для S7–400
- для SFC 38 = "0"

Маска ошибок доступа

На рисунке 10–3 мы показываем Вам битовый шаблон маски ошибок для ошибок доступа. Маска ошибок для ошибок доступа стоит в параметрах ACCFLT_... Объяснение ошибок доступа Вы найдете в таблицах 10–3.

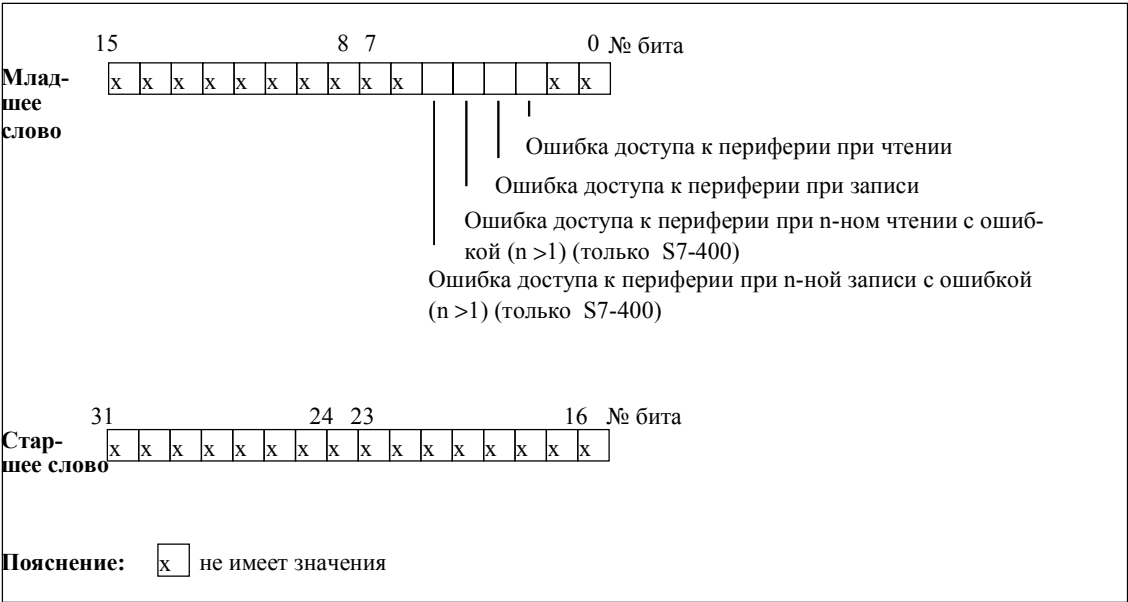


Рис. 10-3. Маска ошибок доступа

Несущественные биты

На рисунке 10–3 знак x обозначает для ...

- ... входных параметров SFC 36, 37, 38 = "0"
- ... входных параметров SFC 36, 37 = "1" для S7–300
= "0" для S7–400
- для SFC 38 = "0"

Пример

Рисунок 10–4 показывает Вам, как выглядит младшее слово маски ошибок для ошибок доступа со всеми маскированными ошибками

- как входной параметр для SFC 36
- как выходной параметр для SFC 36.

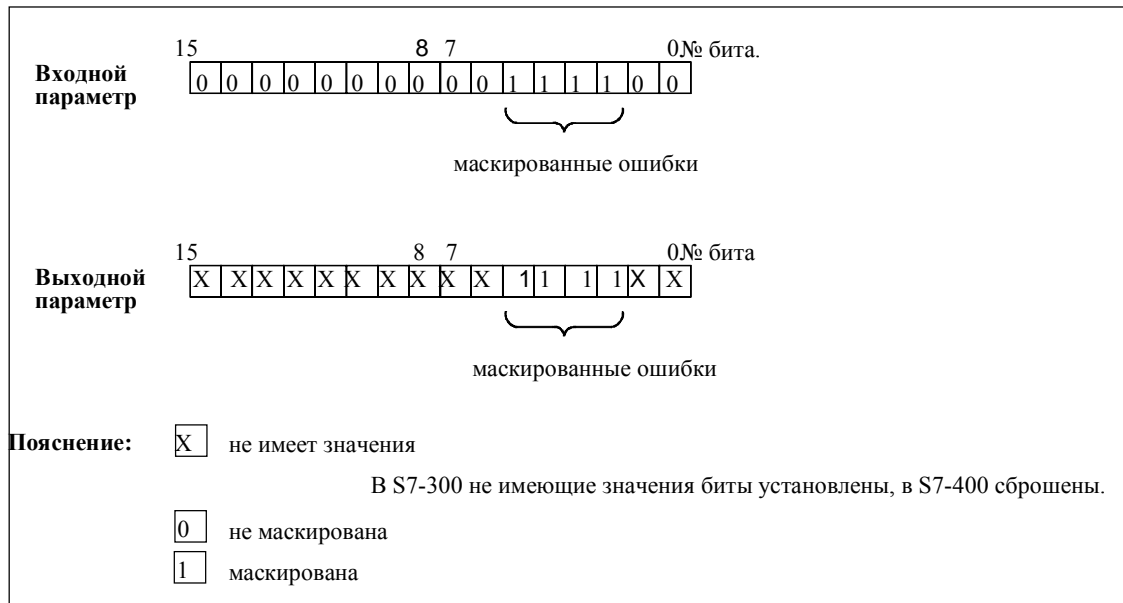


Рис. 10-4. Пример маски ошибок

**Ошибки
программирования
Младшее слово**

Таблица 10–1 содержит ошибки, которые сопоставлены младшему слову маски ошибок для ошибок программирования.
Соответствующим ошибкам сопоставлены возможные причины ошибок.

Таблица 10–1. Возможные причины ошибок программирования, младшее слово

Ошибка	ID события (W#16#...)	Ошибка появляется, ...
Ошибка BCD–преобразования	2521	... когда преобразуемое значение не является BCD–числом (например, 5E8).
Ошибка длины области при чтении	2522	... когда адресованный операнд не полностью находится внутри возможной области операндов. Пример: Должен читаться операнд MW 320, несмотря на то, что область меркеров имеет размер только 256 байтов.
Ошибка длины области при записи	2523	... когда адресованный операнд не полностью находится внутри возможной области операндов. Пример: Должно записываться значение в MW 320, несмотря на то, что область меркеров имеет размер только 256 байтов.
Ошибка области при чтении	2524	... когда при косвенной адресации, выходящей за пределы области, для операнда указывается неправильный код области. Пример: Верно: LAR1 P#E 12.0 L W[AR1, P#0.0] Неверно: LAR1 P#12.0 L W[AR1, P#0.0] При этой операции объявляется ошибка области.
Ошибка области при записи	2525	... когда при косвенной адресации, выходящей за пределы области, для операнда указывается неправильный код области. Пример: Верно: LAR1 P#E 12.0 T W[AR1, P#0.0] Неверно: LAR1 P#12.0 T W[AR1, P#0.0] При этой операции объявляется ошибка области.
Ошибка номера таймера	2526	... когда выполняется доступ к несуществующему таймеру. Пример: SI T [MW 0] с MW 0 = 129; должен запускаться таймер 129, хотя существует только 128 таймеров.
Ошибка номера счетчика	2527	... когда выполняется доступ к несуществующему счетчику. Пример: ZV Z [MW 0] с MW 0 = 600; выполняется доступ к счетчику 600, хотя существует только 512 счетчиков (CPU 416–1).
Ошибка выравнивания при чтении	2528	... когда операнд длиной в байт, слово или двойное слово адресуется битовым адресом _ 0. Пример: Верно: LAR1 P#M12.0 L B[AR1, P#0.0] Неверно: LAR1 P#M12.4 L B[AR1, P#0.0]
Ошибка выравнивания при записи	2529	... когда операнд длиной в байт, слово или двойное слово адресуется битовым адресом _ 0. Пример: Верно: LAR1 P#M12.0 T B[AR1, P#0.0] Неверно: LAR1 P#M12.4 T B[AR1, P#0.0]

**Ошибки
программирования
Старшее слово**

Таблица 10–2 содержит ошибки, которые сопоставлены старшему слову маски ошибок для ошибок программирования. Соответствующим ошибкам сопоставлены возможные причины ошибок.

Таблица 10–2. Возможные причины ошибок программирования, старшее слово

Ошибка	ID события (W#16#...)	Ошибка появляется, ...
Ошибка записи блока данных	2530	... когда блок данных, в который происходит запись, защищен от записи.
Ошибка записи экземпляра блока данных	2531	... когда экземпляр блока данных, в который происходит запись, защищен от записи.
Ошибка номера блока DB	2532	... когда открывается блок данных, номер которого больше, чем наибольший допустимый номер.
Ошибка номера блока DI	2533	... когда открывается экземпляр блока данных, номер которого больше, чем наибольший допустимый номер.
Ошибка номера блока FC	2534	... когда вызывается функция, номер которой больше, чем наибольший допустимый номер.
Ошибка номера блока FB	2535	... когда вызывается функциональный блок, номер которого больше, чем наибольший допустимый номер.
DB не загружен	253A	... когда открываемый блок данных не загружен.
FC не загружен	253C	... когда вызываемая функция не загружена.
SFC отсутствует	253D	... когда вызываемая системная функция отсутствует.
FB не загружен	253E	... когда вызываемый функциональный блок не загружен.
SFB отсутствует	253F	... когда вызываемый системный/стандартный функциональный блок отсутствует.

Ошибки доступа

Таблица 10–3 содержит ошибки, которые сопоставлены маске ошибок для ошибок доступа. Соответствующим ошибкам сопоставлены возможные причины ошибок.

Таблица 10–3. Возможные причины ошибок доступа

Ошибка	ID события (W#16#...)	в случае	Ошибка появляется, ...
Ошибка доступа к периферии при чтении	2942	S7–300	... когда адресу в периферийной области не сопоставлен сигнальный модуль или ... когда доступ к этой периферийной области не квитируется в пределах установленного контрольного времени модуля (задержка квитиования).
		S7–400	при первом ошибочном доступе для чтения (задержка квитиования).
	2944	S7–400	... когда адресу в периферийной области не сопоставлен сигнальный модуль или при n–ом ошибочном доступе для чтения ($n > 1$).
Ошибка доступа к периферии при записи	2943	S7–300	... когда адресу в периферийной области не сопоставлен сигнальный модуль или ... когда доступ к этой периферийной области не квитируется в пределах установленного контрольного времени модуля (задержка квитиования).
		S7–400	при первом ошибочном доступе для записи (задержка квитиования).
	2945	S7–400	... когда адресу в периферийной области не сопоставлен сигнальный модуль или при n–ом ошибочном доступе для записи ($n > 1$).

10.2 Маскирование событий, вызывающих синхронные ошибки, с помощью SFC 36 "MSK_FLT"

Описание

Вы используете SFC 36 "MSK_FLT" (mask synchronous faults) для того, чтобы управлять реакцией CPU на события, вызывающие синхронные ошибки. Для этого Вы маскируете соответствующие события (маски ошибок смотрите в разделе 10.1). Путем вызова SFC 36 Вы маскируете события, вызывающие синхронные ошибки, в действующем классе приоритета.

Если Вы устанавливаете в "1" отдельные биты масок синхронных ошибок во входных параметрах, то другие ранее уже установленные биты сохраняют свое значение "1". Следовательно, Вы получаете новые маски ошибок, которые Вы можете считывать через выходные параметры. Маскированные Вами события, вызывающие синхронные ошибки, не вызывают ОВ, а только записываются в регистр состояний события. Вы можете считывать регистр состояний события с помощью SFC 38 "READ_ERR" (смотрите раздел 10.4).

Параметры

Таблица 10–4. Параметры для SFC 36 "MSK_FLT"

Параметр	Объявление	Тип данных	Область памяти	Описание
PRGFLT_SET_MASK	INPUT	DWORD	E, A, M, D, L, Konst.	Маскируемые ошибки программирования
ACCFLT_SET_MASK	INPUT	DWORD	E, A, M, D, L, Konst.	Маскируемые ошибки доступа
RET_VAL	OUTPUT	INT	E, A, M, D, L	Информация об ошибках
PRGFLT_MASKED	OUTPUT	DWORD	E, A, M, D, L	Маскированные ошибки программирования
ACCFLT_MASKED	OUTPUT	DWORD	E, A, M, D, L	Маскированные ошибки доступа

Информация об ошибках

Таблица 10–5. Специфическая информация об ошибках для SFC 36 "MSK_FLT"

Код ошибки (W#16#...)	Описание
0000	Никакая из ошибок еще не была маскирована.
0001	По крайней мере, одна из ошибок была уже маскирована, несмотря на это маскируются другие ошибки.

10.3 Демаскирование событий, вызывающих синхронные ошибки, с помощью SFC 37 "DMSK_FLT"

Описание Вы используете SFC 37 "DMSK_FLT" (unmask synchronous faults) для того, чтобы демаскировать события, вызывающие ошибки, маскированные с помощью SFC 36 "MSK_FLT". Для этого Вы должны установить в "1" соответствующие биты масок ошибок во входных параметрах (маски ошибок смотрите в разделе 10.1). Посредством вызова SFC 37 Вы демаскируете соответствующие события действующего класса приоритета. Одновременно стираются опрашиваемые записи в регистре состояний события. Вы можете считывать новые маски ошибок через выходные параметры.

Параметры

Таблица 10–6. Параметры для SFC 37 "DMSK_FLT"

Параметр	Объявление	Тип данных	Область памяти	Описание
PRGFLT_RESET_MASK	INPUT	DWORD	E, A, M, D, L, Konst.	Демаскируемые ошибки программирования
ACCFLT_RESET_MASK	INPUT	DWORD	E, A, M, D, L, Konst.	Демаскируемые ошибки доступа
RET_VAL	OUTPUT	INT	E, A, M, D, L	Информация об ошибке
PRGFLT_MASKED	OUTPUT	DWORD	E, A, M, D, L	Все еще маскированные ошибки программирования
ACCFLT_MASKED	OUTPUT	DWORD	E, A, M, D, L	Все еще маскированные ошибки доступа

Информация об ошибках

Таблица 10–7. Специфическая информация об ошибках для SFC 37 "DMSK_FLT"

Код ошибки (W#16#...)	Описание
0000	Все указанные ошибки были демаскированы.
0001	По крайней мере, одна из ошибок не была маскирована, несмотря на это другие ошибки демаскируются.

10.4 Чтение регистра состояний события с помощью SFC 38 "READ_ERR"

Описание

С помощью SFC 38 "READ_ERR" (read error registers) Вы считываете регистр состояний события. Регистр состояний события по своей структуре соответствует маскам ошибок программирования и доступа, которые Вы можете программировать с помощью SFC 36 и 37 в качестве входных параметров (смотрите раздел 10.1).

Во входные параметры Вы записываете то, какие события, вызывающие синхронные ошибки, Вы хотите опрашивать из регистра состояний события. Посредством вызова SFC 38 Вы считываете желаемые записи из регистра состояний события и одновременно стираете эти записи.

Из регистра состояний события Вы берете информацию о том, какие маскированные синхронные ошибки в действующем приоритетном классе появились, по крайней мере, один раз. Установленный бит означает, что соответствующая маскированная синхронная ошибка появилась, по крайней мере, один раз.

Параметры

Таблица 10–8. Параметры для чтения регистра состояний события

Параметр	Объявление	Тип данных	Область памяти	Описание
PRGFLT_QUERY	INPUT	DWORD	E, A, M, D, L, Konst.	Опрос ошибок программирования
ACCFLT_QUERY	INPUT	DWORD	E, A, M, D, L, Konst.	Опрос ошибок доступа
RET_VAL	OUTPUT	INT	E, A, M, D, L	Информация об ошибках
PRGFLT_ESR	OUTPUT	DWORD	E, A, M, D, L	Появившаяся ошибка программирования
ACCFLT_ESR	OUTPUT	DWORD	E, A, M, D, L	Появившаяся ошибка доступа

Информация об ошибках

Таблица 10–9. Специфическая информация об ошибках для SFC 38 "READ_ERR"

Код ошибки (W#16#...)	Описание
0000	Все опрошенные ошибки маскированы.
0001	По крайней мере, одна из опрошенных ошибок не маскирована.