

3 Функции копирования и функции управления для блоков

3.1 Копирование переменных с помощью SFC20 "BLKMOV"

Описание

SFC 20 "BLKMOV" (block move [перемещение блока]) используется для копирования содержимого области памяти (= исходной области) в другую область памяти (= целевую область).

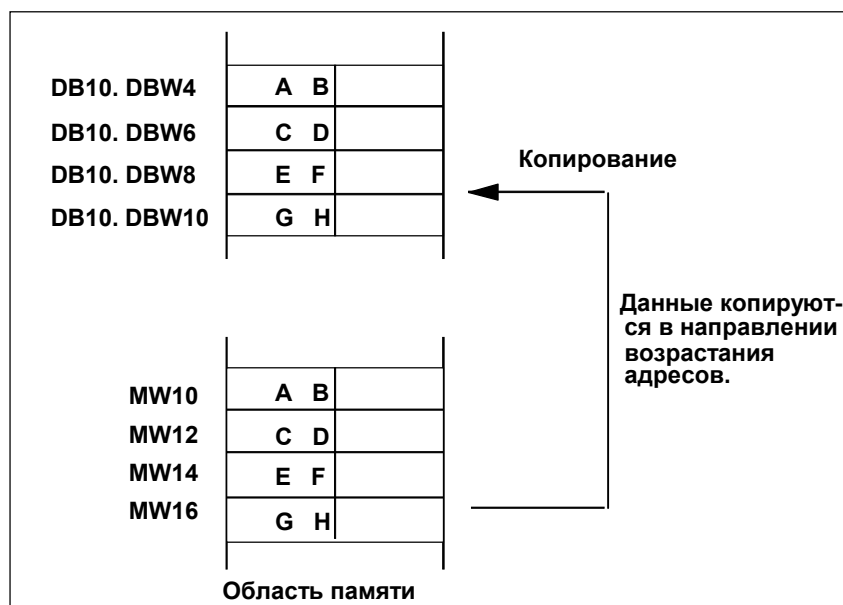
С помощью SFC20 "BLKMOV" можно копировать следующие области памяти:

- части DB-блоков (data blocks)
- области меркеров (memory bits)
- область образа процесса (part process image) для входов
- область образа процесса (part process image) для выходов.

Исходные параметры могут быть включены в загружаемую память (область памяти блока, который не имеет значения для исполнения программы) (DB, скомпилированного с ключевым словом UNLINKED).

Примечание

Если в Вашем CPU присутствует функция SFC 83, то Вы можете использовать ее для считывания блоков данных, которые не имеют значения для исполнения программы. Если для этого Вы будете использовать функцию SFC 20, то система выдаст ошибку W#16#8092.



Возможность прерывания

Пока исходная область не является частью блока данных, который существует только в загрузочной памяти, нет ограничений на глубину вложения.

Тем не менее, если SFC 20 прерывается при копировании из DB, не влияющего на выполнение программы, то исполнение SFC 20 более не может быть вложенным.

Параметр	Описание	Тип данных	Область памяти	Характеристика
SRCBLK	INPUT	ANY	I, Q, M, D, L	Определяет область памяти, которая должна копироваться (исходная область). Массивы с типом данных STRING, недопустимы.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время исполнения функции происходит ошибка, то возвращаемое значение содержит код ошибки.
DSTBLK	OUTPUT	ANY	I, Q, M, D, L	Определяет область памяти, в которую будут копироваться данные (целевая область). Массивы с типом данных STRING, недопустимы.

Примечание

Исходная и целевая области не должны перекрываться. Если заданная целевая область больше, чем исходная, то функция копирует в целевую область лишь столько данных, сколько содержится в исходной области.

Если заданная целевая область (параметр DSTBLK) меньше, чем исходная (параметр SRCBLK), то функция копирует лишь столько данных, сколько может быть записано в целевую область.

Если целевая область или исходная области фактически меньше, чем указано в соответствующих параметрах для них (параметры DSTBLK, SRCBLK), то данные не будут пересылаться.

Если указатель ANY (источника или адресата) имеет тип данных BOOL, то заданная длина должна делиться на 8; в противном случае SFC не будет выполняться.

Параметры для исходной или целевой областей (или для обеих областей) должны иметь строковый тип данных (STRING). Если для исходной области задан тип String, то только фактические символы строки будут скопированы. Если для исходной и целевой областей задан тип String, то будет использована фактическая длина скопированных символов будет использована при записи данных.

Если необходимо скопировать строку включающую максимальную длину и фактическую длину, выполните следующие действия: Создайте собственный указатель ANY для использования с параметрами DSTBLK и SRCBLK. Для этого используйте тип данных BYTE.

Особенности:

Если несвязанный блок данных копируется в ОЗУ с помощью SFC 20 BLKMOV и одновременно загружается, например, посредством PG, то выполнение функции SFC может быть задержано на несколько миллисекунд. Это приводит к увеличению длительности цикла OB, и может нарушать текущий цикл мониторинга. Избегайте загрузки блока в CPU во время копирования блока с помощью SFC 20.

Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Нет ошибок
8091	Превышена глубина вложения
8092	SFC20 "BLKMOV" не может быть выполнен из-за обращения к невыполняемому блоку. Используйте для операции перемещения SFC 83.
8хуу	Информация об общих ошибках: см. раздел "Проверка ошибок с выходным параметром RET_VAL"

3.2 Непрерываемое копирование переменных с помощью SFC 81 "UBLKMOV"

Описание

С помощью SFC 81 "UBLKMOV" (uninterruptable block move [непрерываемое перемещение блоков]) Вы можете копировать содержимое одной области памяти (=исходная область) в другую область памяти (= целевую область). Эта операция копирования не может прерываться другими действиями операционной системы.

Используя SFC 81 "UBLKMOV", Вы можете копировать все области памяти, кроме:

- следующих типов блоков: FB, SFB, FC, SFC, OB, SDB
- счетчиков
- таймеров
- областей памяти периферийных входов/выходов

Исходные параметры могут также быть включены в загружаемую область памяти несвязанных блоков данных (скомпилированных с опцией UNLINKED)!

Возможность прерывания, время реакции на прерывание

Копирование не может прерываться. Помните, что, если Вы используете SFC 81 "UBLKMOV", то это может увеличить время реакции Вашего CPU на прерывание.

Параметр	Описание	Тип	Область памяти	Характеристика
SRCBLK	INPUT	ANY	I, Q, M, D, L	Определяет копируемую область памяти (исходную область). Массивы типа STRING недопустимы.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении функции происходит ошибка, то возвращаемое значение содержит код ошибки.
DSTBLK	OUTPUT	ANY	I, Q, M, D, L	Определяет область памяти, в которую будут копироваться данные (целевая область). Массивы типа STRING недопустимы.

Примечание

Исходная и целевая области не должны перекрываться. Если заданная целевая область больше, чем исходная, то функция копирует в целевую область лишь столько данных, сколько содержится в исходной области.

Если заданная целевая область (параметр DSTBLK) меньше, чем исходная (параметр SRCBLK), то функция копирует лишь столько данных, сколько может быть записано в целевую область.

Если целевая область или исходная области фактически меньше, чем указано в соответствующих параметрах для них (параметры DSTBLK, SRCBLK), то данные не будут пересылаться.

Если указатель ANY (источника или адресата) имеет тип данных BOOL, то заданная длина должна делиться на 8; в противном случае SFC не будет выполняться.

Если указатель ANY имеет тип STRING, то заданная длина должна быть равна 1.

Информация об ошибках

Код ошибки (W#16#...)	Характеристика
0000	Ошибки нет
8091	Исходная область находится в несвязанном блоке данных
8092	Общая информация об ошибках (см. пункт Проверка ошибок с выходным параметром RET_VAL)

3.3 Инициализация области памяти с помощью SFC 21 "FILL"

Описание

С помощью SFC 21 "FILL" Вы можете предварительно заполнить область памяти (целевую область) содержимым другой области памяти (исходной области). SFC копирует содержимое в заданную целевую область, пока эта область памяти не будет полностью записана.

Примечание

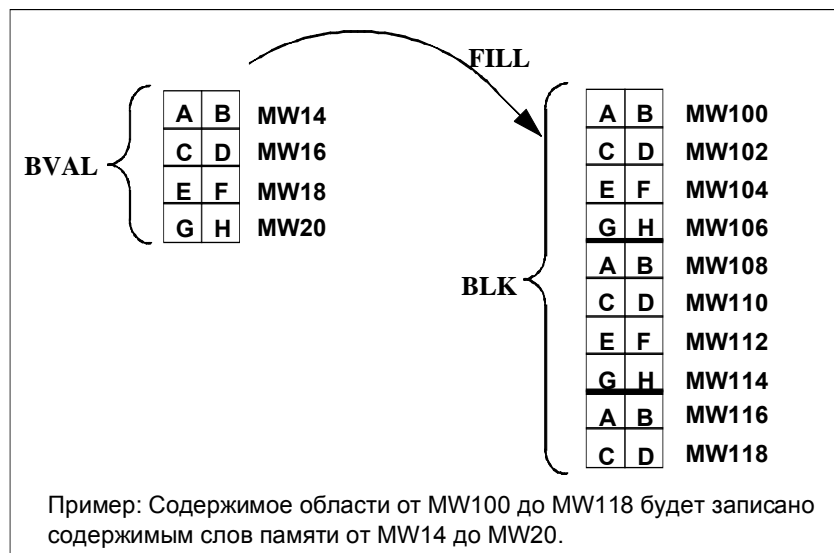
Исходный и целевой массив не должны перекрываться.

Если целевая область, подлежащая инициализации, не является целым кратным длины входного параметра BVAL, то она, тем не менее, записывается до последнего байта.

Если целевая область, подлежащая инициализации, меньше, чем исходная область, то копируется лишь столько данных, сколько может быть записано в целевую область.

Если целевая область или исходная области фактически меньше, чем указано в соответствующих параметрах для них (параметры BVAL, BLK), то данные не будут пересылаться.

Если указатель ANY (источника или адресата) имеет тип данных BOOL, то заданная длина должна делиться на 8; в противном случае SFC не будет выполняться.



Исключения

Следующие области памяти не могут использоваться как исходные:

- область счетчиков
- область таймеров

С помощью SFC 21 нельзя записывать значения в:

- следующие типы блоков: FB, SFB, FC, SFC, SDB
- счетчики
- таймеры
- области памяти периферийных входов/выходов

Параметр	Описание	Тип данных	Область памяти	Характеристика
BVAL	INPUT	ANY	I, Q, M, D, L	Параметр BVAL содержит значение или описание области, содержимое которой будет использовано для инициализации целевой области (исходная область). Массивы, относящиеся к типу данных STRING, недопустимы.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время обработки функции появляется ошибка, то возвращаемое значение содержит код ошибки.
BLK	OUTPUT	ANY	I, Q, M, D, L	Параметр BLK содержит описание области, подлежащей инициализации (целевой области). Массивы, относящиеся к типу данных STRING, недопустимы.

Входной параметр имеет тип структуры (Structure)

Если в качестве входного параметра передается структура, то помните следующую особенность: STEP 7 всегда определяет длину структуры как четное число байтов. В результате структуре потребуется один байт дополнительного пространства в памяти, если Вы опишете структуру с нечетным числом байтов.

Пример

Структура описана следующим образом:

```
TYP_5_BYTE_STRUCTURE : STRUCT
```

```
    BYTE_1_2 : WORD
```

```
    BYTE_3_4 : WORD
```

```
    BYTE_5 : BYTE
```

```
END_STRUCT
```

Описанная структура "TYP_5_BYTE_STRUCTURE" требует 6 байтов памяти.

Информация об ошибках

SFC 21 "FILL" выдает только информацию об общих (неспецифических) ошибках. См. раздел Проверка ошибок с выходным параметром RET_VAL

3.4 Создание блока данных с помощью SFC 22 "CREAT_DB"

Описание

С помощью SFC 22 "CREAT_DB" (create data block [создать блок данных]), Вы можете создать блок данных, который не содержит предварительно установленных значений. Первоначально в новом блоке содержатся случайные значения. SFC 22 создает блок данных выбираемой длины с номером, взятым из заданного диапазона. SFC присваивает DB минимально возможный номер из заданного диапазона. Если Вы хотите создать DB с конкретным номером, то просто выберите диапазон, указывая одно и то же значение для верхней и нижней границы. Вы не можете назначить номер, если DB с таким же номером уже существует в программе пользователя. Длина DB должна быть определена четным числом байтов.

Возможность прерывания

SFC22 "CREAT_DB" может прерываться более приоритетными OB. Если SFC22 "CREAT_DB" вновь вызывается в OB более высокого приоритета, то этот вызов отвергается с кодом ошибки W#16#8091.

Параметр	Описание	Тип данных	Область памяти	Характеристика
LOW_LIMIT	INPUT	WORD	I, Q, M, D, L, константа	Нижнее граничное значение является минимальным номером в диапазоне номеров, которые Вы можете присвоить своему блоку данных.
UP_LIMIT	INPUT	WORD	I, Q, M, D, L, константа	Верхнее граничное значение является максимальным номером в диапазоне номеров, которые Вы можете присвоить своему блоку данных.
COUNT	INPUT	WORD	I, Q, M, D, L, константа	Числовое значение задает количество байтов данных, которое Вы хотите зарезервировать для своего блока данных. Здесь Вы должны задать четное число байтов (максимум 65534).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время обработки функции появляется ошибка, то возвращаемое значение содержит код ошибки.
DB_NUMBER	OUTPUT	WORD	I, Q, M, D, L	Номер блока данных является номером созданного блока данных. Если происходит ошибка (бит 15 в RET_VAL был установлен), то в DB_NUMBER вводится значение 0.

Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Нет ошибки.
8091	Вы выполнили вложенный вызов SFC22.
8092	<p>Функция "Создать DB" не может быть в данный момент выполнена, так как:</p> <ul style="list-style-type: none"> • в данный момент активна функция "Сжатие памяти пользователя" • в CPU уже находится максимально возможное количество DB-блоков • Н CPU выполняет функции соединения или обновления. • программа из ПО WinAC Software CPU обнаружила ошибку в ОС компьютера, на котором установлена система WinAC.
80A1	<p>Ошибка в номере DB:</p> <ul style="list-style-type: none"> • Номер равен 0. • Номер превышает количество DB для конкретного CPU. • Значение параметра LOW_LIMIT [нижняя граница] больше значения UP_LIMIT [верхняя граница].
80A2	<p>Ошибка в длине DB:</p> <ul style="list-style-type: none"> • Длина равна 0. • Длина была задана нечетным числом. • Длина больше, чем допускает CPU.
80B1	Нет свободных номеров DB.
80B2	Недостаточно свободной памяти.
80B3	Недостаточно непрерывной памяти (рекомендация: необходимо выполнить сжатие!)

3.5 Удаление блока данных с помощью SFC23 "DEL_DB"

Описание

С помощью SFC 23 "DEL_DB" (delete data block [удалить блок данных]) Вы можете удалить блок данных, находящийся в рабочей памяти, и, если имеется, в загрузочной памяти CPU. Удаляемый DB не должен быть открыт ни в текущем, ни в каком-либо более низком классе приоритета, иными словами, он не должен быть внесен ни в один из двух регистров DB, ни в В-стек. В противном случае CPU при вызове SFC 23 запускает OB 121. Если OB 121 не загружен, то CPU переходит в состояние STOP. В системах S7-300 (за исключением CPU 318) DB-блок удаляется без вызова OB 121.

Примечание

Не рекомендуется удалять экземплярные блоки DB (instance DB) с использованием функции SFC 23 "DEL_DB", так как это всегда приводит к программным ошибкам. Избегайте удалять экземплярные блоки DB (instance DB) с использованием функции SFC 23!

В следующей таблице поясняется, когда DB может быть удален с помощью SFC23 "DEL_DB".

Если...	то...
DB был создан путем вызова SFC22 "CREAT_DB",	SFC23 может его удалить.
DB был передан в CPU посредством STEP 7 и не был создан с ключевым словом UNLINKED,	SFC23 может его удалить.
DB находится на флэш-карте,	SFC23 не может его удалить.

Возможность прерывания

SFC23 "DEL_DB" может быть прерван более высокими классами приоритета. Если этот SFC вызывается там вновь, то этот второй вызов игнорируется и в RET_VAL записывается W#16#8091.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DB_NUMBER	INPUT	WORD	I, Q, M, D, L, константа	Номер удаляемого DB
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке

Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8091	Вызовы SFC23 были вложенными, и была превышена максимальная глубина вложения для используемого CPU.
8092	Функция "Удаление DB" в данный момент не может быть выполнена, так как <ul style="list-style-type: none"> • в данный момент активна функция "Сжатие памяти пользователя" • в данный момент производится операция копирования DB, который необходимо уничтожить, из CPU в offline-проект • Н CPU выполняет функции соединения или обновления. • программа из ПО WinAC Software CPU обнаружила ошибку в ОС компьютера, в котором инсталлирована система WinAC.
80A1	Ошибка во входном параметре DB_NUMBER: выбранный фактический параметр <ul style="list-style-type: none"> • имеет значение 0 • больше, чем максимально допустимый номер DB для используемого CPU.
80B1	DB с заданным номером в CPU не существует.
80B2	DB с заданным номером был создан с ключевым словом UNLINKED.
80B3	DB находится на флэш-карте.
80B4	DB не может быть удален, так как <ul style="list-style-type: none"> • является блоком F-программы • является экземплярным блоком (instance DB) блока для S7-коммуникаций (только в системах S7-400). • является технологическим блоком (technology DB).
80C1	Функция уничтожения DB "Delete DB" не может быть выполнена в настоящее время из-за временного критического состояния параметров.
8хуу	Информация об общих ошибках: см. раздел "Проверка ошибок с выходным параметром RET_VAL"

3.6 Тестирование блока данных с помощью SFC 24 "TEST_DB"

Описание

С помощью функции SFC 24 "TEST_DB" (test data block [тестировать блок данных]) Вы получаете информацию о блоке данных, находящемся в рабочей памяти CPU. SFC запрашивает сведения о количестве байтов данных в выбранном DB и проверяет, защищен ли DB от записи.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DB_NUMBER	INPUT	WORD	I, Q, M, D, L, константа	Номер тестируемого DB
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках
DB_LENGTH	OUTPUT	WORD	I, Q, M, D, L	Количество байтов данных, которое содержит выбранный DB.
WRITE_PROT	OUTPUT	BOOL	I, Q, M, D, L	Информация о признаке защиты выбранного DB от записи (1 означает, что он защищен от записи).

Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
80A1	Ошибка во входном параметре DB_NUMBER: выбранный фактический параметр <ul style="list-style-type: none"> • имеет значение 0 • больше, чем максимально допустимый номер DB для используемого CPU.
80B1	DB с заданным номером в CPU не существует.
80B2	DB был создан с ключевым словом UNLINKED.
8хуу	Информация об общих ошибках: см. раздел "Проверка ошибок с выходным параметром RET_VAL"

3.7 Сжатие памяти пользователя с помощью SFC 25 "COMPRESS"

Пропуски в памяти

Пропуски могут возникать как в рабочей, так и в загрузочной памяти вследствие многократного удаления и повторной загрузки блоков. Эти пропуски сокращают эффективную область памяти.

Описание

С помощью SFC 25 "COMPRESS" Вы можете запустить процедуру сжатия раздела ОЗУ как загрузочной, так и рабочей памяти. Функция сжатия здесь та же самая, что и при внешнем запуске в режиме RUN-P (положение переключателя режимов работы).

Если процедура сжатия была запущена извне и все еще активна, то вызов SFC 25 приведет к появлению сообщения об ошибке.

Примечание

Блоки данных, имеющие длину более 1000 байтов, не сдвигаются функцией SFC 25 "COMPRESS". Это значит, что после сжатия в рабочей памяти могут остаться пропуски.

Параметр	Описание	Тип данных	Область памяти	Характеристика
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке
BUSY	OUTPUT	BOOL	I, Q, M, D, L	Показывает, активна ли еще функция сжатия, запущенная вызовом SFC25. (1 означает, что активна)
DONE	OUTPUT	BOOL	I, Q, M, D, L	Показывает, успешно ли завершился процесс сжатия, который был запущен посредством SFC 25 (1 означает "успешно").

Проверка функции сжатия

При однократном вызове SFC 25 "COMPRESS" запускается функция сжатия; однако Вы не можете проверить, успешно ли выполнилось сжатие.

Если Вы хотите проверить функцию сжатия, выполните описанные ниже шаги:

Вызовите SFC 25 циклически. После каждого вызова сначала оценивайте параметр RET_VAL. Если его значение равно 0, можно оценить параметры BUSY и DONE. Если BUSY = 1, а DONE = 0, то это указывает, что функция сжатия еще активна. Когда BUSY примет значение 0, а DONE - значение 1, то это указывает, что функция сжатия успешно завершена. Если потом вновь вызвать SFC 25, то вновь запустится сжатие.

Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было. Функция сжатия была запущена SFC 25. Только в этом случае имеет смысл оценивание в программе пользователя выходных параметров BUSY и DONE (см. выше).
8091	Функция сжатия была запущена извне и еще активна.
8092	Функция "Сжатие памяти пользователя" в данный момент не может быть выполнена, так как <ul style="list-style-type: none"> была запущена извне в среде STEP 7 и все еще активна функция "Удаление блока данных" функция тестирования и запуска в настоящее время требует конкретного блока (например, статус) была запущена извне и все еще активна функция "Копирование блоков". основной CPU выполняет функции соединения или обновления.
8хуу	Информация об общих ошибках: см. раздел "Проверка ошибок с выходным параметром RET_VAL"

3.8 Передача заменяющего значения в аккумулятор 1 с помощью SFC 44 "REPL_VAL"

Описание

С помощью SFC 44 "REPL_VAL" (replace value [заменить значение]) Вы передаете значение в аккумулятор 1 класса приоритета, вызвавшего ошибку.

Ограничение на вызов функции: только в ОБ обработки синхронных ошибок

Вы можете вызывать SFC 44 "REPL_VAL" только в ОБ синхронных ошибок (ОВ 121, ОВ 122).

Пример применения

Если модуль ввода поврежден до такой степени, что из него более нельзя считывать значения, то при каждом обращении к этому модулю запускается ОБ 122. В ОБ 122 с помощью SFC 44 "REPL_VAL" в аккумулятор 1 прерванного класса приоритета может быть передано подходящее значение, так что программа может продолжать работу с этим заменяющим значением. Информация, нужная для выбора заменяющего значения (например, блок, в котором встретилась ошибка, или поврежденный операнд), находится локальных переменных ОБ 122.

Параметр	Описание	Тип данных	Область памяти	Характеристика
VAL	INPUT	DWORD	I, Q, M, D, L, константа	Заменяющее значение
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время обработки функции встретилась ошибка, то возвращаемое значение содержит код ошибки.

Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибок не было. Было введено заменяющее значение.
8080	SFC 44 не вызывался ни одним из ОБ синхронных ошибок (ОВ 121, ОВ 122).
8хуу	Информация об общих ошибках: см. раздел "Проверка ошибок с выходным параметром RET_VAL"

3.9 Генерация блоков данных в загружаемой памяти с помощью SFC 82 "CREA_DBL"

Описание

С помощью SFC 82 "CREA_DBL" (create data block in load memory [создание DB в загружаемой памяти]) Вы можете создать новый блок данных в загружаемой памяти (Micro Memory Card – карта памяти). SFC 82 генерирует DB с размером, заданным по умолчанию, и с номером из определенного ряда. Функция SFC 82 назначает блоку наименьший из возможных номеров. Вы можете задать для DB и определенный номер - путем назначения данного номера для верхнего и нижнего граничных значений для вышеуказанного специального ряда номеров. Невозможно задать уже используемый номер DB. Если уже существует блок с данным номером в рабочей памяти и/или загружаемой памяти или DB существует как копия, то SFC завершает работу и выдается сообщение об ошибке.

Примечание

С помощью SFC 24 "TEST_DB" Вы можете определить, существует ли уже DB с заданным номером.

Содержание области данных, на которую указывает параметр SRCBLK (исходный блок), записывается в DB. Эта область данных должна быть DB или областью из DB. Чтобы обеспечить консистентность данных, Вы не должны изменять эту область данных, во время работы SFC 82 (то есть, пока параметр BUSY имеет значение TRUE (ИСТИНА)).

С помощью SFC 82 может быть создан и инициализирован DB-блок с атрибутом READ_ONLY ("только чтение").

SFC 82 не изменяет контрольную сумму программы пользователя.

Принцип работы

SFC 82 "CREA_DBL" работает асинхронно, т.е. время ее работы может перекрывать несколько вызовов функции. Запуск функции SFC 82 осуществляется с REQ =1.

Состояние работы функции можно контролировать с помощью выходных параметров RET_VAL и BUSY.

Также для асинхронно работающих функций контролируйте значения REQ, RET_VAL и BUSY.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	REQ = 1: Запрос на генерацию блока данных
LOW_LIMIT	INPUT	WORD	I, Q, M, D, L	Наименьший номер в последовательности номеров из ряда, указанного для назначения вновь создаваемым DB
UP_LIMIT	INPUT	WORD	I, Q, M, D, L	Наибольший номер в последовательности номеров из ряда, указанного для назначения вновь создаваемым DB
COUNT	INPUT	WORD	I, Q, M, D, L	Значение, определяющее количество байтов данных, которые резервируются для DB. Здесь необходимо определить четное число байтов.
ATTRIB	INPUT	BYTE	I, Q, M, D, L	Характеристики DB: Бит 0 = 1: UNLINKED: (несвязанный): DB существует только в загружаемой памяти. Бит 1 = 1: READ_ONLY: (только чтение): DB защищен от записи. Бит 2 = 1: NON_RETAIN: (перманентность): DB перманентный. Бит 3 ... 7: Резерв
SRCBLK	INPUT	ANY	D	Указатель на область данных, в которой DB инициализирован
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Код ошибки (возвращаемое значение)
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: Процесс не завершен
DB_NUM	OUTPUT	WORD	I, Q, M, D, L	Номер созданного DB

Информация об ошибках

Код ошибки (W#16#....)	Описание
0000	Ошибок не было
0081	Область назначения (целевая) больше области исходной (источника). Исходная информация полностью записывается в область назначения. Свободные байты области назначения заполняются нулями.
7000	Первый вызов с REQ=0: передача данных не активирована; параметр BUSY имеет значение 0.
7001	Первый вызов с REQ=1: передача данных не активирована; параметр BUSY имеет значение 1.

Код ошибки (W#16#....)	Описание
7002	Промежуточное обращение (значение REQ - любое): передача данных уже активирована; BUSY имеет значение 1.
8081	Область назначения (целевая) меньше области исходной (источника). Область назначения полностью заполняется. Непоместившаяся в ней часть исходной информации игнорируется.
8091	Имеются вложенные вызовы функции SFC 82
8092	Генерация DB не выполняется, т.к.: <ul style="list-style-type: none"> активирован процесс сжатия памяти приложения ("Compress Application Memory") для Вашего CPU уже достигнуто максимальное количество блоков
8093	Для параметра SCRBLK нет блока данных DB или блок данных находится вне рабочей памяти
8094	Еще не поддерживается атрибут, который был определен для параметра ATTRIB
80A1	Ошибка нумерации DB: <ul style="list-style-type: none"> номер блока равен 0 минимальный номер для диапазона номеров больше максимального
80A2	Ошибка размера DB: <ul style="list-style-type: none"> длина блока равна 0 длина блока равна нечетному числу размер блока больше максимально допустимого для CPU
80B1	Нет свободных номеров для DB
80B2	Недостаточно свободной памяти
80BB	Недостаточно загружаемой памяти
80C0	Область назначения используется в настоящее время другой функцией SFC или функцией связи
80C3	Уже достигнуто максимальное количество одновременно работающих функций SFC 82
8хуу	Общая ошибка программы, например: <ul style="list-style-type: none"> исходный DB не существует или доступен только в виде копии исходная область в DB не существует См. раздел Проверка ошибок с выходным параметром RET_VAL

3.10 Считывание информации из блока данных в загружаемой памяти с помощью SFC 83 "READ_DBL"

Описание

С помощью SFC 83 "READ_DBL" (read date block in load memory [чтение DB в загружаемой памяти]) Вы можете копировать DB или область в DB типа в загружаемой памяти (Micro Memory Card – карта памяти) в область данных DB назначения. DB назначения (целевой DB) должен быть назначен выполняемому блок, т.е., не должен быть создан с атрибутом UNLINKED. Содержимое загружаемой памяти не изменяется в процессе выполнения операции чтения.

Для обеспечения консистентности данных нельзя изменять область назначения во время работы SFC 83 (то есть, пока параметр BUSY имеет значение TRUE (ИСТИНА)).

На параметры SRCBLK и DSTBLK накладываются следующие ограничения:

- Для указателя ANY типа BOOL длина должна делиться на 8.
- Если указатель ANY имеет тип STRING, то заданная длина должна быть равна 1.

При необходимости Вы можете определить размер исходной области с помощью функции SFC 24 "TEST_DB".

Примечание

Функция SFC 83 может работать асинхронно, поэтому следует избегать частого (или циклического) считывания переменных из загружаемой памяти.

Примечание

Будучи запущенной, функция SFC 83 всегда должна без прерывания завершить работу. Если достигнуто максимальное количество одновременно работающих функций SFC 83, и при этом вновь вызывается SFC 83 с высшим приоритетом, то вызов игнорируется и выдается сообщение с кодом ошибки W#16#80C3.

Принцип работы

Функция SFC 83 предназначена для асинхронной работы, т.е. время ее работы может перекрываться с несколькими вызовами этой же функции. Запуск функции SFC 83 осуществляется по условию REQ = 1.

Состояние работы функции можно контролировать с помощью выходных параметров RET_VAL и BUSY.

Также для асинхронно работающих функций контролируйте значения REQ, RET_VAL и BUSY.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	REQ = 1: Запрос на чтение блока данных
SRCBLK	INPUT	ANY	D	Указатель на область данных DB в загружаемой области, из которой производится чтение
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Код ошибки
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: (состояние работы): Процесс не завершен
DSTBLK	OUTPUT	ANY	D	Указатель на область данных целевого DB

Информация об ошибках

Код ошибки (W#16#....)	Описание
0000	Ошибок не было
8081	Область назначения (целевая) меньше области исходной (источника). Область назначения полностью заполняется. Непоместившаяся в ней часть исходной информации игнорируется.
7000	Первый вызов с REQ=0: передача данных не активирована; параметр BUSY имеет значение 0.
7001	Первый вызов с REQ=1: передача данных не активирована; параметр BUSY имеет значение 1.
7002	Промежуточное обращение (REQ - любое значение): передача данных уже активирована; BUSY имеет значение 1.
0081	Область назначения (целевая) больше области исходной (источника). Исходная информация полностью записывается в область назначения. Значения свободных байтов области назначения не изменяются.
8093	Для параметра DSTBLK нет блока данных или блок данных отсутствует в рабочей памяти.
80B1	Для параметра SRCBLK нет блока данных или блок данных показывает, что нет объекта в загружаемой памяти (например, DB сгенерированный SFC 22)
80B4	DB с F-атрибутом не может быть считан
80C0	Область назначения используется в настоящее время другой функцией SFC или функцией связи
80C3	Уже достигнуто максимальное количество одновременно работающих функций SFC 83
8хуу	Общая ошибка. См. пункт Проверка ошибок с выходным параметром RET_VAL

3.11 Запись блока данных в загружаемой памяти с помощью SFC 84 "WRIT_DBL"

Описание

С помощью SFC 84 "WRIT_DBL" (write date block in load memory [запись DB в загружаемой памяти]) Вы можете передать содержимое DB или данные из исходного DB в рабочей памяти в DB или в область DB в загружаемой памяти (Micro Memory Card – карта памяти). Исходный DB должен быть связанным (выполняемым), т.е. с параметром UNLINKED = 0. Тем не менее, исходный DB может быть сгенерирован с помощью функции SCF 22 "CREAT_DB".

Для обеспечения консистентности данных нельзя изменять область исходных данных во время работы SFC 84 (то есть, пока параметр BUSY имеет значение TRUE (ИСТИНА)).

На параметры SRCBLK и DSTBLK накладываются следующие ограничения:

- Для указателя ANY типа BOOL длина должна делиться на 8.
- Если указатель ANY имеет тип STRING, то заданная длина должна быть равна 1.

При необходимости Вы можете определить размер исходной области с помощью функции SFC 24 "TEST_DB".

SFC 84 не изменяет контрольной суммы пользовательской программы, если Вы описали DB-блок, созданный с помощью SFC. Тем не менее, при первой записи в загруженный DB изменяется контрольная сумма пользовательской программы.

Примечание

Функция SFC 84 не годится для частой (или циклической) записи переменных в загружаемую память, потому что технология модулей памяти Micro Memory Cards обеспечивает лишь некоторое конечное число процедур записи в эти модули. Для получения более подробной информации см. справочное руководство *"SIMATIC S7-300 Programmable Controller CPU Data: CPU 31xC and CPU 31x"* ("Данные центральных процессоров для систем SIMATIC S7-300: CPU 31xC и CPU 31x").

Принцип работы

SFC 84 "WRITE_DBL" предназначена для асинхронной работы, т.е. во время ее работы могут производиться другие вызовы функции SFC 84. Запуск функции SFC 84 осуществляется при условии REQ = 1.

Состояние работы функции можно контролировать с помощью выходных параметров RET_VAL и BUSY.

Также для асинхронно работающих функций контролируйте значения REQ, RET_VAL и BUSY.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	REQ = 1: Запрос на запись
SRCBLK	INPUT	ANY	D	Указатель на область данных DB, из которой производится чтение
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Код ошибки
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: (состояние работы): Процесс не завершен
DSTBLK	OUTPUT	ANY	D	Указатель на область данных целевого DB в загружаемой памяти

Информация об ошибках

Код ошибки (W#16#....)	Описание
0000	Ошибок не было
0081	Область назначения (целевая) больше области исходной (источника). Исходная информация полностью записывается в область назначения. Значения свободных байтов области назначения не изменяются.
7000	Первый вызов с REQ=0: передача данных не активирована; параметр BUSY имеет значение 0.
7001	Первый вызов с REQ=1: передача данных не активирована; параметр BUSY имеет значение 1.
7002	Промежуточное обращение (REQ не имеет значения): передача данных уже активирована; BUSY имеет значение 1.
8081	Область назначения (целевая) меньше области исходной (источника). Область назначения полностью заполняется. Непоместившаяся в ней часть исходной информации игнорируется.
8092	Недопустимый рабочий режим: Во время работы функции SFC 84 CPU переходит в режим STOP. Код ошибки возвращается при последующем переходе к режиму выполнения (RUN). SFC 84 вызывается вновь.
8093	Для параметра SRCBLK нет блока или нет соответствующего выполняемого блока (например, DB сгенерированный SFC 22)
80B1	Для параметра DSTBLK нет блока или нет соответствующего выполняемого блока (например, DB сгенерированный SFC 22)
80B4	DB с F-атрибутом не может быть изменен
80C0	Область назначения используется в настоящее время другой функцией SFC или функцией связи. Пример: Вы выгружаете DB из CPU в программатор PG. Вы хотите изменить содержимое DB посредством SFC 84.
80C3	Уже достигнуто максимальное количество одновременно работающих функций SFC 84.
8хуу	Общая ошибка. См. пункт Проверка ошибок с выходным параметром RET_VAL

3.12 Создание блока данных с помощью SFC 85 "CREA_DB"

Описание

С помощью SFC 85 "CREA_DB" (создание блока данных) Вы можете создавать блок данных без значений по умолчанию в пользовательской программе. Вместо них DB будет содержать случайные данные. SFC создает блок данных с номером из определенной области и с заданными размерами. SFC присваивает наименьший возможный номер из этой области DB. Для создания DB с определенными номерами, введите значения для нижнего и верхнего предела для области номеров по умолчанию. Вы не можете использовать номера DB, уже использованные в пользовательской программе. Длина DB должна быть определена четным числом.

В зависимости от выбора параметра ATTRIB созданный DB может иметь свойство RETAIN или NON_RETAIN:

- RETAIN означает, что DB создается в ретанентной части рабочей области памяти (work memory). Это значит, что текущие значения DB сохраняются после каждого перехода выключение/включение питания и после каждого перезапуска ("теплого" - "warm").
- NON_RETAIN означает, что DB создается в неретанентной части рабочей памяти (work memory). Это значит, что текущие значения DB после каждого перехода выключение/включение питания и после каждого перезапуска ("теплого" - "warm") будут неопределенными.

Если эти части рабочей памяти (work memory), не различаются, то параметр ATTRIB игнорируется. Это значит, что значения DB сохраняются после каждого перехода выключение/включение питания и после каждого перезапуска ("теплого").

Возможность прерывания функции

Системная функция SFC 85 "CREA_DB" может быть прервана OB с более высоким приоритетом. Если SFC 85 "CREA_DB" вновь вызывается в этом OB с высоким приоритетом, то такой вызов будет игнорирован с кодом ошибки W#16#8091.

Параметр	Объявление	Тип данных	Область памяти	Описание
LOW_LIMIT	INPUT	WORD	I, Q, M, D, L константа	Нижнее предельное значение - это наименьший номер из диапазона номеров, который Вы можете назначить для Вашего блока данных
UP_LIMIT	INPUT	WORD	I, Q, M, D, L константа	Верхнее предельное значение - это наименьший номер из диапазона номеров, который Вы можете назначить для Вашего блока данных
COUNT	INPUT	WORD	I, Q, M, D, L константа	Это значение показывает число байтов данных, которое Вы хотите назначить для Вашего блока данных. Вы должны определить четное число байтов (максимум 65534).
ATTRIB	INPUT	BYTE	I, Q, M, D, L константа	Атрибуты DB: <ul style="list-style-type: none"> • В#16#00: RETAIN (реманентные данные) • В#16#04: NON_RETAIN (нереманентные данные)
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время выполнения произошла ошибка, то возвращаемое значение функции будет содержать код ошибки.
DB_NUMBER	OUTPUT	WORD	I, Q, M, D, L	Значение параметра - это номер созданного блока данных. В случае ошибки (бит 15 значения RET_VAL устанавливается), а в DB_NUMBER вводится значение 0.

Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Нет ошибок
8091	Вы вызвали вложенную функцию SFC85.
8092	Функция SFC85 "CREA_DB" в настоящее время не доступна, так как: <ul style="list-style-type: none"> • активирована функция сжатия пользовательской памяти "Compress User Memory" • обнаружена ошибка в ОС компьютера, на котором установлено ПО WinAC.
8094	Некорректное значение параметра ATTRIB
80A1	Ошибка в номере блока DB: <ul style="list-style-type: none"> • Номер равен 0 • Номер превышает верхний предел для DB, определенный для данного CPU • Нижний предел больше верхнего предела
80A2	Ошибка размера блока DB: <ul style="list-style-type: none"> • Размер блока равен 0 • Размер блока задан нечетным числом • Размер блока больше допустимого для данного CPU значения
80B1	Нет свободных номеров для DB.
80B2	Не достаточно свободной памяти.
80B3	Не достаточно непрерывной свободной памяти (запустите функцию "сжатия" памяти - compress).
8хуу	Информация об общих ошибках: см. раздел "Проверка ошибок с выходным параметром RET_VAL"

