ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ

СТАНДАРТНЫЕ РУТИНЫ ДЛЯ СИСТЕМЫ ФАСТБАС

ГОСТ Р 34.1341–93
(МЭК 1052–91)

Издание официальное

Госстандарт России

Москва — 1994

gost.e-ecolog.ru

СОДЕРЖАНИЕ

ПРЕДИСЛОВИЕ 7
ВВЕДЕНИЕ 8

ГЛАВА 1. ВВЕДЕНИЕ, ПРЕДМЕТ И ОБЛАСТЬ РАСПРОСТРАНЕНИЯ СТАНДАРТА 9
1.1. Область распространения и предмет стандарта 9
1.2. Интерпретация указаний в этом стандарте 9
1.3. Обзор документа 10

ГЛАВА 2. ОСНОВНЫЕ ПОЛОЖЕНИЯ 3
2.1. Терминология 3
2.2. Среда 3
2.2.1. Обзор 3
2.2.2. Определение понятия «Среда» 4
2.2.3. Идентификатор Среды 5
2.2.4. Режим исполнения 5
2.2.5. Доступ к Фастбасу в многопользовательской системе 5
2.2.6. Сессия Фастбас 5
2.2.7. Рутины действия 5
2.2.8. Время вызова и время исполнения 6
2.3. Соглашения о наименованиях 6
2.4. Параметры для Рутин: типы и направления 6
2.5. Параметры Буферов 7
2.6. Категории Рутин 8

ГЛАВА 3. ОРГАНИЗАЦИЯ СРЕДЫ И ОТЛОЖЕННОЕ ИСПОЛНЕНИЕ 9
3.1. Открытие и закрытие Фастбас-сессии 9
3.1.1. Открытие Фастбас-сессии 9
3.1.2. Закрытие Фастбас-сессии 9
3.2. Рутины, организующие Среду 10
3.2.1. Создание Среды с Режимом немедленного исполнения 10
3.2.2. Создание Среды с Режимом отложенного исполнения 10
3.2.3. Среда с освобождением ресурсов 11
3.2.4. Среда с восстановлением ресурсов 11
3.2.5. Копирование Среды 12
3.2.6. Получение Среды 12
3.2.7. Установка Среды 12
3.3. Отложенное исполнение и подтверждение правильности Списка 13
3.3.1. Исполнение Списка 13
3.3.2. Подтверждение правильности Списка 13
3.3.3. Установка текущего указателя в определенном Списке 13

ГЛАВА 4. ОПЕРАЦИОННЫЕ ПАРАМЕТРЫ 15
4.1. Введение 15
4.1.1. Операционные параметры в Режиме отложенного исполнения 15
4.2. Определение операционных параметров 15
4.3. Рутины Операционных параметров 16
4.3.1. Операционный параметр инициализации 16
4.3.2. Операционный параметр установки 16
4.3.3. Операционный параметр получения 16
4.3.4. Операционный параметр чтения 17
4.3.5. Операционный параметр записи 17
4.3.6. Проталкивание текущих Операционных параметров в стек 17
4.3.7. Выталкивание текущих Операционных параметров из стека 18
4.4. Всеобщий и Обрабатывающий ошибки Операционные параметры 18
4.4.1. Идентификатор Порта контроллера 18
4.4.2. Не ждать завершения действия 19
4.4.3. Отложенное исполнение 19
4.4.4. Объем среды 19
4.4.5. Генерация ограниченного статуса 19

4.4.6. Возвратный код содержит только Серьезность....20
4.4.7. Порог Серьезности для обработчика ошибок....20
4.4.8. Порог Серьезности для извещателя об ошибках....20
4.4.9. Порог Серьезности исключительной ситуации....20
4.4.10. Порог Серьезности сообщения....21
4.4.11. Доклад сжатого сообщени....21
4.4.12. Доклад о каждом действии....21
4.5. Фастбас-протокол, ОП повтора и превышения лимита времени....22
4.5.1. Уровень Арбитража....22
4.5.2. Протокол Арбитража с обеспеченным доступом....22
4.5.3. Протокол Арбитража с приоритетным доступом....23
4.5.4. Владение магистралью на время передачи целого блока....23
4.5.5. Размер блочка при передаче блока....23
4.5.6. Принудительная передача блока данных....23
4.5.7. Период циклов данных при Принудительной передаче блока данных....24
4.5.8. Устройство с фиксированным Адресом следующей передачи (NTA)....24
4.5.9. Передача данных короткими словами....24
4.5.10. Размер короткого слова....24
4.5.11. Генерация контроля четности....25
4.5.12. Действие Географической адресации....25
4.5.13. Без Цикла арбитража....25
4.5.14. Без Первичного адресного цикла....26
4.5.15. Без Вторичного адресного цикла для Операций в Области данных....26
4.5.16. Без Вторичного адресного цикла в Области регистров CSR....26
4.5.17. Без Цикла данных....26
4.5.18. Удерживать Владение магистралью....27
4.5.19. Удерживать Владение магистралью, если не выставлен сигнал AR....27
4.5.20. Удерживать связку с адресом....27
4.5.21. Удерживать сигналы Мастера при ошибке....28
4.5.22. Долгий таймер....28
4.5.23. Отключение Долгого таймера....28
4.5.24. Удерживать Долгий таймер....28
4.5.25. Таймер ожидания....29
4.5.26. Отключить Таймер ожидания....29
4.5.27. Адресный таймер....29
4.5.28. Отключить Адресный таймер....29
4.5.29. Таймер данных....30
4.5.30. Отключить Таймер данных....30
4.5.31. Программный таймер....30
4.5.32. Отключить Программный таймер....31
4.5.33. Удержать Программный таймер....31
4.5.34. Число повторений....31
4.5.35. Время повторения....31

ГЛАВА 5. БУФЕРЫ ДАННЫХ....32
5.1. Аргументы доступа к буферам....32
5.1.1. Буфер специфицирован в качестве переменной или массива....32
5.1.2. Буфер специфицирован в качестве значения....32
5.1.3. Буфер специфицирован в качестве идентификатора Последовательного буфера....32
5.2. Рутины Последовательных буферов данных....33
5.2.1. Объявить Внутренний буфер....33
5.2.2. Объявить Внешний буфер....33
5.2.3. Отключить Внутренний последовательный буфер данных....33
5.2.4. Считать указатель Последовательного буфера....34
5.2.5. Записать указатель Последовательного буфера....34
5.2.6. Сместить указатель Последовательного буфера....35

ГЛАВА 6. РУТИНЫ ПРОСТЫХ ПЕРЕДАЧ....36
6.1. Условия, управляющие передачами, и сложные Рутины....36
6.1.1. Арбитраж....36
6.1.2. Первичный адресный цикл....36

6.1.3. Вторичный адресный цикл записи....37
6.1.4. Циклы данных....37
6.1.5. Передачи блоков....37
6.1.6. Разъединение....38
6.2. Рутины простых передач Фастбас....38
6.2.1. Передача одиночного слова Данных....38
6.2.2. Передача Блока....39
6.2.3. Рутины Вторичного Адреса....40
6.2.4. Считать длину последней передачи данных....41
ГЛАВА 7. РУТИНЫ СЛОЖНЫХ ПЕРЕДАЧ....42
7.1. Доступ к маршрутной таблице Соединителей сегментов....42
7.2. Передача данных между Фастбас-устройствами....42
7.3. Размещение Фастбас-информации при Чтении-Модификации-Записи....43
7.4. Рутины Объединения Данных....44
7.4.1. Считать данные из последовательных адресов Устройств....44
7.4.2. Читать данные из Устройств, откликающихся на сканирование Т-штырьков....45
7.4.3. Присоединить Фастбас-устройство....45
7.4.4. Отсоединить Фастбас-устройство....46
7.5. Послать сообщение о Фастбас-прерывании....46

ГЛАВА 8. РУТИНЫ ПРИМИТИВНЫХ ФАСТБАС-ДЕЙСТВИЙ....47
8.1. Введение....47
8.2. Параметры....47
8.3. Рутины одиночных циклов....47
8.3.1. Арбитраж....47
8.3.2. Отказ от Владения магистралью....48
8.3.3. Первичный адресный цикл....48
8.3.4. Разъединение....48
8.3.5. Передачи одиночного слова....49
8.3.6. Передачи блока....49
8.4. Доступ к линиям магистрали Фастбас....50
8.4.1. Читать отдельную (-ные) линию (-ии) магистрали....51
8.4.2. Записать отдельную (-ные) линию (-ии) магистрали....51
8.4.3. Читать внутреннюю (-ие) линию (-ии)....52

ГЛАВА 9. ФАСТБАС-РУТИНЫ ЗАПРОСОВ НА SR И СООБЩЕНИЙ О ПРЕРЫВАНИЯХ....53
9.1. Введение....53
9.2. Запрос на обслуживание SR....53
9.2.1. Очистка установленного запроса на обслуживание....53
9.2.2. Присоединить Рутину к SR....54
9.2.3. Отсоединить Рутину от SR....54
9.2.4. Разрешить связи с SR....54
9.2.5. Отменить связи с SR....55
9.3. Рутины реакции на Сообщение о Фастбас-прерывании....55
9.3.1. Связать Рутину с FIR....55
9.3.2. Отсоединить Рутину от FIR....56
9.3.3. Разрешить связи с FIR....56
9.3.4. Отменить связи с FIR....57

ГЛАВА 10. РУТИНЫ СИНХРОНИЗАЦИИ, СИСТЕМНЫХ РЕСУРСОВ И ПОРТОВ....58
10.1. Средства синхронизации....58
10.1.1. Ждать завершения операции....58
10.1.2. Проверить состояние завершения....58
10.1.3. Присоединить Рутину завершения....58
10.1.4. Отсоединить Рутину завершения....59
10.2. Сброс Порта Фастбас....59
10.3. Назначение Порта Фастбас....59
10.3.1. Назначить Порт Фастбас....60
10.3.2. Отменить назначение Порта Фастбас....60
10.3.3. Назначить Порт Фастбас с таймаутом....60
10.4. Выдать номера версий....60

10.5. Сообщить об ошибке Порта Фастбас 61
10.6. Сформировать сообщение об ошибке Порта 61

ГЛАВА 11. ОБРАБОТКА СОСТОЯНИЙ И ОШИБОК 62
11.1. Введение 62
11.2. Коды ошибок и возвратов 62
11.3. Суммарный и Дополнительный Статус 62
11.4. Усеченная генерация статусной информации 63
11.5. Статус Режима задержанного исполнения 63
11.6. Серьезность 63
11.6.1. FB_SEV_SUCCES (FSSUCC) 63
11.6.2. FB_SEV_INFO (FSINFO) 63
11.6.3. FB_SEV_WARNING (FSWARN) 63
11.6.4. FB_SEV_ERROR (FSERR) 64
11.6.5. FB_SEV_FATAL (FSFTL) 64
11.6.6. FB_SEV_NEVER (FSNEV) 64
11.7. Установка Серьезности Кода Ошибки 64
11.8. Отклик на ошибки при исполнении 65
11.8.1. FB_RESP_IGNORE 65
11.8.2. FB_RESP_ABORT 65
11.8.3. FB_RESP_RETRY_ABORT 65
11.8.4. FB_RESP_RETRY_IGNORE 66
11.8.5. FB_RESP_ABORT_ACTION 66
11.8.6. FB_RESP_RETRY_ABORT_ACTION 66
11.8.7. Серьезность Кодов ошибок, выработанных для соответствующих Откликов на Ошибки. 66
11.8.8. Специфицировать и определить отклик на ошибку 67
11.9. Автоматический Докладчик ошибок 67
11.9.1. Присоединить Рутину обработки ошибок Пользователя 68
11.9.2. Отсоединить Рутину обработки ошибок Пользователя 68
11.9.3. Подробности Рутины обработки ошибок Пользователя 68
11.10. Доступ к информации Суммарного сатуса 69
11.10.1. Декодировать Суммарный сатус 69
11.11. Доступ к информации Дополнительного сатуса 69
11.11.1. Декодировать информацию Дополнительного статуса 70
11.11.2. Декодировать Дополнительный статус при исполнении Списка 70
11.11.3. Поиск информации Дополнительного статуса 71
11.11.4. Найти специфический Дополнительный статус в Списке 71
11.12. Рутины Обработки Кодов Ошибки 72
11.12.1. Проверить Серьезность Кода ошибки 72
11.12.2. Проверить Код Статуса в Коде ошибки 72
11.12.3. Проверить Серьезность в Коде ошибки 72
11.12.4. Возвратить Код ошибки специфицированной Серьезности 73
11.13. Доложить Фастбас-ошибку 73
11.14. Выработать Сообщение об Ошибке 73

ГЛАВА 12. КОДЫ ОШИБОК 75
12.1. Стандартный набор Кодов ошибок 75

Приложение А. Перечень имен Рутин и Параметров 84
А1. Рутины, сортированные по Краткому имени 86
А2. Рутины, сортированные по Полному имени 97

Приложение В. Список зарезервированных имен 92
В1. Список, сортированный по Полному имени 92
В2. Список, сортированный по Краткому имени 97

Приложение С. Типы данных ФОРТРАН-77 103

Указатель 104

Предисловие научного редактора русского текста

Разработка этого стандарта началась в США в процессе создания первых устройств на основе стандарта Фастбас, утвержденного впервые в ноябре 1982 г. комитетом NIM (Nuclear Instruments Modules) Департамента энергии США. На годичном Nuclear Science Symposium в августе 1984 г. было объявлено об издании спецификации «Fastbus Standard Routines». После уточнения этот документ в апреле 1987 г. был представлен в Международную электротехническую комиссию в качестве проекта международного стандарта. Ровно через четыре года вышел в свет стандарт МЭК 1052(1991), идентичный американскому стандарту ANSI/ IEEE 1177-1989. ГОСТ 34.341 является полным аутентичным переводом международного стандарта и предназначен для использования совместно с основным стандартом Фастбас, который издан на русском языке в виде ГОСТ 34.340.

Разработка стандартных Рутин стала возможной благодаря тому, что в гл. 8 основного стандарта впервые в мире были определены и стандартизованы функции всех битов регистров управления и состояний. Во многих Рутинах имеются прямые ссылки на биты РУС и их функции. Рутины позволяют создать комплект БИС, аппаратно воплощающих стандарт и обеспечивающих максимально быстрое программирование, трансляцию и исполнение программ. Рутины можно применить и другим способом - с использованием универсальных программ на языке высокого уровня при существенно более медленной работе. Такие программы будут учитывать логические требования, описанные в Рутинах, и явятся, скорее всего, промежуточной стадией на пути к комплекту БИС.

Точные формулировки логических требований, данные в описаниях Рутин настоящего стандарта, обеспечивают совместимость программ, созданных в разных фирмах, с аппаратурой, соответствующей стандарту Фастбас. Оба стандарта совместно определяют *5* уровней Модели открытых систем Международной организации стандартов, начиная с физического уровня и кончая сессионным уровнем (п. 3.1). Охват пяти уровней всего двумя стандартами является уникальным в мире.

В гл. 10 основного стандарта определены Соединители сегментов, обеспечивающие передачу информации между Крейт-сегментами при помощи Кабель-сегментов с использованием параллельных кодов. В комитетах NIM и IEEE в настоящее время завершается разработка третьего стандарта Фастбас на оптоволоконные каналы связи между Крейт-сегментами для передачи последовательным кодом. Ожидается, что и этот стандарт станет международным и в дальнейшем государственным стандартом России.

При обоих способах передачи обеспечивается образование сети произвольной конфигурации. Поскольку информация между модулями и между крейтами передается на основе одного и того же логического протокола Фастбас, Рутины действительны для Работы как внутри крейта, так и в сети. Это свойство системы Фастбас также уникально в мире. В целом система Фастбас с присущей ей собственной универсальной сетью оказывается наиболее эффективной из магистрально-модульных многопроцессорных информационно-измерительно-управляющих систем.

Многие термины, употребляемые в вычислительной технике, в стандартах Фастбас имеют специальный смысл, определенный стандартами. Например, Рутина в этом стандарте это не стандартная собственно *программа*, а логические *требования* к созданию программы. Поэтому Первичный адрес, Вторичный адрес, Среда, Воплощение и многие другие термины в оригинальных стандартах и в ГОСТ 34.340 и ГОСТ 34.341 написаны с заглавной буквы, чтобы предотвратить ошибки, возможные при попытках использовать в системе Фастбас термины в общепринятом смысле.

Гл. 6–9 перевела М.А.Пирогова, гл. 10, 11 перевел В.А. Краюшкин, гл. 1–5 – К.Э.Эрглис. Он же выполнил научное редактирование и корректуру всего текста.

К.Эрглис

МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ СТАНДАРТ МЭК-1052: СТАНДАРТНЫЕ ПРОГРАММЫ ДЛЯ СИСТЕМЫ ФАСТБАС

Предисловие

1. Формальные решения или соглашения по техническим вопросам, подготовленные Техническими комитетами, в которых представлены заинтересованные Национальные комитеты, выражают, по возможности наиболее точно, согласованное международное мнение по рассматриваемым стандартам.

2. Стандарты имеют статус рекомендаций для международного использования и приняты Национальными комитетами именно в этом смысле.

3. В целях распространения международной унификации, МЭК выражает пожелание, чтобы все Национальные комитеты приняли текст рекомендаций МЭК для своих национальных норм, насколько позволят национальные условия. Любое различие между рекомендацией МЭК и соответствующими национальными нормами должно быть, насколько возможно, ясно указано в последних.

4. МЭК не формулирует какие-либо процедуры, касающиеся маркировки в знак признания продукции, и не несет ответственности, когда продукция объявлена соответствующей одной из рекомендаций.

Информация

Настоящий стандарт подготовлен Техническим комитетом 45: Ядерное приборостроение. Текст стандарта основан на следующих документах:

По Правилу шести месяцев	По Отчету о голосовании
45(ЦБ)198	45(ЦБ)205

Полную информацию о голосовании по одобрению этого стандарта можно получить из Отчета о голосовании, указанного в выше.

В стандарте использована Публикация МЭК 935(1990): ФАСТБАС - Модульная быстродействующая система сбора данных.

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

УДК 681.118.087:006.354 Группа П70

ГОСТ Р 34.1341–93 (МЭК 1052–91)

ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ

СТАНДАРТНЫЕ РУТИНЫ ДЛЯ СИСТЕМЫ ФАСТБАС

Information technology

Fastbus Standard Routines

ОКП 42 2000 Дата введения 01.07.94

Глава 1. ВВЕДЕНИЕ, ПРЕДМЕТ И ОБЛАСТЬ РАСПРОСТРАНЕНИЯ СТАНДАРТА

Настоящий стандарт определяет стандартные программы для доступа к системе ФАСТБАС – стандартизованной модульной быстродействующей системе сбора данных и управления, определенной ГОСТ 34.340 – ФАСТБАС – модульная быстродействующая система сбора данных.

Принято, что читатель знаком с системой Фастбас.

1.1 Область распространения и предмет стандарта

Стандартные программы (Рутины), определенные настоящим стандартом, применимы к системам Фастбас, соответствующим ГОСТ 34.340. Назначение документа – обеспечить разработчиков стандартными программами, имеющими единый смысл для всех пользователей, чтобы уменьшить дублирование в разработках программ и максимально повысить взаимозаменяемость в программах и аппаратуре Фастбас.

1.2 Интерпретация указаний в этом стандарте

Пункты стандарта, в которых использовано слово «должен», являются обязательными.

Пункты, в которых использовано слово «следует», являются рекомендацией предпочтительной практики, которой нужно следовать, если нет серьезных причин к иным решениям.

Пункты, в которых использовано слово «можно», указывают на допустимые решения и оставляют разработчику свободу выбора.

Отдельные Воплощения стандарта, отнесенные к случаям, которые определены в этом стандарте как «зависящие от воплощения», должны быть ясно описаны в соответствующей документации.

Воплощение, противоречащее любому обязательному параграфу, нельзя рассматривать как воплощающее этот стандарт.

Для использования этого стандарта не требуется ни лицензий, ни иных разрешений.

1.3 Обзор документа

В гл.2 описаны некоторые основные положения, общие для всех Рутин.

Гл. 3 посвящена Рутинам, применяемым в связи с использованием Среды, с которой работает система Фастбас.

В гл. 4 описаны Операционные параметры, которые образуют часть Среды Фастбас.

В гл. 5 описаны буферные режимы, которые действующая программа использует для управления доступом к буферам данных пользователя.

В гл. 6 описаны Рутины для выполнения простых передач Фастбас.

В гл. 7 описаны Рутины для выполнения сложных передач Фастбас.

В гл. 8 описаны Рутины для выполнения примитивных действий Фастбас.

В гл. 9 описаны Рутины для обработки асинхронных событий в системе Фастбас, для реакций на запрос обслуживания и реакций на сообщения о прерывании.

В гл. 10 описаны Рутины, зависящие либо от системы, либо от порта.

В гл. 11 описаны отзывы Воплощения на ошибки. Описаны Рутины, управляющие отзывами, и Рутины для извлечения статусной информации о предыдущих действиях.

Гл. 12 содержит список всех кодов ошибок наряду со степенью их серьезности.

Глава 2. ОСНОВНЫЕ ПОЛОЖЕНИЯ

2.1 Терминология

В настоящем стандарте использованы термины, имеющие смысл, специфический для системы Фастбас. Такие термины, как «принудительная передача», «широкая адресация», «Сегмент» и другие, определены в стандарте Фастбас. Их определения здесь не повторяются.

Рутины в этом стандарте определены как средство связи между «Пользователем», который их вызывает, и «Воплощением», которое вызываемо Рутиной. В сферу Фастбас-сессии входят понятия «Пользователь» или «Процесс пользователя». Некоторые общие термины определены ниже.

«Фастбас-действия» непосредственно управляют линиями в сегменте Фастбас или считывают с линий. Не все Воплощения Фастбас поддерживают все действия этого типа, возможные в принципе.

«Примитивным Фастбас-действием» является одно из следующих возможных логически завершенных поддействий:

Цикл арбитража.
Первичный адресный цикл.
Вторичный адресный цикл данных.
Цикл данных.
Освобождение устройства (Завершающая последовательность).
Освобождение магистрали.

Не все Воплощения в Фастбасе поддерживают исполнение каждого одиночного поддействия, но все Воплощения должны поддерживать Операции и Передачи так, как они определены ниже, чтобы их можно было квалифицировать как Воплощения этого стандарта.

«Операция» состоит из Первичного адресного цикла, за которым следует ноль или более Циклов данных и Завершающая последовательность.

«Простая операция» это Операция, в которой первый Цикл данных после Первичного адресного цикла может определять только Вторичный адрес.

«Передача» состоит в получении Мастером владения магистралью, сопровождаемого исполнением одной или больше Операций с последующим освобождением магистрали.

«Простая передача» это передача с исполнением единственной Простой операции.

«Контроллер» является зачинателем вызовов Фастбас-действий, описанных в этом стандарте.

«Порт» Фастбас связывает Контроллер с Сегментом Фастбас.

Между Контроллером и Сегментом Фастбас могут существовать множественные маршруты. Для различения разных маршрутов должны быть использованы идентификаторы портов port_id.

Понятие "Стандарт Фастбас", используемое здесь, ГОСТ 34.340 – ФАСТБАС – модульная быстродействующая система сбора данных.

2.2 Среда

2.2.1 Обзор

Стандарт Фастбас определяет протокол, по которому Мастер сначала получает владение магистралью, устанавливает соединение с одним или несколькими Слугами, а затем передает данные. Мастер может прервать соединение со Слугами и, не оставляя контроль над Сегментом, может

установить соединение с другим Слугой. В любой момент Мастер может освободить магистраль, чтобы другой Мастер мог получить управление магистралью сразу же после прекращения связки с адресатом.

Все эти действия имеют место в Среде, относительно которой Пользователь имеет некоторые средства управления. Именно степень управления, доступная Пользователю, будет изменяться от одного Воплощения к другому. Контроллер может или может не быть способным изменять параметры, влияющие на Фастбас-действия, а также и система программирования процессора может или не может допускать использование определенных операций или техники. Чтобы сделать возможным использование широкого диапазона аппаратных и программных ресурсов, которые могут быть доступны при различных приложениях, в этом стандарте проведены различия между действиями, которые имеют место в собственно Сегменте Фастбас, и Средой, в которой происходят действия. Рутины определены так, чтобы управлять как характером Среды, так и действиями, происходящими в Сегменте.

Процессы в системе Фастбас определяются инструкциями, выдаваемыми Контроллером. Контроллер может быть большой вычислительной системой со многими стандартными периферийными устройствами, как диски и магнитофоны (в дополнение к Фастбасу), и с многопользовательской многозадачной операционной системой, которая обеспечивает возможность для одновременных работ в широком диапазоне, таких как разработка программ и сбор данных. В другом случае в устройстве Фастбас может быть процессор, имеющий хотя бы только рудиментарную операционную систему для одного пользователя и одной задачи. Такой процессор может выполнять программы, подготовленные на стороне и хранимые в постоянной памяти.

Результатом исполнения одних инструкций, выдаваемых Контроллером Фастбас, будут циклы различного типа в собственно системе Фастбас, в то время как другие инструкции будут управлять Средой, в которой имеют место циклы Фастбас. Эта Среда содержит уровень арбитража, используемый Мастером в соревновании за владение магистралью, различные выдержки времени, условия об удержании или освобождении магистрали в конце текущей Операции и т.д. В этом стандарте предусмотрена возможность спецификации таких условий, хотя от конкретного Порта Фастбас не требуется способность воспринимать все изменения.

В другом аспекте Среда определяет, как система ведет себя после окончания вызванной Рутины Фастбас. Предусмотрен автоматический отклик на ошибочные условия и автоматическое извещение об ошибках. Пользователь имеет контроль над реакциями аппаратных и программных средств в определенных условиях, а также над тем, что выводить или не выводить для показа.

Среда также определяет источник инструкций для Порта Фастбас. Эти инструкции могут исходить непосредственно от стандартной программы, как только она встречается в исполняемой программе, или они могут быть взяты из Списка инструкций, который был создан предварительно. Создан ли в системе Фастбас Список или нет, время вызова Рутины определяется Средой.

2.2.2 Определение понятия «Среда»

«Среда» - это информация, которую хранит Воплощение по воле Пользователя. Эта информация состоит из пяти компонентов:

1. Комплект «Операционных параметров», которые имеют значения, фиксированные по умолчанию, и могут быть явно установлены Пользователем (см. гл. 4).
2. Информация, управляющая откликом Воплощения на ошибки и другие несанкционированные условия (см. гл. 4 и 11).
3. Информация, относящаяся к последовательностным буферам (см. гл. 5).
4. Информация о состояниях, выдаваемая Воплощением, описывающая исполненные действия и ошибки (см. гл. 11).
5. «Список» действий Фастбас. Этот список используют только в Режиме отложенного исполнения в качестве записи действий, подлежащих исполнению позднее. В Среде, которая не поддерживает Режим отложенного исполнения, Список отсутствует (см. гл. 3).

2.2.3 Идентификатор Среды

К Среде обращаются при помощи параметра environment_id - идентификатора Среды, который придается большинству Рутин Фастбас. Воплощение должно предусматривать Среду-по-умолчанию, имеющую идентификатор Среды FB_DEFAULT_EID. Вызовом одной из Рутин, указаиных в разделе 3.2, могут быть созданы другие Среды. Пользователь может работать одновременно с несколькими Средами.

2.2.4 Режим исполнения

Среда может содержать один из двух Режимов исполнения: «Режим немедленного исполнения» или «Режим отложенного исполнения». Режим исполнения указывает состояние Операционного параметра FB_PAR_DELAY_EXEC - соответственно FB_FALSE или FB_TRUE. Первоначальный Режим исполнения зависит от того, как создана Среда. Среда, созданная «немедленно», должна всегда оставаться с Режимом немедленного исполнения. Среда, созданная «отложенно», может быть изменена последовательно на Режим немедленного исполнения и обратно при помощи изменений значения FB_PAR_DELAY_EXEC.

Для Воплощения обязательно поддержание Режима немедленного исполнения, в то время как поддержание Режима отложенного исполнения реализуется по желанию.

Режим исполнения влияет на то, что происходит, когда вызывается Рутина действия.

Действия, вызванные Рутинами действия, когда идентификатор Среды не прошел как параметр, должны всегда исполняться немедленно.

2.2.5 Доступ к Фастбасу в многопользовательской системе

Магистраль Фастбас и адресные связки предусматривают правильное использование магистрали множеством Мастеров. Программное Воплощение должно обеспечивать взаимное исключение нескольких Мастеров, достигающих Фастбас через один и тот же Порт. Воплощение должно предотвращать доступ других Пользователей к магистрали через Порт, если какой-нибудь Пользователь уже установил через этот Порт Владение магистралью (связка GK) или связался с другим адресом (связка AS-AK).

Явное определение Порта является операцией высокого уровня, которая обеспечивает пользователю гарантию исключительного использования Порта Фастбас для выполнения серии операций без соревнования за Порт. Следует заметить, что определение Порта само по себе не гарантирует использование самой магистрали (см. раздел 10.3).

2.2.6 Сессия Фастбас

Вызов Рутины FB_OPEN, описанный в гл.3, определяет сессию Фастбас. В пределах этой сессии пользователь должен иметь доступную Среду-по-умолчанию (отмеченную идентификатором Среды FB_DEFAULT_EID). Любые назначения Порта или Устройства (см. гл. 10) действительны в течение всей выполняемой сессии Фастбас.

2.2.7 Рутины действия

В последующих описаниях Рутин некоторые из них отмечены как «Рутины действия». Этим Рутинам присуще «действие», влияющее на систему Фастбас или «Буферы данных», или на то и другое. Все Рутины, в имени которых есть слова READ или WRITE (вместо GET или SET, см. раздел 4.1), являются Рутинами действия.

В Режиме немедленного исполнения, когда вызвана Рутина действия, действие выполняется сразу. Статусная информация об успехе или неудаче действия запоминается в Среде (Статусная

история, см. гл. 11). Точное действие, подлежащее исполнению, зависит от вызванной Рутины и от состояния Операционных параметров. Передача данных Рутиной действия в буферы Пользователя и обратно происходит в соответствии с определенным Режимом буфера (см. гл. 5).

В Режиме отложенного исполнения, когда вызвана Рутина действия, действие начинается не сразу. При этом действие, определенное Рутиной в соответствии с состоянием Операционных параметров на данный момент, вносится в «Список» действий и может быть выполнено в любое более позднее время. Возвратный код Рутины дает индикацию лишь успеха в добавлении действия к Списку, но на этой стадии Статусная история не создается и предыдущая Статусная история остается нетронутой. На этой стадии не происходят передачи данных в Буферы данных (за исключением единственного слова WRITE [записать] данные - в режиме FB_BUFFER_VALUE - см. гл. 5).

Рутина FB_LIST_EXECUTE вызывает исполнение по порядку всех действий, запомненных в Списке. Когда вызвана эта Рутина, создается Статусная история и передаются данные в Буферы данных Пользователя и обратно, как полагается для каждого действия.

2.2.8 Время вызова и время исполнения

При описании результатов работы Рутин действия используют два термина, причем один и тот же термин может быть применен к описанию режимов как немедленного, так и отложенного исполнения. "Время вызова" относится к моменту вызова Рутины действия, а "Время исполнения" – к моменту, когда начинается само действие. Поэтому в Режиме немедленного исполнения Время исполнения то же самое, что Время вызова, в то время как в Режиме отложенного исполнения Время исполнения это момент вызова Рутины FB_LIST_EXECUTE.

2.3 Соглашения о наименованиях

Каждой Рутине присвоено описательное имя в двух формах: полное имя (до 32 знаков) и краткое имя (до 6 знаков). Все полные имена начинаются с FB_ и все краткие – с F. По этому стандарту оба имени полагается набирать на верхнем регистре клавиатуры.

Воплощение должно узнавать краткие имена. Воплощению следует узнавать полные имена.

2.4 Параметры для Рутин: типы и направления

Имя, тип данных и направление приведены для каждой Рутины в следующей форме.

Параметры:

<имя>	<тип>	<направление>
<имя>	<тип>	<направление>
...	...	...

Поле <имя> используют только для ссылок на параметр внутри последующего описания Рутины. По этому стандарту поле <имя> полагается всегда набирать строчными буквами нижнего регистра.

Поле <тип> показывает использование и, для каждого Воплощения, потребный объем памяти для параметра. Обозначение типа всегда начинается с букв FB_ с последующими строчными буквами.

Многие параметры хранятся как слова из 32 битов. Однако во многих случаях значения аргументов, которые можно взять, ограничены. Некоторые значения, подобные идентификаторам Среды, присвоены Рутиной и не должны быть изменяемы Пользователем. Другие типы должны иметь только символические значения, определенные этим стандартом. Преобразование этих имен в запоминаемые значения зависит от Воплощения. Ниже представлены поля <тип> с указанием их применения:

FB_boolean Принимает значения FB_FALSE или FB_TRUE.

FB_buffer_mode	Определяет значение относительного параметра buffer_specifier (см. гл. 5).
FB_environment_id	Значение FB_DEFAULT_EID может быть использовано всегда. Единственное другое значение, которое может быть использовано, обсуждается в разделе 3.2.
FB_error_code	Код ошибки (см. гл. 11). Содержит статусную информацию и Серьезность.
FB_integer	Принимает любые значения от -2^{31} до $2^{31} - 1$.
FB_line	Идентифицирует сигнал или группу сигналов в Фастбасе. Принимает любое значение, данное в разделе 8.4.
FB_list_pointer	Позиция в Списке действий. Представление зависит от Воплощения.
FB_operator	Идентифицирует арифметическую или логическую операции, которые имеют место в течение Фастбас-цикла Чтение-модификация-запись (см. гл. 7).
FB_parameter_id	Идентифицирует конкретный Операционный параметр. Принимает любое из значений, перечисленных в гл. 4, или значение, зависящее от применения.
FB_parameter_value	Значение Операционного параметра. Допустимый диапазон значений зависящий от параметра, дан в гл. 4.
FB_port_id	Идентифицирует конкретное соединение между Контроллером и Фастбас-сегментом.
FB_response_code	Описывает действие, которое предпринято в результате определенных условий в аппаратуре Фастбас. См. раздел 11.8.
FB_routine	Параметр, используемый для введения программы Пользователя в качестве аргумента к Фастбас-Рутине.
FB_seq_buffer_id	Идентификатор Последовательстного буфера, описанный в гл. 5.
FB_severety	Описывает Серьезность состояния с ошибками. Принимает любое значение из списка в разделе 11.6.
FB_word	32-разрядное слово не определенного значения.

Тип может быть расширен при помощи префикса «array_of», который отмечает массив не определенного размера. Размер массива в байтах всегда дается как отдельный параметр.

Представление этих типов в Среде Фортрана-77 дано в приложении С.

Поле <направление> указывает на направление потока информации и должно иметь одно из следующих значений:

IN	Программа читает аргумент, но не изменяет его.
OUT	Программа возвращает значение аргументу или определенному выражению.

Например. Рутина FRD или FB_READ_DAT имеет следующие параметры:

return_code		FB_error_code	(OUT)	
environment_id		FB_environment_id	(IN)	
primary_address		FB_word	(IN)	
secondary_address		FB_word	(IN)	
buffer_mode		FB_buffer_mode	(IN)	
buffer_specifier		FB_word	(OUT)	(FBVAR)
	или	FB_seq_buffer_id	(IN)	(FBSEQ)

Здесь показано, что параметр в первой строке является Возвращаемым кодом, возвращаемым программой, во второй строке параметр является управляющим идентификатором, переданным программе и не возвращенным. Третий и четвертый параметры это 32-разрядные слова, передаваемые программе. Параметры buffer_mode и buffer_specifier определяют источник или назначение данных в Контроллере и обсуждаются ниже. Параметр buffer_specifier может быть в одной или иных формах, записанных для каждой Рутины.

2.5 Параметры Буферов

Любая Рутина действия среди своих параметров имеет два параметра буферов:

buffer_mode	FB_buffer_mode	(IN)
buffer_specifier	<тип>	<направление>

Здесь значение параметра Буферного режима определяет значение второго параметра, его тип и, возможно, его направление.

В трех возможных Буферных режимах для передачи данных допускается использование переменной, специфицированной величины (литерала) или нумерованного последовательностного буфера. Режимы даны ниже:

Buffer_mode Value [значение Буферного режима]		Buffer_specifier [спецификатор буфера]	
(полное имя)	(краткое)	<тип>	<направление>
FB_BUFFER_VAR	FBVAR	(array_of_)FB_word	IN или OUT
FB_BUFFER_VALUE	FBVAL	FB_word	IN
FB_BUFFER_SEQ	FBSEQ	FB_seq_buffer_id	IN

Буферные режимы и их использование в Средах Режимов отложенного и немедленного исполнения обсуждаются более детально в гл. 5.

В приложении С дано предлагаемое представление типов Фастбас-данных в Среде Фортрана-77.

2.6 Категории Рутин

В этом стандарте каждой Рутине присвоена одна из категорий:

Рутины категории А обязательны для любого Воплощения.

Рутины категории В обязательны для любого Воплощения, поддерживающего Среду Режима отложенного исполнения.

Рутины категории С должны придаваться тем Контроллерам Фастбас, которые могут быть запрограммированы для выполнения одиночных Фастбас-циклов или управления отдельными сигнальными линиями в магистрали Фастбас.

Рутины, соответствующие категории I, должны создаваться, когда Контроллер предназначен для работы в качестве Слуги, и принимать сообщения о прерываниях Фастбас или выполнять действия, связанные с выставлением сигнала SR.

Рутины категории S являются системными и зависят от Воплощения. Их Воплощение необязательно и их Операция может не быть одной и той же для всех Воплощений.

Рутины категории Е произвольны для любого Воплощения.

Глава 3. ОРГАНИЗАЦИЯ СРЕДЫ и ОТЛОЖЕННОЕ ИСПОЛНЕНИЕ

Конкретное Воплощение Рутин Фастбас может зависеть от архитектуры Контроллера и Порта. В поддержании или отсутствии Режима отложенного исполнения заключается главное различие между Воплощениями. Это различие отображается в Воплощении (см. гл. 2).

Все Воплощения должны обеспечивать Среду-по-умолчанию, обозначаемую идентификатором среды FB_DEFAULT_EID.Эта Среда должна быть только Средой Режима немедленного исполнения.

В Воплощениях, которые не поддерживают Режим отложенного исполнения, Среда-по-умолчанию,равно как и любая Среда, создаваемая пользователем, должна содержать операционные параметры, отклик на ошибки и статусную инфрмацию,а также информацию о Последовательном буфере, если он есть. В Воплощениях, которые поддерживают Режим отложенного исполнения, любые Среды, созданные Пользователем с Режимом отложенного исполнения, должны быть способны содержать дополнительно Список действий, подлежащих исполнению.

3.1 Открытие и Закрытие Фастбас-сессии

Описаны две Рутины, которые должны быть вызываемы любой программой до и в конце Фастбас-сессии.

3.1.1 Открытие Фастбас-сессии

До вызова любой Рутины, определенной этим стандартом, Пользователем должна быть вызвана нижеследующая Рутина. Параметры, зависящие от воплощения, могут быть введены для спецификации деталей, указывающих, как должна быть использована Система Фастбас.

Ни один из Операционных параметров не должен влиять на эту Рутину.

Имя: FBOPEN или FB_OPEN

Категория: A

Параметры:

return_code FB_error_code (OUT)

дальнейшие параметры зависят от Воплощения

Действие:

Этот вызов может быть использован для выполнения программной и аппаратной инициализации Контроллера, но не должен выполнять какого либо Фастбас-действия на магистрали.

Не влияя на Фастбас-сегмент или выполнение какой-либо операции Фастбас, в Среде-по-умолчанию все Операционные параметры должны принимать свои состояния-по-умолчанию.

Если FB_OPEN была вызвана раньше, чем FB_CLOSE, Рутина FB_CLOSE должна быть обязательно выполнена.

3.1.2 Закрытие Фастбас-сессии

Если использование системы Фастбас больше не требуется, Пользователь должен вызвать следующую Рутину:

Имя: FBCLOS или FB_CLOSE

Категория: A

Параметры:

return_code	FB_error_code	(OUT)	

Действие:

Если Пользователь покидает Порт Фастбас, когда на на каких-либо линиях магистрали выставлены сигналы, эти сигналы должны быть сняты. Ни один из операционных параметров не должен влиять на эту Рутину. Если, после вызова Рутины FB_CLOSE, Пользователь вызывает любую Рутину, определенную в этом стандарте, до нового вызова Рутины FB_OPEN, должно быть выдано извещение об ошибке.

Статусная история не генерируется.

3.2 Рутины, организующие Среду

3.2.1 Создание Среды с Режимом немедленного исполнения

Имя: FCIENV или FB_CREATE_IMMEDIATE_ENVIRONMENT

Категория: A

Параметры:

return_code	FB_error_code	(OUT)	
environment_id	FB_environment_id	(OUT)	(см.текст)

Действие:

Должна быть создана или назначена Среда, состояние которой по умолчанию определяется Воплощением в Режиме немедленного исполнения (FB_PAR_DELAY_EXEC имеет значение FB_FALSE). Для Сред, создаваемых таким путем, Операционный параметр FB_PAR_DELAY_EXEC должен быть только читаемым. Возвращаемое значение environment_id (идентификатора Среды) должно быть использовано в качестве входного параметра для определения Среды для других Рутин.

Для Воплощений, распределяющих память, эта Рутина должна обеспечить доступность необходимой для Среды зоны памяти и возвращать идентификатор среды.

(В тех Воплощениях, где распределение памяти должен выполнять Пользователь, параметр идентификатора Среды указывает Пользователю назначенную зону адресов, которая должна быть использована для хранения и идентификации определения Среды. В этом случае направлением параметра является IN).

3.2.2 Создание Среды с Режимом отложенного исполнения

Имя: FCDENV или FB_CREATE_DELAYED_ENVIRONMENT

Категория: B

Параметры:

return_code	FB_error_code	(OUT)	
environment_id	FB_environment_id	(OUT)	
maximum_calls	FB_integer	(IN)	(см.текст)

Действие:

Должна быть создана или назначена Среда с Режимом отложенного исполнения. Операционный параметр FB_PAR_DELAY_EXEC имеет значение FB_TRUE, а в других отношениях Среда находится в состоянии, которое по умолчанию определено Воплощением. Для Сред, создаваемых таким путем, Операционный параметр FB_PAR_DELAY_EXEC должен быть read/write [чтение/запись]. Должен быть определен соответствующий идентификатор Среды environment_id и использован для определения Среды для других Рутин этого стандарта. Параметр maximum_calls [максимальное число вызовов] должен задавать максимальное число Фастбас-действий, которые могут быть указаны в Списке.

Созданная Среда должна содержать пустой или нулевой Список. Для Воплощений, распределяющих память, эта Рутина должна обеспечивать доступность необходимой для Среды зоны памяти и возвращать идентификатор Среды.

(В тех Воплощениях, где распределение памяти должен выполнять Пользователь, параметр идентификатора Среды указывает Пользователю назначенную зону адресов, которая должна быть использована для хранения и идентификации определения Среды и относящегося к ней Списка. В этом случае направлением параметра является IN).

3.2.3 Среда с освобождением ресурсов

Имя: FRLENV или FB_RELEASE_ENVIRONMENT

Категория: A

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие:

Все ресурсы системы, связанные со Средой, определенной идентификатором environment_id, должны быть освобождены. Этот идентификатор должен стать недействительным для использования в последующих вызовах Рутин этого стандарта.

3.2.4 Среда с восстановлением ресурсов

Имя: FRSENV или FB_RESET_ENVIRONMENT

Категория: A

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие:

Среда, определенная идентификатором environment_id, должна быть восстановлена в состояние, которое определено по умолчанию Воплощением. В Воплощениях с исполнением Списка действий Среда должна содержать пустой или нулевой Список. Идентификатор environment_id должен оставаться действительным для последующих вызовов Рутин этого стандарта.

3.2.5 Копирование Среды

Имя: FCOENV или FB_COPY_ENVIRONMENT

Категория: E

Параметры:

return_code	FB-error_code	(OUT)
from_environment_id	FB_environment_id	(IN)
to_environment_id	FB_environment_id	(OUT) (см. текст)

Действие:

Эта Рутина должна создавать новую Среду, определяемую идентификатором to_environment_id и идентичную текущему состоянию Среды, которое выражается идентификатором from_environment_id. К идентификатору to_environment_id должны применяться пп. 3.2.1и 3.2.2 описания идентификатора environment_id соответственно для Сред с Режимами немедленного и отложенного исполнения.

3.2.6 Получение Среды

Имя: FGTENV или FB_GET_ENVIRONMENT

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
user_array	array_of_FB_word	(IN)
array_size	FB-integer	(IN)
byte_count	FB_integer	(OUT)

Действие:

Информация в Среде, определенной идентификатором environment_id, должна быть передана в массив user_array. Этот массив должен иметь длину array_size байтов. Действительное число переданных байтов должно быть возвращено в виде byte_count. Если user_array недостаточно велик,чтобы содержать определенную Среду, должно быть возвращено сообщение об ошибке, а возвращенное значение byte_count должно быть числом байтов, необходимых для безошибочной передачи.

3.2.7 Установка Среды

Имя: FSTENV или FB_SET_ENVIRONMENT

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
user_array	array_of_FB_word	(IN)
byte_count	FB_integer	(IN)

Действие:

Информация байтов byte_count в массиве user_array должна быть передана Среде, определенной идентификатором environment_id.

Информация должна быть запомнена в массиве user_array в предыдущем вызове FB_GET_EN-

VIRONMENT. Для Воплощений, распределяющих память, создается новая Среда, идентифицированная возвращенным параметром environment_id. Для Воплощений, в которых распределение памяти должен выполнять Пользователь, параметр environment_id указывает массив, определяемый Пользователем, который будет содержать описание Среды.

3.3 Отложенное исполнение и подтверждение правильности Списка

3.3.1 Исполнение Списка

Имя: FLEXEC или FB_LIST_EXECUTE

Категория: B

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие:

В Среде, определенной идетификатором environment_id, эта Рутина должна запускать подтверждение правильности Списка (если подтверждение не сделано раньше) и затем запускать исполнение Списка (см. раздел 2.2). Состояние Операционного параметра FB_PAR_DELAY_EXEC должно быть игнорировано.

Создается Статусная история (см. гл. 11).

3.3.2 Подтверждение правильности Списка

Имя: FLVAL или FB_LIST_VALIDATE

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие:

Успешное исполнение этой Рутины должно показать, если оно может быть индицировано, что, Список в Среде, определенной идентификатором environment_id, будет успешно исполнен при наличном идентификаторе port_id.

3.3.3 Установка текущего указателя в определенном Списке

Имя: FLGPTR или FB_LIST_GET_POINTER

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
list_pointer	FB_list_pointer	(OUT)

Действие:

Эта Рутина должна возвращать указатель на следующее свободное место в Списке, относящемся к рассматриваемой Среде.

После исполнения Списка возвращенное значение может быть использовано для извлечения статусной информации о Списке (см. гл. 11).

Глава 4. ОПЕРАЦИОННЫЕ ПАРАМЕТРЫ

4.1 Введение

Операционные параметры (далее ОП) позволяют скроить Среду для конкретных применений. Когда создают Среду, ей придают набор ОП, которые все имеют значение по умолчанию, возможно, зависящие от их назначения. Текущее значение ОП может быть изменено Рутиной FB_PAR_SET, а ее текущее значение определяется Рутинами FB_PAR_GET и FB_PAR_READ.

4.1.1 Операционные параметры в Режиме отложенного исполнения

Имеются два типа ОП, различие между которыми становится ясным только в Средах с Режимом отложенного исполнения. Параметры первого типа, Всеобщий и Обрабатывающий ошибки ОП (раздел 4.4), должны, по самой своей природе, оставаться неизменными в течение исполнения Списка. Параметры второго типа, Фастбас-протокола (раздел 4.5), относятся непосредственно к действиям в системе Фастбас и могут быть изменены Рутинами действия, входящими в список. Всеобщий и Обрабатывающий ошибки ОП могут быть изменены только Рутиной FB_PAR_SET, в то время как ОП Фастбас-протокола могут быть изменены как Рутиной FB_PAR_SET, так и Рутиной FB_PAR_WRITE.

Рутина FB_PAR_SET переопределяет текущее значение ОП и, следовательно, переопределяет исходную Среду для исполнения Списка (для Среды с Режимом отложенного исполнения). Рутина FB_PAR_WRITE позволяет изменить ОП Фастбас-протокола во время исполнения Списка, но без влияния на действия набора ОП, когда Список исполняется повторно. Эффект любого действия FB_PAR_WRITE может быть изменен между последующими исполнениями Списка изменением значений в соответствующем буфере данных. Рутина FB_PAR_READ позволяет опрашивать ОП Фастбас-протокола во время исполнения Списка.

4.2 Определение Операционных параметров

Операционные параметры определяют часть Среды. ОП, относящиеся к различным идентификаторам environment_id, полностью независимы. ОП представляется одним 32-битным значением. Некоторые ОП принимают значения только FB_TRUE и FB_FALSE. Некоторые ОП могут быть только читаемы. Установка только читаемого параметра в значение, фиксированное Воплощением, не должна рассматриваться как ошибка. Могут быть также и ОП, специфичные для Воплощения.

Краткая форма имен всех ОП, определенных ниже, должна быть узнаваема любым Воплощением.

От Воплощения, предлагаемого этим стандартом, не требуется доступ ко всем ОП. ОП, которые не поддерживаются программным или аппаратным Воплощением, если к ним обращаются, должны возвращать извещение об ошибке (см. гл. 12).

Воплощение должно определять значения по умолчанию для всех поддерживаемых параметров. Если значения по умолчанию определены в этом стандарте, то именно они должны быть использованы во всех Воплощениях.

Воплощение может ограничивать доступ к некоторым ОП (и возвращать код ошибки, указывающий на недостаточную привилегию для предпринятой операции, когда сделана попытка ее проведения). Это может быть особенно необходимо, когда Воплощение допускает множество Пользователей.

4.3 Рутины Операционных параметров

Для обеспечения доступа к ОП должны быть использованы следующие Рутины. Параметры всегда должны быть вызываемы по их именам, определенным последующими пунктами стандарта.

4.3.1 Операционный параметр инициализации

Имя: FBPINI или FB_PAR_INIT

Категория: A

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
parameter_id	FB_parameter_id	(IN)

Действие:

Эта Рутина должна вызвать восстановление ОП, специфицированного через parameter_id, к свому значению по умолчанию, определенному Воплощением в конкретной Среде. Если parameter_id придан Рутине FB_PAR_ALL (FPALL), то все параметры должны быть установлены в их значения по умолчанию.

4.3.2 Операционный параметр установки

Имя: FBPSET или FB_PAR_SET

Категория: A

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
parameter_id	FB_parameter_id	(IN)
parameter_value	FB_parameter_value	(IN)

Действие:

После вызова эта Рутина должна присваивать значения parameter_value Операционному параметру, специфицированному через parameter_id в конкретной Среде. После обращения к этой Рутине значение parameter_value должно стать текущим значением для названного ОП.

4.3.3 Операционный параметр получения

Имя: FBPGET или FB_PAR_GET

Категория: A

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
parameter_id	FB_parameter_id	(IN)
parameter_value	FB_parameter_value	(OUT)

Действие:

После вызова эта Рутина должна возвращать текущее значение названного ОП в конкретную Среду в качестве значения parameter_value.

4.3.4 Операционный параметр чтения

Имя: FBPRD или FB_PAR_READ

Категория: B, Рутина действия

Параметры:

return_code		FB_error_code	(OUT)	
environment_id		FB_environment_id	(IN)	
parameter_id		FB_parameter_id	(IN)	
buffer_mode		FB_buffer_mode	(IN)	
buffer_specifier		FB_word	(OUT)	(FBVAR)
	или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие:

Независимо от режима исполнения в конкретной Среде, эта Рутина должна во Время исполнения поместить текущее значение названного ОП в определенный буфер данных. Эта Рутина должна быть использована только для ОП, описанных в разделе 4.5.

4.3.5 Операционный параметр записи

Имя: FBPWRT или FB_PAR_WRITE

Категория: B, Рутина действия

Параметры:

return_code		FB_error_code	(OUT)	
environment_id		FB_environment_id	(IN)	
parameter_id		FB_parameter_id	(IN)	
buffer_mode		FB_buffer_mode	(IN)	
buffer_specifier		FB_word	(OUT)	(FBVAR)
	или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие:

Независимо от Режима исполнения в конкретной Среде, эта Рутина должна во Время исполнения изменять текущее значение названного ОП на значение Времени исполнения в определенном буфере данных. Действия FB_PAR_WRITE не должны переходить от одного исполнения Списка на последующее исполнение. Эта Рутина должна быть использована только для ОП, описанных в разделе 4.5.

4.3.6 Проталкивание текущих Операционных параметров в стек

Имя: FBPUSH или FB_PAR_PUSH

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие:

После вызова эта Рутина должна переместить текущие ОП конкретной Среды во внутренний стек ОП, специфичных для Среды.

4.3.7 Выталкивание текущих Операционных параметров из стека

Имя: FBPOP или FB_PAR_POP

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие:

После вызова эта Рутина должна заменить ОП названной Среды параметрами, извлеченными с верха внутреннего стека ОП, специфичных для Среды.

4.4 Всеобщий и Обрабатывающий ошибки Операционные параметры

Всеобщие операционные параметры это параметры, связанные с действиями или элементами, которые не относятся прямо к деталям действий собственно системы Фастбас (последние описаны в разделе 4.5). Всеобщими операционными параметрами являются, например, определенность Порта Фастбас, указание, ожидать ли завершения Фастбас-действия до возвращения служебной информации к вызывающему или нет, объемы информации и Режим исполнения в Среде. Обрабатывающие ошибки ОП используются для управления извещениями о статусе. Они обсуждаются более детально в гл. 11. Термины Суммарный статус и Дополнительный статус определены в разделе 11.3.

4.4.1 Идентификатор Порта контроллера

Имя: FPPORT или FB_PAR_PORT

Значение: Зависит от Воплощения

По умолчанию: Зависит от Воплощения

Этот ОП должен идентифицировать Контроллер и Порт Фастбас, через которые проходит любое действие. Допустимо, чтобы данный Контроллер имел более одного Порта Фастбас. Этот параметр, зависящий от Воплощения, определяет, какой Порт использовать для последующих запросов действий, когда в Фастбасе доступны несколько маршрутов.

В режиме отложенного исполнения:

Заметим, что замена Порта, используемого данной Средой после того как составлен Список действий, подлежащих исполнению, может привести к тому, что невведенные или незаконные

параметры будут присвоены Порту. Это затруднение может быть преодолено лишь при следующем вызове Рутины FB_LIST_EXECUTE.

4.4.2 Не ждать завершения действия

Имя: FPNOWT или FB_PAR_NO_WAIT

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот ОП установлен в FB_FALSE, то любая Рутина действия или вызов FB_LIST_EXECUTE должны выдать возвращение вызывающей Рутине только лишь сразу после завершения запрошенных действий - с ошибками или без ошибок. Если параметр установлен в FB_TRUE, возвращение вызывающей Рутине должно быть выполнено немедленно после того, как начались запрошенные действия, возможно до их завершения.

4.4.3 Отложенное исполнение

Имя: FPDLAY или FB_PAR_DELAY_EXEC

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE - для Среды, созданной с Режимом немедленного исполнения (только чтение)

FB_TRUE - для Среды, созданной с Режимом отложенного исполнения (чтение/запись)

Этот ОП определяет Режим исполнения в Среде. Он имеет значение FB_FALSE для Среды с Режимом немедленного исполнения или значение FB_TRUE для Среды с Режимом отложенного исполнения.

4.4.4 Объем Среды

Имя: FPENVS или FB_PAR_ENVIRONMENT_SIZE

Значение: Зависит от Воплощения

По умолчанию: Зависит от Воплощения

Этот параметр (только для чтения) должен выражать текущий объем Среды в байтах.

4.4.5 Генерация ограниченного статуса

Имя: FPNSTA или FB_PAR_NO_STATUS

Значения: FB_TRUE или FB_FALSE

По умолчанию: Зависят от Воплощения

Установка этого ОП в FB_TRUE разрешает более быстрое исполнение Рутин действия или FB_LIST_EXECUTE при пожертвовании детальной статусной информации о выполненных действиях.

Если этот ОП имеет значение FB_FALSE, то каждая Рутина действия или вызов

FB_LIST_EXECUTE должны сохранить как Суммарную, так и Дополнительную статусную информацию (см. гл. 11). При значении FB_TRUE должна быть сохранена лишь Суммарная статусная информация.

4.4.6 Возвратный код содержит только Серьезность

Имя: FPRSEV или FB_PAR_RETURN_SEVERETY

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Установка этого ОП в FB_TRUE разрешает более быстрое исполнение Рутин действия или FB_LIST_EXECUTE при пожертвовании более детального Возвратного кода.

Если этот ОП имеет значение FB_TRUE, Возвратный код, возвращаемый Рутиной, должен содержать лишь Серьезность любой встреченной ошибки. То есть он должен быть одним из кодов ошибок: FB_ERR_SUCCESS, FB_ERR_INFO, FB_ERR_WARN, FB_ERR_ERROR или FB_ERR_FATAL. Более детальная статусная информация может быть получена вызовом одной из Рутин, декодирующих статус (см. гл. 11).

Если этот ОП имеет значение FB_FALSE, Возвратный код должен быть Кодом ошибки, содержащим статусную информацию и соответствующую Серьезность.

4.4.7 Порог Серьезности для обработчика ошибок

Имя: FPHATH или FB_PAR_HANDLER_THRESHOLD

Значения: Любое значение типа FB_severity

По умолчанию: FB_SEV_WARNING

Этот ОП должен определять порог Серьезности, при котором будет вызван Обработчик ошибок Пользователя. Если в результате исполнения Действий или Списка действий, наиболее серьезный код ошибки равен или больше порога, то Обработчик ошибок Пользователя (с использованием Рутины FB_STATUS_CONNECT) должен быть вызван до того, как начнет работать Автоматический извещатель об ошибках.

4.4.8 Порог Серьезности для извещателя об ошибках

Имя: FPRETH или FB_PAR_REPORT_THRESHOLD

Значения: любое значение типа FB_severity

По умолчанию: FB_SEV_WARNING

Этот ОП должен определять порог Серьезности, при котором будет происходить автоматическое извещение об ошибках. Средства для извещения об ошибках зависят от Воплощения.

Если Серьезность Возвратного кода, возникающего в результате вызова любой Рутины этого стандарта (модифицированной любым вызовом Обработчика ошибок Пользователя), равна или больше этого порога, автоматическое извещение об ошибке или ошибках должно происходить до получения Возвратного кода от Рутины.

4.4.9 Порог Серьезности исключительной ситуации

Имя: FPEXTH или FB_PAR_EXCEPTION_THRESHOLD

Значения: любое значение типа FB_severity

По умолчанию: FB_SEV_ERROR

Сразу же после окончания любого автоматического извещения об ошибках, этот ОП должен определить порог Серьезности, при котором будет генерировано состояние исключительной ситуации - обычно прекращение программы. Детали этого состояния должны зависеть от Воплощения.

Если Серьезность Возвратного кода, возникающего в результате вызова любой Рутины этого стандарта (модифицированной любым вызовом Обработчика ошибок Пользователя), равна или больше этого порога, должно быть дано извещение об исключительной ситуации.

4.4.10 Порог Серьезности сообщения

Имя: FPMETH или FB_PAR_MESSAGE_THRESHOLD

Значения: Любое значение типа FB_severity

По умолчанию: FB_SEV_SUCCESS

Этот ОП должен определять порог Серьезности, при котором будет происходить автоматическое извещение об индивидуальных кодах ошибок.

Возвратный код, возникающий в результате вызова Рутины (модифицированной любым вызовом Обработчика ошибок Пользователя), и любые коды ошибок в Суммарном или Дополнительном статусах, чья Серьезность равна порогу или больше его, должны быть доложены.

4.4.11 Доклад сжатого сообщения

Имя: FPRSUM или FB_PAR_REPORT_TERSE

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Этот ОП должен управлять количеством статусной информации, которая должна быть доложена при любом вызове автоматического докладчика ошибок.

Если значение параметра FB_FALSE, будет доложена вся имеющаяся статусная информация, связанная с любым кодом ошибок, предназначенным для доклада. Если значение FB_TRUE, то автоматический докладчик ошибок передаст только одно сообщение.

Детали содержания и объем сообщения зависят от Воплощения.

4.4.12 Доклад о каждом действии

Имя: FBRACT или FB_PAR_REPORT_ACTIONS

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Этот ОП должен устанавливать, нужно ли автоматическому докладчику ошибок переда-

вать информацию об исполненных действиях.

Если значение FB_FALSE, такую информацию докладывать не нужно.

Если значение FB_TRUE, автоматический докладчик ошибок будет докладывать о каждом действии, когда оно происходит.

Детали и объем сообщения зависят от Воплощения.

4.5 Фастбас-протокол, ОП повтора и превышения лимита времени

Операционные параметры, связанные с Фастбас-протоколом, обусловливают действия в магистралях Фастбас, выполняемые Портом. Эти параметры могут быть доступны через Рутины FB_PAR_GET, FB_PAR_SET, FB_PAR_READ и FB_PAR_WRITE. Действия Рутин FB_PAR_READ и FB_PAR_WRITE могут быть отложены в зависимости от Среды, а действия FB_PAR_GET и FB_PAR_SET безотлагательны.

Стандарт Фастбас определяет четыре таймера в Мастере Фастбас. Для управления этими таймерами (когда они могут быть управляемы Воплощением) предусмотрены Операционные параметры. Определен также всеобщий программный таймер. По умолчанию все таймеры должны быть разрешены.

Если в течение Фастбас-цикла на линиях SS (статус Слуги) возникнет неожиданный сигнал, Воплощение может отозваться на ошибку повторением цикла. Средства для спецификаций этого отзыва описаны в гл. 11. Определены ОП для спецификации числа повторов и временного интервала между ними.

4.5.1 Уровень арбитража

Имя: FPARBL или FB_PAR_ARBITRATION_LEVEL

Значения: От 0 до 63

По умолчанию: Зависит от Воплощения

Этот параметр должен определять значение кода, который нужно использовать в Регистре уровня арбитража (CSR#8<05:00>) у соответствующего Мастера для последующего вызова Рутины действия. Кроме хранения Уровня арбитража Мастера, этот регистр содержит также информацию относительно протокола, который Мастеру нужно использовать при выставлении сигнала AR (см. стандарт Фастбас - ГОСТ 34.340- или МЭК 935, п. 6.3.1 и раздел 8.11).

Этот параметр, как и другие, указанные ниже и влияющие на процесс арбитража, следует устанавливать только через Процессор-хозяин.

4.5.2 Протокол Арбитража с обеспеченным доступом

Имя: FPASSA или FB_PAR_ASSURED_ACCESS

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Этот параметр должен определять состояние CSR#8<07> (бита обеспеченного доступа в Регистре уровня арбитража) для последующих вызовов Рутины действия. Если FB_TRUE, этот бит должен быть установлен, а если FB_FALSE, бит должен быть очищен. Когда CSR#8<07> =1, Мастер при выставлении AR должен следовать Протоколу обеспеченного доступа.

4.5.3 Протокол Арбитража с приоритетным доступом

Имя: FPPRIA или FB_PAR_PRIORITIZED_ACCESS

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Этот параметр должен определять состояние CSR#8<06> (бита Приоритетного доступа в Регистре уровня арбитража) для последующих вызовов Рутины действия. Если FB_TRUE, этот бит должен быть установлен, а если FB_FALSE, бит должен быть очищен. Когда CSR#8<06> =1, Мастер при выставлении AR должен следовать Протоколу приоритетного доступа.

4.5.4 Владение магистралью на время передачи целого блока

Имя: FPWBLK или FB_PAR_WHOLE_BLOCK

Значения: FB_TRUE или FB_FALSE

По умолчанию: Зависит от Воплощения

Этот параметр должен устанавливать, будет или не будет освобождена магистраль Фастбас во время любой последующей Операции передачи блока.

Если значение параметра FB_FALSE, Мастер может любым доступным ему способом разделить передачу целого блока на ряд передач последующих частей блока - блочков (см. п. 6.1.5).

Операционные параметры FB_PAR_HOLD_BUS, FB_PAR_HOLD_BUS_NO_AR, FB_PAR_HOLD_AS и FB_PAR_HOLD_ON_ERROR, определенные ниже, имеют преимущество, переопределяя этот параметр в решении, будет или не будет отменено Владение магистралью.

4.5.5 Размер блочка при передаче блока

Имя: FPBLSZ или FB_PAR_BLOCKLET_SIZE

Значения: Число Фастбас-циклов данных (фронтов DS)

По умолчанию: Зависит от Воплощения

Этот параметр должен определять максимальный размер блочка для последующих передач блоков.

4.5.6 Принудительная передача блока данных

Имя: FPPIPE или FB_PAR_PIPELINE

Значения: FB_TRUE_ или FB_FALSE

По умолчанию: FB_FALSE

Если этот параметр имеет значение FB_TRUE, любые последующие передачи блоков должны быть выполнены в Режиме принудительной передачи (MS = 3).

Временной интервал между фронтами сигнала (от DS(u) до DS(d)) получается от параметра FB_PAR_CLOCK, если Воплощение имеет к нему доступ.

4.5.7 Период циклов данных при Принудительной передаче блока данных

Имя: FPCLK или FB_PAR_CLOCK

Значения: Время между фронтами DS в наносекундах

По умолчанию: Зависит от Воплощения

Когда происходит Принудительная передача блока данных, фронты сигнала DS следуют с периодом, задаваемым Мастером. Если этот период может быть установлен программно, он должен быть определен этим параметром.

4.5.8 Устройство с фиксированным Адресом следующей передачи (NTA)

Имя: FPFNTA или FB_PAR_FIXED_NTA

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Когда значение этого параметра FB_TRUE, он должен быть использован для указания о том, что передача данных происходит к Устройству или от Устройства, которое не изменяет число в Регистре адреса следующей передачи (NTA) (например устройства, подобные ПВПВ).

Такое Устройство принимает или посылает множество слов данных из единственного Внутреннего адреса. Порт, передающий данные к или от такого Устройства, должен быть уверен в такой особенности, чтобы корректно повторить Операцию, если случится ошибка, а также в случае разбиения блочной передачи на несколько более коротких блочных передач.

Заметим, что любая Операция с таким Устройством может быть повторена корректно, если только в дополнение, Устройство имеет защитный буферный регистр, описанный в разделе 11.3 стандарта Фастбас. (Присутствие защитного буфера прозрачно для Мастера).

4.5.9 Передача данных короткими словами

Имя: FPSDW или FB_PAR_SHORT_DATA_WORD

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот параметр имеет величину FB_FALSE, каждое слово данных, передаваемое к или от определенного Буфера данных, должно быть шириной в 32 бита. Если этот параметр установлен в FB_TRUE, каждое слово данных должно иметь ширину, задаваемую параметром FB_PAR_SHORT_WORD_SIZE.

Биты FB_PAR_SHORT_WORD_SIZE передаются по Фастбас-линиям адрес/данные с низшими номерами битов. Остающиеся линии AD очищаются записью Фастбас и игнорируются при чтении. Этот параметр должен применяться только в Циклах данных, но не в Первичном или Вторичном адресных циклах.

4.5.10 Размер короткого слова

Имя: FPSWDS или FB_PAR_SHORT_WORD_SIZE

Значение: зависит от Воплощения

По умолчанию: зависит от Воплощения

Этот Операционный параметр выражается числом битов короткого слова, зависит от Воплощения и предназначен только для чтения.

4.5.11 Генерация контроля четности

Имя: FPPRTY или FB_PAR_PARITY

Значения:
FB_PARITY_ODD (FPPODD), генерируется нечетность
FB_PARITY_EVEN (FPPEVN), генерируется четность
FB_PARITY_NONE (FPPNON), четность не генерируется

По умолчанию: зависит от Воплощения (должно быть ODD или NONE)

Этот параметр должен использоваться портом Фастбас для управления генерацией четности.

Заметим, что по стандарту Фастбас создается либо проверка на нечетность, либо отсутствие проверки, поэтому четность генерировать не следует, за исключением генерации в целях диагностики.

4.5.12 Действие Географической адресации

Имя: FPEG или FB_PAR_EG_UP

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот ОП имеет значение FB_TRUE, Порт должен установить линию EG, чтобы обеспечить узнавание Географического адреса при выполнении Первичного адресного цикла в запрошенной Операции. Если же Порт не меет средств управления линией EG, этот параметр должен быть не поддерживаемым.

4.5.13 Без Цикла арбитража

Имя: FPNOAR или FB_PAR_NO_ARBITRATION

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот ОП имеет значение FB_TRUE, Цикл арбитража должен отсутствовать в запрошенной Передаче Фастбас.

Этот параметр игнорируется Рутинами примитивных действий в гл. 8.

4.5.14 Без Первичного адресного цикла

Имя: FPNOPA или FB_PAR_NO_PRIM_ADDR

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот ОП имеет значение FB_TRUE, в запрошенную Операцию не должен быть включен Первичный адресный цикл. Параметр Первичного адреса должен быть проигнорирован.

Если Операция, подлежащая исполнению, имеет этот параметр установленным в FB_TRUE и предварительно не было установлено соединение с адресом, должна быть возвращена ошибка и Операция не должна быть выполнена.

Этот параметр игнорируется Рутинами примитивных действий в гл. 8.

4.5.15 Без Вторичного адресного цикла для Операций в Области данных

Имя: FPNOSA или FB_PAR_NO_SEC_ADDR

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Этот ОП должен действовать только при Операциях в Области данных. Если он имеет значение FB_TRUE, Вторичный адресный цикл не должен быть выполнен в качестве части Фастбас-операции. Параметр Вторичного адреса должен быть проигнорирован. Если он имеет значение FB_FALSE, Вторичный адресный цикл должен быть выполнен.

Этот параметр игнорируется Рутинами примитивных действий в гл. 8.

4.5.16 Без Вторичного адресного цикла в Области регистров CSR

Имя: FPNOSC или FB_PAR_NO_SEC_ADDR_CSR

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Этот ОП должен действовать только при Операциях в Области регистров CSR. Если он имеет значение FB_TRUE, Вторичный адресный цикл не должен быть выполнен в качестве части Фастбас-операции. Параметр Вторичного адреса должен быть проигнорирован. Если он имеет значение FB_FALSE, Вторичный адресный цикл должен быть выполнен.

Заметим, что стандарт Фастбас требует почти во всех случаях, чтобы Операции в области CSR содержали Вторичный адресный цикл. Этот параметр игнорируется Рутинами примитивных действий в гл. 8.

4.5.17 Без Цикла данных

Имя: FPNODA или FB_PAR_NO_DATA_CYCLE

Значеня: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот параметр имеет значение FB_TRUE, Цикл данных не должен быть выполнен как часть запрошенной Фастбас-операции.

Этот параметр игнорируется Рутинами примитивных действий в гл. 8.

4.5.18 Удерживать Владение магистралью

Имя: FPHOLD или FB_PAR_HOLD_BUS

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот параметр имеет значение FB_TRUE, Владение магистралью должно удерживаться до завершения данной Операции, то есть сигнал GK должен быть выставленным. В противном случае Владение магистралью должно быть прекращено до или при завершении Операции, если только не выдается сигнал Фастбас-ошибки и к тому же параметр FB_PAR_HOLD_BUS_ON_ERROR имеет значение FB_TRUE.

Этот параметр игнорируется Рутинами примитивных действий в гл. 8.

4.5.19 Удерживать Владение магистралью, если не выставлен сигнал AR

Имя: FPHOAR или FB_PAR_HOLD_BUS_NO_AR

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот параметр имеет значение FB_TRUE, Владение магистралью должно удерживаться до завершения данной Операции, если сигнал AR не выставлен на магистрали. Как только обнаружено выставление сигнала AR, магистраль должна быть освобождена. Иначе говоря, Владение магистралью должно быть прекращено до или при завершении Операции. Владение должно удерживаться, если ОП FB_PAR_HOLD_BUS имеет значение FB_TRUE или выдается Фастбас-ошибка и при этом значение параметра FB_PAR_HOLD_BUS_ON_ERROR равно FB_TRUE.

Этот параметр игнорируется Рутинами примитивных действий в гл. 8.

4.5.20 Удерживать связку с адресом

Имя: FPHOAS или FB_PAR_HOLD_AS

Значение: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот параметр имеет значение FB_TRUE, любая Фастбас-связка с адресом должна поддерживаться и после окончания Операции, то есть сигнал AS должен быть выставленным. В противном случае связка с адресом должна быть прекращена при завершении Операции, если только не случается фатальная Фастбас-ошибка и не задействован параметр FB_PAR_HOLD_ON_ERROR.

Этот параметр игнорируется Рутинами примитивных действий в гл. 8.

4.5.21 Удерживать сигналы Мастера при ошибке

Имя: FPHOER или FB_PAR_HOLD_BUS_ON_ERROR

Значения: FB_TRUE или FB_FALSE

По умолчанию: FB_FALSE

Если этот параметр имеет значение FB_TRUE и Операция обрывается вследствие Фастбас-ошибки, Порт должен поддерживать любые сигналы, выставленные в настоящий момент. Сигналы должны поддерживаться до намеренного их снятия.

4.5.22 Долгий таймер

Имя: FPLOT или FB_PAR_LONG_TIMER

Значение: выдержка времени в наносекундах

По умолчанию: зависит от Воплощения

Если выдержка времени Долгого таймера может быть установлена программно, она должна быть определена этим параметром. Если она читаема, в этом параметре следует содержать ее значение. Если случается окончание лимита времени, должен быть возвращен соответствующий код ошибки.

Выдержка времени Долгого таймера будет также записана в регистр Порта CSR#1C (если регистр реализован).

Долгий таймер запускается при начале Действия или исполнения Списка (если он уже не работает благодаря значению FB_TRUE параметра FB_PAR_LONG_TIMER_ON). Он останавливается и сбрасывается при завершении Действия или исполнения Списка, если значение параметра FB_PAR_LONG_TIMER_ON не равно FB_TRUE (смотри п. 4.5.24).

Заметим, что Долгий таймер действует в течение всей Фастбас-передачи и таким образом позволяет функционировать таймерам Арбитража или GK.

4.5.23 Отключение Долгого таймера

Имя: FPDLOT или FB_PAR_DISABLE_LONG_TIMER

Значения: FB_FALSE или FB_TRUE

По умолчанию: FB_FALSE

Этот параметр должен быть установлен в значение FB_TRUE для отключения выдержки времени Долгого таймера (в бите CSR#9<04>).

4.5.24 Удерживать Долгий таймер

Имя: FPOLOT или FB_PAR_LONG_TIMER_ON

Значения: FB_FALSE или FB_TRUE

По умолчанию: FB_FALSE

Этот параметр должен быть установлен в FB_TRUE, если требуется подавить сброс Долгого таймера при завершении Действия или исполнения Списка и перезапустить таймер в начале

Действия или исполнения Списка (при этом таймер продолжает счет времени).

4.5.25 Таймер ожидания

Имя: FPWTT или FB_PAR_WT_TIMER

Значения: выдержка времени в наносекундах

По умолчанию: зависит от Воплощения

Если выдержка времени Таймера ожидания может быть установлена программно, она должна быть определена этим параметром. Если она читаема, ее следует возвращать с этим параметром. Этот таймер начинает работать, когда Мастер обнаруживает на линии WT выставленный сигнал, а когда сигнал WT не выставлен, таймер сбрасывается и запрещается. Если случается окончание этого лимита времени, должен быть возвращен соответствующий Код ошибки.

Выдержка Таймера ожидания будет также записана в регистр Порта CSR#1D (если он реализован).

4.5.26 Отключить Таймер ожидания

Имя: FPDWTT или FB_PAR_DISABLE_WT_TIMER

Значения: FB_FALSE или FB_TRUE

По умолчанию: FB_FALSE

Этот параметр должен быть установлен в FB_TRUE, если нужно устранить лимит времени в Таймере ожидания.

4.5.27 Адресный таймер

Имя: FPAKT или FB_PAR_AK_TIMER

Значения: выдержка времени в наносекундах

По умолчанию: зависит от Воплощения

Если выдержка времени в Адресном таймере может быть установлена программно, она должна быть определена этим параметром. Если она читаема, ее следует возвращать с этим параметром. Когда бы Порт ни генерировал фронт сигнала AS (AS(u) или AS(d)), таймер запускается для получения подтверждающего фронта сигнала AK. Если случается окончание лимита времени, должен быть возвращен соответствующий Код ошибки.

Выдержка времени Адресного таймера будет также записана в регистр Порта CSR#1E (если он реализован).

4.5.28 Отключить Адресный таймер

Имя: FPDAKT или FB_PAR_DISABLE_AK_TIMER

Значения: FB_FALSE или FB_TRUE

По умолчанию: FB_FALSE

Этот параметр должен быть установлен в FB_TRUE, если нужно устранить лимит

времени в Адресном таймере (в регистре CSR#9<06>).

4.5.29 Таймер данных

Имя: FPDKT или FB_PAR_DK_TIMER

Значения: выдержка времени в наносекундах

По умолчанию: зависит от Воплощения

Если выдержка времени в Таймере данных может быть установлена программно, она должна быть определена этим параметром. Если она читаема, ее следует возвращать с этим параметром. Когда бы Порт ни генерировал фронт сигнала DS, таймер должен запускаться для приема соответствующего фронта сигнала DK. Если случается окончание лимита времени, должен быть возвращен соответствующий Код ошибки.

Выдержка времени Таймера данных будет также записана в регистр Порта CSR#1F (если он реализован).

4.5.30 Отключить Таймер данных

Имя: FPDDKT или FB_PAR_DISABLE_DK_TIMER

Значения: FB_FALSE или FB_TRUE

По умолчанию: FB_FALSE

Этот параметр должен быть установлен в FB_TRUE, если нужно устранить лимит времени в Таймере данных (в регистре CSR#9<07>).

4.5.31 Программный таймер

Имя: FPSOFT или FB_PAR_SOFT_TIMER

Значения: выдержка времени в миллисекундах

По умолчанию: зависит от Воплощения

Этот параметр обеспечивает единообразное определение одного или нескольких Программных таймеров, которые будут содержаться в любом Воплощении. Он дает средство для возврата управления вызывающей Рутине, когда Действие или Список оказываются слишком длинными для исполнения.

В сети Фастбас этот параметр установкой бита в регистре CSR#9 не устраняется.

Истечение лимита времени Программного таймера не вызывает никаких изменений в состоянии системы Фастбас.

Программный таймер запускается в начале Действия или исполнения Списка (если он уже не работает благодаря значению FB_TRUE параметра FB_PAR_SOFT_TIMER_ON), или когда его значение установлено с помощью параметра FB_PAR_SOFT_TIMER. Таймер останавливается и сбрасывается при окончании Действия или Списка, если значение параметра FB_PAR_SOFT_TIMER_ON не равно FB_TRUE (см. п. 4.5.33).

4.5.32 Отключить Программный таймер

Имя: FPDSOT или FB_PAR_DISABLE_SOFT_TIMER

Значения: FB_FALSE или FB_TRUE

По умолчанию: FB_FALSE

Этот параметр должен быть установлен в FB_TRUE, если нужно устранить лимит времени в Программном таймере.

4.5.33 Удержать Программный таймер

Имя: FPOSOT или FB_PAR_SOFT_TIMER_ON

Значения: FB_FALSE или FB_TRUE

По умолчанию: FB_FALSE

Этот параметр должен быть установлен в FB_TRUE, если нужно подавить сброс таймера при окончании Действия или Списка и перезапустить таймер в начале Действия или исполнения Списка.

4.5.34 Число повторений

Имя: FPRTRY или FB_PAR_NUM_RETRY

Значения: число повторений

По умолчанию: 0

Этот параметр должен определять число повторений Фастбас-циклов, приводящих к выдаче Кода ошибки с Серьезностью, не равной значению FB_SEV_SUCCESS, если соответствующий Ответный код (см. раздел 11.6) указывает, что следует предпринять повторения.

Заметим, что если желательно большее число повторений, то на уровне Пользователя могут быть повторены все действия.

4.5.35 Время повторения

Имя: FPRTRT или FB_PAR_ADDRESS_CYCLE_RETRY_TIME

Значения: среднее время выдержки в наносекундах

По умолчанию: зависит от Воплощения

Когда должны быть повторены Цикл данных или Вторичный адресный цикл, повторение должно быть выполнено как можно быстрее. Однако, когда должен быть повторен Первичный адресный цикл, иногда необходимо выжидать конечный отрезок времени между каждыми попытками. Если код SS=1 (сеть занята), Мастер должен не повторять адресный цикл немедленно, чтобы избежать возможного системного зависания. Этот параметр применяют для определения среднего значения случайнго выжидания, которое следует вводить между повторениями.

Глава 5. БУФЕРЫ ДАННЫХ

5.1 Аргументы доступа к Буферам

Для каждой Рутины действия, выполняющей передачу данных к «Буферу данных» или от Буфера, режим передачи и местоположение Буфера определяет пара аргументов. Первый аргумент пары («режим передачи») устанавливает, как должен быть интерпретирован второй аргумент. Второй аргумент пары («спецификатор буфера») является либо идентификатором Последовательного буфера, либо переменной или идентификатором массива, либо самим значением данных. Допустимые Режимы буфера, определенные в гл. 2, описаны ниже.

5.1.1 Буфер специфицирован в качестве переменной или массива

Поддержание этого Режима буфера должно быть предусмотрено для Режима немедленного исполнения. Оно не обязательно для Режима отложенного исполнения.

Когда параметр Режима буфера имеет величину FB_BUFFER_VAR, спецификатор Буфера должен быть переменной. Во Время исполнения данные должны быть записаны от или прочитаны на определенную переменную.

5.1.2 Буфер специфицирован в качестве значения

Этот Режим буфера должен допускаться только для Операций записи единственного слова и должен быть предусмотрен для Сред с Режимом отложенного исполнения. Он не обязателен для Сред с Режимом немедленного исполнения.

Когда параметр режима буфера имеет величину FB_BUFFER_VALUE, спецификатор буфера должен иметь значение в виде одного слова данных, значение которого в Момент вызова приписывается Времени исполнения.

5.1.3 Буфер специфицирован в качестве идентификатора Последовательного буфера

Поддержание этого Режима буфера должно быть предусмотрено для Режима отложенного исполнения.

Когда параметр Режима буфера установлен на FB_BUFFER_SEQ, последующие Фастбас-действия при том же спецификаторе буфера (идентификатор последовательного буфера) выполняют поочередно запись в или чтение из того же самого Последовательного буфера. В этой главе описаны Рутины FB_READ_BUFFER_POINTER, FB_WRITE_BUFFER_POINTER, и FB_DISPLACE_BUFFER_POINTER, обеспечивающие доступ к указателю местоположения текущих данных в Последовательном буфере.

Может быть объявлен тип Последовательного буфера - Внутренний или Внешний.

«Внутренний буфер» должен быть доступен для вызывающей программы непосредственно. (Поскольку в этом стандарте не предусмотрен механизм для достижения параметров в разных Средах, ссылки на Внутренние буферы данных должны происходить в каждом случае только из одной Среды).

По желанию возможно создание «Внешних буферов», в которых Буфер данных может и не быть непосредственно доступен для программы, которая его использует.

Значение идентификатора буфера определяется Пользователем и должно быть единственным внутри Среды. «Последовательный буфер» должен быть объявлен с использованием одной из Рутин, описанных ниже, до исполнения любой операции, в которой имеется ссылка на буфер.

5.2 Рутины Последовательных буферов данных

5.2.1 Объявить Внутренний буфер

Имя: FBDSEQ или FB_DECLARE_SEQ_BUFFER

Категория: B

Параметры:

return code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
seq_buffer_id	FB_seq_buffer_id	(IN)
buffer_array	FB_word_array	(IN)
buffer_size	FB_integer	(IN)

Действие:

Эта Рутина должна связывать специфицированный идентификатор Последовательного буфера (seq_buffer_id) со специфицированным идентификатором Внутреннего буфера (buffer_array), содержащим байты размера буфера (buffer_size). Указатель буфера сбрасывается. Внутренний буфер является локальным для вызывающей программы и доступен для нее непосредственно. Фастбас-действия с их параметрами buffer-mode, установленным в FB_BUFFER_SEQ, и buffer_specifier, установленным в то же значение, что и значение seq_buffer_id, должны использовать тот же самый Внутренний буфер.

5.2.2 Объявить Внешний буфер

Имя: FBDEXT или FB_DECLARE_EXT-SEQ_BUFFER

Категория: B

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
seq_buffer_id	FB_seq_buffer_id	(IN)
buffer_location	FB_word_array	(IN)
buffer_size	FB_integer	(IN)

Действие:

Эта Рутина должна связывать специфицированный идентификатор Последовательного буфера со специфицированным Внешним буфером, который характеризуется аргументом buffer_location размером buffer_size байтов. Буфер является внешним по отношению к Пользователю и может не быть непосредственно доступен для него (буфер может быть размещен, например, в Порте). Любые Фастбас-операции с их параметрами buffer_mode, установленным в значение FB_BUFFER_SEQ, и buffer_specifier, установленным в то же значение, что и seq_buffer_id, должны использовать один и тот же Внешний буфер. Параметры buffer_location и buffer_size должны зависеть от Воплощения.

5.2.3 Отключить Внутренний последовательный буфер данных

Имя: FBREL или FB_RELEASE_SEQ_BUFFER

Категория: B

Параметры:

return_code		FB-error_code	(OUT)
environment_id		FB_environment_id	(IN)
seq_buffer_id		FB_seq_buffer_id	(IN)

Действие:

Эта Рутина при будущем использовании должна отключать специфицированный идентификатор буфера. Его значение становится недействительным до тех пор, пока он не будет заново объявлен как Внутренний или Внешний буфер, как описано выше.

5.2.4 Считать указатель Последовательного буфера

Имя: FBRPTR или FB_READ_BUFFER_POINTER

Категория: E, Рутина действия

Параметры:

return_code		FB_error_code	(OUT)	
environment_id		FB_environment_id	(IN)	
seq_buffer_id		FB_seq_buffer_id	(IN)	
buffer_mode		FB_buffer_mode	(IN)	
buffer_specifier		FB_word	(OUT)	(FBVAR, FBVAL)
	или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие:

Эта Рутина должна копировать текущий указатель Последовательного буфера, связанный с параметром seq_buffer_id, в Буфер, специфицированный параметрами доступа к буферу buffer_mode и buffer_specifier. Значение указателя Последовательного буфера должно быть выражено в байтах относительно начала Последовательного буфера. При указании на первое место Последовательного буфера значение указателя должно быть равным нулю.

5.2.5 Записать указатель Последовательного буфера

Имя: FBWPTR или FB_WRITE_BUFFER_POINTER

Категория: E, Рутина действия

Параметры:

return_code		FB_error_code	(OUT)	
environment_id		FB_environment_id	(IN)	
seq_buffer_id		FB_seq_buffer_id	(IN)	
buffer_mode		FB_buffer_mode	(IN)	
buffer_specifier		FB_word	(IN)	(FBVAR, FBVAL)
	или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие:

Указатель данных в специфицированном Последовательном буфере устанавливается в значение, содержащееся в Буфере данных, специфицированном параметрами доступа к буферу buffer_mode и buffer_specifier. Специфицированное значение должно быть выражено в байтах относительно начала Последовательного буфера. При указании на первое место Последовательного буфера значение указателя должно быть равно нулю.

5.2.6 Сместить указатель Последовательного буфера

Имя: FBDPTR или FB_DISPLACE_BUFFER_POINTER

Категория: E, Рутина действия

Параметры:

return_code		FB_error_code	(OUT)	
environment_id		FB_environment_id	(IN)	
seq_buffer_id		FB_seq_buffer_id	(IN)	
buffer_mode		FB_buffer_mode	(IN)	
buffer_specifier		FB_word	(IN)	(FBVAR, FBVAL)
	или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие:

Во Время исполнения эта Рутина должна сместить текущий указатель Последовательного буфера, связанный с параметром seq_buffer_id, на величину (в байтах), содержащуюся в Буфере данных, специфицированном параметрами доступа к Буферу buffer_mode и buffer_specifier.

Глава 6. РУТИНЫ ПРОСТЫХ ПЕРЕДАЧ.

Рутины, описанные в этой главе, вызывают передачи Фастбас, которые являются последовательностями циклов Фастбас.

6.1 Условия, управляющие передачами, и сложные Рутины

В этом разделе представлены условия, при которых различные циклы Фастбас явно или неявно вызываются Рутинами, описанными в этой и следующих главах. Сами циклы описаны в стандарте Фастбас.

6.1.1 Арбитраж

Арбитраж должен выполняться автоматически, если Порт не владеет магистралью и если параметр FB_PAR_NO_ARBITRATION имеет зачение FB_FALSE . Если этот параметр имеет значение FB_TRUE или если Порт уже владеет магистралью, арбитраж должен подавляться.

Используемый уровень арбитража должен быть определен ОП FB_PAR_ARBITRATION_LEVEL.

Перед установкой AR Мастер должен учитывать состояние линии AI, если ОП FB_PAR_ASSURED_ACCESS имеет значение FB_TRUE, или же учитывать уровень арбитража текущего мастера, если параметр FB_PAR_PRIORITIZED_ACCESS имеет значение FB_TRUE.

6.1.2 Первичный адресный цикл

Первичный адресный цикл должен выполняться до тех пор, пока ОП FB_PAR_NO_PRIM_ADDR не будет равен FB_TRUE, в то время как вызвана Рутина Действия. Используемый Первичный адрес должен быть (во время вызова) величиной параметра «primary_address» в Рутине.

Если ОП FB_PAR_EG_UP имеет значение FB_TRUE, в то время как вызвана Рутина, линия магистрали EG должна быть, в случае соответствия, установлена Портом во время Первичного адресного цикла.

Автоматический возврат по ошибке должен быть определен путем использования Рутины FB_STATUS_RESPONSE, описанной в гл. 11.

После каждого Первичного адресного цикла магистраль должна быть освобождена (т.е. GK сброшен) перед или в течение последнего цикла данных Операции, если только во время вызова Рутины не были выполнены следующие условия:

параметр FB_PAR_HOLD_BUS имел значение FB_TRUE или

параметр FB_PAR_HOLD_BUS_ON_ERROR имел значение FB_TRUE и встретилась ошибка, которая остановит дальнейшее выполнение Операций Фастбас, или

параметр FB_PAR_HOLD_BUS_NO_AR имел значение FB_TRUE и было обнаружено, что AR не установлен.

С Первичным адресным циклом связаны следующие состояния с ошибками:

FEAKTO	или FB_ERR_AK_TIMEOUT	Нет AK-отклика в течение лимита времени
FEASSx	или FB_ERR_PRIMARY_ADDRESS_SSx	Неожидаемый SS-отклик

6.1.3 Вторичный адресный цикл записи

Если ОП FB_PAR_NO_SEC_ADDR имеет значение FB_FALSE, для Рутин Области адресов данных за Первичным адресным циклом должен следовать Вторичный адресный цикл. Если этот ОП имеет значение FB_TRUE, Вторичный адресный цикл должен быть пропущен и параметр «primary_address» должен игнорироваться.

Если ОП FB_PAR_NO_SEC_ADDR_CSR имеет значение FB_FALSE, для Рутин Области адресов регистров управления и состояний (CSR) за Первичным адресным циклом должен следовать Вторичный адресный цикл. Если этот ОП имеет значение FB_TRUE, Вторичный адресный цикл должен быть пропущен и параметр «primary_address» должен игнорироваться.

Используемый Вторичный адрес должен быть (во время вызова) величиной параметра «secondary_address» в Рутине (если не утверждается обратное в описании Рутины).

Автоматические повторные попытки в случае возникновения ошибок должны быть определены с использованием Рутины FB_STATUS_SET_RESPONSE, описанной в гл. 11.

Со Вторичными адресными циклами связаны следующие состояния с ошибками:

FESATO	или	FB_ERR_SECONDARY_ADDRESS_TIMEOUT	Нет DK-отклика в течение лимита времени
FEESSx	или	FB_ERR_SECONDARY_ADDRESS_SSx	Неожидаемый SS отклик.

6.1.4 Циклы данных

Циклы данных вызываются большинством Рутин, описанных в этой главе, как установлено ниже для каждой Рутины, если только FB_PAR_NO_DATA_CYCLE не имеет значения FB_TRUE при вызове Рутины.

Автоматические повторные попытки в случае возникновения ошибок должны быть определены с использованием Рутины FB_STATUS_SET_RESPONSE, описанной в гл. 11.

Параметр FB_PAR_SHORT_DATA_WORD, будучи равным FB_TRUE, вызывает в течение цикла данных упаковку в буфер данных или распаковку из буфера данных слов длиной менее 32 бит.

ОП FB_PAR_PIPELINE должен определять, выполняется ли Рутинами передач блоков Передача блока или Принудительная передача.

С Циклами данных связаны следующие состояния с ошибками:

FEDKTO	или	FB_ERR_DK_TIMEOUT	Нет DK-отклика в течение установленного времени
FEDSSx	или	FB_ERR_DATA_SSx	Неверный SS-отклик
FEBSS2	или	FB_ERR_BLOCK_DATA_SS2	SS2-отклик на Передачу блока.

6.1.5 Передачи блоков

Управляя передачей блока, Мастер может следовать одной из нескольких стратегий, в зависимости от его возможностей в отношении аппаратного обеспечения и величин некоторых ОП.

Мастер должен передавать целый блок за одну непрерывную операцию, если ОП FB_PAR_WHOLE_BLOCK имеет значение FB_TRUE.

Если этот параметр имеет значение FB_FALSE, Мастер может делить передачу целого

блока на передачу нескольких блочков меньшего размера. Максимальный размер этих блочков определяется ОП FB_PAR_BLOCKLET_SIZE.

После передачи каждого из этих блочков Мастер должен либо освободить магистраль и принять участие в следующем цикле арбитража, либо проверить состояние AR и освободить магистраль, если только определено, что AR установлен.

Мастеры, которые способны реагировать на состояние AR, могут вместо использования блочков фиксированого размера непрерывно контролировать состояние AR в течение передачи и реагировать надлежащим образом, когда устанавливается AR.

6.1.6 Разъединение

После того как Операция вызвана Рутиной, описанной в этой главе, должно произойти полное разъединение со Слугой, если только не были выполнены следующие условия: параметр FB_PAR_HOLD_AS имел значение FB_TRUE во время вызова, или параметр FB_PAR_HOLD_BUS_ON_ERROR имел значение FB_TRUE и встретилась ошибка, которая остановит дальнейшее выполнение Операций Фастбас.

6.2 Рутины простых передач Фастбас

6.2.1 Передача одиночного слова Данных

Имя:

FRD или FB_READ_DAT	Считать слово из Области адресов данных
FWD или FB_WRITE_DAT	Записать Слово в Область адресов данных
FRC или FB_READ_CSR	Считать Слово из Области адресов CSR
FWC или FB_WRITE_CSR	Записать Слово в Область адресов CSR
FRDM или FB_READ_DAT_MULT	Считать Слово, Область адресов данных, широкий адрес
FWDM или FB_WRITE_DAT_MULT	Записать Слово, Область адресов данных, широкий адрес
FRCM или FB_READ_CSR_MULT	Считать Слово, Область адресов регистров CSR, широкий адрес
FWCM или FB_WRITE_CSR_MULT	Записать Слово, Область адресов регистров CSR, широкий адрес

Категория: A, Рутина Действия

Параметры:

return_code		FB_error_code	(OUT)		
environment_id		FB_environment_id	(IN)		
primary_address		FB_word	(IN)		
secondary_address		FB_word	(IN)		
buffer_mode		FB_buffer_mode	(IN)		
buffer_specifier		FB_word	(IN)	FWxx	(FBVAR, FBVAL)
	или	FB_word	(OUT)	FRxx	(FBVAR)
	или	FB_seq_buffer_id	(IN)		(FBSEQ)

Действие:

Должна происходить передача одиночного слова данных.

Код MS выбора режима передачи, использованный для Первичного адресного цикла, и состояние линии чтения RD для цикла данных должны устанавливаться в зависимости от вызываемых Рутин следующим образом:

Рутина	MS	RD
FB_READ_DAT	MS=0	RD=1
FB_WRITE_DAT	MS=0	RD=0
FB_READ_CSR	MS=1	RD=1
FB_WRITE_CSR	MS=1	RD=0
FB_READ_DAT_MULT	MS=2	RD=1
FB_WRITE_DAT_MULT	MS=2	RD=0
FB_READ_CSR_MULT	MS=3	RD=1
FB_WRITE_CSR_MULT	MS=3	RD=0

Цикл Данных должен иметь значение MS=0.

Должно употребляться управление исполнением отдельных циклов, определенное в разделе 6.1.

6.2.2 Передача Блока

Имя:

FRDB или FB_READ_DAT_BLOCK	Считать Блок, Область адресов данных
FWDB или FB_WRITE_DAT_BLOCK	Записать Блок, Область адресов данных
FRCB или FB_READ_CSR_BLOCK	Считать Блок, Область адресов CSR
FWCB или FB_WRITE_CSR_BLOCK	Записать Блок, Область адресов CSR
FRDBM или FB_READ_DAT_BLOCK_MULT	Считать Блок, Область адресов данных, широкий адрес
FWDBM или FB_WRITE_DAT_BLOCK_MULT	Записать Слово, Область адресов данных, широкий адрес
FRCBM или FB_READ_CSR_BLOCK_MULT	Считать Слово, Область адресов CSR, широкий адрес
FWCBM или FB_WRITE_CSR_BLOCR_MULT	Записать Слово, Область адресов CSR, широкий адрес

Категория: A, Рутина Действия

Параметры:

return_code		FB_error_code	(OUT)		
environment_id		FB_environment_id	(IN)		
primary_address		FB_word	(IN)		
secondary_address		FB_word	(IN)		
buffer_mode		FB_buffer_mode	(IN)		
buffer_specifier		array_of_FB_word	(IN)	FWxx	(FBVAR)
	или	array_of_FB_word	(OUT)	FRxx	(FBVAR)
	или	FB_seq_buffer_id	(IN)		(FBSEQ)
max_bytes		FB_integer	(IN)		

Действие:

Должна происходить передача блока.

Код MS выбора режима передачи, использованный для Первичного адресного цикла, и значение RD линии чтения для передачи блока должны устанавливаться в зависимости от вызываемых Рутин следующим образом:

Рутина	MS	RD
FB_READ_DAT_BLOCK	MS=0	RD=1
FB_WRITE_DAT_BLOCK	MS=0	RD=0
FB_READ_CSR_BLOCK	MS=1	RD=1
FB_WRITE_CSR_BLOCK	MS=1	RD=0
FB_READ_DAT_BLOCK_MULT	MS=2	RD=1
FB_WRITE_DAT_BLOCK_MULT	MS=2	RD=0
FB_READ_CSR_BLOCK_MULT	MS=3	RD=1
FB_WRITE_CSR_BLOCK_MULT	MS=3	RD=0

Код MS выбора режима передачи, использованный для передачи блока, должен зависеть от значения ОП FB_PAR_PIPELINE. Если значение этого параметра равно FB_FALSE, то MS=1. Если значение этого параметра равно FB_TRUE, то MS=3.

Должно употребляться управление исполнением отдельных циклов, определенное в разделе 6.1.

6.2.3 Рутины Вторичного адреса

Имя:

FRDSA или FB_READ_DAT_SA	Считать Вторичный адрес в Области адресов данных
FWDSA или FB_WRITE_DAT_SA	Записать Вторичный адрес в Области адресов данных
FRCSA или FB_READ_CSR_SA	Считать Вторичный адрес в Области адресов данных CSR
FWCSA или FB_WRITE_CSR_SA	Записать Вторичный адрес в Области адресов данных CSR

Категория: A, Рутина Действия

Параметры:

return_code		FB_error_code	(OUT)		
environment_id		FB_environment_id	(IN)		
primary_address		FB_word	(IN)		
buffer_mode		FB_buffer_mode	(IN)		
buffer_specifier		FB_word	(IN)	FWxSA	(FBVAR, FBVAL)
	или	FB_word	(OUT)	FRxSA	(FBVAR)
	или	FB_seq_buffer_id	(IN)		(FBSEQ)

Действие:

Должна производиться передача одиночного слова данных в Регистр адреса следующей передачи NTA или из него. Код MS выбора режима передачи, использованный для Первичного адресного цикла, и значение RD линии чтения для Вторичного адресного цикла должны устанавливаться в зависимости от вызываемых Рутин следующим образом:

Рутина	MS	RD
FB_READ_DAT_SA	MS=0	RD=1
FB_WRITE_DAT_SA	MS=0	RD=0
FB_READ_CSR_SA	MS=1	RD=1
FB_WRITE_CSR_SA	MS=1	RD=0

В Цикле данных должно иметь место MS=2. Должно употребляться управление исполнением отдельных циклов, определенное в разделе 6.1, за исключением Вторичного адресного цикла, включающего считывание или запись с передачей данных в или из Буфера данных во Время исполнения.

ОП FB_PAR_NO_SEC_ADDR должен быть активным. Так, если этот ОП установлен в FB_TRUE, за Первичным адресным циклом не последует никакого действия, отличного от завершающей последовательности.

6.2.4 Считать длину последней передачи данных

Имя: FRLEN или FB_READ_LENGTH

Категория: E, Рутина Действия

Параметры:

return_code		FB_error_code	(OUT)	
environment_id		FB_environment_id	(IN)	
buffer_mode		FB_buffer_mode	(IN)	
buffer_specifier		FB_word	(OUT)	(FBVAR)
	или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие:

При окончании времени исполнения в Буфер данных Рутины FB_READ_LENGTH должно быть возвращено число байтов, переданных в или из буфера пользователя во время последнего Фастбас-действя (в специфицированной среде) с использованием спецификатора буфера пользователя.

Глава 7. РУТИНЫ СЛОЖНЫХ ПЕРЕДАЧ

Определяемые в этой главе Рутины являются подмножеством большого количества возможных сложных Передач Фастбас. Отдельные Фастбас-действия выполняются при тех же самых условиях, которые описаны в разделе 6.1.

7.1 Доступ к маршрутной таблице Соединителей сегментов

Возможные варианты доступа к маршрутной таблице Соединителя сегментов (СС) определяются следующими рутинами.

Имя:

FRRT	или	FB_READ_ROUTE_TABLE (одиночное слово)
FWRT	или	FB_WRITE_ROUTE_TABLE (одиночное слово)
FRRTB	или	FB_READ_ROUTE_TABLE_BLOCK
FWRTB	или	FB_WRITE_ROUTE_TABLE_BLOCK

Категория: E, Рутина Действия

Параметры:

return_code	FB_error_code	(OUT)		
environment_id	FB_environment_id	(IN)		
primary_address	FB_word	(IN)		
route_address	FB_word	(IN)		
buffer_mode	FB_buffer_mode	(IN)		
buffer_specifier	FB_word	(IN)	FWRT	(FBVAL, FBVAR)
или	FB_word	(OUT)	FRRT	(FBVAR)
или	array_of_FB_word	(IN)	FWRTB	(FBVAR)
или	array_of_FB_word	(OUT)	FRRTB	(FBVAR)
или	FB_seq_buffer_id	(IN)	любой	(FBSEQ)
max_bytes	FB_integer	(IN)	FxRTB только	

Действие:

Порт слуги в СС при географическом адресе, определяемом ОП primary_address, должен быть адресован в Области адресов CSR. За Вторичным адресным циклом со значением адреса 40h следует Цикл записи данных при MS=0 и значении ОП route_address, который специфицирует (начальный) адрес для доступа в маршрутную таблицу. За Циклом записи данных должен следовать Вторичный адресный цикл со значением адреса 41h - регистр данных Маршрутной таблицы. Для Рутин FB_READ_ROUTE_TABLE и FB_WRITE_ROUTE_TABLE должен быть выполнен Цикл данных при MS=0, а для рутин FB_READ_ROUTE_TABLE_BLOCK и FB_WRITE_ROUTE_TABLE_BLOCK должна быть выполнена передача блока при MS=1. Данные должны быть переданы в специфицируемый Буфер данных или из специфицированного Буфера данных.

7.2 Передача данных между Фастбас-устройствами

Имя:

FMOVD или FB_MOVE_DAT	Передача слова в Область адресов данных
FMOVC или FB_MOVE_CSR	Передача слова в Область адресов CSR
FMOVDB или FB_MOVE_DAT_BLOCK	Передача блока в Область адресов данных
FMOVCB или FB_MOVE_CSR_BLOCK	Передача блока в Область адресов CSR

Категория: E, Рутина действия

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
primary_address1	FB_word	(IN)

secondary_address1		FB_word	(IN)		
primary_address2		FB_word	(IN)		
secondary_address2		FB_word	(IN)		
buffer_mode		FB_buffer_mode	(IN)	FMOVxB	только
buffer_specifier		FB_word	(OUT)	FMOVxB	(FBVAR)
	или	FB_seq_buffer_id	(IN)	FMOVxB	(FBSEQ)
max_bytes		FB_integer	(IN)	FMOVxB	только

Действие:

Со Слугой должна быть выполнена Операция чтения (слова или блока), использующая ОП primary_address1 и secondary_address1. Если чтение завершено без ошибки, то считанные этой Операцией данные должны быть затем записаны другому Слуге с использованием параметров primary_address2 и secondary_address2. Если в качестве промежуточного Буфера данных используется Последовательный буфер (Sequential Buffer), то указатель буфера должен быть перемещен со считывания на запись так, чтобы считанные данные не могли быть записаны вновь.

Соображения, изложенные в разделе 6.1, должны быть применены как для Операций считывания, так и для Операций записи.

При передаче одиночного слова данные должны быть считаны во внутренний Буфер данных, как определяется в Воплощении, т.е. в вызов Рутины передачи не будут включены ОП, определяющие буфер. Для передачи блока необходимо специфицировать промежуточный буфер.

7.3 Размещение Фастбас-информации при Чтении - Модификации - Записи

Имя:

FMODD или FB_MODIFY_DAT	Модификация слова в Области адресов данных
FMODC или FB_MODIFY_CSR	Модификация Слова в Области адресов регистров CSR

Категория: E, Рутина Действия

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
primary_address	FB_word	(IN)
secondary_address	FB_word	(IN)
operator_id	FB_operator	(IN)
operand	FB_word	(IN)

Действие:

Должно быть считано слово из регистра Слуги, специфицированного ОП primary_address и secondary_address. Владение магистралью должно поддерживаться, пока выполняется арифметическая или логическая операция, специфицированная ОП operator_id. При выполнении операции используется значение, считанное как первый параметр, и операнд, как второй параметр в ОП operator_id. Результат этой операции записывается в тот же самый регистр Слуги. Затем Владение магистралью снимается.

Значения параметра operator_id приводятся в следующей таблице:

Краткое имя	Полное имя	Значение
FMSET	FB_MOD_SET	Установить позиции битов в соответствии с операндом
FMCLR	FB_MOD_CLEAR	Сбросить позиции битов в соответствии с операндом

FMAND	FB_MOD_AND	Выполнить поразрядную операцию «И» между операндом и считанными данными
FMOR	FB_MOD_OR	Выполнить поразрядную операцию «ИЛИ» между операндом и считанными данными
FMXOR	FB_MOD_XOR	Выполнить поразрядную операцию «исключающее ИЛИ» между операндом и считанными данными
FMNOT	FB_MOD_NOT	Выполнить поразрядную операцию «НЕ» над считанными данными
FMRSH	FB_MOD_RSHIFT	Выполнить поразрядный сдвиг считанных данных вправо столько раз, сколько указано в операнде
FMLSH	FB_MOD_LSHIFT	Выполнить поразрядный сдвиг считанных данных влево столько раз, сколько указано в операнде

7.4 Рутины объединения данных

Эти рутины используют для чтения блоков данных последовательно из нескольких различных Фастбас-устройств. Каждая Рутина содержит отличающуюся процедуру выбора Фастбас-адресов, с которых будет начинаться чтение блока.

7.4.1 Считать данные из последовательных адресов Устройств

Имя :

FRDSEQ или FB_READ_DAT_SEQUENTIAL	Читать в область адресов данных последовательно
FRCSEQ или FB_READ_CSR_SEQUENTIAL	Читать в область адресов CSR последовательно

Категория : E, Рутина действия

Параметры :

return_code		FB_error_code	(OUT)	
control_id		FB_control_id	(IN)	
start_primad		FB_word	(IN)	
last_primad		FB_word	(IN)	
primad_step		FB_word	(IN)	
secondary_address		FB_word	(IN)	
buffer_mode		FB_buffer_mode	(IN)	
buffer_specifier		array_of_FB_word	(OUT)	(FBVAR)
	или	FB_seq_buffer_id	(IN)	(FBSEQ)
buffer_size		FB_integer	(IN)	
max_bytes		FB_integer	(IN)	

Действие :

Должна быть выполнена передача блока размером до max_bytes по адресу, специфицированному параметрами start_primad и secondary_address. Данные должны быть переданы из специфицированного Буфера данных или переданы в него. Порт освобождает присоединенного Слугу без прекращения Владения магистралью и вычисляет новый первичный адрес путем прибавления значения primad_step к текущему значению параметра первичного адреса. Если новый первичный адрес не превосходит значения параметра last_primad, должен быть передан еще один блок данных размером до величины max_bytes, данные присоединяются к данным предварительно переданного блока. Этот процесс должен повторяться до тех пор, пока не будет превышено значение last_primad, после чего магистраль должна быть освобождена.

7.4.2 Читать данные из Устройств, откликающихся на сканирование Т-штырьков

Имя:

FRDTPS или FB_READ_DAT_TP_SCAN	Сканировать, читать в область адресов данных
FRCTPS или FB_READ_CSR_TP_SCAN	Сканировать, читать в область адресов CSR

Категория : E, Рутина действия

Параметры :

return_code		FB_error_code	(OUT)	
environment_id		FB_environment_id	(IN)	
primary_address		FB_word	(IN)	
secondary_address		FB_word	(IN)	
buffer_mode		FB_word	(IN)	
buffer_specifier		FB_word	(OUT)	(FBVAR)
	или	FB_seq_buffer_id	(IN)	(FBSEQ)
buffer_size		FB_integer	(IN)	
max_bytes		FB_integer	(IN)	

Действие :

После Цикла широкой первичной адресации с использованием параметра primary_address, при значении MS=2 (FRDTPS) или MS=3 (FRCTPS), должен следовать Цикл чтения одиночного слова данных при значении MS=0. Параметр primary_address должен иметь сброшенным Глобальный бит (бит 1) и должен быть специфицирован в соответствии с пп. 3,3 а, 4, 5 или 6 Табл. III Стандарта Фастбас.

Для каждого набора бит в считанном слове создается Географический адрес, состоящий из поля GP (адрес группы), специфицированного параметром primary_address, и декодированного значения позиции установленного бита в слове данных. Для каждого такого адреса, за Вторичным адресным циклом, использующим значение параметра secondary_address, должна следовать передача блока в виде Операции чтения данных, выполняемая для числа байт, не превосходящего значения параметра max_bytes. Данные из успешно переданного блока должны быть каскадированы в специфицированном Буфере данных.

7.4.3 Присоединить Фастбас-устройство

Имя: FDVAL или FB_DEVICE_ALLOCATE

Категория : E, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
primary_address	FB_word	(IN)

Действие :

За Первичным адресным циклом в Области адресов регистров CSR должен следовать Вторичный адресный цикл, определяющий адрес 4. Цикл записи данных в виде одиночного слова выполняется затем с адресом Фастбас-порта как с данными. Соображения, изложенные в разделе 6.1, должны быть применены за исключением того, что значение Вторичного адреса всегда равно 4.

7.4.4 Отсоединить Фастбас-устройство

Имя: FDVDE или FB_DEVICE_DEALLOCATE

Категория : E, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
primary_address	FB_word	(IN)

Действие :

После Первичного адресного цикла в Области адресов CSR должен следовать Вторичный адресный цикл, определяющий адрес, равный 0. Затем должен быть выполнен Цикл записи данных одиночного слова со значением 80000 (шестнадцатиричное). Соображения, изложенные в разд. 6.1, должны быть применены за исключением того, что значение Вторичного адреса будет всегда равно 0.

7.5 Послать сообщение о Фастбас-прерывании

Имя: FBFIM или FB_SEND_FIM

Категория : E, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)	
environment_id	FB_environment_id	(IN)	
primary_address	FB_word	(IN)	
receiver_block_num	FB_word	(IN)	
buffer_mode	FB_buffer_mode	(IN)	
buffer_specifier	array_of_FB_word	(IN)	(FBVAR)
или	FB_seq_buffer_id	(IN)	(FBSEQ)
message_len_bytes	FB_integer	(IN)	

Действие :

Эта рутина должна выполнять Передачу блока данных в точном соответствии с п. 6.2.2 при длине блока, равной одной четверти значения message_len_bytes (в 32-разрядных словах).

Первичный адрес должен быть определен в Области адресов CSR и может быть либо Географическим, либо Логическим, либо Широким. Вторичный адрес должен был выработан Рутиной по значению, специфицируемому номером блока адресов в Устройстве обслуживания прерываний (см. Стандарт Фастбас, разд. 8.2).

Сообщение должно быть скопировано Воплощением во Внутренний буфер, а младшие четыре бита первого слова данных должны быть заменены на значение, соответствующее числу передаваемых слов данных и вычисляемое по формуле (message_len_bytes/4-1).

Воплощению следует заменить второе слово сообщения на адрес Мастера, посылающего сообщение.

Глава 8. РУТИНЫ ПРИМИТИВНЫХ ФАСТБАС-ДЕЙСТВИЙ

8.1 Введение

Рутины примитивных Фастбас-действий разделены на две группы. Рутины для одиночных Фастбас-циклов изложены в разделе 8.3, а Рутины для работы с отдельными линиями магистрали - в разд. 8.4.

При своем завершении Рутина примитивных действий может оставить Фастбас в состоянии, в котором магистраль не может быть использована другими пользователями (см. п. 2.2.5)

8.2 Параметры

В Рутинах, представленных в этой главе, не применяют следующие ОП (описанные в гл. 4) :

FB_PAR_NO_WAIT
FB_PAR_NO_ARBITRATION
FB_PAR_NO_PRIM_ADDR
FB_PAR_NO_SEC_ADDR
FB_PAR_NO_SEC_ADDR_CSR
FB_PAR_NO_DATA_CYCLE
FB_PAR_HOLD_BUS
FB_PAR_HOLD_BUS_NO_AR
FB_PAR_HOLD_AS.

Как Владение магистралью, так и любое адресное соединение не может быть отменено при завершении Рутины примитивных действий, если только отмена не вызвана Операцией специально.

8.3 Рутины одиночных циклов

Эти Рутины формируют полный набор, с помощью которого могут быть выработаны Рутины передач.

Если Рутина одиночного цикла вызывает фронт импульса на AS-или DS-линиях, следует дожидаться фронта подтверждения AK или DK и проверить соответствующий SS-код.

8.3.1 Арбитраж

Имя: FCARB или FB_CYCLE_ARBITRATE

Категория : C, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действия :

Эта Рутина должна принудить Мастера провести арбитраж на Владение магистралью (см. разд. 6.1).

8.3.2 Отказ от Владения магистралью

Имя: FCREL или FB_CYCLE_RELEASE_BUS

Категория : C, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие :

Эта Рутина должна вызывать сброс линии GK Мастером.

8.3.3 Первичный адресный цикл

Имя:

FCPD или FB_CYCLE_PA_DAT	Первичный адрес в Области адресов данных
FCPC или FB_CYCLE_PA_CSR	Первичный адрес в Области адресов CSR
FCPDM или FB_CYCLE_PA_DAT_MULT	Первичный адрес в Области адресов данных, широкий адрес
FCPCM или FB_CYCLE_PA_CSR_MULT	Первичный адрес в Области адресов CSR, широкий адрес

Категория : C, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)	
environment_id	FB_environment_id	(IN)	
buffer_mode	FB_buffer_mode	(IN)	
buffer_specifier	FB_word	(IN)	(FBVAR,FBVAL)
или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие :

Первичный адресный цикл должен быть выполнен так, как предписано п. 6.1.2, с использованием данных из специфицированного Буфера данных в качестве Первичного адреса и при значении кода MS , соответствующего вызванной Рутине :

Рутина	MS
FB_CYCLE_PA_DAT	MS=0
FB_CYCLE_PA_CSR	MS=1
FB_CYCLE_PA_DAT_MULT	MS=2
FB_CYCLE_PA_CSR_MULT	MS=3

8.3.4 Разъединение

Имя: FCDISC или FB_CYCLE_DISCONNECT

Категория : C, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие :

Все сигналы, за исключением GK, Мастер должен снять с магистрали.

8.3.5 Передачи одиночного слова

Имя:

FCRW или FB_CYCLE_READ_WORD	Чтение одиночного слова
FCRSA или FB_CYCLE_READ_SA	Чтение Вторичного адреса
FCWW или FB_CYCLE_WRITE_WORD	Запись одиночного слова
FCWSA или FB_CYCLE_WRITE_SA	Запись Вторичного адреса

Категория : C, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)		
environment_id	FB_environment_id	(IN)		
buffer_mode	FB_buffer_mode	(IN)		
buffer_specifier	FB_word	(IN)	FCWxx	(FBVAR,FBVAL)
или	FB_word	(OUT)	FCRxx	(FBVAR)
или	FB_seq_buffer_id	(IN)		(FBSEQ)

Действие :

Цикл данных или Вторичный адресный цикл должны быть выполнены и данные должны быть переданы в Буфер данных или из него так, как предписано п. 6.2.1. В зависимости от вызываемой Рутины коды RD и MS должны иметь значения :

Рутина	RD	MS
FB_CYCLE_READ_WORD	RD=1	MS=0
FB_CYCLE_READ_SA	RD=1	MS=2
FB_CYCLE_WRITE_WORD	RD=0	MS=0
FB_CYCLE_WRITE_SA	RD=0	MS=2

8.3.6 Передачи блока

Имя:

FCRB или FB_CYCLE_READ_BLOCK
FCWB или FB_CYCLE_WRITE_BLOCK

Категория : C, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)		
environment_id	FB_environment_id	(IN)		
buffer_mode	FB_buffer_mode	(IN)		
buffer_specifier	array_of_FB_word	(IN)	FCWB	(FBVAR)
или	array_of_FB_word	(OUT)	FCRB	(FBVAL)
или	FB_seq_buffer_id	(IN)		(FBSEQ)
max_bytes	FB_integer	(IN)		

Действие

Передача блока должна быть выполнена так, как предписано п. 6.2.2. Направление

передачи данных должно зависеть от вызываемых Рутин следующим образом :

Рутина	RD
FB_CYCLE_READ_BLOCK	RD = 0
FB_CYCLE_WRITE_BLOCK	RD = 1

Используемое значение кода MS должно зависеть от значения параметра FB_PAR_PIPE-LINE следующим образом :

FB_PAR_PIPELINE =FB_FALSE специфицирует MS=1
FB_PAR_PIPELINE =FB_TRUE специфицирует MS=3

Длина передачи данных может быть определена вызовом Рутины FB_READ_LEGTH, описанной в п. 6.2.4.

8.4 Доступ к линиям магистрали Фастбас

Эти Рутины составляют законченный набор, который дает возможность выполнить любую Фастбас-операцию, включая функции диагностики и тестирования.

Когда Рутины этого раздела вызывают фронты на линиях строба AS или DS, то не предусматривается никакой проверки соответствующего подтверждения или проверки состояния SS-линий.

В Воплощении, поддерживающем Рутины FB_LINE_READ и FB_LINE_WRITE, должны быть определены имена для всех линий магистрали Фастбас, однако не требуется обеспечивать доступ к каждой линии и можно возвращать код ошибки для индикации того, что доступ к конкретной линии не поддерживается.

В Рутинах, о которых пойдет речь ниже, диапазон значений параметров типа FB_integer определяются значениями параметров типа FB_LINE_ в соответствии со следующей таблицей :

ТАБЛИЦА 1. Определение типа FB LINE и соответствующего диапазона

Имя линии		Диапазон значений
Краткое	Полное	
FBLAS	FB_LINE_AS	0,1
FBLAK	FB_LINE_AK	0,1
FBLEG	FB_LINE_EG	0,1
FBLMS	FB_LINE_MS	0...7
FBLAD	FB_LINE_AD	$0...2^{32}-1$
FBLSS	FB_LINE_SS	0...7
FBLDS	FB_LINE_DS	0,1
FBLDK	FB_LINE_DK	0,1
FBLRD	FB_LINE_RD	0,1
FBLPE	FB_LINE_PE	0,1
FBLPA	FB_LINE_PA	0,1
FBLWT	FB_LINE_WT	0,1
FBLAR	FB_LINE_AR	0,1
FBLAG	FB_LINE_AG	0,1
FBLAL	FB_LINE_AL	0...63
FBLGK	FB_LINE_GK	0,1

ТАБЛИЦА 1.*(продолжение)* Определение типа FB_LINE и соответствующего диапазона

Имя линии		Диапазон значений
Краткое	Полное	
FBLAI	FB_LINE_AI	0, 1
FBLSR	FB_LINE_SR	0, 1
FBLRB	FB_LINE_RB	0, 1
FBLBH	FB_LINE_BH	0, 1
FBLGA	FB_LINE_GA	0...31
FBLTP	FB_LINE_TP	0, 1

8.4.1 Читать отдельную (-ные) линию (-ии) магистрали

Имя: FLR или FB_LINE_READ

Категория : C, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)	
environment_id	FB_environment_id	(IN)	
line	FB_line	(IN)	
buffer_mode	FB_buffer_mode	(IN)	
buffer_specifier	FB_word	(OUT)	(FBVAR)
или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие :

Текущее состояние Фастбас-линий (линии), определяемое параметром line, должно быть возвращено в специфицированный Буфер данных.

8.4.2 Записать отдельную (-ные) линию (-ии) магистрали

Имя: FLW или FB_LINE_WRITE

Категория : C, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)	
environment_id	FB_environment_id	(IN)	
line	FB_line	(IN)	
buffer_mode	FB_buffer_mode	(IN)	
buffer_specifier	FB_word	(OUT)	(FBVAR,FBVAL)
или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие :

На линиях, определяемых параметром line, должны быть выставлены значения сигналов, взятые из специфицированного Буфера данных.

8.4.3 Читать внутреннюю (-ие) линию (-ии)

Имя: FLRI или FB_LINE_READ_INTERNAL

Категория : C, Рутина действия

Параметры :

return_code	FB_error_code	(OUT)	
environment_id	FB_environment_id	(IN)	
line	FB_line	(IN)	
buffer_mode	FB_buffer_mode	(IN)	
buffer_specifier	FB_word	(OUT)	(FBVAR)
или	FB_seq_buffer_id	(IN)	(FBSEQ)

Действие :

Текущее состояние сигнала (-ов), внутренних для Порта Фастбас и произведенных Портом, относящихся к линии (-ям), которые определены параметром line, должно быть возвращено в спецификированный Буфер данных.

Точное значение возвращаемой величины определяется Воплощением.

Глава 9. ФАСТБАС-РУТИНЫ ЗАПРОСОВ SR И СООБЩЕНИЙ О ПРЕРЫВАНИЯХ

9.1 Введение

Рутины, описываемые в настоящей главе, устанавливают связь между программным обеспечением Мастера и обнаружением портом Мастера выставления сигнала SR или же получением Фастбас-сообщения о прерывании.

Существует два стандартных протокола прерываний в FASTBUS:

1. Любой Модуль в любое время может выставить сигнал SR запроса на обслуживание, это может быть обнаружено Мастером.
2. Мастер может послать Фастбас-сообщение о прерывании (длиной до 16 слов) в Блок адресов CSR приема прерывания в принимающем Устройстве обслуживания прерываний (УОП).

(Подробные описания содержатся в Стандарте Фастбас, в гл. 9.)

9.2. Запрос на обслуживание SR

Сигнал SR может вызвать Рутину пользователя на исполнение в асинхронном режиме (через прерывание контроллера) или значение сигнала может быть планомерно опрошено .

Сигнал SR может быть опрошен Примитивной рутиной FB_LINE_READ (FLR). Пользователь может нацелить Рутину пользователя так, чтобы она асинхронно вызывалась при установке сигнала SR.

Одним из параметров, обеспечивающих такую связь, является параметр SR_Source_Address (адрес источника запроса на обслуживание). Им может быть Географический адрес модуля, установившего сигнал SR, по которому вызвана Рутина пользователя, или FB_SR_DEFAULT (см. ниже).

Процесс, с помощью которого устройства устанавливают сигнал SR, зависит от программного и аппаратного обеспечения. Настоящий стандарт не требует от Воплощения того, чтобы оно было в состоянии определять источник сигнала SR, а только рекомендует делать это, если возможно. Воплощение со знанием топологии системы Фастбас может производить сканирование штырьков TP. Кроме того, Воплощение может планомерно опрашивать адрес каждого из источников сигнала SR, для которых предусмотрена описываемая связь, чтобы определить, кем выставлен SR (возможно,что более чем одним устройством).

Для резервной рутины FB_SR_DEFAULT должен быть предусмотрен параметр SR_source_address, чтобы учесть случаи, когда Воплощение не может определить адрес источника сигнала SR и когда Пользоваатель желает иметь связь со всеми выставляемыми сигналами SR независимо от конкретного источника. Если Рутина связана с названным параметром SR_source_address, должна быть вызвана программа Пользователя, когда обнаружено,что ни одно из присоединенных устройств не выставляет SR или Воплощение не может определить источник выставления SR.

Предусмотрены Рутины, позволяющие привилегированному Пользователю устанавливать или сбрасывать связи с SR из конкретного Порта. Связи по SR устанавливаются по умолчанию, как только выполнена явная связь.

9.2.1 Очистка установленного запроса на обслуживание

Воплощению следует обеспечить отмену установленных сигналов SR в устройстве, выставившем SR, до вызова присоединенной Рутины (с использованием битов CSR#0, описанных в п. 8.3.2

стандарта Фастбас). К области действий присоединенной Рутины относятся сброс источника SR и повторное выставление сигналов SR в устройстве.

9.2.2 Присоединить Рутину к SR

Имя: FBSRC или FB_SR_CONNECT

Категория : I

Параметры :

return_code	FB_error_code	(OUT)
SR_source_adress	FB_word	(IN)
port_id	FB_port_id	(IN)
routine	FB_routine	(IN)

Действие :

Когда обнаружен сигнал SR, и устройство, специфицированное параметром SR_source_address, выставляет SR, должна быть вызвана специфицированная Рутина, если доступен Порт. Если SR_source_address определен как FB_SR_DEFAULT, все обнаруженные запросы SR от источников, которые не присоединены явно, будут переданы специфицированной Рутине.

Параметр, передаваемый присоединенной Рутине, должен быть следующим:

SR_source_address FB_word (IN)

ОП SR_sourse_adress должен быть первичным адресом Слуги, выставляющего SR, за исключением случая, когда FB_SR_DEFAULT был передан к FB_SR_CONNECT. В этом случае FB_SR_DEFAULT также должен быть передан Рутине Пользователя.

9.2.3 Отсоединить Рутину от SR

Имя: FBSRD или FB_SR_DISCONNECT

Категория : I

Параметры :

return_code	FB_error_code	(OUT)
SR_source_code	FB_word	(IN)
port_id	FB_port_id	(IN)

Действие :

После вызова этой Рутины должна быть разорвана ранее установленная связь (для специфицированного port_id) между выставлением сигнала SR устройством с Географическим адресом SR_source_address и Рутиной, специфицированной Пользователем. С этого момента выставление сигнала SR устройством не будет приводить к вызову Рутины, специфицированной Пользователем. Если специфицирован SR_source_address для FB_SR_SOURCE_ALL, все подобные связи, созданные для специфицированного Порта, должны быть разорваны.

9.2.4 Разрешить связи с SR

Имя: FBSREN или FB_SR_ENABLE

Категория : I

Параметры :

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)

Действие :

Специфицированный Порт должен получить возможность отвечать на сигнал SR путем вызова какой-либо присоедененной Рутины.

9.2.5 Отменить связи с SR

Имя: FBSRDS или FB_SR_DISABLE

Категория : I

Параметры :

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)

Действие :

После вызова этой Рутины Воплощение не должно вызывать никакую присоединенную Рутину, вследствие выставления SR, обнаруженного специфицированным port_id. Любой ожидающий обработки сигнал SR должен быть игнорирован.

9.3 Рутины реакции на Сообщение о Фастбас-прерывании

Приведенные ниже описания используют для объявления таких Рутин, которые служат для приема сообщения о Фастбас-прерывании Портом Фастбас. Предполагается, что Пользователь не имеет непосредственного доступа к приемнику Фастбас-прерываний [Fastbus Interrupt Receiver - FIR] *). Прием сообщений и приведение в действие FIR производится Воплощением, которое отфильтровывает и диспетчеризует приходящие к Пользователю сообщения.

Если Порт лишен возможности принимать Сообщения о прерывании, никакая из присоединенных Рутин не будет вызвана.Порт по умолчанию должен иметь возможность принимать сообщения и передавать их присоединенным Рутинам.

9.3.1 Связять Рутину с FIR

Имя: FBFIRC или FB_FIR_CONNECT

Категория : I

Параметры :

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)
receiver_block_num	FB_integer	(IN)
filter_mask	FB_word	(IN)
filter_value	FB_word	(IN)
filter_word_num	FB_integer	(IN)

---

* В разделе 9.1 основного стандарта МЭК 935 Система Фастбас FIR назван иначе: ISD - Interrupt Service Device, а в ГОСТ 34.340 соответственно УОП - Устройство обслуживания прерываний. Поскольку в названия Рутин входят FIR, далее использовано это обозначение. Примечание научного редактора.

routine	FB_routine	(IN)

Действие :

Эта Рутина должна создавать связь между специфицированной Рутиной и специфицированным Блоком приемника прерываний) идентифицированного Порта. Значение параметра receiver_block_num, равное 0, должно специфицировать Блок приемника прерываний в диапазоне адресов от CSR#100 до CSR#10F, а значение параметра receiver_block_num, равное 1, должно специфицировать блок в диапазоне от CSR#110 до CSR#11F, и т.д.*

Когда сообщение принято в специфицированный Блок приемника прерываний, содержимое слова filter_word_num сообщения о прерывании должно быть сложено со значением параметра filter_mask путем операции поразрядного И, а результат должен быть сопоставлен со значением параметра filter_value. В случае равенства Рутина должна быть вызвана, в противном случае никаких дальнейших действий не производится.

Все присоединенные Рутины, удовлетворяющие этому критерию, должны быть вызваны в порядке, зависящем от Воплощения.

Передаваемые присоединенной Рутине параметры должны быть такими :

receiver_block_no	FB_integer	(IN)
message_buffer	array_of_FB_word	(IN)
message_lenght_bytes	FB_integer	(IN)
port_id	FB_port_id	(IN)

9.3.2 Отсоединить Рутину от FIR

Имя: FBFIRD или FB_FIR_DISCONNECT

Категория : I

Параметры :

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)
receiver_block_num	FB_integer	(IN)
filter_mask	FB_word	(IN)
filter_value	FB_word	(IN)
filter_word_num	FB_integer	(IN)
routine	FB_routine	(IN)

Действие :

Эта Рутина должна побудить Воплощение отменить дальнейшее ассоциирование специфицированных Рутины и фильтра, с одной стороны, с приемом сообщения о прерывании в Блоке приемника прерывания специфицированного Порта, с другой стороны.

9.3.3 Разрешить связи с FIR

Имя: FBFIRE или FB_FIR_ENABLE

Категория : I

---

*) В ГОСТ 34.340- иное название: Блок адресов CSR приема прерываний в Устройстве обслуживания прерываний (УОП). Примечание научного редактора.

Параметры :

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)

Действие :

Воплощение должно получить возможность установить доступ к Сообщениям о Фастбас-прерываниях для Порта, специфицированного параметром port_id, а затем передать эти сообщения предназначенным присоединенным Рутинам.

9.3.4 Отменить связи с FIR

Имя: FBFIRS или FB_FIR_DISABLE

Категория : I

Параметры :

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)

Действие :

Воплощение делается неспособным к вызову любой присоединенной Рутины пользователя в результате приема сообщения специфицированным Портом.

Глава 10. РУТИНЫ СИНХРОНИЗАЦИИ,СИСТЕМНЫХ РЕСУРСОВ И ПОРТОВ

Рутины системных ресурсов и Портов обеспечивают доступ к общим средствам синхронизации, а в машинно-независимом режиме - к специфическим функциям Порта.

Необходимо отметить, что доступ к Рутинам системных ресурсов в общем случае будет воздействовать на других пользователей многопользовательской системы, в связи с чем может потребоваться некоторая система приоритетов. Если пользователю не предоставляется доступ к функции, должна быть возвращена ошибка FB_ERR_NO_PRIVILEGE.

10.1 Средства синхронизации

Нижеприведенные Рутины позволяют пользователю проверять состояние Рутины действия или вызова FB_LIST_EXECUTE.Вообще эти средства наиболее полезны в случае, когда операционный параметр FB_PAR_NO_WAIT установлен в FB_TRUE и когда возврат в вызывающую Рутину был произведен прежде завершения запрошенного действия. Пользователь может непосредственно объявить Рутину для вызова на выполнение и протестировать или ожидать завершения Рутины действия или выполнения FB_LIST_EXECUTE.

10.1.1 Ждать завершения операции

Имя: FCOMWT или FB_COMPLETION_WAIT

Категория: S

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие:

Рутина должна ожидать завершения всех операций, связанных со специфицированным environment_id.

10.1.2 Проверить состояние завершения

Имя: FCOMTS или FB_COMPLETION_TEST

Категория: S

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
boolean_value	FB_boolean	(IN)

Действие:

Рутина должна проверять, завершены ли все операции, специфицированные для среды. Если все операции завершены, должно быть возвращено булевское значение FB_TRUE, иначе FB_FALSE.

10.1.3 Присоединить Рутину завершения

Имя: FBCOMC или FB_COMPLETION_CONNECT

Категория: S

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
routine	FB_routine	(IN)

Действие:

Рутина должна устанавливать связь между специфицированной Рутиной и завершением Рутины действия либо Списка действий. В конкретный момент времени может быть присоединена только одна Рутина завершения. При попытке установить связь ещё с одной Рутиной должна быть возвращена ошибка.

10.1.4 Отсоединить Рутину завершения

Имя: FBCOMD или FB_COMPLETION_DISCONNECT

Категория: S

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие:

Эта Рутина должна побуждать Воплощение далее не соотносить присоединенную Рутину разъединения с завершением работы Рутины действия либо Списка действий.

10.2 Сброс Порта Фастбас

Имя: FBPRST или FB_PORT_RESET

Категория: A, Рутина действия

Параметры:

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)

Действие:

Эта Рутина должна сбрасывать Порт Фастбас. Подробно действие такого сброса должно быть определено в Воплощении.

Все сигналы, установленные Портом на магистрали, должны быть сброшены.

Сброс Порта может влиять на других Пользователей. В многопользовательской системе Пользователь обычно будет требовать привилегии для успешного выполнения этой Рутины.

10.3 Назначение Порта Фастбас

В многопользовательском Воплощении нижеприведенные Рутины могут применяться Пользователем в целях присвоения и отмены присвоения Порта для его исключительного использования.

10.3.1 Назначить Порт Фастбас

Имя: FNPALL или FB_PORT_ALLOCATE

Категория: S, I

Параметры:

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)

Действие:

Эта Рутина должна пытаться назначить право исключительного использования специфицированного Порта Пользователю, связанному с вызываемой Рутиной. В случае неуспеха должна быть возвращена ошибка. Назначение Порта будет воздействовать на других пользователей из-за предотвращения их доступа к этому Порту.

10.3.2 Отменить назначение Порта Фастбас

Имя: FNPDEA или FB_PORT_DEALLOCATE

Категория: S, I

Параметры:

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)

Действие:

Эта Рутина должна отменять назначение специфицированого Порта.

10.3.3 Назначить Порт Фастбас с таймаутом

Имя: FNPALT или FB_PORT_ALLOCATE_TMO

Категория: S, I

Параметры:

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)
timeout_parameter	лимит времени в миллисекундах	

Действие:

Эта Рутина должна пытаться назначить право исключительного использования специфицированного Порта пользователю, связанному с вызываемой Рутиной.В случае неуспеха в течение времени, определенного Рутиной, должна быть возвращена ошибка.

10.4 Выдать номера версий.

Имя: FBVERS или FB_GET_SOFTWARE_VERSION

Категория: A

Параметры:

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)

hw_type	FB-word	(OUT)
hw_version	FB_word	(OUT)
sw_type	FB_word	(OUT)
sw_version	FB_word	(OUT)

Действие:

Рутина должна возвращать тип и номер версии аппаратного (hw) и программного (sw) обеспечения, используемых в специфицированном Порте.

Параметр типа аппаратного обеспечения устанавливают равным значению идентификатора Модуля Фастбас (определен в регистре CSR#0-, примечание переводчика). Версия аппаратного обеспечения может быть определена тем, кто воплощает стандарт.

Тип программного обеспечения присваивают способом, аналогичным назначению идентификатора модуля, как описано в стандарте Фастбас, п. 8.3.1. Таким образом, значение старших битов 16-разрядного идентификатора программного обеспечения присваивается Президентом Комитета NIM по запросу организаций, желающих получить эти значения.

Версия программного обеспечения определяется тем, кто воплощает стандарт, и должна быть включена в документацию, сопровождающую Воплощение.

10.5 Собщить об ошибке Порта Фастбас

Имя: FBPRPT или FB_PORT_STATUS_REPORT

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)
error_code	FB_error_code	(IN)

Действие:

В соответствии с параметром error_code должно быть выработано сообщение об ошибке. Должно быть произведено некоторое действие, зависимое от Воплощения, чтобы привлечь внимание Пользователя.

10.6 Сформировать сообщение об ошибке Порта

Имя: FBPTRN или FB_PORT_STATUS_TRANSLATE

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
port_id	FB_port_id	(IN)
error_code	FB_error_code	(IN)
message	FB_string	(OUT)

Действие:

В соответствии с параметром error_code должно быть выработано сообщение об ошибке. Результат возвращается в параметре message.

Глава 11. ОБРАБОТКА СОСТОЯНИЙ И ОШИБОК.

11.1 Введение

В этой главе описаны Рутины для управления обработкой ошибок, обнаруженных при исполнении стандартных Рутин, и Рутины для получения подробной ирформации об условиях, приведших к возникновению этих ошибок.

11.2 Коды ошибок и возвратов

Вызов любой из Рутин настоящего стандарта должен сопровождаться выработкой кода ошибки, состоящего из Кода состояния и его Серьезности.

Код состояния определяет в основных терминах результат выполнения Рутины. Коды состояния отмечают, завершилась ли Рутина нормально или встретились идентифицированные затруднения. Кроме того, Коды состояния служат для получения подробной информации и конечных условий выполненного Действия.

«Серьезность» [Severity] Кода ошибки используют как переключатель в различных автоматических действиях сбора сообщений об ошибках. Почти каждая Рутина в настоящем стандарте вырабатывает «Код Возврата» [Return Code], который по умолчанию - есть Код Ошибки и, следовательно, содержит как статусную информацию , так и информацию о Серьезности (см. п. 4.4.6). Серьезность может быть изменена относительно значения, заданного по умолчанию, посредством вызова FB_STATUS_SET_SEVERITY или в результате отклика на появление ошибки (см.ниже). Вне зависимости от модификаций Серьезности, Код ошибки, выработанный для занесения в Суммарный статус, должен быть одним из тех, которые описаны в гл. 12.

11.3 Суммарный и Дополнительный Статусы

В дополнение к Коду Возврата, в Режиме немедленного исполнения Рутина действия вырабатывает статусную информацию, состоящую из Кода Ошибки, общего числа байтов, считанных Рутиной, и общего числа байтов, записанных Рутиной. Эта информация образует Суммарный статус (Одно или оба сосчитанных числа байтов могут и не относиться к какой-либо конкретной Рутине).

Суммарный Статус должен всегда быть доступен через вызов FB_STATUS_GET_SUMMARY.

Добавочная информация о состоянии для каждой Рутины действия может быть собрана и запомнена как Дополнительный статус [Supplementary Status], связанный с Суммарным статусом. Дополнительный Статус обычно зависит от Контроллера и Воплощения стандарта.

Каждый терм Дополнительного статуса должен содержать Код ошибки.

Необязательные термы, входящие в состав Дополнительной статусной информации, содержат как связанный числовой параметр, определяющий Код Ошибки, и номер индекса, который по способу, зависящему от Воплощения, отмечает часть действия Рутины, заносимой в доклад.

Серьезность Суммарного статуса должна быть равна или больше Серьезности любого из Дополнительных статусов, относящихся к этому Суммарному.

Такой набор Суммарного и Дополнительных статусов образует Историю статусов .

Автоматический докладчик ошибок (см. ниже) использует уровень Серьезности Суммарного статуса так же как любой Дополнительный статус для определения того, что надо и что не надо включать в доклад. Существуют Рутины, предназначенные для определения подробностей, предоставляемых Историей статусов.

11.4 Усеченная генерация статусной информации

ОП FB_PAR_NO_STATUS при значении FB_TRUE должен ограничивать накопление статусной информации до уровня Суммарного статуса для достижения ускоренного выполнения действий. Дополнительные статусы не сохраняются.

Некоторые Воплощения могут не иметь возможности для генерации Дополнительного статуса. Для таких Воплощений ОП FB_ PAR_NO_STATUS следует делать параметром, защищенным по записи, и со значением по умолчанию FB_TRUE.

11.5 Статус Режима задержанного исполнения

Когда любая из Рутин действия, кроме FB_LIST_EXECUTE, вызывается в Среде Режима задержанного исполнения, она вырабатывает только Код возврата. Вызов Рутины не вырабатывает никакой Истории статуса.

Выполнение Списка посредством вызова FB_LIST_EXECUTE должно приводить в результате к генерации Суммарного статуса, который должен содержать код первой обнаруженной ошибки с наивысшим из встреченных уровней Серьезности.

В дополнение к выработке этого Суммарного статуса, выполнение Списка может также, если ОП FB_PAR_NO_STATUS установлен в FB_FALSE, генерировать Дополнительный статус для каждого из действий Списка.

11.6 Серьезность

Каждый из Кодов ошибки содержит соответствующую Серьезность. Определяемые Серьезности в порядке их возрастания должны быть следующие :

Полное имя Серьезности	Краткое имя Серьезности
FB_SEV_SUCCESS	FSSUCC
FB_SEV_INFO	FSINFO
FB_SEV_WARNING	FSWARN
FB_SEV_ERROR	FSERR
FB_SEV_FATAL	FSFTL
FB_SEV_NEVER	FSNEV

11.6.1 FB_SEV_SUCCES (FSSUCC)

Функция, запрошенная Рутиной, выполнена успешно.

11.6.2 FB_SEV_INFO (FSINFO)

Функция, запрошенная Рутиной, выполнена, но с некоторыми замечаниями. Например, при завершении запрошенной Фастбас-активности было сохранено владение магистралью.

11.6.3 FB_SEV_WARNING (FSWARN)

Функция, запрошенная Рутиной, была завершена несмотря на то, что была обнаруженна некоторая ошибка (ошибки). Эта ошибка не сделала дальнейший процесс бессмысленным либо невозможным. Например, запрошенное изменение параметра тайм-аута не исполнено в связи с тем,

что такое изменение не поддерживается на аппаратном уровне.

11.6.4 FB_SEV_ERROR (FSERR)

Запрошенная функция не была выполнена успешно . Ошибки такого рода обычно делают дальнейшие вызовы или действия Фастбас невозможными или их результаты непредсказуемыми. Например, первичный адресный цикл принял отличный от нуля SS-ответ или переданный параметр был запрещенным.

11.6.5 FB_SEV_FATAL (FSFTL)

Запрошенная функция не была выполнена успешно. Такая неудача достаточно существенна для аварийного завершения программы.

11.6.6 FB_SEV_NEVER (FSNEV)

Одно из применений иерархии уровней Серьезности ошибок состоит в обеспечении метода применения устанавливаемого порога для инициации Определенных специфических действий.

Эта Серьезность, FB_SEV_NEVER, должна быть такой, чтобы пороговые условия никогда не могли встретиться.

11.7 Установка Серьезности кода ошибки

Следующая Рутина должна использоваться для изменения уровня Серьезности в зависимости от текущего состояния Кода ошибки или его значения по умолчанию.

В Режиме отложенного исполнения Серьезность, связанная с Кодом ошибки, в первую очередь должна быть текущей при вызове FB_LIST_EXECUTE, а не принимаемой при построении Списка.

Имя : FSSSEV или FB_STATUS_SET_SEVERITY

Категория : E

Параметры :

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
error_code	FB_error_code	(IN)
severity	FB_severity	(IN)

Действие :

Серьезность, связанная со специфицируемым Кодом ошибки, должна устанавливаться в данное значение. Это значение изменяет Серьезность, присвоенную по умолчанию. Серьезность по умолчанию дана в таблице в гл. 12 для каждого Кода ошибки.

Коды ошибок, чья Серьезность была модифицирована в результате отклика на ошибку, должны иметь наименьшую из двух Серьезностей: полученной из последнего отклика и установленной этой Рутиной.

Не все Коды Ошибок могут иметь свои Серьезности измененными. При попытке изменить Серьезность для таких Кодов ошибок будет возвращена ошибка.

11.8 Отклик на ошибки при исполнении

Во время исполнения каждого Фастбас-цикла Мастером может быть принят один из восьми возможных Кодов состояния Слуги, который, в зависимости от обстоятельств, может содержать либо не содержать ошибку. Также возможно, что цикл не сможет завершиться или даже начаться из-за неспособности Мастера принять возвратную часть подтверждения в течение лимита времени, либо какая-либо другая аппаратная или программная выдержка времени истечет до завершения Цикла или Операции. Код отклика используют для управления ответом на такие условия. Специфицированные ответы для каждого аппаратного состояния должны содержать следующие, определенные ниже, отклики .

Определяемыми откликами должны быть :

Код Отклика	
Полное имя	Краткое имя
FB_RESP_IGNORE	FBRIG
FB_RESP_ABORT	FBRAB
FB_RESP_RETRY_ABORT	FBRRAB
FB_RESP_RETRY_IGNORE	FBRRIG
FB_RESP_ABORT_ACTION	FBRABA
FB_RESP_RETRY_ABORT_ACTION	FBRRAA

В Режиме отложенного исполнения, когда может быть выполнен Список действий, выработка откликов FB_RESP_ABORT и FB_RESP_RETRY_ ABORT приводит к прекращению выполнения Списка действий. Активизация дополнительных Кодов отклика, FB_RESP_ABORT_ACTION и FB_RESP_RETRY_ABORT_ACTION, приводит к прекращению того действия,во время выполнения которого обнаружена ошибочная ситуация, но при этом исполнение Списка действий будет продолжено.

В Режиме немедленного исполнения FB_RESP_ABORT_ACTION действует как FB_RESP_ABORT, а FB_RESP_RETRY_ABORT_ACTION действует так же, как и FB_RESP_RETRY_ABORT. Однако, как рассматривается далее, Серьезности результирующего Кода ошибки будут различными.

11.8.1 FB_RESP_IGNORE

Если состояние аппаратных средств таково, что соответствует Коду отклика FB_RESP_IGNORE, Воплощение должно пытаться продолжить действия так, как если бы ошибка не имела места.

В общем случае, возникшая ошибка может предотвратить продолжение любого действия до того, как возникнет заметная опасность.

11.8.2 FB_RESP_ABORT

Эта Рутина должна быть откликом Воплощения по умолчанию на все аппаратные ошибочные состояния, за исключением SS=0 и, при передаче блока, SS=2. Если существует специфицированный Код отклика на обнаруженные состояния аппаратуры, исполнение Списка Действий должно быть прекращено.

11.8.3 FB_RESP_RETRY_ABORT

Если имеет место состояние аппаратуры,соответствующее специфицированному Коду отклика, Фастбас-действие , которое вызвало ошибку, будет повторяться. Если после числа попыток

повтора, задаваемого текущим значением ОП FB_PAR_NUM_RETRY, состояние ошибки остается, исполнение Списка действий должно быть прекращено.

11.8.4 FB_RESP_RETRY_IGNORE

Если имеет место состояние аппаратуры, соответствующее специфицированному Коду отклика, Фастбас-действие, которое вызвало ошибку, будет повторяться. Если после числа попыток повтора, задаваемого текущим значением ОП FB_PAR_NUM_RETRY, состояние ошибки будет сохраняться, ошибка должна быть проигнорирована и исполнение Списка действий будет продолжено как при FB_RESP_IGNORE.

11.8.5 FB_RESP_ABORT_ACTION

Если имеет место состояние аппаратуры, соответствующее специфицированному Коду отклика, действие должно быть отменено. Если Список действий исполняем, производится попытка исполнения Действия, следующего в списке.

11.8.6 FB_RESP_RETRY_ABORT_ACTION

Если имеет место состояние аппаратуры, соответствующее специфицированному Коду отклика, Фастбас-действие, которое вызвало ошибку, будет повторяться. Если после числа попыток повтора, задаваемого текущим значением ОП FB_PAR_NUM_RETRY, состояние ошибки будет сохраняться, действие должно быть прекращено.Если Список действий исполняем, производится попытка исполнения Действия, следующего в Списке.

11.8.7 Серьезность Кодов ошибок, выработанных для соответствующих откликов на ошибки.

Если результатом значения Кода отклика является продолжение Действия или набора Действий, а не прекращение всех Действий, должны быть выработаны дополнительные Коды ошибок по FB_ERR_IGNORE (если ошибка была проигнорирована) или по FB_ERR_RETRY (если были предприняты повторы), или по FB_ERR_ABORT_ACTION (если было прекращено отдельное действие из Списка действий).

Как Код ошибки для первичной ошибки, так и дополнительный Код ошибки имеют соответствующие Серьезности, но в этом случае оба Кода ошибки должны иметь меньшую Серьезность из двух. За счет того, что Серьезностью FB_ER_RETRY является FB_SEV_WARN, Серьезность Кода ошибки по умолчанию или установлению, равная FB_SEV_ERROR, понижается, когда попытки успешны.

Значения по умолчанию Серьезностей для Кодов ошибок, соответствующих Кодам Отклика, представлены в таблице .

Код Ошибки	Присвоенная Серьезность
FB_ERR_IGNORE	FB_SEV_SUCCES
FB_ERR_RETRY	FB_SEV_INFO
FB_ERR_ABORT_ACTION	FB_SEV_WARN

Кроме того, если это возможно, число попыток повторного исполнения должно быть занесено в Дополнительную статусную информацию.

11.8.8 Специфицировать и определить отклик на ошибку

Следующие Рутины должны использоваться для специфицирования и получения доступа к Кодам откликов. Первая Рутина позволяет специфицировать Код отклика для данного состояния, а вторая — определять текущий Код отклика для данного состояния.

Имя: FSSR или FB_STATUS_SET_RESPONSE
FSGR или FB_STATUS_GET_RESPONSE

Категория: E

Параметры:

return_code		FB_error_code	(OUT)	
environment_id		FB_environment_id	(IN)	
hardware_error		FB_error_code	(IN)	
error_response		FB_response_code	(IN)	(FSSR)
	или	FB_response_code	(OUT)	(FSGR)

Параметр hardware_error должен принимать одно из следующих значений (могут также использоваться соответствующие полные имена из гл. 12):

FEASS0,	FEASS1,	FEASS2,	FEASS3,	FEASS4,	FEASS5,	FEASS6,	FEASS7
FESSS0,	FESSS1,	FESSS2,	FESSS3,	FESSS4,	FESSS5,	FESSS6,	FESSS7
FEDSS0,	FEDSS1,	FEDSS2,	FEDSS3,	FEDSS4,	FEDSS5,	FEDSS6,	FEDSS7
FEWTTO,	FEBSS2,	FEDPE,	FEBOV,	FEAKTO,	FESATO,	FEDKTO,	

где FExSSy есть Код статуса (SS), полученный Мастером в цикле первичного адреса (x=A), вторичного адреса (x=S) или в цикле данных (x=D), принимающий значение y от 0 до 7 .

Действие:

Отклик на состояние, соответствующее специфицированному Коду ошибки, должен устанавливаться посредством Рутины FB_STATUS_SET_RESPONSE. Значение по умолчанию откликов на тайм-ауты и ненулевые SS-отклики, за исключением SS=2 для циклов передачи блоков, должно быть FB_RESP_ABORT. Значение по умолчанию отклика на все SS=0 должно быть FB_RESP_IGNORE. Значение по умолчанию отклика на SS=2 для циклов передачи блоков должно быть FB_RESP_ABORT_ACTION.

Последнее установленное значение или, если не было установления, значение по умолчанию отклика системы на состояние аппаратуры данного типа, должно быть возвращено Рутиной FB_STATUS_GET_RESPONSE.

11.9 Автоматический Докладчик ошибок

Если Воплощение обеспечивает «автоматичесое составление доклада об ошибках», оно должно работать следующим образом:

После выполнения Рутины, перед возвращением управления Пользователю, должен быть вызван Автоматический Докладчик ошибок (АДО). АДО должен сначала проверить Серьезность Кода ошибки в Суммарном статусе и сравнить ее со значением, специфицированным по FB_PAR_HANDLER_THRESHOLD. Если она равна или превосходит FB_PAR_HANDLER_THRESHOLD и Обработчик ошибок Пользователя был подключен, как описано ниже, должен быть вызван Обработчик ошибок Пользователя.

Далее Серьезность Кода ошибки Суммарного статуса сравнивается с серьезностью, специфицированной по FB_PAR_REPORT_THRESHOLD. Если она равна или превосходит это пороговое значение, АДО должен продолжать работу, в противном случае управление должно быть возвращено Пользователю.

Если Код ошибки в Суммарном статусе имеет Серьезность равную или превосходящую FB_PAR_MESSAGE_THRESHOLD, должен быть выработан доклад об ошибке. Если FB_PAR_REPORT_TERSE имеет значение FB_FALSE, должен быть выработан также доклад для любого из Кодов ошибок Дополнительных статусов, чья Серьезность также равна или превосходит FB_PAR_MESSAGE_THRESHOLD. Если FB_PAR_REPORT_TERSE имеет значение FB_TRUE, то автоматически доклад для Дополнительного статуса вырабатываться не должен. Следует отметить, что если ОП FB_PAR_NO_STATUS имеет значение FB_TRUE, будет выработан только Суммарный статус.

Среда связи, которую использует АДО для передачи и/или отображения сообщений, зависит от Воплощения.

Наконец, если Код ошибки в Суммарном статусе имеет Серьезность, которая равна или превосходит значение, специфицированное FB_PAR_EXCEPTION_THRESHOLD, то должно быть объявлено исключительное состояние или выполнено действие по прекращению выполнения вызывающей программы.

11.9.1 Присоединить Рутину обработки ошибок Пользователя

Имя: FSCON или FB_STATUS_CONNECT

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
User_routine	FB_routine	(IN)

Действие:

Рутина обработки ошибок Пользователя User_routine должна соответствовать специфицированной Среде. Любая Рутина, ранее связанная с тем же самым environment_id , автоматически становится отсоединенной.

11.9.2 Отсоединить Рутину обработки ошибок Пользователя

Имя: FSDISC или FB_STATUS_DISCONNECT

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)

Действие:

После исполнения этой Рутины никакая Рутина обработки ошибок Пользователя не должна быть ассоциирована с поименованной Средой.

11.9.3 Подробности Рутины обработки ошибок Пользователя

Имя: (Рутина обработки ошибок Пользователя)

Категория: E

Параметры:

return_code	FB_error_code	(OUT)

return_error	FB_error_code	(OUT)
input_error	FB_error_code	(IN)
environment_id	FB_environment_id	(IN)

Действие:

Рутина обработки ошибок Пользователя должна иметь вышеозначенные параметры. Возвращаемый Код ошибки должен быть использован АДО по месту значения в Суммарном статусе. Если Обработчик ошибок Пользователя возвращает return_code, обозначающий, что исполнение не прошло успешно, АДО должен игнорировать Код ошибки, возвращённый Рутиной.

Рутина Пользователя не должна вызывать никакую Рутину Действия.

11.10 Доступ к информации Суммарного статуса

Суммарный статус вырабатывается всегда при исполнении Рутины действия в Среде Режима непосредственного исполнения или при вызове Рутины FB_LIST_EXECUTE. Нижеприведенная Рутина декодирует информацию Суммарного статуса.

11.10.1 Декодировать Суммарный статус

Имя: FSGSUM или FB_STATUS_GET_SUMMARY

Категория: A

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
error_code	FB_error_code	(OUT)
bytes_read	FB_integer	(OUT)
bytes_written	FB_integer	(OUT)

Действие:

Эта Рутина должна возвращать следующие значения из специфицированной Среды:

error_code — *Код ошибки наивысшей Серьезности, записанный в Среде;*

bytes_read — *Общее число байтов, считанных в любой Буфер Данных в результате вызова Рутины действия либо вызова FB_LIST_EXECUTE;*

bytes_written — *Общее число байтов, записанных из любого из Буферов данных в результате вызова Рутины действия либо вызова FB_LIST_EXECUTE.*

11.11 Доступ к информации Дополнительного статуса

В дополнение к информации Суммарного статуса Воплощение может собирать дополнительную информацию о выполнении Действий - «Дополнительный статус». Величина и тип такого Дополнительного статуса будет зависеть от Воплощения, поэтому точное определение всех параметров не может быть произведено.

Механизм доступа к такой информации определяется так, что каждый вызов одной какой-либо Рутины, описываемой ниже, должен возвращать единичную запись в информацию Дополнительного статуса.

Каждая запись подразумевается состоящей из Кода ошибки, описывающего Статус, и его Серьезности, параметра «associated_parameter», если он применяется, и параметра «where_occured»

[где_обнаружен], который отмечает, какое действие, либо какая часть действия привели к появлению Кода ошибки. От Пользователя зависит осведомленность о том, какой из Кодов ошибки имеет ассоциированный параметр.

Например, информационный Код ошибки FB_ERR_LAST_PAD должен иметь в качестве ассоциированного параметра значение последнего использованного Первичного адреса.

Если Рутина, используемая для доступа к записям Дополнительного статуса, не может больше выдавать информацию, она должна возвращать Код ошибки FB_ERR_NO_MORE_STATUS.

Повторные вызовы Рутин доступа могут быть использованы для извлечения всей доступной статусной информации.

Дополнительный статус Списка исполнения:

Если доступна информация Дополнительного статуса из вызова FB_LIST_EXECUTE, параметры where_occured и list_pointer являются указателями на действия в пределах Списка, к которому относится статусная информация. В частности, параметр list_pointer (входной параметр Рутин декодирования статуса, см. ниже) может быть получен при построении Списка посредством использования Рутины FB_LIST_GET_POINTER.

11.11.1 Декодировать информацию Дополнительного статуса

Имя: FSGSUP или FB_STATUS_GET_SUPPLEMENTARY

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
error_code	FB_error_code	(OUT)
associated_parameter	FB_word	(OUT)
where_occurred	FB_integer	(OUT)

Действие:

Эта Рутина должна возвращать дополнительную статусную информацию о действии.

Первый вызов этой Рутины должен возвращать тот же Код ошибки, что и код, возвращаемый Рутиной FB_STATUS_GET_SUMMARY.

Информация о Действиях, исполненных в Режиме отложенного исполнения:

Первый вызов этой Рутины, которая вырабатывает информацию о конкретном действии из Списка , должен возвратить Код ошибки, эквивалентный тому коду, который был бы занесен в Суммарный статус, если бы действие выполнялось в Режиме непосредственного исполнения.

11.11.2 Декодировать Дополнительный статус при исполнении Списка

Имя: FSGLSU или FB_STATUS_LIST_SUPPLEMENTARY

Категория: E

Параметры:

return_code	FB_error_code	(OUT)

environment_id	FB_environment_id	(IN)
list_pointer	FB_list_pointer	(IN)
error_code	FB_error_code	(OUT)
associated_parameter	FB_word	(OUT)
where_occurred	FB_integer	(OUT)

Действие:

Эта Рутина должна возвращать подробную статусную информацию из Истории статуса для специфицированного вызова действия из Списка по способу, идентичному FB_STATUS_GET_SUPPLEMENTARY.

Параметр list_pointer должен быть возвращен посредством вызова FB_LIST_GET_POINTER, производимого непосредственно перед вызовом Рутины, для действия которой должен быть получен статус.

Первый вызов этой Рутины , результатом которого является информация о конкретном действии из Списка, должен возвращать Код ошибки, эквивалентный тому коду, который мог бы быть получен в Суммарном статусе, если бы это действие было выполнено в Режиме непосредственного исполнения.

11.11.3 Поиск информации Дополнительного статуса

Имя: FSFSUP или FB_FIND_SUPPLEMENTARY

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
error_code	FB_error_code	(IN)
associated_parameter	FB_word	(OUT)
where_occurred	FB_integer	(OUT)

Действие:

Эта Рутина действует идентично Рутине FB_STATUS_GET_SUPPLEMENTARY, за исключением того, что параметр error_code придается вызывающей стороной. Рутина позволяет пользователю извлекать специфицированную запись информации Дополнительного статуса.

Если специфицированный Код ошибки не представлен в Дополнительном статусе, должна быть возвращена ошибка FB_ERR_STATUS_NOT_FOUND.

11.11.4 Найти специфический Дополнительный статус в Списке

Имя: FSFLSU или FB_FIND_LIST_SUPPLEMENTARY

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
list_pointer	FB_list_pointer	(IN)
error_code	FB_error_code	(IN)
associated_parameter	FB_word	(OUT)
where_occurred	FB_integer	(OUT)

Действие:

Эта Рутина действует идентично FB_STATUS_LIST_SUPPLEMENTARY, за исключением того, что параметр error_code придается вызывающей стороной, чтобы определить, где существует специфический Дополнительный статус.

Рутина должна возвращать информацию только о том действии, для которого Код ошибки и указатель в Списке соответствуют величинам, специфицированным Пользователем.

Должна быть возвращена ошибка FB_ERR_STATUS_NOT_FOUND , если в Дополнительном статусе не обнаружено никакого дополнительного присутствия Кода ошибки.

11.12 Рутины обработки Кодов ошибки

Необходимо отметить, что FB_STATUS_SEVERITY, FB_STATUS_MATCH и FB_STATUS_THRESHOLD являются функциями, а не подпрограммами.

11.12.1 Проверить Серьезность Кода ошибки

Имя: FSEVER или FB_STATUS_SEVERITY

Категория: E

Параметры:

error_code	FB_error_code	(IN)

Действие:

FB_STATUS_SEVERITY является функцией,которая должна возвращать значение Серьезности, специфицированной параметром error_code.

11.12.2 Проверить Код статуса в Коде ошибки

Имя: FMATCH или FB_STATUS_MATCH

Категория: E

Параметры:

error_code_1	FB_error_code	(IN)
error_code_2	FB_error_code	(IN)

Действие:

Эта функция должна возвращать значения FB_TRUE, если два Кода ошибки идентичны (исключительно по Серьезности), и FB_FALSE в противном случае.

11.12.3 Проверить Серьезность в Коде ошибки

Имя: FSTHR или FB_STATUS_THRESHOLD

Категория: E

Параметры:

error_code_1	FB_error_code	(IN)

severity	FB_severity	(IN)

Действие:

Эта функция должна возвращать значения FB_TRUE, если Серьезность Кода ошибки равна или больше специфицированной Серьезности, и FB_FALSE в противном случае.

11.12.4 Возвратить Код ошибки специфицированной Серьезности

Имя: FSENC или FB_STATUS_ENCODE

Категория: E

Параметры:

new_error_code	FB_error_code	(OUT)
error_code	FB_error_code	(IN)
severity	FB_severity	(IN)

Действие:

Эта Рутина должна возвращать Код ошибки с той же самой статусной информацией, что и в параметре error_code, но со специфицированной Серьезностью.

11.13 Доложить Фастбас-ошибку

Имя: FSRPT или FB_STATUS_REPORT

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
error_code	FB_error_code	(IN)
associated_parameter	FB_word	(IN)
where_occurred	FB_integer	(IN)

Действие:

По значениям параметров error_code, associated_parameter и where_occurred должно быть выработано сообщение об ошибке. Некоторые Действия, зависящие от Воплощения, должны быть произведены для привлечения внимания Пользователя к этому сообщению.

11.14 Выработать Сообщение об ошибке

Имя: FSTRAN или FB_STATUS_TRANSLATE

Категория: E

Параметры:

return_code	FB_error_code	(OUT)
environment_id	FB_environment_id	(IN)
error_code	FB_error_code	(IN)
associated_parameter	FB_word	(IN)
where_occurred	FB_integer	(IN)
message	FB_string	(OUT)

Действие:

По значениям параметров error_code, associated_parameter и where_occurred должно быть выработано сообщение об ошибке. Результат возвращается в параметр message.

Глава 12. КОДЫ ОШИБОК

12.1 Стандартный набор кодов ошибок

Как уже обсуждалось в гл. 11, Код ошибки содержит статусную информацию - Код статуса (описание некоторых состояний, встречающихся при вызове Рутин этого стандарта) и Серьезность.

Описанные в настоящей главе Коды ошибок характеризуют все возможные статусные состояния, которые могут встретиться при выполнении Рутин этого стандарта. Каждый описываемый Код ошибки имеет Серьезность, соответствующую по умолчанию данному статусному состоянию.

Во всех Воплощениях должно использовать краткие имена Кодов ошибок, перечисленные в настоящей главе.

Если Воплощение не включает FB_STATUS_SET_SEVERITY и/или FB_STATUS_SET_RESPONSE (см. гл. 11), то Серьезность каждого Кода ошибки должна быть такой, как описано ниже.

Рутины FB_STATUS_SET_RESPONSE и FB_STATUS_SET_SEVERITY могут изменять ассоциированную с Кодом статуса Серьезность. Если используются эти Рутины, то Серьезность Кода ошибки изменяется.

FB_ERR_ACCESS_DATA_BUFFER Краткое имя: FEBUF, Серьезность: FB_SEV_ERROR

Во время передачи данных в Буфер данных или из него была обнаружена ошибка доступа. Такое могло бы произойти при обращении к ОЗУ, внутреннему или внешнему последовательному буферу.

FB_ERR_ABORT_ACTION Краткое имя: FEABA, Серьезность: FB_SEV_WARNING

Действие, содержащееся в Списке действий и исполняемое посредством вызова FB_LIST_EXECUTE, было преждевременно прекращено из-за наличия ошибки, однако прекращения Списка действий не последовало. Соответствующий возникшей ошибке Код ошибки доступен и связан с этой ошибкой.

Серьезность FB_ERR_ABORT_ACTION должно использовать для модификации Серьезности, соответствующей возникшей ошибке.

FB_ERR_ADDRESS_RELEASE_TIMEOUT Краткое имя: FEAKDO, Серьезность: FB_SEV_ERROR

В течение лимита времени после AS(d) не был обнаружен AK(d).

FB_ERR_AK_TIMEOUT Краткое имя: FEAKTO, Серьезность: FB_SEV_ERROR

Превышено время таймера AK.

FB_ERR_ALREADY_CONNECT Краткое имя: FEACON, Серьезность: FB_SEV_ERROR

Попытка присоединиться к Рутине окончилась безуспешно из-за того, что присоединение уже было выполнено.

FB_ERR_BAD_PARAMETER_VALUE Краткое имя: FEBPRV, Серьезность: FB_SEV_WARNING

При вызове FB_PAR_SET значение данного параметра не являлось правильным, однако значение его (возможно преобразованное) было принято каким-либо образом и произошло изменение Среды.

FB_ERR_BLOCK_DATA_SS2 Краткое имя: FEBSS2, Серьезность: FB_SEV_INFO

Для Цикла данных во время передачи блока SS=2 (см. FB_ERR_DATA_SS2).

FB_ERR_BUFFER_OVERFLOW Краткое имя: FEBOV, Серьезность: FB_SEV_ERROR

Запрошенное действие переполнит или переполнило Буфер Данных.

FB_ERR_CLOSED Краткое имя: FECLSD, Серьезность: FB_SEV_ERROR

Сессия Фастбас была завершена. В этом состоянии никакая другая Рутина, кроме FB_OPEN, не может быть вызвана (то же значение ошибки будет возвращено при FB_CLOSE если сессия не была открыта).

FB_ERR_CONNECT Краткое имя: FECON, Серьезность: FB_SEV_ERROR

Попытка присоединения Рутины закончилась безуспешно и при этом предварительного присоединения не было.

FB_ERR_DATA_PARITY_ERROR Краткое имя: FEDPE, Серьезность: FB_SEV_ERROR

При выполнении Цикла чтения данных возникла ошибка, связанная с контролем по четности (значение SS могло быть любым, включая 0).

FB_ERR_DATA_SS0 Краткое имя: FEDSS0, Серьезность: FB_SEV_SUCCESS

В Цикле данных Фастбас было возвращено значение кода SS = 0.

FB_ERR_DATA_SS1 Краткое имя: FEDSS1, Серьезность: FB_SEV_ERROR

В Цикле данных было возвращено значение кода SS = 1.

FB_ERR_DATA_SS2 Краткое имя: FEDSS2, Серьезность: FB_SEV_ERROR

В Цикле данных было возвращено значение кода SS = 2. Необходимо отметить, что во время передачи блока при значении кода SS=2, по умолчанию следует вырабатывать ошибку FB_ERR_BLOCK_DATA_SS2, которой соответствует серьезность FB_SEV_INFO.

FB_ERR_DATA_SS3 Краткое имя: FEDSS3, Серьезность: FB_SEV_ERROR

В Цикле данных было возвращено значение кода SS = 3.

FB_ERR_DATA_SS4 Краткое имя: FEDSS4, Серьезность: FB_SEV_ERROR

В Цикле данных было возвращено значение кода SS = 4.

FB_ERR_DATA_SS5 Краткое имя: FEDSS5, Серьезность: FB_SEV_ERROR

В Цикле данных было возвращено значение кода SS = 5.

FB_ERR_DATA_SS6 Краткое имя: FEDSS6, Серьезность: FB_SEV_ERROR

В Цикле данных было возвращено значение кода SS = 6.

FB_ERR_DATA_SS7 Краткое имя: FEDSS7, Серьезность: FB_SEV_ERROR

В Цикле данных было возвращено значение кода SS = 7.

FB_ERR_DISCONNECT Краткое имя: FEDCON, Серьезность: FB_SEV_ERROR

При попытке произвести разъединение выполнение Рутины было прекращено. Никакое текущее соединение не может существовать.

FB_ERR_DK_RELEASE_TIMEOUT Краткое имя: FEDKDO, Серьезность: FB_SEV_ERROR

В течение лимита времени после DS(d) не обнаружено DK(d).

FB_ERR_DK_TIMEOUT Краткое имя: FEDKTO, Серьезность: FB_SEV_ERROR

В течение лимита времени после фронта DS не обнаружен фронт DK.

FB_ERR_ENV_ACTIVE Краткое имя: FEENAC, Серьезность: FB_SEV_ERROR

Специфицированная среда является используемой, но может быть недоступной. Ошибка такого рода может иметь место, если предпринята попытка обращения к Среде, когда действие или множество действий выполняются при значении ОП FB_PAR_NO_WAIT, равном FB_TRUE.

FB_ERR_ENV_ID_OVERFLOW Краткое имя: FEEIOV, Серьезность: FB_SEV_ERROR

Невозможность создания Среды из-за того, что максимально возможное число Сред уже было создано.

FB_ERR_ENV_OVERFLOW Краткое имя: FEENOV, Серьезность: FB_SEV_ERROR

Не существует достаточно пространства для создания Среды.

FB_ERR_ENV_RELEASE Краткое имя: FEEREL, Серьезность: FB_SEV_ERROR

Попытка освобождения Среды окончилась безуспешно. Однако параметр environment_id был действительным.

FB_ERR_ERROR Краткое имя: FEERR, Серьезность: FB_SEV_ERROR

Обнаружена ошибка с Серьезностью FB_SEV_ERROR. Дополнительная информация может быть доступна через Статусную Историю, если она существует.

FB_ERR_FATAL Краткое имя: FEFTL, Серьезность: FB_SEV_FATAL

Обнаружена ошибка с Серьезностью FB_SEV_FATAL. Дополнительная информация может быть доступна через Статусную Историю, если она существует.

FB_ERR_HARDWARE_ERROR Краткое имя: FEHERR, Серьезность: FB_SEV_ERROR

Произошла аппаратная ошибка, зависящая от Воплощения. Любая доступная дополнительная информация об ошибке может быть получена из Статусной истории.

FB_ERR_HARDWARE_WARNING Краткое имя: FEHWRN, Серьезность: FB_SEV_WARNING

Получено предупреждение об аппаратной ошибке в Порте, зависящей от Воплощения. Любая доступная дополнительная информация может быть получена из Статусной истории.

FB_ERR_HOLDING_ADDRESS Краткое имя: FEHDAS, Серьезность: FB_SEV_INFO

Поддерживается адресное соединение (связка AS/AK) либо вследствие того, что один из ОП FB_PAR_HOLD_AS или FB_HOLD_BUS_ON_ERROR установлен в FB_TRUE, либо из-за того, что было выполнено Примитивное Фастбас-действие.

FB_ERR_HOLDING_BUS Краткое имя: FEHDFB, Серьезность: FB_SEV_INFO

Сохранено Владение магистралью Фастбас (GK было установлено Портом). Это произошло

либо из-за того, что один из ОП FB_PAR_HOLD_BUS, FB_PAR_HOLD_BUS_NO_AR или FB_HOLD_BUS_ON_ERROR установлен в FB_TRUE, либо из-за того, что было выполнено примитивное Фастбас-действие.

FB_ERR_IGNORE Краткое имя: FEIGNO, Серьезность: FB_SEV_SUCCESS

Обнаружена ошибка, которая была игнорирована. Может быть предпринята попытка повтора действия. Код ошибки, соответствующий первоначальной ошибке, должен быть доступен вместе с кодом этой ошибки.

Серьезность FB_ERR_IGNORE должна быть использована для модификации серьезности, доложенной с оригинальной ошибкой.

FB_ERR_ILL_PARAMETER_VALUE Краткое имя: FEIPRV, Серьезность: FB_SEV_ERROR

При вызове FB_SET_PAR значение указываемого ОП недопустимо. В Среде изменений не было произведено.

FB_ERR_ILL_SEQ_BUFFER_ID Краткое имя: FEILSI, Серьезность: FB_SEV_ERROR

Возвращено из FB_DECLARE_SEQ_BUFFER или FB_DECLARE_EXT_SEQ_BUFFER ; значение последовательного параметра bufer_id является неверным для Воплощения.

FB_ERR_IMPLEMENTATION Краткое имя: FEIMP, Серьезность: FB_SEV_ERROR

В конкретном Воплощении обнаружена неожидаемая ошибка. Никакая другая иформация не доступна.

FB_ERR_INFO Краткое имя: FEINFO, Серьезность: FB_SEV_INFO

Обнаружена ошибка с серьезностью FB_SEV_INFO. Дополнительная информация может быть получена из Статусной истории, если она существует.

FB_ERR_INVALID_ENV_ID Краткое имя: FEINEI, Серьезность: FB_SEV_ERROR

Не существует Среды, соответствующей данному значению параметра environmet_id.

FB_ERR_INVALID_LIST_POINTER Краткое имя: FEINLI, Серьезность: FB_SEV_ERROR

Неверное значение специфицирует Указатель списка.

FB_ERR_INVALID_PORT_ID Краткое имя: FEINPO, Серьезность: FB_SEV_ERROR

Не существует Порта, соответствующего данному значению параметра port_id.

FB_ERR_INVALID_SEQ_BUFFER_ID Краткое имя: FEINSI, Серьезность: FB_SEV_ERROR

Не существует последовательного буфера (внешнего или внутреннего), соответствующего данному значению параметра buffer_id.

FB_ERR_LAST_PRIMARY_ADDRESS Краткое имя: FELPAD, Серьезность: FB_SEV_INFO

Значение последнего первичного адресной связки в Действии или при вызове FB_LIST_EXECUTE.

FB_ERR_LAST_SECONDARY_ADDRESS
Краткое имя: FELSAD, Серьезность: FB_SEV_INFO

Значение последнего вторичного адреса в Действии или при вызове FB_LIST_EXECUTE.

FB_ERR_LAST_LIST_POINTER Краткое имя: FELPTR, Серьезность: FB_SEV_ERROR

Конечное значение указателя списка в вызове FB_LIST_EXECUTE.

FB_ERR_LIST_OVERFLOW Краткое имя: FELOV, Серьезность: FB_SEV_ERROR

При составлении Списка превышено пространство, допустимое для размещения Списка в Среде.

FB_ERR_LONG_TIMEOUT Краткое имя: FELOTO, Серьезность: FB_SEV_ERROR

Превышено значение времени Долгого таймера.

FB_ERR_LONG_TIMER_ON Краткое имя: FELOON, Серьезность: FB_SEV_INFO

Продолжается работа Долгого таймера (т.е. продолжается отсчет времени).

FB_ERR_NO_ADDRESS_CONNECTION Краткое имя: FENOAS, Серьезность: FB_SEV_ERROR

В отсутствие адресного соединения (связки AS/AK) была предпринята попытка выполнить Цикл данных или Вторичный адресный цикл.

FB_ERR_NO_MORE_STATUS Краткое имя: FENMST, Серьезность: FB_SEV_ERROR

Не существует никакого Дополнительного статуса для конкретного действия , Списка или специфицированного Указателя списка.

FB_ERR_NO_PRIVILEGE Краткое имя: FENPRV, Серьезность: FB_SEV_ERROR

Пользователь,вызвавший Рутину, не имел достаточных привилегий. Этот Код ошибки может быть возвращен Воплощением, поддерживающим многопользовательский режим, когда вызванная Рутина может приводить к конфликтам-столкновениям с другими Пользователями системы.

FB_ERR_NO_SUPPLEMENTARY_STATUS Краткое имя: FENSST, Серьезность: FB_SEV_INFO

Нет Дополнительного статуса. Либо еще не была вызвана ни одна Рутина действия, использующая данную Среду, либо ОП FB_PAR_NO_STATUS был установлен при вызове последней Рутины действия. Этот Код ошибки должен быть возвращен посредством вызова FB_STATUS_FIND_SUPPLEMENTARY или FB_STATUS_FIND_LIST_SUPPLEMENTARY, когда не был найден запрошенный Код ошибки Дополнительного статуса. Подобный Код ошибки также должен быть возвращен посредством вызова FB_STATUS_GET_LIST_SUPPLEMENTARY либо FB_STATUS_FIND_LIST_SUPPLEMENTARY, когда нет Дополнительного статуса, соответствующего специфицированному Указателю списка.

FB_ERR_NORMAL Краткое имя: FENORM, Серьезность: FB_SEV_SUCCESS

Нормальное завершение. Никаких неожидаемых состояний не обнаружено. Дополнительная информация может быть получена из Статусной истории.

FB_ERR_NOT_CONNECTED Краткое имя: FENCON, Серьезность: FB_SEV_WARNING

Попытка отсоединения Рутины завершилась безуспешно из-за того, что не существовало соединения.

FB_ERR_NOT_HOLDING_MASTERSHIP Краткое имя: FENHMA, Серьезность: FB_SEV_SUCCESS

Владения магистралью Фастбас не было.

FB_ERR_NOT_MASTER Краткое имя: FENOMA, Серьезность: FB_SEV_ERROR

Предпринята попытка любого Цикла Фастбас, кроме Арбитража, и при этом Порт не владел магистралью. Это может произойти тогда, когда в требуемом порядке не вызваны Рутины примитивных действий, либо ОП FB_PAR_NO_ARBITRATION установлен в FB_TRUE, но Порт не поддерживал Владение магистралью.

FB_ERR_OPEN Краткое имя: FEOPEN, Серьезность: FB_SEV_WARNING

Во время уже открытой Фастбас-сессии был произведен вызов FB_OPEN. Сессия была закрыта и затем повторно открыта. Установленные в предыдущей сессии параметры environment_id отменяются.

FB_ERR_OPERATION_PENDING Краткое имя: FEPEND, Серьезность: FB_SEV_WARNING

Запрошенное действие или множество действий не было еще завершено. Такой Код ошибки может быть возвращен , если в Режиме немедленного исполнения вызвана Рутина действия или же была вызвана FB_LIST_EXECUTE с ОП FB_PAR_NO_WAIT, установленным в FB_TRUE. Такой Код ошибки отмечает то, что запрошенное действие (действия) было начато без фатальной ошибки.

FB_ERR_PIPELINE_TRANSFER Краткое имя: FEPPL, Серьезность: FB_SEV_ERROR

Этот Код ошибки возвращается только при режиме Принудительной передачи. Код индицирует наличие таких проблем, возникающих при выполнении передачи, которые не могут быть доложены посредством других более подробных Кодов ошибок. Например, число полученных фронтов DK не соответствует числу фронтов DS.

FB_ERR_POINTER_OVERFLOW Краткое имя: FEPOV, Серьезность: FB_SEV_ERROR

При попытке переместить указатель Последовательного буфера превышена граница буфера.

FB_ERR_PORT_ACTIVE Краткое имя: FEPAC, Серьезность: FB_SEV_ERROR

Попытка выполнения Действия была невозможной из-за того, что Порт был активен.Это может произойти в многопользовательской системе, если Порт выполняет действие, запрошенное другим Пользователем,и не используется очередь запросов.

FB_ERR_PORT_ALREADY_ALLOCATED Краткое имя: FEPAL, Серьезность: FB_SEV_ERROR

Попытка захватить Порт не была успешной из-за того, что Порт уже был захвачен другим Пользователем.

FB_ERR_PORT_NOT_ALLOCATED Краткое имя: FEPNAL, Серьезность: FB_SEV_ERROR

Попытка выполнения Действия, требовавшего захваченности Порта, была прекращена из-за того, что Порт не был захвачен, либо была попытка освободить Порт, который не был захвачен.

FB_ERR_PORT_NOT_AVAILABLE Краткое имя: FEPNOP, Серьезность: FB_SEV_ERROR

Порт не доступен для использования потому, что он не находится в оперативном состоянии.

FB_ERR_PORT_OWNED Краткое имя: FEPOWN, Серьезность: FB_SEV_INFO

Запрос на захват Порта был произведен по отношению к Порту, уже захваченному предыдущим Пользователем.

FB_ERR_PRIMARY_ADDRESS_SS0 Краткое имя: FEASS0, Серьезность: FB_SEV_SUCCESS

В Первичном адресном Цикле обнаружено значение SS=0.

FB_ERR_PRIMARY_ADDRESS_SS1 Краткое имя: FEASS1, Серьезность: FB_SEV_ERROR

В Первичном адресном Цикле обнаружено значение SS=1.

FB_ERR_PRIMARY_ADDRESS_SS2 Краткое имя: FEASS2, Серьезность: FB_SEV_ERROR

В Первичном адресном Цикле обнаружено значение SS=2.

FB_ERR_PRIMARY_ADDRESS_SS3 Краткое имя: FEASS3, Серьезность: FB_SEV_ERROR

В Первичном адресном Цикле обнаружено значение SS=3.

FB_ERR_PRIMARY_ADDRESS_SS4 Краткое имя: FEASS4, Серьезность: FB_SEV_ERROR

В Первичном адресном Цикле обнаружено значение SS=4.

FB_ERR_PRIMARY_ADDRESS_SS5 Краткое имя: FEASS5, Серьезность: FB_SEV_ERROR

В Первичном адресном Цикле обнаружено значение SS=5.

FB_ERR_PRIMARY_ADDRESS_SS6 Краткое имя: FEASS6, Серьезность: FB_SEV_ERROR

В Первичном адресном Цикле обнаружено значение SS=6.

FB_ERR_PRIMARY_ADDRESS_SS7 Краткое имя: FEASS7, Серьезность: FB_SEV_ERROR

В Первичном адресном Цикле обнаружено значение SS=7.

FB_ERR_READ_ONLY_PARAMETER Краткое имя: FEROPA, Серьезность: FB_SEV_ERROR

Была предпринята попытка установить для ОП режим защиты по записи.

FB_ERR_RETRY Краткое имя: FERTY, Серьезность: FB_SEV_INFO

По крайней мере один повтор запрошенного действия был предпринят в связи с обнаружением ошибки. Действие в конце концов выполнено успешно. Код ошибки, соответствующий первоначальной ошибке, следует делать вместе с этим Кодом ошибки.

Серьезность кода FB_ERR_RETRY должна быть использована для модификации Серьезности, доложенной с первоначальной ошибкой.

FB_ERR_SAD_TIMEOUT Краткое имя: FESATO, Серьезность: FB_SEV_ERROR

В течение Вторичного адресного цикла обнаружено превышение лимита времени.

FB_ERR_SECONDARY_ADDRESS_SS0 Краткое имя: FESSS0, Серьезность: FB_SEV_SUCCESS

Во Вторичном адресном цикле обнаружено значение SS=0.

FB_ERR_SECONDARY_ADDRESS_SS1 Краткое имя: FESSS1, Серьезность: FB_SEV_ERROR

Во Вторичном адресном цикле обнаружено значение SS=1.

FB_ERR_SECONDARY_ADDRESS_SS2 Краткое имя: FESSS2, Серьезность: FB_SEV_ERROR

Во Вторичном адресном цикле обнаружено значение SS=2.

FB_ERR_SECONDARY_ADDRESS_SS3 Краткое имя: FESSS3, Серьезность: FB_SEV_ERROR

Во Вторичном адресном цикле обнаружено значение SS=3.

FB_ERR_SECONDARY_ADDRESS_SS4 Краткое имя: FESSS4, Серьезность: FB_SEV_ERROR

Во Вторичном адресном цикле обнаружено значение SS=4.

FB_ERR_SECONDARY_ADDRESS_SS5 Краткое имя: FESSS5, Серьезность: FB_SEV_ERROR

Во Вторичном адресном цикле обнаружено значение SS=5.

FB_ERR_SECONDARY_ADDRESS_SS6 Краткое имя: FESSS6, Серьезность: FB_SEV_ERROR

Во Вторичном адресном цикле обнаружено значение SS=6.

FB_ERR_SECONDARY_ADDRESS_SS7 Краткое имя: FESSS7, Серьезность: FB_SEV_ERROR

Во Вторичном адресном цикле обнаружено значение SS=7.

FB_ERR_SEQ_BUFFER_ACTIVE Краткое имя: FESBAC, Серьезность: FB_SEV_ERROR

Последовательный буфер, соответствующий данному значению параметра buffer_id, используется, но может не быть доступен для Рутин, изменяющих Указатель буфера.

FB_ERR_SEQ_BUFFER_EXISTS Краткое имя: FESBEX, Серьезность: FB_SEV_WARNING

Уже существовал последовательный буфер, соответствующий данному значению параметра buffer_id, при вызовах FB_DECLARE_SEQ_BUFFER или FB_DECLARE_EXT_SEQ_BUFFER. Последовательный буфер был переобъявлен, а указатели переинициализированы.

FB_ERR_SEQ_BUFFER_ID_OVERFLOW Краткое имя: FESIOV, Серьезность: FB_SEV_ERROR

Пространство недостаточно для объявления другого последовательного буфера.

FB_ERR_SOFT_TIMER_ON Краткое имя: FESOON, Серьезность: FB_SEV_INFO

Поддерживается работа программного таймера (т.е. счет продолжается).

FB_ERR_SOFTWARE_TIMEOUT Краткое имя: FESOTO, Серьезность: FB_SEV_ERROR

Превышено значение лимита времени программного таймера.

FB_ERR_STATUS_NOT_FOUND Краткое имя: FENSF, Серьезность: FB_SEV_ERROR

Не может быть найдена Статусная информация для входного значения Указателя списка.

FB_ERR_STATUS_OVERFLOW Краткое имя: FESOV, Серьезность: FB_SEV_ERROR

Нет пространства, достаточного для запоминания Статусной истории.

FB_ERR_SUCCESS Краткое имя: FESUCC, Серьезность: FB_SEV_SUCCESS

Обнаружено неожидаемое состояние, не оказывающее влияния на выполнение Действий. Дополнительная информация может быть получена из Статусной истории.

FB_ERR_UNKNOWN Краткое имя: FEOOPS, Серьезность: FB_SEV_ERROR

Повстречался неизвестный Код ошибки.

FB_ERR_UNKNOWN_PARAMETER Краткое имя: FEUNPR, Серьезность: FB_SEV_ERROR

В Рутину был передан неизвестный параметр (не определенный Стандартом или Воплощением).

FB_ERR_UNS_PORT_OPERATION Краткое имя: FEUPOP, Серьезность: FB_SEV_ERROR

Запрошенное Действие не поддерживается текущим Портом. Этот код может быть выработан в Режиме немедленного исполнения или в Режиме отложенного исполнения при проверке или выполнении Списка.

FB_ERR_UNS_PARAMETER Краткое имя: FEUPAR, Серьезность: FB_SEV_WARNING

В стандартную Рутину был передан неподдерживаемый ОП.

FB_ERR_UNS_ROUTINE Краткое имя: FEUROU, Серьезность: FB_SEV_ERROR

Была вызвана Рутина, определяемая настоящим Стандартом, но неподдерживаемая Воплощением.

FB_ERR_USER_ARRAY_OVERFLOW Краткое имя: FEUAOV, Серьезность: FB_SEV_ERROR

Вызов FB_GET_ENVIRONMENT привел к переполнению массива Пользователя.

FB_ERR_WAIT_TIMEOUT Краткое имя: FEWTTO, Серьезность: FB_SEV_ERROR

Превышен лимит времени таймера WAIT.

FB_ERR_WARNING Краткое имя: FEWARN, Серьезность: FB_SEV_WARNING

Обнаружена ошибка с Серьезностью FB_SEV_WARNING. Дополнительная информация может быть получена из Статусной истории, если она существует.

Приложение А. ПЕРЕЧЕНЬ ИМЕН РУТИН И ПАРАМЕТРОВ

Это Приложение содержит список всех Имен Рутин, определенных данным стандартом, и аргументы этих Рутин. Рутины приведены в той последовательности, в какой они встречаются в стандарте.

Обозначения некоторых аргументов расшифровываются следующим образом:

retcod	=Return_code
eid	=Environment_id
pad	=Primary_Address
sad	=Secondary_Address
bufmode	=Buffer_Mode
buf	=buffer_specifier

Краткое имя		Полное имя	
FBOPEN	или	FB_OPEN	[retcod, rest are implementation dependent]
FBCLOS	или	FB_CLOSE	[retcod]
FCIENV	или	FB_CREATE_IMMEDIATE_ENVIRONMENT	[retcod, eid]
FCDENV	или	FB_CREATE_DELAYED_ENVIRONMENT	[retcod, eid, maxcalls]
FRLENV	или	FB_RELEASE_ENVIRONMENT	[retcod, eid]
FRSENV	или	FB_RESET_ENVIRONMENT	[retcod, eid]
FCOENV	или	FB_COPY_ENVIRONMENT	[retcod, from_eid, to_eid]
FGTENV	или	FB_GET_ENVIRONMENT	[retcod, eid, array_size, byte_count]
FSTENV	или	FB_SET_ENVIRONMENT	[retcod, eid, array, byte_count]
FLEXEC	или	FB_LIST_EXECUTE	[retcod, eid]
FLVAL	или	FB_LIST_VALIDATE	[retcod, eid]
FLGPTR	или	FB_LIST_GET_POINTER	[retcod, eid, list_pointer]
FBPINI	или	FB_PAR_INIT	[retcod, eid, param_id]
FBPSET	или	FB_PAR_SET	[retcod, eid, param_id, param_value]
FBPGET	или	FB_PAR_GET	[retcod, eid, param_id, param_value]
FBPRD	или	FB_PAR_READ	[retcod, eid, param_id, bufmode, buf]
FBPWRT	или	FB_PAR_WRITE	[retcod, eid, param_id, bufmode, buf]
FBPUSH	или	FB_PAR_PUSH	[retcod, eid]
FBPOP	или	FB_PAR_POP	[retcod, eid]
FBDSEQ	или	FB_DECLARE_SEQ_BUFFER	[retcod, eid, bufid, buffer, buf_size]
FBDEXT	или	FB_DECLARE_EXT_SEQ_BUFFER	[retcod, eid, bufid, buffer, buf_size]
FBREL	или	FB_RELEASE_SEQ_BUFFER	[retcod, eid, bufid]
FBRPTR	или	FB_READ_BUFFER_POINTER	[retcod, eid, bufid, bufmode, buf]
FBWPTR	или	FB_WRITE_BUFFER_POINTER	[retcod, eid, bufid, bufmode, buf]
FBDPTR	или	FB_DISPLACE_BUFFER_POINTER	[retcod, eid, bufid, bufmode, buf]
FRD	или	FB_READ_DAT	[retcod, eid, pad, sad, bufmode, buf]
FWD	или	FB_WRITE_DAT	[retcod, eid, pad, sad, bufmode, buf]
FRC	или	FB_READ_CSR	[retcod, eid, pad, sad, bufmode, buf]
FWC	или	FB_WRITE_CSR	[retcod, eid, pad, sad, bufmode, buf]
FRDM	или	FB_READ_DAT_MULT	[retcod, eid, pad, sad, bufmode, buf]
FWDM	или	FB_WRITE_DAT_MULT	[retcod, eid, pad, sad, bufmode, buf]
FRCM	или	FB_READ_CSR_MULT	[retcod, eid, pad, sad, bufmode, buf]
FWCM	или	FB_WRITE_CSR_MULT	[retcod, eid, pad, sad, bufmode, buf]
FRDB	или	FB_READ_DAT_BLOCK	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FWDB	или	FB_WRUTE_DAT_BLOCK	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FRCB	или	FB_READ_CSR_BLOCK	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FWCB	или	FB_WRITE_CSR_BLOCK	[retcod, eid, pad, sad, bufmode, buf, maxbytes]

FRDBM	или	FB_READ_DAT_BLOCK_MULT	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FWDBM	или	FB_WRITE_DAT_BLOCK_MULT	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FRCBM	или	FB_READ_CSR_BLOCK_MULT	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FWCBM	или	FB_WRITE_CSR_BLOCK_MULT	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FRDSA	или	FB_READ_DAT_SA	[retcod, eid, pad, bufmode, buf]
FWDSA	или	FB_WRITE_DAT_SA	[retcod, eid, pad, bufmode, buf]
FRCSA	или	FB_READ_CSR_SA	[retcod, eid, pad, bufmode, buf]
FWCSA	или	FB_WRITE_CSR_SA	[retcod, eid, pad, bufmode, buf]
FRLEN	или	FB_READ_LENGTH	[retcod, eid, bufmode, buf]
FRRT	или	FB_READ_ROUTE_TABLE	[retcod, eid, pad, route_sad, bufmode, buf]
FWRT	или	FB_WRITE_ROUTE_TABLE	[retcod, eid, pad, route_sad, bufmode, buf]
FRRTB	или	FB_READ_ROUTE_TABLE_BLOCK	[retcod, eid, pad, route_sad, bufmode, buf, maxbytes]
FWRTB	или	FB_WRITE_ROUTE_TABLE_BLOCK	[retcod, eid, pad, route_sad, bufmode, buf, maxbytes]
FMOVD	или	FB_MOVE_DAT	[retcod, eid, pad1, sad1, pad2, sad2]
FMOVC	или	FB_MOVE_CSR	[retcod, eid, pad1, sad1, pad2, sad2]
FMOVDB	или	FB_MOVE_DAT_BLOCK	[retcod, eid, pad1, sad1, pad2, sad2, bufmode, buf, maxbytes]
FMOVCB	или	FB_MOVE_CSR_BLOCK	[retcod, eid, pad1, sad1, pad2, sad2, bufmode, buf, maxbytes]
FMODD	или	FB_MODIFY_DAT	[retcod, eid, pad, sad, oper_id, operand]
FMODC	или	FB_MODIFY_CSR	[retcod, eid, pad, sad, oper_id, operand]
FRDSEQ	или	FB_READ_DAT_SEQUENTIAL	[retcod, eid, start_pad, end_pad, step_pad, sad, bufmode, buf, buf_size, maxbytes]
FRCSEQ	или	FB_READ_CSR_SEQUENTIAL	[retcod, eid, start_pad, end_pad, step_pad, sad, bufmode, buf, buf_size, maxbytes]
FRDTPS	или	FB_READ_DAT_TP_SCAN	[etcod, eid, pad, sad, bufmode, buf, buf_size, maxbytes]
FRCTPS	или	FB_READ_CSR_TP_SCAN	[etcod, eid, pad, sad, bufmode, buf, buf_size, maxbytes]
FDVAL	или	FB_DEVICE_ALLOCATE	[retcod, eid, pad]
FDVDE	или	FB_DEVICE_DEALLOCATE	[retcod, eid, pad]
FBFIM	или	FB_SEND_FIM	[retcod, eid, pad, receiver_block, bufmode, buf, message_len]
FCARB	или	FB_CYCLE_ARBITRATE	[retcod, eid]
FCREL	или	FB_CYCLE_RELEASE_BUS	[retcod, eid]
FCPD	или	FB_CYCLE_PA_DAT	[retcod, eid, bufmode, buf]
FCPC	или	FB_CYCLE_PA_CSR	[retcod, eid, bufmode, buf]
FCPDM	или	FB_CYCLE_PA_DAT_MULT	[retcod, eid, bufmode, buf]
FCPCM	или	FB_CYCLE_PA_CSR_MULT	[retcod, eid, bufmode, buf]
FCDISC	или	FB_CYCLE_DISCONNECT	[retcod, eid]
FCRW	или	FB_CYCLE_READ_WORD	[retcod, eid, bufmode, buf]
FCRSA	или	FB_CYCLE_READ_SA	[retcod, eid, bufmode, buf]
FCWW	или	FB_CYCLE_WRITE_WORD	[retcod, eid, bufmode, buf]
FCWSA	или	FB_CYCLE_WRITE_SA	[retcod, eid, bufmode, buf]
FCRB	или	FB_CYCLE_READ_BLOCK	[retcod, eid, bufmode, buf, maxbytes]
FCWB	или	FB_CYCLE_WRITE_BLOCK	[retcod, eid, bufmode, buf, maxbytes]
FLR	или	FB_LINE_READ	[retcod, eid, line, bufmode, buf]
FLW	или	FB_LINE_WRITE	[retcod, eid, line, bufmode, buf]

FLRI	или	FB_LINE_READ_INTERNAL	[retcod, eid, line, bufmode, buf]
FBSRC	или	FB_SR_CONNECT	[retcod, SRsource, port, routine]
FBSRD	или	FB_SR_DISCONNECT	[retcod, SRsource]
FBSREN	или	FB_SR_ENABLE	[retcod, port]
FBSRDS	или	FB_SR_DISABLE	[retcod, port]
FBFIRC	или	FB_FIR_CONNECT	[retcod, port, receiver, filter_mask, filter_value, filter_w или d, routine]
FBFIRD	или	FB_FIR_DISCONNECT	[retcod, port, receiver, routine]
FBFIRE	или	FB_FIR_ENABLE	[retcod, port]
FBFIRS	или	FB_FIR_DISABLE	[retcod, port]
FBCOMC	или	FB_COMPLETION_CONNECT	[retcod, eid, routine]
FBCOMD	или	FB_COMPLETION_DISCONNECT	[retcod, eid]
FCOMWT	или	FB_COMPLETION_WAIT	[retcod, eid]
FCOMTS	или	FB_COMPLETION_TEST	[retcod, eid, result]
FBPRST	или	FB_PORT_RESET	[retcod, port]
FNPALL	или	FB_PORT_ALLOCATE	[retcod, port]
FNPDEA	или	FB_PORT_DEALLOCATE	[retcod, port]
FNPALT	или	FB_PORT_ALLOCATE_TMO	[retcod, port, timeout]
FBPRPT	или	FB_PORT_STATUS_REPORT	[retcod, port, error_code]
FBPRPN	или	FB_PORT_STATUS_TRANSLATE	[retcod, port, error_code, message]
FBVERS	или	FB_GET_SOFTWARE_VERSION	[retcod, port, hw_type, hw_version, sw_type, sw_version]
FSSSEV	или	FB_STATUS_SET_SEVERITY	[retcod, eid, error_code, severity]
FSSR	или	FB_STATUS_SET_RESPONSE	[retcod, eid, error_code, response]
FSGR	или	FB_STATUS_GET_RESPONSE	[retcod, eid, error_code, response]
FSCON	или	FB_STATUS_CONNECT	[retcod, eid, routine]
FSDISC	или	FB_STATUS_DISCONNECT	[retcod, eid]
FSGSUM	или	FB_STATUS_GET_SUMMARY	[retcod, eid, error_code, bytes_read, bytes_written]
FSGSUP	или	FB_STATUS_GET_SUPPLEMENTARY	[retcod, eid, error_code, param, pointer]
FSGLSU	или	FB_STATUS_LIST_SUPPLEMENTARY	[retcod, eid, list_pointer, error_code, param, pointer]
FSFSUP	или	FB_FIND_SUPPLEMENTARY	[retcod, eid, error_code, param, pointer]
FSFLSU	или	FB_FIND_LIST_SUPPLEMENTARY	[retcod, eid, list_pointer, error_code, param, pointer]
FSEVER	или	FB_STATUS_SEVERITY	[error_code]
FMATCH	или	FB_STATUS_MATCH	[error_code1, error_code2]
FSTHR	или	FB_STATUS_THRESHOLD	[error_code, severity]
FSENC	или	FB_STATUS_ENCODE	[new_error_code, old_error_code, severity]
FSRPT	или	FB_STATUS_REPORT	[retcod, eid, error_code, param, pointer]
FSTRAN	или	FB_STATUS_TRANSLATE	[retcod, eid, error_code, param, pointer, message]

А1. Рутины, сортированные по Краткому имени

Краткое имя		Полное имя	
FBCLOS	или	FB_CLOSE	[retcod]
FBCOMC	или	FB_COMPLETION_CONNECT	[retcod, eid, routine]
FBCOMD	или	FB_COMPLETION_DISCONNECT	[retcod, eid]
FBDEXT	или	FB_DECLARE_EXT_SEQ_BUFFER	[retcod, eid, bufid, buffer, buf_size]
FBDPTR	или	FB_DISPLACE_BUFFER_POINTER	[retcod, eid, bufid, bufmode, buf]
FBDSEQ	или	FB_DECLARE_SEQ_BUFFER	[retcod, eid, bufid, buffer, buf_size]
FBFIM	или	FB_SEND_FIM	[retcod, eid, pad, recciver_block, bufmode, buf, message_len]

FBFIRC	или	FB_FIR_CONNECT	[retcod, port, receiver, filter_mask, filter_value, filter_w или d, routine]
FBFIRD	или	FB_FIR_DISCONNECT	[retcod, port, receiver, routine]
FBFIRE	или	FB_FIR_ENABLE	[retcod, port]
FBFIRS	или	FB_FIR_DISABLE	[retcod, port]
FBOPEN	или	FB_OPEN	[retcod, rest are implementation dependent]
FBPGET	или	FB_PAR_GET	[retcod, eid, param_id, param_value]
FBPINI	или	FB_PAR_INIT	[retcod, eid, param_id]
FBPOP	или	FB_PAR_POP	[retcod, eid]
FBPRD	или	FB_PAR_READ	[retcod, eid, param_id, bufmode, buf]
FBPRPN	или	FB_PORT_STATUS_TRANSLATE	[retcod, port, error_code, message]
FBPRPT	или	FB_PORT_STATUS_REPORT	[retcod, port, error_code]
FBPRST	или	FB_PORT_RESET	[retcod, port]
FBPSET	или	FB_PAR_SET	[retcod, eid, param_id, param_value]
FBPUSH	или	FB_PAR_PUSH	[retcod, eid]
FBPWRT	или	FB_PAR_WRITE	[retcod, eid, param_id, bufmode, buf]
FBREL	или	FB_RELEASE_SEQ_BUFFER	[retcod, eid, bufid]
FBRPTR	или	FB_READ_BUFFER_POINTER	[retcod, eid, bufid, bufmode, buf]
FBSRC	или	FB_SR_CONNECT	[retcod, SRsource, port, routine]
FBSRD	или	FB_SR_DISCONNECT	[retcod, SRsource]
FBSRDS	или	FB_SR_DISABLE	[retcod, port]
FBSREN	или	FB_SR_ENABLE	[retcod, port]
FBVERS	или	FB_GET_SOFTWARE_VERSION	[retcod, port, hw_type, hw_version, sw_type, sw_version]
FBWPTR	или	FB_WRITE_BUFFER_POINTER	[retcod, eid, bufid, bufmode, buf]
FCARB	или	FB_CYCLE_ARBITRATE	[retcod, eid]
FCDENV	или	FB_CREATE_DELAYED_ENVIRONMENT	[retcod, eid, maxcalls]
FCDISC	или	FB_CYCLE_DISCONNECT	[retcod, eid]
FCIENV	или	FB_CREATE_IMMEDIATE_ENVIRONMENT	[retcod, eid]
FCOENV	или	FB_COPY_ENVIRONMENT	[retcod, from_eid, to_eid]
FCOMTS	или	FB_COMPLETION_TEST	[retcod, eid, result]
FCOMWT	или	FB_COMPLETION_WAIT	[retcod, eid]
FCPC	или	FB_CYCLE_PA_CSR	[retcod, eid, bufmode, buf]
FCPCM	или	FB_CYCLE_PA_CSR_MULT	[retcod, eid, bufmode, buf]
FCPD	или	FB_CYCLE_PA_DAT	[retcod, eid, bufmode, buf]
FCPDM	или	FB_CYCLE_PA_DAT_MULT	[retcod, eid, bufmode, buf]
FCRB	или	FB_CYCLE_READ_BLOCK	[retcod, eid, bufmode, buf, maxbytes]
FCREL	или	FB_CYCLE_RELEASE_BUS	[retcod, eid]
FCRSA	или	FB_CYCLE_READ_SA	[retcod, eid, bufmode, buf]
FCRW	или	FB_CYCLE_READ_WORD	[retcod, eid, bufmode, buf]
FCWB	или	FB_CYCLE_WRITE_BLOCK	[retcod, eid, bufmode, buf, maxbytes]
FCWSA	или	FB_CYCLE_WRITE_SA	[retcod, eid, bufmode, buf]
FCWW	или	FB_CYCLE_WRITE_WORD	[retcod, eid, bufmode, buf]
FDVAL	или	FB_DEVICE_ALLOCATE	[retcod, eid, pad]
FDVDE	или	FB_DEVICE_DEALLOCATE	[retcod, eid, pad]
FGTENV	или	FB_GET_ENVIRONMENT	[retcod, eid, array_size, byte_count]
FLEXEC	или	FB_LIST_EXECUTE	[retcod, eid]
FLGPTR	или	FB_LIST_GET_POINTER	[retcod, eid, list_pointer]
FLR	или	FB_LINE_READ	[retcod, eid, line, bufmode, buf]
FLRI	или	FB_LINE_READ_INTERNAL	[retcod, eid, line, bufmode, buf]
FLVAL	или	FB_LIST_VALIDATE	[retcod, eid]
FLW	или	FB_LINE_WRITE	[retcod, eid, line, bufmode, buf]
FMATCH	или	FB_STATUS_MATCH	[error_code1, error_code2]
FMODC	или	FB_MODIFY_CSR	[retcod, eid, pad, sad, oper_id, operand]
FMODD	или	FB_MODIFY_DAT	[retcod, eid, pad, sad, oper_id, operand]
FMOVC	или	FB_MOVE_CSR	[retcod, eid, pad1, sad1, pad2, sad2]
FMOVCB	или	FB_MOVE_CSR_BLOCK	[retcod, eid, pad1, sad1, pad2, sad2,

			bufmode, buf, maxbytes]
FMOVD	или	FB_MOVE_DAT	[retcod, eid, pad1, sad1, pad2, sad2]
FMOVDB	или	FB_MOVE_DAT_BLOCK	[retcod, eid, pad1, sad1, pad2, sad2, bufmode, buf, maxbytes]
FNPALL	или	FB_PORT_ALLOCATE	[retcod, port]
FNPALT	или	FB_PORT_ALLOCATE_TMO	[retcod, port, timeout]
FNPDEA	или	FB_PORT_DEALLOCATE	[retcod, port]
FRC	или	FB_READ_CSR	[retcod, eid, pad, sad, bufmode, buf]
FRCB	или	FB_READ_CSR_BLOCK	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FRCBM	или	FB_READ_CSR_BLOCK_MULT	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FRCM	или	FB_READ_CSR_MULT	[retcod, eid, pad, sad, bufmode, buf]
FRCSA	или	FB_READ_CSR_SA	[retcod, eid, pad, bufmode, buf]
FRCSEQ	или	FB_READ_CSR_SEQUENTIAL	[retcod, eid, start_pad, end_pad, step_pad, sad, bufmode, buf, buf_size, maxbytes]
FRCTPS	или	FB_READ_CSR_TP_SCAN	[etcod, eid, pad, sad, bufmode, buf, buf_size, maxbytes]
FRD	или	FB_READ_DAT	[retcod, eid, pad, sad, bufmode, buf]
FRDB	или	FB_READ_DAT_BLOCK	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FRDBM	или	FB_READ_DAT_BLOCK_MULT	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FRDM	или	FB_READ_DAT_MULT	[retcod, eid, pad, sad, bufmode, buf]
FRDSA	или	FB_READ_DAT_SA	[retcod, eid, pad, bufmode, buf]
FRDSEQ	или	FB_READ_DAT_SEQUENTIAL	[retcod, eid, start_pad, end_pad, step_pad, sad, bufmode, buf, buf_size, maxbytes]
FRDTPS	или	FB_READ_DAT_TP_SCAN	[etcod, eid, pad, sad, bufmode, buf, buf_size, maxbytes]
FRLEN	или	FB_READ_LENGTH	[retcod, eid, bufmode, buf]
FRLENV	или	FB_RELEASE_ENVIRONMENT	[retcod, eid]
FRRT	или	FB_READ_ROUTE_TABLE	[retcod, eid, pad, route_sad, bufmode, buf]
FRRTB	или	FB_READ_ROUTE_TABLE_BLOCK	[retcod, eid, pad, route_sad, bufmode, buf, maxbytes]
FRSENV	или	FB_RESET_ENVIRONMENT	[retcod, eid]
FSCON	или	FB_STATUS_CONNECT	[retcod, eid, routine]
FSDISC	или	FB_STATUS_DISCONNECT	[retcod, eid]
FSENC	или	FB_STATUS_ENCODE	[new_error_code, old_error_code, severity]
FSEVER	или	FB_STATUS_SEVERITY	[error_code]
FSFLSU	или	FB_FIND_LIST_SUPPLEMENTARY	[retcod, eid, list_pointer, error_code, param, pointer]
FSFSUP	или	FB_FIND_SUPPLEMENTARY	[retcod, eid, error_code, param, pointer]
FSGLSU	или	FB_STATUS_LIST_SUPPLEMENTARY	[retcod, eid, list_pointer, error_code, param, pointer]
FSGR	или	FB_STATUS_GET_RESPONSE	[retcod, eid, error_code, response]
FSGSUM	или	FB_STATUS_GET_SUMMARY	[retcod, eid, error_code, bytes_read, bytes_written]
FSGSUP	или	FB_STATUS_GET_SUPPLEMENTARY	[retcod, eid, error_code, param, pointer]
FSRPT	или	FB_STATUS_REPORT	[retcod, eid, error_code, param, pointer]
FSSR	или	FB_STATUS_SET_RESPONSE	[retcod, eid, error_code, response]
FSSSEV	или	FB_STATUS_SET_SEVERITY	[retcod, eid, error_code, severity]
FSTENV	или	FB_SET_ENVIRONMENT	[retcod, eid, array, byte_count]
FSTHR	или	FB_STATUS_THRESHOLD	[error_code, severity]

FSTRAN	или	FB_STATUS_TRANSLATE	[retcod, eid, error_code, param, pointer, message]
FWC	или	FB_WRITE_CSR	[retcod, eid, pad, sad, bufmode, buf]
FWCB	или	FB_WRITE_CSR_BLOCK	[retcod, eid, pad, sad, bufmode, buf,
FWCBM	или	FB_WRITE_CSR_BLOCK_MULT	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FWCM	или	FB_WRITE_CSR_MULT	[retcod, eid, pad, sad, bufmode, buf]
FWCSA	или	FB_WRITE_CSR_SA	[retcod, eid, pad, bufmode, buf]
FWD	или	FB_WRITE_DAT	[retcod, eid, pad, sad, bufmode, buf]
FWDB	или	FB_WRUTE_DAT_BLOCK	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FWDBM	или	FB_WRITE_DAT_BLOCK_MULT	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FWDM	или	FB_WRITE_DAT_MULT	[retcod, eid, pad, sad, bufmode, buf]
FWDSA	или	FB_WRITE_DAT_SA	[retcod, eid, pad, bufmode, buf]
FWRT	или	FB_WRITE_ROUTE_TABLE	[retcod, eid, pad, route_sad, bufmode, buf]
FWRTB	или	FB_WRITE_ROUTE_TABLE_BLOCK	[retcod, eid, pad, route_sad, bufmode, buf, maxbytes] maxbytes]

А2. Рутины, сортированные по Полному имени

Полное имя		Краткое имя	
FB_CLOSE	или	FBCLOS	[retcod]
FB_COMPLETION_CONNECT	или	FBCOMC	[retcod, eid, routine]
FB_COMPLETION_DISCONNECT	или	FBCOMD	[retcod, eid]
FB_COMPLETION_TEST	или	FCOMTS	[retcod, eid, result]
FB_COMPLETION_WAIT	или	FCOMWT	[retcod, eid]
FB_COPY_ENVIRONMENT	или	FCOENV	[retcod, from_eid, to_eid]
FB_CREATE_DELAYED_ENVIRONMENT	или	FCDENV	[retcod, eid, maxcalls]
FB_CREATE_IMMEDIATE_ENVIRONMENT	или	FCIENV	[retcod, eid]
FB_CYCLE_ARBITRATE	или	FCARB	[retcod, eid]
FB_CYCLE_DISCONNECT	или	FCDISC	[retcod, eid]
FB_CYCLE_PA_CSR	или	FCPC	[retcod, eid, bufmode, buf]
FB_CYCLE_PA_CSR_MULT	или	FCPCM	[retcod, eid, bufmode, buf]
FB_CYCLE_PA_DAT	или	FCPD	[retcod, eid, bufmode, buf]
FB_CYCLE_PA_DAT_MULT	или	FCPDM	[retcod, eid, bufmode, buf]
FB_CYCLE_READ_BLOCK	или	FCRB	[retcod, eid, bufmode, buf, maxbytes]
FB_CYCLE_READ_SA	или	FCRSA	[retcod, eid, bufmode, buf]
FB_CYCLE_READ_WORD	или	FCRW	[retcod, eid, bufmode, buf]
FB_CYCLE_RELEASE_BUS	или	FCREL	[retcod, eid]
FB_CYCLE_WRITE_BLOCK	или	FCWB	[retcod, eid, bufmode, buf, maxbytes]
FB_CYCLE_WRITE_SA	или	FCWSA	[retcod, eid, bufmode, buf]
FB_CYCLE_WRITE_WORD	или	FCWW	[retcod, eid, bufmode, buf]
FB_DECLARE_EXT_SEQ_BUFFER	или	FBDEXT	[retcod, eid, bufid, buffer, buf_size]
FB_DECLARE_SEQ_BUFFER	или	FBDSEQ	[retcod, eid, bufid, buffer, buf_size]
FB_DEVICE_ALLOCATE	или	FDVAL	[retcod, eid, pad]
FB_DEVICE_DEALLOCATE	или	FDVDE	[retcod, eid, pad]
FB_DISPLACE_BUFFER_POINTER	или	FBDPTR	[retcod, eid, bufid, bufmode, buf]
FB_FIND_LIST_SUPPLEMENTARY	или	FSFLSU	[retcod, eid, list_pointer, error_code, param, pointer]
FB_FIND_SUPPLEMENTARY	или	FSFSUP	[retcod, eid, error_code, param, pointer]
FB_FIR_CONNECT	или	FBFIRC	[retcod, port, receiver, filter_mask, filter_value, filter_w или d, routine]
FB_FIR_DISABLE	или	FBFIRS	[retcod, port]
FB_FIR_DISCONNECT	или	FBFIRD	[retcod, port, receiver, routine]
FB_FIR_ENABLE	или	FBFIRE	[retcod, port]
FB_GET_ENVIRONMENT	или	FGTENV	[retcod, eid, array_size, byte_count]

FB_GET_SOFTWARE_VERSION	или	FBVERS	[retcod, port, hw_type, hw_version, sw_type, sw_version]
FB_LINE_READ	или	FLR	[retcod, eid, line, bufmode, buf]
FB_LINE_READ_INTERNAL	или	FLRI	[retcod, eid, line, bufmode, buf]
FB_LINE_WRITE	или	FLW	[retcod, eid, line, bufmode, buf]
FB_LIST_EXECUTE	или	FLEXEC	[retcod, eid]
FB_LIST_GET_POINTER	или	FLGPTR	[retcod, eid, list_pointer]
FB_LIST_VALIDATE	или	FLVAL	[retcod, eid]
FB_MODIFY_CSR	или	FMODC	[retcod, eid, pad, sad, oper_id, operand]
FB_MODIFY_DAT	или	FMODD	[retcod, eid, pad, sad, oper_id, operand]
FB_MOVE_CSR	или	FMOVC	[retcod, eid, pad1, sad1, pad2, sad2]
FB_MOVE_CSR_BLOCK	или	FMOVCB	[retcod, eid, pad1, sad1, pad2, sad2, bufmode, buf, maxbytes]
FB_MOVE_DAT	или	FMOVD	[retcod, eid, pad1, sad1, pad2, sad2]
FB_MOVE_DAT_BLOCK	или	FMOVDB	[retcod, eid, pad1, sad1, pad2, sad2, bufmode, buf, maxbytes]
FB_OPEN	или	FBOPEN	[retcod, rest are implementation dependent]
FB_PAR_GET	или	FBPGET	[retcod, eid, param_id, param_value]
FB_PAR_INIT	или	FBPINI	[retcod, eid, param_id]
FB_PAR_POP	или	FBPOP	[retcod, eid]
FB_PAR_PUSH	или	FBPUSH	[retcod, eid]
FB_PAR_READ	или	FBFRD	[retcod, eid, param_id, bufmode, buf]
FB_PAR_SET	или	FBPSET	[retcod, eid, param_id, param_value]
FB_PAR_WRITE	или	FBPWRT	[retcod, eid, param_id, bufmode, buf]
FB_PORT_ALLOCATE	или	FNPALL	[retcod, port]
FB_PORT_ALLOCATE_TMO	или	FNPALT	[retcod, port, timeout]
FB_PORT_DEALLOCATE	или	FNPDEA	[retcod, port]
FB_PORT_RESET	или	FBPRST	[retcod, port]
FB_PORT_STATUS_REPORT	или	FBPRPT	[retcod, port, error_code]
FB_PORT_STATUS_TRANSLATE	или	FBPRPN	[retcod, port, error_code, message]
FB_READ_BUFFER_POINTER	или	FBRPTR	[retcod, eid, bufid, bufmode, buf]
FB_READ_CSR	или	FRC	[retcod, eid, pad, sad, bufmode, buf]
FB_READ_CSR_BLOCK	или	FRCB	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FB_READ_CSR_BLOCK_MULT	или	FRCBM	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FB_READ_CSR_MULT	или	FRCM	[retcod, eid, pad, sad, bufmode, buf]
FB_READ_CSR_SA	или	FRCSA	[retcod, eid, pad, bufmode, buf]
FB_READ_CSR_SEQUENTIAL	или	FRCSEQ	[rètcod, eid, start_pad, end_pad, step_pad, sad, bufmode, buf, buf_size, maxbytes]
FB_READ_CSR_TP_SCAN	или	FRCTPS	[etcod, eid, pad, sad, bufmode, buf, buf_size, maxbytes]
FB_READ_DAT	или	FRD	[retcod, eid, pad, sad, bufmode, buf]
FB_READ_DAT_BLOCK	или	FRDB	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FB_READ_DAT_BLOCK_MULT	или	FRDBM	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FB_READ_DAT_MULT	или	FRDM	[retcod, eid, pad, sad, bufmode, buf]
FB_READ_DAT_SA	или	FRDSA	[retcod, eid, pad, bufmode, buf]
FB_READ_DAT_SEQUENTIAL	или	FRDSEQ	[retcod, eid, start_pad, end_pad, step_pad, sad, bufmode, buf, buf_size, maxbytes]
FB_READ_DAT_TP_SCAN	или	FRDTPS	[etcod, eid, pad, sad, bufmode, buf, buf_size, maxbytes]
FB_READ_LENGTH	или	FRLEN	[retcod, eid, bufmode, buf]
FB_READ_ROUTE_TABLE	или	FRRT	[retcod, eid, pad, route_sad, bufmode, buf]
FB_READ_ROUTE_TABLE_BLOCK	или	FRRTB	[retcod, eid, pad, route_sad, bufmode,

			buf, maxbytes]
FB_RELEASE_ENVIRONMENT	или	FRLENV	[retcod, eid]
FB_RELEASE_SEQ_BUFFER	или	FBREL	[retcod, eid, bufid]
FB_RESET_ENVIRONMENT	или	FRSENV	[retcod, eid]
FB_SEND_FIM	или	FBFIM	[retcod, eid, pad, receiver_block, bufmode, buf, message_len]
FB_SET_ENVIRONMENT	или	FSTENV	[retcod, eid, array,byte_count]
FB_SR_CONNECT	или	FBSRC	[retcod, SRsource, port, routine]
FB_SR_DISABLE	или	FBSRDS	[retcod, port]
FB_SR_DISCONNECT	или	FBSRD	[retcod, SRsource]
FB_SR_ENABLE	или	FBSREN	[retcod, port]
FB_STATUS_CONNECT	или	FSCON	[retcod, eid, routine]
FB_STATUS_DISCONNECT	или	FSDISC	[retcod, eid]
FB_STATUS_ENCODE	или	FSENC	[new_error_code, old_error_code, severity]
FB_STATUS_GET_RESPONSE	или	FSGR	[retcod, eid, error_code, response]
FB_STATUS_GET_SUMMARY	или	FSGSUM	[retcod, eid, error_code, bytes_read, bytes_written]
FB_STATUS_GET_SUPPLEMENTARY	или	FSGSUP	[retcod, eid, error_code, param, pointer]
FB_STATUS_LIST_SUPPLEMENTARY	или	FSGLSU	[retcod, eid, list_pointer, error_code, param, pointer]
FB_STATUS_MATCH	или	FMATCH	[error_code1, error_code2]
FB_STATUS_REPORT	или	FSRPT	[retcod, eid, error_code, param, pointer]
FB_STATUS_SET_RESPONSE	или	FSSR	[retcod, eid, error_code, response]
FB_STATUS_SET_SEVERITY	или	FSSSEV	[retcod, eid, error_code, severity]
FB_STATUS_SEVERITY	или	FSEVER	[error_code]
FB_STATUS_THRESHOLD	или	FSTHR	[error_code, severity]
FB_STATUS_TRANSLATE	или	FSTRAN	[retcod, eid, error_code, param, pointer, message]
FB_WRITE_BUFFER_POINTER	или	FBWPTR	[retcod, eid, bufid, bufmode, buf]
FB_WRITE_CSR	или	FWC	[retcod, eid, pad, sad, bufmode, buf]
FB_WRITE_CSR_BLOCK	или	FWCB	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FB_WRITE_CSR_BLOCK_MULT	или	FWCBM	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FB_WRITE_CSR_MULT	или	FWCM	[retcod, eid, pad, sad, bufmode, buf]
FB_WRITE_CSR_SA	или	FWCSA	[retcod, eid, pad, bufmode, buf]
FB_WRITE_DAT	или	FWD	[retcod, eid, pad, sad, bufmode, buf]
FB_WRITE_DAT_BLOCK_MULT	или	FWDBM	[retcod, eid, pad, sad, bufmode, buf, maxbytes]
FB_WRITE_DAT_MULT	или	FWDM	[retcod, eid, pad, sad, bufmode, buf]
FB_WRITE_DAT_SA	или	FWDSA	[retcod, eid, pad, bufmode, buf]
FB_WRITE_ROUTE_TABLE	или	FWRT	[retcod, eid, pad, route_sad, bufmode, buf]
FB_WRITE_ROUTE_TABLE_BLOCK	или	FWRTB	[retcod, eid, pad, route_sad, bufmode, buf, maxbytes]
FB_WRUTE_DAT_BLOCK	или	FWDB	[retcod, eid, pad, sad, bufmode, buf, maxbytes]

Приложение В. СПИСОК ЗАРЕЗЕРВИРОВАННЫХ ИМЕН

Имена, помеченные знаком *** , определены только в этой таблице и нигде более в этом стандарте не встречаются.

В1. Список, сортированный по Полному имени

Полное имя	Краткое имя
FB_BUFFER_SEQ	FBSEQ
FB_BUFFER_VALUE	FBVAL
FB_BUFFER_VAR	FBVAR
FB_CLOSE	FBCLOS
FB_COMPLETION_CONNECT	FBCOMC
FB_COMPLETION_DISCONNECT	FBCOMD
FB_COMPLETION_TEST	FCOMTS
FB_COMPLETION_WAIT	FCOMWT
FB_COPY_EVIRONMENT	FCOENV
FB_CREATE_DELAYED_EVIRONMENT	FCCDENV
FB_CREATE_IMMEDIATE_EVIRONMENT	FCCIENV
FB_CYCLE_ARBITRATE	FCARB
FB_CYCLE_DISCONNECT	FCDISC
FB_CYCLE_PA_CSR	FCPC
FB_CYCLE_PA_CSR_MULT	FCPCM
FB_CYCLE_PA_DAT	FCPD
FB_CYCLE_PA_DAT_MULT	FCPDM
FB_CYCLE_READ_BLOCK	FCRB
FB_CYCLE_READ_SA	FCRSA
FB_CYCLE_READ_WORD	FCRW
FB_CYCLE_RELEASE_BUS	FCREL
FB_CYCLE_WRITE_BLOCK	FCWB
FB_CYCLE_WRITE_SA	FCWSA
FB_CYCLE_WRITE_WORD	FCWW
FB_DECLARE_EXT_SEQ_BUFFER	FBDEXT
FB_DECLARE_SEQ_BUFFER	FBDSEQ
FB_DEFAULT_EID	FBDEID***
FB_DEVICE_ALLOCATE	FDVAL
FB_DEVICE_DEALLOCATE	FDVDE
FB_DISPLACE_BUFFER_POINTER	FBDPTR
FB_ERR_ABORT_ACTION	FEABA
FB_ERR_ACCESS_DATA_BUFFER	FEBUF
FB_ERR_ADDRESS_RELEASE_TIMEOUT	FEAKDO
FB_ERR_AK_TIMEOUT	FEAKTO
FB_ERR_ALREADY_CONNECT	FEACON
FB_ERR_BAD_PARAMETER_VALUE	FEBPRV
FB_ERR_BLOCK_DATA_SS	FEBSS2
FB_ERR_BUFFER_OVERFLOW	FEBOV
FB_ERR_CLOSED	FECLSD
FB_ERR_CONNECT	FECON
FB_ERR_DATA_PARITY_ERROR	FEDPE
FB_ERR_DATA_SS0	FEDSS0
FB_ERR_DATA_SS1	FEDSS1
FB_ERR_DATA_SS2	FEDSS2
FB_ERR_DATA_SS3	FEDSS3
FB_ERR_DATA_SS4	FEDSS4
FB_ERR_DATA_SS5	FEDSS5
FB_ERR_DATA_SS6	FEDSS6

FB_ERR_DATA_SS7	FEDSS7
FB_ERR_DISCONNECT	FEDCON
FB_ERR_DK_RELEASE_TIMEOUT	FEDKDO
FB_ERR_TIMEOUT	FEDKTO
FB_ERR_ENV_ACTIVE	FEENAC
FB_ERR_ENV_ID_OVERFLOW	FEEIOV
FB_ERR_ENV_OVERFLOW	FEENOV
FB_ERR_ENV_RELEASE	FEEREL
FB_ERR_ERROR	FEERR
FB_ERR_FATAL	FEFTL
FB_ERR_HARDWARE_ERROR	FEHERR
FB_ERR_HARDWARE_WARNING	FEHWRN
FB_ERR_HOLDING_ADDRESS	FEHDAS
FB_ERR_HOLDING_BUS	FEHDFB
FB_ERR_IGNORE	FEIGNO
FB_ERR_ILL_PARAMETER_VALUE	FEIPRV
FB_ERR_ILL_SEQ_BUFFER_ID	FEILSI
FB_ERR_IMPLEMENTATION	FEIMP
FB_ERR_INFO	FEIFO
FB_ERR_INVALID_ENV_ID	FEINEI
FB_ERR_INVALID_LIST_POINTER	FEINLI
FB_ERR_INVALID_PORT_ID	FEINPO
FB_ERR_INVALID_SEQ_BUFFER_ID	FEINSI
FB_ERR_LAST_LIST_POINTER	FELPTR
FB_ERR_LAST_PRIMARY_ADDRESS	FELPAD
FB_ERR_LAST_SECONDARY_ADDRESS	FELSAD
FB_ERR_LIST_OVERFLOW	FELOV
FB_ERR_LONG_TIMEOUT	FELOTO
FB_ERR_NORMAL	FENORM
FB_ERR_NOT_CONNECTED	FENCON
FB_ERR_NOT_HOLDING_MASTERSHIP	FENHMA
FB_ERR_NOT_MASTER	FENOMA
FB_ERR_NO_ADDRESS_CONNECTION	FENOAS
FB_ERR_NO_MORE_STATUS	FENMST
FB_ERR_NO_PRIVELEGE	FENPRV
FB_ERR_NO_SUPPLEMENTARY_STATUS	FENSST
FB_ERR_OPEN	FEOPEN
FB_ERR_OPERATION_PENDING	FEPEND
FB_ERR_PIPELINE_TRANSFER	FEPPL
FB_ERR_POINTER_OVERFLOW	FEPOV
FB_ERR_PORT_ACTIVE	FEPAC
FB_ERR_PORT_ALREADY_ALLOCATED	FEPAL
FB_ERR_PORT_NOT_ALLOCATED	FEPNAL
FB_ERR_PORT_NOT_AVAILABLE	FEPNOP
FB_ERR_PORT_OWNED	FEPOWN
FB_ERR_PORT_STATUS_REPORT	FEPRPT
FB_ERR_PORT_STATUS_TRANSLATE	FBPTRN
FB_ERR_PRIMARY_ADDRESS_SS0	FEASS0
FB_ERR_PRIMARY_ADDRESS_SS1	FEASS1
FB_ERR_PRIMARY_ADDRESS_SS2	FEASS2
FB_ERR_PRIMARY_ADDRESS_SS3	FEASS3
FB_ERR_PRIMARY_ADDRESS_SS4	FEASS4
FB_ERR_PRIMARY_ADDRESS_SS5	FEASS5
FB_ERR_PRIMARY_ADDRESS_SS6	FEASS6
FB_ERR_PRIMARY_ADDRESS_SS7	FEASS7
FB_ERR_READ_ONLY_PARAMETER	FEROPA
FB_ERR_RETRY	FERRTY
FB_ERR_SECONDARY_ADDRESS_TIMEOUT	FESATO
FB_ERR_SECONDARY_ADDRESS_SS0	FESSS0
FB_ERR_SECONDARY_ADDRESS_SS1	FESSS1

FB_ERR_SECONDARY_ADDRESS_SS2	FESSS2
FB_ERR_SECONDARY_ADDRESS_SS3	FESSS3
FB_ERR_SECONDARY_ADDRESS_SS4	FESSS4
FB_ERR_SECONDARY_ADDRESS_SS5	FESSS5
FB_ERR_SECONDARY_ADDRESS_SS6	FESSS6
FB_ERR_SECONDARY_ADDRESS_SS7	FESSS7
FB_ERR_SEQ_BUFFER_ACTIVE	FESBAC
FB_ERR_SEQ_BUFFER_EXISTS	FESBEX
FB_ERR_SEQ_BUFFER_ID_OVERFLOW	FESIOV
FB_ERR_SOFTWARE_TIMEOUT	FESOTO
FB_ERR_STATUS_NOT_FOUND	FENSF
FB_ERR_STATUS_OVERFLOW	FESOV
FB_ERR_SUCCESS	FESUCC
FB_ERR_UNKNOWN	FEOOPS
FB_ERR_UNKNOWN_PARAMETER	FEUNPR
FB_ERR_UNS_PARAMETER	FEUPAR
FB_ERR_UNS_PORT_OPERATION	FEUPOP
FB_ERR_UNS_ROUTINE	FEUROU
FB_ERR_USER_ARRAY_OVERFLOW	FEUAOV
FB_ERR_WAIT_TIMEOUT	FEWTTO
FB_ERR_WARNING	FEWARN
FB_FALSE	FFALSE***
FB_FIND_LIST_SUPPLEMENTARY	FSFLSU
FB_FIND_SUPPLEMENTARY	FSFSUP
FB_FIR_CONNECT	FBFIRC
FB_FIR_DISABLE	FBFIRS
FB_FIR_DISCONNECT	FBFIRD
FB_FIR_ENABLE	FBFIRE
FB_GET_ENVIRONMENT	FGTENV
FB_GET_SOFTWARE_VERSION	FBVERS
FB_LINE_AD	FBLAD
FB_LINE_AG	FBLAG
FB_LINE_AI	FBLAI
FB_LINE_AK	FBLAK
FB_LINE_AL	FBLAL
FB_LINE_AR	FBLAR
FB_LINE_AS	FBLAS
FB_LINE_BH	FBLBH
FB_LINE_DK	FBLDK
FB_LINE_DS	FBLDS
FB_LINE_EG	FBLEG
FB_LINE_GA	FBLGA
FB_LINE_GK	FBLGK
FB_LINE_MS	FBLMS
FB_LINE_PA	FBLPA
FB_LINE_PE	FBLPE
FB_LINE_RB	FBLRB
FB_LINE_RD	FBLRD
FB_LINE_READ	FLR
FB_LINE_READ_INTERNAL	FLRI
FB_LINE_SR	FBLSR
FB_LINE_SS	FBLSS
FB_LINE_TP	FBLTP
FB_LINE_WRITE	FLW
FB_LINE_WT	FBLWT
FB_LIST_EXECUTE	FLEXEC
FB_LIST_GET_POINTER	FLGPTR
FB_LIST_VALIDATE	FLVAL
FB_MODIFY_CSR	FMODC
FB_MODIFY_DAT	FMODD

FB_MOVE_CSR	FMOVC
FB_MOVE_CSR_BLOCK	FMOVCB
FB_MOVE_DAT	FMOVD
FB_MOVE_DAT_BLOCK	FMOVDB
FB_OPEN	FBOPEN
FB_PARITY_EVEN	FPPENV
FB_PARITY_NONE	FPPNON
FB_PARITY_ODD	FPPODD
FB_PAR_ADDRESS_CYCLE_RETRY_TIME	FPRTRT
FB_PAR_AK_TIMER	FPAKT
FB_PAR_ALL	FPALL
FB_PAR_ARBITRATION_LEVEL	FPARBL
FB_PAR_ASSURED_ACCESS	FPASSA
FB_PAR_BLOCKLET_SIZE	FPBLSZ
FB_PAR_CLOCK	FPCLK
FB_PAR_DELAY_EXEC	FPDLAY
FB_PAR_DISABLE_AK_TIMER	FPDAKT
FB_PAR_DISABLE_DK_TIMER	FPDDKT
FB_PAR_DISABLE_LONG_TIMER	FPDLOT
FB_PAR_DISABLE_SOFT_TIMER	FPDSOT
FB_PAR_DISABLE_WT_TIMER	FPDWTT
FB_PAR_DK_TIMER	FPDKT
FB_PAR_EG_UP	FPEG
FB_PAR_ENVIRONMENT_SIZE	FPENVS
FB_PAR_EXCEPTION_THRESHOLD	FPEXTH
FB_PAR_FIXED_NTA	FPFNTA
FB_PAR_GET	FBPGET
FB_PAR_HANDLER_THRESHOLD	FPHATH
FB_PAR_HOLD_AS	FPHOAS
FB_PAR_HOLD_BUS	FPHOLD
FB_PAR_HOLD_BUS_NO_AR	FPHOAR
FB_PAR_HOLD_BUS_ON_ERROR	FPHOER
FB_PAR_INIT	FBPINI
FB_PAR_LONG_TIMER	FPLOT
FB_PAR_LONG_TIMER_ON	FPLOON
FB_PAR_MESSAGE_THRESHOLD	FPMETH
FB_PAR_NO_ARBITRATION	FPNOAR
FB_PAR_NO_DATA_CYCLE	FPNODA
FB_PAR_NO_PRIM_ADDR	FPNOPA
FB_PAR_NO_SEC_ADDR	FPNOSA
FB_PAR_NO_SEC_ADDR_CSR	FPNOSC
FB_PAR_NO_STATUS	FPNSTA
FB_PAR_NO_WAIT	FPNOWT
FB_PAR_NUM_RETRY	FPRTRY
FB_PAR_PARITY	FPPRTY
FB_PAR_PIPELINE	FPPIPE
FB_PAR_POP	FBPOP
FB_PAR_PORT	FPPORT
FB_PAR_PRIORITIZED_ACCESS	FPPRIA
FB_PAR_PUSH	FBPUSH
FB_PAR_READ	FBPRD
FB_PAR_REPORT_TERSE	FPRSUM
FB_PAR_REPORT_THRESHOLD	FPRETH
FB_PAR_RETURN_SEVERITY	FPRSEV
FB_PAR_SET	FBPSET
FB_PAR_SHORT_DATA_WORD	FPSDW
FB_PAR_SHORT_WORD_SIZE	FPSWDS
FB_PAR_SOFT_TIMER_	FPSOFT
FB_PAR_SOFT_TIMER_ON	FPSOON
FB_PAR_WHOLE_BLOCK	FPWBLK

FB_PAR_WRITE	FBPWRT
FB_PAR_WT_TIMER	FPWTT
FB_PAR_ALLOCATE	FNPALL
FB_PAR_ALLOCATE_TMO	FNPALT
FB_PAR_DEALLOCATE	FNDEA
FB_PORT_RESET	FBPRST
FB_READ_BUFFER_POINTER	FBRPTR
FB_READ_CSR	FRC
FB_READ_CSR_BLOCK	FRCB
FB_READ_CSR_BLOCK_MULT	FRCBM
FB_READ_CSR_MULT	FRCM
FB_READ_CSR_SA	FRCSA
FB_READ_CSR_SEQUENTIAL	FRCSEQ
FB_READ_CSR_TP_SCAN	FRCTPS
FB_READ_DAT	FRD
FB_READ_DAT_BLOCK	FRDB
FB_READ_DAT_BLOCK_MULT	FRDBM
FB_READ_DAT_MULT	FRDM
FB_READ_DAT_SA	FRDSA
FB_READ_DAT_SEQUENTIAL	FRDSEQ
FB_READ_DAT_TP_SCAN	FRDTPS
FB_READ_LENGTH	FRLEN
FB_READ_ROUTE_TABLE	FRRT
FB_READ_ROUTE_TABLE_BLOCK	FRRTB
FB_RELEASE_ENVIRONMENT	FRLENV
FB_RELEASE_SEQ_BUFFER	FBREL
FB_RESET_ENVIRONMENT	FRSENV
FB_RESP_ABORT	FBRAB
FB_RESP_ABORT_ACTION	FBRABA
FB_RESP_IGNORE	FBRIG
FB_RESP_RETRY_ABORT	FBRRAB
FB_RESP_RETRY_ABORT_ACTION	FBRRAA
FB_RESP_RETRY_IGNORE	FBRRIG
FB_SEND_FIM	FBFIM
FB_SET_ENVIRONMENT	FSTENV
FB_SEV_ERROR	FSERR
FB_SEV_FATAL	FSFTL
FB_SEV_INFO	FSINFO
FB_SEV_NEVER	FSNEV
FB_SEV_SUCCESS	FSSUCC
FB_SEV_WARNING	FSWARN
FB_SR_CONNECT	FBSRC
FB_SR_DEFAULT	FBSRDF***
FB_SR_DISABLE	FBSRDS
FB_SR_DISCONNECT	FBSRD
FB_SR_ENABLE	FBSREN
FB_SR_SOURCE_ALL	FBSRSA***
FB_STATUS_CONNECT	FSCON
FB_STATUS_DISCONNECT	FSDISC
FB_STATUS_ENCODE	FSENC
FB_STATUS_GET_RESPONSE	FSGR
FB_STATUS_GET_SUMMARY	FSGSUM
FB_STATUS_GET_SUPPLEMENTARY	FSGSUP
FB_STATUS_LIST_SUPPLEMENTARY	FSGLSU
FB_STATUS_MATCH	FMATCH
FB_STATUS_REPORT	FSRPT
FB_STATUS_SET_RESPONSE	FSSR
FB_STATUS_SET_SEVERITY	FSSSEV
FB_STATUS_SEVERITY	FSEVER
FB_STATUS_THRESHOLD	FSTHR

FB_STATUS_TRANSLATE	FSTRAN
FB_TRUE	FTRUE***
FB_WRITE_BUFFER_POINTER	FBWPTR
FB_WRITE_CSR	FWC
FB_WRITE_CSR_BLOCK	FWCB
FB_WRITE_CSR_BLOCK_MULT	FWCBM
FB_WRITE_CSR_MULT	FWCM
FB_WRITE_CSR_SA	FWCSA
FB_WRITE_DAT	FWD
FB_WRITE_DAT_BLOCK	FWDB
FB_WRITE_DAT_BLOCK_MULT	FWDBM
FB_WRITE_DAT_MULT	FWDM
FB_WRITE_DAT_SA	FWDSA
FB_WRITE_ROUTE_TABLE	FWRT
FB_WRITE_ROUTE_TABLE_BLOCK	FWRTB

В2. Список, сортированный по Краткому имени

Краткое имя	Полное имя
FBCLOS	FB_CLOSE
FBCOMC	FB_COMPLETION_CONNECT
FBCOMD	FB_COMPLETION_DISCONNECT
FBDEID	FB_DEFAULT_EID***
FBDEXT	FB_DECLARE_EXT_SEQ_BUFFER
FBDPTR	FB_DISPLACE_BUFFER_POINTER
FBDSEQ	FB_DECLARE_SEQ_BUFFER
FBFIM	FB_SEND_FIM
FBFIRC	FB_FIR_CONNECT
FBFIRD	FB_FIR_DISCONNECT
FBFIRE	FB_FIR_ENABLE
FBFIRS	FB_FIR_DISABLE
FBLAD	FB_LINE_AD
FBLAG	FB_LINE_AG
FBLAI	FB_LINE_AI
FBLAK	FB_LINE_AK
FBLAL	FB_LINE_AL
FBLAR	FB_LINE_AR
FBLAS	FB_LINE_AS
FBLBH	FB_LINE_BH
FBLDK	FB_LINE_DK
FBLDS	FB_LINE_DS
FBLEG	FB_LINE_EG
FBLGA	FB_LINE_GA
FBLGK	FB_LINE_GK
FBLMS	FB_LINE_MS
FBLPA	FB_LINE_PA
FBLPE	FB_LINE_PE
FBLRB	FB_LINE_RB
FBLRD	FB_LINE_RD
FBLSR	FB_LINE_SR
FBLSS	FB_LINE_SS
FBLTP	FB_LINE_TP
FBLWT	FB_LINE_WT
FBOPEN	FB_OPEN
FBPGET	FB_PAR_GET
FBPINI	FB_PAR_INIT
FBPOP	FB_PAR_POP
FBPRD	FB_PAR_READ

FBPRST	FB_PORT_RESET
FBPSET	FB_PAR_SET
FBPTRN	FB_ERR_PORT_STATUS_TRANSLATE
FBPUSH	FB_PAR_PUSH
FBPWRT	FB_PAR_WRITE
FBRAB	FB_RESP_ABORT
FBRABA	FB_RESP_ABORT_ACTION
FBREL	FB_RELEASE_SEQ_BUFFER
FBRIG	FB_RESP_IGNORE
FBRPTR	FB_READ_BUFFER_POINTER
FBRRAA	FB_RESP_RETRY_ABORT_ACTION
FBRRAB	FB_RESP_RETRY_ABORT
FBRRIG	FB_RESP_RETRY_IGNORE
FBSEQ	FB_BUFFER_SEQ
FBSRC	FB_SR_CONNECT
FBSRD	FB_SR_DISCONNECT
FBSRDF	FB_SR_DEFAULT***
FBSRDS	FB_SR_DISABLE
FBSREN	FB_SR_ENABLE
FBSRSA	FB_SR_SOURCE_ALL***
FBVAL	FB_BUFFER_VALUE
FBVAR	FB_BUFFER_VAR
FBVERS	FB_GET_SOFTWARE_VERSION
FBWPTR	FB_WRITE_BUFFER_POINTER
FCARB	FB_CYCLE_ARBITRATE
FCCDENV	FB_CREATE_DELAYED_EVIRONMENT
FCCIENV	FB_CREATE_IMMEDIATE_EVIRONMENT
FCDISC	FB_CYCLE_DISCONNECT
FCOENV	FB_COPY_EVIRONMENT
FCOMTS	FB_COMPLETION_TEST
FCOMWT	FB_COMPLETION_WAIT
FCPC	FB_CYCLE_PA_CSR
FCPCM	FB_CYCLE_PA_CSR_MULT
FCPD	FB_CYCLE_PA_DAT
FCPDM	FB_CYCLE_PA_DAT_MULT
FCRB	FB_CYCLE_READ_BLOCK
FCREL	FB_CYCLE_RELEASE_BUS
FCRSA	FB_CYCLE_READ_SA
FCRW	FB_CYCLE_READ_WORD
FCWB	FB_CYCLE_WRITE_BLOCK
FCWSA	FB_CYCLE_WRITE_SA
FCWW	FB_CYCLE_WRITE_WORD
FDVAL	FB_DEVICE_ALLOCATE
FDVDE	FB_DEVICE_DEALLOCATE
FEABA	FB_ERR_ABORT_ACTION
FEACON	FB_ERR_ALREADY_CONNECT
FEAKDO	FB_ERR_ADDRESS_RELEASE_TIMEOUT
FEAKTO	FB_ERR_AK_TIMEOUT
FEASS0	FB_ERR_PRIMARY_ADDRESS_SS0
FEASS1	FB_ERR_PRIMARY_ADDRESS_SS1
FEASS2	FB_ERR_PRIMARY_ADDRESS_SS2
FEASS3	FB_ERR_PRIMARY_ADDRESS_SS3
FEASS4	FB_ERR_PRIMARY_ADDRESS_SS4
FEASS5	FB_ERR_PRIMARY_ADDRESS_SS5
FEASS6	FB_ERR_PRIMARY_ADDRESS_SS6
FEASS7	FB_ERR_PRIMARY_ADDRESS_SS7
FEBOV	FB_ERR_BUFFER_OVERFLOW
FEBPRV	FB_ERR_BAD_PARAMETER_VALUE
FEBSS2	FB_ERR_BLOCK_DATA_SS
FEBUF	FB_ERR_ACCESS_DATA_BUFFER

FECLSD	FB_ERR_CLOSED
FECON	FB_ERR_CONNECT
FEDCON	FB_ERR_DISCONNECT
FEDKDO	FB_ERR_DK_RELEASE_TIMEOUT
FEDKTO	FB_ERR_TIMEOUT
FEDPE	FB_ERR_DATA_PARITY_ERROR
FEDSS0	FB_ERR_DATA_SS0
FEDSS1	FB_ERR_DATA_SS1
FEDSS2	FB_ERR_DATA_SS2
FEDSS3	FB_ERR_DATA_SS3
FEDSS4	FB_ERR_DATA_SS4
FEDSS5	FB_ERR_DATA_SS5
FEDSS6	FB_ERR_DATA_SS6
FEDSS7	FB_ERR_DATA_SS7
FEEIOV	FB_ERR_ENV_ID_OVERFLOW
FEENAC	FB_ERR_ENV_ACTIVE
FEENOV	FB_ERR_ENV_OVERFLOW
FEEREL	FB_ERR_ENV_RELEASE
FEERR	FB_ERR_ERROR
FEFTL	FB_ERR_FATAL
FEHDAS	FB_ERR_HOLDING_ADDRESS
FEHDFB	FB_ERR_HOLDING_BUS
FEHERR	FB_ERR_HARDWARE_ERROR
FEHWRN	FB_ERR_HARDWARE_WARNING
FEIFO	FB_ERR_INFO
FEIGNO	FB_ERR_IGNORE
FEILSI	FB_ERR_ILL_SEQ_BUFFER_ID
FEIMP	FB_ERR_IMPLEMENTATION
FEINEI	FB_ERR_INVALID_ENV_ID
FEINLI	FB_ERR_INVALID_LIST_POINTER
FEINPO	FB_ERR_INVALID_PORT_ID
FEINSI	FB_ERR_INVALID_SEQ_BUFFER_ID
FEIPRV	FB_ERR_ILL_PARAMETER_VALUE
FELOTO	FB_ERR_LONG_TIMEOUT
FELOV	FB_ERR_LIST_OVERFLOW
FELPAD	FB_ERR_LAST_PRIMARY_ADDRESS
FELPTR	FB_ERR_LAST_LIST_POINTER
FELSAD	FB_ERR_LAST_SECONDARY_ADDRESS
FENCON	FB_ERR_NOT_CONNECTED
FENHMA	FB_ERR_NOT_HOLDING_MASTERSHIP
FENMST	FB_ERR_NO_MORE_STATUS
FENOAS	FB_ERR_NO_ADDRESS_CONNECTION
FENOMA	FB_ERR_NOT_MASTER
FENORM	FB_ERR_NORMAL
FENPRV	FB_ERR_NO_PRIVELEGE
FENSF	FB_ERR_STATUS_NOT_FOUND
FENSST	FB_ERR_NO_SUPPLEMENTARY_STATUS
FEOOPS	FB_ERR_UNKNOWN
FEOPEN	FB_ERR_OPEN
FEPAC	FB_ERR_PORT_ACTIVE
FEPAL	FB_ERR_PORT_ALREADY_ALLOCATED
FEPEND	FB_ERR_OPERATION_PENDING
FEPNAL	FB_ERR_PORT_NOT_ALLOCATED
FEPNOP	FB_ERR_PORT_NOT_AVAILABLE
FEPOV	FB_ERR_POINTER_OVERFLOW
FEPOWN	FB_ERR_PORT_OWNED
FEPPL	FB_ERR_PIPELINE_TRANSFER
FEPRPT	FB_ERR_PORT_STATUS_REPORT
FEROPA	FB_ERR_READ_ONLY_PARAMETER
FERRTY	FB_ERR_RETRY

FESATO	FB_ERR_SECONDARY_ADDRESS_TIMEOUT
FESBAC	FB_ERR_SEQ_BUFFER_ACTIVE
FESBEX	FB_ERR_SEQ_BUFFER_EXISTS
FESIOV	FB_ERR_SEQ_BUFFER_ID_OVERFLOW
FESOTO	FB_ERR_SOFTWARE_TIMEOUT
FESOV	FB_ERR_STATUS_OVERFLOW
FESSS0	FB_ERR_SECONDARY_ADDRESS_SS0
FESSS1	FB_ERR_SECONDARY_ADDRESS_SS1
FESSS2	FB_ERR_SECONDARY_ADDRESS_SS2
FESSS3	FB_ERR_SECONDARY_ADDRESS_SS3
FESSS4	FB_ERR_SECONDARY_ADDRESS_SS4
FESSS5	FB_ERR_SECONDARY_ADDRESS_SS5
FESSS6	FB_ERR_SECONDARY_ADDRESS_SS6
FESSS7	FB_ERR_SECONDARY_ADDRESS_SS7
FESUCC	FB_ERR_SUCCESS
FEUAOV	FB_ERR_USER_ARRAY_OVERFLOW
FEUNPR	FB_ERR_UNKNOWN_PARAMETER
FEUPAR	FB_ERR_UNS_PARAMETER
FEUPOP	FB_ERR_UNS_PORT_OPERATION
FEUROU	FB_ERR_UNS_ROUTINE
FEWARN	FB_ERR_WARNING
FEWTTO	FB_ERR_WAIT_TIMEOUT
FFALSE	FB_FALSE***
FGTENV	FB_GET_ENVIRONMENT
FLEXEC	FB_LIST_EXECUTE
FLGPTR	FB_LIST_GET_POINTER
FLR	FB_LINE_READ
FLRI	FB_LINE_READ_INTERNAL
FLVAL	FB_LIST_VALIDATE
FLW	FB_LINE_WRITE
FMATCH	FB_STATUS_MATCH
FMODC	FB_MODIFY_CSR
FMODD	FB_MODIFY_DAT
FMOVC	FB_MOVE_CSR
FMOVCB	FB_MOVE_CSR_BLOCK
FMOVD	FB_MOVE_DAT
FMOVDB	FB_MOVE_DAT_BLOCK
FNDEA	FB_PAR_DEALLOCATE
FNPALL	FB_PAR_ALLOCATE
FNPALT	FB_PAR_ALLOCATE_TMO
FPAKT	FB_PAR_AK_TIMER
FPALL	FB_PAR_ALL
FPARBL	FB_PAR_ARBITRATION_LEVEL
FPASSA	FB_PAR_ASSURED_ACCESS
FPBLSZ	FB_PAR_BLOCKLET_SIZE
FPCLK	FB_PAR_CLOCK
FPDAKT	FB_PAR_DISABLE_AK_TIMER
FPDDKT	FB_PAR_DISABLE_DK_TIMER
FPDKT	FB_PAR_DK_TIMER
FPDLAY	FB_PAR_DELAY_EXEC
FPDLOT	FB_PAR_DISABLE_LONG_TIMER
FPDSOT	FB_PAR_DISABLE_SOFT_TIMER
FPDWTT	FB_PAR_DISABLE_WT_TIMER
FPEG	FB_PAR_EG_UP
FPENVS	FB_PAR_ENVIRONMENT_SIZE
FPEXTH	FB_PAR_EXCEPTION_THRESHOLD
FPFNTA	FB_PAR_FIXED_NTA
FPHATH	FB_PAR_HANDLER_THRESHOLD
FPHOAR	FB_PAR_HOLD_BUS_NO_AR
FPHOAS	FB_PAR_HOLD_AS

FPHOER	FB_PAR_HOLD_BUS_ON_ERROR
FPHOLD	FB_PAR_HOLD_BUS
FPLOON	FB_PAR_LONG_TIMER_ON
FPLOT	FB_PAR_LONG_TIMER
FPMETH	FB_PAR_MESSAGE_THRESHOLD
FPNOAR	FB_PAR_NO_ARBITRATION
FPNODA	FB_PAR_NO_DATA_CYCLE
FPNOPA	FB_PAR_NO_PRIM_ADDR
FPNOSA	FB_PAR_NO_SEC_ADDR
FPNOSC	FB_PAR_NO_SEC_ADDR_CSR
FPNOWT	FB_PAR_NO_WAIT
FPNSTA	FB_PAR_NO_STATUS
FPPENV	FB_PARITY_EVEN
FPPIPE	FB_PAR_PIPELINE
FPPNON	FB_PARITY_NONE
FPPODD	FB_PARITY_ODD
FPPORT	FB_PAR_PORT
FPPRIA	FB_PAR_PRIORITIZED_ACCESS
FPPRTY	FB_PAR_PARITY
FPRETH	FB_PAR_REPORT_THRESHOLD
FPRSEV	FB_PAR_RETURN_SEVERITY
FPRSUM	FB_PAR_REPORT_TERSE
FPRTRT	FB_PAR_ADDRESS_CYCLE_RETRY_TIME
FPRTRY	FB_PAR_NUM_RETRY
FPSDW	FB_PAR_SHORT_DATA_WORD
FPSOFT	FB_PAR_SOFT_TIMER_
FPSOON	FB_PAR_SOFT_TIMER_ON
FPSWDS	FB_PAR_SHORT_WORD_SIZE
FPWBLK	FB_PAR_WHOLE_BLOCK
FPWTT	FB_PAR_WT_TIMER
FRC	FB_READ_CSR
FRCB	FB_READ_CSR_BLOCK
FRCBM	FB_READ_CSR_BLOCK_MULT
FRCM	FB_READ_CSR_MULT
FRCSA	FB_READ_CSR_SA
FRCSEQ	FB_READ_CSR_SEQUENTIAL
FRCTPS	FB_READ_CSR_TP_SCAN
FRD	FB_READ_DAT
FRDB	FB_READ_DAT_BLOCK
FRDBM	FB_READ_DAT_BLOCK_MULT
FRDM	FB_READ_DAT_MULT
FRDSA	FB_READ_DAT_SA
FRDSEQ	FB_READ_DAT_SEQUENTIAL
FRDTPS	FB_READ_DAT_TP_SCAN
FRLEN	FB_READ_LENGTH
FRLENV	FB_RELEASE_ENVIRONMENT
FRRT	FB_READ_ROUTE_TABLE
FRRTB	FB_READ_ROUTE_TABLE_BLOCK
FRSENV	FB_RESET_ENVIRONMENT
FSCON	FB_STATUS_CONNECT
FSDISC	FB_STATUS_DISCONNECT
FSENC	FB_STATUS_ENCODE
FSERR	FB_SEV_ERROR
FSEVER	FB_STATUS_SEVERITY
FSFLSU	FB_FIND_LIST_SUPPLEMENTARY
FSFSUP	FB_FIND_SUPPLEMENTARY
FSFTL	FB_SEV_FATAL
FSGLSU	FB_STATUS_LIST_SUPPLEMENTARY
FSGR	FB_STATUS_GET_RESPONSE
FSGSUM	FB_STATUS_GET_SUMMARY

FSGSUP	FB_STATUS_GET_SUPPLEMENTARY
FSINFO	FB_SEV_INFO
FSNEV	FB_SEV_NEVER
FSRPT	FB_STATUS_REPORT
FSSR	FB_STATUS_SET_RESPONSE
FSSSEV	FB_STATUS_SET_SEVERITY
FSSUCC	FB_SEV_SUCCESS
FSTENV	FB_SET_ENVIRONMENT
FSTHR	FB_STATUS_THRESHOLD
FSTRAN	FB_STATUS_TRANSLATE
FSWARN	FB_SEV_WARNING
FTRUE	FB_TRUE***
FWC	FB_WRITE_CSR
FWCB	FB_WRITE_CSR_BLOCK
FWCBM	FB_WRITE_CSR_BLOCK_MULT
FWCM	FB_WRITE_CSR_MULT
FWCSA	FB_WRITE_CSR_SA
FWD	FB_WRITE_DAT
FWDB	FB_WRITE_DAT_BLOCK
FWDBM	FB_WRITE_DAT_BLOCK_MULT
FWDM	FB_WRITE_DAT_MULT
FWDSA	FB_WRITE_DAT_SA
FWRT	FB_WRITE_ROUTE_TABLE
FWRTB	FB_WRITE_ROUTE_TABLE_BLOCK

Приложение С. Типы данных ФОРТРАН/77

В следующей таблице предлагается представление типов данных Фастбас при реализации на ФОРТРАНе 77.

FB_boolean	INTEGER *4
FB_buffer_mode	INTEGER *4
FB_environment_id	INTEGER *4
FB_error_code	INTEGER *4
FB_integer	INTEGER *4
FB_line	INTEGER *4
FB_list_pointer	INTEGER *4
FB_parameter_id	INTEGER *4
FB_parameter_value	INTEGER *4
FB_port_id	INTEGER *4
FB_routine	передается точкой входа "внешней" процедуры
FB_seq_buffer_id	INTEGER *4
FB_severity	INTEGER *4
FB_word	INTEGER *4
FB_string	CHARACTER

АЛФАВИТНЫЙ УКАЗАТЕЛЬ ТЕРМИНОВ

Русский алфавит

А

Автоматический докладчик ошибок 5
Адрес
  Вторичный 45, 48
  Первичный 45
  Широкий 11
  NTA 48
Адресный таймер 37
Адресный цикл
  Вторичный 45, 48
  Первичный 44, 53
Аппаратные ошибки 73

Б

Байтов счет
  запись 70
  при передаче блока 49
  число записанных 77
  чтение 70, 77
Без 33, 34
  Вторичного адресного цикла 34
Блоки данных 45
Блочки 46
Буферы данных 13
  аргументы доступа 40
  параметры 16
  Последовательные 40
    Внешний 41
    Внутренний 41
    доступ к указателю 41
    записать указатель 42
    объявление 41
    отключить 41
    размер 41
    сместить указатель 43
    считать указатель 42
  режим 16, 40
  спецификатор 40

В

Версия программы 68
Владение магистралью 44, 45
  отказ 56
Воплощение 11
Время вызова 14
Время исполнения 14
Время повторения попыток 39
Вторичный адрес 45, 48
Вызовы 18, 19

Г

Географическая адресация 33
Генерация контроля четности 33

Д

Данные
  Буферы 13, 40
  Рутины объединения 52
    управление циклом 45
  таймер 38
Длина блока 49
Долгий таймер 36
Дополнительный статус 70, 77
Доступ
  к линиям Фастбас 58
  к маршрутной таблице 50
  Обеспеченный 44

Ж

Ждать завершения операции 66

З

Закрытие Фастбас-сессии 17
Запись 13, 25
  данных 46
  сигнала на линии 59
Запрос на обслуживание 61
Значение в качестве параметра 40

И

Имя краткое 14, 58
  полное 14, 58
Исполнение
  время 14
  Режимы 13
  Списка 21

К

Категории Рутин 16
Код ошибки 70
Контроллер 11
Копирование среды 20
Краткое имя 14, 58

Л

Линии магистрали Фастбас 58, 59
  доступ к линиям 58
  сигналы на линиях 59
Логическая адресация 34

М

Максимальное число вызовов 18, 19
Массив как параметр Буфера 40

Н

Назначение Порта Фастбас 67
Назначение Порта с таймаутом 68
Направление параметра 14
Немедленное исполнение 13

Номер версии 68

О

Обработка ошибок Пользователя 76
Ожидание завершения операции 66
Ожидания таймер 37
Операционные параметры 12
  определение 23
Операция 11
Освобождение магистрали 44
Ответный код 39
  определение 73
Отказ от Владения магистралью 56
Отклики на ошибки 73, 74, 75
Открытие Фастбас-сессии 17
Отложенное исполнение 13
Отмена связи
  с FIR 65
  с SR 63
Отсоединение
  от Рутины завершения 67
  от Слуги 56
  от FIR 64
  от SR 62
  Порта 68
  Устройства 53
Ошибки
  Докладчик 75
  код 70
  отклик 73

П

Память,распределение для Среды 18
Параметры 12
  Рутин 14
  Буферов 16
Передача 11
  данных 50
  Принудительная 11, 31
  простая 11
  Рутины 44, 46
Передачи блоков 45, 47
  в Примитивных циклах 57
  длина 49
  условия управления 45
Переменная в качестве параметра 40
Повторные попытки
  время повторения 39
  коды возвратов 70
  число повторений 39
Получение Среды 20
Пользователь 11
  обработка ошибок 76
Порт 11
  назначение 68
  отмена 68
  сброс 67
Послать FIM 54
Последовательный буфер 40
Примитивные действия 11
Принудительная передача 11, 31
Приоритетный доступ 44
Присвоенная серьезность 74
Присоединение
  к Слуге 56
  Рутины завершения 66
  Устройства 53
  FIR 63
  SR 62
Проверка состояния завершения 66
Программный таймер 38
Простая операция 11
Простая передача 11
  Рутины 44, 46

Р

Размер Буфера 41
Размер Среды 19
Разъединение 46, 56
Распределение памяти для Среды 18
Режим
  Буфера данных 16, 40
  немедленного исполнения 13
  отложенного исполнения 13
Рутины
  Действия 13
  категории 16
  одиночного слова 46
  простых передач 44, 46

С

Сброс Порта 67
Сегмент 11
Серьезность 70, 71
  присвоенная 74
Сигнальные линии, доступ 58
Сканирование редких данных 52
Список 12
  исполнение 21
  подтверждение правильности 21
  указатель 21, 78
Среда
  восстановление 19
  идентификатор 13
  копирование 20
  обзор 11
  определение 12
  освобождение 19
  получение в user_array 20
  по умолчанию 17
  размер 20
  создание 18
Статус (состояние)
  Дополнительный 70, 77
  История 71
  код 70
  Суммарный 70

Т

Таймер
- Адресный 37
- данных 38
- Долгий 36
- ожидания 37
- Программный 38

Тип параметра 14

У

Удержание
- Владения магистралью 35
- при ошибке 36
- связки с адресом 35

Указатель
- Буфера данных 41, 42, 43
- возвращения 21
- Списка 21, 78

Управление
- арбитражем 44
- Циклом данных 45

Устройство
- отсоединение 53
- присоединение 53

Ф

Фастбас-действие 11

Фастбас
- линии см. Линии
- Порт 67
- сессия 17
- Устройство 53

Фастбас-прерывания
- сообщение 63
  - посылка 54
- приемник 63

Фронты AS, DS, AK, DK 55

Ч

Четность 33

Число повторений 39

Чтение 13, 25
- внутренней линии 60
- данных 46
- линий Порта 60
- сигналов Фастбас 58, 59

Чтение-модификация-запись 51

Ш

Широкая адресация 11

Латинский алфавит

Array_of 15

Associated_parameter 77

EG 37

Environment_id 13

GK (фронт) 55

FIR 63, 64, 65

IN 15

MS (коды) 47, 48, 49

NTA 48

OUT 15

RD (состояние) 47, 48

SET 13, 24

SR 61, 62, 63

TP-сканирование 53

Where_occured 77

WRITE 13, 25

WT 37

Имена Рутин (по алфавиту)

FBCLOS 17

FBCOMC 66

FBCOMD 67

FBDEXT 41

FBDPTR 42

FBDSEQ 41

FBFIM 54

FBFIRC 63

FBFIRD 64

FBFIRE 64

FBFIRS 65

FBLAD 58

FBLAG 58

FBLAI 59

FBLAK 58

FBLAL 58

FBLAR 58

FBLAS 58

FBLBH 59

FBLDK 58

FBLDS 58

FBLEG 58

FBLGA 59

FBLGK 58

FBLMS 58

FBLPA 58

FBLPE 58

FBLRB 59

FBLRD 58

FBLSR 59

FBLSS 58

FBLTP 59

FBLWT 58

FBOPEN 17

FBPGET 24

FBPINI 24

FBPOP 26

FBPRD 25

FBPRPT 69

FBPRST 67

FBPSET 24

FBPTRN 69

FBPUSH 26

FBPWRT 25

FBRAB 73

FBRABA 73, 74

FBREL 41

FBRIG 73
FBRPTR 42
FBRRAA 73, 74
FBRRAB 73
FBRRIG 73, 74
FBSRC 62
FBSRD 62
FBSRDS 63
FBSREN 62
FBVERS 68
FBWPTR 42
FB_boolean 14
FB_buffer_mode 15
FB_BUFFER_SEQ 40
FB_BUFFER_VALUE 40
FB_BUFFER_VAR 40
FB_CLOSE 17
FB_COMPLETION_CONNECT 66
FB_COMPLETION_DISCONNECT 67
FB_COMPLETION_TEST 66
FB_COMPLETION_WAIT 66
FB_COPY_EVIRONMENT 20
FB_CREATE_DELAYED_EVIRONMENT 18
FB_CREATE_IMMEDIATE_EVIRONMENT 18
FB_CYCLE_ARBITRATE 55
FB_CYCLE_DISCONNECT 56
FB_CYCLE_PA_CSR 56
FB_CYCLE_PA_CSR_MULT 56
FB_CYCLE_PA_DAT 56
FB_CYCLE_PA_DAT_MULT 56
FB_CYCLE_READ_BLOCK 57
FB_CYCLE_READ_SA 57
FB_CYCLE_READ_WORD 57
FB_CYCLE_RELEASE_BUS 56
FB_CYCLE_WRITE_BLOCK 57
FB_CYCLE_WRITE_SA 57
FB_CYCLE_WRITE_WORD 57
FB_DECLARE_EXT_SEQ_BUFFER 41
FB_DECLARE_SEQ_BUFFER 41
FB_DEFAULT_EID 13, 17
FB_DEVICE_ALLOCATE 53
FB_DEVICE_DEALLOCATE 53
FB_DISPLACE_BUFFER_POINTER 43
FB_envirionment_id 15
FB_error_code 15
FB_ERR_ABORT_ACTION 74, 83
FB_ERR_ACCESS_DATA_BUFFER 83
FB_ERR_ADDRESS_RELEASE_TIMEOUT 75
FB_ERR_AK_TIMEOUT 44, 83
FB_ERR_ALREADY_CONNECT 83
FB_ERR_BAD_PARAMETER_VALUE 83
FB_ERR_BLOCK_DATA_SS2 45, 84
FB_ERR_BUFFER_OVERFLOW 84
FB_ERR_CLOSED 84
FB_ERR_CONNECT 84
FB_ERR_DATA_PARITY_ERROR 84
FB_ERR_DATA_SSx 45
FB_ERR_DATA_SS0 84
FB_ERR_DATA_SS1 84
FB_ERR_DATA_SS2 84
FB_ERR_DATA_SS3 84
FB_ERR_DATA_SS4 84
FB_ERR_DATA_SS5 84
FB_ERR_DATA_SS6 84
FB_ERR_DATA_SS7 84
FB_ERR_DISCONNECT 84
FB_ERR_DK_RELEASE_TIMEOUT 85
FB_ERR_TIMEOUT 45, 85
FB_ERR_ENV_ACTIVE 85
FB_ERR_ENV_ID_OVERFLOW 85
FB_ERR_ENV_OVERFLOW 85
FB_ERR_ENV_RELEASE 85
FB_ERR_ERROR 85
FB_ERR_FATAL 85
FB_ERR_HARDWARE_ERROR 85
FB_ERR_HARDWARE_WARNING 85
FB_ERR_HOLDING_ADDRESS 85
FB_ERR_HOLDING_BUS 85
FB_ERR_IGNORE 74, 86
FB_ERR_ILL_PARAMETER_VALUE 86
FB_ERR_ILL_SEQ_BUFFER_ID 86
FB_ERR_IMPLEMENTATION 86
FB_ERR_INFO 86
FB_ERR_INVALID_ENV_ID 86
FB_ERR_INVALID_LIST_POINTER 86
FB_ERR_INVALID_PORT_ID 86
FB_ERR_INVALID_SEQ_BUFFER_ID 86
FB_ERR_LAST_LIST_POINTER 86
FB_ERR_LAST_PRIMARY_ADDRESS 86
FB_ERR_LAST_SECONDARY_ADDRESS 86
FB_ERR_LIST_OVERFLOW 87
FB_ERR_LONG_TIMEOUT 87
FB_ERR_LONG_TIMER_ON 87
FB_ERR_NORMAL 87
FB_ERR_NOT_CONNECTED 87
FB_ERR_NOT_HOLDING_MASTERSHIP
FB_ERR_NOT_MASTER 87
FB_ERR_NO_ADDRESS_CONNECTION 87
FB_ERR_NO_MORE_STATUS 87
FB_ERR_NO_PRIVELEGE 87
FB_ERR_NO_SUPPLEMENTARY_STATUS 87
FB_ERR_OPEN 88
FB_ERR_OPERATION_PENDING 88
FB_ERR_PIPELINE_TRANSFER 88
FB_ERR_POINTER_OVERFLOW 88
FB_ERR_PORT_ACTIVE 88
FB_ERR_PORT_ALREADY_ALLOCATED 88
FB_ERR_PORT_NOT_ALLOCATED 88
FB_ERR_PORT_NOT_AVAILABLE 88
FB_ERR_PORT_OWNED 88
FB_ERR_PRIMARY_ADDRESS_SSx 44
FB_ERR_PRIMARY_ADDRESS_SS0 88
FB_ERR_PRIMARY_ADDRESS_SS1 88
FB_ERR_PRIMARY_ADDRESS_SS2 89
FB_ERR_PRIMARY_ADDRESS_SS3 89
FB_ERR_PRIMARY_ADDRESS_SS4 89
FB_ERR_PRIMARY_ADDRESS_SS5 89
FB_ERR_PRIMARY_ADDRESS_SS6 89
FB_ERR_PRIMARY_ADDRESS_SS7 89
FB_ERR_READ_ONLY_PARAMETER 89

FB_ERR_RETRY 74, 89
FB_ERR_SAD_TIMEOUT 89
FB_ERR_SECONDARY_ADDRESS_SSx 89
FB_ERR_SECONDARY_ADDRESS_SS0 89
FB_ERR_SECONDARY_ADDRESS_SS1 89
FB_ERR_SECONDARY_ADDRESS_SS2 89
FB_ERR_SECONDARY_ADDRESS_SS3 89
FB_ERR_SECONDARY_ADDRESS_SS4 89
FB_ERR_SECONDARY_ADDRESS_SS5 90
FB_ERR_SECONDARY_ADDRESS_SS6 90
FB_ERR_SECONDARY_ADDRESS_SS7 90
FB_ERR_SECONDARY_ADDRESS_TIMEOUT 45
FB_ERR_SEQ_BUFFER_ACTIVE 90
FB_ERR_SEQ_BUFFER_EXISTS 90
FB_ERR_SEQ_BUFFER_ID_OVERFLOW 90
FB_ERR_SOFTWARE_TIMEOUT 90
FB_ERR_SOFT_TIMER_ON 90
FB_ERR_STATUS_NOT_FOUND 90
FB_ERR_STATUS_OVERFLOW 90
FB_ERR_SUCCESS 90
FB_ERR_UNKNOWN 90
FB_ERR_UNKNOWN_PARAMETER 90
FB_ERR_UNS_PARAMETER 91
FB_ERR_UNS_PORT_OPERATION 90
FB_ERR_UNS_ROUTINE 91
FB_ERR_USER_ARRAY_OVERFLOW 91
FB_ERR_WAIT_TIMEOUT 91
FB_ERR_WARNING 91
FB_FIND_LIST_SUPPLEMENTARY 79
FB_FIND_SUPPLEMENTARY 79
FB_FIR_CONNECT 63
FB_FIR_DISABLE 65
FB_FIR_DISCONNECT 64
FB_FIR_ENABLE 64
FB_GET_ENVIRONMENT 20
FB_GET_SOFTWARE_VERSION 68
FB_integer 15
FB_line 15
FB_LINE_AD 58
FB_LINE_AG 58
FB_LINE_AI 59
FB_LINE_AK 58
FB_LINE_AL 58
FB_LINE_AR 58
FB_LINE_AS 58
FB_LINE_BH 59
FB_LINE_DK 58
FB_LINE_DS 58
FB_LINE_EG 58
FB_LINE_GA 59
FB_LINE_GK 58
FB_LINE_MS 58
FB_LINE_PA 58
FB_LINE_PE 58
FB_LINE_RB 59
FB_LINE_RD 58
FB_LINE_READ 59
FB_LINE_READ_INTERNAL 59
FB_LINE_SR 59
FB_LINE_SS 58
FB_LINE_TP 59
FB_LINE_WRITE 59
FB_LINE_WT 58
FB_LIST_EXECUTE 14, 21, 71
FB_LIST_GET_POINTER 21, 79
FB_list_pointer 15
FB_LIST_VALIDATE 21
FB_MODIFY_CSR 51
FB_MODIFY_DAT 51
FB_MOD_AND 52
FB_MOD_CLEAR 51
FB_MOD_LSHIFT 52
FB_MOD_NOT 52
FB_MOD_OR 52
FB_MOD_RSHIFT 52
FB_MOD_SET 51
FB_MOD_XOR 52
FB_MOVE_CSR 50
FB_MOVE_CSR_BLOCK 50
FB_MOVE_DAT 50
FB_MOVE_DAT_BLOCK 50
FB_OPEN 17
FB_operator 15
FB_parameter_id 15
FB_parameter_value 15
FB_PARITY_EVEN 33
FB_PARITY_NONE 33
FB_PARITY_ODD 33
FB_PAR_ADDRESS_CYCLE_RETRY_TIME 39
FB_PAR_AK_TIMER 37
FB_PAR_ALL 24
FB_PAR_ARBITRATION_LEVEL 30
FB_PAR_ASSURED_ACCESS 30
FB_PAR_BLOCKLET_SIZE 31
FB_PAR_CLOCK 32
FB_PAR_DELAY_EXEC 27
FB_PAR_DISABLE_AK_TIMER 37
FB_PAR_DISABLE_DK_TIMER 38
FB_PAR_DISABLE_LONG_TIMER 36
FB_PAR_DISABLE_SOFT_TIMER 38
FB_PAR_DISABLE_WT_TIMER 37
FB_PAR_DK_TIMER 38
FB_PAR_EG_UP 33
FB_PAR_ENVIRONMENT_SIZE 27
FB_PAR_EXCEPTION_THRESHOLD 29
FB_PAR_FIXED_NTA 32
FB_PAR_GET 24
FB_PAR_HANDLER_THRESHOLD 28, 75
FB_PAR_HOLD_AS 35
FB_PAR_HOLD_BUS 35
FB_PAR_HOLD_BUS_NO_AR 35
FB_PAR_HOLD_BUS_ON_ERROR 36
FB_PAR_INIT 24
FB_PAR_LONG_TIMER 36
FB_PAR_LONG_TIMER_ON 36
FB_PAR_MESSAGE_THRESHOLD 29, 75
FB_PAR_NO_ARBITRATION 33
FB_PAR_NO_DATA_CYCLE 34
FB_PAR_NO_PRIM_ADDR 34
FB_PAR_NO_SEC_ADDR 34, 45

FB_PAR_NO_SEC_ADDR_CSR 34, 45
FB_PAR_NO_STATUS 27, 71
FB_PAR_NO_WAIT 27
FB_PAR_NUM_RETRY 39
FB_PAR_PARITY 33
FB_PAR_PIPELINE 31
FB_PAR_POP 26
FB_PAR_PORT 26
FB_PAR_PRIORITIZED_ACCESS 31
FB_PAR_PUSH 26
FB_PAR_READ 25
FB_PAR_REPORT_ACTIONS 29
FB_PAR_REPORT_TERSE 29, 76
FB_PAR_REPORT_THRESHOLD 28, 75
FB_PAR_RETURN_SEVERITY 28
FB_PAR_SET 24
FB_PAR_SHORT_DATA_WORD 32
FB_PAR_SHORT_WORD_SIZE 33
FB_PAR_SOFT_TIMER_ 38
FB_PAR_SOFT_TIMER_ON 39
FB_PAR_WHOLE_BLOCK 31
FB_PAR_WRITE 25
FB_PAR_WT_TIMER 37
FB_PAR_ALLOCATE 68
FB_PAR_ALLOCATE_TMO 68
FB_PAR_DEALLOCATE 68
FB_port_id 15
FB_PORT_RESET 67
FB_PORT_STATUS_REPORT 69
FB_PORT_STATUS_TRANSLATE 69
FB_READ_BUFFER_POINTER 42
FB_READ_CSR 46
FB_READ_CSR_BLOCK 47
FB_READ_CSR_BLOCK_MULT 47
FB_READ_CSR_MULT 46
FB_READ_CSR_SA 48
FB_READ_CSR_SEQUENTIAL 52
FB_READ_CSR_TP_SCAN 53
FB_READ_DAT 46
FB_READ_DAT_BLOCK 47
FB_READ_DAT_BLOCK_MULT 47
FB_READ_DAT_MULT 46
FB_READ_DAT_SA 48
FB_READ_DAT_SEQUENTIAL 52
FB_READ_DAT_TP_SCAN 53
FB_READ_LENGTH 49
FB_READ_ROUTE_TABLE 50
FB_READ_ROUTE_TABLE_BLOCK 50
FB_RELEASE_ENVIRONMENT 19
FB_RELEASE_SEQ_BUFFER 41
FB_RESET_ENVIRONMENT 19
FB_response_code 15
FB_RESP_ABORT 73
FB_RESP_ABORT_ACTION 73, 74
FB_RESP_IGNORE 73
FB_RESP_RETRY_ABORT 73
FB_RESP_RETRY_ABORT_ACTION 73
FB_RESP_RETRY_IGNORE 73, 74
FB_routine 15
FB_SEND_FIM 54
FB_seq_buffer 15
FB_SET_ENVIRONMENT 20
FB_severity 15
FB_SEV_ERROR 71, 72
FB_SEV_FATAL 71, 72
FB_SEV_INFO 71
FB_SEV_NEVER 71, 72
FB_SEV_SUCCESS 71
FB_SEV_WARNING 71
FB_SR_CONNECT 62
FB_SR_DEFAULT 62
FB_SR_DISABLE 63
FB_SR_DISCONNECT 62
FB_SR_ENABLE 62
FB_STATUS_CONNECT 28, 76
FB_STATUS_DISCONNECT 76
FB_STATUS_ENCODE 81
FB_STATUS_GET_RESPONSE 75
FB_STATUS_GET_SUMMARY 77
FB_STATUS_GET_SUPPLEMENTARY 78
FB_STATUS_LIST_SUPPLEMENTARY 78
FB_STATUS_MATCH 80
FB_STATUS_REPORT 81
FB_STATUS_SET_RESPONSE 75
FB_STATUS_SET_SEVERITY 72
FB_STATUS_SEVERITY 80
FB_STATUS_THRESHOLD 80
FB_STATUS_TRANSLATE 81
FB_word 15
FB_WRITE_BUFFER_POINTER 42
FB_WRITE_CSR 47
FB_WRITE_CSR_BLOCK 47
FB_WRITE_CSR_BLOCK_MULT 47
FB_WRITE_CSR_MULT 47
FB_WRITE_CSR_SA 48
FB_WRITE_DAT 47
FB_WRITE_DAT_BLOCK 47
FB_WRITE_DAT_BLOCK_MULT 47
FB_WRITE_DAT_MULT 47
FB_WRITE_DAT_SA 48
FB_WRITE_ROUTE_TABLE 50
FB_WRITE_ROUTE_TABLE_BLOCK 50
FCARB 55
FCDENV 18
FCDISC 56
FCIENV 18
FCOENV 20
FCOMTS 66
FCOMWT 66
FCPC 56
FCPCM 56
FCPD 56
FCPDM 56
FCRB 57
FCREL 56
FCRSA 57
FCRW 57
FCWB 56
FCWSA 57
FCWW 57

FDVAL 53
FDVDE 53
FEABA 83
FEACON 83
FEAKDO 83
FEAKTO 83
FEASSx 44
FEASS0..88
FEASS1..88
FEASS2..88
FEASS3..88
FEASS4..88
FEASS5..88
FEASS6..88
FEASS7..88
FEBOV 84
FEBPRV 83
FEBSS2 45, 84
FEBUF 83
FECLSD 84
FECON 84
FEDCON 84
FEDKDO 85
FEDKTO 45, 85
FEDPE 84
FEDSSx 45
FEDSS0 84
FEDSS1 84
FEDSS2 84
FEDSS3 84
FEDSS4 84
FEDSS5 84
FEDSS6 84
FEDSS7 84
FEEIOV 85
FEENAC 85
FEENOV 85
FEEREL 85
FEERR 85
FEFTL 85
FEHDAS 85
FEHDFB 85
FEHERR 85
FEHWRN 85
FEIGNO 86
FEILSI 86
FEIMP 86
FEINEI 86
FEINFO 86
FEINLI 86
FEINPO 86
FEINSI 86
FEIPRV 86
FELOON 87
FELOTO 87
FELOV 87
FELPAD 86
FELPTR 86
FELSAD 86
FENCON 87
FENHMA 87
FENMST 87
FENOAS 87
FENOMA 87
FENORM 87
FENPRV 87
FENSF 90
FENSST 87
FEOOPS 90
FEOPEN 88
FEPAC 88
FEPAL 88
FEPEND 88
FEPNAL 88
FEPNOP 88
FEPOV 88
FEPOWN 88
FEPPL 88
FEROPA 89
FERTY 89
FESATO 45, 89
FESBAC 90
FESBEX 90
FESIOV 90
FESOON 90
FESOTO 90
FESOV 98
FESSSx 45
FESSS0 89
FESSS1 89
FESSS2 89
FESSS3 89
FESSS4 89
FESSS5 89
FESSS6 89
FESSS7 89
FESUCC 90
FEUAOV 91
FEUNPR 90
FEUPAR 91
FEUPOP 90
FEUROU 91
FEWARN 91
FEWTTO 91
FGTENV 20
FLEXEC 21
FLGPTR 21
FLR 59
FLRI 60
FLVAL 21
FLW 59
FMAND 52
FMATCH 80
FMCLR 51
FMLSH 52
FMNOT 52
FMODC 51
FMODD 51
FMOR 52
FMOVC 50

FMOVCB 50
FMOVD 50
FMOVDB 50
FMRSH 52
FMSET 51
FMXOR 52
FNPALL 68
FNPALT 68
FNPDEA 68
FPAKT 37
FPALL 24
FPARBL 30
FPASSA 30
FPBLSZ 31
FPCLK 32
FPDAKT 37
FPDDKT 38
FPDKT 38
FPDLAY 27
FPDLOT 36
FPDSOT 39
FPDWTT 37
FPEG 33
FPENVS 27
FPEXTH 29
FPFNTA 32
FPHATH 28
FPHOAR 35
FPHOAS 35
FPHOER 36
FPHOLD 35
FPLOT 36
FPMETH 29
FPNOAR 33
FPNODA 34
FPNOPA 34
FPNOSA 34
FPNOSC 34
FPNOWT 27
FPNSTA 27
FPOLOT 36
FPOSOT 39
FPPEVN 33
FPPIPE 31
FPPNON 33
FPPODD 33
FPPORT 26
FPPRIA 31
FPPRTY 33
FPRACT 29
FPRETH 28
FPRSEV 28
FPRSUM 29
FPRTRT 39
FPRTRY 39
FPSDW 32
FPSOFT 38
FPSWDS 33
FPWBLK 31
FPWTT 37
FRC 46
FRCB 47
FRCBM 47
FRCM 46
FRCSA 48
FRCSEQ 52
FRCTPS 53
FRD 46
FRDB 47
FRDBM 47
FRDM 46
FRDSA 48
FRDSEQ 52
FRDTPS 53
FRLEN 49
FRLENV 19
FRRT 50
FRRTB 50
FRSENV 19
FSCON 76
FSDISC 76
FSENC 81
FSERR 64, 65
FSEVER 80
FSFLSU 79
FSFSUP 79
FSFTL 71, 72
FSGLSU 78
FSGR 75
FSGSUM 77
FSGSUP 78
FSINFO 71
FSNEV 71, 72
FSRPT 81
FSSR 75
FSSSEV 72
FSSUCC 71
FSTENV 20
FSTHR 80
FSTRAN 81
FSWARN 71
FWC 46
FWCB 47
FWCBM 47
FWCM 46
FWCSA 48
FWD 46
FWDB 47
FWDBM 47
FWDM 46
FWDSA 48
FWRT 50
FWRTB 50

ИНФОРМАЦИОННЫЕ ДАННЫЕ

1. ПОДГОТОВЛЕН И ВНЕСЕН Всероссийским научно-исследовательским институтом стандартизации и сертификации в машиностроении (ВНИИНМАШ)

2. УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Постановлением Госстандарта России от 24.06.93 № 159.
Настоящий стандарт разработан на основе прямого применения Публикации МЭК 1052–91 и определяет стандартные программы для доступа к системе Фастбас стандартизованной модульной быстродействующей системе сбора данных и управления, определенной стандартом ГОСТ 34.340–91 (МЭК 935–90)

3. ССЫЛОЧНЫЕ НОРМАТИВНО-ТЕХНИЧЕСКИЕ ДОКУМЕНТЫ

Обозначение отечественного нормативно-технического документа, на который дана ссылка	Обозначение соответствующего стандарта МЭК	Глава, пункт, подпункт, в котором приведена ссылка
34.340–91	МЭК 935–90	1; 1.1; 2.1

Редактор *В.П. Огурцов*
Технический редактор *В.Н. Прусакова*
Корректор *В.С. Черная*

gost.e-ecolog.ru

но в набор 15.07.93. Подписано в печать 30.08.93. Усл.печ.л. 15,0. Усл.кр.-отт. 15,25
Уч.-изд. л. 10,53. Тираж 306. С559

Ордена "Знак Почета" Издательство стандартов,
107076, Москва, Колодезный пер., 14.
Калужская типография стандартов, ул. Московская, 256. Зак. 1860