ГОСТ Р ИСО/МЭК 10038—99

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационная технология

ПЕРЕДАЧА ДАННЫХ И ОБМЕН ИНФОРМАЦИЕЙ МЕЖДУ СИСТЕМАМИ

ЛОКАЛЬНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ

МОСТЫ НА ПОДУРОВНЕ УПРАВЛЕНИЯ ДОСТУПОМ К СРЕДЕ

Издание официальное

БЗ 4—98/593

ГОССТАНДАРТ РОССИИ
Москва

gost.e-ecolog.ru

Предисловие

1 РАЗРАБОТАН Московским научно-исследовательским центром (МНИЦ) Комитета при Президенте Российской Федерации по политике информатизации

ВНЕСЕН Техническим комитетом по стандартизации ТК 22 «Информационные технологии»

2 ПРИНЯТ И ВВЕДЕН В ДЕЙСТВИЕ Постановлением Госстандарта России от 7 октября 1999 г. № 331-ст

Настоящий стандарт содержит полный аутентичный текст международного стандарта ИСО/МЭК 10038—93. «Информационная технология. Передача данных и обмен информацией между системами. Локальные вычислительные сети. Мосты на подуровне управления доступом к среде»

3 ВВЕДЕН ВПЕРВЫЕ

© ИПК Издательство стандартов, 2000

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Госстандарта России

Содержание

1 Общие положения . . . 1
1.1 Область применения . . . 1
1.2 Нормативные ссылки . . . 2
1.3 Определения . . . 2
1.4 Сокращения . . . 2
1.5 Соответствие . . . 3
1.6 Рекомендации . . . 4
1.6.1 Административное управление . . . 4
1.7 Специфичные для УДС мостовые методы . . . 4
2 Обеспечение услуг УДС . . . 4
2.1 Общие положения . . . 5
2.2 Постоянство услуг УДС . . . 5
2.3 Обеспечение качества услуг . . . 5
2.3.1 Доступность услуг . . . 5
2.3.2 Потеря кадров . . . 6
2.3.3 Нарушение последовательности кадров . . . 6
2.3.4 Дублирование кадров . . . 6
2.3.5 Транзитная задержка . . . 7
2.3.6 Время существования кадра . . . 7
2.3.7 Коэффициент необнаруженных ошибок кадра . . . 7
2.3.8 Максимальный размер сервисного блока данных . . . 7
2.3.9 Приоритет . . . 7
2.3.10 Пропускная способность . . . 8
2.4 Внутренние услуги подуровня, обеспечиваемые внутри моста УДС . . . 8
2.5 Поддержка внутренних услуг подуровня специфичными для УДС процедурами . . . 9
2.5.1 Обеспечение услуг ЛВС КДОН/ОК по ГОСТ 34.913.3 . . . 9
2.5.2 Обеспечение услуг ЛВС по ГОСТ 34.913.4 (маркерный метод доступа к шине) . . . 10
2.5.3 Обеспечение услуг ЛВС по ГОСТ Р ИСО/МЭК 8802—5 (маркерный метод доступа к кольцу) . . . 11
2.5.4 Обеспечение услуг волоконно-оптическим интерфейсом передачи данных (ВОРИПД) 12
3 Принципы работы . . . 13
3.1 Операции моста . . . 13
3.1.1 Ретрансляция . . . 13
3.1.2 Информация фильтрации . . . 13
3.1.3 Диспетчер моста . . . 14
3.2 Архитектура моста . . . 14
3.3 Модель операций . . . 16
3.4 Информация о состоянии порта . . . 18
3.5 Прием кадров . . . 18
3.6 Передача кадров . . . 18
3.7 Продвижение кадра . . . 19
3.7.1 Условия продвижения . . . 19
3.7.2 Проверка дублирования адресов УЛЗ . . . 19
3.7.3 Постановка кадров в очередь . . . 19
3.7.4 Преобразование приоритетов . . . 19
3.7.5 Перерасчет КПК . . . 20
3.7.6 Модель . . . 20
3.8 Процесс обучения . . . 20
3.9 База данных фильтрации . . . 21
3.9.1 Статические записи . . . 21
3.9.2 Динамические записи . . . 21
3.9.3 Постоянная база данных . . . 22

3.9.4 Модель операций . . . 22
3.10 Протокольный логический объект моста . . . 22
3.11 Диспетчер моста . . . 22
3.12 Адресация . . . 23
3.12.1 Оконечные станции . . . 23
3.12.2 Порты моста . . . 23
3.12.3 Протокольные логические объекты моста . . . 23
3.12.4 Логические объекты диспетчера моста . . . 24
3.12.5 Уникальный идентификатор моста . . . 25
3.12.6 Зарезервированные адреса . . . 25
4 Алгоритм и протокол покрывающего дерева . . . 25
4.1 Требования к алгоритму . . . 25
4.2 Требования к мостам УДС . . . 26
4.3 Общие сведения . . . 26
4.3.1 Активная топология и ее вычисление . . . 26
4.3.2 Распространение информации о топологии . . . 29
4.3.3 Реконфигурация . . . 29
4.3.4 Изменение состояния порта . . . 30
4.3.5 Информирование об изменении топологии . . . 31
4.4 Состояния порта . . . 32
4.4.1 Блокирование . . . 32
4.4.2 Прослушивание . . . 32
4.4.3 Обучение . . . 33
4.4.4 Продвижение . . . 33
4.4.5 Неактивное состояние . . . 33
4.5 Параметры и тайм-ауты протокола . . . 34
4.5.1 Параметры ПБДМ «конфигурация» . . . 34
4.5.2 Параметры ПБДМ «уведомление об изменении топологии» . . . 35
4.5.3 Параметры моста . . . 35
4.5.4 Тайм-ауты моста . . . 37
4.5.5. Параметры порта . . . 37
4.5.6 Тайм-ауты порта . . . 38
4.6 Элементы процедур . . . 39
4.6.1 Передача ПБДМ «конфигурация» . . . 39
4.6.2 Регистрация информации о конфигурации . . . 39
4.6.3 Регистрация значений тайм-аутов, содержащихся в ПБДМ «конфигурация» . . . 40
4.6.4 Генерация ПБДМ «конфигурация» . . . 40
4.6.5 Ответ на ПБДМ «конфигурация» . . . 41
4.6.6 Передача ПБДМ «уведомление об изменении топологии» . . . 41
4.6.7 Изменение конфигурации . . . 41
4.6.8 Выбор корневого моста . . . 41
4.6.9 Выбор назначенного порта . . . 42
4.6.10 Становление назначенного порта . . . 43
4.6.11 Выбор состояния порта . . . 43
4.6.12 Выполнение продвижения . . . 44
4.6.13 Выполнение блокирования . . . 44
4.6.14 Обнаружение изменения топологии . . . 44
4.6.15 Уведомление об изменении топологии подтверждено . . . 44
4.6.16 Подтверждение изменения топологии . . . 45
4.7 Операции протокола . . . 45
4.7.1 Прием ПБДМ «конфигурация» . . . 45
4.7.2 Прием ПБДМ «уведомление об изменении топологии» . . . 46
4.7.3 Истечение тайм-аута заявки . . . 46
4.7.4 Истечение тайм-аута времени существования сообщения . . . 46
4.7.5 Истечение тайм-аута задержки продвижения . . . 46

4.7.6 Истечение тайм-аута уведомления об изменении топологии . . . 46
4.7.7 Истечение тайм-аута изменения топологии . . . 46
4.7.8 Истечение тайм-аута удержания . . . 46
4.8 Диспетчер логического объекта мостового протокола . . . 46
4.8.1 Инициализация . . . 47
4.8.2 Активизация порта . . . 47
4.8.3 Деактивизация порта . . . 48
4.8.4 Установка приоритета моста . . . 48
4.8.5 Установка приоритета порта . . . 48
4.8.6 Установка стоимости маршрута . . . 48
4.9 Модель процедур . . . 49
4.9.1 Общие сведения . . . 49
4.10 Рабочие характеристики . . . 66
4.10.1 Требования . . . 66
4.10.2 Значения параметров . . . 67
5 Кодирование ПБДМ . . . 67
5.1 Структура . . . 68
5.1.1 Передача и представление октетов . . . 68
5.1.2 Компоненты . . . 68
5.2 Кодирование типов параметров . . . 68
5.2.1 Кодирование идентификатора протокола . . . 68
5.2.2 Кодирование идентификатора версии протокола . . . 68
5.2.3 Кодирование типов ПБДМ . . . 68
5.2.4 Кодирование флагов . . . 69
5.2.5 Кодирование идентификатора моста . . . 69
5.2.6 Кодирование стоимости маршрута к корневому мосту . . . 69
5.2.7 Кодирование идентификатора порта . . . 69
5.2.8 Кодирование значений тайм-аутов . . . 69
5.3 Форматы и параметры ПБДМ . . . 69
5.3.1 ПБДМ «конфигурация» . . . 69
5.3.2 ПБДМ «уведомление об изменении топологии» . . . 70
5.3.3 Действительность принимаемых ПБДМ . . . 70
6 Диспетчер моста . . . 71
6.1 Функции диспетчера . . . 71
6.1.1 Диспетчер конфигурации . . . 71
6.1.2 Диспетчер неисправностей . . . 71
6.1.3 Диспетчер рабочих характеристик . . . 71
6.1.4 Диспетчер защиты . . . 71
6.1.5 Диспетчер административного управления учетом . . . 72
6.2 Администрируемые объекты . . . 72
6.3 Типы данных . . . 72
6.4 Логический объект диспетчера моста . . . 72
6.4.1 Конфигуратор моста . . . 72
6.4.2 Конфигурация порта . . . 74
6.5 Логические объекты управления доступом к среде . . . 74
6.6 Процесс продвижения . . . 74
6.6.1 Счетчики порта . . . 74
6.6.2 Приоритет передачи . . . 75
6.7 База данных фильтрации . . . 75
6.7.1 Общие положения . . . 75
6.7.2 Статическая запись . . . 76
6.7.3 Динамическая запись . . . 76
6.7.4 Постоянная база данных фильтрации . . . 76
6.7.5 Основные операции базы данных фильтрации . . . 77
6.8 Логический объект протокола моста . . . 78

6.8.1 Логический объект протокола моста . . . 78
6.8.2 Порт моста . . . 79
7 Протокол диспетчера . . . 80
7.1 Определение операций . . . 80
7.1.1 Определение операции «получение» . . . 80
7.1.2 Определение операции «установка» . . . 80
7.1.3 Определение операции «действие» . . . 80
7.2 Операции . . . 81
7.3 Кодирование . . . 81
8 Рабочие характеристики моста . . . 89
8.1 Гарантированная скорость фильтрации порта . . . 89
8.2 Гарантированная скорость маршрутизации моста . . . 89
Приложение А Форма заявки о соответствии реализации протоколу . . . 90
А.1 Введение . . . 90
А.2 Сокращения и специальные символы: символы факультативного статуса . . . 90
А.3 Инструкции по заполнению копии формы ЗСРП . . . 90
А.4 Идентификация требований . . . 90
Приложение В Вычисление параметров покрывающего дерева . . . 101
В.1 Общие положения . . . 101
В.2 Сокращения и специальные символы . . . 101
В.3 Вычисление . . . 102
В.3.1 Время существования, диаметр и транзитная задержка . . . 102
В.3.2 Передача ПБДМ . . . 102
В.3.3 Точность времени существования сообщения . . . 102
В.3.4 Время заявки . . . 103
В.3.5 Распространение мостового протокольного сообщения . . . 103
В.3.6 Время удержания . . . 103
В.3.7 Максимальное время существования . . . 103
В.3.8 Задержка продвижения . . . 104
В.4 Выбор диапазонов параметров . . . 104
В.4.1 Абсолютные максимальные значения . . . 104
В.4.2 Время удержания . . . 105
В.4.3 Диапазон времени заявки . . . 105
В.4.4 Наибольшие требуемые значения максимального времени существования и задержки продвижения . . . 105
В.4.5 Наименьшие значения максимального времени существования и задержки продвижения . . . 105
В.4.6 Взаимоотношения между максимальным временем существования и задержкой продвижения . . . 105
Приложение С Операции моста по прозрачной маршрутизации со стороны отправителя . . . 106
С.1 Общие положения . . . 106
С.1.1 Назначение . . . 106
С.1.2 Определения . . . 106
С.1.3 Ссылки . . . 107
С.1.4 Соответствие . . . 108
С.2 Обеспечение услуг УДС . . . 108
С.2.1 Общие требования . . . 108
С.2.2 Сохранение услуг УДС . . . 108
С.2.3 Обеспечение качества услуг . . . 108
С.2.4 Внутренние услуги подуровня . . . 109
С.2.5 Обеспечение внутренних услуг подуровня . . . 110
С.3 Принципы операций . . . 111
С.3.1 Операция моста при маршрутизации со стороны отправителя . . . 111
С.3.2 Архитектура моста . . . 111
С.3.3 Операции моста . . . 112

C.3.4 Информация состояния порта . . . 117
C.3.5 Прием кадров . . . 117
C.3.6 Передача кадров . . . 117
C.3.7 Продвижение кадров . . . 117
C.3.8 Адресация . . . 123
C.4 Диспетчер моста . . . 123
C.4.1 Логический объект диспетчера моста . . . 123
C.4.2 Процесс продвижения . . . 124
C.4.3 Логический объект диспетчера моста ПМСО . . . 124
C.4.4 База данных пары портов моста ПМСО . . . 125
C.5 Протокол диспетчера . . . 126
C.5.1 Операции . . . 126
C.5.2 Кодирование . . . 127
Приложение D Форма ЗСРП для операций моста при прозрачной маршрутизации со стороны отправителя . . . 130
D.1 Введение . . . 130
D.2 Сокращения и специальные символы. Символы факультативного статуса . . . 130
D.3 Инструкции по заполнению копии формы ЗСРП . . . 130
D.4 Идентификация требований . . . 131
D.5 Соответствие стандартам по УДС и УЛЗ . . . 131
D.6 Ретрансляция и фильтрация кадров . . . 131
D.7 Номера мостов и ИД ЛВС . . . 132
D.8 Диспетчер моста . . . 132
Приложение E Стандарты IEEE и ANSI . . . 133

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационная технология

ПЕРЕДАЧА ДАННЫХ И ОБМЕН ИНФОРМАЦИЕЙ МЕЖДУ СИСТЕМАМИ

ЛОКАЛЬНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СЕТИ
МОСТЫ НА ПОДУРОВНЕ УПРАВЛЕНИЯ ДОСТУПОМ К СРЕДЕ

Information technology. Telecommunications and information exchange between systems. Local area networks. Media access control (MAC) bridges

Дата введения 2000—07—01

1 Общие положения

Локальные вычислительные сети (ЛВС) любых типов могут быть объединены вместе с помощью мостов на подуровне управления доступом к среде (мостов УДС). Каждая отдельная ЛВС имеет свой собственный независимый подуровень УДС. Создание мостовой ЛВС позволяет взаимодействовать станциям, подключенным к отдельным ЛВС, так, как если бы они были подключены к одной ЛВС. Мосты УДС выполняют операции ниже границы услуг УДС, и они прозрачны для протоколов, функционирующих выше этой границы на подуровне управления логическим звеном (УЛЗ) по ГОСТ 28907 и на сетевом уровне по ГОСТ Р ИСО/МЭК 7498-1. Мостовая ЛВС может обеспечивать:

1) взаимосвязь станций, подключенных к ЛВС различных типов;

2) эффективное увеличение физической протяженности, числа допустимых подключений и суммарной производительности ЛВС;

3) разделение физической ЛВС по административным или эксплуатационным соображениям.

1.1 **Область применения**

Для совместимого взаимодействия оборудования обработки данных, которое использует услуги подуровня УДС, предоставляемые взаимосвязанными стандартными ЛВС с различными или одинаковыми методами управления доступом к среде, настоящий стандарт определяет общий метод работы мостов УДС. С этой целью стандарт:

1) определяет место мостовой функции в рамках архитектурного описания подуровня УДС;

2) определяет принципы работы моста УДС с точки зрения обеспечения и сохранения услуг УДС и поддержки качества этих услуг;

3) определяет внутренние услуги подуровня УДС, предоставляемые отдельными ЛВС для функций, не зависимых от метода доступа к среде, которые обеспечивают ретрансляцию кадров мостом;

4) идентифицирует функции, которые должен выполнять мост, и обеспечивает архитектурную модель внутренних операций моста в терминах процессов и логических объектов, обеспечивающих эти функции;

5) устанавливает требования к протоколу взаимодействия мостов мостовых ЛВС для организации сети и определяет метод распределенного вычисления активной топологии покрывающего дерева;

6) определяет кодирование протокольных блоков данных моста (ПБДМ);

7) устанавливает требования к административному управлению мостом в мостовых ЛВС, идентифицируя администрируемые объекты и определяя операции административного управления;

8) определяет способ доступа удаленного диспетчера к операциям административного управления, используя описание протокола и архитектуры, приведенное в TEEE Std 802.1B;

Издание официальное

9) устанавливает требования к соответствию, рекомендуемые значения по умолчанию и допустимые диапазоны рабочих параметров моста;

10) устанавливает требования к оборудованию, претендующему на соответствие настоящему стандарту;

11) определяет критерии использования специфичных для УДС методов работы моста.

Настоящий стандарт определяет операции мостов УДС, которые подключены непосредственно к ЛВС, как определено в соответствующих стандартных или практически реализованных технологиях УДС. Несмотря на то, что волоконно-оптический распределенный интерфейс передачи данных (ВОРИПД) в сети кольцевого типа с маркерным доступом, определенный в ГОСТ Р 50452, не входит в семейство стандартных ЛВС, он может рассматриваться как эквивалент УДС. Если не оговорено иное, то ссылки в настоящем стандарте на стандартные ЛВС подразумевают и ВОРИПД.

Определение удаленных мостов ЛВС, использующих для передачи кадров между мостами среду глобальной вычислительной сети (ГВС), не входит в предмет рассмотрения настоящего стандарта.

1.2 **Нормативные ссылки**

Настоящий стандарт содержит ссылки на следующие стандарты. Ссылки на стандарты IEEE или ANSI являются справочными и приведены в приложении Е.

ГОСТ 34.913.3—91 (ИСО/МЭК 8802-3—92) Информационная технология. Локальные вычислительные сети. Метод случайного доступа к шине (КДОН/ОК) и спецификация физического уровня

ГОСТ 34.913.4—91 (ИСО/МЭК 8802-4—90) Информационная технология. Локальные вычислительные сети. Метод маркерного доступа к шине и спецификация физического уровня

ГОСТ 34.936—91 (ИСО 10039—91) Информационная технология. Локальные вычислительные сети. Определение услуг уровня управления доступом к среде

ГОСТ 28907—91 (ИСО 8802-2—89) Системы обработки информации. Локальные вычислительные сети. Протокол и услуги уровня управления логическим звеном данных

ГОСТ Р 50452—92 (ИСО 9314-1—89) Системы обработки информации. Волоконно-оптический распределенный интерфейс передачи данных (ВОРИПД). Часть 1. Протокол физического уровня кольца с маркерным доступом.

ГОСТ Р ИСО/МЭК 8802-5—99 Информационная технология. Локальные и региональные вычислительные сети. Часть 5. Метод доступа к кольцу с передачей маркера и спецификация физического уровня

ГОСТ Р ИСО/МЭК 8824—93 Информационная технология. Взаимосвязь открытых систем. Спецификация абстрактно-синтаксической нотации версии один (АСН.1)

ГОСТ Р ИСО/МЭК 8825—93 Информационная технология. Взаимосвязь открытых систем. Описание базовых правил кодирования для абстрактно-синтаксической нотации версии один (АСН.1)

ГОСТ Р ИСО/МЭК 9595—99 Информационная технология. Взаимосвязь открытых систем. Определение общих услуг административного управления

ИСО 6937-2—83* Информационная технология. Набор кодированных знаков для передачи текста. Графические знаки латинского и нелатинского алфавита

ИСО 9314-2—89* Системы обработки информации. Волоконно-оптический распределенный интерфейс передачи данных (ВОРИПД). Часть 2. Уровень управления доступом к среде (УДС) кольца с маркерным доступом ВОРИПД

ИСО/МЭК 9596—90* Системы обработки информации. Взаимосвязь открытых систем. Спецификация протокола общего административного управления

ИСО/МЭК 10178—92* Информационная технология. Передача данных и обмен информацией между системами. Структура и кодирование адресов подуровня управления логическим звеном в локальных вычислительных сетях.

1.3 **Определения**

Для настоящего стандарта специфично следующее определение:

мостовая локальная вычислительная сеть: Цепочка отдельных ЛВС, взаимосвязанных мостами УДС.

1.4 **Сокращения**

Для настоящего стандарта специфично следующее сокращение:

ПБДМ — протокольный блок данных моста.

---

* Международные стандарты ИСО/МЭК — во ВНИИКИ Госстандарта России.

1.5 **Соответствие**

Мост УДС, претендующий на соответствие настоящему стандарту:

1) должен удовлетворять:

a) соответствующим стандартным или реализованным технологиям УДС, таким как ГОСТ 34.913.3, ГОСТ 34.913.4, ГОСТ Р ИСО/МЭК 8802-5 или ИСО 9314-2, как изложено в 2.5, и

b) ГОСТ 28907 для реализации УЛЗ класса I или II при обеспечении операций типа 1 согласно требованиям подразделов 3.2, 3.3 и 3.12;

2) должен осуществлять ретрансляцию и фильтрацию кадров согласно 3.1, 3.5—3.7;

3) должен либо:

a) отфильтровывать кадры с одинаковыми адресами отправителя и получателя, либо

b) не отфильтровывать кадры с одинаковыми адресами отправителя и получателя, как описано 3.7;

4) может обеспечивать возможность управления преобразованием приоритетов продвигаемых кадров согласно 3.7;

5) должен сохранять информацию, требуемую для принятия решения о необходимости фильтрации кадров, согласно 3.1, 3.8 и 3.9;

6) должен использовать установленные значения следующих параметров базы данных фильтрации (см. 3.9):

a) размер базы данных фильтрации — максимальное число возможных содержащихся в ней записей,

b) размер постоянной базы данных — максимальное число возможных содержащихся в ней записей;

7) может обеспечивать возможности чтения и обновления базы данных фильтрации и постоянной базы данных согласно 3.9;

8) может обеспечивать возможности установления времени существования сообщения согласно 3.9. Мост, который обеспечивает эту возможность, должен реализовывать полный диапазон значений, определенных в таблице 3—3;

9) должен обеспечивать возможности адресации, определенные в 3.12;

10) может реализовывать факультативные возможности адресации административного управления моста, определенные в 3.12.4, для логической увязки адреса моста с портом моста согласно 3.12.5 и для реконфигурации групповых адресов в постоянной базе данных согласно 3.12.6;

11) должен обеспечивать:

a) средства назначения групповых адресов УДС с целью идентификации логического объекта протокола моста, если не используется 48-битовая универсально администрируемая адресация,

b) средства назначения уникальных адресов в пределах мостовой ЛВС с целью уникальной идентификации моста в случае использования 16-битовой локально администрируемой адресации согласно 4.5 и

c) индивидуальный идентификатор порта для каждого порта моста согласно 4.5 и в соответствии с требованиями подраздела 4.2 к алгоритму и протоколу покрывающего дерева;

12) должен реализовывать алгоритм и протокол покрывающего дерева, описанный в разделе 4, согласно 4.7;

13) не должен превышать значений, приведенных в 4.10.2, для следующих параметров:

a) максимальная мостовая транзитная задержка,

b) максимальная переоценка приращения времени существования сообщения,

c) максимальная задержка передаваемого ПБДМ;

14) должен использовать значения, приведенные в таблице 4—3 для параметра «время захвата»;

15) может обеспечивать присвоение значений следующим параметрам с учетом конфигурации активной топологии покрывающего дерева:

a) приоритет моста,

b) приоритет порта,

c) стоимость пути для каждого порта. Мост, который обеспечивает эту возможность, должен реализовывать диапазон значений, определенных в 4.10.2 и таблицах 4—4 и 4—5;

16) может обеспечивать возможность установления значений следующих параметров алгоритма и протокола покрывающего дерева:

a) максимальное мостовое время существования сообщения,

b) мостовое время заявки,

c) мостовая задержка продвижения.

Мост, который обеспечивает эту возможность, должен реализовывать диапазон значений, определенных в 4.10.2 и таблице 4—3;

17) должен кодировать передаваемые ПБДМ и объявлять действительными принимаемые ПБДМ, как определено в разделе 5;

18) может обеспечивать административное управление мостом.

Мосты, претендующие на обеспечение административного управления, должны поддерживать все объекты и операции административного управления, определенные в разделе 6.

19) может обеспечивать удаленное административное управление, стандартизованное IEEE Std 802.1B. Мосты, претендующие на обеспечение удаленного административного управления согласно IEEE Std 802.1B, должны:

a) соответствовать стандарту IEEE Std 802.1B,

b) обеспечивать удаленные операции административного управления путем использования операций сетевого административного управления и кодирования, определенных в разделе 7;

20) должен определять гарантированный коэффициент фильтрации порта для каждого порта, гарантированный коэффициент ретрансляции моста и устанавливать соотношение между временными интервалами T и T, определенными в разделе 8. Операции моста в рамках специфицированных параметров не должны нарушать ни одного из требований, предъявляемых настоящим стандартом.

Поставщик реализации, претендующей на соответствие настоящему стандарту, должен заполнить копию формы ЗСРП, приведенной в приложении А, и представить информацию, необходимую для идентификации как поставщика, так и реализации.

1.6 **Рекомендации**

1.6.1 Административное управление

Настоятельно рекомендуется обеспечение объектов и операций административного управления, определенных в разделе 6.

1.7 **Специфичные для УДС мостовые методы**

Могут существовать специфичные для подуровня УДС методы организации мостов. Использование специфичного для УДС мостового метода и метода, определенного настоящим стандартом, на одной и той же ЛВС должно:

1) не препятствовать взаимосвязи между станциями мостовой ЛВС;

2) сохранять услуги и УДС;

3) сохранять характеристики каждого мостового метода в пределах своего региона;

4) обеспечивать возможность одновременного сосуществования обоих мостовых технологий в ЛВС без каких-либо неблагоприятных влияний друг на друга.

2 Обеспечение услуг УДС

Мосты УДС взаимоувязывают отдельные ЛВС, образуя ЛВС путем ретрансляции кадров между подуровнями УДС отдельных мостовых ЛВС. Место мостовой функции в подуровне УДС показано на рисунке 2.1.

Услуги УДС предоставляются их пользователю на оконечной станции и поддерживаются функцией ретрансляции моста. Это позволяет исключать требования, специфичные для отдельных методов доступа к среде.

В данном разделе рассматриваются вопросы предоставления услуг УДС оконечным станциям, поддержка услуг УДС мостами, сохранение услуг УДС, обеспечиваемых мостовой ЛВС, поддержание качества услуг в мостовой ЛВС и обеспечение внутренних услуг подуровня, предоставляемых для самих мостов с целью подключения отдельных ЛВС.

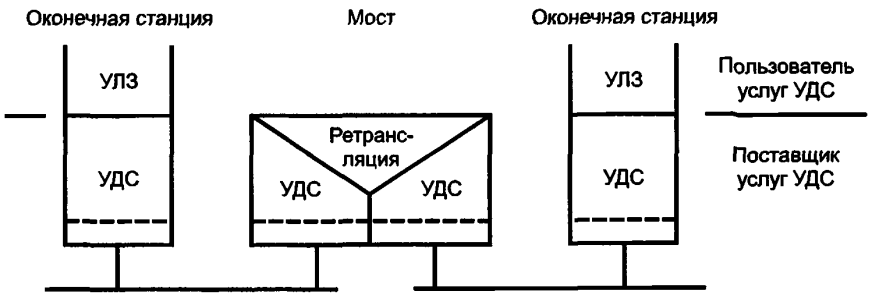


Рисунок 2.1 — Внутренняя организация подуровня УДС

2.1 **Общие положения**

Услуги УДС, предоставляемые оконечным станциям мостовой ЛВС, обеспечиваются мостами УДС этой мостовой ЛВС. Мосты выдают примитивы УД-БЛОК-ДАННЫХ запрос и соответствующие примитивы УД-БЛОК-ДАННЫХ индикация, переносящие данные пользователя, без выдачи на них подтверждений. Услуги подтверждения оконечными станциями, взаимодействующими через мост, не обеспечиваются.

Стиль выполнения операций моста увеличивает доступность услуг УДС для оконечных станций и облегчает эксплуатацию мостовой ЛВС.

Поэтому желательно наделить мосты способностью организовывать мостовую ЛВС таким образом, чтобы:

1) можно было обеспечить резервные пути между оконечными станциями, позволяющими мостовой ЛВС продолжать предоставление своих услуг в случае выхода из строя какой-либо ее компоненты (моста или ЛВС);

2) можно было прогнозировать и автоматически изменять пути между оконечными станциями при заданной доступности к компонентам мостовой ЛВС.

2.2 **Постоянство услуг УДС**

Услуги подуровня УДС, предоставляемые мостовой ЛВС, которая состоит из отдельных соединенных мостами ЛВС, такие же, как и услуги отдельной ЛВС. Благодаря этой особенности:

1) мост непосредственно не адресуется взаимодейсвующими оконечными станциями, за исключением случаев, когда оконечная станция используется для целей административного управления: кадры, передаваемые между оконечными станциями переносят в своем поле «адрес получателя» адрес УДС равноправной оконечной станции, а не адрес моста (при его наличии);

2) все адреса УДС должны быть уникальными и назначаться в пределах мостовой ЛВС;

3) адреса УДС оконечной станции не ограничиваются топологией и конфигурацией мостовой ЛВС.

2.3 ***Обеспечение качества услуг***

Качество услуг УДС, обеспечиваемое мостом, не должно быть значительно ниже качества, обеспечиваемого отдельной ЛВС. Подлежат рассмотрению те параметры качества услуг, которые имеют отношение к:

1) доступности услуг;
2) потере кадров;
3) нарушению последовательности кадров;
4) дублированию кадров;
5) транзитной задержке, испытываемой кадром;
6) времени существования кадра;
7) коэффициенту необнаруженных ошибок в кадре;
8) обеспечиваемому максимальному размеру сервисного блока данных;
9) приоритету пользователя и
10) пропускной способности.

2.3.1 Доступность услуг

Подуровень УДС предоставляет услуги УДС оконечным станциям, подключенным к ЛВС или мостовой ЛВС. Доступность услуг измеряется как часть некоторого общего времени, в течение которого предоставляется услуга. Работа моста может увеличить или снизить доступность услуг.

Доступность услуг может быть увеличена путем автоматической реконфигурации мостовой ЛВС с целью обхода неисправной компоненты (например передатчика, кабеля или соединителя) на пути данных. Доступность услуг может быть снижена неисправностью самого моста по причине отклонения услуги или отфильтровывания кадров мостом.

В случае автоматической реконфигурации мост может отклонить услугу и аннулировать кадры (см. 2.3.2) с целью сохранения других аспектов услуг УДС (см. 2.3.3, 2.3.4). Услуги могут быть отклонены оконечной станцией, которая не получает преимуществ от реконфигурации; следовательно, доступность услуг для таких оконечных станций снижается. Мосты могут отфильтровывать кадры для локализации трафика в мостовой ЛВС. Как только оконечная станция изменит свое местоположение, она может оказаться неспособной принимать кадры от других оконечных станций до тех пор, пока не будет обновлена информация о фильтрации, хранимая мостом.

В мосту должны быть предусмотрены соответствующие меры по увеличению доступности услуг, исключению потерь услуг или задержек в предоставлении услуг, за исключением ситуаций, возникающих в результате неисправностей, удаления или подсоединения компонент мостовой ЛВС, или в результате перемены местоположения оконечной станции. Эти ситуации считаются экстраординарными событиями. Таким образом, операции любого дополнительного протокола, необходимые для поддержания качества услуг, ограничиваются конфигурацией мостовой ЛВС и не зависят от конкретных случаев обеспечения услуг.

2.3.2 Потеря кадров

Услуги, предоставляемые подуровнем УЛС, не гарантируют доставку сервисных блоков данных. Кадры, передаваемые станцией-отправителем, с высокой вероятностью поступают без искажений на станцию - получатель. Работа моста вносит минимальную дополнительную потерю кадров.

Кадры, передаваемые станцией - отправителем, могут поступать на станцию - получатель поврежденными в результате:

1) искажения кадра на физическом уровне в процессе передачи или приема;
2) аннулирования кадра мостом в случае:
   a) невозможности передать его в течение некоторого максимального времени, в результате чего требуется удалить кадр, чтобы предотвратить превышение максимального времени существования кадра (см. 2.3.6),
   b) невозможности сохранять кадр вследствие исчерпания емкости внутреннего буфера, например кадры, подлежащие хранению, прибывают со скоростью, превышающей скорость их возможной дальнейшей передачи,
   c) когда размер сервисного блока данных, переносимого кадром, превышает максимальный, поддерживаемый процедурами УДС, реализуемыми в той ЛВС, в которую должен ретранслироваться кадр,
   d) изменений в объединенной топологии мостовой ЛВС, вынуждающих аннулировать кадры в течение ограниченного периода времени, чтобы сохранить другие аспекты качества услуг.

2.3.3 Нарушение последовательности кадров

Услуги, обеспечиваемые подуровнем УДС, не допускают нарушения последовательности передачи кадров с заданным приоритетом пользователя. Сервисный примитив УД-БЛОК-ДАННЫХ индикация, соответствующий примитиву УД-БЛОК-ДАННЫХ запрос с тем же запрошенным приоритетом, должен приниматься в той же последовательности, в которой обрабатывался соответствующий примитив запроса. Работа моста не должна приводить к нарушению последовательности кадров, передаваемых с одним и тем же приоритетом пользователя.

В случае, когда мосты в мостовой ЛВС способны соединить отдельные УДС таким образом, что между любой парой отправитель — получатель может образоваться множество путей, протокол должен обеспечить исключение такой возможности.

2.3.4 Дублирование кадров

Услуги, обеспечиваемые подуровнем УДС, не допускают дублирования кадров. Работа моста не приводит к дублированию кадров данных пользователя.

Возможность дублирования кадров в мостовой ЛВС возникает вследствие возможного дублирования принимаемых кадров при последующих передачах в пределах моста или вследствие возникновения множества путей между оконечными станциями — отправителем и получателем.

Мосты не должны дублировать кадры данных пользователя.

2.3.5 Транзитная задержка

Услуги, предоставляемые подуровнем УДС, вносят транзитную задержку кадра, которая зависит от особенностей физической среды и используемого метода управления доступом к среде. Транзитная задержка кадра представляет собой время, прошедшее между выдачей примитива УД-БЛОК-ДАННЫХ запрос и выдачей соответствующего примитива УД-БЛОК-ДАННЫХ индикация. Значение истекшего времени вычисляется только для успешно переданных сервисных блоков данных.

Поскольку услуги УДС обеспечиваются в абстрактном интерфейсе оконечной станции, то не представляется возможным точно определить общую транзитную задержку кадра. Однако можно измерить те компоненты задержки, которые связаны с доступом к среде или с передачей и приемом кадра; может быть также измерена транзитная задержка, вносимая промежуточной системой, в данном случае мостом.

Минимальная дополнительная транзитная задержка, вносимая мостом, представляет собой время, затрачиваемое на прием кадра, плюс время, затрачиваемое на доступ к физической среде, в которую предполагается ретранслировать кадр. Заметим, что кадр должен быть принят полностью прежде, чем он будет ретранслирован, поскольку должна быть вычислена контрольная последовательность кадра (КПК) и в случае ошибки кадр должен быть аннулирован.

2.3.6 Время существования кадра

Услуги, обеспечиваемые подуровнем УДС, гарантируют наличие для каждого отдельного сеанса взаимосвязи верхней границы транзитной задержки. Это максимальное время существования кадра устанавливается для гарантии правильной работы протоколов вышерасположенных уровней. Дополнительная транзитная задержка, вносимая мостом, рассмотрена выше.

Для того, чтобы выдержать максимальное время существования кадра, мосту может потребоваться аннулировать некоторые другие кадры. Поскольку информация, предоставляемая мосту подуровнем УДС, не содержит значения транзитной задержки, которой подвергается любой отдельный кадр, мосты должны аннулировать кадры, чтобы не превысить максимальную задержку в каждом мосту.

Значение максимальной мостовой транзитной задержки основано как на максимальной задержке, налагаемой всеми мостами мостовой ЛВС, так и на требуемом максимальном времени существования кадра. Рекомендуемое и абсолютное максимальное значения транзитной задержки указаны в таблице 4-2.

2.3.7 Коэффициент необнаруженных ошибок кадра

Услуги, предоставляемые подуровнем УДС, вносят в передаваемые кадры очень незначительный коэффициент необнаруженных ошибок. Защита от необнаруженных ошибок осуществляется путем использования КПК, которая присоединяется к кадру подуровнем УДС станции - отправителя перед передачей и проверяется станцией - получателем при приеме.

Вычисленная КПК для данного сервисного блока данных зависит от используемого метода управления доступом к среде. Поэтому необходимо повторно вычислять КПК внутри моста, обеспечивающего функцию ретрансляции между стандартными УДС различных типов. Это вносит вероятность дополнительных необнаруженных ошибок, возникающих в процессе работы моста. Для кадров, ретранслируемых между ЛВС одного и того же типа УДС, мосты не должны превышать коэффициент необнаруженных ошибок кадра, выше того, который был бы достигнут при сохранении КПК неизменной.

2.3.8 Максимальный размер сервисного блока данных

Максимальный размер сервисного блока данных, который может быть обеспечен ЛВС, меняется в зависимости от метода управления доступом к среде и его соответствующих параметров (быстродействия и др.). Он может быть ограничен владельцем ЛВС.

Этот размер, обеспечиваемый мостом между двумя ЛВС, меньше размера, обеспечиваемого отдельными ЛВС. Мост не должен пытаться выполнять ретрансляцию кадра в ЛВС, которая не обеспечивает размер сервисного блока данных, переносимого данным кадром.

2.3.9 Приоритет

К услугам, обеспечиваемым подуровнем УДС, относится приоритет пользователя, рассматриваемый как параметр качества услуг. Примитив УД-БЛОК-ДАННЫХ запрос с более высоким приоритетом может преобладать над остальными примитивами запроса, выдаваемыми на той же станции, или в других станциях, подключенных к той же ЛВС, и может обгонять предшествующие примитивы УД-БЛОК-ДАННЫХ индикация.

Подуровень УДС преобразует запрашиваемые приоритеты пользователя в приоритеты, обеспечиваемые отдельными методами управления доступа к среде. Запрошенный приоритет пользователя может быть передан на станцию-получатель.

Поскольку мост не должен изменять последовательность кадров, создаваемых из примитивов УД-БЛОК-ДАННЫХ запрос одинакового приоритета, то преобразование приоритетов должно быть статическим.

2.3.10 Пропускная способность

Общая пропускная способность, обеспечиваемая мостовой ЛВС, может быть значительно выше, чем обеспечиваемая эквивалентной отдельной ЛВС. Мосты могут локализовать трафик внутри мостовой ЛВС путем фильтрации кадров.

Пропускная способность между оконечными станциями отдельных ЛВС, взаимодействующих через мост, может быть снижена путем аннулирования кадров в мосту из-за невозможности передавать их в течение длительного периода с требуемой скоростью в ЛВС, расположенную на пути к получателю.

2.4 **Внутренние услуги подуровня, обеспечиваемые внутри моста УДС**

Внутренние услуги подуровня, предоставляемые объектами УДС логическому объекту ретрансляции УДС внутри моста, — это те услуги, которые предоставляются отдельными независимыми УДС. Мост наблюдает за соответствующими процедурами УДС и протоколом каждой ЛВС, с которой он соединен. Никакие кадры управления (известные также как кадры УДС), руководящие работой отдельной ЛВС, не выдаются из ЛВС — источника этих кадров.

Внутренние услуги подуровня образуются из услуг УДС, стандартизованных ГОСТ 34.936, путем добавления данной спецификации с элементами, необходимыми для выполнения функций ретрансляции. В подключенной оконечной станции эти дополнительные элементы могут рассматриваться либо ниже границы услуг УДС и относиться только к операциям поставщика услуг, либо как локальные вопросы, не имеющие отношения к равноправным взаимоотношениям услуг УДС. К перечню параметров, относящихся к примитивам УД-БЛОК-ДАННЫХ запрос и УД-БЛОК-ДАННЫХ индикация, определенным в ГОСТ 34.936, добавляются три параметра: «тип кадра», «действие УДС» и «контрольная последовательность кадра». Определение внутренних услуг подуровня не добавляет никаких новых сервисных примитивов к примитивам, определенным в ГОСТ 34.936.

Внутренние услуги подуровня не содержат специфичных для УДС средств и процедур, работа которых ограничена в рамках отдельных ЛВС. К примитивам услуги БЛОК-ДАННЫХ, описывающих данную услугу, относятся следующие:

УД-БЛОК-ДАННЫХ индикация (
тип кадра,
действие удс,
адрес получателя,
адрес отправителя,
сервисный блок данных удс,
приоритет пользователя,
контрольная последовательность кадра
)

Каждый привлекаемый примитив индикации соответствует приему кадра УДС из отдельной ЛВС.

Параметр «тип кадра» указывает класс кадра. Этот параметр принимает одно из следующих значений: кадр данных пользователя, специфичный кадр удс или зарезервированный кадр.

Параметр «действие удс» указывает действие, запрошенное от логического объекта УДС, получившего примитив индикации. Если параметр «тип кадра» имеет значение «кадр данных пользователя», то параметр «действие удс» принимает одно из следующих значений: «запрос с ответом», «запрос без ответа» или «ответ». Для значений «специфичный кадр удс» и «зарезервированный кадр» этот параметр не применяется.

Параметр «адрес получателя» представляет собой либо адрес отдельного логического объекта УДС, либо адрес группы логических объектов УДС.

Параметр «адрес отправителя» — это индивидуальный адрес логического объекта УДС - отправителя.

Параметр «сервисный блок данных удс» содержит сервисный блок данных.

Параметр «приоритет пользователя» представляет собой приоритет, запрошенный инициирующим пользователем услуг. Значение этого параметра, если оно специфицировано, находится в диапазоне от 0 (наименьшее) до 7 (наибольшее).

Параметр «контрольная последовательность кадра» явно обеспечивается как параметр примитива таким образом, чтобы его можно было использовать без перерасчетов в качестве параметра соответствующего примитива запроса.

Идентификация ЛВС, из которой получены конкретные кадры, является локальным вопросом и не выражается в параметрах сервисных примитивов.

УДБЛОКДАННЫХ запрос (
тип кадра,
действие удс,
адрес получателя,
адрес отправителя,
сервисный блок данных удс,
приоритет пользователя,
приоритет доступа
контрольная последовательность кадра
)

Примитив запроса привлекается для передачи кадра в отдельную ЛВС.

Параметр «тип кадра» указывает класс кадра.

Параметр «действие удс» указывает действие, требуемое от логического объекта УДС.

Параметр «адрес получателя» представляет собой либо адрес отдельного логического объекта УДС, либо группы логических объектов УДС.

Параметр «адрес отправителя» содержит индивидуальный адрес логического объекта УДС - отправителя.

Параметр «сервисный блок данных удс» содержит сервисный блок данных.

Параметр «приоритет пользователя» представляет собой приоритет, запрошенный инициирующим пользователем услуг. Значение этого параметра, если оно специфицировано, находится в диапазоне от 0 (наименьшее) до 7 (наибольшее).

Параметр «приоритет доступа» представляет собой приоритет, используемый локальным поставщиком услуг для передачи запроса. Он может использоваться для определения приоритета подлежащих передаче кадров, поставленных в очередь локальным логическим объектом УДС как локально, так и среди других станций, подключенных к одной и той же отдельной ЛВС (если такую возможность допускает метод доступа к среде). Значение этого параметра, если оно специфицировано, находится в диапазоне от 0 (наименьшее) до 7 (наибольшее).

Параметр «контрольная последовательность кадра» явно предусматривается как параметр примитива, с тем чтобы он мог быть использован без пересчета.

Идентификация ЛВС, в которую передается кадр, является локальным вопросом и не выражается в параметрах сервисных примитивов.

2.5 **Поддержка внутренних услуг подуровня специфичными для УДС процедурами**

В этом подразделе определяется преобразование внутренних услуг подуровня УДС в протокол и процедуры УДС каждого отдельного стандартного типа УДС и кодирование параметров услуг в кадрах УДС. Преобразование осуществляется путем ссылок на стандарты по ЛВС, которые определяют отдельные методы доступа к среде. В вопросах этих преобразований обращают внимание на все специальные виды ответственности мостов, подключенных к ЛВС данного типа УДС.

2.5.1 Обеспечение услуг ЛВС КДОН/ОК по ГОСТ 34.913.3

Коллективный метод доступа с опознаванием несущей и обнаружением конфликтов (КДОН/ОК) определен в ГОСТ 34.913.3. В разделе 3 настоящего стандарта определена структура кадра УДС, а в разделе 4 — сам метод управления доступом к среде.

При приеме примитива УД-БЛОК-ДАННЫХ запрос локальный логический объект УДС выполняет компоновку передаваемых данных и формирует кадр, используя предоставляемые параметры в соответствии с изложенным ниже. Перед вручением кадра передающей компоненте «административное

управление доступом к среде» подуровня УДС для его передачи он присоединяет преамбулу и начальный ограничитель кадра (см. 4.2.3 ГОСТ 34.913.3).

При получении кадра УДС приемной компонентой «административное управление доступом к среде» он направляется компоненте «раскомпоновка принимаемых данных», которая проверяет действительность КПК и расформировывает кадр, как определено ниже, на параметры, которые поставляются с примитивом УД-БЛОК-ДАННЫХ индикация (см. 4.2.4 ГОСТ 34.913.3).

Параметр «тип кадра» принимает единственное значение «кадр данных пользователя» и явно не кодируется в кадре УДС.

Параметр «действие удс» принимает единственное значение «запрос без ответа» и явно не кодируется в кадре УДС.

Параметр «адрес получателя» преобразуется в поле «адрес получателя» кадра УДС (см. 3.2.3 ГОСТ 34.913.3).

Параметр «адрес отправителя» преобразуется в поле «адрес отправителя» кадра УДС (см. 3.2.3 ГОСТ 34.913.3).

Число октетов в параметре «сервисный блок данных удс» преобразуется в поле «длина» кадра УДС (см. 3.2.6 ГОСТ 34.913.3), а октеты данных кодируются в поле «данные» (см. 3.2.7 ГОСТ 34.913.3).

Параметр «приоритет пользователя» не кодируется в кадре УДС и принимает неопределенное значение в соответствующем примитиве УД-БЛОК-ДАННЫХ индикация.

Параметр «контрольная последовательность кадра» преобразуется в поле КПК кадра УДС (см. 3.2.8 ГОСТ 34.913.3). КПК вычисляется как функция содержимого полей «адрес получателя», «адрес отправителя», «длина», «данные» и «заполнитель». Следовательно, этот параметр переносит также значение поля «заполнитель», необходимый для удовлетворения требований минимального размера кадра (см. 3.2.73 ГОСТ 34.913.3). Если примитив УД-БЛОК-ДАННЫХ запрос не сопровождается этим параметром, КПК вычисляется в соответствии с 3.2.8 ГОСТ 34.913.3.

Для обеспечения внутренних услуг подуровня УДС методом доступа КДОН/ОК никаких специальных действий, помимо тех, которые определены для использования услуг УДС подуровнем УЛЗ, не требуется

2.5.2 Обеспечение услуг ЛВС по ГОСТ 34.913.4 (маркерный метод доступа к шине)

Маркерный метод доступа к шине определен в ГОСТ 34.913.4. В разделе 4 настоящего стандарта определены форматы кадра, в разделе 5 обсуждаются базовые концепции протокола доступа, а в разделе 7 дано определение спецификации операций УДС ЛВС шинного типа с маркерным доступом.

При получении примитива УД-БЛОК-ДАННЫХ запрос интерфейсный конечный автомат (ИНТ-КА) локального логического объекта УДС ставит в очередь кадр для его передачи конечным автоматом управления доступом (УД-КА) (см. раздел 7 ГОСТ 34.913.4). При передаче кадра его поля устанавливаются с использованием предоставляемых параметров, как определено ниже, и добавляются поля «преамбула», «начальный ограничитель» и «коненый ограничитель» (см. раздел 4 ГОСТ 34.913.4).

При получении приемным конечным автоматом (ПМ-КА) действительного кадра УДС (см. 7.1.5 ГОСТ 34.913.4) генерируется примитив УД-БЛОК-ДАННЫХ индикация с параметрами, полученными из полей кадра, как определено ниже.

Параметр «тип кадра» кодируется в битах FF (битовые позиции 1 и 2) поля «управление кадра» (см. 4.1.3.1 и 4.1.3.2 ГОСТ 34.913.4). Битовая комбинация 01 означает «кадр данных пользователя», битовая комбинация 00 или 10 — специфичный кадр удс», а битовая комбинация 11 — «зарезервированный кадр».

Параметр «действие удс» кодируется битами МММ (битовые позиции 3—5) поля «управление кадра» (см. 4.1.3.2 ГОСТ 34.913.4). Для «кадра данных пользователя» они принимают значение 000 при «запросе без ответа», 001 — при «запросе с ответом» и 010 — при «ответе».

Параметр «адрес получателя» преобразуется в поле «адрес получателя» кадра УДС (см. 4.1.4.1 ГОСТ 34.913.4).

Параметр «адрес отправителя» преобразуется в поле «адрес отправителя» кадра УДС (см. 4.1.4.2 ГОСТ 34.913.4).

Параметр «сервисный блок данных удс» преобразуется в поле «блок данных УДС» (см. 4.1.5 ГОСТ 34.913.4).

Параметр «приоритет пользователя» кодируется битами РРР (битовые позиции 6—8) поля «управление кадра» (см. 4.1.3.2 и 5.1.7 ГОСТ 34.913.4).

Параметр «контрольная последовательность кадра» преобразуется в поле КПК кадра УДС (см. 4.1.6 ГОСТ 34.913.4). КПК вычисляется как функция содержимого полей «управление кадра», «адрес получателя», «адрес отправителя» и «данные». Если примитив УД-БЛОК-ДАННЫХ запрос не сопровождается этим параметром, КПК вычисляется в соответствии с 4.1.6 ГОСТ 34.913.4.

Для обеспечения внутренних услуг подуровня УДС маркерным методом доступа к шине никаких специальных действий, помимо тех, которые определены для использования услуг УДС подуровнем УЛЗ, не требуется

2.5.3 Обеспечение услуг ЛВС по ГОСТ Р ИСО/МЭК 8802-5 (маркерный метод доступа к кольцу)

Маркерный метод доступа к кольцу определен в ГОСТ Р ИСО/МЭК 8802-5. В разделе 3 настоящего стандарта определены форматы и услуги, а в разделе 4 — протокол маркерного метода доступа к кольцу.

При получении примитива УД-БЛОК-ДАННЫХ запрос локальный логический объект УДС формирует кадр, используя поставляемые параметры, как определено ниже, добавляя к данным пользователя поля «управление кадра», «адрес получателя», «адрес отправителя» и КПК, и ставит его в очередь для передачи при получении приемлемого маркера (см. 4.1.1 ГОСТ Р ИСО/МЭК 8802-5). При передаче кадра к нему добавляются поля «начальный ограничитель», «управление доступом», «конечный ограничитель» и «состояние кадра».

При получении действительного кадра УДС (см. 4.1.4 ГОСТ Р ИСО/МЭК 8802-5), который не передавался локальным логическим объектом УДС порта моста, с битом «указатель информации маршрутизации» (занимающим ту же позицию в поле «адрес отправителя», что и бит «групповой адрес» в поле «адрес получателя») в значении ноль генерируется примитив УД-БЛОК-ДАННЫХ индикация с параметрами, полученными из полей кадра, как определено ниже.

Параметр «тип кадра» кодируется битами типа кадра (биты FF) поля «управление кадра» (см. 3.2.3.1 ГОСТ Р ИСО/МЭК 8802-5). Битовая комбинация 01 указывает «кадр данных пользователя», битовая комбинация 00 или 10 — «специфичный кадр удс» и битовая комбинация 11 — «зарезервированный кадр».

Параметр «действие удс» принимает единственное значение «запрос без ответа» и явно не кодируется в кадре УДС.

Параметр «адрес получателя» преобразуется в поле «адрес получателя» кадра УДС (см. 3.2.4.1 ГОСТ Р ИСО/МЭК 8802-5).

Параметр «адрес отправителя» преобразуется в поле «адрес отправителя» кадра УДС (см. 3.2.4.2 ГОСТ Р ИСО/МЭК 8802-5).

Параметр «сервисный блок данных удс» преобразуется в поле «информация» (см. 3.2.6 ГОСТ Р ИСО/МЭК 8802-5).

Параметр «приоритет пользователя», связанный с «кадром данных пользователя», кодируется битами YYY поля «управление кадра» (см. 3.2.3 ГОСТ Р ИСО/МЭК 8802-5).

Параметр «контрольная последовательность кадра» преобразуется в поле КПК кадра УДС (см. 3.2.7 ГОСТ Р ИСО/МЭК 8802-5). КПК вычисляется как функция содержимого полей «управление кадра», «адрес получателя», «адрес отправителя» и «информация». Если примитив УД-БЛОК-ДАННЫХ запрос не сопровождается этим параметром, КПК вычисляется в соответствии с 3.2.7 ГОСТ Р ИСО/МЭК 8802-5.

Биты «распознавание адреса» (А) в поле «состояние кадра» (см. 3.2.9 ГОСТ Р ИСО/МЭК 8802-5) могут быть установлены в значение 1, если сгенерирован примитив УД-БЛОК-ДАННЫХ индикация с параметрами «тип кадра» и «действие удс» в значениях «кадр данных пользователя» и «запрос без ответа» соответственно или если бы такая индикация была сгенерирована при наличии достаточной емкости буферной памяти; в противном случае биты А не должны устанавливаться, за исключением случаев, оговоренных в ГОСТ Р ИСО/МЭК 8802-5.

Если биты А установлены в 1, биты «копирования кадра» (С) также могут быть установлены в 1 для отражения доступности приемных буферов; в противном случае биты С не должны устанавливаться.

Для обеспечения внутренних услуг подуровня УДС мост с маркерным методом доступа к кольцу должен быть способен распознавать и удалять кадры, передаваемые самому себе, даже если они могут содержать адрес отправителя, отличный от того порта моста, который передавал их.

2.5.4 Обеспечение услуг волоконно-оптическим интерфейсом передачи данных (ВОРИПД)

Метод доступа ВОРИПД определен в ГОСТ Р 50452 и ИСО 9314-2. В разделе 6 настоящего стандарта определены услуги, а в разделах 7 и 8 — средства и операции, соответственно.

При получении действительного кадра (см. 8.3.1 ИСО 9314-2), который не передавался локальным логическим объектом УДС порта моста, с первым битом адреса отправителя, равным нулю, генерируется примитив УД-БЛОК-ДАННЫХ индикация. Параметры этого примитива вырабатываются из полей кадра, как изложено ниже.

Указатель «адрес опознан» (А) поля «состояние кадра» кадра (см. 7.3.8 ИСО 9314-2) в кольце, из которого он был получен, не должен устанавливаться, за исключением случаев, оговоренных в ИСО 9314-2. Указатель «кадр скопирован» (С) (см. 7.3.8 ИСО 9314-2) может быть установлен, если сгенерирован примитив УД-БЛОК-ДАННЫХ индикация с параметрами «тип кадра» и «действие удс» в значениях «кадр данных пользователя» и «запрос без ответа», соответственно, если кадр должен продвигаться и если доступен приемный буфер. В противном случае указатель С не должен устанавливаться, за исключением ситуаций, оговоренных в ИСО 9314-2.

Примечания

1 Изложенные выше требования по установке указателя С мостами УДС дополняют требования ИСО 9314—2 и рассматриваются в рабочей группе ANSI ASC X3T9.5 под названием УДС-2. Однако стандарт IEEE Std. 802.1D не требует выполнения всех функций УДС-2 или полного соответствия УДС-2.

2 Согласно требованиям ИСО 9314-2 мосты могут изменять указатели А и/или С при получении кадра, адресованного мосту как оконечной станции ВОРИПД, или кадра, относящегося к операциям УДС ВОРИПД.

К параметрам, связанным с примитивом УД-БЛОК-ДАННЫХ индикация, который генерируется при получении кадра, в настоящее время относятся следующие.

Параметр «тип-кадра» кодируется битами формата кадра (биты CL, FF и ZZZZ) поля «управление кадра» (см. 7.3.3 ИСО 9314-2). Комбинация битов 0L01rXXX означает «кадр данных пользователя» (асинхронный кадр УЛЗ, где бит L означает длину адреса и может принимать значение 0 или 1; бит r является резервным и может принимать значение 0 или 1; и биты XXX могут принимать значения от 000 до 111). Все остальные битовые комбинации образуют значение «не кадр данных пользователя» параметра «тип кадра».

Параметр «действие удс» принимает единственное значение «запрос без ответа» и явно не кодируется в кадре УДС.

Параметр «адрес получателя» преобразуется в поле «адрес получателя» кадра УДС (см. 7.3.4, 7.3.4.1 ИСО 9314-2).

Параметр «адрес отправителя» преобразуется в поле «адрес отправителя» кадра УДС (см. 7.3.4.2 ИСО 9314-2).

Параметр «сервисный блок данных удс» преобразуется в поле «информация» (см. 7.3.5 ИСО 9314-2).

Параметр «приоритет пользователя», относящийся к «кадрам данных пользователя», кодируется битами PPP в поле «управление кадра» (см. 7.3.3.4 ИСО 9314-2), если кадр является асинхронно передаваемым кадром УЛЗ, поле которого «управление кадра» имеет значение 0L010PPP, где бит L означает длину адреса (см. 7.3.3.2 ИСО 9314-2).

Параметр «контрольная последовательность кадра» преобразуется в поле «контрольная последовательность кадра» кадра УДС (см. 7.3.6 ИСО 9314-2). КПК вычисляется как функция содержимого полей «управление кадра», «адрес получателя», «адрес отправителя» и «информация».

При получении примитива УД-БЛОК-ДАННЫХ запрос локальный логический объект УДС формирует кадр, используя обеспечиваемые параметры, как определено выше, добавляет к данным пользователя поля «управление кадра», «адрес получателя», «адрес отправителя» и «контрольная последовательность кадра» и ставит кадр в очередь на передачу при получении приемлемого маркера (см. 8.3.1 ИСО 9314-2). При передаче кадра к нему присоединяются поля «преамбула», «начальный ограничитель», «конечный ограничитель» и «состояние кадра».

Если примитив УД-БЛОК-ДАННЫХ запрос не сопровождается контрольной последовательностью кадра, она вычисляется в соответствии с 7.3.6 ИСО 9314-2.

Комбинация битов поля «управление кадра» должна иметь значение 0L01rPPP, указывая на асинхронный кадр УЛЗ, где бит L означает длину адреса (см. 7.3.3.2 ИСО 9314-2), бит r зарезервирован и установлен в ноль, а биты PPP означают приоритет кадра (см. 7.3.3.4 ИСО 9314-2).

Если значение параметра «приоритет пользователя» специфицировано, оно кодируется битами PPP, а приоритет доступа (см. 8.1.4 ИСО 9314-2) образуется из значения тайм-аута удержания маркера (ТУМ); в противном случае он определяется разработчиком. Если значение параметра «приоритет пользователя» не специфицировано, биты PPP должны быть установлены в ноль и приоритет доступа образуется из значения ТУМ.

Для обеспечения внутренних услуг подуровня УДС мост ВОРИПД удаляет кадры, передаваемые самому себе, согласно требованиям ИСО 9314-2, даже если они могут содержать адрес отправителя, отличный от адреса того порта моста, который их передавал.

3 Принципы работы

В этом разделе определяются принципы и модель работы моста. В частности, в данном разделе:

1) поясняются основные элементы операций моста и перечисляются выполняемые им функции;

2) определяется архитектурная модель моста, управляющая обеспечением этих функций;

3) определяется модель операций моста в терминах процессов и логических объектов, обеспечивающих указанные функции;

4) подробно рассматриваются требования к адресации в мостовой ЛВС и определяются принципы адресации логических объектов моста.

3.1 **Операции моста**

К основным элементам операций моста относятся:

1) ретрансляция и фильтрация кадров;

2) обеспечение информацией, необходимой для принятия решения о фильтрации кадров;

3) административное управление перечисленными выше элементами операций.

3.1.1 Ретрансляция

Мост УДС ретранслирует отдельные кадры УДС между различными УДС мостовой ЛВС, подключенными к его портам. При этом последовательность поступления кадров в один порт и их передачи в другой порт сохраняется неизменной.

К функциям, обеспечивающим ретрансляцию кадров и сохранение качества услуг, выполняемым мостом, относятся:

1) прием кадров;

2) аннулирование кадров, полученных с ошибкой (см. 2.3.2);

3) аннулирование кадров, у которых параметр «тип кадра» не является «кадром данных пользователя» или значение его параметра «действие удс» не равно «запрос без ответа» (см. 2.4);

4) аннулирование кадров в результате применения информации фильтрации;

5) аннулирование кадров, полученных с чрезмерно длинным сервисным блоком данных (см. 2.3.8);

6) продвижение принятых кадров в направлении других портов моста;

7) аннулирование кадров для предотвращения превышения максимальной транзитной задержки (см. 2.3.6);

8) выбор приоритета исходящего доступа (см. 2.3.9);

9) преобразование сервисных блоков данных и перерасчет контрольной последовательности кадра (см. 2.3.7);

10) передача кадров.

3.1.2 Информация фильтрации

Мосты отфильтровывают кадры, т. е. не продвигают полученные портом моста кадры в другие порты данного моста, чтобы не допустить дублирования кадров (см. 2.3.4). Передача кадров между двумя оконечными станциями может быть ограничена только теми ЛВС, которые формируют маршрут между этими станциями (см. 2.3.10).

К функциям, которые обеспечивают использование и поддержание информации фильтрации, относятся:

1) обеспечение постоянной структуры резервных адресов;

2) четкое построение статической информации фильтрации;

3) автоматическое изучение динамической информации путем наблюдения за трафиком мостовой ЛВС;

4) перевод в разряд устаревшей автоматически изученной информации фильтрации;

5) вычисление и построение топологии мостовой ЛВС.

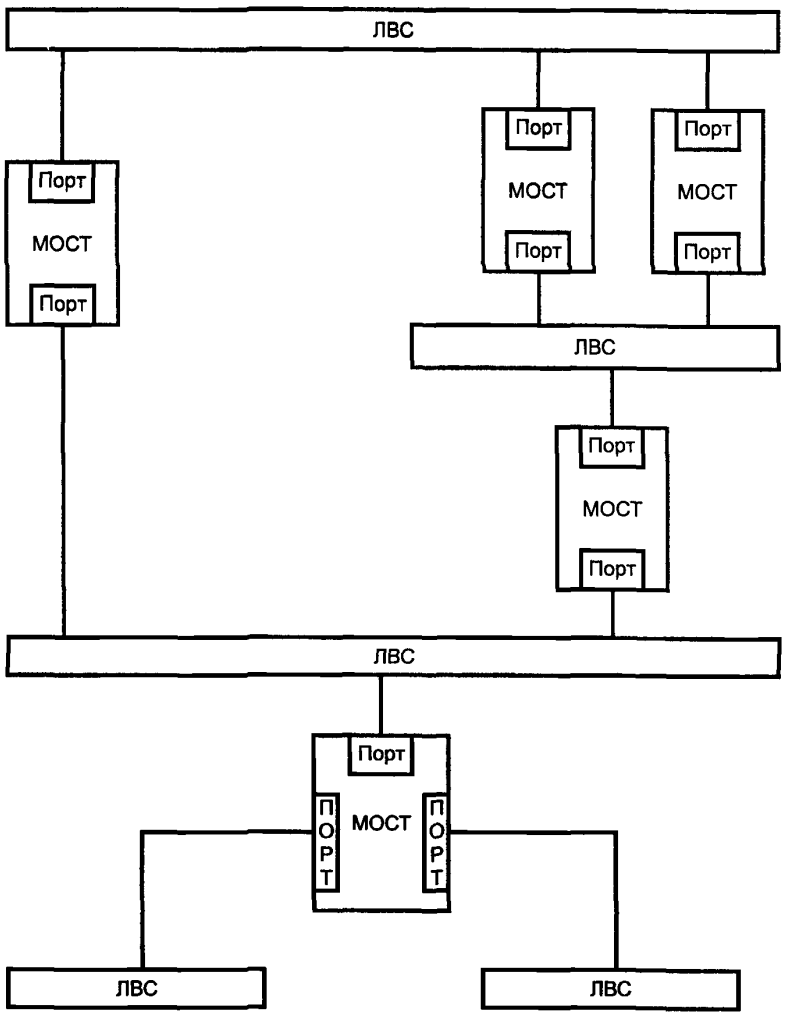


Рисунок 3.1 — Мостовая локальная вычислительная сеть

3.1.3 Диспетчер моста

Функции диспетчера моста обеспечивают контроль и управление выполнением перечисленных выше функций. Они определяются в разделе 6.

3.2 **Архитектура моста**

Каждый мост принимает и передает кадры в ЛВС, к которой он подключен, используя услуги отдельного логического объекта УДС, связанного с этим портом. Логический объект УДС каждого порта оперирует функциями, зависимыми от метода доступа к среде (протокол и процедуры УДС), как определено в соответствующем стандарте по конкретной технологии УДС.

Ретрансляционный логический объект УДС, не зависимый от метода доступа к среде, выполняет функции ретрансляции кадров между портами моста, фильтрации кадров и изучения информации фильтрации.

Протокольный логический объект моста вычисляет и строит топологию мостовой ЛВС.

Ретрансляционный логический объект УДС использует внутренние услуги и подуровня, обеспечиваемые отдельными логическими объектами УДС каждого порта. Эти услуги и их обеспечение определены в 2.4 и 2.5. Они не зависят от конкретной технологии УДС.

Протокольный логический объект моста и другие пользователи протоколов более высоких уровней, например диспетчер моста, используют процедуры управления логическим звеном. Эти процедуры обеспечиваются отдельно для каждого порта и используют услуги УДС, обеспечиваемые отдельными логическими объектами УДС.

На рисунках 3.2 и 3.3 показан мост и его порты, а также архитектура моста с двумя портами. Мосты могут иметь более двух портов.

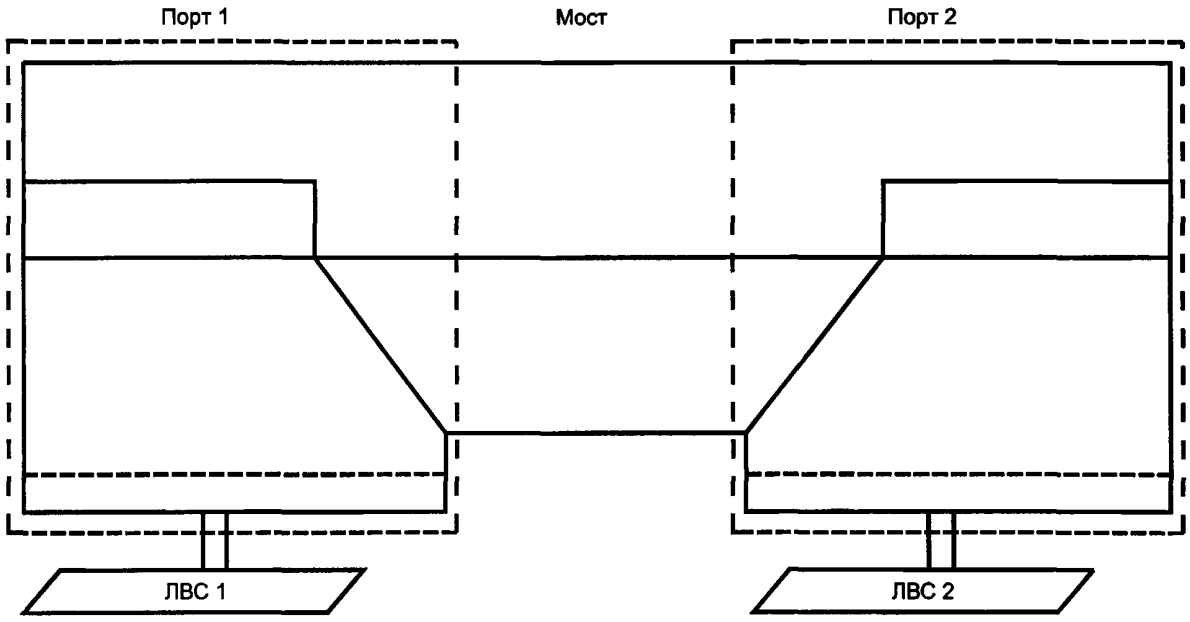


Рисунок 3.2 — Порты моста

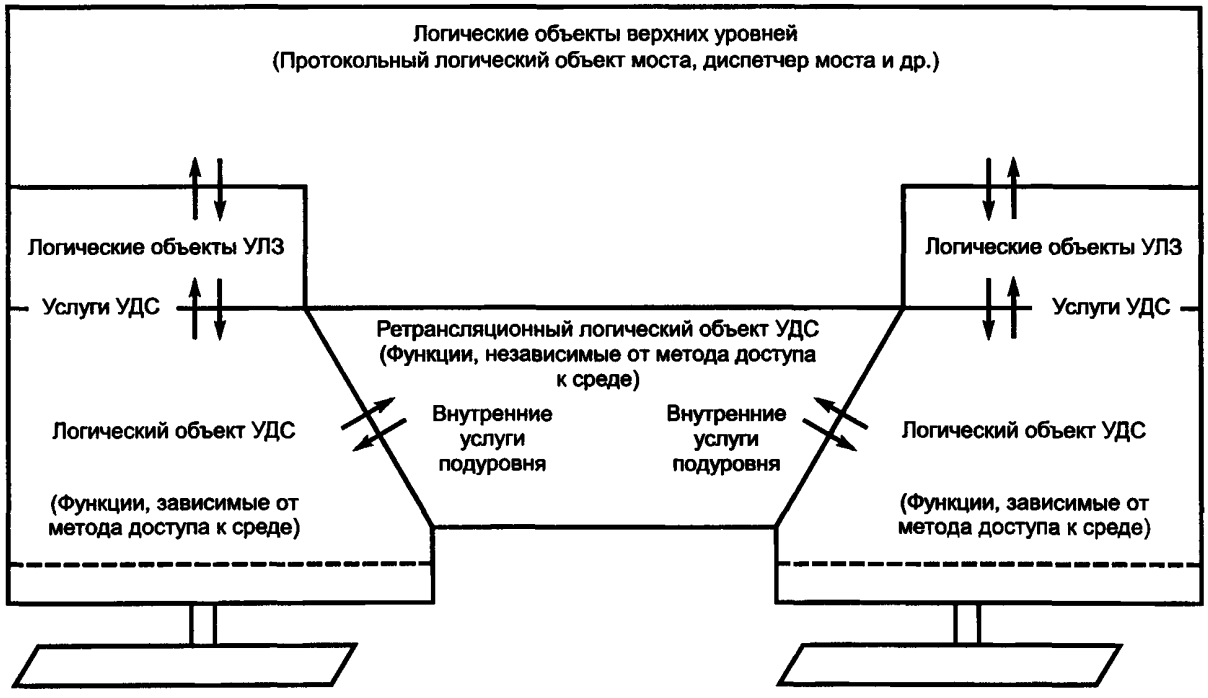


Рисунок 3.3 — Архитектура моста

3.3 **Модель операций**

Использование ретрансляционным логическим объектом УДС внутренних услуг подуровня, обеспечиваемых отдельными логическими объектами УДС, связанными с каждым портом моста, определены в 3.5 и 3.6.

Кадры принимаются из процессов и логических объектов, которые моделируют операции ретрансляционных логических объектов УДС моста, для их последующей передачи, а при получении доставляются указанным процессам и объектам. К этим процессам и объектам относятся:

1) процесс продвижения (см. 3.7), который продвигает получаемые кадры, направляя их в остальные порты моста, отфильтровывает кадры на основе информации, содержащейся в базе данных фильтрации (см. 3.9), и исходя из состояний портов моста (см. 3.4);

2) процесс обучения (см. 3.8), который путем наблюдения за адресами отправителя кадров, получаемых каждым портом, обновляет базу данных фильтрации (см. 3.9), основываясь на состоянии порта (см. 3.4), в котором наблюдаются эти кадры;

3) база данных фильтрации (см. 3.9), которая оперирует информацией фильтрации.

Каждый порт моста функционирует также как оконечная станция, предоставляющая услуги УДС подуровню УЛЗ, который обеспечивает:

1) протокольный логический объект моста (см. 3.10), реализующий протокол построения конфигурации между мостами на подуровне УДС (см. раздел 4) и частично определяющий состояние каждого порта моста (см. 3.4) и их участие в работе активной топологии мостовой ЛВС.

2) других пользователей УЛЗ, таких как протоколы, обеспечивающие функции административного управления моста.

Каждый порт моста должен выполнять операции процедур УЛЗ типа 1 для обеспечения операций протокольного логического объекта моста. Порты моста могут обеспечивать другие типы процедур УЛЗ, которые могли бы использоваться другими протоколами.

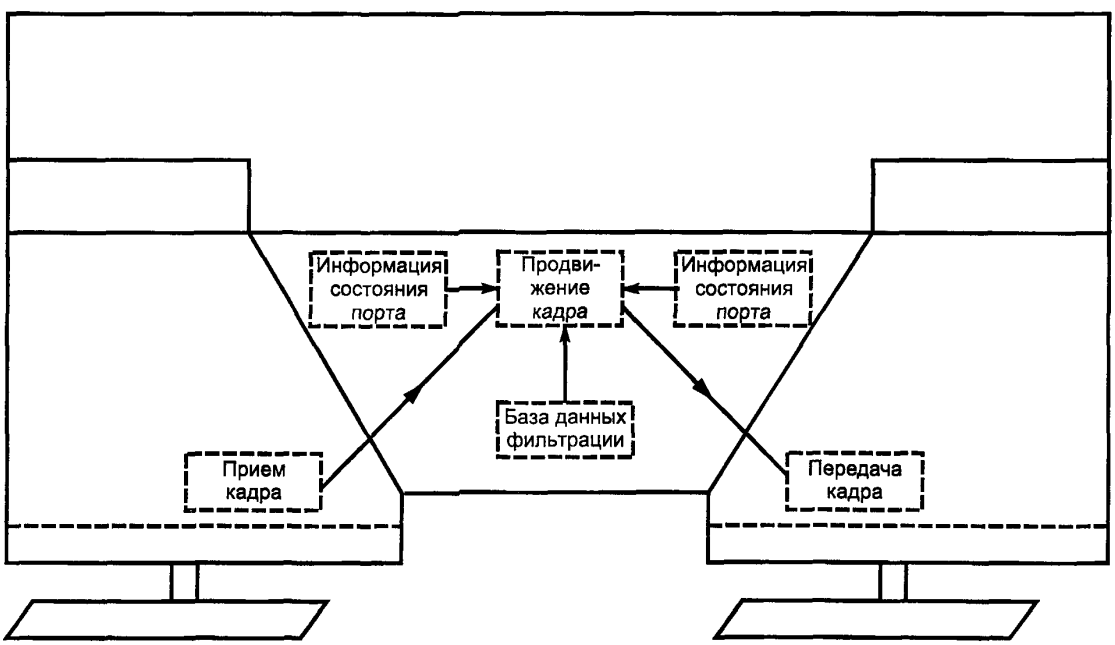


Рисунок 3.4 — Ретрансляция кадров УДС

На рисунке 3.4 показан отдельный пример ретрансляции кадров между портами двухпортового моста.

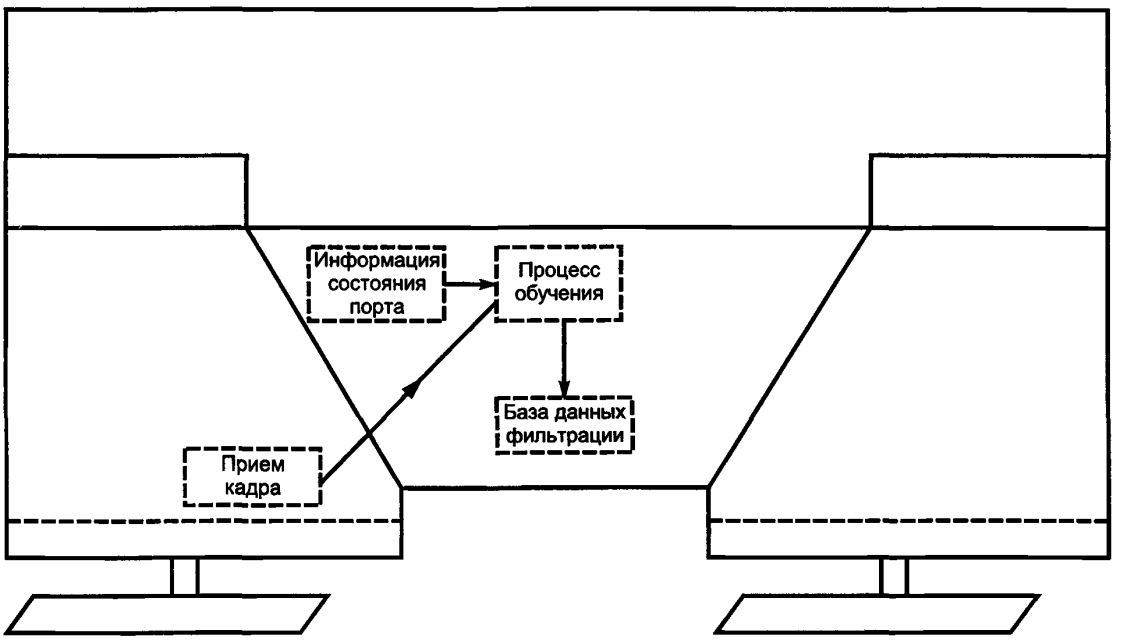


Рисунок 3.5 — Вид сетевого трафика

На рисунке 3.5 показано введение в базу данных фильтрации информации, переданной одним кадром и полученной одним из портов двухпортового моста.

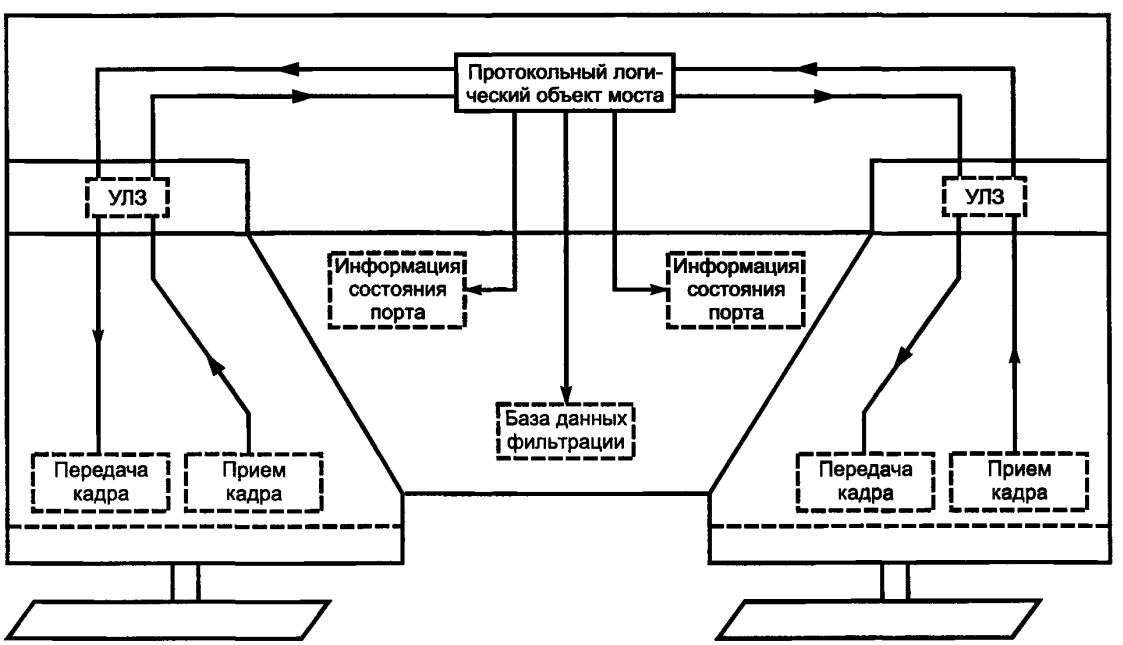


Рисунок 3.6 — Операции межмостового протокола

На рисунке 3.6 показаны прием и передача ПБДМ протокольным логическим объектом моста.

3.4 **Информация о состоянии порта**

Информация о состоянии каждого порта управляет их участием в работе мостовой ЛВС. Если диспетчер допускает порт к участию в ретрансляции кадров и этот порт способен выполнять данную функцию, порт считается активным.

Настоящий стандарт определяет использование алгоритма и протокола покрывающего дерева, которые сводят топологию мостовой ЛВС к простой активно связанной топологии (см. раздел 4). В подразделе 4.4 специфированы состояния порта, относящиеся к этому механизму. Порты, участвующие в ретрансляции кадров, описываются как находящиеся в состоянии продвижения.

Введение информации о месте нахождения оконечной станции в базу данных фильтрации посредством процесса обучения также зависит от активной топологии. Если информация, относящаяся к принятым портом кадрам, подлежит введению в базу данных фильтрации процессом обучения, то этот порт характеризуется как находящийся в состоянии обучения, в противном случае, он не находится в этом состоянии.

На рисунке 3.4 показано использование процессом продвижения информации состояния порта, получившего кадр, с целью определения необходимости ретрансляции принятого кадра через другие порты моста, а также использование информации состояния порта для второго порта моста с целью определения необходимости продвижения через этот порт передаваемого кадра.

На рисунке 3.5 показано использование процессом обучения информации состояния порта, получившего кадр, с целью определения необходимости введения в базу данных фильтрации информации о месте нахождения станции.

На рисунке 3.6 показаны операции протокольного логического объекта моста, реализующие алгоритм и протокол покрывающего дерева, и операции по модификации информации состояния порта, выполняемые ими как часть процесса определения активной топологии мостовой ЛВС.

3.5. **Прием кадров**

Отдельный логический объект УДС, связанный с каждым активным портом моста, просматривает все кадры, передаваемые по ЛВС, к которой он подключен.

Кадры с ошибками, определенными соответствующими методами доступа к среде, должны быть аннулированы; к ним относятся кадры, имеющие ошибку КПК.

Все остальные кадры направляются к процессу обучения.

Кадры примитива УД-БЛОК-ДАННЫХ индикация с параметрами «тип кадра» и «действие удс» в значениях «кадр данных пользователя» и «запрос без ответа» соответственно (см. 2.5) направляются процессу продвижения. Кадры, у которых указанные параметры имеют другие значения, такие как «запрос с ответом» и «ответ», аннулируются и не должны регранслироваться мостом.

Кадры, адресованные порту моста, который ведет себя как обычная оконечная станция, и имеющие параметр «тип кадра» в значении «кадр данных пользователя», должны передаваться подуровню УЛЗ. Такие кадры переносят в поле «адрес получателя» либо индивидуальный адрес УДС порта, либо групповой адрес, связанный с этим портом (см. 3.12).

Кадры, ретранслируемые в данный порт моста из других портов того же моста и адресуемые этому порту моста как оконечной станции, должны также передаваться подуровню УЛЗ.

3.6 **Передача кадров**

Отдельный логический объект УДС, связанный с каждым активным портом моста, передает кадры, направляемые ему ретрансляционным логическим объектом УДС.

Ретранслируемые кадры направляются для передачи процессом продвижения. Примитив УД-БЛОК-ДАННЫХ запрос, относящийся к таким кадрам, переносит значения полей «адрес получателя» и «адрес отправителя», полученных в соответствующем примитиве УД-БЛОК-ДАННЫХ индикация.

Протокольный блок данных УЛЗ передается подуровнем УЛЗ, который ведет себя как пользователь услуг УДС, предоставляемых портом моста. Передаваемые кадры, которые переносят такие ПБД, содержат в поле «адрес отправителя» индивидуальный адрес УДС порта.

Каждый кадр передается процедурами УДС, определяемыми конкретной технологии стандартных ЛВС. Параметры «тип кадра» и «действие удс» соответствующего примитива «УДС-БЛОК-ДАННЫХ запрос» имеют значения «кадр данных пользователя» и «запрос без ответа» соответственно (см. 2.5).

Кадры, переданные по запросу пользователя подуровня УЛЗ услуг УДС, обеспечиваемых портом моста, также должны направляться ретрансляционному логическому объекту УДС.

3.7 **Продвижение кадра**

Процесс продвижения направляет полученные кадры, подлежащие ретрансляции, в другие порты моста, фильтруя их на основе информации, содержащейся в базе данных фильтрации, и с учетом состояния портов моста.

3.7.1 Условия продвижения

Кадры, поступившие в активный порт моста и направленные процессу продвижения, ставятся в очередь на передачу в каждом другом активном порту моста только в том случае, если:

1) порт, в который поступит кадр, находится в состоянии продвижения (см. 4.4) и

2) порт, в который должен быть передан кадр, находится в состоянии продвижения, и

3) либо

a) база данных фильтрации указывает, что кадр с таким значением поля «адрес получателя» должен быть продвинут через передающий порт (например это может быть, если в базе данных фильтрации отсутствует адрес получателя), либо

b) значения полей «адрес получателя» и «адрес отправителя» одинаковые и мост не выполняет фильтрацию этого кадра, и

4) максимальная длина сервисного блока данных обеспечивается ЛВС, к которой подключен передающий порт, и не будет превышена.

3.7.2 Проверка дублирования адресов УЛЗ

Мост может либо:

1) отфильтровывать кадры, в которых поля «адрес отправителя» и «адрес получателя» на подуровне УДС имеют одинаковые значения с целью локализации трафика, либо

2) продвигать такие кадры для обеспечения факультативной функции проверки дублирования адресов УЛЗ.

3.7.3 Постановка кадров в очередь

Процесс продвижения обеспечивает память для постановки кадров в очередь в ожидании возможности их передачи отдельным логическим объектам УДС, связанным с каждым портом моста. При этом должен поддерживаться определенный порядок передачи кадров из очереди. Кадр, поставленный в очередь процессом продвижения для передачи в порт, должен быть изъят из этой очереди при передаче отдельному логическому объекту УДС этого порта; не следует продолжать попытки передачи кадра в этот порт, если становится известно, что передача окажется безуспешной.

Кадр, поставленный в очередь процессом продвижения для передачи в порт, может быть удален из этой очереди без последующей передачи, если время, на которое гарантирована буферизация, истекло для этого кадра.

Кадр, поставленный в очередь для передачи в порт, должен быть удален из этой очереди без последующей передачи отдельному логическому объекту УДС этого порта, если максимальная мостовая транзитная задержка (см. 2.3.6) может быть превышена на время, в течение которого этот кадр мог бы передаваться.

Кадр, поставленный в очередь для передачи в порт, должен быть удален из этой очереди, если этот порт выходит из состояния продвижения.

Удаление кадра в каком-либо порту из очереди на передачу само по себе еще не означает, что он должен быть удален из очереди на передачу в любом другом порту.

3.7.4 Преобразование приоритетов

Процесс продвижения выполняет преобразование приоритетов продвигаемых кадров (см. 2.3.9). Он определяет значения параметров «приоритет пользователя» и «приоритет доступа», используемых при ретрансляции кадров.

Параметр «приоритет пользователя» в примитиве УД-БЛОК-ДАННЫХ запрос (см. 2.5) должен быть:

1) равен значению параметра «приоритет пользователя» соответствующего примитива УД-БЛОК-ДАННЫХ индикация, если он специфицирован, т. е. если кадр получен из ЛВС, использующей маркерный метод доступа к шине, маркерный метод доступа к кольцу или метод доступа ВОРИПД, либо

2) установлен в значение параметра «приоритет пользователя при отправлении» (см. ниже) для передающего порта, если его значение в соответствующем примитиве УД-БЛОК-ДАННЫХ индикация не было специфицировано, т. е. если кадр был получен из ЛВС, использующей метод доступа КДОН/ОК.

Параметр «приоритет доступа» в примитиве УД-БЛОК-ДАННЫХ запрос (см. 2.5) должен быть:

a) установлен в значение параметра «приоритет доступа при отправлении» (см. ниже) для передающего порта или

b) равен значению параметра «приоритет пользователя» в примитиве запроса, или

c) для случая ВОРИПД (ИСО 9314-2) образуется из значения ТУМ в виде значения по умолчанию; в противном случае его значение выбирается разработчиком.

Параметры «приоритет пользователя при отправлении» и «приоритет доступа при отправлении» могут быть установлены диспетчером. Если такая возможность обеспечивается, значения этих параметров должны быть установлены независимо для каждого передающего порта, и мост должен быть способен использовать полный ряд значений в диапазоне, приведенном в таблицах 3.1 и 3.2. Если такая возможность не обеспечивается, мост должен использовать значения по умолчанию, приведенные в таблицах 3.1 и 3.2.

Заметим, что метод доступа КДОН/ОК рассматривает все значения параметра «приоритет доступа» одинаковыми и не передает значения параметра «приоритет пользователя». Параметры «приоритет пользователя при отправлении» и «приоритет доступа при отправлении» для этого метода доступа определяются таким образом, чтобы обеспечить согласованный подход к административному управлению портов моста различных типов УДС.

Таблица 3.1 — Приоритеты пользователя при отправлении

Параметр	Значение рекомендуемое или по умолчанию	Диапазон
Приоритет пользователя при отправлении (КДОН/ОК, ГОСТ 34.913.3)	0	0—7
Приоритет пользователя при отправлении (шина с маркерным доступом, ГОСТ 34.913.4)	0	0—7
Приоритет пользователя при отправлении (кольцо с маркерным доступом, ГОСТ Р ИСО/МЭК 8802-5)	0	0—7
Приоритет пользователя при отправлении (ВОРИПД, ГОСТ Р 50452)	0	0—7

Таблица 3.2 — Приоритеты доступа при отправлении

Параметр	Значение рекомендуемое или по умолчанию	Диапазон
Приоритет доступа при отправлении (КДОН/ОК, ГОСТ 34.913.3)	0	0—7
Приоритет доступа при отправлении (шина с маркерным доступом, ГОСТ 34.913.4)	0	0—7
Приоритет доступа при отправлении (кольцо с маркерным доступом, ГОСТ Р ИСО/МЭК 8802-5)	4	0—7

3.7.5 Перерасчет КПК

В тех случаях, когда кадр продвигается между двумя отдельными логическими объектами УДС одного и того же типа стандартных ЛВС (включая ЛВС типа ВОРИПД), КПК, полученную в примитиве УД-БЛОК-ДАННЫХ индикация, можно передать в соответствующем примитиве УД-БЛОК-ДАННЫХ запрос и повторно не вычислять (см. 2.3.7).

В случаях использования ЛВС различных типов это сделать невозможно и КПК должна пересчитываться в соответствии с конкретными процедурами УДС передающего логического объекта УДС.

3.7.6 Модель

На рисунке 3.4 на примере ретрансляции кадра между портами двухпортового моста показаны операции процесса продвижения.

3.8 **Процесс обучения**

Процесс обучения наблюдает адреса отправителя кадров, получаемых каждым портом моста, и модифицирует базу данных с учетом состояния принимающего порта.

Кадры направляются процессу обучения отдельными логическими объектами УДС, связанными с каждым портом моста, как определено в 3.5.

Процесс обучения может определить маршрут через мостовую ЛВС для конкретных оконечных станций путем просмотра поля «адрес отправителя» получаемых кадров. Он может создавать или изменять динамические записи (см. 3.9, 3.9.2) в базе данных фильтрации, относящиеся к порту, в котором был получен кадр с адресом УДС в поле «адрес отправителя» только в том случае, если:

1) порт, получивший кадр, находится в состоянии, позволяющем выполнять обучение (см. 4.4), и
2) поле кадра «адрес отправителя» указывает определенную оконечную станцию, т. е. не групповой адрес, и
3) статическая запись (см. 3.9, 3.9.1) соответствующего адреса УДС уже не существует, и
4) суммарное число записей не превышает емкости базы данных фильтрации.

Если емкость базы данных фильтрации уже заполнена и должна быть введена новая запись, то существующая запись может быть удалена, чтобы освободить место для новой записи.

На рисунке 3.5 показаны операции процесса обучения по введению в базу данных фильтрации информации о местонахождении станции, переданной отдельным кадром и полученной одним из портов моста.

3.9 **База данных фильтрации**

База данных фильтрации содержит информацию фильтрации, которая либо явно сформирована действием диспетчера, либо автоматически введена процессом обучения. Она обеспечивает различные запросы со стороны процесса продвижения, например, о необходимости передачи кадров с заданными значениями адреса получателя УДС в данный порт.

База данных фильтрации должна содержать в себе статические записи (см. 3.9.1) и обеспечивать возможность хранения динамических записей (см. 3.9.2). Для заданного адреса УДС не должны иметь место оба типа записей: динамическая запись не должна создаваться при наличии статической записи; создание статической записи должно приводить к удалению динамической записи для того же адреса УДС, если такая запись уже существует.

База данных фильтрации может быть опрошена и откорректирована диспетчером. Такая диспетчеризация может быть выполнена либо локальными или частными средствами, либо путем использования возможностей удаленного диспетчера, обеспечиваемых логическим объектом диспетчера моста (см. 3.11), с использованием операций, определенных в разделе 6.

3.9.1 Статические записи

Статические записи могут быть занесены в базу данных фильтрации и удалены из нее под непосредственным управлением диспетчера. Они не удаляются автоматически никаким механизмом таймирования.

Статические записи определяют:

1) адрес УДС-получателя, для которого определена фильтрация;
2) для каждого входящего порта, в котором принимаются кадры, портовое преобразование, определяющее для каждого исходящего порта, из которого могут быть переданы кадры, необходимость фильтрации и продвижения кадров в этот порт.

Адреса УДС, которые могут быть специфицированы, должны включать в себя групповые адреса и глобальный адрес.

3.9.2 Динамические записи

Динамические записи создаются и обновляются процессом обучения, как определено в 3.8. Они должны удаляться по истечении определенного времени с момента их создания или последней коррекции. Такое таймирование записей гарантирует, что оконечные станции, которые переместились в другую часть мостовой ЛВС, не будут иметь постоянных препятствий по получению кадров. Оно учитывает также изменения активной топологии мостовой ЛВС, которые могут быть причиной того, что с точки зрения моста оконечные станции меняют свое местоположение, т. е. маршрут к этим оконечным станциям со временем меняется, проходя через другой порт моста.

Значения тайм-аутов или времени старения, по истечении которого динамическая запись автоматически удаляется, могут устанавливаться диспетчером (см. раздел 6). Рекомендуемые значения по умолчанию для использования в мостовой ЛВС приведены в таблице 3.3; они предложены с тем, чтобы в большинстве случаев исключить необходимость установления явных указанных значений. В таблице 3.3 определен также диапазон используемых значений. Если значение времени старения может

быть установленно диспетчером, мост должен иметь возможность использовать значения заданного диапазона кратностью 1 с.

Определение в разделе 4 алгоритм и протокол покрывающего дерева охватывают процедуры уведомления всех мостов мостовой ЛВС об изменениях топологии и устанавливают короткий тайм-аут, действующий после любого изменения топологии. В периоды отсутствия изменений топологии эти процедуры допускают обычную длительность тайм-аута, достаточную для того, чтобы охватить периоды, в течение которых сами адресуемые оконечные системы не генерируют кадры (возможно, по причине отключения питания) и в то же время не уменьшающую возможностей мостовой ЛВС продолжать обеспечение услуг после автоматической реконфигурации.

Динамические записи определяют:

1) адрес УДС, для которого определена фильтрация;

2) номер порта.

Кадры с определенным адресом УДС - получателя могут продвигаться только в заданный порт. Динамическая запись ведет себя подобно статической записи с единственным портом, выбранным в операции портового преобразования.

Адреса УДС, определенные в динамической записи, не должны содержать групповые и глобальные адреса.

Таблица 3.3 — Значение параметра времени старения

Параметр	Рекомендуемое значение по умолчанию	Диапазон
Время старения, с	300,0	$10,0 - 1,0 \times 10^6$

3.9.3 Постоянная база данных

База данных фильтрации содержит в себе постоянную базу данных, которая обеспечивает фиксированную память для статических записей. База данных фильтрации должна быть инициирована статическими записями, содержащимися в этой постоянной памяти данных.

Статические записи могут быть введены и удалены из постоянной базы данных явными управляющими действиями диспетчера.

3.9.4 Модель операций

На рисунке 3.4 на примере ретрансляции кадра между портами двухпортового моста показано использование базы данных фильтрации процессом продвижения

На рисунке 3.5 показаны создание и корректировка динамических записей в базе данных фильтрации процессом обучения.

На рисунке 3.6 показаны операции протокольного логического объекта моста (см. 3.10), который реализует алгоритм и протокол покрывающего дерева, и информирование им базы данных фильтрации об изменениях активной топологии.

3.10 **Протокольный логический объект моста**

Реализует алгоритм и протокол покрывающего дерева.

Протокольные логические объекты мостов, подключенные через свои порты к одним и тем же отдельным ЛВС в мостовой ЛВС, взаимодействуют между собой путем обмена протокольными блоками данных моста.

На рисунке 3.6 показаны операции протокольного логического объекта моста, включая прием и передачу кадров, содержащих ПБДМ, модификацию информации о состоянии, относящуюся к отдельным портам моста, и информирование базы данных фильтрации об изменении активной топологии.

3.11 **Диспетчер моста**

Мост может обеспечивать средства удаленного административного управления.

Необходимые для этого средства и обеспечиваемые ими операции определены в разделе 6.

В разделе 7 определены протокольные операции, идентификаторы и конкретные значения, подлежащие использованию в реализуемых операциях административного управления согласно требований стандарта IEEE Std 802.1, часть B.

Протоколы диспетчера моста используют услуги, предоставляемые процедурами УЛЗ, которые, в свою очередь, используют услуги УДС, обеспечиваемые в мостовой ЛВС.

3.12 **Адресация**

Все логические объекты УДС мостовой ЛВС должны использовать 48-битовые адреса, которые могут быть либо универсально, либо локально администрируемыми адресами, либо теми и другими.

Мосты могут использовать:

1) 48-битовую универсально администрируемую адресацию или

2) 48-битовую локально администрируемую адресацию, или

3) 16-битовую локально администрируемую адресацию.

Каждый конкретный адрес УДС, используемый в мостовой ЛВС, должен быть уникальным в пределах данной сети с целью недвусмысленной спецификации адресуемой станции.

3.12.1 Оконечные станции

Кадры, передаваемые между оконечными станциями с использованием услуг УДС, обеспечиваемых мостовой ЛВС, содержат адрес УДС - отправителя и получателя равноправных оконечных станций в полях кадра «адрес получателя» и «адрес отправителя» соответственно. Адрес или другие средства идентификации моста не переносятся в кадрах, передаваемых между равноправными пользователями, с целью ретрансляции кадров в мостовой ЛВС.

Широковещательный адрес и другие групповые адреса в целом применимы для использования услуг УДС, обеспечиваемых мостовой ЛВС. Кадры с такими значениями поля «адрес получателя» при отсутствии явно администрируемой конфигурации базы данных фильтрации (см. 3.9 и раздел 6) ретранслируются через мостовую ЛВС.

3.12.2 Порты моста

Отдельный логический объект УДС, связанный с каждым портом моста, должен иметь отдельный специфичный адрес УДС. Этот адрес используется процедурами УДС, которые реализует логический объект УДС, если это требуется конкретным используемым методом доступа к среде.

Кадры, поступающие из ЛВС, к которой подключен порт, и содержащие в поле «адрес получателя» адрес УДС данного порта, доставляются пользователю услуг УДС (УЛЗ) точно так же, как и оконечной станции.

3.12.3 Протокольные логические объекты моста

Принимают и передают только те кадры, которые содержат ПБДМ. Они передаются и принимаются только из других протокольных логических объектов моста (а в ситуациях, когда два порта моста подсоединены к одной и той же ЛВС, передаются и принимаются только между этими портами).

Для передачи ПБДМ протокольный логический объект моста использует примитив ЗД-БЛОК-ДАННЫХ запрос, обеспечиваемый отдельными логическими объектами УЛЗ, связанными с каждым активным портом моста. ПБДМ поступают в соответствующем примитиве ЗД-БЛОК-ДАННЫХ индикация. Параметры примитива ЗД-БЛОК-ДАННЫХ запрос «адрес отправителя» и «адрес получателя» должны указывать стандартные ПДУЗ, назначенные протоколу покрывающего дерева моста.

Каждый примитив ЗД-БЛОК-ДАННЫХ индикация выдается для передачи командного ПБД НИ УЛЗ, который переносит ПБДМ в своем поле информации. Поля «адрес получателя» и «адрес отправителя» ПДУЗ принимают значения, поступившие в соответствующем примитиве запроса. Значение, присвоенное ПДУЗ протокола покрывающего дерева моста, приведено в таблице 3.4.*

Таблица 3.4 — Стандартное присвоение ПДУЗ

Назначение	Значение
Протокол покрывающего дерева моста	01000010
Кодовое представление — бит младшей значимости приведенного значения — крайний слева. Значимость битов возрастает слева направо.	

В настоящем стандарте определяется содержащееся во всех ПБДМ (см. раздел 5) поле «идентификатор протокола», которое служит для идентификации различных протоколов, обеспечиваемых протокольными логическими объектами моста, в сфере действия присвоений ПДУЗ. В настоящем стандарте определяется одно значение идентификатора протокола, которое приведено в разделе 5. Это значение служит для идентификации ПБДМ, обмениваемых между протокольными логическими объектами моста, реализующими алгоритм и протокол покрывающего дерева согласно раздела 4. Все остальные значения этого поля зарезервированы для последующей стандартизации.

* Полный перечень стандартных присвоений ПДУЗ и критерии присвоения содержатся в ИСО/МЭК ТО 10178.

Протокольный логический объект моста, получивший ПБДМ с неизвестным идентификатором протокола, должен аннулировать такой ПБДМ.

Протокольный логический объект моста, реализующий алгоритм и протокол покрывающего дерева согласно раздела 4 настоящего стандарта, всегда передает ПБДМ, адресованные всем остальным протокольным логическим объектам моста, которые подключены к той ЛВС, по которой передавались кадры с ПБДМ. Для адресации данной группы логических объектов в поле «адрес получателя» должен быть использован групповой адрес. Этот групповой адрес должен быть сформирован в постоянной базе данных (см. 3.12.6) с целью ограничения числа ПБДМ, передаваемых в отдельную ЛВС.

Для этой цели назначается 48-битовый универсальный адрес, известный как групповой адрес моста. Значения этого адреса определены в таблице 3.5. Мосты, которые используют 48-битовую универсально администрируемую адресацию, должны вводить этот адрес в поле «адрес получателя» всех кадров УДС, содержащих ПБДМ.

Таблица 3.5 — Зарезервированные адреса

Назначение	Значение	Назначение	Значение
Групповой адрес моста	01—80—C2—00—00—00	Зарезервирован для дальнейшей стандартизации	01—80—C2—00—00—08
Зарезервирован для дальнейшей стандартизации	01—80—C2—00—00—01	То же	01—80—C2—00—00—09
То же	01—80—C2—00—00—02	»	01—80—C2—00—00—0A
»	01—80—C2—00—00—03	»	01—80—C2—00—00—0B
»	01—80—C2—00—00—04	»	01—80—C2—00—00—0C
»	01—80—C2—00—00—05	»	01—80—C2—00—00—0D
»	01—80—C2—00—00—06	»	01—80—C2—00—00—0F
»	01—80—C2—00—00—07		

В поле «адрес отправителя» кадра УДС, содержащего ПБДМ, передается конкретный адрес УДС того порта моста, через который проходит этот ПБДМ (см. 3.12.2).

3.12.4 Логические объекты диспетчера моста

Передают и принимают ПБД с использованием услуг, предоставляемых отдельными логическими объектами УЛЗ, связанными с каждым портом моста. Каждый из этих объектов УЛЗ, в свою очередь, использует услуги УДС, предоставляемые отдельными логическими объектами УДС, связанными с каждым портом, и обеспечиваемые мостовой ЛВС в целом.

Логический объект диспетчера моста, как пользователь услуг УДС, предоставляемых мостовой ЛВС, может быть подключен к любой точке мостовой ЛВС. Кадры, адресованные логическому объекту диспетчера моста, могут при необходимости ретранслироваться мостом, чтобы достичь ЛВС, к которой он подключен.

Чтобы исключить получение кадров - дубликатов, должно соблюдаться основное требование, предъявляемое к отдельной или мостовой ЛВС: каждому пункту подключения должен быть присвоен уникальный адрес.

Логическому объекту диспетчера конкретного моста присваивается один или несколько специфичных адресов УДС в сочетании с идентификатором протокола верхнего уровня и адресной информацией. Вместе с портами данного моста этот объект может участвовать в использовании одного или нескольких пунктов подключения к мостовой ЛВС. Рекомендуется, чтобы этот объект использовал услуги УДС, предоставляемые всеми логическими объектами УДС, связанными с каждым портом моста, т. е. чтобы к нему был обеспечен доступ через каждый порт моста с использованием кадров, содержащих в поле «адрес получателя» специфичный адрес УДС этого порта.

В настоящем стандарте определяется стандартный групповой адрес общего пользования, который служит для передачи запросов административного управления логическому объекту диспетчера моста, связанного со всеми портами моста, подключенными к мостовой ЛВС. Запрос административного управления, который передается в кадре УДС, содержащем такое значение адреса в поле «адрес получателя», как правило, будет приводить к выдаче многих ответов из отдельного моста. Этот адрес известен как групповой адрес диспетчера моста всех ЛВС и имеет значения, приведенные в таблице 3.6.

3.12.5 Уникальный идентификатор моста

Для функционирования протокола моста, определенного в разделе 4, желательно и необходимо, чтобы с каждым мостом был связан один уникальный идентификатор. Этот идентификатор образуется из уникального адреса УДС моста, известного как адрес моста.

Таблица 3.6 — Адресация диспетчера моста

Назначение	Значение
Групповой адрес диспетчера моста всех ЛВС	01—80—С2—00—00—10

В мостовой ЛВС, использующей локально присвоенные 16-битовые адреса, такой адрес должен быть уникален среди всех других адресов, присвоенных в мостовой ЛВС.

В мостовой ЛВС, использующей 48-битовую адресацию, такой адрес должен присваиваться как уникальный в глобальном масштабе, т. е. он должен быть 48-битовым уникально администрируемым адресом.

Рекомендуется, чтобы этот адрес был специфичным адресом УДС порта моста с наименьшим номером (порт 1).

Уникальный идентификатор моста, используемый алгоритмом и протоколом покрывающего дерева, образуется из адреса моста, как определено в 4.5.3 и 5.2.5.

3.12.6 Зарезервированные адреса

Кадры, содержащие в своем поле «адрес получателя» какой-либо из адресов, указанных в таблице 3.5, не ретранслируются мостом. Они должны храниться в постоянной базе данных. Диспетчер не должен иметь возможность удалять такие адреса из постоянной базы данных фильтрации. Эти групповые адреса зарезервированы для присвоения в стандартных протоколах согласно установленным критериям (см. 5.5 IEEE Std 802).

Другие групповые адреса могут быть сформированы в виде статических записей в постоянной базе данных без необходимости их явной инициализации диспетчером. Диспетчер может иметь возможность удаления таких адресов из постоянной базы данных фильтрации.

4 Алгоритм и протокол покрывающего дерева

Алгоритм и протокол конфигурации, описываемые в данном разделе, сводит топологию мостовой ЛВС к отдельному покрывающему дереву.

4.1 **Требования к алгоритму**

Алгоритм покрывающего дерева и соответствующий ему протокол моста работает для обеспечения, поддержки и сохранения качества услуг УДС относительно всех аспектов КУ, рассмотренных в разделе 2. Для выполнения этой функции алгоритм должен удовлетворять следующим требованиям, каждое из которых относится к вопросам данного раздела:

1) он должен организовывать активную топологию мостовой ЛВС любой топологии в единое покрывающее дерево таким образом, чтобы между двумя оконечными станциями существовал только один маршрут данных, исключая шлейфы данных (см. 2.3.3 и 2.3.4);
2) он должен обеспечить устойчивость к неисправностям путем автоматической реконфигурации *топологии покрывающего дерева при неисправностях моста или разрыва тракта данных в* рамках доступных компонентов мостовой ЛВС, и автоматическую адаптацию моста или порта моста, добавленного к мостовой ЛВС, без образования шлейфов данных (см. 2.1);
3) вся активная топология должна быть стабильной в мостовой ЛВС любых размеров. Она должна с высокой вероятностью придти в устойчивое состояние за короткий установленный промежуток времени с целью уменьшения времени, в течение которого услуги оказываются недоступны для использования любой парой оконечных станций (см. 2.1);
4) активная топология должна быть предсказуема, воспроизводима и допускать выбор путем администрирования параметров алгоритма, обеспечивая тем самым применимость диспетчера конфигурации на основе анализа трафика в соответствии с задачами административного управления рабочими характеристиками (см. 2.1 и 2.3.10);

5) для оконечных станций работа алгоритма должна быть прозрачной с тем, чтобы при использовании услуг УДС они не заботились о том, подключены они к отдельной или мостовой ЛВС (см. 2.2);

6) полоса пропускания канала связи, используемая мостом при установлении и использовании покрывающего дерева в любой конкретной ЛВС, должна составлять небольшую долю от общей доступной полосы частот и не должна зависеть от общего трафика, обеспечиваемого мостовой ЛВС, при любом общем количестве мостов или ЛВС (см. 2.3.10).

Кроме того, алгоритм и протокол должны удовлетворять следующим целям, которые ограничивают сложность мостов и их конфигурации:

1) требования к памяти, относящейся к каждому порту моста, не должны зависеть от количества мостов и ЛВС в мостовой ЛВС;

2) мостам не следует задавать конфигурацию заранее до их включения в мостовую ЛВС, за исключением наличия адресов УДС, определенных путем использования обычных процедур.

4.2 Требования к мостам УДС

Для работы мостового протокола требуется следующее:

1) уникальный групповой адрес УДС, распознаваемый всеми мостами мостовой ЛВС, который идентифицирует протокольные логические объекты всех мостов, подключенных к отдельной ЛВС;

2) идентификатор каждого моста, уникальный в пределах мостовой ЛВС;

3) отличный от других идентификатор каждого порта моста, который может быть назначен независимо от значений, используемых в других местах.

Значения каждого из этих параметров и механизм определения их значений должны обеспечиваться каждым мостом. В случае, когда мосты УДС используют 48-битовую универсально администрированную адресацию, уникальный адрес УДС, идентифицирующий протокольный логический объект моста, представляет собой групповой адрес моста (см. 3.12.3).

Кроме того, для обеспечения административного управления конфигурацией покрывающего дерева активной топологии требуется следующее:

1) средства назначения относительного приоритета каждого моста в группе мостов мостовой ЛВС;

2) средства назначения относительного приоритета каждого порта в группе портов отдельного моста;

3) средства, позволяющие определить для каждого порта моста компоненту «стоимость маршрута».

Эти параметры могут устанавливаться диспетчером.

Уникальный идентификатор каждого моста частично образуется из адреса моста (см. 3.12.5) и частично из администрируемой компоненты «приоритет» (см. 5.2.5). Относительный приоритет мостов определяется численным сравнением уникальных идентификаторов, где наименьшее численное значение указывает наивысший приоритет идентификатора.

Одна часть идентификатора каждого порта является фиксированной и различной для каждого порта моста, а другая — администрируемой компонентой «приоритет» (см. 5.2.7). Относительный приоритет портов определяется численным сравнением уникальных идентификаторов, где наименьшее численное значение указывает наивысший приоритет идентификатора.

Стоимость маршрута, связанная с каждым портом моста, может быть администрируемой. Дополнительно в 4.10.2 даны их значения по умолчанию для портов, подключенных к ЛВС к конкретными типами УДС и скоростей.

4.3 **Общие сведения**

4.3.1 Активная топология и ее вычисление

Алгоритм и протокол покрывающего дерева образуют простую взаимоувязанную активную топологию из произвольно соединенных компонентов мостовой ЛВС. Кадры продвигаются через определенные порты моста мостовой ЛВС, минуя заблокированные. В любой момент времени мост эффективно соединяет только те ЛВС, к которым подключены порты, находящиеся в состоянии продвижения. Кадры продвигаются в обоих направлениях через те мостовые порты, которые находятся в состоянии продвижения. Порты, находящиеся в состоянии блокирования, не продвигают кадры ни в одном из

направлений, но могут входить в состав активной топологии, т. е. могут быть переведены в состояние продвижения, если компоненты сети выходят из строя, удаляются или вводятся в сеть.

На рисунке 2.2 показан пример мостовой ЛВС. На рисунке 4.1 показана активная, т. е. логическая увязанная топология одной и той же мостовой ЛВС после ее образования.

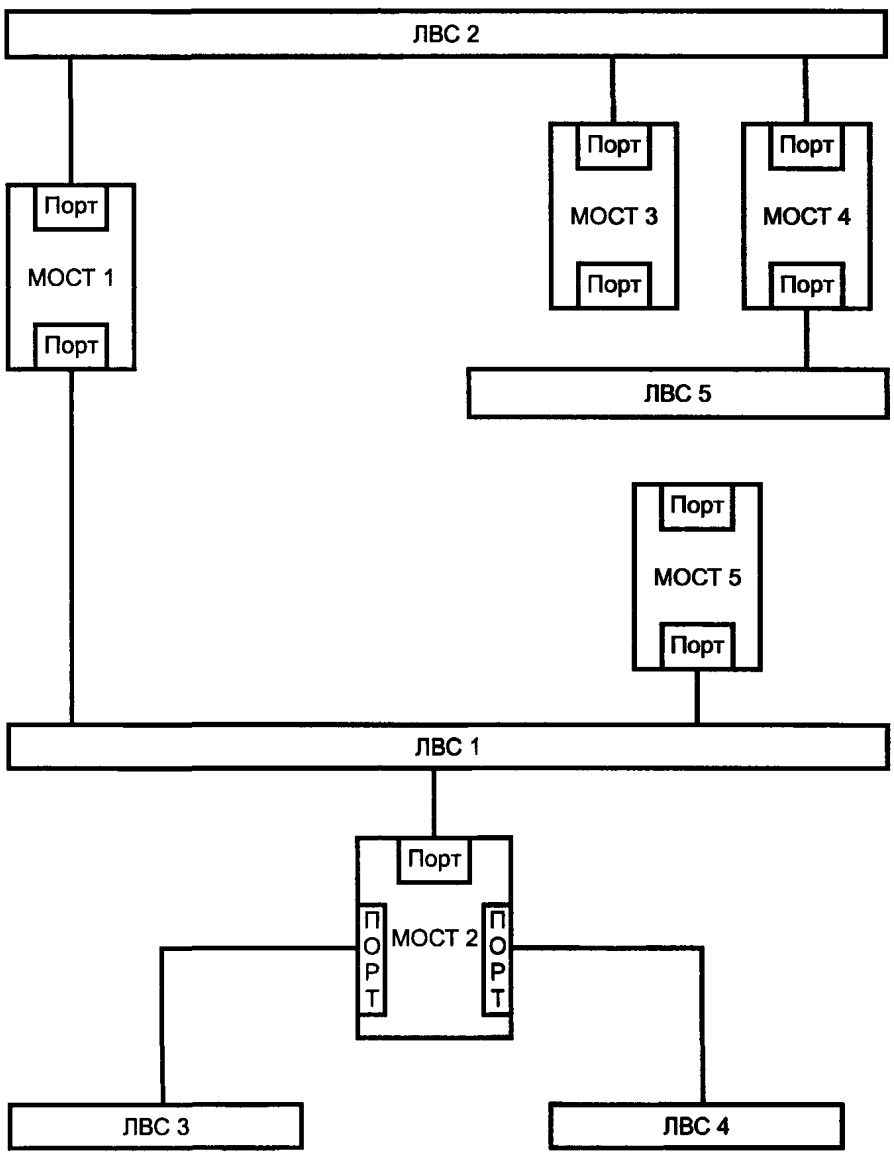


Рисунок 4.1 — Активная топология

Один из мостов называется «корневым мостом» или «корнем» мостовой ЛВС. Каждая отдельная ЛВС соединена с одним из портов моста, который продвигает кадры из этой ЛВС к корневому мосту и из корневого моста к данной ЛВС. Этот порт называется «назначенным портом» для данной ЛВС, а мост, в котором находится этот порт, — «назначенным мостом» для этой ЛВС. Корневой мост является назначенным мостом для всех тех ЛВС, с которыми он связан. Порты каждого моста, находящиеся в состоянии продвижения, являются корневыми портами (расположенными ближе всего к корневому мосту — см. ниже) и портами назначения (если таковые имеются).

На рисунке 4.1 в качестве корневого моста выбран мост 1 (хотя на первый взгляд не видно, какой из мостов является корневым), и он же является назначенным мостом для ЛВС1 и ЛВС2. Мост

2 является назначенным мостом для ЛВС 3 и ЛВС 4, а мост 3 — для ЛВС 5. На рисунке 4.2 показана логическая древовидная топология этой конфигурации мостовой ЛВС.

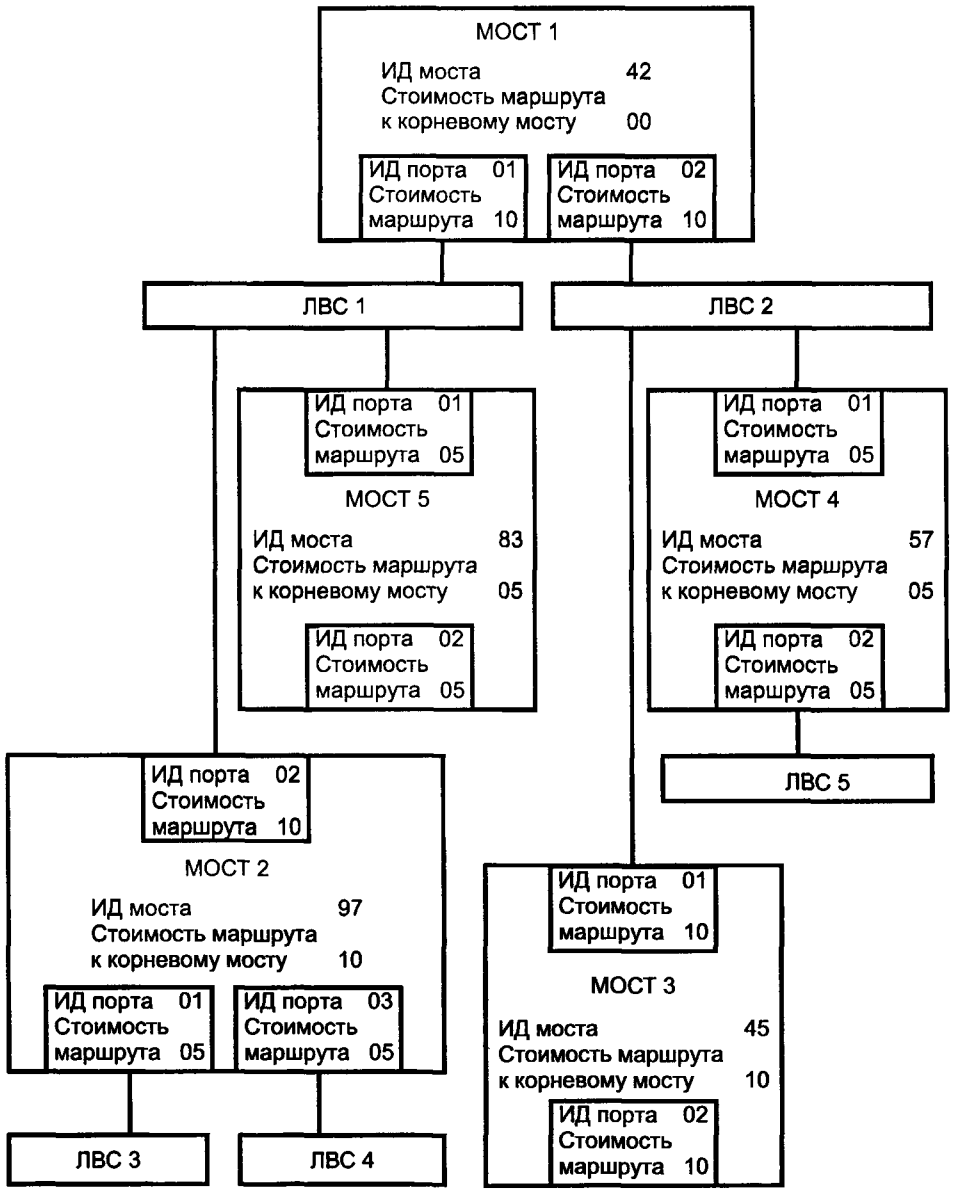


Рисунок 4.2 — Связующее дерево

Устойчивая активная топология мостовой ЛВС определяется:

1) уникальными идентификаторами моста, связанными с каждым мостом;

2) стоимостью маршрута, связанного с каждым портом моста;

3) идентификатором порта, связанного с каждым портом моста.

Мост, у которого идентификатор имеет наивысший приоритет, является корневым (для удобства вычислений этому идентификатору присваивают наименьшее числовое значение). Каждый мост и порт моста мостовой ЛВС имеет соответствующий параметр «стоимость маршрута к корневому мосту» (СМКМ). Это сумма стоимостей маршрутов к каждому порту, принимающему кадры из корневого моста по самому дешевому маршруту к мосту. Назначенным портом для каждой ЛВС является тот порт моста, для которого стоимость маршрута к корневому мосту наименьшая: если два или более портов имеют одинаковое значение СМКМ, то для выхода из тупиковой ситуации сначала используются мостовые

идентификаторы этих мостов, а затем идентификаторы их портов. Таким образом один из портов моста выбирается в качестве назначенного порта для каждой ЛВС и по такому же расчету выбирается среди собственных портов моста корневой порт, что в конечном счете и определяет полностью активную топологию.

Компонента «мостовой идентификатор» каждого моста, а также «стоимость маршрута» и «идентификатор порта» каждого порта моста могут подвергаться административному управлению, что позволяет руководителю выбирать активную топологию мостовой ЛВС.

4.3.2 Распространение информации о топологии

Мосты передают друг другу ПБДМ, известные под названием ПБДМ «конфигурация», для обеспечения взаимосвязи и вычисления указанной выше информации. Кадр УДС, содержащий ПБДМ, переносит групповой адрес моста в поле «адрес получателя» и принимается всеми мостами, подключенными к той ЛВС, через которую передаются кадры.

ПБДМ не продвигаются непосредственно мостами, но содержащаяся в них информация может быть использована мостом при вычислении своего собственного ПБДМ, подлежащего передаче, и может стимулировать его передачу. ПБДМ «конфигурация», который переносит конкретные значения между портами моста, подключенного к отдельной ЛВС, распознается среди сообщений о конфигурации, осуществляющих распространение информации по всей мостовой ЛВС.

Каждый ПБДМ «конфигурация» наряду с другими параметрами содержит уникальный идентификатор того моста, который считает передающий мост корневым, стоимость маршрута к корневому мосту из передающего порта, идентификатор передающего моста и идентификатор передающего порта. Принимающему мосту этой информации достаточно, чтобы определить: имеет ли передающий порт более высокие основания стать назначенным портом для ЛВС, из которой получен ПБДМ «конфигурация», по сравнению с тем портом, который в данный момент расчитывает стать назначенным, и следует ли принимающему порту стать корневым портом для данного моста, если такового еще нет.

Своевременное распространение по всей мостовой ЛВС информации, необходимой всем портам для определения своего состояния (блокирование или продвижение), достигается тремя основными механизмами:

1) мост, который расчитывает стать корневым (все мосты после включения питания надеются стать корневыми, пока не обнаружится наличие другого корневого моста), выдает через регулярные интервалы времени «сообщение о конфигурации» (передаваемое в ПБДМ «конфигурация») во все ЛВС, к которым он подключен;
2) мост, который получает ПБДМ «конфигурация» и приходит к выводу, что его порт корневого моста передает более значимую информацию (идентификатор порта наивысшего приоритета, наименьшая стоимость маршрута к корневому мосту, передающий мост или порт наивысшего приоритета), продвигает эту информацию во все ЛВС, для которых этот порт надеется стать назначенным;
3) мост, который получает менее значимую информацию через порт, рассматриваемый им как назначенный для ЛВС, к которой он подключен, передает в ответ свою собственную информацию для ее опознания всеми другими мостами, подключенными к этой ЛВС.

Следовательно, маршруты покрывающего дерева в направлении к мосту с наивысшим приоритетом идентификатора корневого моста становятся быстро известны в пределах мостовой ЛВС, а менее значимая информация о потенциальных корневых мостах и маршрутах не воспринимается.

4.3.3 Реконфигурация

Чтобы обеспечить реконфигурацию мостовой ЛВС при исключении компонентов сети или при изменении диспетчером параметров, определяющих топологию, информация о топологии, распространяемая по всей мостовой ЛВС, должна иметь ограниченное время существования. Это достигается путем передачи в каждом ПБДМ «конфигурация» «возраста» переносимой информации (времени, прошедшего с момента генерации в корневом мосту сообщения о конфигурации). Каждый мост хранит информацию, полученную назначенным портом из каждой ЛВС, к котором подключены его порты, и следит за «возрастом» этой информации.

При нормальной устойчивой работе регулярная передача корневым мостом сообщений о конфигурации гарантирует своевременность информации о топологии.

Если мост отсчитывает тайм-аут хранения информации для порта, этот порт будет пытаться стать назначенным портом для ЛВС, к которой он подключен, и будет передавать в эту ЛВС протокольную информацию, получаемую из корневого моста через его корневой порт.

При истечении для данного корневого моста установленного тайм-аута в качестве корневого может быть выбран другой порт, при этом информация, передаваемая через ЛВС, для которой данный мост является назначенным, может быть вычислена на основе информации, полученной новым корневым портом.

Если в текущем корневом мосту никакой зарегистрированной информации не осталось, мост будет осуществлять реконфигурацию, объявив сам себя корневым мостом. Если корневой мост выходит из строя, другие мосты будут так же таймировать протокольную информацию, и информация, относящаяся к наилучшему преемнику корневого моста, а также новая топология будут быстро распространяться по всей мостовой ЛВС. Возможно также, что путь к текущему корневому мосту изменился (например, вследствие возрастания стоимости) и реконфигурирующий мост установил таймирование, поскольку он должен рассматривать самую последнюю информацию от корневого моста как менее значимую с момента наличия у него самого дорогостоящего маршрута к корневому мосту. В последнем случае соседние мосты должны немедленно отвечать на ПБДМ, передаваемые мостом, стремящимся стать корневым.

Чтобы все мосты мостовой ЛВС одинаково понимали, когда необходимо начинать таймирование прежней информации, значение тайм-аута передается из корневого моста во всех сообщениях о конфигурации. Это значение учитывает задержки распространения передаваемых и принимаемых ПБДМ в каждой отдельной ЛВС мостовой ЛВС и, тем самым, задержки распространения протокольной информации вниз по покрывающему дереву. Чтобы уменьшить вероятность запуска механизма реконфигурации при потере сообщения о конфигурации, это значение включает дополнительные временные интервалы, в течение которых эти сообщения передаются корневым мостом.

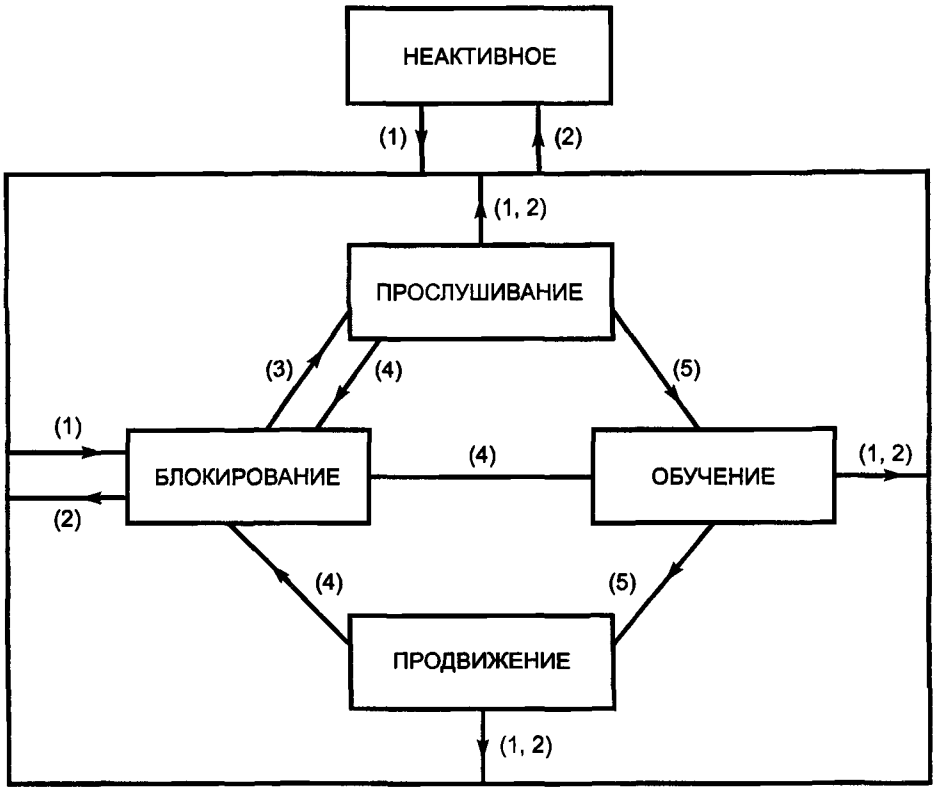
4.3.4 Изменение состояния порта

Поскольку при продвижении протокольной информации по всей ЛВС возникают задержки распространения, резкий переход от одной активной топологии к другой невозможен. Изменения топологии в различных частях мостовой ЛВС могут происходить в различные моменты времени и переход порта моста непосредственно из состояния неучастия в активной топологии в состояние продвижения связан с риском появления временных шлейфов данных, дублирований и нарушения порядка следования кадров. Желательно также перед началом продвижения кадров предоставлять другим мостам время для ответа на менее значимую протокольную информацию.

Следовательно, порты моста должны ожидать информацию о новой топологии для ее распространения по всей мостовой ЛВС и перед продвижением кадров они должны ожидать истечения старой информации об активной топологии с целью определения времени существования любых продвигаемых кадров.

В течение этого времени желательно таймировать информацию о местоположении станции в базе данных фильтрации, которая может оказаться ужу недействительной, и в последней части этого интервала изучать новую информацию о местоположении станции, чтобы уменьшить влияние начального потока кадров при вхождении порта в состояние продвижения. Поэтому когда алгоритм принимает решение о переводе порта в состояние продвижения, он должен сначала ввести его в состояние прослушивания, в котором он должен ожидать протокольной информации, поскольку предполагается, что он может вернуться в заблокированное состояние, и истечения протокольного таймера, которое переводит его в состояние обучения. В этом состоянии он продолжает блокировать продвигаемые кадры, но информация о местоположении станции вводится процессом обучения в базу данных фильтрации. Наконец, истечение протокольного таймера переводит его в состояние продвижения, где разрешено как продвижение ретранслируемых кадров, так и изучение местоположения станции.

Переходы состояний порта показаны на рисунке 4.3.



1 — порт активизирован диспетчером или после инициализации, *2* — порт деактивизирован диспетчером или вышел из строя, *3* — алгоритм выбрал данный порт назначенным или портом корневого моста, *4* — алгоритм не выбрал данныи порт назначенным или портом корневого моста, *5* — истек протокольный таим-аут (тайм-аут продвижения)

Рисунок 4 3 — Состояния порта

4.3.5 Информирование об изменении топологии

При нормальной устойчивой работе информация о местоположении станции в базе данных фильтрации нуждается в обновлении только после физического перемещения станции. Следовательно, может оказаться желательным допустить длительное время старения записей в базе данных фильтрации, особенно когда многие оконечные станции передают кадры после включения питания перемещенных станций, что может привести к необходимости повторного изучения информации о местоположении станции.

Однако перестройка активной топологии мостовой ЛВС для моста может выглядеть как перемещение в сети оконечных станций. Это может иметь место даже в том случае, если состояние портов этого моста остается неизменным. Даже перестройке одной части мостовой ЛВС необходимо повторное изучение местоположения станции.

Алгоритм и протокол покрывающего дерева обеспечивают для моста процедуры по обнаружению изменений в активной топологии, по надежному уведомлению корневого моста об изменениях, а для корневого моста — по последующей передаче изменений во все мосты. При этом мосты используют небольшое значение времени существования записей в базе данных фильтрации в течение установленного периода времени.

Когда мост, не являющийся корневым, изменяет активную топологию мостовой ЛВС, он передает ПБДМ «уведомление об изменении топологии» в ЛВС, которая подключена к порту корневого моста. Такая передача повторяется до тех пор, пока мост не получит подтверждения от моста — получателя этой ЛВС. Подтверждение посылается в ПБДМ «конфигурация», поэтому уведомление либо будет подтверждено, либо произойдет последующая реконфигурация. Назначенный мост направляет уведомление непосредственно корневому мосту или в его направлении, используя ту же процедуру.

Если корневой мост получает такое уведомление или сам изменяет топологию, он должен в течение некоторого времени во всех сообщениях о конфигурации установить флаг «изменение топологии» Это время должно быть таким, чтобы все мосты смогли принять одно или несколько сообщений о конфигурации, или можно было осуществить дальнейшую реконфигурацию Если такой флаг установлен, мосты будут использовать значение «задержка продвижения» (интервал времени, затрачиваемыи в каждом из состояний «прослушивание» или «обучение») для определения времени существования динамических записеи Когда указанный флаг сбрасывается, мосты возвращаются к использованию времени старения записей в базе данных фильтрации

4 4 **Состояния порта**

Операции отдельного порта моста описываются в терминах состоянии порта и процессов (см 3 3), которые обеспечивают и поддерживают функции, необходимые для работы моста (см 3 1)

Состояние каждого порта управляет обработкой кадров, получаемых из отдельного логического объекта УДС, связанного с данным портом (см 3 5), предоставлением кадров логическому объекту УДС для их последующей передачи (см 3 6) и возможностью включения порта в активную топологию мостовои ЛВС

Операции алгоритма и протокола покрывающего дерева служат для поддержания и изменения состояний порта с целью обеспечения требовании, предъявляемых к алгоритму (см 4 1) Возможные состояния порта и соответствующие правила относительно обработки кадров специфичны для данного алгоритма и протокола моста

Ниже для каждого из пяти состояний, т е «неактивное», «прослушивание», «обучение», «продвижение» и «блокирование», в которых может находиться порт, определяется следующее

1) назначение состояния,
2) аннулирует ли процесс продвижения (см 3 7) полученные кадры,
3) направляет ли процесс продвижения (см 3 7) продвигаемые кадры для дальнейшеи передачи,
4) каким образом процесс обучения (см 3 8) обрабатывает полученные кадры,
5) включает ли протокольныи логический объект моста (см 3 10) порт в свои вычисления активной топологии,
6) условия, при которых порт входит в состояние и выходит из него

4 4 1 Блокирование

В этом состоянии порт не участвует в ретрансляции кадров, предотвращая тем самым дублирование кадров вследствие их прохождения через многие маршруты, существующие в активной топологии мостовои ЛВС

Процесс продвижения должен аннулировать получаемые кадры Он не должен направлять продвигаемые кадры для передачи Процесс обучения не должен вводить информацию о местоположении станции в базу данных фильтрации

Протокольный логический объект моста должен включать порт в свои вычисления активной топологии Принимаемые ПБДМ должны обрабатываться в соответствии с алгоритмом и протоколом покрывающего дерева

В состояние «блокирование» порт входит либо в результате инициализации моста, либо из состояния «неактивное», когда порт активизируется операциями диспетчера В это состояние порт может перейти из состояний прослушивания, обучения или продвижения в результате операций алгоритма и протокола покрывающего дерева Порт входит в состояние «блокирование» при получении информации о том, что другой мост является назначенным мостом для ЛВС, к которой подключен данный порт

Из этого состояния порт может выйти при истечении протокольного тайм-аута или при получении ПБДМ «конфигурация» этим или другим портом с переходом в состояние «прослушивание» в результате операций алгоритма и протокола покрывающего дерева Из этого состояния порт может перейти в состояние «неактивное» в результате действий диспетчера

4 4 2 Прослушивание

В этом состоянии порт готовится к участию в ретрансляции кадров Ретрансляция кадров временно деактивизируется с целью исключения временных шлейфов, которые могут возникнуть в мостовой ЛВС в течение времени существования этого состояния, при изменении активной топологии мостовой ЛВС Состояние обучения деактивизировано, поскольку изменения в активной топологии могут привести к некорректности полученной информации, когда активная топология становится устойчивой

Процесс продвижения должен аннулировать поступающие кадры Он не должен направлять про-

двигаемые кадры для передачи. Процесс обучения не должен вводить информацию о местоположении станции в базу данных фильтрации.

Протокольный логический объект моста должен включать порт в свои вычисления активной топологии. Полученные ПБДМ должны обрабатываться в соответствии с алгоритмом и протоколом покрывающего дерева. ПБДМ могут выдаваться для передачи.

В это состояние порт входит из состояния «блокирование», когда операции алгоритма и протокола покрывающего дерева устанавливают, что порт должен участвовать в ретрансляции кадров.

Из этого состояния порт может выйти при истечении протокольного тайм-аута с переходом в состояние «обучение» в результате операций алгоритма и протокола покрывающего дерева. Из этого состояния порт может выйти при получении ПБДМ этим или другим портом и войти в состояние «блокирование» в результате операций алгоритма и протокола покрывающего дерева. Из этого состояния порт может перейти в состояние «неактивное» или «блокирование» в результате действий диспетчера.

4.4.3 Обучение

В этом состоянии порт готовится к участию в ретрансляции кадров. Ретрансляция кадров временно деактивизируется с целью предотвращения временных шлейфов данных, которые могут появиться в мостовой ЛВС во время существования этого состояния, при изменении активной топологии мостовой ЛВС. Обучение активизировано с тем, чтобы иметь возможность получать информацию до ретрансляции кадров с целью снижения объема ретрансляции необязательных кадров.

Процесс продвижения должен аннулировать принимаемые кадры. Он не должен направлять продвигаемые кадры для передачи. Процесс обучения должен вводить информацию о местоположении станции в базу данных фильтрации.

Протокольный логический объект моста должен включать порт в свои вычисления активной топологии. Поступающие ПБДМ должны обрабатываться в соответствии с требованиями алгоритма и протокола покрывающего дерева. ПБДМ могут направляться для передачи.

В это состояние порт вводится из состояния «прослушивание» операциями алгоритма и протокола покрывающего дерева по истечении протокольного тайм-аута.

Из этого состояния порт может выйти при истечении протокольного тайм-аута с переходом в состояние «продвижение» в результате операций алгоритма и протокола покрывающего дерева. Из этого состояния порт может выйти при получении ПБДМ этим или другим портом и войти в состояние «блокирование» в результате операций алгоритма и протокола покрывающего дерева. Из этого состояния порт может перейти в состояние «неактивное» или «блокирование» в результате действий диспетчера.

4.4.4 Продвижение

В этом состоянии порт участвует в ретрансляции кадров. Процесс продвижения может продвигать полученные кадры. Он может предоставлять продвинутые кадры для передачи. Процесс обучения должен вводить информацию о местоположении станции в базу данных фильтрации.

Протокольный логический объект моста должен включать порт в свои вычисления активной топологии. Принятые ПБДМ должны обрабатываться в соответствии с алгоритмом и протоколом покрывающего дерева. ПБДМ могут направляться для передачи.

В это состояние порт вводится из состояния обучения операциями алгоритма и протокола покрывающего дерева по истечении протокольного тайм-аута.

Из этого состояния порт может выйти при получении ПБДМ этим или другим портом и войти в состояние «блокирование» в результате операций алгоритма и протокола покрывающего дерева. Из этого состояния порт может перейти в состояние «неактивное» или «блокирование» в результате действий диспетчера.

4.4.5 Неактивное состояние

В этом состоянии порт не участвует в ретрансляции кадров и в операциях алгоритма и протокола покрывающего дерева.

Процесс продвижения должен аннулировать полученные кадры. Он не должен направлять кадры для передачи. Процесс обучения не должен вводить информацию о местоположении станции в базу данных фильтрации.

Протокольный логический объект моста не должен включать порт в свои вычисления активной топологии. Поступающие ПБДМ не должны обрабатываться в соответствии с алгоритмом и протоколом покрывающего дерева. ПБДМ не должны направляться для передачи.

В это состояние порт вводится из любого другого состояния операциями диспетчера.

Порт выходит из этого состояния по разрешению диспетчера и входит в состояние «блокирование».

4.5 **Параметры и тайм-ауты протокола**

Информация передается между протокольными логическими объектами отдельных мостов путем передачи кадров, содержащих ПБДМ. В данном подразделе определены параметры, содержащиеся в двух типах ПБДМ: «конфигурация» и «уведомление об изменении топологии». Кодирование этих параметров и дополнительных информационных элементов определено в разделе 5.

Каждый протокольный логический объект моста обеспечивает множество параметров и тайм-аутов независимо от отдельных портов, количества тайм-аутов и параметров каждого порта. В данном подразделе определяются эти параметры, их использование и условия их модификации.

4.5.1 Параметры ПБДМ «конфигурация»

4.5.1.1 *Идентификатор корневого моста*

Это уникальный идентификатор того моста, который мост, передающий ПБДМ «конфигурация», считает корневым.

Он передается для того, чтобы позволить всем мостам согласовать этот корневой мост.

4.5.1.2 *Стоимость маршрута к корневому мосту*

Этот параметр определяет стоимость маршрута к корневому мосту, указанному идентификатором корневого моста, выданным передающим мостом.

Он передается для того, чтобы мост принял решение: какой из мостов, подключенных к ЛВС, из которой получен ПБДМ «конфигурация», обеспечит самую дешевую стоимость маршрута к корневому мосту для данной ЛВС.

4.5.1.3 *Идентификатор моста*

Уникальный идентификатор моста, передающего ПБДМ «конфигурация».

Этот параметр передается для того, чтобы мост:

1) определил в случае с ЛВС, к которой подключено два или более мостов, какие из них обеспечивают равноценные по стоимости маршруты к корневому мосту и какой следует определить как назначенный мост для данной ЛВС;
2) выявил случаи, когда два или более портов одного и того же моста подключены к одной и той же ЛВС, т. е. находятся в прямой взаимосвязи через маршрут мостовой ЛВС, ни один из компонентов которой не работает по алгоритму и протоколу покрывающего дерева.

4.5.1.4 *Идентификатор порта*

Это идентификатор порта передающего моста, через который передан ПБДМ «конфигурация». Он однозначно идентифицирует порт этого моста.

Этот параметр передается для того, чтобы в случае подключения к ЛВС двух или более портов одного и того же моста мост мог решить, какие из портов подсоединены.

4.5.1.5 *Время существования сообщения*

Этот параметр указывает время существования сообщения «конфигурация», отсчитываемое от момента генерации ПБДМ «конфигурация» корневым мостом, который стимулирует генерацию этого ПБДМ.

Он передается мосту для того, чтобы разрешить ему аннулировать информацию, время существования которой превысило максимальное время существования (см. ниже).

4.5.1.6 *Максимальное время существования*

Этот параметр указывает значение тайм-аута, которое должны использовать все мосты мостовой ЛВС. Это значение устанавливается корневым мостом.

Он передается каждому мосту мостовой ЛВС с тем, чтобы они имели согласованные значения, относительно которых проверяется время существования хранимой информации о конфигурации.

4.5.1.7 *Время заявки*

Этот параметр указывает интервал времени, отсчитываемый с момента генерации корневым мостом ПБДМ «конфигурация».

Он не используется непосредственно алгоритмом связующего дерева, однако переносится в ПБДМ «конфигурация», чтобы облегчить функциям диспетчера управление работой протокола.

4.5.1.8 *Задержка продвижения*

Этот параметр указывает значение тайм-аута, которое должны использовать все мосты мостовой ЛВС. Значение задержки продвижения устанавливается корневым мостом.

Он передается для того, чтобы все мосты мостовой ЛВС использовали согласованное значение тайм-аута задержки продвижения при переходе порта из текущего состояния в состояние продвижения. Этот параметр используется также как значение тайм-аута для времени старения динамических записей в базе данных фильтрации, отсчитываемое после изменения активной топологии.

4.5.1.9 *Подтверждение изменения топологии*

Флаг, устанавливаемый мостом, который является назначенным для данной ЛВС и который передает ПБДМ «конфигурация» в ответ на полученный ПБДМ «уведомление об изменении топологии».

Этот параметр передается для того, чтобы обеспечить протокол надежного подтверждения с целью уведомления корневого моста об изменениях в активной топологии.

4.5.1.10 *Изменение топологии*

Флаг, устанавливаемый коневым мостом во всех ПБДМ, передаваемых после уведомления об изменении топологии или при обнаружении такого изменения.

Этот параметр передается для информирования всех мостов мостовой ЛВС о произошедшем изменении активной топологии в части мостовой ЛВС и о необходимости быстрого обновления устаревших записей в базе данных фильтрации с целью ограничения влияния временной изоляции оконечных систем, подключенных к мостовой ЛВС, обусловленной использованием некорректной информации в базе данных фильтрации.

Значение времени старения применительно к динамическим записям базы данных фильтрации становится равным значению параметра задержки продвижения, хранимого для данного моста; т. е. после истечения тайм-аута задержки продвижения при установленном флаге «изменение топологии» во всех сообщениях конфигурации, полученных из корневого моста, в базе данных фильтрации остаются только те динамические записи, которые были созданы или смодифицированы в этот период.

4.5.2 Параметры ПБДМ «уведомление об изменении топологии»

Этот ПБДМ не содержит параметров.

4.5.3 Параметры моста

4.5.3.1 *Корневой назначенный мост*

Это уникальный идентификатор моста, который предполагается корневым мостом. Этот параметр используется в качестве значения параметра «идентификатор корневого моста» во всех ПБДМ «конфигурация», передаваемых этим мостом.

4.5.3.2 *Стоимость маршрута к корневому мосту*

Этот параметр указывает стоимость маршрута от данного моста к корневому мосту. Она равна сумме значений параметров «назначенная стоимость» и «стоимость маршрута», хранимых для данного порта моста. Если данный мост является корневым, этот параметр имеет значение ноль.

Этот параметр используется:

1) для проверки значения параметра «стоимость маршрута к корневому мосту», доставленного в полученном ПБДМ «конфигурация»;
2) в качестве значения параметра «стоимость маршрута к корневому мосту», предложенного во всех ПБДМ «конфигурация», переданных данным мостом.

4.5.3.3 *Порт корневого моста*

Это идентификатор порта, обеспечивающего наименьшую стоимость маршрута к корневому мосту, т. е. порта, для которого сумма значений параметров «назначенная стоимость» и «стоимость маршрута», хранимых для данного порта, является наименьшей.

Если два или более портов обеспечивают одинаковые самые дешевые маршруты к корневому мосту, то портом корневого моста выбирается тот из них, у которого наиболее высокоприоритетный параметр «идентификатор моста», хранимый в виде параметра «назначенный мост» для этого порта.

Если два или более портов обеспечивают одинаковые самые дешевые маршруты к корневому мосту и имеют одинаковые значения параметра «назначенный мост», то портом корневого моста выбирается тот из них, который имеет наиболее высокоприоритетный параметр «назначенный порт», хранимый для данного порта.

И наконец, если два или более портов обеспечивают равные по стоимости маршруты к корневому мосту и имеют одинаковые значения параметров «назначенный мост» и «назначенный порт», то портом корневого моста становится тот из них, который имеет наиболее высокоприоритетный параметр «идентификатор порта». Для различных портов одного и того же моста гарантируется свой, отличный от других, идентификатор и тем самым обеспечивается развязка тупиковых ситуаций.

Этот параметр используется для идентификации порта, через который устанавливается маршрут к корневому мосту. Он теряет свою значимость, когда мост становится корневым и устанавливается в ноль.

4.5.3.4 *Максимальное время существования*

Этот параметр указывает максимальное время существования полученной протокольной информации, отсчитываемое до момента ее аннулирования.

4.5.3.5 *Время заявки*

Этот параметр указывает интервал времени между передачами ПБДМ «конфигурация» мостом, который пытается стать корневым или уже является таковым.

4.5.3.6 *Задержка продвижения*

Этот параметр указывает время, затрачиваемое в состоянии «прослушивание» на переход из состояния «блокирование» в состояние «обучение».

Это время, затрачиваемое в состоянии «обучение» на переход из состояния «прослушивание» в состояние «продвижение».

Этот параметр используется для отсчета времени существования динамических записей в базе данных фильтрации, когда в протокольных сообщениях, полученных из корневого моста, установлен флаг «изменение топологии»

4.5.3.7 *Идентификатор моста*

Это уникальный идентификатор моста, используемый в качестве значения:

1) параметра «идентификатор моста» во всех ПБДМ «конфигурация», передаваемых данным мостом;

2) параметра «корневой назначенный мост», когда мост является корневым или пытается стать таковым после истечения всей информации, касающейся текущего корневого моста, или в результате действий диспетчера.

Этот параметр содержит две части, одна из которых образуется из уникального адреса моста (см. 3.12.5) и гарантирует уникальность идентификатора моста в мостовой ЛВС, а другая позволяет отрегулировать приоритетность идентификатора моста и считается наиболее значимой частью при сравнении приоритетов. Эта приоритетная часть данного параметра может быть скорректирована действиями диспетчера.

4.5.3.8 *Максимальное время существования моста*

Это значение параметра «максимальное время существования», которое он принимает, когда мост является корневым или пытается стать таковым.

Этот параметр может быть скорректирован действиями диспетчера.

4.5.3.9 *Мостовое время заявки*

Этот параметр указывает значение параметра «время заявки», которое он принимает, когда мост является корневым или пытается стать таковым.

Он определяет интервал времени между передачами ПБДМ «уведомление об изменении топологии» по направлению к корневому мосту, когда мост пытается уведомить назначенный мост данной ЛВС, к которой подключен его порт корневого моста, об изменении топологии.

Этот параметр может быть скорректирован действиями диспетчера.

4.5.3.10 *Мостовая задержка продвижения*

Этот параметр указывает значение параметра «задержка продвижения», которое он принимает, когда мост является корневым или пытается стать таковым.

Он может быть скорректирован действиями диспетчера.

4.5.3.11 *Обнаружено изменение топологии*

Это булевый параметр, устанавливаемый в значение «истинно» для регистрации изменений топологии, которые были обнаружены мостом или сообщены мосту.

Он используется:

1) с целью содействия регулярности передач через интервалы, определяемые параметром «время заявки моста» ПБДМ «уведомление об изменении топологии» в направлении корневого моста, где данный мост не является корневым;

2) в значении «истинно» чтобы установить параметр «изменение топологии» для данного моста в значение «истинно», если этот мост является или становится корневым.

4.5.3.12 *Изменение топологии*

Это булевый параметр, устанавливаемый для регистрации значения флага «изменение топологии»

в ПБДМ «конфигурация», подлежащих передаче мостом в те ЛВС, для которых данный мост является назначенным.

Если этот параметр имеет значение «истинно», то тайм-аут существования записей в базе данных фильтрации принимает значение параметра «задержка продвижения». Динамические записи, время существования которых превысило это значение, удаляются из базы данных фильтрации.

Если этот параметр имеет значение «ложно», то тайм-аут старения записей в базе данных фильтрации принимает значение параметра «время старения». Время старения может быть установлено диспетчером (см. раздел 6).

4.5.3.13 *Время изменения топологии*

Этот параметр указывает период времени, в течение которого ПБДМ передаются с флагом «изменение топологии», установленным корневым мостом после обнаружения изменения топологии.

Значение этого параметра равно сумме значений параметров «максимальное мостовое время существования» и «мостовая задержка продвижения». Любой из этих составляющих параметров может быть скорректирован действиями диспетчера.

4.5.3.14 *Время удержания*

Этот параметр определяет минимальный период времени между передачами ПБДМ «конфигурация» через данный порт корневого моста. В любой период времени удержания не должно передаваться более двух ПБДМ «конфигурация».

Он является постоянным параметром моста. Его значение определено в таблице 4.3.

4.5.4 Тайм-ауты моста

4.5.4.1 *Тайм-аут заявки*

Служит для обеспечения периодической передачи ПБДМ «конфигурация» мостом, когда он является или пытается стать корневым.

Значение этого тайм-аута определяется мостовым параметром «время заявки».

4.5.4.2 *Тайм-аут уведомления об изменении топологии*

Должен гарантировать, что назначенный мост ЛВС, к которой подключен мост с портом корневого моста, будет проинформирован об обнаруженном изменении топологии.

Значение этого тайм-аута определяется мостовым параметром «время заявки моста».

4.5.4.3 *Тайм-аут изменения топологии*

Служит для определения периода времени, в течение которого передаются ПБДМ «конфигурация» с флагом «изменение топологии», установленным мостом, когда он становится корневым после обнаружения изменения топологии.

Значение этого тайм-аута определяется мостовым параметром «время изменения топологии».

4.5.5 Параметры порта

4.5.5.1 *Идентификатор порта*

Это идентификатор соответствующего порта, используемый в виде значения параметра «идентификатор порта» всех ПБДМ «конфигурация», передаваемых в соответствующий порт.

Этот параметр состоит из двух частей. Одна часть отражает фиксированное отношение к физическим портам, обеспечиваемым реальным оборудованием. Порты идентифицируются небольшими числами от единицы и выше. Эта часть параметра гарантирует уникальность идентификатора порта среди всех портов одного моста. Другая часть параметра позволяет регулировать приоритет порта и при сравнении приоритетов считается наиболее значимой частью. Приоритетная часть этого параметра может быть скорректирована действиями диспетчера.

4.5.5.2 *Состояние*

Этот параметр указывает текущее состояние порта (т. е. неактивное, прослушивание, обучение, продвижение или блокирование).

Он используется для управления приемом кадров из логического объекта УДС соответствующего порта, процессами обучения и продвижения, продвижением кадров процессом продвижения в данный логический объект УДС, передачей и приемом ПБДМ (см. 4.4).

Он модифицируется действиями протокола и может также корректироваться действиями диспетчера.

4.5.5.3 *Стоимость маршрута*

Этот параметр указывает долю стоимости маршрута, проходящего через данный порт, когда он является портом корневого моста, в общей стоимости маршрута к корневому мосту для данного моста.

Значение этого параметра, добавленное к значению параметра «назначенная стоимость» для порта

корневого моста, используется в качестве значения параметра «стоимость маршрута к корневому мосту», содержащегося во всех ПБДМ «конфигурация», передаваемых мостом, когда он не является корневым.

Этот параметр может быть скорректирован действиями диспетчера.

4.5.5.4 *Корневой назначенный мост*

Это уникальный идентификатор моста, зарегистрированного как корневой мост в параметре «идентификатор корневого моста» ПБДМ «конфигурация», передаваемых назначенным мостом для той ЛВС, к которой подключен этот порт.

Этот параметр используется для проверки значения параметра «идентификатор корневого моста», содержащегося в полученном ПБДМ «конфигурация».

4.5.5.5 *Назначенная стоимость*

Этот параметр указывает стоимость маршрута к корневому мосту, установленную назначенным портом той ЛВС, к которой этот порт подключен.

Он используется для проверки значения параметра «стоимость маршрута к корневому мосту», содержащегося в полученном ПБДМ «конфигурация».

4.5.5.6 *Назначенный мост*

Это уникальный идентификатор моста, предполагающего стать назначенным мостом для ЛВС, связанной с данным портом.

Этот параметр используется:

1) вместе с параметрами порта «назначенный порт» и «идентификатор порта» для того, чтобы определить, должен ли этот порт стать назначенным для ЛВС, к которой он подключен;
2) для проверки значения параметра «идентификатор моста», содержащегося в полученном ПБДМ «конфигурация».

4.5.5.7 *Назначенный порт*

Это идентификатор порта, который предполагает стать назначенным портом для ЛВС, к которой этот порт подключен.

Этот параметр используется:

1) вместе с параметрами порта «назначенный мост» и «идентификатор порта» для того, чтобы определить, должен ли этот порт стать назначенным портом для ЛВС, к которой он подключен;
2) диспетчером для определения топологии мостовой ЛВС.

4.5.5.8 *Подтвердить изменение топологии*

Этот параметр указывает значение флага «подтверждение изменения топологии» в следующем ПБДМ «конфигурация», который должен быть передан соответствующему порту.

Он используется для того, чтобы зарегистрировать необходимость установки флага «подтверждение изменения топологии» в ответ на полученный ПБДМ «уведомление об изменении топологии».

4.5.5.9 *Торможение конфигурации*

Это булевый параметр, устанавливаемый для регистрации необходимости передачи ПБДМ «конфигурация» по истечении тайм-аута удержания для соответствующего порта.

Он используется вместе с тайм-аутом удержания данного порта для того, чтобы ПБДМ «конфигурация» не передаются слишком часто, но при обеспечении неустарелости передаваемой информации.

4.5.6 Тайм-ауты порта

4.5.6.1 *Тайм-аут времени существования*

Служит для измерения времени существования полученной протокольной информации, регистрируемой для данного порта, и для гарантии того, что эта информация будет аннулирована, когда ее время существования превысит значение параметра «максимальное время существования», зарегистрированное мостом.

Значение этого тайм-аута равно мостовому параметру «максимальное время существования».

4.5.6.2 *Тайм-аут задержки*

Служит для контроля времени, затрачиваемого портом в состояниях «прослушивание» и «обучение».

Значение этого тайм-аута равно мостовому параметру «задержка продвижения».

4.5.6.3 *Тайм-аут удержания*

Служит для гарантии того, что ПБДМ «конфигурация» не будут передаваться слишком часто через любой порт моста.

Значение этого тайм-аута равно параметру «время удержания» для данного моста.

4.6 **Элементы процедур**

4.6.1 Передача ПБДМ «конфигурация»

4.6.1.1 *Назначение*

Этот ПБДМ предназначен для передачи другим портам моста, подключенным к той же ЛВС, что и порт, из которого передан ПБДМ «конфигурация», сведений о корневом назначенном мосте, стоимости маршрута к корневому мосту, назначенном мосте, назначенном порту и значений протокольных тайм-аутов.

4.6.1.2 *Использование*

Эта процедура используется:

4.6.1.2.1 как часть процедуры генерации ПБДМ «конфигурация» (см. 6.4.6);

4.6.1.2.2 как часть процедуры ответа на ПБДМ «конфигурация» (см. 4.6.5);

4.6.1.2.3 после истечения тайм-аута удержания для порта (см. 4.7.8), если флаг «торможение конфигурации» для данного порта установлен в результате предыдущего привлечения этой процедуры.

4.6.1.3 *Процедура*

4.6.1.3.1 Если тайм-аут удержания для порта находится в состоянии отсчета, должен быть установлен флаг «торможение конфигурации» для данного порта.

4.6.1.3.2 В противном случае, если тайм-аут удержания не находится в состоянии отсчета, ПБДМ «конфигурация» передается через выбранный порт в пределах времени максимальной задержки передачи ПБДМ (как определено в 4.10.2) после любого привлечения этой процедуры. Параметры ПБДМ «конфигурация» устанавливаются следующим образом:

1) параметр «идентификатор корневого моста» — в значение параметра моста «корневой назначенный мост»;
2) параметр «стоимость маршрута к корневому мосту» — в значение параметра моста «стоимость маршрута к корневому мосту»;
3) параметр «идентификатор моста» — в значение параметра моста «идентификатор моста»;
4) параметр «идентификатор порта» — в значение параметра «идентификатор порта», хранимое в порту моста, через который передан ПБДМ «конфигурация»;
5) если мост выбран как корневой, т. е. если параметры «корневой назначенный мост» и «идентификатор моста» имеют одинаковые значения, то параметр «время существования сообщения» устанавливается в значение ноль;
6) в противном случае значение параметра «время существования сообщения» следует устанавливать таким, чтобы передаваемый ПБДМ «конфигурация» не передавал заниженной оценки времени существования протокольного сообщения, полученного портом корневого моста; т. е. передаваемое значение должно быть не меньше значения, зарегистрированного тайм-аутом «время существования сообщения» для данного порта, должно быть больше полученного значения и может учитывать любую задержку передачи. Значение этого параметра не должно превышать его действующего значения больше чем на максимальное время существования сообщения, увеличивая завышенную оценку как определено в 4.10.2;
7) параметры «максимальное время существования», «время заявки» и «задержка продвижения» следует установить в значение параметров моста «максимальное время существования», «время заявки» и «задержка продвижения»;
8) флаг «подтверждение изменения топологии» следует установить в значение флага порта «подтвердить изменение топологии». Параметр «подтвердить изменение топологии» сбрасывается;
9) флаг «изменение топологии» следует установить в значение флага моста «изменение топологии»;
10) флаг порта «торможение конфигурации» сбрасывается;
11) начинается отсчет тайм-аута удержания для порта.

4.6.2 Регистрация информации о конфигурации

4.6.2.1 *Назначение*

Эта процедура предназначена для регистрации протокольных параметров, содержащихся в ПБДМ «конфигурация», принятым данным портом.

4.6.2.2 *Использование*

Эта процедура используется после получения ПБДМ «конфигурация», содержащего протокольную информацию, которая заменяет уже хранимую информацию, т. е. если:

4.6.2.2.1 параметр «идентификатор корневого моста» определяет мост более высокого приоритета относительно моста, зарегистрированного как корневой назначенный мост, или

4.6.2.2.2 параметр «идентификатор корневого моста» имеет такое же значение, что и параметр «корневой назначенный мост», а параметр «стоимость маршрута к корневому мосту» ниже значения стоимости, зарегистрированной как «назначенная стоимость» для данного порта, или

4.6.2.2.3 параметры «идентификатор корневого моста» и «стоимость маршрута к корневому мосту» соответствуют параметрам, зарегистрированным портом, и «идентификатор моста» означает мост, имеющий более высокий приоритет относительно моста, зарегистрированного как «назначенный мост» для данного порта, или

4.6.2.2.4 параметры «идентификатор корневого моста» и «стоимость маршрута к корневому мосту» такие же, как и зарегистрированные для данного порта, и «идентификатор моста» является таким же, как и зарегистрированный в виде «назначенный мост» для данного порта, и либо:

1) мост, принимающий ПБДМ, не является «назначенным мостом» для данного порта;
2) «идентификатор порта» означает порт с более высоким приоритетом относительно порта, зарегистрированного как «порт назначения».

4.6.2.3 *Процедура*

4.6.2.3.1 Параметры порта «корневой назначенный мост», «назначенная стоимость», «назначенный мост» и «назначенный порт» устанавливаются в значения параметров «идентификатор корневого моста», «стоимость маршрута к корневому мосту», «идентификатор моста» и «идентификатор порта», содержащихся в принятом ПБДМ «конфигурация».

4.6.2.3.2 Запускается тайм-аут «время существования сообщения» для данного порта, начиная со значения параметра «время существования сообщения», содержащегося в принятом ПБДМ «конфигурация».

4.6.3 Регистрация значений тайм-аутов, содержащихся в ПБДМ «конфигурация»

4.6.3.1 *Назначение*

Эта процедура предназначена для замены параметров «максимальное время существования», «время заявки», «задержка продвижения» и флага «изменение топологии» на последние значения, полученные из корневого моста.

4.6.3.2 *Использование*

Эта процедура используется после получения ПБДМ «конфигурация» портом корневого моста, который привлекает процедуру регистрации информации ПБДМ «конфигурация» для данного порта (см. 4.6.2.2).

4.6.3.3 *Процедура*

Хранимые мостом параметры «максимальное время существования», «время заявки», «задержка продвижения» и «изменение топологии» устанавливаются в значения, содержащиеся в полученном ПБДМ «конфигурация».

4.6.4 Генерация ПБДМ «конфигурация»

4.6.4.1 *Назначение*

Эта процедура предназначена для передачи мостам, подключенным к каждой ЛВС, для которой данный мост является назначенным, сведений о корневом назначенном мосте, стоимости маршрута к корневому мосту, назначенному мосту, назначенному порту и значений пртокольных тайм-аутов.

4.6.4.2 *Использование*

Эта процедура используется:

4.6.4.2.1 после приема ПБДМ «конфигурация» портом корневого моста, который привлекает процедуру регистрации информации ПБДМ «конфигурация» для этого порта (см. 4.6.2.2);

4.6.4.2.2 после истечения времени заявки;

4.6.4.2.3 после назначения моста корневым назначенным мостом процессом «изменение конфигурации» при истечении тайм-аута «время существования сообщения» для данного порта моста;

4.6.4.2.4 после назначения моста корневым назначенным мостом действиями диспетчера.

4.6.4.3 *Процедура*

Для каждого порта, который является назначенным для той ЛВС, к которой он подключен (т. е. значения параметров «назначенный мост» и «назначенный порт», хранимые в данном порту, равны

значениям параметров «идентификатор моста» и «идентификатор порта» для данного порта соответственно, и порт не находится в состоянии «неактивное»), используется процедура передачи ПБДМ «конфигурация» (см. 4.6.1).

4.6.5 Ответ на ПБДМ «конфигурация»

4.6.5.1 *Назначение*

Эта процедура предназначена для того, что определить назначенные мост и порт для ЛВС в случае, когда еще один порт моста передал в эту ЛВС ПБДМ «конфигурация». Такая ситуация возникает, если ПБДМ «конфигурация» из текущего корневого моста не был получен передающим мостом или потому, что этот корневой мост только что установлен, или вследствие потери ПБДМ и последующего истечения тайм-аута существования сообщения.

4.6.5.2 *Использование*

Эта процедура используется после получения назначенным портом для данной ЛВС, к которой он подключен, ПБДМ «конфигурация», который не корректирует информацию, хранимую для этого порта, то есть не удовлетворяет условиям использования процедуры регистрации информации (см. 4.6.2.2).

4.6.5.3 *Процедура*

Используется процедура передачи ПБДМ «конфигурация» (см. 4.6.1) для порта, в котором был получен ПБДМ «конфигурация».

4.6.6 Передача ПБДМ «уведомление об изменении топологии»

4.6.6.1 *Назначение*

Эта процедура должна уведомить мост, находящийся на пути к корневому мосту о том, что передающим мостом обнаружено изменение топологии. В конечном счете это должно привести к уведомлению об изменении топологии корневого моста.

4.6.6.2 *Использование*

Эта процедура используется:

4.6.6.2.1 после обнаружения или получения сообщения об изменении топологии мостом, который не является корневым;

4.6.6.2.2 после истечения тайм-аута уведомления об изменении топологии.

4.6.6.3 *Процедура*

ПБДМ «уведомление об изменении топологии» следует передавать через порт корневого моста во время максимальной задержки передачи ПБДМ (см. 4.10.2).

4.6.7 Изменение конфигурации

4.6.7.1 *Назначение*

Эта процедура должна изменить информацию о конфигурации, хранимую мостом и портами моста.

4.6.7.2 *Использование*

Эта процедура используется:

4.6.7.2.1 после получения ПБДМ «конфигурация», который привлекает процедуру регистрации информации ПБДМ «конфигурация» для данного порта (см. 4.6.2.2);

4.6.7.2.2 после того, когда порт становится назначенным для ЛВС, к которой он подключен, при истечении тайм-аута существования сообщения для этого порта;

4.6.7.2.3 после изменения состояния порта действиями диспетчера.

4.6.7.3 *Процедура*

4.6.7.3.1 Процедуру выбора корневого моста (см. 4.6.8) следует использовать для выбора корневого назначенного порта и порта корневого моста, и для вычисления стоимости маршрута для этого моста.

4.6.7.3.2 Процедуру выбора назначенного порта (см. 4.6.9) следует использовать для того, чтобы для каждого порта определить, станет ли он назначенным портом для ЛВС, к которой он подключен.

4.6.8 Выбор корневого моста

4.6.8.1 *Назначение*

Эта процедура предназначена для выбора корневого назначенного моста и порта корневого моста, и вычисления стоимости маршрута к корневому мосту для этого моста.

4.6.8.2 *Использование*

Эта процедура используется процедурой изменения конфигурации (см. 4.6.7).

4.6.8.3 *Процедура*

4.6.8.3.1 Порт корневого моста устанавливается для идентификации порта среди портов, не являющихся назначенными портами в ЛВС, к которой они подключены, не находящихся в состоянии «неактивное» и имеющими параметр «корневой назначенный мост» с приоритетом, превышающим приоритет мостового идентификатора данного моста, который:

1) имеет наивысший приоритет относящегося к нему корневого моста, т. е. зарегистрированного как «корневой назначенный мост» для этого порта;
2) из двух и более портов с наивысшим приоритетом параметра «корневой назначенный мост» имеет более низкую стоимость маршрута к его корневому мосту, т. е. самую низкую сумму параметров «назначенная стоимость» и «стоимость маршрута» для любого порта, или
3) из двух или более портов с наивысшим приоритетом параметра «корневой назначенный мост» и наименьшим значением соответствующей стоимости маршрута к корневому мосту имеет наивысший приоритет параметра «идентификатор моста», хранимого как «назначенный мост» для ЛВС, к которой этот порт подключен, или
4) из двух или более портов с наивысшим приоритетом параметра «корневой назначенный мост», наименьшим значением соответствующей стоимости маршрута к корневому мосту и наивысшим приоритетом параметра «назначенный мост» имеет наивысший приоритет параметра «идентификатор порта», зарегистрированного как «назначенный порт» для ЛВС, к которой этот порт подключен, или
5) из двух или более портов с наивысшим приоритетом параметра «корневой назначенный мост» наименьшим значением соответствующей стоимости маршрута к корневому мосту и наивысшим приоритетом параметров «назначенный мост» и «назначенный порт» имеет наивысший приоритет параметра «идентификатор порта».

4.6.8.3.2 Если такого порта нет, значение параметра «порт корневого моста» устанавливается в ноль и

1) параметр «корневой назначенный мост», хранимый мостом, устанавливается в значение параметра «идентификатор моста», хранимого этим мостом, и
2) значение параметра «стоимость маршрута к корневому мосту», хранимого мостом, устанавливается в ноль.

4.6.8.3.3 В противном случае, т. е. если один из портов моста идентифицируется как «порт корневого моста», тогда:

1) параметр «корневой назначенный мост», хранимый мостом, устанавливается в значение параметра «корневой назначенный мост», хранимого портом корневого моста, и
2) значение параметра моста «стоимость маршрута к корневому мосту» устанавливается равным значению параметра «стоимость маршрута к корневому мосту», относящегося к порту корневого моста, т. е. сумме значений параметров «назначенная стоимость» и «стоиомсть маршрута» для порта корневого моста.

4.6.9 Выбор назначенного порта

4.6.9.1 *Назначение*

Для каждого порта эта процедура должна определить, должен ли данный порт стать назначенным портом для ЛВС, к которой он подключен.

4.6.9.2 *Использование*

Эта процедура используется как часть процедуры изменения конфигурации (см. 4.6.7).

4.6.9.3 *Процедура*

Процедура становления назначенного порта (см. 4.6.10) должна привлекаться для каждого порта, который:

4.6.9.3.1 уже выбран как назначенный порт для ЛВС, к которой он подключен, т. е. параметры порта «назначенный мост» и «назначенный порт» имеют те же значения, что и параметры «идентификатор моста» и «идентификатор порта» для этого порта соответственно, или для которого

4.6.9.3.2 параметр «корневой назначенный мост», зарегистрированный для моста, отличается от параметра, зарегистрированного для данного порта (заметим, что данная процедура следует за выбором корневого моста), или

4.6.9.3.3 мост обеспечивает маршрут к корневому мосту более низкой стоимости для ЛВС, к

которой подключен данный порт, т. е. стоимость маршрута к корневому мосту, зарегистрированная мостом, дешевле назначенной стоимости, зарегистрированной портом, или

4.6.9.3.4 мост обеспечивает маршрут к корневому мосту одинаковой стоимости и «идентификатор моста» указывает мост более высокого приоритета, чем мост, зарегистрированный как назначенный для данного порта, или

4.6.9.3.5 мост обеспечивает маршрут к корневому мосту одинаковой стоимости к корневому мосту и является назначенным для ЛВС, к которой подключен его порт, а «идентификатор порта» данного порта имеет более высокий приоритет, чем этот же параметр порта, зарегистрированного как назначенный.

4.6.10 Становление назначенного порта

4.6.10.1 *Назначение*

При условии, что порт является назначенным для ЛВС, к которой он подключен, эта процедура должна присвоить соответствующие значения тем параметрам порта, которые определяют активную топологию мостовой ЛВС.

4.6.10.2 *Использование*

Эта процедура используется:

4.6.10.2.1 после истечения тайм-аута времени существования сообщения для данного порта;

4.6.10.2.2 после выбора порта в качестве назначенного порта для ЛВС, к которой он подключен, процедурой выбора назначенного порта (см. 4.6.9) как часть процедуры изменения конфигурации (см. 4.6.7);

4.6.10.2.3 после изменения состояния порта действиями диспетчера (см. 4.8.1—4.8.3 и 4.8.5).

4.6.10.3 *Процедура*

4.6.10.3.1 Параметр порта «корневой назначенный мост» устанавливается мостом в значение параметра моста «корневой назначенный мост».

4.6.10.3.2 Параметр порта «назначенная стоимость» устанавливается мостом в значение параметра моста «стоимость маршрута к корневому мосту».

4.6.10.3.3 Параметр порта «назначенный мост» устанавливается в значение параметра моста «идентификатор моста».

4.6.10.3.4 Параметр порта «назначенный порт» устанавливается в значение параметра этого порта «идентификатор порта».

4.6.11 Выбор состояния порта

4.6.11.1 *Назначение*

Эта процедура должна определить состояние портов моста, основываясь на обновленной информации о конфигурации, которая указывает для каждого порта его участие в активной топологии мостовой ЛВС, т. е. должен ли он:

1) стать портом корневого моста для данного моста;

2) стать назначенным портом;

3) стать дублирующим портом в избыточно связанной мостовой ЛВС.

4.6.11.2 *Использование*

Эта процедура должна использоваться вслед за использованием процедуры изменения конфигурации и после:

4.6.11.2.1 получения ПБДМ «конфигурация», содержащего протокольную информацию, которая заменяет информацию, зарегистрированную портом;

4.6.11.2.2 истечения тайм-аута времени существования сообщения для порта, который побуждает данный порт стать назначенным для ЛВС, к которой он подключен;

4.6.11.2.3 изменения в состоянии порта, обусловленного действиями диспетчера.

4.6.11.3 *Процедура*

Для каждого порта моста:

4.6.11.3.1 если порт является портом корневого моста для моста, то:

1) флаги «торможение конфигурации» и «подтверждение изменения топологии» сбрасываются,

2) для этого порта используется процедура продвижения (см. 4.6.12);

4.6.11.3.2 если порт является назначенным для ЛВС, к которой он подключен, т. е. параметр порта «назначенный мост» такой же как параметр моста «идентификатор моста», а параметры порта «назначенный порт» и «идентификатор порта» одинаковы, и порт не находится в состоянии «неактивное», то:

1) прекращается отсчет тайм-аута порта «время существования сообщения», если он отсчитывался,

2) используется процедура продвижения (см. 4.6.12) для данного порта;

4.6.11.3.3 если порт должен стать резервным, т. е. ни портом корневого моста, ни назначенным портом, то:

1) флаги порта «торможение конфигурации» и «подтверждение изменения топологии» сбрасываются,

2) используется процедура блокирования (см. 4.6.13).

4.6.12 Выполнение продвижения

4.6.12.1 *Назначение*

Эта процедура должна позволить порту участвовать в маршрутизации кадров после истечения соответствующего интервала времени, который гарантирует, что временные шлейфы в мостовой ЛВС не приведут к дублированию кадров.

4.6.12.2 *Использование*

Эта процедура используется как часть процедуры выбора состояния порта (см. 4.6.11).

4.6.12.3 *Процедура*

Если порт находится в состоянии блокирования, то:

4.6.12.3.1 он переходит в состояние прослушивания;

4.6.12.3.2 Запускается тайм-аут порта «задержка продвижения».

4.6.13 Выполнение блокирования

4.6.13.1 *Назначение*

Эта процедура должна завершить участие порта в ретрансляции кадров.

4.6.13.2 *Использование*

Эта процедура используется как часть процедуры выбора состояния порта (см. 4.6.11).

4.6.13.3 *Процедура*

Если порт не находится в состоянии «неактивное» или «блокирование», то:

4.6.13.3.1 В случае нахождения порта в состоянии «продвижение» или «обучение» привлекается процедура «обнаружение изменения топологии» (4.6.14);

4.6.13.3.2 порт устанавливается в состояние блокирования;

4.6.13.3.3 прекращается отсчет тайм-аута порта «задержка продвижения».

4.6.14 Обнаружение изменения топологии

4.6.14.1 *Назначение*

Эта процедура должна зарегистрировать изменение топологии, обнаруженное или переданное мостом. Выполнить действия по уведомлению корневого моста о событии «обнаружено изменение топологии».

4.6.14.2 *Использование*

Эта процедура используется:

4.6.14.2.1 при получении ПБДМ «уведомление об изменении топологии» портом, назначенным для ЛВС, к которой он подключен;

4.6.14.2.2 при входе порта моста в состояние продвижения после истечения тайм-аута порта «время задержки продвижения» при условии, что мост является назначенным по меньшей мере для одной ЛВС, к которой подключены его порты;

4.6.14.2.3 при переходе порта моста из состояния «продвижение» или «обучение» в состояние «блокирование»;

4.6.14.2.4 когда мост становится корневым.

4.6.14.3 *Процедура*

4.6.14.3.1 Если мост определен как корневой, т. е. параметры моста «корневой назначенный мост» и «идентификатор моста» одинаковы, то:

1) устанавливается флаг моста «изменение топологии»;

2) запускается мостовой тайм-аут «изменение топологии».

4.6.14.3.2 Если мост не выбран как корневой и флаг моста «изменение топологии» уже сброшен, и мост является назначенным для ЛВС, к которой подключены его порты, то:

1) привлекается процедура передачи ПБДМ «уведомление об изменении топологии»;

3) запускается тайм-аут «сообщение об изменении топологии».

4.6.15 Уведомления об изменении топологии подтверждено

4.6.15.1 *Назначение*

Эта процедура должна завершить передачу ПБДМ «уведомление об изменении топологии».

4.6.15.2 *Использование*

Процедура используется при получении ПБДМ «конфигурация» с установленным флагом «подтверждение изменения топологии» из назначенного моста ЛВС, к которой подключен порт корневого моста.

4.6.15.3 *Процедура*

Сбрасывается флаг моста «обнаружение изменения топологии» и прекращается отсчет тайм-аута «уведомление об изменении топологии».

4.6.16 Подтверждение изменения топологии

4.6.16.1 *Назначение*

Эта процедура должна подтвердить уведомление об обнаруженном изменении топологии, переданное другим мостом.

4.6.16.2 *Использование*

Процедура используется после получения ПБДМ «уведомление об изменении топологии» портом, назначенным для ЛВС, к которой он подключен.

4.6.16.3 *Процедура*

Устанавливается флаг порта «подтвердить изменение топологии» и для порта используется процедура передачи ПБДМ «конфигурация» (см. 4.6.1).

4.7 **Операции протокола**

Логический объект мостового протокола должен:

1) взаимодействовать с его равноправными логическими объектами в других мостах путем передачи ПБДМ;

2) обновлять хранимые протокольные переменные и тайм-ауты;

3) изменять состояния портов моста после:

a) получения ПБДМ;

b) истечения тайм-аутов порта и моста в соответствии с определениями, приведенными ниже (см. 4.7.1—4.7.8). В случаях неоднозначности ссылки должны даваться на процедурную модель (см. 4.9), которая образует определяющее описание операций протокола.

Настоящая спецификация использует элементы протокольных процедур, описанные в 4.6, который в сочетании с операциями данного подраздела, параметрами и тайм-аутами протокола, определенными в 4.5, обеспечивает абстрактное описание алгоритма и протокола покрывающего дерева. Соответствие настоящей спецификации достигается через обеспечение параметров и тайм-аутов протокола и передачи ПБДМ в соответствии с изложенным выше. На реализацию не налагается никаких других ограничений, в частности, к отдельным элементам процедур требования соответствия не предъявляются.

4.7.1 Прием ПБДМ «конфигурация»

4.7.1.1 Если полученный ПБДМ «конфигурация» содержит протокольную информацию, которая заменяет информацию, уже хранимую для порта, как определено в 4.6.2.2, применяется следующая последовательность процедур:

4.7.1.1.1 процедура регистрации информации ПБДМ «конфигурация» для порта (см. 4.6.2);

4.7.1.1.2 процедура изменения конфигурации (см. 4.6.7);

4.7.1.1.3 процедура выбора состояния порта (см. 4.6.11);

4.7.1.1.4 если мост был определен как корневой до обновления конфигурации, но не дольше, прекращается отсчет тайм-аута заявки (см. 4.5.4.1);

4.7.1.1.5 если мост был определен как корневой до обновления конфигурации, но не дольше, и флаг «обнаружено изменение топологии» установлен, прекращается отсчет тайм-аута «изменение топологии», используется процедура передачи «уведомление об изменении топологии» (см. 4.6.6) и запускается тайм-аут «уведомление об изменении топологии»;

4.7.1.1.6 если ПБДМ «конфигурация» получен портом корневого моста (т. е. данный порт выбран процедурой обновления конфигурации как порт корневого моста), применяются процедуры записи значений тайм-аутов ПБДМ «конфигурация» (см. 4.6.3) и генерации ПБДМ «конфигурация» (см. 4.6.4);

4.7.1.1.7 если ПБДМ «конфигурация» получен портом корневого моста и флаг «подтверждено изменение топологии» установлен, применяется процедура подтверждения изменения топологии (см. 4.6.15).

4.7.1.2 Если полученный ПБДМ «конфигурация» не содержит информации, которая заменяет информацию, хранимую данным портом, и этот порт является назначенным для ЛВС, к которой он подключен, т. е. значение параметров порта «назначенный мост» и «назначенный порт» равны значениям параметров моста «идентификатор моста» и порта «идентификатор порта» соответственно, то используется процедура ответа на ПБДМ «конфигурация» (см. 4.6.5).

4.7.2 Прием ПБДМ «уведомление об изменении топологии»

Если порт, в который поступил ПБДМ «уведомление об изменении конфигурации», является назначенным для ЛВС, к которой он подключен, то:

4.7.2.1 используется процедура обнаружения изменения топологии (см. 4.6.14);

4.7.2.2 используется процедура «подтвердить изменение топологии» (см. 4.6.16).

4.7.3 Истечение тайм-аута заявки

Используется процедура генерации ПБДМ «конфигурация» (см. 4.6.4) и запускается тайм-аут заявки (см. 4.5.4.1).

4.7.4 Истечение тайм-аута времени существования сообщения

4.7.4.1 Для порта, для которого истек тайм-аут времени существования сообщения, используется процедура становления назначенного порта (см. 4.6.10), а также:

4.7.4.2 процедура изменения конфигурации (см. 4.6.7);

4.7.4.3 процедура выбора состояния порта (см. 4.6.11);

4.7.4.4 если мост выбран процедурой изменения конфигурации как корневой, то:

4.7.4.4.1 параметры моста «максимальное время существования», «время заявки» и «задержка продвижения» устанавливаются в значения параметров «мостовое максимальное время существования», «мостовое время заявки» и «мостовая задержка продвижения»;

4.7.4.4.2 используется процедура обнаружения изменения топологии (см. 4.6.14);

4.7.4.4.3 прекращается отсчет тайм-аута уведомления об изменении топологии (4.5.4.2);

4.7.4.4.4 используется процедура генерации ПБДМ «конфигурация» (см. 4.6.4) и запускается тайм-аут заявки.

4.7.5 Истечение тайм-аута задержки продвижения

4.7.5.1 Если порт, для которого истек тайм-аут задержки продвижения (см. 4.5.6.2), находился в состоянии «прослушивание», то:

4.7.5.1.1 состояние порта устанавливается в состояние «обучение»;

4.7.5.1.2 повторно запускается тайм-аут задержки продвижения.

4.7.5.2 Если порт, для которого истек тайм-аут задержки продвижения, находился в состоянии «обучение», то:

4.7.5.2.1 он переводится в состояние продвижения;

4.7.5.2.2 для моста, являющегося назначенным для более чем одной ЛВС, к которой подключены его порты, привлекается процедура «обнаружено изменение топологии» (см. 4.6.14).

4.7.6 Истечение тайм-аута уведомления об изменении топологии

Для порта, для которого истек тайм-аут уведомления об изменении топологии:

4.7.6.1 используется процедура передачи ПБДМ «уведомление об изменении топологии» (см. 4.6.6);

4.7.6.2 повторно запускается тайм-аут уведомления об изменении топологии (см. 4.5.4.2).

4.7.7 Истечение тайм-аута изменения топологии

4.7.7.1 Сбрасывается флаг моста «обнаружено изменение топологии».

4.7.8 Истечение тайм-аута удержания

Если флаг «торможение конфигурации» установлен для порта, для которого истек тайм-аут удержания (см. 4.5.6.3), то для этого порта привлекается процедура передачи ПБДМ «конфигурация» (см. 4.6.1).

4.8 **Диспетчер логического объекта мостового протокола**

Административное управление логическим объектом мостового протокола, выполняющего алгоритм и протокол покрывающего дерева, необходим для:

1) удовлетворения всем требованиям к локальной информации и услугам конфигурации;

2) поддержки операций удаленного диспетчера.

В этом подразделе определяются следующие действия диспетчера над параметрами и процедурами протокола и алгоритма покрывающего дерева:

a) инициализация;

b) активизация отдельного порта;

c) деактивизация отдельного порта;

d) изменение приоритетной части параметра «идентификатор моста»;

e) изменение приоритетной части параметра «идентификатор порта»;

f) изменение стоимости маршрута, относящейся к отдельному порту.

Такие операции должны модифицировать параметры и тайм-ауты протокола и передавать ПБДМ, как описано ниже (см. 4.8.1 — 4.8.6). На реализацию не налагается никаких других ограничений, в частности, к отдельным элементам процедур требования соответствия не предъявляются. В случаях неоднозначности ссылки должны даваться на процедурную модель (см. 4.9), которая образует определяющее описание этих операций.

В данном разделе не определяется, какие операции становятся доступны для станции удаленного диспетчера, каким образом они комбинируются и каким образом передаются. Операции и средства, обеспечиваемые для удаленного диспетчера, подробно описаны в разделе 6 «Диспетчер моста». Точно также в данном разделе не определяется доступность локальной информации и процедур конфигурации.

4.8.1 Инициализация

4.8.1.1 Параметр моста «корневой назначенный мост» устанавливается в значение параметра «идентификатор моста», а значение параметра моста «стоимость маршрута» устанавливается в ноль.

4.8.1.2 Параметры моста «максимальное время существования», «время заявки» и «задержка продвижения» устанавливаются в значения параметров «мостовое максимальное время существования», «мостовое время заявки» и «мостовая задержка продвижения» соответственно.

4.8.1.3 Флаги моста «обнаружено изменение топологии» и «изменение топологии» сбрасываются, и прекращается отсчет тайм-аутов «уведомление об изменении топологии» (см. 4.5.4.2) и «изменение топологии» (см. 4.5.4.3), если они отсчитывались.

4.8.1.4 Для каждого порта моста:

4.8.1.4.1 используется процедура становления назначенного порта (см. 4.6.10) для присвоения значений параметрам порта «корневой назначенный мост», «назначенная стоимость», «назначенный мост» и «назначенный порт»;

4.8.1.4.2 порт устанавливается в состояние «блокирование», если он активизирован вследствие инициализации; как вариант, порт устанавливается в состояние «неактивное»;

4.8.1.4.3 флаг «подтверждение изменения топологии» сбрасывается.

4.8.1.4.4 флаг «торможение конфигурация» сбрасывается;

4.8.1.4.5 прекращается отсчет тайм-аута времени существования сообщения (см. 4.5.6.1), если он отсчитывался;

4.8.1.4.6 прекращается отсчет тайм-аута задержки продвижения (см. 4.5.6.2), если он отсчитывался;

4.8.1.4.7 прекращается отсчет тайм-аута удержания (см. 4.5.6.3), если он отсчитывался.

4.8.1.5 Для выбора состояния каждого порта моста используется процедура «выбор состояния порта» (см. 4.6.11).

4.8.1.6 Для генерации ПБДМ «конфигурация» привлекается процедура генерации ПБДМ «конфигурация» (см. 4.6.4) и запускается тайм-аут заявки (см. 4.5.4.1).

4.8.2 Активизация порта

4.8.2.1 Процедура становления назначенного порта (см. 4.6.10) используется для присвоения значений параметрам порта «корневой назначенный мост», «назначенная стоимость», «назначенный мост» и «назначенный порт».

4.8.2.2 Порт устанавливается в состояние «блокирование».

4.8.2.3 Сбрасывается флаг «подтвердить изменение топологии».

4.8.2.4 Сбрасывается флаг «торможение конфигурация».

4.8.2.5 Прекращается отсчет тайм-аута времени существования сообщения (см. 4.5.6.1), если он отсчитывался.

4.8.2.6 Прекращается отсчет тайм-аута задержки продвижения (см. 4.5.6.2), если он отсчитывался.

4.8.2.7 Прекращается отсчет тайм-аута удержания (см. 4.5.6.3), если он отсчитывался.

4.8.2.8 Используется процедура выбора состояния порта (см. 4.6.11).

4.8.3 Деактивизация порта

4.8.3.1 Для присвоения значений параметрам порта «корневой назначенный мост», «назначенная стоимость», «назначенный мост» и «назначенный порт» используется процедура становления назначенного порта (см. 4.6.10).

4.8.3.2 Порт устанавливается в состояние «неактивное».

4.8.3.3 Флаг «подтвердить изменение топологии» сбрасывается.

4.8.3.4 Флаг «торможение конфигурации» сбрасывается.

4.8.3.5 Прекращается отсчет тайм-аута времени существования сообщения (см. 4.5.6.1), если он отсчитывался.

4.8.3.6 Прекращается отсчет тайм-аута задержки продвижения (см. 4.5.6.2), если он отсчитывался.

4.8.3.7 Используется процедура изменения конфигурации (см. 4.6.7).

4.8.3.8 Используется процедура выбора состояния порта (см. 4.6.11).

4.8.3.9 Если мост выбран корневым в результате изменения конфигурации, то:

4.8.3.9.1 параметры моста «максимальное время существования сообщения», «время заявки» и «задержка продвижения» устанавливаются в значения параметров «мостовое максимальное время существования», «мостовое время заявки» и «мостовая задержка продвижения» соответственно;

4.8.3.9.2 используется процедура обнаружения изменения топологии (см. 4.6.14);

4.8.3.9.3 прекращается отсчет тайм-аута уведомления об изменении конфигурации (см. 4.5.4.2);

4.8.3.9.4 используется процедура генерации ПБДМ «конфигурация» (см. 4.6.4) и запускается тайм-аут заявки.

4.8.4 Установка приоритета моста

4.8.4.1 Вычисляется новое значение идентификатора моста.

4.8.4.2 Значение параметра «назначенный мост», хранимое каждым портом, который выбирается как назначенный порт для ЛВС, к которой он подключен (т. е. для тех портов, у которых значения параметров «назначенный мост» и «назначенный порт» равны значениям «идентификатор моста» и «идентификатор порта» этого порта соответственно), устанавливается в новое значение «идентификатор моста».

4.8.4.3 Параметр «идентификатор моста», хранимый мостом, устанавливается в новое значение.

4.8.4.4 Используется процедура изменения конфигурации (см. 4.6.7).

4.8.4.5 Используется процедура выбора состояния порта.

4.8.4.6 Если мост был выбран в качестве корневого, то:

4.8.4.6.1 параметры «максимальное время существования», «время заявки» и «задержка продвижения» устанавливаются в значения параметров «мостовое максимальное время существования», «мостовое время заявки» и «мостовая задержка продвижения» соответственно;

4.8.4.6.2 используется процедура обнаружения изменения топологии (см. 4.6.14);

4.8.4.6.3 прекращается отсчет тайм-аута уведомления об изменении топологии (см. 4.5.4.2);

4.8.4.6.4 используется процедура генерации ПБДМ «конфигурация» (4.6.4) и запускается тайм-аут заявки.

4.8.5 Установка приоритета порта

4.8.5.1 Вычисляется новое значение идентификатора порта.

4.8.5.2 Если порт был выбран в качестве назначенного для ЛВС, к которой он подключен (т. е. значение параметров «мост назначения» и «назначенный порт» равны значениям параметров «идентификатор моста» и «идентификатор порта» соответственно), то параметр «назначенный порт», хранимый портом, устанавливается в новое значение параметра «идентификатор порта».

4.8.5.3 Параметр «идентификатор порта», хранимый портом, устанавливается в новое значение.

4.8.5.4 Если значение параметра «назначенный мост», хранимого портом, равно значению параметра «идентификатор моста», хранимого мостом, и новое значение идентификатора порта имеет более высокий приоритет, чем зарегистрированное для параметра «назначенный порт», то:

4.8.5.4.1 для присвоения значений параметрам порта «корневой назначенный мост», «назначенная стоимость», «назначенный мост» и «назначенный порт» используется процедура становления порта назначения (см. 4.6.10);

4.8.5.4.2 используется процедура выбора состояния порта (см. 4.6.11).

4.8.6 Установка стоимости маршрута

4.8.6.1 Параметр «стоимость маршрута» для порта устанавливается в новое значение.

4.8.6.2 Используется процедура изменения конфигурации (см. 4.6.7).

4.8.6.3 Используется процедура выбора состояния порта.

4.9 **Модель процедур**

Данный раздел содержит определительное описание операций алгоритма и протокола связующего дерева. Текст на естественном языке в подразделах 4.6—4.8 данного стандарта предназначен для неформального представления определяемой здесь семантики операций. При обнаружении различия в интерпретации между текстом на естественном языке и моделью процедур предпочтение следует отдать последней.

4.9.1 Общие сведения

Параметры, тайм-ауты, элементы процедур и операции протокола представлены ниже в виде компилируемой программы на языке программирования Си (ANSI X3.159).

Целью представления этой программы является точное и недвусмысленное определение операций алгоритма и протокола. Описание операций протокола на языке программирования никоим образом не ограничивает способы реализации протокола; реальная реализация может использовать любую подходящую технологию.

Соответствие оборудования настоящему стандарту рассматривается только в отношении наблюдаемости протокола. Программа, приведенная в этом разделе, содержит подробные элементы модели, которые являются локальным вопросом для реализации; в отношении этих элементов не устанавливаются требования соответствия.

Текст на естественном языке в подразделах 4.6—4.8 соответствует тексту на машинном языке, представленному в данном подразделе. С целью сохранения компактности текста программы все комментарии даны путем ссылок на текст естественного языка в подразделах 4.6—4.8 и имеют форму 4.n.n.n . В случаях, когда оператор программы привлекает элемент процедуры, следует ссылка на конкретные условия из перечисленных для данной процедуры, которые обусловили привлечение этой процедуры.

```
/* * * *
```

```
#define No_of_ports 2
        /* произвольный выбор, чтобы обеспечить компиляцию приводимого ниже кода */

#define All_ports No_of_ports+1
        /* порты начинаются с 1, а массивы в с—с 0 */

#define Default_path_cost 10
        /* произвольное */

#define Message_age_increment 1
        /* минимально возможное приращение для исключения заниженной оценки времени
        существования, допустимого для времени передачи ПБДМ */

#define No_port 0
        /* зарезервированное значение для параметра порта «корневой мост», указывающего порт
        некорневого моста и используемого когда мост становится корневым */

/* * * *
```

```
        Time hello_time;                        /* (4.5.1.7) */
        Time forward_delay;                     /* (4.5.1.8) */
        Flag topology_change_acknowledgment;    /* (4.5.1.9) */
        Flag topology_change;                   /* (4.5.1.10) */

} Config_bpdu;

/** Параметры ПБДМ «уведомление об изменении топологии» (4.5.2) **/

typedef struct
{
        Bpdu_type type;
} Tcn bpdu;
/** Параметры моста (4.5.3) **/
typedef struct
{
        Identifier designated_root;             /* (4.5.3.1) */
        Cost root_path_cost;                    /* (4.5.3.2) */
        Int root_port;                          /* (4.5.3.3) */
        Time max_age;                           /* (4.5.3.4) */
        Time hello-time;                        /* (4.5.3.5) */
        Time forward_delay;                     /* (4.5.3.6) */
        Identifier bridge_id;                   /* (4.5.3.7) */
        Time bridge_max_age;                    /* (4.5.3.8) */
        Time bridge_hello_time;                 /* (4.5.3.9) */
        Time bridge forward_delay;              /* (4.5.3.10) */
        Boolean topology_change_detected;       /* (4.5.3.11) */
        Boolean topology_change;                /* (4.5.3.12) */
        Time topology_change_time;              /* (4.5.3.13) */
        Time hold_time;                         /* (4.5.3.14) */
} Bridge data;
/** Параметры порта (4.5.5) **/
typedef struct
{
        Port_id port_id;                        /* (4.5.5.1) */
        State state;                            /* (4.5.5.2) */
        Int path_cost;                          /* (4.5.5.3) */
        Identifier designated_root;             /* (4.5.5.4) */
        Int designated_cost;                    /* (4.5.5.5) */
        Identifier designated_bridge;           /* (4.5.5.6) */
        Port_id designated_port;                /* (4.5.5.7) */
        Boolean topology_change_acknowledge;    /* (4.5.5.8) */
        Boolean config_pending;                 /* (4.5.5.9) */
} Port data;
/** типы для поддержки тайм-аутов данной псевдореализации. **/
typedef struct
{
        Boolean active; /* используемый тайм-аут. */
        Time value;  /* текущее значение отсчитывающего тайм-аута. */
        } Timer;
```

```
/* * * *
```

```
    start_hold_timer (port_no);                                    /* (4.6.1.3.2(11)) */
}
}
    /* где
    send_config_bpdu(port_no, bpdu)
    Int port_no;
    Config_bpdu *bpdu;
    — конкретная программа псевдореализации, которая передает пбдм в конкретный порт в течение
    заданного времени.
    */
    /* и */
    Boolean root_bridge()
    {
    return(bridge_info.designated_root == bridge_info.bridge_id);
    }
Boolean_supersedes_port_info(port_no, config)                      /* (4.6.2.2) */
Int port_no;
Config_bpdu *config;
{
    return (
    ( config—>root_id
      < port_info[port_no].designated_root                         /* (4.6.2.2.1) */
    )
    | |
    ( ( config—>root_id
        == port_info[port_no].designated_root
      )
      &&
      ( ( config—>root_path_cost
          < port_info[port_no].designated_cost                     /* (4.6.2.2.2) */
        )
        | |
        ( ( config—>root_path_cost
            == port_info[port_no).designated_cost
          )
          &&
          ( ( config—>bridge_id
              < port_info[port_no].designated_bridge               /* (4.6.2.2.3) */
            )
            | |
            ( ( config—>bridge_id
                = = port_info[port_no].designated_bridge           /* (4.6.2.2.4) */
              )
              &&
              ( ( config—>bridge_id ! = bridge_info.bridge_id      /* (4.6.2.2.4(1)) */
                )
                | |
                (   config—>port_id
                    <= port_info[port_no].designated_port          /* (4.6.2.2.4(2)) */
                )
    ) ) ) ) ) ) )
          );
}
record_config_information(port_no, config)
Int port_no;
```

```
Config_bpdu *config;
{
    port_info[port_no).designated_root = config—>root_id;                    /* (4.6.2.3.1) */
    port_info[port_no].designated_cost = config—>root_path_cost;
    port_info[port_no].designated_bridge = config—>bridge_id;
    port_info[port_no].designated_port = config—>port_id;
    start_message_age_timer (port_no, config—>message_age);                  /* (4.6.2.3.2) */
}
record_config_timeout_values (config)                                        /* (4.6.3)     */
Config_bpdu *config;
{
    bridge_info.max_age = config—>max_age;                                   /* (4.6.3.3) */
    bridge_info.hello_time = config—>hello_time;
    bridge_info.forward_delay = config—>forward_delay;
    bridge_info.topology_change = config—>topology_change;
}
config_bpdu_generation( )                                                    /* (4.6.4) */
{
    Int port_no;
    for (port_no = One; port_no <—No_of_ports; port_no++)                    /* (4.6.4.3) */
    {
        if ( designated_port(port_no)                                        /* (4.6.4.3) */
             &&
             (port_info[port_no].state ! = Disabled)
           )
           {
             transmit_config(port_no);                                       /* (4.6.4.3) */
           }                                                                 /* (4.6.1.2) */
    }
}
             /* где */
             Boolean_designated_port(port_no)
             Int_port_no;
             {
                return ( ( port_info[port_no].designated_bridge = = bridge_info.bridge_id
                           )
                           &&
                           (port_info[port_no].designated_port = = port_info[port_no).port_id
                           )
                       );
             }
reply(port_no)                                                               /* (4.6.5) */
Int port_no;
{
    transmit_config(port_no);                                                /* (4.6.5.3) */
}
transmit_tcn ( )                                                             /* (4.6.6) */
(
    Int port_no;
    port_no = bridge_info.root_port;
    tcn_bpdu[port_no).type = Tcn_bpdu_type;
    send_tcn_bpdu(port_no, &tcn_bpdu[bridge_info.root_port]);                /* (4.6.6.3) */
}
      /* where
      send_tcn_bpdu(port_no, bpdu)
```

```
    Int port_no;
    Tcn_bpdu *bpdu;
    — конкретная программа псевдореализации, которая передает пбдм в конкретный порт в тече-
      ние заданного времени.
    */
configuration_update()                                          /* (4.6.7) */
{
  root_selection();                                             /* (4.6.7.3.1) */
                                                                /* (4.6.8.2) */
  designated_port_selection();                                  /* (4.6.7.3.2) */
                                                                /* (4.6.9.2) */
}
root_selection()                                                /* (4.6.8) */
{
  Int root_port;
  Int port_no;
  root_port = No_port;
  for (port_no = One; port_no <— No of_ports; port_no++)        /* (4.6.8.3.1) */
  {
    if ( ( (designated_port(port_no))
           &&
           (port_info[port_no].state ! = Disabled)
           &&
           (port_info(port_no].designated_root
             <bridge_info.bridge_id)
         )
         &&
         ( (root_port = = No_port)
         | |
         ( port_info[port_no].designated_root
           < port_info[root_port].designated_root               /* (4.6.8.3.1(1)) */
         )
         | |
         ( ( port_info(port_no).designated_root
             = = port_info[root_port].designated_root
           )
           &&
           ( ( ( port_info[port_no].designated_cost
                 + port_info[port_no].path_cost
               )
               <
               ( port_info[root_port].designated_cost
                 + port_info[root_port].path_cost               /* (4.6.8.3.1(2)) */
               )
             )
             | |
             ( ( ( port_info[port_no].designated_cost
                   + port_info(port_no).pnth_cost
                 )
                 = =
                 ( port_info[root_port].designated_cost
                   + port_info[root_port].path_cost
                 )
               )
               &&
               ( ( port_info[port_no].designated_bridge
                   <port_info[root_port].designated_bridge      /* (4.6.8.3.1(3)) */
```

```
                    )
                    | |
                    ( ( port_info[port_no].designated_bridge
                        = = port_info[root_port].designated_bridge
                      )
                      &&
                      ( ( port_info[port_no].designated_port
                          < port_info[root_port].designated_port          /* (4.6.8.3.1(4)) */
                        )
                        | |
                        ( ( port_info[port_no].designated_port
                            = = port_info[root_port].designated_port
                          )
                          &&
                          ( port_info[port_no].port_id
                            <port_info[root_port].port_id                 /* (4.6.8.3.1(5)) */
                          )
      ) ) ) ) ) ) ) )
    {
      root_port = port_no;
    }
  }
  bridge_info.root_port = root_port;                                      /* (4.6.8.3.1) */
  if (root_port = = No_port)                                              /* (4.6.8.3.2) */
  {
    bridge_info.designated_root = bridge_info.bridge_id;                  /* (4.6.8.3.2(1)) */
    bridge_info.root_path_cost = Zero;                                    /* (4.6.8.3.2(2)) */
  }
  else                                                                    /* (4.6.8.3.3) */
  {
    bridge_info.designated_root
    = port_info[root_port].designated_root;                               /* (4.6.8.3.3(1)) */
    bridge_info.root_path_cost = (port_info[root_port].designated_cost
    + port_info[root_port].path_cost                                      /* (4.6.8.3.3(2)) */
                                  );
  }
}
designated_port_selection( )                                              /* (4.6.9) */
{
  Int port_no;
  for (port_no = One; port_no <= No_of_ports; port_no++)                  /* (4.6.9.3) */
                                                                          /* (4.6.9.3.1) */
  {
    if ( designated_port(port_no)
         | |
         (
           port_info[port_no].designated_root
           ! = bridge_info.designated_root                                /* (4.6.9.3.2) */
         )
         | |
         ( bridge_info.root_path_cost
           <port_info[port_no].designated_cost                            /* (4.6.9.3.3) */
```

```
            )
            | |
            ( ( bridge_info.root_path_cost
                = = port_info[port_no].designated_cost
              )
              &&
              ( ( bridge_info.bridge_id
                  = = port_info[port_no].designated_bridge                    /* (4.6.9.3.4) */
                )
                | |
                ( ( bridge_info.bridge_id
                    = = port_info[port_no].designated_bridge
                  )
                  &&
                  ( port_info.[port_no].port_id
                    <= port_info[port_no].designated_port                     /* (4.6.9.3.5) */
                  )
        ) ) ) )
    {
        become_designated_port(port_no);
    }
    }
}
become_designated_port(port_no)
Int port_no;
{
    port_info[port_no].designated_root
    = bridge_info.designated_root;                                            /* (4.6.10.3.1) */
    port_info[port_no].designated_cost
    = bridge_info.root_path_cost;                                             /* (4.6.10.3.2) */
    port_info[port_no].designated_bridge
    = bridge_info.bridge_id;                                                  /* (4.6.10.3.3) */
    port_info[port_no].designated_port
    = port_info[port_no].port_id;                                             /* (4.6.10.3.4) */
}
port_state_selection( )                                                       /* (4.6.11) */
{
    Int port_no;
    for (port_no = One; port_no <= No_of_ports; port_no ++)
    {
        if (port_no = = bridge_info.root_port)                                /* (4.6.11.3.1) */
        {
            port_info[port_no].config_pending = False;                        /* (4.6.11.3.1(1)) */
            port_info[port_no].topology_change_acknowledge = False;
            make_forwarding (port_no);                                        /* (4.6.11.3.1(2)) */
        }
        else if (designated_port (port_no))                                   /* (4.6.11.3.2) */
        {
            stop_message_age_timer (port_no);                                 /* (4.6.11.3.2(1)) */
            make_forwarding (port_no);                                        /* (4.6.11.3.2(2)) */
        }
        else                                                                  /* (4.6.11.3.3) */
```

```
        {
            port_info[port_no].config_pending = False;                    /* (4.6.11.3.3(1)) */
            port_info[port_no].tapology_change_acknowledge = False;
            make_blocking(port_no);                                       /* (4.6.11.3.3(2)) */
        }
    }
}
make_forwarding(port_no)                                                  /* (4.6.12) */
Int port_no;
{
    if (port_info[port_no].state = = Blocking)                            /* (4.6.12.3) */
    {
        set_port_state(port_no, Listening);                               /* (4.6.12.3.1) */
        start_forward_delay_timer(port_no);                               /* (4.6.12.3.2) */
    }
}
make_blocking(port_no)                                                    /* (4.6.13) */
Int port_no;
{
    if ( (port_info[port_no].state ! = Disabled)
         &&
         (port_info[port_no].state ! = Blocking)                          /* (4.6.13.3) */
       )
    {
        if ( (port_info[port_no].state = = Forwarding)
             | |
             (port_info[port_no].state = = Learning)
           )
        {
            topology_change_detection();                                  /* (4.6.13.3.1) */
                                                                          /* (4.6.14.2.3) */
        }
        set_port_state(port_no, Blocking);                                /* (4.6.13.3.2) */
        stop_torward_delay_timer(port_no);                                /* (4.6.13.3.3) */
    }
}
            /* где */
            set_port_state(port_no, state)
            Int port_no;
            State state;
            {
                port_info[port_no].state = state;
            }
topology_change_detection()                                               /* (4.6.14) */
{
    if (root_bridge())                                                    /* (4.6.14.3.1) */
    {
        bridge_info.topology_change = True;                               /* (4.6.14.3.1(1)) */
        start topology change timer ();                                   /* (4.6.14.3.1(2)) */
    }
    else if (bridge_info.topology_change_detected = = False)              /* (4.6.14.3.2) */
    {
        transmit_tcn();                                                   /* (4.6.14.3.2(1)) */
        start_tcn_timer();                                                /* (4.6.14.3.2(2)) */
    }
```

```
    bridge_info.topology_change_detected = True;                    /* (4.6.14.3.3) */
}
topology_change_acknowledged()                                      /* (4.6.15) */
{
    bridge_info.topology_change_detected = False;                   /* (4.6.15.3.1) */
    stop_tcn_timer();                                               /* (4.6.15.3.2) */
}
acknowledge_topology_change(port_no)                                /* (4.6.16) */
Int port_no;
{
    port_info(port_no].topology_change_acknowledge = True;          /* (4.6.16.3.1) */
    transmit_config(port_no);                                       /* (4.6.16.3.2) */
}
/** Операции протокола (4.7) **/
received_config_bpdu(port_no, config)                               /* (4.7.1) */
Int port_no;
Config_bpdu *config;
{
    Boolean_root;
    root = root_bridge();
    if (port_info[port_no].state ! = Disabled)
    {
      if (supersedes_port_info(port_no, config))                    /* (4.7.1.1) */
                                                                    /* (4.6.2.2) */
      {
        record_config_information(port_no, config);                 /* (4.7.1.1.1) */
                                                                    /* (4.6.2.2) */
        configuration_update();                                     /* (4.7.1.1.2) */
                                                                    /* (4.6.7.2.1) */
        port_state_selection();                                     /* (4.7.1.1.3) */
                                                                    /* (4.6.11.2.1) */
        if ((!root_bridge()) && root)                               /* (4.7.1.1.4) */
        {
          stop_hello_timer();
          if (bridge_info.topology_change_detected)                 /* (4.7.1.1.5) */
          {
            stop_topology_change_timer();
            transmit_tcn();                                         /* (4.6.6.1) */
            start_tcn_timer();
          }
        }
        if (port_no = = bridge_info.root_port)
        {
          record_config_timeout_values(config);                     /* (4.7.1.1.6) */
                                                                    /* (4.6.3.2) */
          config_bpdu_generation();                                 /* (4.6.4.2.1) */
          if (config —> topology_change_acknowledgment)             /* (4.7.1.1.7) */
          {
            topology_change_acknowledged();                         /* (4.6.15.2) */
          }
        }
      }
      else if (designated_port(port_no))                            /* (4.7.1.2) */
      {
        reply(port_no);                                             /* (4.7.1.2.1) */
                                                                    / * (4.6.5.2) */
      }
```

```
    }
}
received_tcn_bpdu(port_no, tcn)                                    /* (4.7.2) */
Int port_no;
Tcn_bpdu *tcn;
{
    if (port_info[port_no].state ! = Disabled)
    {
        if (designated_port(port_no))
        {
            topology_change_detection();                           /* (4.7.2.1) */
                                                                   /* (4.6.14.2.1) */
            acknowledge_topology_change(port_no);                  /* (4.7.2.2) */
                                                                   /* (4.6.16.2) */
        }
    }
}
    hello_timer_expiry( )                                          /* (4.7.3) */
    {
        config_bpdu_generation();                                  /* (4.6.4.2.2) */
        start_hello_timer();
    }
    message_age_timer_expiry(port no)                              /* (4.7.4) */
    Int port_no;
    {
        Boolean_root;
        root = root_bridge();
        become_designated_port(port_no);                           /* (4.7.4.1) */
                                                                   /* (4.6.10.2.1) */
        configuration_update( );                                   /* (4.7.4.2) */
                                                                   /* (4.6.7.2.2) */
        port_state_selection( );                                   /* (4.7.4.3) */
                                                                   /* (4.6.11.2.2) */
        if ((root_bridge()) && (!root))                            /* (4.7.4.4) */
        {
            bridge_info.max_age = bridge_info.bridge_max_age;      /* (4.7.4.4.1) */
            bridge_info.hello_time = bridge_info.bridge_hello_time;
            bridge_info.forward_delay = bridge_info.bridge_forward_delay;
            topology_change_detection();                           /* (4.7.4.4.2) */
                                                                   /* (4.6.14.2.4) */
            stop_tcn_timer( );                                     /* (4.7.4.4.3) */
            config_bpdu_generation();                              /* (4.7.4.4.4) */
                                                                   /* (4.6.4.4.3) */
            start_hello_timer();
        }
    }
forward_delay_timer_expiry(port_no)                                /* (4.7.5) */
Int port_no;
{
    if (port_info[port_no].state = = Listening)                    /* (4.7.5.1) */
    {
        set_port_state(port_no, Learning);                         /* (4.7.5.1.1) */
        start_forward_delay_timer(port_no);                        /* (4.7.5.1.2) */
    }
    else if (port_info[port_no].state = = Learning)                /* (4.7.5.2) */
```

```
    {
        set_port_state(port_no, Forwnrding);                    /* (4.7.5.2.1) */
        if (designated_for_some_port())                         /* (4.7.5.2.2) */
        {
            topology_change_detection();                        /* (4.6.14.2.2) */
        }
    }
}
        /* где */
        Boolean_designated_for_some_port()
        {
            Int port_no;
            for (port_no = One; port_no <= No_of_ports; port_no++)
            {
                if ( port_info[port_no].designated_bridge
                    = = bridge_info.bridge_id
                )
                {
                    return(True);
                }
            }
            return(False);
        }
tcn_timer_expiry()                                              /* (4.7.6) */
{
    transmit_tcn();                                             /* (4.7.6.1) */
    start_tcn_timer();                                          /* (4.7.6.2) */
}
topology_change_timer_expiry()                                  /* (4.7.7) */
{
    bridge_info.topology_change_detected = False;               /* (4.7.7.1) */
    bridge_info.topology_change = False;                        /* (4.7.7.2) */
}
hold_timer_expiry(port_no)                                      /* (4.7.8) */
Int port_no;
{
    if (port_info(port_no].config_pending)
    {
        transmit_config(port_no);                               /* (4.7.8.1) */
    }                                                           /* (4.6.1.2.3) */
}
/** Диспетчер логического объекта мостового протокола          (4.8) **/
initialisation()                                                /* (4.8.1) */
{
    Int port_no;
    bridge_info.designated_root = bridge_info.bridge_id;        /* (4.8.1.1) */
    bridge_info.root_path_cost = Zero;
    bridge_info.root_port = No_port;
    bridge_info.max_age = bridge_info.bridge_max_age;           /* (4.8.1.2) */
    bridge_info.hello_time = bridge_info.bridge_hello_time;
    bridge_info.forward_delay = bridge_info.bridge_forward_delay;
    bridge_info.topology_change_detected = False;               /* (4.8.1.3) */
    bridge_info.topology_change = False;
    stop_tcn_timer();
    stop_topology_change_timer();
    for (port_no = One; port_no <= No_of_ports; port_no++)      /* (4.8.1.4) */
```

```
    {
        initialize_port(port_no);
    }
    port_state_selection( );                                    /* (4.8.1.5) */
    config_bpdu_generation();                                   /* (4.8.1.6) */
    start_hello_timer( );
    }
                                                                /*
                                                                */
    initialize_port(port_no)
    Int port_no;
    {
        become_designated_port(port_no);                        /* (4.8.1.4.1) */
        set_port_state(port_no, Blocking);                      /* (4.8.1.4.2) */
        port_info[port_no].topology_change_acknowledge = False; /* (4.8.1.4.3) */
        port_info[port_no].config_pending = False;              /* (4.8.1.4.4) */
        stop_message_age_timer(port_no);                        /* (4.8.1.4.5) */
        stop_forward_delay_timer(port_no);                      /* (4.8.1.4.6) */
        stop_hold_timer(port_no);                               /* (4.8.1.4.7) */
    }
enable_port(port_no)                                            /* (4.8.2) */
Int port_no;
{
    initialize_port(port_no);
    port_state_selection();                                     /* (4.8.2.7) */
}
                                                                /*
                                                                */
disable_port(port_no)                                           /* (4.8.3) */
Int port no;
{
    Boolean_root;
    root = root_bridge();
    become_designated_port(port_no);                            /* (4.8.3.1) */
    set_port_state(port_no, Disabled);                          /* (4.8.3.2) */
    port_info[port_no].topology_change_acknowledge = False;     /* (4.8.3.3) */
    port_info[port_no].config_pending = False;                  /* (4.8.3.4) */
    stop_message_age_timer(port_no);                            /* (4.8.3.5) */
    stop_forward_delay_timer(port_no);                          /* (4.8.3.6) */
    configuration_update();
    port_state_selection();                                     /* (4.8.3.7) */
    if  ((root_bridge()) && (!root))                            /* (4.8.3.8) */
    {
        bridge_info.max_age = bridge_info.bridge_max_age;       /* (4.8.3.8.1) */
        bridge_info.hello_time = bridge_info.bridge_hello_time;
        bridge_info.forward_delay = bridge_info.bridge_forward_delay;
        topology_change_detection( );                           /* (4.8.3.8.2) */
        stop_tch_timer( );                                      /* (4.8.3.8.3) */
        config_bpdu_generation( );                              /* (4.8.3.8.4) */
        start_hello_timer( );
    }
}
set_bridge_priority(new_bridge_id)                              /* (4.8.4) */
Identifier new_bridge_id;                                       /* (4.8.4.1) */
{
    Boolean root;
    Int port_no;
    root = root_bridge();
    for (port_no = One; port_no <= No_of_ports; port_no++)      /* (4.8.4.2) */
```

```
    {
        if (designated_port(port_no)
        {
          port_info[port_no].designated_port = new_bridge_id;
        }
    }
    bridge_info.bridge_id = new_bridge_id;                          /* (4.8.4.3) */
    configuration_update();                                         /* (4.8.4.4) */
    port_state_selection();                                         /* (4.8.4.5) */
    if  ((root_bridge()) && (!root))                                /* (4.8.4.6) */
    {
        bridge_info.max_age = bridge_info.bridge_max_age;           /* (4.8.4.6.1) */
        bridge_info.hello_timer = bridge_info.bridge_hello_time;
        bridge_info.forward_delay = bridge_info.bridge_forward_delay;
        topology_change_detection( );                               /* (4.8.4.6.2) */
        stop_tch_time();                                            /* (4.8.4.6.3) */
        config_bpdu_generation();                                   /* (4.8.4.6.4) */
        start_hello_timer();
    }
}
set_port_priority(port_no, new_port_id)                             /* (4.8.5) */
Int port_no;
Port_id new_port_id;                                                /* (4.8.5.1)
{
    if (designaten_port(port_no))                                   /* (4.8.5.2) */
    {
        port_info[port_no].designated_port = new_port_id;
    }
    port_info[port_no].port_id =  new_port_id;                      /* (4.8.5.3) */
    if ( (  bridge_info.bridge_id
            = = port_info[port_no].designated_bridge                /* (4.8.5.4) */
        )
        &&
        ( port_info[port_no].port_id
          <port_info[port_no].designated_port
        )
       )
    {
        become_designated_port(port_no);                            /* (4.8.5.4.1) */
        port_state_selection ();                                    /* (4.8.5.4.2) */
    }
}
                                                                        /*
                                                                        */
set_path_cost(port_no, path_cost)                                   /* (4.8.6) */
Int_port_no;
Cost_path_cost;
{
    port_info[port_no].path_cost = path_cost;                       /* (4.8.6.1) */
    configuration_update();                                         /* (4.8.6.2) */
    port_state_selection ();                                         /* (4.8.6.3) */
}
```

```
/** поддержка отсчета тайм-аута конкретной псевдореализации. **/
tick()
{
    Int_port_no;
    if (hello_timer_expired())
    { hello_timer_expiry();
    }
    if (tcn_timer_expired())
    { tcn_timer_expiry();
    }
    if (topology_change_timer_expired())
    { topology_change_timer_expiry();
    }
    for (port_no = One; port_no <= No_of_ports; port_no++)
    {
        if (forward_delay_timer_expired(port_no))
        {
          forward_delay_timer_expiry(port_no);
        }
        if (message_age_timer_expired(port_no))
        {
          message_age_timer_expiry(port_no);
        }
        if (hold_timer_expired(port_no))
        {
          hold_timer_expiry(port_no);
        }
    }
}
/* где */
start_hello_timer()
{   hello_timer.value = (Time) Zero;
    hello_tirner.active = True;
}
stop_hello_timer()
{     hello_timer.active = False;
}
Boolean hello_timer_expired()
(   if (hello_timer.active && (++hello_timer.value >= bridge_info.hello_time))
    {   hello_timer.active = False;
        return(True);
    }
    return(False);
}
start_tcn_timer()
{   tcn_timer.vnlue = (Time) Zero;
    tcn_timer.active = True;
}
stop_tcn_timer()
{   tcn_timer.active = False;
}
Boolean tcn_timer_expired()
(   if (tcn_timer.active && (++tcn_timer.value >= bridge_info.bridge_hello_time))
    {   tcn_timer.active = False;
        return(True);
```

```
    }
    return(False);
}
start_topology_change_timer()
{  topology_change_timer.value = (Time) Zero;
    topology_change_timer.active = True;
}
stop_topology_change_timer()
{  topology_change_timer.active = False;
}
Boolean topology_change_timer_expired()
(  if ( topology change timer.active
        && ( ++topology_change_timer.value >= bridge_info.topology_change_time
            )
      )
    { topology_change_timer.active = False;
      return(True);
    }
    return(False);
}
start_message_age_timer(port_no, message_age)
Int port_no;
Time message age;
{  message_age_timer[port_no].value = message_age;
    message_age_timer[port_no].active = True;
}
stop_message_age_timer(port_no)
Int port_no;
{  message_age_timer[port_no].active = False;
}
Boolean_message_age_timer_expired(port_no)
Int port_no;
{  if (message_age_timer[port_no].active &&
      (++message_age_timer[port_no].value >= bridge_info.max_age))
    { message_age_timer[port_no].active = False;
      return(True);
    }
      return(False);
}
start_forward_delay_timer(port_no)
Int port_no;
{  forward_delay_timer[port_no].value = Zero;
    forward_delay_timer[port_no].active = True;
}
stop_forward_delay_timer(port_no)
Int port_no;
{  forward_delay_timer[port_no].active = False;
}
Boolean_ forward_delay_timer_expired(port_no)
Int port_no;
{  if ( forward_delay_timer[port_no].active &&
        (++forward_delay_timer[port_no].value >= bridge_info.forward_delay))
    {   forward_delay_timer[port_no].active = False;
        return(True);
    }
    return(False);
```

```
}
start_hold_timer(port_no)
Int port_no;
{  hold_timer[port_no].value = Zero;
   hold_timer[port_no].active = True;
}
stop_hold_timer(port_no)
Int port_no;
{  hold_timer[port_no].active = False;
}
Boolean hold_timer_expired(port_no)
Int port_no;
{  if (hold_timer[port_no].active &&
       (++hold_timer[port_no].value >= bridge_info.hold_time))
   {  hold_timer[port_no].active = False;
      return(True);
   }
   return(False);
}
/** программы передачи конкретной псевдореализации **/
#include «transmit.c»
```

4.10 **Рабочие характеристики**

В данном разделе устанавливаются требования к рабочим характеристикам мостов в мостовой ЛВС и к установке значений параметров алгоритма и протокола покрывающего дерева. Они необходимы для того, чтобы гарантировать правильную работу алгоритма и протокола.

Приводятся рекомендуемые эксплуатационные значения рабочих параметров. Они определены для того, чтобы избежать необходимости устанавливать эти значения заранее до выполнения операций, и выбраны с точки зрения упрощения взаимодействия компонентов мостовой ЛВС.

Определены абсолютные максимальные значения рабочих параметров, а также диапазон приемлемых значений с тем, чтобы содействовать выбору эксплуатационных значений и обеспечить руководство для разработчиков.

4.10.1 Требования

При правильной работе параметры и конфигурация мостов мостовой ЛВС гарантируют, что:

1) мосты не инициируют реконфигурацию, если в ней никто не нуждается. Это означает, что сообщение протокола моста не будет подвергаться таймированию до получения последующего сообщения, если только не произойдет сбой;
2) после реконфигурации кадры не должны продвигаться по новой активной топологии, если кадры, ранее продвинутые по предшествующей активной топологии, уже находятся в мостовой ЛВС. Это гарантирует отсутствие дублирования кадров;

Эти требования сводятся к установлению ограничений на:

a) максимальный диаметр мостовой ЛВС — максимальное число мостов между любыми двумя пунктами подключений оконечных станций;
b) максимальную транзитную задержку моста — максимальное время, прошедшее между приемом и передачей мостом продвигаемых кадров, которые в случае превышения установленного предела будут аннулированы;
c) максимальную задержку передачи ПБДМ — максимальная задержка, предшествующая передаче ПБДМ, устанавливаемая ввиду необходимости передачи таких ПБДМ согласно правилам 4.7;
d) максимальную переоценку приращения времени существования сообщения, которая может быть выполнена для значения параметра «время существования сообщения» в передаваемых ПБДМ, или времени существования записанной информации мостового протокольного сообщения;
e) значения параметров моста: «время заявки», «максимальное время существования», «задержка продвижения» и «время удержания».

К тому же мост не должен:

i) недооценивать приращение для параметра «время существования» в передаваемых ПБДМ;

ii) недооценивать время задержки продвижения;

iii) переоценивать интервал времени заявки, когда мост действует как корневой.

4.10.2 Значения параметров

Рекомендуемые значения, абсолютный максимум и диапазон параметров определяются в таблицах 4.1—4.5.

Таблица 4.1 — Максимальный мостовой диаметр

Параметр	
Максимальный мостовой диаметр	7

Таблица 4.2 — Транзитная задержка и задержка передачи

В секундах

Параметр	Рекомендуемое значение	Абсолютный максимум
Максимальная мостовая транзитная задержка	1,0	4,0
Максимальная задержка передачи ПБДМ	1,0	4,0
Максимальная переоценка приращения времени существования сообщения	1,0	4,0

Таблица 4.3 — Значения алгоритма и протокола покрывающего дерева

В секундах

Параметр	Рекомендуемое значение	Абсолютный максимум	Диапазон
Мостовое время заявки	2,0	—	1,0—10,0
Мостовое максимальное время существования	20,0	—	6,0—40,0
Мостовая задержка продвижения	15,0	—	4,0—30,0
Время удержания	—	1,0	—

Примечание — Знак «—» означает, что параметр не применяется.

Пункт 4.10.2 вводит взаимосвязь между максимальным мостовым временем существования и мостовой задержкой продвижения.

Таблица 4.4 — Значения параметра приоритета моста и порта

Параметр	Рекомендуемое значение	Диапазон
Приоритет моста	327688	0 — 65536
Приоритет порта	128	0 — 255

Таблица 4.5 — Значения параметра стоимости маршрута

Параметр	Рекомендуемое значение	Абсолютный максимум	Диапазон
Стоимость маршрута	См. 4.10.2	1	1 — 65536

Мост не должен превышать абсолютные максимальные значения параметров, определенных в таблице 4.2, для максимальной мостовой транзитной задержки, максимальной задержки передачи ПБДМ и максимальной переоценки приращения времени существования сообщения.

Если значения мостового времени заявки, максимального мостового времени существования и мостовой задержки продвижения могут быть установлены диспетчером, мост должен иметь возможность использовать полный ряд значений параметра в диапазоне, определенном в таблице 4.3 с разбиением в 1 с.

Мост должен использовать значение времени удержания, приведенное в таблице 4.3.

Мост должен использовать следующие зависимости:

2 × (мостовая задержка продвижения — 1,0 с) ≥ максимального времени существования

максимальное мостовое время существования ≥ 2 × (мостовое время заявки +1,0 с)

Рекомендуется, чтобы предлагаемые значения параметра «стоимость маршрута» для каждого порта моста были получены на основе следующей формулы

Стоимость маршрута = 100/скорость подключенной ЛВС в Мбит/с,

применяя которую для ЛВС, работающей со скоростью 10 Мбит/с, рекомендуемое значение стоимости маршрута составляет 100.

Если значения приоритетов моста и порта для каждого порта могут быть установлены диспетчером, мост должен иметь возможность использовать полный ряд значений параметра в диапазоне, определенном в таблице 4.4.

Если значение стоимости маршрута может быть установлено диспетчером, мост должен иметь возможность использовать полный ряд значений параметра в диапазоне, определенном в таблице 4.4 с разбиением 1.

5 Кодирование ПБДМ

В этом разделе определяется структура и кодирование ПБДМ, передаваемых между логическими объектами протокола моста.

5.1 **Структура**

5.1.1 Передача и представление октетов

Все ПБДМ должны содержать в себе целое число октетов. Октеты в ПБДМ нумеруются, начиная с 1 и возрастая последовательно в порядке их включения в сервисный блок данных уровня звена данных (СБДЗ). Биты в октете нумеруются от 1 до 8, где бит 1 является битом старшей значимости.

Если последовательные октеты используются для представления двоичного числа, то октет с младшим номером является октетом старшей значимости.

Все логические объекты протокола моста соблюдают эти соглашения по расположению битов и октетов, обеспечивая тем самым взаимодействие.

В случаях, когда в данном разделе кодирование ПБДМ представляется с использованием диаграмм, используется следующее представление:

1) октеты с меньшими номерами располагаются слева, а октеты с большими номерами — справа;

2) в пределах октета бит 8 всегда расположен слева, а бит 1 — справа.

5.1.2 Компоненты

Параметр «идентификатор протокола» кодируется в начальных октетах всех ПБДМ. В настоящем стандарте зарезервировано единственное значение параметра «идентификатор протокола». Настоящий стандарт не налагает дополнительных ограничений на структуру, кодирование или использование ПБДМ с другими значениями поля «идентификатор протокола» (при их наличии) другими стандартами по протоколам.

ПБДМ, используемые логическими объектами протокола моста, выполняющими определенный в разделе 4 алгоритм и протокол покрывающего дерева, используют зарезервированное значение идентификатора протокола и имеют следующую структуру.

Каждый ПБДМ включает постоянное число параметров, необходимых для работы протокола. Каждый параметр кодируется в одном или нескольких октетах и имеет постоянную длину. Порядок расположения параметров в ПБДМ является постоянным.

Параметры ПБДМ определяются типом ПБДМ; все ПБДМ одного и того же типа включают в себя одни и те же параметры в одной и той же последовательности, закодированные одинаковым образом.

5.2 **Кодирование типов параметров**

5.2.1 Кодирование идентификатора протокола

Идентификатор протокола должен кодироваться двумя октетами.

5.2.2 Кодирование идентификатора версии протокола

Идентификатор версии протокола должен кодироваться одним октетом. Если два идентификатора версии протокола интерпритируются как двоичные числа без знаков, то число старшей значимости будет относиться к последней версии протокола.

5.2.3 Кодирование типов ПБДМ

Тип ПБДМ должен кодироваться одним октетом. Битовая комбинация октета служит только для различения типов; никакой упорядоченной взаимосвязи между ПБДМ различных типов не предполагается.

5.2.4 Кодирование флагов

Флаг должен кодироваться одним битом одного октета. Таким образом в одном октете может быть закодировано несколько флагов. Флаг считается установленным, если соответствующий бит октета принимает значение 1. Битовые позиции октета, которые не соответствуют флагам, определенным для данного типа ПБДМ, сбрасываются, т. е. должны принимать значения 0. Для ПБДМ данной версии протокола и типа никаких дополнительных флагов не определено.

5.2.5 Кодирование идентификатора моста

Идентификатор моста должен кодироваться восемью октетами, представляющими абсолютную величину двоичного числа. Два идентификатора моста могут сравниваться численно, и меньшее число должно означать мост более низкого приоритета.

Два октета старшей значимости идентификатора моста содержат компонент приоритета, что позволяет применять административное управление относительным приоритетом моста (см. 4.5.3.7 и раздел 6). Шесть октетов младшей значимости обеспечивают уникальность идентификатора моста; они должны получаться из глобального уникального адреса моста (см. 3.12.5) в соответствии со следующей процедурой.

Третий октет старшей значимости образуется из начального октета адреса УДС, бит младшей значимости этого октета (бит 1) принимает значение первого бита адреса моста, следующий бит более старшей значимости — значение второго бита адреса моста, и так далее. В мостовой ЛВС, использующей 48-битовую адресацию УДС, таким же образом октеты с четвертого по восьмой принимают значения октетов со второго по шестой адреса моста.

В мостовой ЛВС, использующей 16-битовую адресацию УДС, четвертый октет принимает значение второго октета адреса моста, а октеты с пятого по восьмой принимают значения 0000 0000.

5.2.6 Кодирование стоимости маршрута к корневому мосту

Стоимость маршрута к корневому мосту должна быть закодирована в четырех октетах в виде абсолютного двоичного числа, кратного произвольному числу единиц стоимости. В 4.10.2 приведены рекомендации по вычислению приращения стоимости маршрута к корневому мосту с целью присвоения этому параметру некоторого общего значения, не требуя от диспетчера практической установки моста в мостовой ЛВС.

5.2.7 Кодирование идентификатора порта

Идентификатор порта должен кодироваться двумя октетами, представляющими абсолютное значение двоичного числа. Если два идентификатора порта численно сравниваются, меньшее число должно означать порт более высокого приоритета. Октет старшей значимости идентификатора порта содержит компонент приоритета, который дает возможность административно управлять относительным приоритетом портов, расположенных в одном и том же мосту (см. 4.5.5 и раздел 6). Октет младшей значимости содержит номер порта, выраженный абсолютным значением двоичного числа. Значение 0 в качестве номера порта не используется.

5.2.8 Кодирование значений тайм-аутов

Значения тайм-аутов должны кодироваться двумя октетами, представляющими абсолютное значение двоичного числа, кратного единице времени — 1/256 с. Это позволяет представлять время в диапазоне от 0, но не включая его, до 256 с.

5.3 **Форматы и параметры ПБДМ**

5.3.1 ПБДМ «конфигурация»

Формат ПБДМ «конфигурация» показан на рисунке 5.1. Каждый передаваемый ПБДМ «конфигурация» должен содержать следующие параметры (4.5.1) и никакие другие:

1) идентификатор протокола кодируется в первом и втором октетах ПБДМ. Он имеет значение 0000 0000 0000 0000, которое идентифицирует алгоритм и протокол покрывающего дерева, определенные в разделе 4;
2) идентификатор версии протокола кодируется в третьем октете ПБДМ и принимает значение 0000 0000;
3) тип ПБДМ кодируется в четвером октете ПБДМ. Это поле должно принимать значение 0000 0000. Это означает ПБДМ «конфигурация»;
4) флаг «подтверждение изменения топологии» кодируется восьмым битом пятого октета ПБДМ;
5) флаг «изменение топологии» кодируется первым битом пятого октета ПБДМ;

6) идентификатор корневого моста кодируется в октетах 6—13 ПБДМ;

7) стоимость маршрута к корневому мосту кодируется в октетах 14—17 ПБДМ;

8) идентификатор моста кодируется в октетах 18—25 ПБДМ;

9) идентификатор порта кодируется в октетах 26 и 17 ПБДМ;

10) значение тайм-аута времени существования сообщения кодируется в октетах 28 и 29 ПБДМ;

11) значение тайм-аута максимального времени существования кодируется в октетах 30 и 31 ПБДМ;

12) значение тайм-аута заявки кодируется в октетах 32 и 33 ПБДМ;

13) значение тайм-аута задержки продвижения кодируется в октетах 34 и 35 ПБДМ.

	Октеты
Идентификатор протокола	1 2
Идентификатор версии протокола	3
Тип ПБДМ	4
Флаги	5
Идентификатор корневого моста	6 7 8 9 10 11 12 13
Стоимость маршрута к корневому мосту	14 15 16 17
Идентификатор моста	18 19 20 21 22 23 24 25
Идентификатор порта	26 27
Время существования сообщения	28 29
Максимальное время существования	30 31
Тайм-аут заявки	32 33
Задержка продвижения	34 35

Рисунок 5.1 — Параметры и формат ПБДМ «конфигурация»

5.3.2 ПБДМ «уведомление об изменении топологии»

Формат ПБДМ «уведомление об изменении топологии» показан на рисунке 5.2. Каждый передаваемый ПБДМ «уведомление об изменении топологии» должен содержать следующие параметры (см. 4.5.2) и никакие другие:

1) идентификатор протокола кодируется в первом и втором октерах ПБДМ. Он принимает значение 0000 0000 0000 0000, которое идентифицирует алгоритм и протокол покрывающего дерева, определенное в разделе 4;

2) идентификатор версии протокола кодируется в третьем октете ПБДМ. Он принимает значение 0000 0000;

3) тип ПБДМ кодируется в четвертом октете ПБДМ. Это поле должно принимать значение *1000 0000, которое означает* ПБДМ «уведомление об изменении топологии»;

5.3.3 Действительность принимаемых ПБДМ

Принимаемый ПБДМ должен обрабатываться логическим объектом протокола моста в соответствии с 4.7 только в том случае, если он содержит параметры «идентификатор протокола», «идентификатор версии протокола», «тип ПБДМ» и либо

1) параметр «тип ПБДМ» указывает ПБДМ «конфигурация» и присутствуют все последующие параметры этого ПБДМ, т. е. ПБДМ сформирован, по меньшей мере, из 35 октетов, либо

2) параметр «тип ПБДМ» указывает ПБДМ «уведомление об изменении топологии» и присутствуют все последующие параметры этого ПБДМ, т. е. ПБДМ сформирован, по меньшей мере, из 4 октетов.

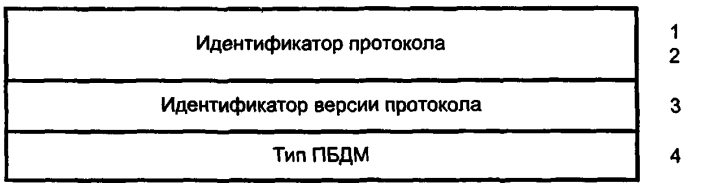


Рисунок 5.2 — Параметры и формат ПБДМ «уведомление об изменении топологии»

6 Диспетчер моста

Средства административного управления обеспечиваются мостами УДС в соответствии с принципами и концепциями основ административного управления ВОС.

В данном разделе:

1) приведена вводная информация о функциональных областях административного управления ВОС для того, чтобы содействовать в определении требований к мостам по обеспечению средств административного управления;
2) устанавливает соответствие между процессами, используемыми для операций моста (см. 3.3), и администрируемыми объектами моста;
3) определяет операции диспетчера, обеспечиваемые каждым администрируемым объектом.

6.1 **Функции диспетчера**

Функции диспетчера относятся к потребностям пользователей в средствах, которые обеспечивают распределение, организацию, наблюдение, управление, защиту и секретность ресурсов передачи и учет их использования. Эти средства могут быть разделены на такие категории, как обеспечиваемые функциональные области конфигурации, неисправности, производительность, защита и административное управление учетом. Каждое из этих средств рассматривается в 6.1.1—6.1.5 вместе со средствами, которые обычно требуются для административного управления коммуникационными ресурсами, и конкретными средствами, предусмотренными в этой функциональной области диспетчером моста.

6.1.1 Диспетчер конфигурации

Служит для идентификации коммуникационных ресурсов, инициализации, сброса и инициирования останова, обеспечения эксплуатационных параметров, а также установления и обнаружения взаимоотношений между ресурсами. Средствами, обеспечиваемыми диспетчером моста в этой функциональной области, являются:

1) идентификация всех мостов, которые совместно образуют мостовую ЛВС и их соответствующие местоположения и, как следствие этой идентификации, местоположения конкретных оконечных станций в конкретных сегментах ЛВС;
2) способность выполнять сброс, т. е. повторную инициализацию определенных мостов;
3) способность управлять приоритетом, с которым порт моста передает кадры;
4) способность устанавливать конкретную конфигурацию покрывающего дерева;
5) способность управлять передачей кадров с конкретными групповыми адресами УДС по определенным маршрутам образованной мостовой ЛВС.

6.1.2 Диспетчер неисправностей

Обеспечивает предотвращение, обнаружение, диагностику и исправление неисправностей. Диспетчер моста в этой функциональной области способен указывать и исправлять неисправности моста, включая регистрацию ошибок и уведомления о них.

6.1.3 Диспетчер рабочих характеристик

Обеспечивает оценку поведения ресурсов передачи данных и эффективности операций по передаче данных. Диспетчер моста в данной функциональной области способен собирать статистические данные о производительности и трафике для анализа. Определенные метрики включают использование сети, подсчитывают продвинутые и аннулированные кадры для конкретных портов внутри моста.

6.1.4 Диспетчер защиты

Используется для защиты ресурсов. Диспетчер моста не обеспечивает никаких средств в этой функциональной области.

6.1.5 Диспетчер административного управления учетом

Диспетчеризация учета обеспечивается для идентификации и распределения стоимости и установленной цены. Диспетчер моста не обеспечивает никаких отдельных средств в этой функциональной области.

6.2 **Администрируемые объекты**

Администрируемые объекты моделируют семантику операций диспетчера. Операции над объектом обеспечивают соответствующую информацию или облегчают контроль над процессом или логическим объектом, связанными с данным администрируемым объектом.

Администрируемые ресурсы моста УДС — это те ресурсы процессов и логических объектов, которые определены в подразделе 3.3. В частности:

1) логический объект диспетчера моста (см. 6.4, 3.11);
2) отдельные логические объекты управления доступом к среде, связанные с каждым портом моста (см. 6.5, 3.2, 3.5 и 3.6);
3) процесс продвижения ретрансляционного логического объекта подуровня УДС (см. 6.6, 3.2 и 3.7);
4) база данных фильтрации ретрансляционного логического объекта подуровня УДС (см. 6.7 и 3.9);
5) логический объект протокола моста (см. 6.8, 3.10 и раздел 4).

Административное управление каждого такого типа ресурса описывается в понятиях администрируемых объектов и операций, приведенных ниже.

6.3 **Типы данных**

В данном разделе определяется семантика операций независимо от их кодирования в протоколе диспетчера. Типы данных рабочих параметров определяются только так, как требуется для данной спецификации.

Используются следующие типы данных:

1) Булево.
2) Перечисление — для сбора поименованных значений.
3) Беззнаковое (абсолютное) — для всех параметров, определенных как «количество» некоторой величины, и для значений приоритета, которые сравниваются численно. В последнем случае меньшее число представляет более высокое значение приоритета.
4) Адрес УДС.
5) Строка знаков латинского алфавита N1 — как определено в ANSI X3.159 для всех строк текста.
6) Интервал времени — абсолютная величина, представляющая положительное целое число в секундах для всех протокольных тайм-аутов.
7) Счетчики — для всех параметров, определенных как «счет» некоторой величины. Счетчик увеличивается и наполняется по модулю от 2 до 64.

6.4 **Логический объект диспетчера моста**

Логический объект диспетчера моста описан в 3.11.

К объектам, которые содержат этот администрируемый ресурс, относятся:

1) конфигуратор моста;
2) конфигуратор порта для каждого порта.

6.4.1 Конфигуратор моста

Объект «конфигуратор моста» моделирует операции, которые модифицируют или запрашивают конфигурацию ресурсов моста. На каждый мост имеется один объект «конфигуратор моста».

К операциям административного управления, которые могут быть выполнены над конфигуратором моста, относятся «открытие моста», «чтение моста», «присвоение имени моста» и «сброс моста».

6.4.1.1 *Открытие моста*

6.4.1.1.1 *Назначение*

Запрос информации, относящейся к мосту(ам) мостовой ЛВС.

6.4.1.1.2 *Входы*

1) Входящий диапазон — набор упорядоченных пар конкретных адресов УДС. Каждая пара определяет диапазон адресов УДС. Мосты должны реагировать только в том случае, если:
   a) для одной из пар численное сравнение их адресов моста с каждым адресом УДС этой пары показывает, что адрес моста больше или равен первому адресу пары, и
   b) меньше или равен второму адресу пары, и
   c) их адреса моста не представлены в описываемом ниже параметре «список исключений».

Численное сравнение одного адреса с другим для выполнения этой операции достигается путем получения числа из адреса УДС в соответствии со следующей процедурой. Последовательность октетов адреса УДС представляется двоичным числом; первый октет, который может передаваться в физическую среду ЛВС при использовании адреса УДС в полях получателя или отправителя кадра УДС, имеет наибольшую значимость, следующий октет — последующую значимость. В пределах каждого октета первый бит является битом младшей значимости.

2) Перечень исключений — перечень определенных адресов УДС.

6.4.1.1.3 *Выходы*

1) Адрес моста — адрес УДС того моста, из которого образуется идентификатор моста, используемый алгоритмом и протоколом покрывающего дерева.
2) Имя моста — строка текста до 32 знаков локально определяемой значимости.
3) Число портов — число портов моста (логических объектов УДС);
4) Адреса порта — следующий перечень для каждого порта:
   a) номер порта — номер порта моста;
   b) адрес порта — конкретный адрес УДС отдельного логического объекта УДС соответствующего порта.
5) Время работы — подсчет в секундах времени, прошедшего с тех пор, когда мост был сброшен или инициирован.

6.4.1.2 *Чтение моста*

6.4.1.2.1 *Назначение*

Получение общей информации, относящейся к мосту.

6.4.1.2.2 *Входы*

Отсутствуют

6.4.1.2.3 *Выходы*

1) Адрес моста — адрес УДС моста, из которого образуется идентификатор моста, используемый алгоритмом и протоколом покрывающего дерева.
2) Имя моста — строка текста до 32 знаков локально определяемой значимости.
3) Число портов — число портов моста (логических объектов УДС).
4) Адресация порта — следующий перечень для каждого порта;
   a) номер порта;
   b) адрес порта — конкретный адрес УДС отдельного логического объекта УДС соответствующего порта.
5) Время работы — подсчет в секундах времени, прошедшего с тех пор, когда мост был сброшен или инициирован.

6.4.1.3 *Присвоение имени мосту*

6.4.1.3.1 *Назначение*

Логическая увязка строки текста, прочитанной операцией «чтение моста», с мостом.

6.4.1.3.2 *Входы*

Имя моста — строка текста до 32 знаков.

6.3.1.3.3 *Выходы*

Отсутствуют.

6.4.1.4 *Сброс моста*

6.4.1.4.1 *Назначение*

Выполнение сброса определенного моста. База данных фильтрации очищается и заполняется записями, определенными в постоянной базе данных, и инициируется логический объект протокола моста (см. 4.8.1).

6.4.1.4.2 *Входы*

Отсутствуют.

6.4.1.4.3 *Выходы*

Отсутствуют.

6.4.2 Конфигурация порта

Объект «конфигурация порта» моделирует операции, которые корректируют или запрашивают информацию о конфигурации портов моста. В каждом мосту имеются стационарные установки портов моста (по одному на каждый интерфейса УДС), каждая из которых определяется постоянным номером порта. Порты нумеруются последовательно, начиная с 1.

Информация, обеспечиваемая объектом «конфигурация порта», содержит сводные данные, определяющие состояние и тип порта. Информация конкретного счетчика, относящаяся к числу продвинутых и отфильтрованных пакетов, а также к числу ошибок, поддерживается ресурсом «процесс продвижения». Операции диспетчера, выполняемые логическим объектом протокола моста, позволяют управлять состояниями каждого порта.

К операциям диспетчера, которые могут быть выполнены над логическим объектом порта, относятся «чтение порта» и «присвоение имени порта».

6.4.2.1 *Чтение порта*

6.4.2.1.1 *Назначение*

Получение общей информации, относящейся к конкретному порту моста.

6.4.2.1.2 *Входы*

Номер порта — номер порта моста.

6.4.2.1.3 *Выходы*

1) Имя порта — строка текста длиной до 32 знаков локально определяемой значимости.
2) Тип порта — тип логического объекта УДС порта (ГОСТ 34.913.3, ГОСТ 34.913.4, ГОСТ Р ИСО/МЭК 8802—5, ГОСТ Р 50452 и др.).

6.4.2.2 *Присвоение имени порта*

6.4.2.2.1 *Назначение*

Логическая увязка строки текста, которую может прочесть операция «чтение порта», с портом моста.

6.4.2.2.2 *Входы*

1) Номер порта.
2) Имя порта — строка текста длиной до 32 знаков.

6.4.2.2.3 *Выходы*

Отсутствуют.

6.5 **Логические объекты управления доступом к среде**

Операции и средства диспетчера, обеспечиваемые логическими объектами УДС, определены в стандартах по административному управлению отдельных подуровней УДС. Логический объект УДС связан с каждым портом моста.

6.6 **Процесс продвижения**

Содержит информацию, относящуюся к продвижению кадров. Предусмотрены счетчики, представляющие информацию относительно количества отфильтрованных, продвинутых кадров и аннулированных в результате ошибок кадров. Процесс продвижения обеспечивает также данные о конфигурации, определяющие способ обработки приоритета кадра.

К объектам, содержащим этот администрируемый ресурс, относятся:

1) счетчики порта;
2) объекты «приоритет передачи» для каждого порта.

6.6.1 Счетчики порта

Объект «счетчики порта» моделируют операции, которые могут выполняться над счетчиками «порт» ресурса «процесс продвижения». В каждом мосту имеется несколько счетчиков порта (по одному на каждый логический объект УДС).

Операцией диспетчера, которая может быть выполнена над счетчиками порта, является «чтение счетчиков продвижения порта».

6.6.1.1 *Чтение счетчиков продвижения порта*

6.6.1.1.1 *Назначение*

Чтение счетчиков продвижения, связанных с конкретным портом моста.

6.6.1.1.2 *Входы*

Номер порта.

6.6.1.1.3 *Выходы*

1) Полученные кадры — число полученных действительных кадров.
2) Входящие аннулированные кадры — число полученных действительных кадров, аннулированных процессом продвижения.
3) Исходящие продвинутые кадры — число кадров, продвинутых соответствующему логическому объекту УДС.
4) Аннулированные кадры из-за недостатка буферной емкости — число кадров, которые должны были быть переданы через соответствующий порт, но были аннулированы из-за недостатка буферной емкости.
5) Аннулированные кадры по истечении транзитной задержки — число кадров, которые должны были быть переданы через соответствующий порт, но были аннулированы по истечении максимальной транзитной задержки моста (может иметь место буферизация).
6) Аннулированные кадры вследствие обнаружения ошибки — число кадров, которые подлежали продвижению к соответствующему подуровню УДС, но не смогли быть переданы (например кадр был слишком длинным).
7) Аннулированные кадры вследствие ошибки элементов — перечень из 16 элементов, каждый из которых содержит адрес отправителя кадра и причину аннулирования кадра (кадр слишком длинный). Перечень поддерживается как циклический буфер. Единственной причиной аннулирования по ошибке является превышение длины передаваемого сервисного блока данных максимально допустимой.

6.6.2 Приоритет передачи

Объект «приоритет передачи» моделирует операции, которые могут выполняться для контроля над тем, правильно ли обрабатывается приоритет кадра для каждого передающего порта в соответствии с 3.7.3. В каждом мосту имеется по несколько объектов «приоритет передачи» (по одному на каждый логический объект УДС).

Операциями диспетчера, которые могут быть выполнены над объектом «приоритет передачи», являются «чтение приоритета передачи» и «установить приоритет передачи».

6.6.2.1 *Чтение приоритета передачи*

6.6.2.1.1 *Назначение*

Чтение набора параметров, управляющих использованием приоритета ретранслируемых кадров.

6.6.2.1.2 *Входы*

Отсутствуют.

6.6.2.1.3 *Выходы*

1) Номер порта.
2) Исходящий приоритет пользователя в диапазоне от 0 до 7.
3) Исходящий приоритет доступа в диапазоне от 0 до 7.

6.6.2.2 *Установление приоритета передачи*

6.6.2.2.1 *Назначение*

Установка параметров, контролирующих использование приоритета ретранслируемых кадров.

6.6.2.2.2 *Входы*

1) Номер порта.
2) Исходящий приоритет пользователя в диапазоне от 0 до 7.
3) Исходящий приоритет доступа в диапазоне от 0 до 7.

6.6.2.2.3 *Выходы*

Отсутствуют.

6.7 **База данных фильтрации**

База данных фильтрации определена в 3.9. Она содержит информацию фильтрации, используемую процессом продвижения (см. 3.7) для того, чтобы определить порт, через который следует продвигать кадры.

К объектам, которые содержат этот администрируемый ресурс, относятся:

1) база данных фильтрации;
2) статические записи;
3) динамические записи;
4) постоянная база данных.

6.7.1 Общие положения

Этот объект моделирует операции, которые могут быть выполнены над базой данных фильтрации в целом или воздействовать на нее. В каждом мосту имеется только один такой объект.

К операциям диспетчера, которые могут быть выполнены над базой данных, относятся «чтение базы данных фильтрации», «установление времени устарения записей в базе данных фильтрации», «удаление записи фильтрации», «создание записи фильтрации», «чтение записи фильтрации» и «чтение набора записей фильтрации», определенные в 6.7.5.

6.7.1.1 *Чтение базы данных фильтрации*

6.7.1.1.1 *Назначение*

Получение общей информации, относящейся к базе данных фильтрации моста.

6.7.1.1.2 *Входы*

Отсутствуют.

6.7.1.1.3 *Выходы*

1) Размер базы данных фильтрации — максимальное число записей, которые может вместить эта база.
2) Число статических записей — количество текущих статических записей в базе данных фильтрации.
3) Число динамических записей — количество текущих динамических записей в базе данных фильтрации.
4) Время старения — определяет момент устарения динамических записей, когда порт, через который проходит запись, находится в состоянии продвижения.

6.7.1.2 *Установление времени старения базы данных фильтрации*

6.7.1.2.1 *Назначение*

Установить время старения динамических записей.

6.7.1.2.2 *Входы*

Время старения.

6.7.1.2.3 *Выходы*

Отсутствуют.

6.7.2 Статическая запись

Этот объект моделирует операции, которые могут быть выполнены над одной статической записью в базе данных фильтрации. Набор объектов статической записи в базе данных фильтрации может изменяться действиями диспетчера.

Объект «статическая запись» поддерживает операции «создание записи фильтрации», «удаление записи фильтрации», «чтение записи фильтрации», определенные в 6.7.5.

6.7.3 Динамическая запись

Этот объект моделирует операции, которые могут быть выполнены над определенной динамической записью в базе данных фильтрации или могут оказать некоторое воздействие на такую запись (т. е. на запись, созданную в базе данных фильтрации процессом обучения в результате наблюдения за сетевым трафиком).

Объект «динамическая запись» поддерживает операции «удаление записи фильтрации» и «чтение записи фильтрации», определенные в 6.7.5.

6.7.4 Постоянная база данных фильтрации

Этот объект моделирует операции, которые могут быть выполнены над постоянной базой данных или воздействовать на нее. В каждой базе данных фильтрации имеется одна постоянная база данных.

К операциям диспетчера, которые могут быть выполнены над постоянной базой данных, относятся «чтение базы данных фильтрации», «создание записи фильтрации», «удаление записи фильтрации», «чтение записи фильтрации» и «чтение набора записей фильтрации», определенные в 6.7.5.

6.7.4.1 *Чтение постоянной базы данных*

6.7.4.1.1 *Назначение*

Получение общей информации, относящейся к постоянной базе данных.

6.7.4.1.2 *Входы*

Отсутствуют.

6.7.4.1.3 *Выходы*

1) Размер постоянной базы данных — максимальное количество записей, которые могут храниться в постоянной базе данных.
2) Число постоянных записей — количество записей, находящихся в данное время в постоянной базе данных.

6.7.5 Основные операции базы данных фильтрации

6.7.5.1 *Создание записи фильтрации*

6.7.5.1.1 *Назначение*

Создание записи в базе данных фильтрации или постоянной базе данных. В базе данных фильтрации могут быть созданы только статические записи.

6.7.5.1.2 *Входы*

1) Идентификатор — база данных фильтрации или постоянная база данных.
2) Адрес — адрес записи на подуровне УДС.
3) Портовая карта — перечень, определяющий:
   a) для входящего порта — номер порта моста;
   b) для исходящих портов — ряд булевых указателей, по одному на каждый порт моста.

Если один из исходящих портов имеет значение «истинно», запись разрешено продвигать в соответствующий порт. Тот исходящий порт, который представляет входящий порт, принимает значение «ложно».

6.7.5.1.3 *Выходы*

Отсутствуют.

6.7.5.2 *Удаление записи фильтрации*

6.7.5.2.1 *Назначение*

Удаление записи из базы данных фильтрации или из постоянной базы данных.

6.7.5.2.2 *Входы*

1) Идентификатор — база данных фильтрации или постоянная база данных.
2) Адрес — адрес требуемой записи на подуровне УДС.

6.7.5.2.3 *Выходы*

Отсутствуют.

6.7.5.3 *Чтение записи фильтрации*

6.7.5.3.1 *Назначение*

Чтение записи из базы данных фильтрации или из постоянной базы данных.

6.7.5.3.2 *Входы*

1) Идентификатор — база данных фильтрации или постоянная база данных.
2) Адрес — адрес требуемой записи на подуровне УДС.

6.7.5.3.3 *Выходы*

1) Адрес — адрес требуемой записи на подуровне УДС.
2) Тип записи — динамическая или статическая.
3) Либо:
   a) номер порта, если тип записи — динамическая;
   b) портовая карта, определенная для операции «создание статической записи», если тип записи — статическая.

6.7.5.4 *Чтение набора записей фильтрации*

6.7.5.4.1 *Назначение*

Чтение набора записей из базы данных фильтрации или постоянной базы данных.

Поскольку число значений, которые должны быть выданы в запрашиваемом наборе, может превысить вместимость сервисного блока данных, содержащего ответ диспетчера, то указанный набор записей идентифицируется. Индексы, определяющие этот набор, могут принимать значения от нуля до величины «размер базы данных фильтрации минус единица».

6.7.5.4.2 *Входы*

1) Идентификатор — база данных фильтрации или постоянная база данных.
2) Индекс начала — содержит начальный индекс требуемого набора записей.
3) Индекс конца — содержит конечный индекс требуемого набора записей.

6.7.5.4.3 *Выходы*

1) Индекс начала — содержит начальный индекс выдаваемого набора записей.
2) Индекс конца — содержит конечный индекс выдаваемого набора записей.
3) Для каждой записи фильтрации выдается:
   a) адрес — адрес требуемой записи на подуровне УДС;
   b) тип записи — динамическая или статическая;
   c) либо:
      i) номер порта, если тип записи — динамическая;
      ii) портовая карта, определенная для операции «создание записи фильтрации», если тип записи — статическая.

6.8 **Логический объект протокола моста**

Этот объект определен в 3.10 и в разделе 4.

К объектам, содержащим этот администрируемый ресурс, относятся:

1) сам логический объект протокола моста;

2) управляемые им порты.

6.8.1 Логический объект протокола моста

Этот объект моделирует операции, которые могут быть выполнены над операциями алгоритма и протокола покрывающего дерева.

В каждом мосту имеется по одному логическому объекту протокола моста, поэтому он может идентифицироваться как отдельный фиксированный компонент ресурса «логический объект протокола».

К операциям диспетчера, которые могут быть выполнены над логическим объектом протокола моста, относятся «чтение параметров протокола моста» и «установка параметров протокола моста».

6.8.1.1 *Чтение параметров протокола моста*

6.8.1.1.1 *Назначение*

Получение информации о логическом объекте протокола моста.

6.8.1.1.2 *Входы*

Отсутствуют.

6.8.1.1.3 *Выходы*

1) Идентификатор моста — как определено в 4.5.3.

2) Время, прошедшее после изменения топологии, — время в секундах, после последней установки флага моста «изменение топологии» в значение «истинно» (см. 4.5.3.12).

3) Число изменений топологии — количество случаев установки флага моста «изменение топологии» (т. е. переходов из значения «ложно» в значение «истинно») после включения питания моста или его инициализации.

4) Изменение топологии (см. 4.5.3.12).

5) Назначенный корневой мост (см. 4.5.3.1).

6) Стоимость маршрута к корневому мосту (см. 4.5.3.2).

7) Порт корневого моста (см. 4.5.3.3).

8) Максимальное время существования (см. 4.5.3.4).

9) Время заявки (см. 4.5.3.5).

10) Задержка продвижения (см. 4.5.3.6).

11) Максимальное мостовое время существования (см. 4.5.3.8).

12) Мостовое время заявки (см. 4.5.3.9).

13) Мостовая задержка продвижения (см. 4.5.3.10).

14) Время удержания (см. 4.5.3.14).

6.8.1.2 *Установка параметров моста*

6.8.1.2.1 *Назначение*

Обновление параметров логического объекта протокола моста с целью создания конфигурации покрывающего дерева и/или установки времени реконфигурации в соответствии с конкретной топологией.

6.8.1.2.2 *Входы*

1) Максимальное мостовое время существования — новое значение (см. 4.5.3.8).

2) Мостовое время заявки — новое значение (см. 4.5.3.9).

3) Мостовая задержка продвижения — новое значение (см. 4.5.3.10).

4) Приоритет моста — новое значение приоритетной части идентификатора моста (см. 4.5.3.7).

6.8.1.2.3 *Выходы*

Отсутствуют.

6.8.1.2.4 *Процедура*

Значения входных параметров проверяются на соответствие требованиям 4.10.2. Если обнаруживается несоответствие либо значение максимального мостового времени существования или мостовой задержки продвижения меньше нижнего предела диапазона, определенного в таблице 4.3, то никаких действий относительно любого из обеспечиваемых параметров не должно предприниматься. Если значение любого из параметров «максимальное мостовое время существования», «мостовая задержка продвижения» или «мостовое время заявки» выходит за рамки диапазона, определенного в таблице 4.3, мост не должен выполнять никаких действий.

В противном случае мостовые параметры «максимальное время существования», «мостовое время заявки» и «мостовая задержка продвижения» устанавливаются в обеспечиваемые значения. Процедура установки параметра «приоритет моста» (см. 4.8.4) используется для установки приоритетной части идентификатора моста в обеспечиваемое значение.

6.8.2 Порт моста

Этот объект моделирует операции, относящиеся к отдельному порту моста, в сочетании с операциями алгоритма и протокола покрывающего дерева.

В каждом мосту имеется постоянный набор портов и, следовательно, будучи фиксированным компонентом ресурса «логический объект протокола», каждый порт может быть идентифицирован постоянным локальным идентификатором порта.

К операциям диспетчера, которые могут быть выполнены над портом моста, относятся «чтение параметров порта», «установка состояния порта» и «установка параметров порта».

6.8.2.1 *Чтение параметров порта*

6.8.2.1.1 *Назначение*

Получение информации о конкретном порте от логического объекта протокола моста.

6.8.2.1.2 *Входы*

Номер порта — номер порта моста.

6.8.2.1.3 *Выходы*

1) Время работы — время в секундах, прошедшее с момента последнего сброса или инициализации порта.

2) Состояние — текущее состояние порта (т. е. неактивное, прослушивание, обучение, продвижение и блокирование) (см. 4.4, 4.5.5.2).

3) Идентификатор порта — уникальный идентификатор порта, содержащий две части: номер порта и поле «приоритет порта» (см. 4.5.5.1).

4) Стоимость маршрута (см. 4.5.5.3).

5) Назначенный корневой мост (см. 4.5.5.4).

6) Назначенная стоимость (см. 4.5.5.5).

7) Назначенный мост (см. 4.5.5.6).

8) Назначенный порт (см. 4.5.5.7).

9) Подтверждаемое изменение топологии (см. 4.5.5.8).

6.8.2.2 *Установление состояния порта*

6.8.2.2.1 *Назначение*

Установление определенного порта в состояние «деактивизировано» или «заблокировано».

6.8.2.2.2 *Входы*

1) Номер порта — номер порта моста.

2) Состояние — деактивизированное или заблокированное (см. 4.4, 5.5.2).

6.8.2.2.3 *Выходы*

Отсутствуют.

6.8.2.2.4 *Процедура*

Если для определенного порта выбрано состояние «деактивизировано», используется процедура «деактивизация порта» (см. 4.8.3). Если выбрано состояние «заблокировано», используется процедура «активизация порта» (см. 4.8.2).

6.8.2.3 *Установка параметров моста*

6.8.2.3.1 *Назначение*

Обновление параметров порта в логическом объекте протокола моста для создания конфигурации покрывающего дерева.

6.8.2.3.2 *Входы*

1) Номер порта — номер порта моста.

2) Стоимость маршрута — новое значение (см. 4.5.5.3).

3) Приоритет порта — новое значение поля «приоритет» идентификатора порта (см. 4.5.5).

6.8.2.3.3 *Выходы*

Отсутствуют.

6.8.2.3.4 *Процедура*

Используется процедура «установление стоимости маршрута» (см. 4.8.6) с целью установления значения параметра «стоимость маршрута» для определенного порта, а также процедура «установление

приоритета порта» (см. 4.8.5) для установления приоритетной части идентификатора порта (см. 4.5.5.1) в обеспечиваемое значение.

7 Протокол диспетчера

В данном разделе определяется, каким образом реализуются средства диспетчеризации, обеспечиваемые мостами УДС и определенные в разделе 6, с использованием набора средств административного управления, установленных в стандарте IEEE Std 802.1B.

В данном разделе определяется:

1) использование протокола административного управления сетевого уровня для выполнения операций диспетчеризации над объектами, определенными в разделе 6;
2) идентификаторы параметров и значения, связанные с каждой операцией;
3) кодирование идентификаторов параметров и типы значений.

Примечание — В стадии разработки находится раздел, который должен определить способ реализации средств диспетчеризации, обеспечиваемых мостами УДС, путем использования услуг общей информации административного управления (УОИУ) (ГОСТ Р ИСО/МЭК 9595) и протокола общей информации административного управления (ПОИУ) (ИСО/МЭК 9596). Это планируемое дополнение будет определять кодирование АСН.1 (ГОСТ Р ИСО/МЭК 8824), используемое ПОИУ, для выполнения определенных операций. Оно будет разработано как часть требований к нотации и кодированию ПОИУ.

7.1 **Определение операций**

Каждая операция над объектом выполняется путем использования одной из следующих операций административного управления сетевого уровня:

1) получение;
2) установка;
3) действие.

7.1.1 Определение операции «получение»

При каждом выполнении операции «получение» определяется:

1) для привлекаемого примитива и связанного с ним ПБД «запрос»;
   a) код идентификатора (КодИд) — целое число, указывающее объект (параметр), к которому предполагается доступ. Он указывается значением кода идентификатора и именем соответствующей макрокоманды вызова ATTRIBUTE, используемой в определении кодирования АСН.1 (ГОСТ Р ИСО/МЭК 8824);
   b) тип идентификатора (определитель ТипаИд), который определяет код идентификатора для того, чтобы выбрать атрибуты или подчиненные объекты, относящиеся к данной операции. Он указывается именем соответствующего ТипаИД в макрокоманде вызова ATTRIBUTE (см. выше). Он используется не всегда;
2) для ответного примитива и связанного с ним ПБД «ответ»:
   a) код идентификатора — как определено для соответствующего привлекаемого примитива;
   b) тип идентификатора — как определено для соответствующего привлекаемого примитива;
   c) тип значения, которое содержит результат операции «получение». Он указывается именем соответствующего ТипаЗначения макрокоманды вызова ATTRIBUTE, рассмотренной выше.

7.1.2 Определение операции «установка»

При каждом выполнении операции «установка» определяется:

1) для привлекаемого примитива и связанного с ним ПБД «запрос»:
   a) код идентификатора,
   b) тип идентификатора (при его наличии),
   c) тип значения, определяющего желаемые значения атрибутов объекта;
2) для ответного примитива и связанного с ним ПБД «ответ»:
   a) код идентификатора,
   b) тип идентификатора,
   c) тип значения, содержащего результат операции «установка».

7.1.3 Определение операции «действие»

При каждом выполнении операции «действие» определяется:

1) для привлекаемого примитива и связанного с ним ПБД «запрос»:
a) идентификатор действия — целое число, идентифицирующее действие, которое должно быть выполнено. Он указывается именем и значением кода идентификатора действия, определенного в АСН.1 (ГОСТ Р ИСО/МЭК 8824),
b) код идентификатора — так же, как и для операций «получение» и «установка»,
c) тип идентификатора — так же, как и для операций «получение» и «установка»,
d) тип значения «действие», определяющего желаемые комбинации атрибутов объекта. Он указывается именем соответствующего ТипаЗначения макрокоманды вызова ATTRIBUTE для объекта, над которым должно выполняться действие;
2) для ответного примитива и связанного с ним ПБД «ответ»:
a) идентификатор действия,
b) код идентификатора,
c) тип идентификатора.

7.2 **Операции**

В таблицах 7.1 —7.4 определено преобразование операций диспетчера моста для логического объекта диспетчера моста (см. 6.4), процесса продвижения (см. 6.6), базы данных фильтрации (см. 6.7) и логического объекта протокола моста (см. 6.8) в операции протокола, определенные в IEEE Std 802.1B.

7.3 **Кодирование**

В данном подразделе определяется кодирование идентификаторов параметров и значения с использованием абстрактно-синтаксической нотации один (АСН.1), определенной в ГОСТ Р ИСО/МЭК 8824 и ГОСТ Р ИСО/МЭК 8825). Общие определения установлены в документе P802.1F/D13.

Таблица 7.1 — Преобразование операций логического объекта диспетчера моста в операции протокола административного управления

Операция диспетчера	Операция протокола	Идентификатор действия	Тип идентификатора	Код идентификатора	Тип значения
Открыть мост	Получение	—	bridgeEntry	BridgeSelect	BridgeConfig
Читать мост	Получение	—	bridgeEntry	—	BridgeConfig
Установить имя моста	Установка	—	bridgeEntry	BridgeNameId	BridgeName
Сбросить мост	Действие	reset	bridgeEntry	—	—
Читать порт	Получение	—	bridgeEntry	PortId	PortConfig
Установить имя порта	Установка	—	bridgeEntry	PortNameId	PortName

Таблица 7.2 — Преобразование операций процесса продвижения в операции протокола административного управления

Операция диспетчера	Операция протокола	Идентификатор действия	Тип идентификатора	Код идентификатора	Тип значения
Читать счетчики продвижения порта	Получение	—	forwardProc	PortCountsId	PortCounts
Читать приоритет передачи	Получение	—	forwardProc	TransmitPriorityId	TransmitPriority
Установить приоритет передачи	Установка	—	forwardProc	TransmitPriorityId	TransmitPriority

Таблица 7.3 — Преобразование операций базы данных фильтрации в операции протокола административного управления

Операция диспетчера	Операция протокола	Идентификатор действия	Тип идентификатора	Код идентификатора	Тип значения
Читать базу данных фильтрации	Получение	—	filterDatabase	FilterDatabaseId	FilterGeneral
Установить время существования записей в базе данных фильтрации	Установка	—	filterDatabase	AgeingTimeId	AgeingTime
Читать постоянную базу данных	Получение	—	filterDatabase	PermanentDatabaseId	PermGeneral
Создать запись фильтрации	Действие	create	filterDatabase	Entry	Filter
Удалить запись фильтрации	Действие	delete	filterDatabase	Entry	Filter
Читать запись фильтрации	Получение	—	filterDatabase	Entry	Filter
Читать ряд записей фильтрации	Получение	—	filterDatabase	EntryRange	FilterRange

Таблица 7.4 — Преобразование операций логического объекта протокола моста в операции протокола административного управления

Операция диспетчера	Операция протокола	Идентификатор действия	Тип идентификатора	Код идентификатора	Тип значения
Читать параметры моста	Получение	—	protocolEntry	ProtocolBridgeId	ProtocolParms
Установить параметры моста	Установка	—	protocolEntry	BridgeParmsId	NewBridgeParms
Читать параметры порта	Получение	—	protocolEntry	ProtocolPortId	PortParms
Установить состояние порта	Действие	ForseState	protocolEntry	ProtocolPortId	NewPortState
Установить параметры порта	Установка	—	protocolEntry	PortParmsId	NewPortParms

```
ZEEE802-1BridgeImeDefinitions DEFINITIONS : : BEGIN
--- rev 1
-- **** Заимствование общих определений и присвоений локальных идентификаторов ****
-- Определение макрокоманды ATTRIBUTE
ATTRIBUTE : : - IEEE802CommonDefinitions.ATTRIBUTE
-- Определение локальных имен для общих типов данных
MACAddress : : - IEEE802CommonDefinitions.MACAddress
ResourceTypeID : : = IEEE802CommonDefinitions.ResourceTypeID
```

```
-- Определение требуемых действий протокола (дополнительно к встроенным операциям протокола:
«получение», «установка»)
IEEE802-1Bridge-Lme-ActionID : : = CHOICE {
                                privateAction  [0] ANY,
                                create         [1] IMPLICIT NULL,
                                delete         [2] IMPLICIT NULL,
                                reset          [3] IMPLICIT NULL,
                                forceState     [4] IMPLICIT NULL }
-- Перечень типичных примеров идентификаторов параметра (idCodes).
-- idCodes параметров определены с использованием макрокоманды ATTRIBUTE.
-- Отрицательные idCodes использованы для реализации конкретных параметров.
-- Взаимоотношения между idCodes, idtypeQualifier (IDTYPES) и VALUETYPES описаны вместе с
-- примерами использования макрокоманды ATTRIBUTE.
--
-- Определения в данном перечне отражают определения, содержащиеся в ссылках ТипаИдентифика-
-- тора на макрокоманды вызова ATTRIBUTE.

-- Примечание. Это соответствие обеспечивается ручным способом без проверки компилятором АСН.1.
-- Перечень определений необходим здесь для спецификации синтаксиса ПБД путем установления взаимоот-
-- ношений между использованием кодов макрокоманды ATTRIBUTE, которая является средством нотации
-- для обобщения всех ссылок на один объект, и всем тем, что входит в ПБД.
--

IEEE802-1Bridge-Lme-ParameterID : : SEQUENCE {
   idCode INTEGER, -- принимает значения кодов макрокоманды ATTRIBUTE
   idtypeQualifier CHOICE {
         bridgeSelect          [0]  IMPLICIT  BridgeSelect,
         bridgeNameId          [1]  IMPLICIT  BridgeNameId,
         portId                [2]  IMPLICIT  PortId,
         portNameId            [3]  IMPLICIT  PortNameId,
         portCountsId          [4]  IMPLICIT  PortCountsId,
         transmitPriorityId    [5]  IMPLICIT  TransmitPriorityId,
         filterDatabaseId      [6]  IMPLICIT  FilterDatabaseId,
         ageingTimeId          [7]  IMPLICIT  AgeingTimeId,
         permanentDatabaseId   [8]  IMPLICIT  PermanentDatabaseId,
         entry                 [9]  IMPLICIT  Entry,
         entryRange            [10] IMPLICIT  EntryRange,
         protocolBridgeId      [11] IMPLICIT  ProtocolBridgeId,
         bridgeParmsId         [12] IMPLICIT  BridgeParmsId,
         protocolPortId        [13] IMPLICIT  ProtocolPortId,
         portParmsId           [14] IMPLICIT  PortParmsId }
-- Перечень типичных примеров значений параметров.
-- Выше перечислен набор ТиповЗначений, определяемых в приводимых ниже примерах использова-
-- ния макрокоманды ATTRIBUTE.
-- Определения в этом перечне отражают определения, содержащиеся в ссылках ТиповЗначений мак-
-- рокоманды вызова ATTRIBUTE.
--
IEEE802-1Bridge-Lme-ParameterValue : : = CHOICE {
               resourceType      [0]  IMPLICIT  ResourceTypeID,
               bridgeConfig      [1]  IMPLICIT  BridgeConfig,
               bridgeName        [2]  IMPLICIT  BridgeName,
               portConfig        [3]  IMPLICIT  PortConfig,
               portName          [4]  IMPLICIT  PortName,
               portCounters      [5]  IMPLICIT  PortCounters,
               transmitPriority  [6]  IMPLICIT  TransmitPriority,
               filterGeneral     [7]  IMPLICIT  FilterGeneral,
               rageingTime       [8]  IMPLICIT  AgeingTime,
```

```
                permGeneral     [9]  IMPLICIT  PermGeneral,
                filter          [10] IMPLICIT  Filter,
                filterRange     [11] IMPLICIT  FilterRange,
                protocolParms   [12] IMPLICIT  ProtocolParms,
                portParms       [13] IMPLICIT  PortParms,
                newBridgeParms  [14] IMPLICIT  NewBridgeParms,
                newPortState    [15] IMPLICIT  NewPortState,
                newPortParms    [16] IMPLICIT  NewPortParms }
--
-- Определение административно управляемых объектов, их идентификаторов параметров (idCodes),
-- определителейТиповИД (ТиповИд) для спецификации выбранных атрибутов объекта, и значений
-- параметров (ТиповЗначений), выбираемых каждым ТипомИд
--
-- Эти определения записываются в форме
--
objectName    ATTRIBUTE
IDTYPES       [<IdtypeQ>] IMPLICIT AnIdtypeQualifierTypeSpecification
              [<IdtypeQ>] IMPLICIT AnotherIdtypeQualifierTypeSpecification
VALUETYPES [<Valtype>] IMPLICIT AValueTypeTypeSpecification
              [<Valtype>] IMPLICIT AnotherValueTypeTypeSpecification
              [<Valtype>] IMPLICIT YetAnotherValueTypeTypeSpecification
  <idCode>
-- Комментарий, отражающий взаимоотношение между операциями административного управления
-- сетевого уровня, ТипамиИд и ТипамиЗначений и операциями диспетчера моста
--
     -- Примечания
     -- 1 <IdtypeQ>, <Valtype> и <idCode> представлены небольшими целыми числами, например 6 При
-- каждом их использовании в макрокоманде ATTRIBUTE их соответствующие значения увеличиваются на
-- единицу Этот механизм действует для любой макрокоманды ATTRIBUTE, поэтому, например, любои
-- <IdtypeQ> уникален в контексте IEEE802—1BridgeLmeDefinitions
     -- 2 Имеется, по меньшей мере, один ТипЗначения для любого объекта, которыи может обрабатывать
-- ся операцией «получение» или «установка» Может иметь место от нуля до нескольких ТиповИд Каждый
--ТипИд, используемый в операциях «получение» или «установка», имеет соответствующее ЗначениеТипа
-- Определение resourseTypeId
-- resourceTypeID ATTRIBUTE
              VALUETYPES [0] ResourceTypeID
  = 0
-- ResourceTypeID на всех уровнях определяется идентификатором 0
-- Логический объект диспетчера моста
bridgeEntity ATTRIBUTE
          IDTYPES       [0] BridgeSelect
                        [1] BridgeNameId
                        [2] PortId
                        [3] PortNameId
          VALUETYPES [1] BridgeConfig
                        [2] BridgeName
                        [3] PortConfig
                        [4] PortName
  = 1
--
-- Доступ к логическому объекту диспетчера моста может осуществляться следующими способами
--
-- Операция «получение», определяющая idCode (1), выдает BridgeConfig
--
-- Операция «получение», определяющая idCode (1) и ТипИд BridgeSelect, выдает BridgeConfig при
-- условии, что адрес моста удовлетворяет критерию выдачи ответов, установленному в BridgeSelect
--
-- Операция «установление», определяющая idCode, ТипИд BridgeNameId и VALUETYPE BridgeName,
-- устанавливает имя моста
--
```

```
-- Операция «действие», с идентификатором действия Reset, определяющая idCode, сбрасывает мост.
--
-- Операция «получение», определяющая idCode и ТипИд PortId выдает ТипЗначения PortConfig.
--
-- Операция «установка», определяющая idCode, ТипИд PortNameId и ТипЗначения PortName, уста-
-- навливает имя порта.
--
BridgeSelect :: = SEQUENCE {
        inclusion [0] IMPLICIT InclusionRange,
        exclusion [1] IMPLICIT ExcluaionList }
   InclusionRange :: = SET OF SEQUENCE {
            lower [0] IMPLICIT MACAddress,
            upper [1] IMPLICIT MACAddress}
   ExclusionList :: = SET OF MACAddress
   BridgeNameId :: = NULL
   PortId :: = INTEGER -- принимает значение от 1 и выше
   PortNameId :: = INTEGER -- принимает то же значение, что и PortId
   BridgeConfig :: = SEQUENCE {
            bridgeAddr    [0] IMPLICIT MACAddress,
            bnarne        [1] IMPLICIT BridgeName,
            portAddresses [2] IMPLICIT PortAddresses,
            uptime        [3] IMPLICIT Counter64}
   BridgeName :: = OCTETSTRING -- распечатываемая строка длиной до 32 октетов (Латинский
алфавит 1)
   PortAddresses :: = SEQUENCE OF PortAddress
   PortAddress :: = SEQUENCE {
            pNumber     [0] IMPLICIT INTEGER,
            -- принимает то же значение, что и PortId portAddr [1] IMPLICIT MACAddress }
-- Имя моста определено выше
PortConfig : : SEQUENCE {
            pName [0] IMPLICIT PortName,
            pType [1] IMPLICIT PortType }
            PortName : : OCTETSTRING -- распечатываемая строка длиной до 32 октетов (Латинский
            алфавит 1)
            PortType : : = INTEGER {
                                p802.3 (83),
                                p802.4 (84),
                                p802.5 (85),
                                FDDI (87) }
    -- Имя порта определено выше
--
-- Процесс продвижения
forwardProc ATTRIBUTE
            IDTYPES         [4] PortCountsId
                            [5] TransmitPriorityId
            VALUETYPES      [5] PortCounters
                            [6] TransmitPriority
: : = 2
--
-- Доступ к процессу продвижения может осуществляться следующими способами:
--
-- Операция «получение», определяющая idCode и ТипИд PortCountId, выдает значения счетчиков
-- порта.
--
-- Операция «получение», определяющая idCode и ТипИд TransmitPriorityId, выдает значение приори-
-- тета передачи.
--
```

```
-- Операция «установка», определяющая idCode, ТипИд TransmitPriorityId и ТипЗначения
-- TransmitPriority, устанавливает приоритеты передачи.
--
   PortCountsId : : = INTEGER -- для идентификации порта принимает значения от 1 и выше
   TransmitPriorityId : : = INTEGER -- для идентификации порта принимает значения от 1 и выше
PortCounters : : = SEQUENCE {
            framesReceived      [0] IMPLICIT Counter64,
            discardInbound      [1] IMPLICIT Counter64,
            forwardOutbound     [2] IMPLICIT Counter64,
            discardBuffers      [3] IMPLICIT Counter64,
            discardTransitDelay [4] IMPLICIT Counter64,
            discardOnError      [5] IMPLICIT Counter64,
            details             [6] IMPLICIT DiscardDetails }
   DiscardDetails : : = SEQUENCE OF DiscardDetail
   DiscardDetail : : = SEQUENCE {
            sourceAddress       [0] IMPLICIT HACAddress,
            errorReason         [1] IMPLICIT INTEGER { reasonTransmitSize (0) }}
TransmitPriority : : = SEQUENCE {
   outboundUserPriority [0] IMPLICIT INTEGER, -- от 0 до 7
   outboundAccessPriority [1] IMPLICIT INTEGER } -- от 0 до 7
--
-- База данных фильтрации
filterDatabase ATTRIBUTE
            IDTYPES             [6] FilterDatabaseId
                                [7] AgeingTimeId
                                [8] PermanentDatabaseId
                                [9] Entry
                                [10] EntryRange
            VALUETYPES          [7] FilterGeneral
                                [8] AgeingTime
                                [9] PermGeneral
                                [10] Filter
                                [11] FilterRange
: : = 3
--
-- Доступ к базе данных фильтрации может осуществляться следующими способами:
--
-- Операция «получение», определяющая idCode (3) и ТипИд FilterDatabaseId, выдает ТипЗначения
-- FilterGeneral.
--
-- Операция «установка», определяющая idCode, ТипИд AgeingTimeId и ТипЗначения AgeingTime,
-- устанавливает время устарения записей в базе данных фильтрации.
--
-- Операция «получение», определяющая idCode и ТипИд PermanentDatabaseId, выдает ТипЗначения
-- PermGeneral.
--
--Операция «действие» с идентификатором действия «создание», определяющим idCode, ТипИд Entry
-- и ТипЗначения Filter, создает запись или в активной базе данных фильтрации, или в постоянной
-- базе данных.
--
-- Операция «действие» с идентификатором действия «удаление», определяющим idCode, ТипИд
-- Entry, удаляет запись из активной базы данных фильтрации или из постоянной базы данных.
-- Операция «получение», определяющая idCode и ТипИд Entry, выдает ТипЗначения Filter.
--
-- Операция «получение», определяющая idCode и ТипИд EntryRange, выдает ТипЗначения FilterRange.
--
```

```
    FilterDatabaseID : : NULL
    PermanentDatabaseID : : NULL
    Entry : : SEQUENCE {
            database [0] IMPLICIT INTEGER {
                                permanentDatabase (0),
                                filteringDatabase (1) }
            addr [1] IMPLICIT MACAddress }
    EntryRange : : = SEQUENCE {
            database [0] IMPLICIT INTEGER {
                                permanentDatabase (0),
                                filteringDatabase (1) }
            startIndex [1] IMPLICIT INTEGER,
            stopIndex [2] IMPLICIT INTEGER }
    FilterGeneral : : = SEQUENCE {
            filterSize          [0] IMPLICIT INTEGER,
            staticEntries       [1] IMPLICIT INTEGER,
            dynamicEntries      [2] IMPLICIT INTEGER,
            ageingTime          [3] IMPLICIT INTEGER }
AgeingTime : : = INTEGER
PermGeneral : : = SEQUENCE {
            permSize            [0] IMPLICIT INTEGER,
            permEntries         [1] IMPLICIT INTEGER }
Filter : : = CHOICE {
            staticEntry         [0] IMPLICIT PortMaps,
            dynamicEntry        [1] IMPLICIT INTEGER }
                                -- для идентификации порта
    PortMaps : : = SEQUENCE OF PortMap
    PortMap : : = SEQUENCE { -- запись для каждого порта, в возрастающем порядке
       inbound [0] IMPLICIT INTEGER -- числа, начиная с 1, для идентификации порта
       outbound [1] IMPLICIT BITSTRING }
       -- бит для каждого порта, в возрастающем порядке; «истинно» указывается значением «1»
FilterRange : : = SEQUENCE {
       startIndex  [0] IMPLICIT INTEGER,
       stopIndex   [1] IMPLICIT INTEGER,
       filterRange [2] IMPLICIT SEQUENCE OF FilterRangeEntry }
    FilterRangeEntry : : = SEQUENCE {
            address  [0] IMPLICIT MACAddress,
            filter   [1] IMPLICIT Filter }
    -- фильтрация определена выше
--
-- Логический объект протокола моста
protocolEntity ATTRIBUTE
--
                    IDTYPES       [11] ProtocolBridgeId
                                  [12] BridgeParmsId
                                  [13] ProtocolPortId
                                  [14] PortParmsId
                    VALUETYPES    [12] ProtocolParms
                                  [13] PortParms
                                  [14] NewBridgeParms
                                  [15] NewPortState
                                  [16] NewPortParms
: : = 4
--
-- Доступ к логическому объекту протокола моста может осуществляться следующими способами:
--
-- Операция «получение», определяющая idCode (4) и ТипИд ProtocolBridgeId, выдает
-- ТипЗначенияProtocolParms.
--
```

```
-- Операция «установка», определяющая idCode, ТипИд BridgeParmsId и ТипЗначения NewBridgeParms,
-- устанавливает параметры моста.
--
-- Операция «получение», определяющая idCode и ТипИд ProtocolPortId, выдает ТипЗначения PortParms.
--
-- Операция «действие» с идентификатором действия ForceState, определяющая idCode, ТипИд
-- ProtocolPortId и ТипЗначения NewPortState, вынуждает порт перейти в определенное состояние.
--
-- Операция «установка», определяющая idCode, ТипИд PortParmsId и ТипЗначения NewPortParms,
-- устанавливает параметры порта.
--
   ProtocolBridgeId : : = NULL
   BridgeparmsId : : = NULL
   ProtocolPortId : : = INTEGER -- принимает значения от 1 и выше
PortParmsId : : = INTEGER -- принимает значения от 1 и выше
ProtocolParms : : = SEQUENCE {
            bridgeid            [0] IMPLICIT BridgeIdentifier,
            timeChange          [1] IMPLICIT INTEGER,
            topologyChangeCount [2] IMPLICIT INTEGER,
            topologyChange      [3] IMPLICIT BOOLEAN,
            designatedRoot      [4] IMPLICIT BridgeIdentifier,
            rootPathCost        [5] IMPLICIT INTEGER,
            rootPort            [6] IMPLICIT INTEGER,
            maxAge              [7] IMPLICIT INTEGER,
            helloTime           [8] IMPLICIT INTEGER,
            forwarDelay         [9] IMPLICIT INTEGER,
            bridgeMaxAge        [10] IMPLICIT INTEGER,
            bridgeHelloTime     [11] IMPLICIT INTEGER,
            bridgeForwardDelay  [12] IMPLICIT INTEGER,
            filterTime          [13] IMPLICIT INTEGER }
            BridgeIdentifier : : = OCTETSTRING /* длина всегда равна 8 */
PortParms : : = SEQUENCE {
            state               [0] IMPLICIT PortStates,
            portid              [1] IMPLICIT PortIdentifier,
            pathCost            [2] IMPLICIT INTEGER,
            desigRoot           [3] IMPLICIT  BridgeIdentifier,
            desigCost           [4] IMPLICIT INTEGER,
            desigBridge         [5] IMPLICIT BridgeIdentifier,
            desigPort           [6] IMPLICIT PortIdentifier,
            stopChangeAck       [7] IMPLICIT BOOLEAN }
--
PortStates : : = INTEGER {
            disabled (0),
            listening (1),
            learning (2),
            forwarding (3),
            blocking (4) }
PortIdentifier : : = SEQUENCE {
            portPriority        [0]  IMPLICIT INTEGER,
            portNumber          [1] IMPLICIT INTEGER }
   -- Идентификатор моста определен выше
--
```

```
NewBridgeParms : : = SEQUENCE {
        maxAge              [0] IMPLICIT INTEGER,
        helloTime           [1] IMPLICIT INTEGER,
        delay               [2] IMPLICIT INTEGER,
        priority            [3] IMPLICIT INTEGER }
NewPortState : : = SEQUENCE {
        number              [0] IMPLICIT INTEGER
        state               [1] IMPLICIT  PortStates }
   -- состояние порта определено выше
NewPortParms : : = SEQUENCE {
        number              [0] IMPLICIT INTEGER,
        cost                [1] IMPLICIT INTEGER,
        priority            [2] IMPLICIT INTEGER }
END
```

8 Рабочие характеристики моста

В данном разделе определен ряд параметров, которые представляют собой рабочие характеристики моста. Эти параметры были выбраны для того, чтобы позволить установить в мосту базовый уровень доверительности при их использовании в начальном определении пригодности этих параметров для заданного применения. Они не могут рассматриваться как исчерпывающее описание рабочих характеристик моста. Рекомендуется в дальнейшем дать дополнительную информацию об этих рабочих характеристиках и о вопросах их применимости в реализации моста.

Определен следующий набор рабочих параметров моста:

1) гарантированная скорость фильтрации порта и связанный с ней интервал времени TF, совместно определяющие трафик, при котором гарантируется фильтрация;
2) гарантированная скорость ретрансляции моста и связанный с ней интервал времени TR.

8.1 **Гарантированная скорость фильтрации порта**

Для определенного порта моста действительная гарантированная скорость фильтрации, измеряемая в единицах кадр/с, — это значение, указывающее, что при любом количестве кадров, выдаваемых из определенного порта моста, подлежащих фильтрации в течение любого интервала TF, процесс продвижения должен отфильтровывать весь набор до тех пор, пока все перечисленные ниже условия не примут значение «истинно»:

1) число кадров в наборе не превышает величину, получаемую при умножении гарантированной скорости фильтрации определенного порта моста на интервал времени TF;
2) гарантированная скорость фильтрации для другого порта(ов) моста не превышена;
3) гарантированная скорость ретрансляции моста не превышена;
4) ретранслируемые кадры не аннулируются из-за перегрузки на выходе (см. 3.7.3).

8.2 **Гарантированная скорость маршрутизации моста**

Для моста действительная гарантированная скорость маршрутизации, измеряемая в единицах кадр/с, — это значение, указывающее, что при любом количестве кадров, выдаваемых из определенного порта моста, подлежащих ретрансляции в течение любого интервала времени TR, процесс продвижения должен ретранслировать весь набор до тех пор, пока все перечисленные ниже условия не примут значение «истинно»;

1) число кадров в наборе не превышает величину, получаемую путем умножения гарантированной скорости ретрансляции моста на интервал времени TR;
2) гарантированная скорость фильтрации для каждого порта не превышена;
3) ретранслируемые кадры не аннулируются из-за перегрузки на выходе (см. 3.7.3).

ПРИЛОЖЕНИЕ А*
(обязательное)

Форма заявки о соответствии реализации протоколу

А.1 **Введение**

Поставщик реализации, которая претендует на соответствие настоящему стандарту, должен заполнить приводимую ниже форму заявки о соответствии реализации протоколу (ЗСРП) и представить информацию, необходимую для идентификации как поставщика, так и реализации.

А. 2 **Сокращения и специальные символы: символы факультативного статуса**

О — обязательно

Ф — факультативно

Фп — факультативно, но, по крайней мере, одна из групп факультативных возможностей, отмеченных одним и тем же номером *n*, обязательна

З — запрещено

позиция — указываемый статус или ответ применим только в том случае, если ЗСРП устанавливает, что позиция, идентифицируемая термином «позиция», обеспечивается

—О— не обязательно, если обеспечивается факультативная возможность, отмеченная тем же номером

А.3 **Инструкции по заполнению копии формы ЗСРП**

Форма ЗСРП представляет собой вопросник фиксированного формата. Она позволяет поставщику обеспечить дополнительной информацию двух видов. При наличии такая информация должна быть представлена в виде позиций, помеченных О.i или Д.i, для целей взаимных ссылок, где i — любой недвусмысленный идентификатор позиции (например обычный номер); никаких других ограничений на ее формат и представление не налагается.

Заполненная форма ЗСРП — это заявка о соответствии реализации протоколу для рассматриваемой реализации.

Ответы на вопросы приводятся в крайней правой колонке либо простой пометкой ответа из ограниченного выбора ответов (обычно «Да» или «Нет»), либо путем записи значения, набора или диапазона значений.

Позиции особой информации обусловлены определенными ответами в вопроснике; они указываются подлежащей заполнению взаимной ссылкой «О.___». Такие позиции имеют место, например, в тех случаях, когда ответ указывает, что возможность, определенная как обязательная, не может быть реализована; наличие позиции особой информации должно иметь соответствующее обоснование.

Форма ЗСРП позволяет поставщику дать дополнительную информацию, позволяющую облегчить интерпретацию ЗСРП. Не предполагается, что будет представлен большой объем такой информации, и ЗСРП может считаться заполненной и без нее. Примерами могут служить описания способов, с помощью которых реализация (отдельная) может быть приспособлена для работы в различных функциональных средах и конфигурациях.

Ссылки на позиции дополнительной информации могут быть даны после любого ответа в вопроснике и могут быть включены в позиции особой информации. Некоторые ответы в вопроснике также требуют дополнительной информации в тех случаях, когда необходимо пояснить, каким образом реализуется возможность, классифицируемая как факультативная.

Примечание — В случае, когда реализацию можно построить несколькими способами, в отдельной ЗСРП можно описать все соответствующие конфигурации. Однако поставщик может предпочесть разработку нескольких ЗСРП, каждая из которых охватывает некоторое подмножество возможных конфигураций реализации, если это облегчает и упрощает представление информации.

А.4 **Идентификация требований**

Информация в каждой из колонок «Функциональная возможность» в форме ЗСРП служит только для наименования требований к соответствию и облегчения их идентификации в настоящем стандарте. Эта информация не заменяет и не дополняет положений, приведенных в основной части настоящего стандарта.

---

* Формы ЗСРП могут свободно копироваться. Пользователи настоящего стандарта могут свободно воспроизводить форму ЗСРП из данного приложения так, чтобы ее можно было использовать для необходимых целей, и могут в дальнейшем публиковать заполненную ЗСРП.

ФОРМА ЗСРП — согласно ГОСТ Р ИСО/МЭК 10038—99

Раздел 1. **Соответствие стандартам по УДС и УЛЗ**

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение	
1a	Соответствует ли реализация метода управления доступом к среде необходимым стандартам по УДС?	КДОН/ОК О	2.5 ГОСТ 34.913.3	Да	Нет: О.__
		Шина с маркерным доступом: О	ГОСТ 34.913.4	Да	Нет: О.__
		Кольцо с маркерным доступом: О	ГОСТ Р ИСО/МЭК 8802-5	Да	Нет: О.__
		ВОРИПД: О	ИСО 9314-2	Да	Нет: О.__
1b	Соответствует ли реализация управления логическим звеном стандарту по УЛЗ?	О	3.2, 3.3, 3.12 ГОСТ 28907	Да	Нет: О.__

Раздел 2. **Ретрансляция и фильтрация кадров**

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение	
2a	Аннулируются ли кадры, полученные с ошибками метода доступа к среде?	О	3.5	Да	Нет: О.__
2b	Направляются ли правильно полученные кадры процессу обучения?	О	3.5	Да	Нет: О.__
2c	Ретранслируются ли только кадры типа «Данные пользователя»?	О	3.5	Да	Нет: О.__
2d	Ретранслируются ли только кадры типа «запрос с ответом»?	О	3.5	Да	Нет: О.__
2e	Направляются ли логическому объекту протокола моста все адресованные ему кадры?	О	3.5	Да	Нет: О.__
2f	Передаются ли только кадры типа «данные пользователя»?	О	3.6	Да	Нет: О.__
2g	Передаются ли только кадры типа «запрос с ответом»?	О	3.6	Да	Нет: О.__
2h	Ставятся ли в очередь на передачу ретранслируемые кадры только при условиях, приведенных в 3.7.1?	О	3.7.1, 3.9.1, 3.9.2, 4.4	Да	Нет: О.__
2i	Сохраняется ли порядок ретранслируемых кадров заданного приоритета пользователя?	О	3.7.3, 3.1.1	Да	Нет: О.__
2j	Направляются ли ретранслируемые кадры для передачи логическому объекту УДС только один раз?	О	2.7.3, 2.3.4	Да	Нет: О.__

Продолжение раздела 2

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение	
2k	Подвергаются ли ретранслируемые кадры максимальной мостовой транзитной задержке?	О	3.7.3	Да	Нет: О.__
2l	Аннулируются ли поставленные в очередь кадры, если порт выходит из состояния продвижения?	О	3.7.3	Да	Нет: О.__
2m	Сохраняется ли приоритет пользователя ретранслируемых кадров там, где это возможно?	О	3.7.4, 2.3.9	Да	Нет: О.__
2n	В противном случае, устанавливается ли приоритет пользователя в значение «исходящий приоритет пользователя?»	О	3.7.4	Да	Нет: О.__
2o	Устанавливается ли приоритет доступа в значение «исходящий приоритет доступа?*»	Ф1	3.7.4	Да	Нет:
2p	Устанавливается ли приоритет доступа в значение «приоритет пользователя?*»	Ф1	3.7.4	Да	Нет:
2q	Может ли мост использовать установленные значения по умолчанию исходящего приоритета доступа?	О	3.7.4, таблица 3—2	Да	Нет: О.__
2r	Превышает ли коэффициент необнаруженных ошибок значение, получаемое путем сохранения КПК там, где это возможно?	З	3.7.5, 2.3.7	Да	Нет: О.__
2s	Сохраняется ли КПК кадров, ретранслируемых между портами УДС одного и того же типа?	Ф	3.7.5	Да	Нет:
2t	Генерирует ли мост примитив УД-БЛОК-ДАННЫХ индикация при получении действительного кадра, передаваемого локальными логическими объектами УДС, относящимися к портам моста?**	З	2.5.4, ИСО 9314-2	Да	Нет: О.__
2u	Используется ли только услуга асинхронной передачи?**	О	ИСО 9314-2, 8.1.4	Да	Нет: О.__
2v	Устанавливает ли мост указатель С при получении из кольцевой ЛВС ВОРИПД кадра для его продвижения?**	Ф	2.5.4, ИСО 9314-2, 7.3.8	Да	Нет
2w	Оставляет ли мост указатель С без изменения при получении из кольцевой ЛВС ВОРИПД кадра для его продвижения?**	Ф	2.5.4, ИСО 9314-2, 7.3.8	Да	Нет: О.__

Окончание раздела 2

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение
3a	Мост фильтрует кадры с одинаковыми адресами отправителя и получателя?	Ф2	3.7.1, 2.3.7	Да Нет
3b	Мост не фильтрует кадры с одинаковыми адресами получателя и отправителя?	Ф2	3.7.1, 3.7.2	Да Нет
4	Обеспечивает ли мост диспетчеризацию приоритетов ретранслируемых кадров?	Ф	3.7.4, таблицы 3.1, 3.2	Да Нет: Д.__
4a	Может ли устанавливаться исходящий приоритет пользователя в любой из диапазонов значений, определенных для каждого порта?	4 : О	3.7.4, таблица 3.1	Да Нет: О.__
4b	Может ли устанавливаться исходящий приоритет доступа в любой из диапазонов, определенных для каждого порта?	4 : О	3.7.4, таблица 3.2	Да Нет: О.__
4c	Соответствует ли исходящий приоритет пользователя для каждого передающего порта определенному в таблице 3.1 значению по умолчанию?	4 : — О	3.7.4, таблица 3.1	Да Нет: О.__
4d	Соответствует ли исходящий приоритет пользователя для каждого передающего порта определенному в таблице 3.2 значению по умолчанию или рекомендуемому значению?	4 : — О	3.7.4, таблица 3.2	Да Нет: О.__

* Относится только к протоколам по ГОСТ 34.913.3, ГОСТ 34.913.4 и ГОСТ Р ИСО/МЭК 8802-5.
** Относится только к протоколу по ИСО 9314-2.

Раздел 3. **Обеспечение информации фильтрации**

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение
5a	Записи в базе данных фильтрации создаются и изменяются только в том случае, если это позволяет состояние порта?	О	3.8, 4.4	Да Нет: О.__
5b	Выполняются ли записи в базе данных фильтрации при приеме кадров с групповым адресом отправителя?	З	3.8, 3.9.2	Нет Да: О.__
5c	Может ли быть создана динамическая запись, которая конфликтует с существующей статической записью?	З	3.8, 3.9	Нет Да: О.__
5d	Обеспечиваются ли статические записи в базе данных фильтрации?	О	3.9	Да Нет: О.__

Продолжение раздела 3

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение
5e	Обеспечиваются ли динамические записи в базе данных фильтрации?	О	3.9	Да Нет: О.__
5f	Удаляется ли любая динамическая запись при создании статической записи для того же адреса?	О	3.9	Да Нет: О.__
5g	Определяет ли каждая статическая запись адрес УДС и преобразование исходящих портов для каждого входящего порта?	О	3.9.1	Да Нет: О.__
5h	Удаляются ли динамические записи из базы данных фильтрации, если они не обновлялись за время существования?	О	3.9.2	Да Нет: О.__
5i	Определяется ли каждой динамической записью адрес УДС и номер порта?	О	3.9.2	Да Нет: О.__
5j	Инициируется ли база данных фильтрации записями, содержащимися в постоянной базе данных?	О	3.9.3	Да Нет: О.__
6a	Констатация размера базы данных фильтрации	О	3.9	—
6b	Констатация размера постоянной базы данных	О	3.9	—
7a	Может ли диспетчер читать базу данных фильтрации?	Ф	3.9	Да: Нет: Д.__
7b	Может ли диспетчер обновлять базу данных фильтрации?	Ф	3.9	Да: Нет Д.__
7c	Могут ли создаваться и удаляться статические записи?	Ф	3.9.1	Да: Нет Д.__
7d	Могут ли выполняться статические записи для индивидуальных адресов УДС?	7c : О	3.9.1	Да Нет: О.__
7e	Могут ли выполняться статические записи для групповых адресов УДС?	7c : О	3.9.1	Да Нет: О.__
7f	Могут ли выполняться статические записи для широковещательного адреса УДС?	7c : О	3.9.1	Да Нет: О.__
7g	Могут ли удаляться и создаваться статические записи в постоянной базе данных?	Ф	3.9.3	Да: Нет Д.__

Окончание раздела 3

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение
8a	Может ли мост автоматически изменяться для использования значения по умолчанию времени существования, рекомендуемого в таблице 3.3?	Ф	3.9.2	Да: Нет Д.__
8b	Может ли мост автоматически изменяться для использования любого диапазона значений времени существования из приведенных в таблице 3.3?	Ф	3.9.2	Да: Нет Д.__

Раздел 4. **Адресация**

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение
9a	Может ли конфигурация моста использовать 48-битовые универсальные адреса?	ФЗ	3.12	Да: Нет Д.__
9b	Может ли конфигурация моста использовать 48-битовые локальные адреса?	ФЗ	3.12	Да: Нет Д.__
9c	Может ли конфигурация моста использовать 16-битовые локальные адреса?	ФЗ	3.12	Да: Нет Д.__
10a	Имеет ли каждый порт отдельный адрес УДС?	О	3.12.2	Да Нет: О.__
10b	Передаются ли все МПБД с одним и тем же групповым адресом?	О	3.12.3, 4.2	Да Нет: О.__
10c	Передаются ли все МПБД с групповым адресом логического объекта протокола моста при использовании универсальной адресации?	9a : О	3.12.3, 4.2	Да Нет: О.__
10d	Является ли адрес отправителя ПБДМ адресом передающего порта?	9a : О	3.12.3	Да Нет: О.__
10e	Является ли адрес моста универсальным адресом?	9a : О 9b : О	3.12.5, 4.2	Да Нет: О.__
10f	Ретранслирует ли мост кадры, направляемые по любому из зарезервированных адресов?	З	3.12.6	Нет Да: О.__
11a	Доступен ли логический объект диспетчера моста через каждый порт с использованием адреса УДС порта и при назначенном ПДУЗ?	13 : Ф	3.12.4	Да Нет
11b	Доступен ли логический объект диспетчера моста через все порты с использованием группового адреса диспетчера моста всех ЛВС?	13 : Ф	3.12.4	Да Нет

Окончание раздела 4

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение
11c	Является ли адрес моста адресом порта 1?	9a : Ф 9c : Ф	3.12.5	Да Нет
11d	Формируются ли заранее в постоянной базе данных групповые адреса дополнительно к резервируемым адресам?	Ф	3.12.6	Да Нет
11e	Могут ли удаляться в базе данных фильтрации дополнительные заранее сформированные записи?	11d : Ф	3.12.6	Да Нет
11f	Формируется ли заранее в постоянной базе данных функциональные адреса для кольца с маркерным методом доступа, определенные в таблице 3.6?	11d : Ф	3.12.6	Да Нет
12a	Может ли назначаться групповой адрес УДС в идентификаторе логического объекта протокола моста?	9b : О 9c : О	4.2	Да: Нет: Д.___ О.__
12b	Может ли назначаться уникальный идентификатор моста?	9c : О	4.2, 4.5.3.7	Да: Нет: Д.___ О.__
12c	Имеет ли каждый порт моста отличный от других идентификатор?	О	4.2, 4.5.5.1	Да Нет: О.__

Раздел 5. **Алгоритм покрывающего дерева**

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение
13a	Обеспечиваются ли все перечисленные ниже параметры моста?	О	4.5.3	Да Нет: О.__
	Назначенный корневой мост		4.5.3.1	
	Стоимость маршрута к корневому мосту		4.5.3.2	
	Порт корневого моста		4.5.3.3	
	Максимальное время существования		4.5.3.4	
	Время заявки		4.5.3.5	
	Задержка продвижения		4.5.3.6	
	Идентификатор моста		4.5.3.7	
	Максимальное мостовое время существования		4.5.3.8	
	Мостовое время заявки		4.5.3.9	
	Мостовая задержка продвижения		4.5.3.10	
	Обнаружение изменения топологии		4.5.3.11	
	Изменение топологии		4.5.3.12	
	Время изменения топологии		4.5.3.13	
	Время удержания		4.5.3.14	

Продолжение раздела 5

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение	
13b	Обеспечиваются ли все перечисленные ниже тайм-ауты моста?	О	454	Да	Нет О__
	Тайм-аут заявки		4541		
	Тайм-аут уведомления об изменении топологии		4542		
	Тайм-аут изменения топологии		4543		
13c	Обеспечиваются ли все перечисленные ниже параметры порта для каждого порта?	О	455	Да	Нет О__
	Идентификатор порта		4551		
	Состояние		4552, 44		
	Стоимость маршрута		4553		
	Назначенныи корневой мост		4554		
	Время заявки		4555		
	Задержка продвижения		4556		
	Идентификатор моста		4557		
	Максимальное мостовое время существования		4558		
	Мостовое время заявки		4559		
13d	Обеспечиваются ли все перечисленные ниже тайм-ауты порта для каждого порта?	О	456	Да	Нет О__
	Тайм-аут времени существования сообщения		4561		
	Тайм-аут задержки продвижения		4562		
	Тайм-аут удержания		4563		
13e	Обеспечиваются ли параметры и тайм-ауты протоколы и передаются ли ПБДМ как требуется при появлении любого из следующих событий?	О	47, 49, 453—456	Да	Нет О__
	Получен ПБДМ «конфигурация»		471		
	Получен ПБДМ «уведомление об изменении топологии»		472		
	Истек тайм-аут заявки		473		
	Истек тайм-аут времени существования сообщения		474		

Продолжение раздела 5

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение	
	Истек тайм-аут задержки продвижения		4.7.5		
	Истек тайм-аут уведомления об изменении топологии		4.7.6		
	Истек тайм-аут времени изменения топологии		4.7.7		
	Истек тайм-аут удержания		4.7.8		
13f	Выполняют ли перечисленные ниже операции модификацию параметров и тайм-аутов протокола и передают ли ПБДМ согласно соответствующим требованиям?	О	4.8, 4.9, 4.5.3, 4.5.4—4.5.6	Да	Нет: О.__
	Инициация		4.8.1		
	Активизация порта		4.8.2		
	Деактивизация порта		4.8.3		
	Установка приоритета моста		4.8.4		
	Установка приоритета порта		4.8.5		
	Установка стоимости маршрута		4.8.6		
14a	Допускает ли мост недооценку приращения параметра времени существования сообщения в передаваемом ПБДМ?	З	4.10.1	Нет	Да: О.__
14b	Допускает ли мост недооценку задержки продвижения?	З	4.10.1	Нет	Да: О.__
14c	Допускает ли мост переоценку интервала времени заявки?	З	4.10.1	Нет	Да: О.__
15a	Использует ли мост значение, определенное для времени удержания?	О	4.10.2, таблица 4.3	Да	Нет: О.__
16	Поддерживает ли мост управление топологией покрывающего дерева?	Ф	4.2	Да	Нет
16a	Может ли устанавливаться относительный приоритет моста?	16 : О	4.2, 4.5.3.7, 4.8.4	Да: Д.___	Нет: О.__
16b	Может ли устанавливаться относительный приоритет среди портов?	16 : О	4.2, 4.5.5.1, 4.8.5	Да: Д.___	Нет: О.__
16c	Может ли устанавливаться стоимость маршрута для каждого порта?	16 : О	4.2, 4.5.5.3, 4.8.6	Да: Д.___	Нет: О.__

Окончание раздела 5

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение	
17	Поддерживает ли мост управление протокольными тайм-аутами?	Ф	4.10	Да	Нет
17a	Можно ли установить любое из значений диапазона, определенного для максимального мостового времени существования?	17 : О	4.10.2, 4.5.3.8, таблица 4.3	Да: Д.___	Нет: О.__
17b	Можно ли установить любое из значений диапазона, определенного для мостового времени заявки?	17 : О	4.10.2, 4.5.3.9, таблица 4.3	Да: Д.___	Нет: О.__
17c	Можно ли установить любое из значений диапазона, определенного для мостовой задержки продвижения?	17 : О	4.10.2, 4.5.3.10, таблица 4.3	Да: Д.___	Нет: О.__
18a	Содержат ли все ПБДМ целое число октетов?	О	5.1.1	Да	Нет: О.__
18b	Кодируются ли все следующие типы параметров ПБДМ в соответствии с требованиями?	О	5.1.1, 5.2	Да	Нет: О.__
	Идентификаторы протокола		5.2.1		
	Идентификаторы версии протокола		5.2.2		
	Тип ПБДМ		5.2.3		
	Флаги		5.2.4		
	Идентификаторы моста		5.2.5		
	Стоимость маршрута к корневому мосту		5.2.6		
	Идентификаторы порта		5.2.7		
	Значения тайм-аутов		5.2.8		
18c	Имеет ли ПБДМ «конфигурация» формат и параметры, соответствующие требованиям?	О	5.3.1	Да	Нет: О.__
18d	Имеет ли ПБДМ «уведомление об изменении топологии» формат и параметры, соответствующие требованиям?	О	5.3.2	Да	Нет: О.__
18e	Проверяется ли правильность принимаемых ПБДМ, как это требуется?	О	5.3.3	Да	Нет: О.__

Раздел 6. **Диспетчер моста**

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение
19	Операции диспетчера моста	Ф	6	Да Нет: О.__
19a	Открыть мост	19 : О	6.4.1.1	Да Нет: О.__
19b	Читать мост	19 : О	6.4.1.2	Да Нет: О.__
19c	Установить имя моста	19 : О	6.4.1.1	Да Нет: О.__
19d	Сброс моста	19 : О	6.4.1.4	Да Нет: О.__
19e	Читать порт	19 : О	6.4.2.1	Да Нет: О.__
19f	Установить имя порта	19 : О	6.4.2.2	Да Нет: О.__
19g	Читать счетчики продвижения порта	19 : О	6.6.1.1	Да Нет: О.__
19h	Читать приоритет передачи	19 : О	6.6.2.1	Да Нет: О.__
19i	Установить приоритет передачи	19 : О	6.6.2.2	Да Нет: О.__
19j	Читать базу данных фильтрации	19 : О	6.7.1.1	Да Нет: О.__
19k	Установить время существования записей в базе данных фильтрации	19 : О	6.7.1.2	Да Нет: О.__
19l	Читать постоянную базу данных	19 : О	6.7.4.1	Да Нет: О.__
19m	Создать запись фильтрации	19 : О	6.7.5.1	Да Нет: О.__
19n	Удалить запись фильтрации	19 : О	6.7.5.2	Да Нет: О.__
19o	Читать запись фильтрации	19 : О	6.7.5.3	Да Нет: О.__
19p	Читать ряд записей фильтрации	19 : О	6.7.5.4	Да Нет: О.__
19q	Читать параметры протокола моста	19 : О	6.8.1.1	Да Нет: О.__
19r	Установить параметры протокола моста	19 : О	6.8.1.2	Да Нет: О.__
19s	Читать параметры порта	19 : О	6.8.2.1	Да Нет: О.__
19t	Установить состояние порта	19 : О	6.8.2.2	Да Нет: О.__
19u	Установить параметры порта	19 : О	6.8.2.3	Да Нет: О.__
20	Удаленное административное управление IEEE 802.1	Ф	7, IEEE Стд 802.1B	Да Нет: О.__
20a	Соответствие стандарту IEEE Стд 802.1B	20 : О	IEEE Стд 802.1B	Да Нет: О.__
20b	Операции административного управления на сетевом уровне и кодирование	20 : О	7	Да Нет: О.__

Раздел 7. **Рабочие характеристики**

Позиция	Функциональная возможность	Статус	Ссылка	Обеспечение
21a	Определение гарантированной скорости фильтрации и соответствующего интервала измерения TF для каждого порта моста в форме, определенной ниже	O	8.1	Д.___
21b	Определение гарантированной скорости ретрансляции и соответствующего интервала измерения TR в форме, определенной ниже Дополнительная информация должна однозначно идентифицировать эти порты	O	8.2	Д.___

Гарантированная скорость ретрансляции моста, кадр/с	*T*, с

Номер (а) порта или другой идентификатор	Гарантированная скорость фильтрации порта (определение для всех портов), кадр/с	*T*, (определение для всех портов), с

ПРИЛОЖЕНИЕ В
(справочное)

Вычисление параметров покрывающего дерева

В данном приложении описывается метод и обоснование вычисления рекомендуемых значений и рабочих диапазонов основных рабочих характеристик параметров алгоритма покрывающего дерева.

В.1 **Общие положения**

Вычисление описывается в виде пронумерованных шагов. На каждом шаге устанавливаются значения для многих параметров, которые затем используются как основа для выполнения последующих шагов.

Данное описание и уравнения относятся к однородной мостовой ЛВС, т. е. к такой сети, в которой все отдельные ЛВС и мосты относятся к одному и тому же типу и работают с одинаковой скоростью. Их легко можно расширить и на неоднородные мостовые ЛВС.

Пояснения иллюстрируются рекомендуемыми значениями для стандартизованных ЛВС. Время указано в секундах.

В.2 **Сокращения и специальные символы**

МДМ — максимальный диаметр моста
МВСК — максимальное время существования кадра
СТЗК — средняя транзитная задержка кадра
СЗДС — средняя задержка доступа к среде
МЗДС — максимальная задержка доступа к среде
ММТЗ — максимальная мостовая транзитная задержка
РСВСС — разрешающая способность времени существования сообщения
МППВС — максимальная переоценка приращения времени существования сообщения
МПВС — максимальная переоценка времени существования сообщения
МЗП — максимальная задержка передачи ПБДМ

МЧПС — максимальное число потерянных протокольных сообщений моста, допустимое до реконфигурации
МВРС — максимальное время распространения мостового протокольного сообщения
ВЗ — время заявки
ВУ — время удержания
МВС — максимальное время существования
ЗП — задержка продвижения

В.3 **Вычисление**

В.3.1 Время существования, диаметр и транзитная задержка

В.3.1.1 *Шаг 1* Выбор максимального диаметра моста мостовой ЛВС и максимальной мостовой задержки продвижения.

Заметим, что в тех случаях, когда отдельная ЛВС обеспечивает диапазон приоритетов передачи, мостовая транзитная задержка может быть изменена в соответствии с приоритетом.

В.3.1.2 *Обоснование выбора*

Время существования кадра равно максимальному диаметру моста, умноженному на максимальную мостовую транзитную задержку, плюс максимальная задержка доступа к среде в начале передачи, т. е.

$$МВСК = (МДМ \times ММТЗ) + МЗДС \qquad (Ур.\ 1)$$

Средняя транзитная задержка кадра между оконечными системами мостовой ЛВС превышает значение задержки, действующей в отдельной ЛВС, на сумму средних значений задержек продвижения и задержек передачи кадра, возникающих в мостах, которые расположены на пути между этими оконечными системами. Эти задержки примерно равны задержкам доступа к среде в слегка нагруженных ЛВС. Так, для систем, расположенных на краях мостовой ЛВС, мы имеем следующее:

$$СТЗК \geq (МДМ \times СЗДС) + СЗДС \qquad (Ур.\ 2)$$

Это ограничивает стремление снижать максимальные транзитные задержки и увеличивать максимальные диаметры мостов.

В.3.1.3 *Рекомендуемые значения для стандартных мостовых ЛВС*

МЗДС $\geq 0{,}5$
МВСК $\leq 7{,}5$
МДМ $= 7$
ММТЗ $= 1{,}0$

В.3.2 Передача ПБДМ

В.3.2.1 *Шаг 2* Выбор приоритета передачи для ПБДМ и значения максимальной задержки передачи ПБДМ.

В.3.2.2 *Обоснование выбора*

В общем случае может быть выбран высокий приоритет передачи, поскольку продолжающаяся работа мостовой ЛВС зависит от успешной передачи и приема ПБДМ. В некоторых случаях может быть более предпочтителен другой трафик, свойственный отдельной ЛВС.

При самом низком значении приоритета, которое может быть выбрано для максимальной задержки передачи ПБДМ, имеет место максимальная задержка доступа к среде для кадров такого приоритета. Осознавая трудности реализации, которые могут возникать при попытке достичь этой величины для кадров с таким приоритетом, представляется более приемлемым выбор значения, равного максимальной мостовой транзитной задержке.

$$МЗП = ММТЗ \qquad (Ур.\ 3)$$

В.3.2.3 *Рекомендуемые значения для стандартных мостовых ЛВС*

Приоритет передачи доступен не для всех стандартных методов доступа к среде. Поэтому, мы выбираем следующее:

МЗП = ММТЗ
МЗП = 1,0

В.3.3 Точность времени существования сообщения

В.3.3.1 *Шаг 3* Выбор соответствующего значения максимальной переоценки приращения времени существования сообщения.

Это максимальная переоценка приращения, добавляемого к значению параметра времени существования сообщения, передаваемого в ПБДМ. Этот параметр позволяет мосту, получившему протокольное сообщение, аннулировать слишком устаревшую информацию.

Вычислите значение этого параметра, который представляет собой максимальную переоценку, даваемую каким-либо мостом относительно времени существования полученной информации мостового протокольного сообщения.

В.3.3.2 *Обоснование выбора*

Выбор максимальной переоценки приращения времени существования сообщения определяется следующим:

1) РСВСС — разрешающей способностью, с которой параметр времени существования сообщения переносится в сообщении о конфигурации;
2) степенью дискретности и точностью тайм-аутов моста;
3) максимальной задержкой передачи ПБДМ.

Предполагая, что тайм-ауты моста не обязательно действуют синхронно с полученными ПБДМ, что они точны, и имеют кратны величине РСВСС, мы имеем, в лучшем случае, следующее:

МППВС = МЗП + РСВСС (Ур. 4)

Это значение следует округлить сверху до ближайшей величины, кратной РСВСС. Здесь важно отметить, что любой мост всегда будет увеличивать значение, по меньшей мере, на одну единицу.

Принимая те же допущения, что и для тайм-аутов моста, принимающего и запоминающего информацию мостового протокольного сообщения, максимальная переоценка времени существования сообщения будет равна максимальной переоценке приращения времени существования сообщения, умноженной на максимальный диаметр моста, минус единица:

МПВС = МППВС × (МДМ — 1) (Ур. 5)

В.3.3.3 *Рекомендуемые значения для стандартных мостовых ЛВС*

МППВС = 1,0
МПВС = 6,0

В.3.4 Время заявки

В.3.4.1 *Шаг 4* Предварительный выбор значения времени заявки.

В.3.4.2 *Обоснование выбора*

Выбор времени заявки осуществляется с учетом его вклада в максимальное время распространения мостового протокольного сообщения (см. шаг 5).

В передаваемых мостовых протокольных сообщениях нет более часто встречающихся интервалов, чем максимальная задержка передачи ПБДМ. В худшем случае, когда мы пытаемся гарантировать правильную работу, эти сообщения должны просто налагаться одно на другое.

В качестве временного решения предлагается удвоенное значение максимальной задержки передачи ПБДМ:

ВЗ = 2 × МЗП (Ур.6)

В.3.4.3 *Рекомендуемые значения для стандартных мостовых ЛВС*

ВЗ = 2,0.

В.3.5 Распространение мостового протокольного сообщения

В.3.5.1 *Шаг 5* Вычисление максимального времени распространения мостового протокольного сообщения.

В.3.5.2 *Обоснование выбора*

Максимальное время распространения мостового протокольного сообщения представляет собой максимальное время, необходимое для переноса информации мостового протокольного сообщения в мостовой ЛВС от одного моста к другому. Оно образуется из следующих составляющих:

1) максимальное время распространения одного мостового протокольного сообщения, необходимое для пересечения мостовой ЛВС, т. е. максимальная задержка передачи ПБДМ, умноженная на максимальный диаметр моста, минус единица;
2) допуск времени заявки, умноженный на максимально допустимое число последовательных потерянных мостовых протокольных сообщений (заметим, что потеря даже одного сообщения должно быть редким явлением);
3) дополнительный допуск времени заявки, поскольку не следует исходить из синхронной работы с корневым мостом, а необходимо выждать это время, прежде чем передать следующий ПБДМ.

МВРС = ((МЧПС + 1) × ВЗ) + МЗП × (МДМ — 1) (Ур.7)

В.3.5.3 *Рекомендуемые значения для стандартных мостовых ЛВС*

Принимая МЧПС = 3, получим
МВРС = 14,0.

В.3.6 Время удержания

В.3.6.1 *Шаг 6* Выбор значения времени удержания.

В.3.6.2 *Обоснование выбора*

Если время удержания превышает максимальную задержку передачи ПБДМ, то максимальное время распространения мостового протокольного сообщения должно быть обусловлено в худшем случае временем удержания в каждом мосту, а не максимальной задержкой передачи ПБДМ. Это может сделать недействительным вывод, приведенный выше в В. 3.5. Поэтому выбирается следующее:

ВУ = МЗП. (Ур.8)

В. 3.6.3 *Рекомендуемые значения для стандартных мостовых ЛВС*

ВУ = 1,0.

В.3.7 Максимальное время существования

В.3.7.1 *Шаг 7* Вычисление нижней границы максимального времени существования для мостовой ЛВС.

В.3.7.2 *Обоснование выбора*

При устойчивых условиях (т. е. отсутствует неисправность, не удаляются и не подключаются мосты и другие компоненты ЛВС), мосты на периферии мостовой ЛВС не должны таймировать корневой мост. В противном случае это может привести к временному локальному отклонению услуг.

Это значит, что максимальное время существования может адекватно справиться с наихудшим случаем задержки распространения и неточностями времени существования протокольного сообщения следующим образом.

Если мост в любое время зависит от информации протокольного сообщения, время существования которого имеет максимальную переоценку, то сумма величин:

1) интервала между передачей последнего протокольного сообщения, который он получил из корневого моста, и первой передачей протокольного сообщения, использующего текущую информацию, и

2) переоценки времени существования текущей информации и

3) времени распространения последнего протокольного сообщения, подлежащего передаче,

может быть меньше максимального времени существования, либо мост будет таймировать информацию протокольного сообщения и пытаться стать корневым мостом.

$$МВС = ((МЧПС + 1) \times ВЗ) + МПВС + (МЗП \times (МДМ - 1)) = МПВС + МВРС. \quad (Ур.9)$$

В.3.7.3 *Рекомендуемые значения для стандартных мостовых ЛВС*

МВС = 20,0.

В.3.8 Задержка продвижения

В.3.8.1 *Шаг 8* Вычисление задержки продвижения

В.3.8.2 *Обоснование вычисления*

Если тайм-аут порта «задержка продвижения» истек и мост начинает продвижение полученных кадров в этом порту, следует позаботиться о том, чтобы в системе больше не оставалось никаких кадров, которые продвигались в предыдущей активной топологии. При наличии таких кадров возникает риск дублирования кадров, либо если при переходе к новой топологии продолжает существовать остаток прежней активной топологии, возникает риск образования шлейфов данных.

Таким образом, периоды прослушивания и обучения, в течение которых действует тайм-аут задержки продвижения, должны охватывать следующие последовательные периоды:

1) от момента входа первого порта моста в состояние прослушивания (и остающегося в этом состоянии в течение последующей реконфигурации) до обнаружения последним мостом мостовой ЛВС изменений в активной топологии;

2) от прекращения продвижения кадров последним мостом, полученных при предыдущей топологии, и до исчезновения последнего продвинутого кадра.

В случае 1) может существовать разница величиной до максимальной переоценки времени существования сообщения между моментами времени, в которые мосты таймируют прежнюю информацию корневого моста и когда они готовятся либо стать корневым мостом, либо прослушивать новый корневой мост. После этого эта величина может достигнуть значения максимального времени распространения мостового протокольного сообщения при передаче новых сведений о новой топологии из нового корневого моста ко всем мостам.

В случае 2) время прекращения продвижения кадров будет равно максимальной задержке прекращения передачи, которая ограничивается максимальной мостовой транзитной задержкой (для всех приоритетов); в дальнейшем кадр будут аннулирован в течение времени существования кадра.

Таким образом мы имеем следующее:

$$2 \times ЗП \geq МПВС + МВРС + ММТЗ + МВСК. \quad (Ур.10)$$

В.3.8.3 *Рекомендуемые значения для стандартных мостовых ЛВС*

ЗП = 15,0.

В.4 **Выбор диапазонов параметров**

В.4.1 Абсолютные максимальные значения

Может потребоваться сформировать ЛВС или мост с более высокой максимальной задержкой доступа к среде, чем предлагается в приведенных выше вычислениях рекомендуемых значений. Это может зависеть от типа трафика, поддерживаемого ЛВС, или от конкретных аспектов реализации моста, ориентированных, например, на увеличение пропускной способности. Однако для обеспечения взаимодействия очень желательно, чтобы абсолютные наибольшие значения максимальной мостовой транзитной задержки, максимальной задержки передачи ПБДМ и максимальной переоценки приращения времени существования сообщения устанавливались на основе настоящего стандарта.

Мосты, работающие с абсолютными максимальными значениями этих параметров, должны сосуществовать с мостами, использующими рекомендуемые значения, в мостовой ЛВС с мостовым диаметром не менее 3. Этот критерий удовлетворяется при следующих условиях:

ММТЗ = 2,0;

МЗП ≤ 2,0;

МППВС ≥ 2,0.

Предполагается, что эти ограничения охватят требования к значениям параметров, превышающим значения, рекомендуемые в В.3.

В.4.2 Время удержания

Невыгодно снижать время удержания ниже рекомендуемого значения максимальной задержки передачи ПБДМ. И ни одна цель не будет достигнута с точки зрения уменьшения использования ширины диапазона или обрабатывающей способности моста путем увеличения времени удержания. Поэтому предлагается зафиксировать значение этого параметра в виде постоянной величины:

ВУ = 1,0.

В.4.3 Диапазон времени заявки

Не требуется, чтобы время заявки было ниже времени удержания. Точно также ни одна цель не будет достигнута путем установки времени заявки в значение, превышающее удвоенное абсолютное максимальное значение максимальной задержки передачи ПБДМ. Поэтому выбирается следующее значение:

$1,0 \le ВЗ \le 4,0$.

В.4.4 Наибольшие требуемые значения максимального времени существования и задержки продвижения

Эти значения вычисляются с использованием уравнений, приведенных в В.3, со следующими значениями параметров:

МДМ = 7;
МЗДС ≤ 2,0;
ММТЗ = 2,0;
МЗП = 2,0;
МППВС = 2,0;
ВЗ = 4,0;
МЧПС = 3,

которые дают следующие результаты:

МВС = 40,0;
ЗП = 30,0.

Несмотря на то, что эти значения предполагаются максимальными, нежелательно препятствовать использованию более высоких значений.

В.4.5 Наименьшие значения максимального времени существования и задержки продвижения

Эти значения вычисляются с использованием уравнений, приведенных в В.3, со следующими значениями параметров:

МДМ = 2;
МЗДС ≤ 0,5;
ММТЗ = 0,5;
МЗП = 0,5;
ВУ = 1,0;
МППВС = 1,0;
ВЗ = 1,0;
МЧПС = 3,

которые дают следующие результаты:

МВС ≤ 6,0;
ЗП = 4,0.

Предлагается, чтобы реализации моста не допускали использования более низких значений максимального времени существования и задержки продвижения для того, чтобы исключить неправильные установки значений.

В.4.6 Взаимоотношения между максимальным временем существования и задержкой продвижения

Для того, чтобы в дальнейшем избежать неправильных установок параметров, что влияет на правильность выполнения операций алгоритма и протокола покрывающего дерева, предлагается, чтобы мосты соблюдали взаимоотношения между максимальным временем существования и задержкой продвижения, установленные в В.3.8, обеспечив следующее соотношение

$2 \times (ЗП - 1,0) \ge МВС$.

ПРИЛОЖЕНИЕ С
(обязательное)

Операции моста по прозрачной маршрутизации со стороны отправителя

С 1 **Общие положения**

Мост прозрачной маршрутизации со стороны отправителя (ПМСО) — это мост УДС, который при приеме кадров с информацией маршрутизации (УИМ = 1) осуществляет маршрутизацию со стороны отправителя, а при приеме кадров без информации маршрутизации (УИМ = 0) осуществляет прозрачные мостовые функции В данном приложении определяются протоколы операций маршрутизации со стороны отправителя, выполняемых мостом ПМСО Форматы кадров, маршрутизируемых со стороны отправителя, и операции моста будут способствовать оконечным станциям в маршрутизации кадров со стороны отправителя Выполнение маршрутизации со стороны отправителя оконечными станциями определено в ИСО/МЭК 8802—2/Доп 5 Маршрутизация со стороны отправителя обеспечивает следующие функции

1) более широкое использование ресурсов путем предоставления многих маршрутов через мостовую ЛВС,

2) более широкое индивидуальное управление кадрами через мостовую ЛВС

С 1 1 Назначение

Для обеспечения совместимого взаимодействия оборудования обработки данных, использующего маршрутизацию со стороны отправителя в мостовой ЛВС, настоящий стандарт определяет операции мостов УДС, обеспечивающих маршрутизацию со стороны отправителя С этой целью он

1) определяет принципы операций моста УДС при обеспечении маршрутизации со стороны отправителя в терминах обеспечения и сохранения услуг УДС, а также обеспечения качества услуг,

2) определяет расширенные внутренние услуги подуровня УДС в их отношении к мостам ПМСО,

3) расширяет архитектурную модель внутренних операций моста,

4) устанавливает требования к диспетчеру моста при выполнении функции маршрутизации со стороны отправителя в мостовой ЛВС, идентифицируя базовые управляемые объекты и операции управления,

5) определяет требования, которым должно отвечать оборудование, претендующее на соответствие настоящему стандарту

Предметом дальнейшего изучения являются следующие вопросы

1) каким образом операции административного управления становятся доступны для удаленного диспетчера с использованием протокола и архитектурного описания, определенного стандартом IEEE Стд 802 1B,

2) требования к рабочим характеристикам, рекомендуемые значения и применяемые диапазоны рабочих параметров моста

К тому же операции ПМСО определяются только для таких УДС, которые имеют определенное поле информации маршрутизации Что касается данной публикации, то такими являются следующие типы УДС

1) маркерный метод доступа к кольцу (ГОСТ Р ИСО/МЭК 8802-5),

2) УДС ВОРИПД (ИСО 9314-2)

Специфичные для УДС вопросы обсуждаются в С 2 5 «Внутренние услуги подуровня»

С 1 2 Определения

Предел ДМ АВМ — максимально допустимое число дескрипторов маршрутов в кадре «анализатор всех маршрутов» Этот предел используется в мостах для ограничения числа мостов, которые пересекают кадры «анализатор всех маршрутов» при их прохождении между оконечными станциями в процессе определения маршрута

Кадр анализатор — кадр, в который мост ПМСО вводит информацию маршрутизации при продвижении кадра Он используется для изучения маршрутов Имеется также кадр «анализатор всех маршрутов», который должен продвигать каждый порт и кадр «анализатор покрывающего дерева», продвигаемый только теми портами, которые назначены для продвижения протоколом покрывающего дерева

ИД входящей ЛВС (ИВхЛВС) — идентификатор ЛВС, из которой мост ПМСО получает кадр

ИД исходящей ЛВС (ИИсхЛВС) — идентификатор ЛВС, которую мост ПМСО передает кадр

Локальная вычислительная сеть (ЛВС) — одна из стандартных локальных вычислительных сетей Например, кольцо с маркерным методом доступа по ГОСТ Р ИСО/МЭК 8802-5 и шина со случайным доступом по ГОСТ 34 913 3 представляют собой ЛВС При образовании мостовой ЛВС может быть объединено несколько ЛВС

Параллельные мосты — два или более мостов, подсоединенных к одной и той же ЛВС

Маршрут — путь, проходящий через несколько ЛВС и мостов ПМСО

Изучение маршрута — процесс получения маршрута к станции назначения

Дескриптор маршрута — двухоктетная область в поле информации маршрутизации, которое определяет ИД ЛВС и номера моста.

Управление маршрутом — первая двухоктетная область в поле информации маршрутизации кадра, маршрутизируемого со стороны отправителя. Эта область указывает тип и характеристики кадра, маршрутизируемого со стороны отправителя.

Информация маршрутизации — область управления маршрутом и последовательность дескрипторов маршрутов, содержащих ИД ЛВС и номера мостов, указывающие маршрут через сеть между двумя станциями.

Маршрутизация со стороны отправителя — механизм моста, предназначенный для маршрутизации кадров через сеть, состоящую из множества ЛВС, посредством определения в кадре маршрута, по которому он будет передаваться.

Предел ДМ АПД — максимально допустимое число дескрипторов маршрута в кадре «анализатор покрывающего дерева». Этот предел реализуется в мосту для ограничения числа мостов, пересекаемых кадрами «анализатор покрывающего дерева».

Прозрачные мостовые функции — мостовой механизм прозрачный для оконечных станций и соединяющий сегменты ЛВС мостами, предназначенными для продвижения кадров путем участия в алгоритме покрывающего дерева.

С.1.2.1 *Сокращения*

В настоящем стандарте определены следующие сокращения:

АВМ — кадр «анализатор всех маршрутов»: УИМ = 1, ТМ = 10х
ПДАВМ — предельное число дескрипторов маршрутов АВМ
НМ — номер моста
ППМ — пара портов моста
ИВхЛВС — ИД входящей ЛВС
ИИсхЛВС — ИД исходящей ЛВС
ДЛ — поле длины
СБД — сервисный блок данных
СБДУД — сервисный блок данных УДС
УМ — управление маршрутизацией
ДМ — дескриптор маршрута
ИМ — информация маршрутизации
ТМ — тип маршрутизации
УИМ — указатель информации маршрутизации
КМСО — кадр, маршрутизируемый по определенному маршруту: УИМ = 1, ТМ = 0хх
ПМСО — прозрачная маршрутизация со стороны отправителя
АПД — кадр «анализатор покрывающего дерева»: УИМ = 0, ТМ = 11х
ПДМАПД — предельное число дескрипторов маршрутов АПД
НМСО — кадр, не маршрутизируемый со стороны отправителя: УИМ = 0, нет поля ИМ

С.1.3 Ссылки

Данное приложение содержит ссылки на следующие стандарты.

ГОСТ 34.936—91 (ИСО 10039—91) Информационная технология. Локальные вычислительные сети. Определение услуг уровня управления доступом к среде

ГОСТ 28696—90 (ИСО 8886—90) Информационная технология. Передача и обмен информацией между системами. Определение услуг звена данных для взаимосвязи открытых систем

ГОСТ 28907—91 (ИСО 8802-2—89) Системы обработки информации. Локальные вычислительные сети. Протокол и услуги уровня управления логическим звеном данных

ГОСТ Р ИСО/МЭК 8802-5—95 Информационная технология. Локальные и региональные вычислительные сети. Часть 5. Метод доступа к кольцу с передачей маркера и спецификация физического уровня

ИСО/МЭК 8802-2—89 Дополнение № 5* Системы обработки информации. Локальные вычислительные сети. Протокол и услуги уровня управления логическим звеном данных. Дополнение № 5. Выполнение оконечными станциями операций маршрутизации со стороны отправителя

ИСО 9314-2—89* Системы обработки информации. Волоконно-оптический распределенный интерфейс передачи данных (ВОРИПД). Часть 2. Уровень управления доступом к среде (УДС) кольца с маркерным доступом ВОРИПД.

---

* До прямого применения данного документа в качестве государственного стандарта распространение его осуществляет секретариат ТК 22 «Информационная технология».

С 1 4 Соответствие

Мост УДС, претендующий на соответствие данному приложению, должен

1) соответствовать основной части настоящего стандарта, дополненного данным приложением, и предоставлять более полный набор функций, описанных в основной части настоящего стандарта, для кадров с информацией маршрутизации,

2) соответствовать статическим и динамическим требованиям, приведенным в С 1 4 1 и С 1 4 2

Кроме того, поставщик реализации, претендующей на соответствие настоящему стандарту, должен заполнить копию формы ЗСРП, приведенную в разделе С 6, и представить информацию, необходимую для идентификации как поставщика, так и реализации

С 1 4 1 *Статические требования к соответствию*

Реализация, претендующая на соответствие настоящему стандарту, должна удовлетворять

1) подуровню УДС и физическому уровню ЛВС, к которои она подключена,

2) статическим характеристикам моста ПМСО согласно С 3

С 1 4 2 *Динамические требования к соответствию*

Реализация, претендующая на соответствие настоящему стандарту, должна

1) проявлять внешнее поведение в соответствии с операциями моста при обработке кадров, маршрутизируемых со стороны отправителя, согласно С 3,

2) сообщать об ошибках согласно С 3

С 2 **Обеспечение услуг УДС**

С 2 1 Общие требования

Услуги УДС, предоставляемые оконечным станциям, подключенным к мостовой ЛВС, обеспечиваются мостами этой мостовой ЛВС, как описано в 2 2

С 2 2 Сохранение услуг УДС

Мосты ПМСО сохраняют услуги УДС, как описано в 2 2

С 2 3 Обеспечение качества услуг

Качество услуг УДС, обеспечиваемое мостом ПМСО, не должно быть значительно ниже качества, обеспечиваемого отдельной ЛВС Полныи перечень параметров качества услуг приведен в 2 3 Параметры, на которые влияет маршрутизация со стороны отправителя, перечислены ниже

1) нарушение последовательности кадров,

2) дублирование кадров,

3) коэффициент необнаруженных ошибок в кадре,

4) обеспечиваемый максимальный размер сервисного блока данных

Кадры без информации маршрутизации относятся к «кадрам, не маршрутизируемым со стороны отправителя» Маршрутизация со стороны отправителя вводит три новых типа кадров В частности, кадры, маршрутизируемые со стороны отправителя, являются основным механизмом транспортировки через сеть, в которой выполняется маршрутизация со стороны отправителя Кадры АПД передаются по той части покрывающего дерева, которая взаимосвязана мостами ПМСО, по мере продвижения они собирают информацию маршрутизации покрывающего дерева Кадры АПД могут использоваться для управления и установления маршрута, но не требуются для передачи данных Все кадры «анализатор всех маршрутов» используются в явном виде в процессе изучения маршрута

С 2 3 1 *Нарушение порядка следования кадров*

Маршрутизация со стороны отправителя допускает наличие и одновременное использование нескольких маршрутов между данной парой станций Если кадры передаются по различным альтернативным маршрутам, они могут поступать в порядке, отличном от порядка их отправки Однако нарушения порядка следования каждого типа кадра по любому конкретному маршруту не происходит Мосты, получающие кадры одного типа (например КМСО), будут передавать их в режиме «первый пришел — первый вышел» (ПВПВ) Относительно конкретных типов кадров

1) порядок следования кадров НМСО не может быть нарушен (см основную часть настоящего стандарта), поскольку они следуют по определенному маршруту покрывающего дерева,

2) порядок следования кадров АПД не может быть нарушен (см основную часть настоящего стандарта), поскольку они следуют по определенному маршруту покрывающего дерева,

3) кадры АВМ используются как кадры управления Нарушение порядка следования не относится к таким кадрам, поскольку по определению они передаются по множеству маршрутов,

4) порядок следования кадров КМСО не может быть нарушен при условии, что отправитель использует только один маршрут со стороны отправителя для доставки последовательности кадров, порядок следования кадров, использующих различные маршруты между однои и тои же парой портов, может быть нарушен

С.2.3.2 *Дублирование кадров*

1) Кадры НМСО и АПД могут дублироваться (см. основную часть настоящего стандарта).

2) Кадры АВМ используются как кадры управления; принимающая сторона должна воспринимать много кадров АВМ (по одному на каждый маршрут).

3) Кадры КМСО не могут быть продублированы, поскольку они продвигаются мостом только в той последовательности, которая определена в поле информации маршрутизации. Более того, мосты предотвращают продвижение кадров с продублированными номерами ЛВС в продвигаемой маршрутной последовательности.

С.2.3.3 *Коэффициент необнаруженных ошибок кадра*

Услуги, предоставляемые подуровнем УДС, вносят в передаваемые кадры очень незначительный коэффициент необнаруженных ошибок (см. 2.3.7). В частности, следует иметь ввиду, что:

1) коэффициент необнаруженных ошибок для НМСО и КМСО остается неизменным;

2) кадры АВМ и АПД собирают информацию маршрутизации при ее продвижении к своему получателю; следовательно, КПК повторно вычисляется в каждом промежуточном мосту. Эти кадры могут иметь высокий коэффициент необнаруженных ошибок, но их не предполагается использовать для транспортировки данных.

С.2.3.4 *Обеспечиваемый максимальный размер СБД*

Полное описание обеспечиваемого максимального размера сервисного блока данных приведено в 2.3.8. Мосты ПМСО отвечают всем требованиям относительно максимального размера СБД. Максимальный размер СБД, обеспечиваемый мостом между двумя ЛВС, меньше размера СБД, обеспечиваемого отдельными ЛВС. Мост не должен пытаться выполнять ретрансляцию кадров в ЛВС, которая не обеспечивает размер СБД, переносимого данным кадром.

Маршрутизация со стороны отправителя снижает вероятность приема мостом кадров неприемлемого размера, поскольку отправитель данных может определить максимальный размер блока данных на выбранном маршруте в процессе открытия маршрута.

С.2.4 Внутренние услуги подуровня

Помимо внутренних услуг подуровня, определенных в 2.4, логические объекты УДС обеспечивают дополнительные расширенные услуги с новым набором примитивов.

Внутренние услуги подуровня исключают те процедуры, действия которых ограничены отдельными ЛВС. Примитивы услуги БЛОК-ДАННЫХ, описывающие данную услугу, приведены ниже. Параметры этих примитивов и их значения идентичны приведенным в 2.4, за исключением добавления нового параметра «информация маршрутизации».

С.2.4.1 *Взаимодействие*

Для того, чтобы запросить УДС передать ПБД, для ретранслирующей функции УДС определены следующие примитивы:

УД-БЛОК-ДАННЫХ запрос

УД-БЛОК-ДАННЫХ индикация.

С.2.4.2 *Подробная спецификация услуг*

Все примитивы определены только в качестве примеров. Каждая услуга присваивает имя конкретному примитиву и запрашиваемой информации, которая передается между УДС и ретрансляционной функцией УДС. Здесь определены только те примитивы, которые не указаны в 2.4.

С.2.4.2.1 УД-БЛОК-ДАННЫХ индикация

Каждый привлекаемый примитив индикации услуги данных УДС соответствует получению кадра из отдельной ЛВС.

Семантика сервисного примитива:

УД-БЛОК-ДАННЫХ индикация (
тип кадра,
действие удс,
адрес получателя,
адрес отправителя,
информация маршрутизации,
сервисный блок данных удс,
приоритет пользователя,
контрольная последовательность кадра
)

Информация маршрутизации. Параметр «информация маршрутизации» определяет одноименное поле полученного кадра.

Действия при генерации. Этот примитив генерируется логическим объектом УДС для ретрансляционной функции УДС с целью информирования о прибытии кадра, подлежащего продвижению. О таких кадрах сообщается только в том случае, если они правильно оформлены (как определено соответствующими подуровнями УДС).

Результат приема. Прием этого примитива побуждает ретрансляционную функцию УДС обработать кадр в соответствии с правилами продвижения, определенными в С. 3.7

С.2.4.2.2 УД-БЛОК-ДАННЫХ запрос

Примитив запроса данных привлекается для передачи кадра в отдельную ЛВС.

Семантика сервисного примитива:

Д-БЛОК-ДАННЫХ запрос (
тип кадра,
действие удс,
адрес получателя,
адрес отправителя,
информация маршрутизации,
сервисный блок данных удс,
приоритет пользователя,
приоритет доступа,
контрольная последовательность кадра
)

Информация маршрутизации. Параметр «информация маршрутизации» определяет одноименное поле кадра, подлежащего продвижению, смодифицированное мостовой функцией.

Примечание — Если КПК обеспечивается, она может продвигаться без повторного вычисления; в противном случае она должна быть вычислена.

Действия при генерации. Этот примитив генерируется ретрансляционной функцией УДС для логического объекта УДС, когда эта функция направляет кадр к логическому объекту УДС для его продвижения.

Результат приема. Прием этого примитива побуждает логический объект УДС присоединить все соответствующие поля и направить правильно сформированный кадр в нижерасположенные уровни протокола для его передачи равноправному логическому объекту или объектам УДС.

С.2.5 Обеспечение внутренних услуг подуровня

В данном разделе определяется преобразование внутренних услуг подуровня в протокол и процедуры УДС и кодирование параметров услуг в кадре УДС.

С.2.5.1 *Обеспечение маркерного метода доступа к кольцу*

Маркерный метод доступа к кольцу определен в ГОСТ Р ИСО/МЭК 8802—5. В разделе 3 указанного стандарта определены форматы и средства, а в разделе 4 — протокол маркерного метода доступа к кольцу.

При получении примитива УД-БЛОК-ДАННЫХ запрос локальный логический объект УДС формирует кадр, используя предоставляемые параметры, как определено выше, добавляя к данным пользователя следующие поля: управление кадра, адрес получателя, адрес отправителя, информацию маршрутизации и КПК. Бит указателя информации маршрутизации устанавливается в единицу, указывая тем самым наличие поля информации маршрутизации. Кадр ставится в очередь на передачу при получении приемлемого маркера (см. 4.1.1). При его передаче добавляется начальный ограничитель, поле управления доступом, конечный ограничитель и поле состояния кадра.

При получении действительного кадра с параметрами «тип кадра» и «действие УДС» в значении «кадр данных пользователя» и «запрос без ответа» соответственно с битом указателя информации маршрутизации в значении единица и при наличии ограничений, при котором либо:

1) тип маршрутизации указывает конкретный маршрутизируемый кадр, и последовательность «ИД входящей ЛВС — номер моста — ИД исходящей ЛВС» соответствует уникальной последовательности «порт — порт», присвоенной данному мосту, либо
2) тип маршрутизации указывает кадр АВМ или АПД, генерируется примитив УД-БЛОК-ДАННЫХ индикация с параметрами, полученными из полей кадра, как определено ниже.

Примечание — При приеме действительного кадра УДС, который не передавался логическим объектом УДС порта моста, с битом указателя информации маршрутизации в значении ноль, генерируется примитив УД-БЛОК-ДАННЫХ индикация в соответствии с требованиями 2.5.3, и кадр обрабатывается мостовой функцией прозрачной маршрутизации.

Параметры «тип кадра», «действие удс», «адрес получателя», «адрес отправителя», «сервисный блок данных удс», «приоритет пользователя» и КПК кодируются, как определено в 2.5.3.

Бит указателя информации маршрутизации кодируется в поле «адрес отправителя». Он занимает ту же позицию в поле адреса отправителя, что и бит указателя группового адреса в поле адреса получателя.

Если примитив УД-БЛОК-ДАННЫХ запрос не сопровождается параметром КПК, она вычисляется в соответствии с 3.2.7 ГОСТ Р ИСО/МЭК 8802-5.

Установка битов А и С осуществляется в соответствии с 2.5.3.

Для обеспечения внутренних услуг подуровня УДС мосты, работающие в кольце с маркерным методом доступа, должны иметь возможность распознавать и удалять кадры, передаваемые самим себе, даже если они содержат адрес отправителя, отличный от адреса порта моста, который их передавал.

С. 2.5.2 *Обеспечение ВОРИПД*

Обеспечение ВОРИПД было введено в стандарт IEEE Std.802.1D добавлением к нему стандарта IEEE Std.802.1i, которые введены в основную часть настоящего стандарта.

При получении действительного кадра (ИСО 9314-2), который не передавался локальным логическим объектом УДС порта моста, генерируется примитив УД-БЛОК-ДАННЫХ индикация. Параметры этого примитива получаются из полей кадра согласно 2.5.4.

При приеме примитива УД-БЛОК-ДАННЫХ запрос локальный логический объект УДС формирует кадр, используя обеспечиваемые параметры, как определено в 2.5.4.

С.3 **Принципы операций**

Данный раздел использует принципы и модель операций моста ПМСО. В нем представлено следующее:

1) пояснение принципиальных элементов работы моста при маршрутизации со стороны отправителя и перечень обеспечиваемых функций;
2) архитектурная модель, определяющая обеспечение этих функций;
3) модель операций моста ПМСО в терминах операций обработки и логических объектов, обеспечивающих эти функции;
4) подробные сведения по дополнительным требованиям к адресации.

С.3.1 Операции моста при маршрутизации со стороны отправителя

К основным элементам операций маршрутизации со стороны отправителя относятся:

1) ретрансляция кадров данных, маршрутизируемых со стороны отправителя;
2) ретрансляция и обработка кадров для передачи и получения информации маршрутизации;
3) управление перечисленными выше функциями.

С.3.1.1 *Ретрансляция кадров данных*

Мост УДС ретранслирует кадры данных пользователя отдельного УДС между различными УДС мостовой ЛВС, подключенных к ее портам. Порядок следования кадров заданного «приоритета пользователя» и «типа кадра», получаемых одним портом и передаваемых в другой, сохраняется.

К функциям, которые обеспечивают ретрансляцию кадров данных, маршрутизируемых со стороны отправителя, и сохраняют качество услуг, обеспечиваемых мостом, относятся:

1) прием кадра;
2) аннулирование кадров, полученных с ошибкой;
3) выбор кадров данных, маршрутизируемых со стороны отправителя;
4) аннулирование кадра, который превышает допустимый для передачи размер СБД (см. С.2.3.4);
5) модификация информации маршрутизации;
6) аннулирование кадра, если максимальная его транзитная задержка превышает допустимое значение;
7) выбор исходящего приоритета доступа;
8) передача кадра.

С.3.1.2 *Распространение информации маршрутизации*

К функциям, которые обеспечивают ретрансляцию и обработку кадров для обеспечения передачи и получения информации маршрутизации, относятся:

1) обработка кадров АВМ,
2) обработка кадров АПД.

С.3.1.3 *Диспетчер моста*

Функции, обеспечивающие управление со стороны диспетчера моста и контроль за предоставлением функций, приведенных в предыдущих подразделах, определены в С.4.

С.3.2 Архитектура моста

Каждый порт моста принимает из ЛВС и передает кадры в ЛВС, к которой он подключен, используя услуги, предоставляемые отдельным логическим объектом УДС, связанным с этим портом, как описано в 3.2. Ретранслирующий логический объект УДС реализует независимые от УДС функции по ретрансляции кадра между портами моста. Если принятый кадр не является маршрутизируемым со стороны отправителя (УИМ = 0), то мост продвигает кадр или аннулирует его, используя логику, описанную в разделе 3 («Логика прозрачной мостовой маршрутизации»). Если принятый кадр является маршрутизируемым со стороны отправителя (УИМ = 1), он обрабатывается, используя логику, описанную в С.3.3 (см. рисунок С.1).

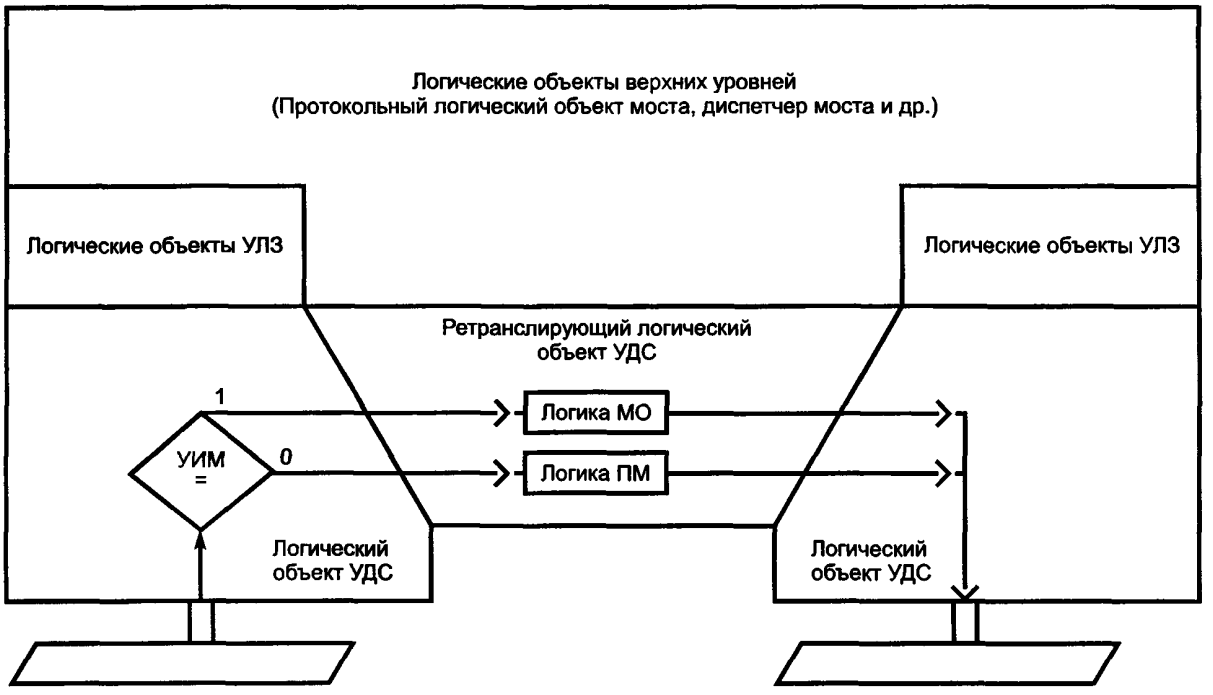


Рисунок С.1 — Логика операций моста ПМСО

При мостовых операциях ПМСО порт обслуживает определенную ЛВС, и этой ЛВС присваивается уникальный ИД ЛВС. Поскольку ПМСО допускает несколько мостов на пути между двумя ЛВС, номер моста (НМ) присваивается каждому мосту на пути следования таким образом, чтобы любые два моста на пути между одними и теми же двумя ЛВС имели различные номера. Из этого распределения следует, что каждый маршрут между ЛВС может быть уникально идентифицирован ИД ЛВС и номером моста. Маршрут через мост определяется в поле ИМ путем указания ИВхЛВС, НМ, ИИсхЛВС, где ИВхЛВС — это ИД ЛВС для принимающего порта и ИИсхЛВС — ИД ЛВС для продвигающего порта.

Следовательно, между каждой уникальной парой портов моста осуществляется мостовая связь (мостовой маршрут). Термин «пара портов моста» (ППМ) будет использоваться для представления пары портов, образующих мостовой маршрут, и каждой ППМ будет присваиваться пара номеров порта и соответствующий им НМ. Таким образом маршрут через мост может быть определен путем установки ИД ЛВС и НМ, присвоенных данной паре портов.

Каждый порт имеет следующие атрибуты:

1) номер порта;

2) ИД ЛВС;

3) самый большой СБДУД;

4) тип порта ПМСО;

5) предельное число ДМ.

При таком архитектурном подходе достаточно описать обработку кадра мостом на основе пары портов моста, как это приведено в С. 3.7.

С.3.3 Операции моста

Использование ретранслирующим логическим объектом УДС внутренних услуг подуровня, которые обеспечиваются отдельными логическими объектами УДС, связанными каждым портом моста, определено в С.3.5 и С.3.6. Операции моста над кадрами без информации маршрутизации (УИМ = 0) описаны в 3.3. Протокольный логический объект моста остается без изменений. Логический объект диспетчера моста расширяется для включения управляемых объектов МСО, как описано в С.4. Ретранслирующий логический объект УДС расширяется для обработки кадров, маршрутизируемых со стороны отправителя, и кадров, не маршрутизируемых со стороны отправителя. Процесс продвижения при маршрутизации со стороны отправителя продвигает полученные кадры в другие порты моста на основе информации, содержащейся в кадре, и на основе состояния портов моста (см. С. 3.7). Адреса отправителя кадров, маршрутизируемых со стороны отправителя (с УИМ = 1), не обрабатываются мостовой функцией прозрачной маршрутизации. Это объясняется

тем, что невозможно логически увязать порт покрывающего дерева с адресом в кадре при маршрутизации со стороны отправителя, поскольку маршруты при маршрутизации со стороны отправителя не всегда совпадают с маршрутами покрывающего дерева.

С.3.3.1 *Общее описание функций маршрутизации со стороны отправителя*

Для кадров, маршрутизируемых со стороны отправителя, решение о их продвижении основывается на информации, содержащейся в поле информации маршрутизации основной части кадра, при наличии такого поля. При его отсутствии решение о продвижении основывается на поле «адрес получателя» кадра (с помощью логики мостовой функции прозрачной маршрутизации).

После того как маршрут определен, станция — отправитель кадра назначает маршрут, по которому будут передвигаться кадры, путем введения описания маршрута в поле информации маршрутизации (ИМ) передаваемого кадра. Мосты копируют и продвигают такие кадры от одного сегмента ЛВС к другому без изменения их содержимого при условии, что типы УДС одинаковы. Если типы УДС различны, содержимое кадров может измениться и КПК должна быть пересчитана. Если данные передаются в кадре, в котором информация маршрутизации модифицируется (например кадр «анализатор всех маршрутов»), КПК должна быть пересчитана и целостность данных пользователя может быть нарушена. Маршрутизация со стороны отправителя обеспечивает возможность кадрам, содержащим данные пользователя, пересекать однотипные мостовые ЛВС без перерасчета КПК.

Станции вначале получают маршрут к данной станции назначения с помощью процесса, называемого «изучение маршрута». Этот процесс позволяет станции по мере надобности динамично изучать маршрут к станции назначения. Каждый мост, который продвигает кадр «изучение (анализатор) маршрута», указывает в поле ИМ конфигурацию моста (ИВхЛВС, НМ, ИИсхЛВС) и самый большой размер кадра, который может обеспечить мост (через отдельную ИсхЛВС), если он меньше уже определенного размера, и повторно генерирует КПК. Таким образом информация маршрутизации формируется мостами по мере продвижения кадра-анализатора от одного сегмента ЛВС к другому. Когда кадр достигает станции назначения, поле ИМ указывает маршрут кадра, прошедшего путь от отправителя к получателю, и максимальный размер кадра, обеспеченный по всему маршруту.

На рисунке С.2 изображены элементы функции маршрутизации со стороны отправителя.

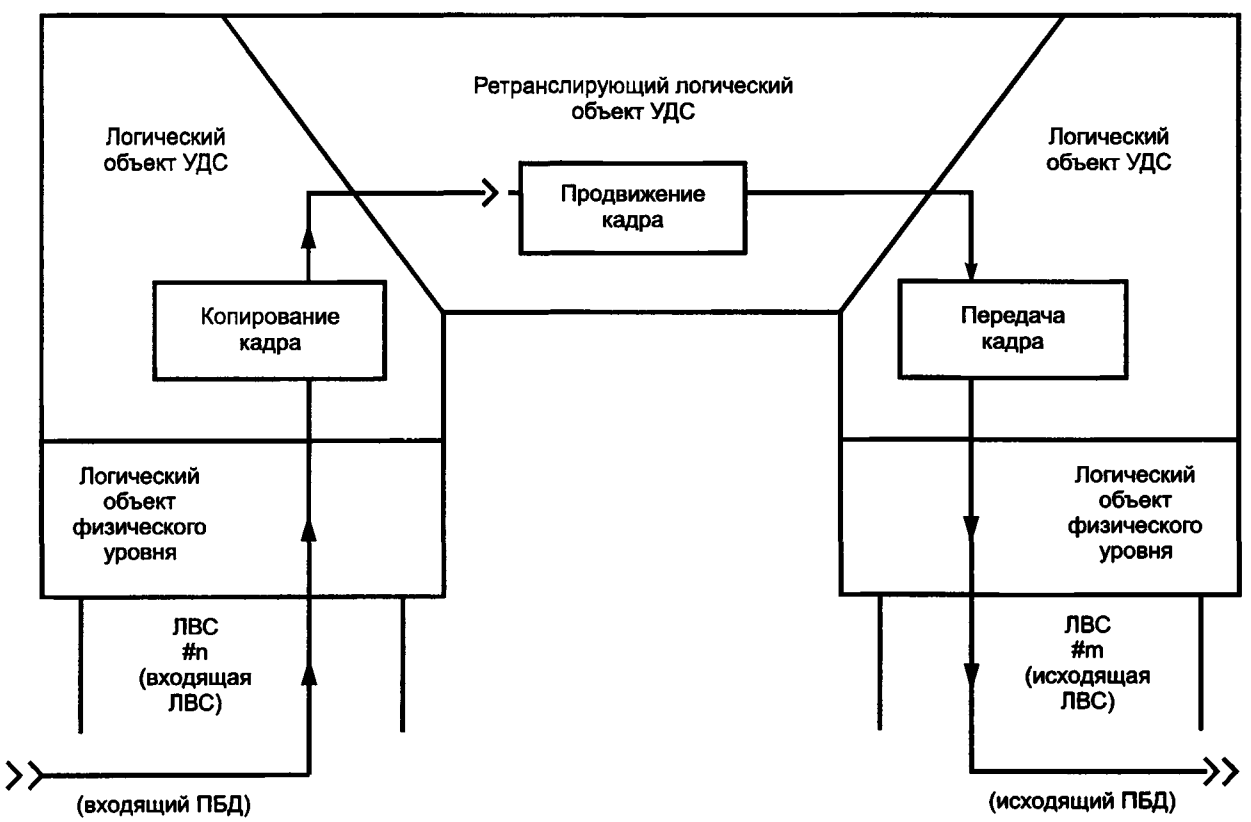
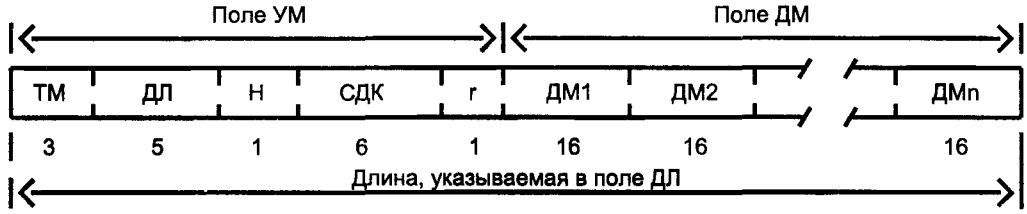


Рисунок С.2 — Элементы мостовой функции маршрутизации со стороны отправителя

С. 3.3.2 *Поле информации маршрутизации со стороны отправителя*
Формат этого поля приведен ниже



УМ — управление маршрутизации (16 битов); ДМn — дескриптор *n* маршрутов (16 битов);
ТМ — тип маршрутизации (3 бита); ДЛ — длина (5 битов); Н — бит направления (1 бит);
СДК — самый длинный кадр (6 битов); r — зарезервированный бит (1 бит)

Поле ИМ передается в физическую среду в соответствии с обычной практикой обработки данных для каждой ЛВС. Рисунок С.3 эквивалентен приведенному выше описанию и представляет поле ИМ во временном аспекте.

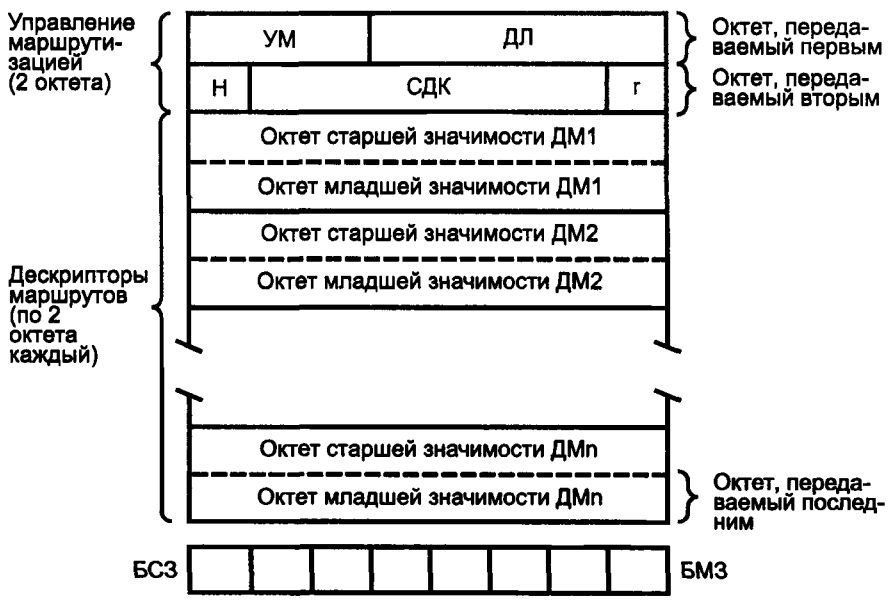


Рисунок С.3 — Поле информации маршрутизации

Поле «тип маршрутизации» (ТМ). Указывает, подлежит ли кадр продвижению через сеть только по одному или по нескольким маршрутам через множество взаимосвязанных ЛВС.

Кадр, маршрутизируемый по конкретному маршруту (ТМ = 0XX). Если бит старшей значимости ТМ установлен в 0, поле ДМ содержит конкретный маршрут через сеть. При передаче через мостовую ЛВС биты XX сохраняются.

Кадр «анализатор всех маршрутов» (ТМ = 10X). Если быты ТМ установлены в значение 10X, указывая кадр «анализатор всех маршрутов», кадр будет передаваться по каждому маршруту сети, разрешенному решением о продвижении АВМ. В результате применения этого типа маршрутизации на станцию назначения прибудет столько кадров, сколько существует различных маршрутов от отправителя к данной станции. Он отправляется оконечной станцией без дескрипторов маршрутов. Дескрипторы маршрутов вводятся в кадр мостами ПМСО по мере его продвижения. При передаче через мостовую ЛВС бит X сохраняется.

Кадр «анализатор покрывающего дерева» (ТМ = 11X). Если биты ТМ установлены в значение 11X, указывая кадр «анализатор покрывающего дерева», кадр из одной ЛВС в другую ретранслируют только те мосты ПМСО, порты которых находятся в состоянии прозрачной мостовой функции продвижения, в результате чего кадр будет продвигаться по всему покрывающему дереву мостовой ЛВС и появляться не более одного раза в каждой ЛВС в сети. Он отправляется оконечной станцией без дескрипторов маршрутов. Дескрипторы маршрутов вводятся в кадр мостами ПМСО по мере его продвижения. При передаче через мостовую ЛВС бит X сохраняется.

Биты длины (ДЛ). Эти пять битов указывают длину (в октетах) поля ИМ. Значение поля длины должно принимать значение от 2 до 30 включ.

Бит направления (Н). Если бит Н установлен в нуль, кадр пересекает ЛВС, образующие сеть, в том порядке, который определен в поле информации маршрутизации (ДМ1 —ДМ2 ... ДМn). В противном случае (бит Н установлен в 1) кадр будет пересекать ЛВС, образующие сеть, в обратном порядке (ДМn — ДМn—1 ... ДМ1). Бит Н имеет значимость только для КМСО. Для кадров АПД и АВМ бит Н должен быть установлен в нуль.

Биты самого длинного кадра (СДК). Биты СДК указывают самый большой размер СБД УДС (поле информации УДС), который может быть передан между двумя взаимодействующими станциями по конкретному маршруту. Биты СДК имеют значимость только для кадров АПД и АВМ. Для КМСО биты СДК игнорируются и не должны изменяться мостом. Станция — отправитель кадра анализатора устанавливает биты СДК в значение максимального размера кадра, который она может обработать. Продвигающие мосты должны устанавливать биты СДК для указания наибольшего значения, которое не должно превышать наименьшей из следующих величин:

1) указанного значения полученных битов СДК;

2) наибольшего размера СБДУД, обеспечиваемого мостом;

3) наибольшего размера СБДУД, обеспечиваемого портом, из которого был получен кадр;

4) наибольшего размера СБДУД, обеспечиваемого портом, в который будет передаваться кадр.

Станция-получатель может в дальнейшем уменьшить значение СДК для указания наибольшей емкости своего кадра.

Кодирование битов СДК образуется из 3-битового базового кодирования и 3-битового расширенного кодирования (см. рисунок С.4).



Обозначения: Н — бит направления; б — бит базового значения; р — бит расширения; г — зарезервированнный бит

Рисунок С.4 — Базовые биты и биты расширения области СДК второго октета поля УМ

Значения базового кодирования битов СДК и их обоснование приведены на рисунке С.5. Если указатель режима СДК установлен в значение «базовый режим», мост должен установить биты СДК в кадрах-анализаторах в соответствии с базовым значением самого длинного кадра (см. таблицу С.1). Если указатель режима СДК установлен в значение «режим расширения», мост должен установить биты СДК в кадрах-анализаторах в соответствии с расширенным значением самого длинного кадра (см. таблицу С.2).

000:	516	октетов	(протокол по ГОСТ Р 34.1952 плюс протокол УЛЗ)
001:	1470	октетов	(протокол по ГОСТ 34.913.3)
010:	2052	октета	(80 × 24 знаков экрана с управляющими символами)
011:	4399	октетов	(протокол по ГОСТ Р ИСО/МЭК 8802-5 для скорости 4 Мбит/с и протокол по ИСО 9314-2)
100:	8130	октетов	(протокол по ГОСТ 34.913.4)
101:	11407	октетов	(протокол по ГОСТ Р ИСО/МЭК 8802-5 при незащищенных 4-битовых пакетах ошибок)
110:	17749	октетов	(протокол по ГОСТ Р ИСО/МЭК 8802-5 для скорости 16 Мбит/с)
111:	41600	октетов	(базовое значение при расширении до 65535 октетов)

Рисунок С.5 — Базовые значения самого длинного кадра и их обоснование

Таблица С.1 — Базовые значения самого длинного кадра

Битовая комбинация базового режима	Значение	Битовая комбинация базового режима	Значение
000	516 октетов	100	8130 октетов
001	1470 октетов	101	11407 октетов
010	2052 октета	110	17749 октетов
011	4339 октетов	111	> 17749 октетов

Таблица С.2 — Расширенные значения самого длинного кадра

Битовые комбинации базового режима	Битовые комбинации режима расширения								
		000	101	010	011	100	101	110	111
	000	516	635	754	873	993	1112	1231	1350
	001	1470	1542	1615	1688	1761	1833	1906	1979
	010	2052	2345	2638	2932	3225	3518	3812	4105
	011	4399	4865	5331	5798	6264	6730	7197	7663
	100	8130	8539	8949	9358	9768	10178	10587	10997
	101	11407	12199	12992	13785	14578	15370	16163	16956
	110	17749	20730	23711	26693	29674	32655	35637	38618
	111	41600	44591	47583	50575	53567	56559	59551	> 59551

Значение СДК — положительное целое число
в (бзсдк + рзсдк × (сбзсдк — бзсдк)/8), где:

бзсдк — базовое значение, представленное в левой колонке 3 битами;

сбзсдк — следующее базовое значение СДК (следующее значение после 111 — 65535);

рзсдк — кодируемый бит расширенного СДК (от 0 до 7).

Примечание — Мосты не обязательно должны обеспечивать все значения из приведенного выше перечня. К тому же при маршрутизации со стороны отправителя оконечные системы объектов УДС с установленными максимальными размерами кадра не должны передавать кадры, в которых заголовок и хвостовик УДС, поля ИМ и поле информации превышают максимальный размер кадра для этого УДС.

В общем случае базовые значения длин в октетах выбраны таким образом, чтобы позволить выбирать оптимальные размеры кадров для передачи по различным физическим средам ЛВС. Эти размеры были вычислены путем вычитания максимального размера заголовка УДС и максимального размера поля информации маршрутизации (30 октетов) из максимального размера физического кадра (см. рисунок С.6). Значения, относящиеся к закодированным битам, представляют максимальную длину поля информации УДС (см. рисунок С.6). Биты расширения используются для обеспечения 7 промежуточных значений между каждым набором базовых значений. Уравнение для вычисления этих значений и фактически вычисленные значения приведены в таблице С.2.

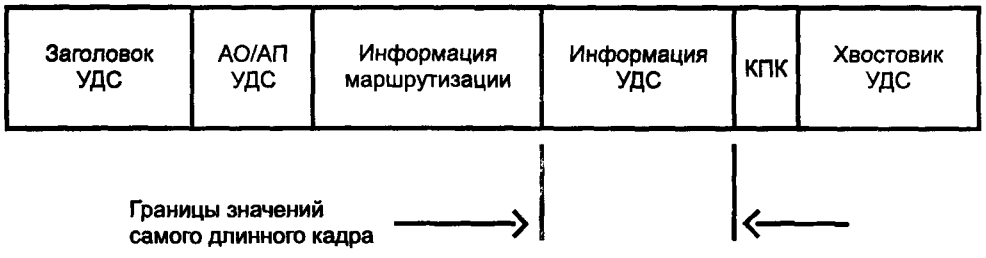


Рисунок С.6 — Границы значений самого длинного кадра

Поле «дескриптор маршрута» (ДМ)
Это поле изображено на рисунке С.7.

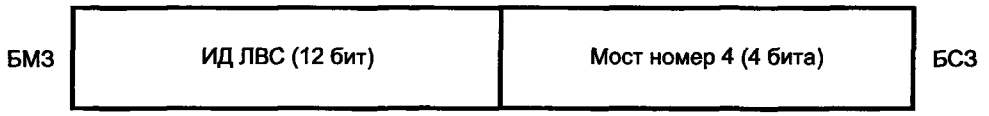


Рисунок С.7 — Поле «дескриптор маршрута»

Последовательность полей ДМ определяет конкретный маршрут через сеть. Каждое поле ДМ содержит уникальный для сети 12-битовый ИД ЛВС, плюс 4-битовый номер моста, который используется для различия двух или более мостов, подсоединенных к одним и тем же двум ЛВС (параллельные мосты). Последний номер моста в поле информации маршрутизации зарезервирован и устанавливается в нуль.

Зарезервированные биты (r). Все зарезервированные биты (r) игнорируются при приеме и передаются в том виде, в котором они были получены.

С.3.3.3 *Типы кадров, маршрутизируемых со стороны отправителя*

Кадры обрабатываются мостами, основываясь на типе маршрутизации (ТМ) кадра. ТМ кадра, подлежащего обработке, указывается в подполе ТМ поля ИМ кадра. Каждый тип маршрутизации определен в следующих подразделах.

С.3.3.3.1 Тип кадра «маршрутизируемый по конкретному маршруту» (ТМ = 0XX)

Поле ДМ содержит конкретный маршрут через сеть. Этот тип используется при обычной маршрутизации кадров данных от отправителя к получателю.

С. 3.3.3.2 Тип кадра «анализатор всех маршрутов» (ТМ = 10X)

Этот кадр передается по всем неповторяющимся маршрутам сети. В результате применения этого кадра, называемого «анализатором всех маршрутов», на станцию назначения поступит столько кадров, сколько имеется допустимых маршрутов к этой станции. По мере продвижения кадра мост добавляет к нему информацию маршрутизации.

С.3.3.3.3 Тип кадра «анализатор покрывающего дерева» (ТМ = 11X).

Кадры АПД продвигаются (логическими схемами маршрутизации со стороны отправителя) только через те порты моста, которые находятся в состоянии продвижения (т. е. по всему покрывающему дереву). В результате кадры поступают на свои станции назначения только по одному разу. Мосты добавляют к этим кадрам дескрипторы маршрутов и фильтрация адресов получателя не требуется. Эти кадры пересекают смежную часть покрывающего дерева, которая подключена мостами ПМСО, приобретая по мере своего продвижения сведения о маршруте.

С.3.3.4 *Обработка мостом кадров, маршрутизируемых со стороны отправителя*

Обработка таких кадров мостами основывается на двустадийной процедуре: прием кадра и продвижение кадра. Четкие определения этих функций приведены в С.3.5 и С.3.7 соответственно. Процесс приема кадра отражает указание кадра для ретранслирующего объекта УДС со стороны логического объекта УДС. Процесс продвижения кадра отражает внутреннюю проверку кадров, которые подвергаются копированию, возможному добавлению информации маршрутизации и действиям, выполняемым для продвижения кадров в другой сегмент ЛВС.

С.3.4 Информация состояния порта

Состояния порта описаны в 4.4.

С.3.5 Прием кадров

Если кадр не имеет информации маршрутизации (УИМ = 0), мост должен обрабатывать кадр в соответствии с положениями раздела 3. Если кадр имеет информацию маршрутизации (УИМ = 1), кадр должен обрабатываться в соответствии с положениями С.7.

С.3.6 Передача кадров

Кадры передаются в соответствии с изложенным в 3.6.

С.3.7 Продвижение кадров

Решение о продвижении кадра основывается на содержимом поля ИМ и на внутреннем состоянии моста. Кадры с заданным приоритетом пользователя и типом кадра должны продвигаться в той последовательности, в которой они копировались из сегмента ЛВС. Процесс продвижения продвигает полученные кадры, которые подлежат ретрансляции, в другие порты моста. Кадры, не маршрутизируемые со стороны отправителя (УИМ = 0), должны продвигаться в соответствии с 3.7. Обработка кадров, маршрутизируемых со стороны отправителя (УИМ = 1), описана ниже. В следующих подпунктах описываются действия моста ПМСО при наличии одной разрешенной пары портов моста (ВхЛВС и ИсхЛВС). Многопортовые мосты содержат несколько пар портов моста, и каждая пара мостов должна функционировать в соответствии с изложенным в данном разделе. Кадры не должны продвигаться через деактивизированные ППМ.

С.3.7.1 *Кадры данных, маршрутизируемые по конкретному маршруту (ТМ = 0XX)*

Мосты, которые обнаруживают в кадре соответствие конфигурации поле ИМ-мост (комбинации ИВхЛВС, НМ, ИИсхЛВС), должны продвигать такие кадры без изменения ИМ при условии, что оба логических объекта УДС используют один и тот же формат кадров (например мост имеет входящим и исходящим кольцо с маркерным методом доступа по ГОСТ Р ИСО/МЭК 8802-5). Если в поле информации маршрутизации кадра ИИЛВС встречается несколько раз, мост должен аннулировать такой кадр и увеличить значение счетчика ДублИИЛВС, если счетчик реализован. Следует иметь ввиду, что КМСО продвигаются или аннулируются на основе таблиц состояний независимо от того, является ли адрес получателя индивидуальным или групповым. К таблице С.3 необходимо дать следующие пояснения.

Таблица С.3 — Таблица состояния продвижения кадра, маршрутизируемого по конкретному маршруту

Ссылка	Условие	Действие
КМСО1	ИВхЛВС—НМ—ИИсхЛВС не совпадает	Не продвигать кадр
КМСО2	(ИВхЛВС—НМ—ИИсхЛВС совпадает) И (наличие ИИсхЛВС = 1) И (ДЛ ИМ = четное число)	Продвижение кадра в ИсхЛВС; увеличение значения счетчика (продвижение КМСО)
КМСО3	(ИВхЛВС—НМ—ИИсхЛВС совпадает) И ((ДЛ ИМ = нечетное число) ИЛИ (ДЛ ИМ = 0))	Аннулирование кадра; увеличение значения счетчика (неправильная ИМ)
КМСО4	(ИВхЛВС—НМ—ИИсхЛВС совпадает) И (наличие ИИсхЛВС > 1)	Аннулирование кадра; увеличение значения счетчика (ДублИИсхЛВС))

Для всех приведенных выше случаев:
УИМ =1, ТМ = 0XX и состояние ППМ — активизированное.
Определения:
аннулирование — кадр не продвигается ни в одну из исходящих ЛВС;
продвижение — кадр передается в исходящую ЛВС.

(КМСО1). Если кадр, маршрутизируемый по конкретному маршруту, получен мостом ПМСО и в поле информации маршрутизации не обнаружено совпадающего сочетания ИВхЛВС—НМ—ИИсхЛВС, этот кадр не должен продвигаться в порт, указанный ИИсхЛВС.

(КМСО2). Если кадр, маршрутизируемый по конкретному маршруту, получен мостом ПМСО и ИВхЛВС—НМ—ИИсхЛВС совпадает, в поле информации маршрутизации нет частых появлений ИИсхЛВС и поле длины имеет четное значение, такой кадр должен быть продвинут в порт и должно быть увеличено значение счетчика «продвижение КМСО», если этот счетчик реализован.

(КМСО3). Если кадр, маршрутизируемый по конкретному маршруту, получен мостом ПМСО и ИВхЛВС—НМ—ИИсхЛВС совпадает, но поле длины имеет нулевое или нечетное значение, такой кадр должен быть аннулирован и должно быть увеличено значение счетчика «неправильная ИМ», если этот счетчик реализован.

(КМСО4). Если кадр, маршрутизируемый по конкретному маршруту, получен мостом ПМСО и ИВхЛВС—НМ—ИИсхЛВС совпадает, но в поле информации маршрутизации часто встречается ИИсхЛВС, такой кадр должен быть аннулирован и должно быть увеличено значение счетчика «ДублИИсхЛВС», если этот счетчик реализован.

С.3.7.2 *Кадры «анализатор всех маршрутов» (ТМ = 10X)*

Для того, чтобы ограничить ненужный трафик, мостам ПМСО разрешается отфильтровывать кадры АВМ, которые являются ненужными или избыточными. Многопортовые мосты ПО могут аннулировать (отфильтровывать) кадры АВМ, которые:

1) уже проходили через мост. Они могут быть обнаружены путем просмотра комбинации ИВхЛВС—НМ—ИИсхЛВС в поле информации маршрутизации полученного кадра;
2) уже проходили через ЛВС, подключенную к этому мосту. Это условие более строгое, чем изложенное в 1). Такие кадры могут быть обнаружены путем анализа поля ИМ полученного кадра на соответствие ИД ЛВС любой из ЛВС, подключенной к мосту, но отличной от ИВхЛВС.

Все мосты ПМСО должны продвигать все остальные кадры с каждым ИИсхЛВС, кроме случаев наличия одного из следующих условий:

1) ИИсхЛВС уже имеется в поле ИМ;
2) последний ИД ЛВС в поле ИМ не соответствует ИВхЛВС;
3) число дескрипторов маршрутов равно предельному числу АВМ или превышает его;
4) длина поля ИМ имеет нечетное значение;

5) длина поля ИМ равна нулю;

6) длина поля ИМ равна четырем (факультативное значение — см. таблицу С.4);

7) бит направления не равен нулю.

При наличии любого из этих условий кадр должен быть аннулирован. При наличии условия 2 увеличивается значение счетчика «несовпадение ИД ЛВС». При наличии условия 4, 5 или 7 увеличивается значение счетчика «неправильная ИМ».

При отсутствии любого из этих условий:

1) если поле ДЛ = 2, мост вводит ДМ с ИВхЛВС в поле ИМ и

2) мост вводит в кадр свой номер моста и номер ИИсхЛВС.

Мост также увеличивает поле длины на два для каждого вводимого им дескриптора маршрута. Перед продвижением кадра мост устанавливает поле «самый длинный кадр» (СДК) в значение, отражающее передающую способность моста, если передающая способность пары портов моста меньше величины, указанной принятым СДК. Передающая способность моста принимает наименьшее значение из следующих величин:

1) максимального размера кадра ВхЛВС;

2) максимального размера кадра ИсхЛВС и

3) максимального размера кадра, который может быть обработан самим мостом.

И наконец, вычисляется КПК. Кадры АВМ обрабатываются в соответствии с таблицей С.4. К таблице С.4 необходимо дать следующие пояснения.

Таблица С.4 — Таблица состояний продвижения кадра «анализатор всех маршрутов»

Ссылка	Условие	Действие
АВМ1	(ИИсхЛВС уже в поле ИМ) ИЛИ (мост в режиме фильтрации)	Не продвигать кадр в ИсхЛВС
АВМ2	Последний ИД ЛВС ¬ = ИВхЛВС	Аннулирование кадра; увеличение значения счетчика (не совпадает ИД ЛВС)
АВМ3	Количество ДМ ≥ ПредельноеЧислоАВМ	Не продвигать; увеличение значения счетчика (превышено предельное число ДМ АВМ)
АВМ4	(ДЛ ИМ = нечетное число) ИЛИ (ДЛ ИМ = 0) ИЛИ (Н ¬ = 0)	Аннулирование кадра; увеличение значения счетчика (неправильная ИМ)
АВМ5	(ДЛ ИМ = 2) И (мост не в состоянии фильтрации) И (предельноеЧисло ДМ ИВС > 1)	Добавление ИВхЛВС, НМ, ИИсхЛВС в поле ИМ; УСТАНОВИТЬ ДЛ ИМ = 6; УСТАНОВИТЬ СДК = НАИМЕНЬШИЙ ИЗ (СДК, ВхЛВС, Мост, ИсхЛВС); пересчет КПК; продвижение кадра; увеличение значения счетчика (продвижение кадров АВМ)
АВМ6	(ДЛ ИМ = 4) И (мост не в состоянии фильтрации) И (последний ИД ЛВС в ИМ = ИВхЛВС) И (предельноеЧисло ДМ ИВС > 1)	Аннулирование кадра; увеличение значения счетчика (неправильная ИМ), ИЛИ (факультативно) Включение НМ, ИИсхЛВС в поле ИМ; ДЛ ИМ = ДЛ ИМ + 2 УСТАНОВИТЬ СДК = МИНИМАЛЬНЫЙ ИЗ (СДК, ВхЛВС, Мост, ИсхЛВС); пересчет КПК; продвижение кадра; увеличение значения счетчика (продвижение кадров АВМ)
АВМ7	(ДЛ ИМ = четное число от 6 до 28 включ.) И (ИИсхЛВС нет в ИМ) И (мост не в состоянии фильтрации) И (последний ИД ЛВС в ИМ = ИВхЛВС) И (количество ОП < предела ДМ АВМ)	Включение ИВхЛВС, НМ, ИИсхЛВС в поле ИМ; ДЛ ИМ = ДЛ ИМ + 2; УСТАНОВИТЬ СДК = МИНИМАЛЬНЫЙ ИЗ (СДК, ВхЛВС, Мост, ИсхЛВС); пересчет КПК; продвижение кадра; увеличение значения счетчика (продвижение кадров АВМ)

Для всех приведенных выше случаев:

УИМ =1, ТМ = 10Х и состояние ППМ — активизированное.

Определения:

аннулирование — кадр не продвигается ни в одну из исходящих ЛВС;

продвижение — кадр передается в исходящую ЛВС.

(АВМ1). Если мост ПМСО получил кадр «анализатор всех маршрутов» с уже имеющимся ИД исходящей ЛВС в поле ИМ или если мост отфильтровывает кадр, этот кадр не должен продвигаться в исходящую ЛВС, указанную ИД исходящей ЛВС.

(АВМ2). Если мост ПМСО получил кадр «анализатор всех маршрутов» с последним ИД ЛВС, не соответствующим ИВхЛВС, этот кадр должен быть аннулирован и должно быть увеличено значение счетчика «несоответствие ИД ЛВС», если этот счетчик реализован.

(АВМ3). Если мост ПМСО получил кадр «анализатор всех маршрутов» с количеством ДМ, соответствующим или превышающим «предельное число ДМ АВМ» порта, указанного ИИсхЛВС, это кадр не должен продвигаться и должно быть увеличено значение счетчика «превышение предельного числа ДМ АВМ» порта, указанного ИИсхЛВС, если этот счетчик реализован.

(АВМ4). Если мост ПМСО получил кадр «анализатор всех маршрутов» ПМСО с полем длины, равным нулю или нечетному числу, или если бит направления не равен нулю, этот кадр должен быть аннулирован и должно быть увеличено значение счетчика «неправильная ИМ», если этот счетчик реализован.

(АВМ5). Если мост ПМСО получил действительный кадр «анализаторы всех маршрутов» с полем длины, равным 2, и предельное значение ДМ АВМ больше 1, мост должен модифицировать этот кадр по мере его продвижения путем введения своего ИД входящей ЛВС (ИВхЛВС), своего номера моста (НМ), своего ИД исходящей ЛВС (ИИсхЛВС) и 4 битов в нулевом значении, установив поле ДЛ в значение 6, поле СДК ИМ — в наименьшее из значений полученного СДК и передающей способности моста и пересчитав КПК, после чего кадр ставится в очередь на передачу в свою исходящую ЛВС и увеличивается значение счетчика «продвижение кадра АВМ», если этот счетчик реализован.

(АВМ6). Если мост ПМСО получил действительный кадр «анализатор всех маршрутов» с длиной ИМ, равной 4, и если предельное значение ДМ АВМ больше 1, то он должен либо:

1) аннулировать кадр и увеличить значение счетчика «неправильная ИМ», если этот счетчик реализован, либо

2) модифицировать кадр по мере его продвижения, установив последние 4 бита предыдущего ДМ в свой номер моста (НМ), добавив следующую двухоктетную область ДМ, состоящую из номера исходящей ЛВС (ИИсхЛВС) и оставшихся 4 битов в значении нуль, увеличив значение поля длины на 2, установив поле самого длинного кадра в минимальное из значений полученного кадра и передающей способности моста, пересчитав КПК, поставить кадр в очередь на передачу в свою исходящую ЛВС и увеличить значение счетчика «продвижение кадра АВМ», если этот счетчик реализован.

(АВМ7). Если мост ПМСО получил действительный кадр «анализатор всех маршрутов» с полем длины от 6 до 28 (содержащем только четные числа) и числом ДМ меньшим предельного числа ДМ АВМ, мост должен модифицировать этот кадр по мере его продвижения, установив последние 4 бита предыдущего ДМ в значение своего номера моста (НМ), добавив следующую двухоктетную область ДМ, состоящую из номера исходящей ЛВС (ИИсхЛВС) и оставшихся 4 битов в значении нуль, увеличив значение поля длины на 2, установив поле «самый длинный кадр» в наименьшее из значений полученного СДК и передающей способности моста, пересчитав КПК, поставить в очередь на передачу в свою исходящую ЛВС и увеличить значение счетчика «продвижение кадров АВМ», если этот счетчик реализован.

С.3.7.3 *Кадры АПД*

Условия для продвижения кадров АПД идентичны условиям продвижения кадров АВМ, за следующими исключениями:

1) нет необходимости отфильтровывать анализаторы из-за наличия алгоритма покрывающего дерева;

2) продвижение обусловливается также состоянием порта исходящей ЛВС и состоянием покрывающего дерева.

Фильтрация адресов назначения для кадров АПД не обязательна. Эти кадры должны пересекать полное покрывающее дерево. Кадры АПД обрабатываются, как указано в таблице С.5.

Таблица С.5 — Таблица состояний продвижения кадра «анализатор покрывающего дерева»

Ссылка	Состояние порта ВхЛВС	ИСхЛВС	Условие	Действие
АПД1	П	П	(ИИсхЛВС уже имеется в поле ИМ)	Аннулирование кадра; увеличение значения счетчика (дублирование ЛМ АПД или ошибка дерева)
АПД2	П	—	Последний ИД ЛВС в ИМ ¬ = ИВхЛВС	Аннулирование кадра; увеличение значения счетчика (несовпадение ИД ЛВС)
АПД3	П	П	Количество ДМ ≥ предельного числа ДМ АПД	Аннулировать кадр; увеличение значения счетчика (превышение предельного числа ДМ АПД)

Окончание таблицы 5

Ссылка	Состояние порта ВхЛВС	Состояние порта ИСхЛВС	Условие	Деистие
АПД4	П	—	(Н ¬ = 0) ИЛИ (ДЛ ИМ = 0) ИЛИ (ДЛ ИМ = нечетное число)	Не продвигать, увеличение значения счетчика (неправильная ИМ)
АПД5	П	П	(ДЛ ИМ = 2) И (предел ДМ АПД > 1)	Включение ИВхЛВС, НМ, ИИсхЛВС в поле ИМ, УСТАНОВИТЬ ДЛ ИМ = 6, УСТАНОВИТЬ СДК = НАИМЕНЬШИЙ ИЗ (СДК, ВхЛВС, мост, ИсхЛВС), пересчет КПК, продвижение кадра, увеличение значения счетчика (продвижение кадров АПД)
АПД6	П	П	(ДЛ ИМ = 4) И (последний ИД ЛВС в ИМ = ИВхЛВС) И (предел ДМ АПД > 1)	Аннулирование кадра увеличение значения счетчика (неправильная ИМ), ИЛИ (факультативно) включение НМ, ИИсхЛВС в поле ИМ, ДЛ ИМ = ДЛ ИМ + 2, УСТАНОВИТЬ СДК = НАИМЕНЬШЕЕ ИЗ (СДК, ВхЛВС, мост, ИсхЛВС), пересчет КПК, продвижение кадра, увеличение значения счетчика (продвижение кадров АПД)
АПД7	П	П	(ДЛ ИМ = четное число между 6 и 28 включительно) И (ИИсхЛВС нет в ИМ) И (последний ИД ЛВС в ИМ = ИВхЛВС) И (количество ДМ < предела ДМ АПД)	Включение НМ, ИИсхЛВС в поле ИМ, ДЛ ИМ = ДЛ ИМ + 2, УСТАНОВИТЬ СДК = НАИМЕНЬШИЙ ИЗ (СДК, ВхЛВС, мост ИсхЛВС), пересчет КПК, продвижение кадра, увеличение значения счетчика (продвижение кадров АПД)
АПД8	— П	—		Не продвигать в ИсхЛВС
АПД9	П	— П		Не продвигать в ИсхЛВС

Для всех приведенных выше случаев
УИМ =1, ТМ = 11Х и состояние ППМ — активизированное
Определения
аннулирование — кадр не продвигается ни в одну из исходящих ЛВС,
продвижение — кадр передается в исходящую ЛВС,
П — порт в состоянии продвижения,
— П — порт не в состоянии продвижения,
— — порт в любом состоянии

К таблице С 5 необходимо дать следующие пояснения

(АПД1) Если мост ПМСО получил кадр «анализатор покрывающего дерева», порт исходящей ЛВС и порт входящеи ЛВС находятся в состоянии продвижения, а в поле ИМ уже имеется ИИсхЛВС, этот кадр должен быть аннулирован и мост должен увеличить значение счетчика «дублирование ДИ ЛВС или ошибка дерева», если этот счетчик реализован

(АПД2) Если мост ПМСО получил кадр «анализатор покрывающего дерева», порт входящей ЛВС находится в состоянии продвижения и последнии ИД ЛВС не совпадает с ИВхЛВС, этот кадр должен быть аннулирован и должно быть увеличено значение счетчика «не совпадение ИД ЛВС», если этот счетчик реализован

(АПД3) Если мост ПМСО получил кадр «анализатор покрывающего дерева», порт входящеи ЛВС и порт исходящей ЛВС находятся в состоянии продвижения и число ДМ равно предельному числу ДМ АПД порта, указанного ИИсхЛВС, или превышает его, этот кадр не должен продвигаться в ИсхЛВС и должно быть увеличено значение счетчика «превышение предела ДМ АПД» порта указанного ИИсхЛВС если этот счетчик реализован

(АПД4) Если мост ПМСО получил кадр «анализатор покрывающего дерева», порт входящей ЛВС находится в состоянии продвижения и поле длины равно нулю или нечетному числу, или бит направления не равен нулю, этот кадр должен быть аннулирован и должно быть увеличено значение счетчика «неправильная ИМ», если этот счетчик реализован

(АПД5). Если мост ПМСО с портами входящей и исходящей ЛВС, находящимися в состоянии продвижения, получил правильный кадр «анализатор покрывающего дерева» с полем длины, равным 2, и предельное число ДМ АПД больше 1, то мост ПМСО должен модифицировать кадр по мере продвижения, введя свой ИД входящей ЛВС (ИВхЛВС), свой номер моста (НМ), свой ИД исходящей ЛВС (ИИсхЛВС) и 4 бита в значении нуль. Мост ПМСО должен также установить поле длины в значение 6, поле самого длинного кадра — в наименьшее из значений полученного СДК и передающей способности моста и пересчитать КПК. Затем мост ПМСО должен поставить кадр в очередь на передачу в свою исходящую ЛВС и увеличить значение счетчика «продвижение кадров АПД», если этот счетчик реализован.

(АПД6). Если мост ПМСО с портами входящей и исходящей ЛВС, находящимися в состоянии продвижения, получил правильный кадр «анализатор покрывающего дерева» с полем длины в значении 4 и предельное число ДМ АПД больше 1, мост должен либо:

1) аннулировать кадр и увеличить значение счетчика «неправильная ИМ», если этот счетчик реализован, или

2) модифицировать кадр по мере продвижения, установив последние 4 бита предыдущего ДМ в значение своего номера моста (НМ), введя следующую двухоктетную область ДМ, состоящую из идентификатора исходящей ЛВС (ИИсхЛВС) и оставшихся 4 битов в значении нуль. Мост ПМСО должен также увеличить длину на 2, установить поле самого длинного кадра в наименьшее из значений полученного СДК и передающей способности моста и пересчитать КПК. После этого мост ПМСО должен поставить кадр в очередь на передачу в свою исходящую ЛВС и увеличить значение счетчика «продвижение кадров АПД», если этот счетчик реализован.

(АПД7). Если мост ПМСО с портами входящей и исходящей ЛВС, находящимися в состоянии продвижения, получил правильный кадр «анализатор покрывающего дерева» с полем длины от 6 до 28 включ. (только четные числа) и количество ДМ меньше предельного числа ДМ АПД, мост должен модифицировать кадр по мере его продвижения, установив последние 4 бита предыдущего ДМ в значение своего номера моста (НМ), введя следующую двухоктетную область ДМ, состоящую из идентификатора исходящей ЛВС (ИИсхЛВС) и оставшихся 4 битов в значении нуль. Мост ПМСО должен также увеличить длину на 2, установить поле самого длинного кадра в наименьшее из значений полученного СДК и передающей способности моста и пересчитать КПК. После этого мост ПМСО должен поставить кадр в очередь на передачу в свою исходящую ЛВС и увеличить значение счетчика «продвижение кадров АПД», если этот счетчик реализован.

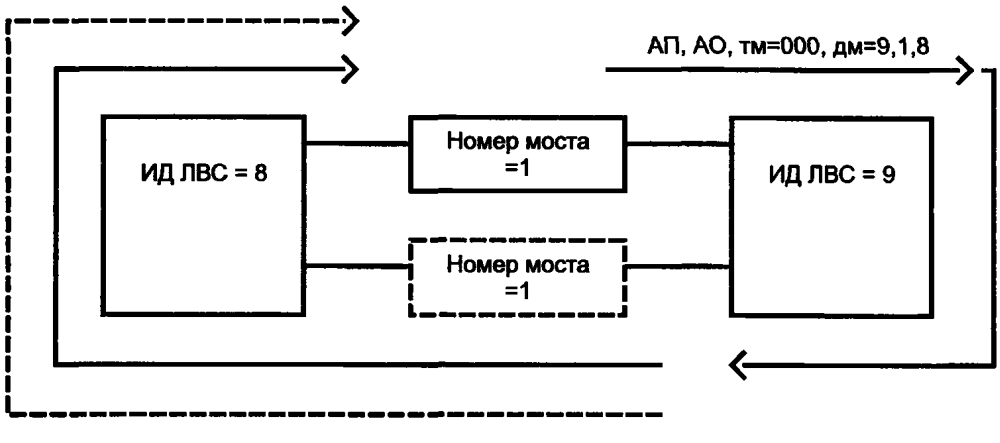
(АПД8). Мост ПМСО должен только продвигать полученные кадры «анализатор покрывающего дерева» в порт, если порт находится в состоянии продвижения.

(АПД9). Мост ПМСО должен только продвигать кадры «анализатор покрывающего дерева» в порт, если порт находится в состоянии продвижения.

С.3.7.4 *Проверка дублирования номера моста*

Диспетчер моста может осуществлять проверку на наличие присвоения одного и того же номера своему мосту и параллельным мостам путем использования приведенной ниже методики.

Для каждой из двух ЛВС, соединенных мостом, диспетчер моста инициирует передачу ПБД УЛЗ «ТЕСТ» с адресом получателя, установленным в значение адреса УДС порта, подключенного к одной ЛВС, и адресом отправителя, установленным в значение адреса УДС порта, подключенного к другой ЛВС. ПБД «ТЕСТ» должен содержать маршрут, состоящий из входящей ЛВС, соответствующей адресу отправителя, номера моста и исходящей ЛВС, соответствующей адресу отправителя. Если диспетчер моста получает более одного ответа на этот ПБД «ТЕСТ», это означает наличие параллельного моста с тем же номером моста, и об этом должен быть проинформирован диспетчер сети (см. рисунок С. 8).



Обозначения:
АП — адрес получателя; ТМ — поле типа маршрутизации; АО — адрес отправителя; ДМ — дескриптор маршрута.

Рисунок С.8 — Иллюстрация проверки дублирования номера моста

С 3 7 5 *Очередность кадров*

Процесс продвижения поддерживает очередность кадров, ожидающих возможности передачи в отдельный логический объект УДС, связанныи с каждым портом моста Должна выдерживаться очередность передачи кадров

Кадр, поставленный в очередь процессом продвижения для передачи в порт, при передаче его в отдельный логический объект УДС этого порта должен быть удален из этой очереди, в дальнейшем не должно предприниматься никаких попыток передать этот же кадр в данный порт даже в том случае, если первая попытка передачи оказалась безуспешной

Кадр, поставленныи процессом продвижения в очередь на передачу в порт, должен быть удален из этой очереди и впоследствии не передаваться, если время, в течение которого гарантируется буферизация, для этого кадра истекло

Кадр, поставленный процессом продвижения в очередь на передачу в порт, должен быть удален из этой очереди и впоследствии не направляться в отдельный логический объект УДС этого порта, если нельзя гарантировать, что максимальная мостовая транзитная задержка (см 2 3 6) не превысит время, в течение которого впоследствии может быть передан этот кадр

Кадр АПД, поставленный в очередь на передачу в порт, должен быть удален из этой очереди, если связанный с ним порт выходит из состояния продвижения

Удаление кадра из очереди на передачу в любой конкретный порт еще не означает, что он должен быть удален из очереди на передачу в любои другой порт

С 3 7 6 *Преобразование приоритетов*

Процесс продвижения при маршрутизации со стороны отправителя определяет значения параметров «приоритет пользователя» и «приоритет доступа», используемых для ретрансляции кадров в соответствии с 3 7 7

С 3 8 Адресация

В мостовых ЛВС с маршрутизацией со стороны отправителя каждой ЛВС и каждому мосту должен быть присвоен номер в соответствии с положениями следующих подпунктов

С 3 8 1 *ИД ЛВС*

Каждой отдельной ЛВС в сети, состоящей из нескольких ЛВС, должен быть присвоен уникальный 12-битовый ненулевои ИД ЛВС, который распознается всеми мостами, подсоединенными к данной ЛВС

С 3 8 2 *Номер моста*

Каждый мост должен иметь номер моста (НМ) Последовательность дескрипторов маршрута (ИВхЛВС — НМ—ИИсхЛВС) должна быть уникальной для каждого маршрута между двумя ЛВС, соединенными мостом

С 3 8 3 *Дескриптор маршрута*

Дескриптор маршрута представляет собой двухоктетное поле (16 битов) Первые 12 битов представляют ИД ЛВС, а следующие 4 бита — номер моста (подробное описание формата кадра приведено в С 3 3 2) Последовательность дескрипторов маршрута определяет уникальный маршрут через мостовую ЛВС Поскольку концом маршрута является ЛВС, а не мост, то часть отдельного моста дескриптора последнего маршрута в поле информации маршрутизации должна быть зарезервирована и установлена в ноль В разделе С 3 7 описан метод построения и продвижения кадров при маршрутизации со стороны отправителя

С 4 **Диспетчер моста**

Для средств диспетчеризации, определенных в разделе 6, выполняются расширения, приведенные в последующих пунктах Они позволяют управлять маршрутами через мостовую ЛВС при маршрутизации со стороны отправителя

С 4 1 Логический объект диспетчера моста

Для логических объектов диспетчера моста, определенных в 6 4, выполняются следующие расширения

С 4 1 1 *Конфигуратор моста*

С 4 1 1 1 *Открытие моста*

Это расширение позволяет операциям административного управления «открыть мост», определенный в 6 4 1 1, сообщить о возможности моста передавать в режиме ПМСО, т е ввести соответствующий параметр в тип значения «конфигуратор моста»

С 4 1 1 1 1 Назначение

Информирование о возможности моста передавать в режиме ПМСО

С 4 1 1 1 2 Дополнительные входы

Отсутствуют

С 4 1 1 1 3 Дополнительные выходы

6) Передающая пропускная способность ПМСО — значение поля, указывающее размер самого длинного ПБДУД кадра, который может быть обработан ретранслирующим объектом УДС моста ПМСО Если режим ПМСО не обеспечивается мостом, это значение должно быть сообщено как нулевое

С 4 1 1 2 *Чтение моста*

Чтобы обеспечить возможность операциям административного управления «чтение моста», определенным в 6 4 1 1, сообщать о передающей пропускной способности моста передавать в режиме ПМСО, последующий параметр вводится в тип значения «конфигуратор моста»

С 4 1 1 2 1 Назначение

Информирование о передающей пропускной способности моста в режиме ПМСО

С 4 1 1 2 2 Дополнительные входы

Отсутствуют

С 4 1 1 2 3 Дополнительные выходы

7) Передающая пропускная способность ПМСО — значение поля, указывающее размер самого длинного ПБДУД кадра, который может быть обработан ретранслирующим объектом УДС моста ПМСО Если режим ПМСО не обеспечивается мостом, это значение должно быть сообщено как нулевое

С 4 2 Процесс продвижения

Для процесса продвижения, определенного в 6 6, выполняются следующие расширения

С 4 2 1 *Счетчики порта*

См 6 6 1

С 4 2 1 1 *Чтение счетчиков продвижения порта*

С 4 2 1 1 1 Назначение

Объекты счетчиков порта моста ПМСО состоят из счетчиков, определенных в 6 6 1 1, и счетчиков, специфичных для режима ПМСО и приведенных в последующих подпунктах

С 4 2 1 1 2 Дополнительные входы

Отсутствуют

С 4 2 1 1 3 Дополнительные выходы

К перечню выходов, определенных в 6 6 1 1 3, добавляются следующие счетчики порта Если соответствующий счетчик не обеспечивается, его значение должно быть сообщено как нулевое

8) Недействительная ИМ — число кадров, причиной аннулирования которых была ошибка формата (т е длина ИМ — нечетное число или нуль)

9) Дублирование ИИсхЛВС — число кадров, аннулированных по причине дублирования ИИсхЛВС в КМСО

10) Несовпадение ИД ЛВС — число кадров АВМ и АПД, которые были аннулированы из-за того, что последний ИД ЛВС в поле информации маршрутизации не был равен ИД входящеи ЛВС

11) Дублирование ИД ЛВС или ошибка дерева — число кадров АПД, которые были аннулированы из-за того, что ИД исходящей ЛВС уже имелся в поле информации маршрутизации

12) Превышено предельное число ДМ АПД — число кадров АПД, причиной аннулирования которых явилось превышение предельного числа ДМ АПД

13) Превышен предел ДМ АВМ — число кадров АВМ, причинои аннулирования которых явилось превышение предельного числа ДМ АВМ

14) Продвинутые КМСО — число прибывших КМСО, продвинутых из одного порта моста в другой

15) Продвинутые кадры АПД — число поступивших кадров АПД, продвинутых из одного порта моста в другой

16) Продвинутые кадры АВМ — число поступивших кадров АВМ, продвинутых из одного порта моста в другой

С 4 3 Логический объект диспетчера моста ПМСО

С 4 3 1 *Конфигуратор моста ПМСО*

Операции конфигуратора моста ПМСО для логического объекта диспетчера моста ПМСО позволяют диспетчеру выполнять операции «чтение предельного числа ДМ моста», «установить предельное число ДМ моста», «чтение режима СДК моста» и «установить режим СДК моста»

С 4 3 1 1 *Читать «режим СДК»*

С 4 3 1 1 1 Назначение

Информирование о режиме, который мост использует для установки битов СДК в кадрах АВМ и АПД

С 4 3 1 1 2 Входы

Отсутствуют

С 4 3 1 1 3 Выходы

Режим СДК — значение поля, указывающее, какой режим использует мост для установки битов СДК базовый или расширенныи

С 4 3 1 2 *Установить «режим СДК»*

С 4 3 1 2 1 Назначение

Установка режима, которыи будет использован мостом для установки битов СДК в кадрах АВМ и АПД

С.4.3.1.2.2 Входы

Режим СДК — значение поля, указывающее, какой режим использует мост для установки битов СДК: базовый или расширенный.

С.4.3.1.2.3 Выходы

Отсутствуют.

С.4.3.2 *Конфигуратор порта ПМСО*

Операции конфигуратора порта ПМСО для логического объекта диспетчера моста ПМСО позволяет диспетчеру читать и устанавливать атрибуты ПМСО для каждого порта (ИД ЛВС и самый большой размер СБДУД). В сопоставлении с операциями, определенными в 6.8.2, над портом ПМСО могут быть выполнены операции, приведенные в следующих подпунктах.

С.4.3.2.1 *Чтение параметров порта*

С.4.3.2.1.1 Назначение

Получение информации, относящейся к конкретному порту внутри логического объекта протокола моста ПМСО.

С.4.3.2.1.2 Входы

Номер порта

С.4.3.2.1.3 Выходы

1) Тип порта ПМСО — значение поля, указывающее доступный тип порта. Существует два типа порта: ТМ и ПМСО.
2) ИД ЛВС — значение поля, указывающее идентификатор, соответствующий ЛВС, к которой подключен данный порт.
3) Самый большой размер СБДУД — значение поля, указывающее самый большой ПБДУД кадра, который может быть обработан данным портом.
4) Предельное число ДМ АВМ — значение поля, указывающее максимальное число дескрипторов маршрутов, которые разрешено содержать в кадре АВМ, продвигаемом мостом.
5) Предельное число ДМ АПД — значение поля, указывающее максимальное число дескрипторов маршрутов, которые разрешено содержать в кадре АПД, продвигаемом мостом.

С.4.3.2.2 *Установка ИД ЛВС*

С.4.3.2.2.1 Назначение

Логическая увязка ИД ЛВС с определенным портом моста.

С.4.3.2.2.2 Входы

1) Номер порта — номер порта моста.
2) ИД ЛВС — идентификатор соответствующей ЛВС, к которой подключен порт. Содержит 12 битов.

С.4.3.2.2.3 Выходы

Отсутствуют.

С.4.3.2.3 *Установка самого большого размера ПБДУД*

С.4.3.2.3.1 Назначение

Логическая увязка самого большого размера ПБДУД с портом моста.

С.4.3.2.3.2 Входы

1) Номер порта — номер порта моста.
2) Самый большой размер СБДУД — значение поля, указывающее максимальный размер ПБДУД, который может быть обработан этим портом.

С.4.3.2.3.3 Выходы

Отсутствуют.

С.4.3.2.4 *Установка предельного числа ДМ*

С.4.3.2.4.1 Назначение

Логическая увязка предельного числа дескрипторов маршрутов, читаемых операцией «чтение предельного числа ДМ», с кадрами АВМ и АПД, продвигаемыми мостом.

С.4.3.2.4.2 Входы

1) Номер порта — номер порта моста.
2) Тип предельного числа ДМ — значение поля, указывающее, к каким кадрам имеет отношение входящий предел: к АВМ или АПД.
3) Значение предельного числа ДМ — значение поля, определяющее максимальное число ДМ, которое может содержаться в поле ИМ кадра, тип которого определен полем «тип предельного числа ДМ». Это поле принимает значение от 0 до 14.

С.4.3.2.4.3 Выходы

Отсутствуют.

С.4.4 База данных пары портов моста ПМСО

Над мостом ПМСО могут быть выполнены следующие операции.

С.4.4.1 *Конфигуратор пары портов моста ПМСО*

С.4.4.1.1 *Чтение размера базы данных пары портов моста ПМСО*

С.4.4.1.1.1 Назначение

Чтение размера базы данных пары портов моста.

С.4.4.1.1.2 Входы

Отсутствуют.

С.4.4.1.1.3 Выходы

Размер базы данных — значение поля, указывающее количество записей в базе данных пары портов моста.

С.4.4.1.2 *Чтение записи в базе данных пары портов моста ПМСО*

С.4.4.1.2.1 Назначение

Чтение атрибутов записи в базе данных пары портов моста.

С.4.4.1.2.2 Входы

1) Номер порта младшей значимости.

2) Номер порта старшей значимости.

С.4.4.1.2.3 Выходы

1) Номер моста, работающего в режиме маршрутизации со стороны отправителя.

2) Состояние моста (открытое или блокировка).

С.4.4.1.3 *Установка записи в базе данных пары портов моста ПМСО*

С.4.4.1.3.1 Назначение

Установка атрибутов записи в базе данных пары портов моста.

С.4.4.1.2.2 Входы

1) Номер порта младшей значимости.

2) Номер порта старшей значимости.

3) Номер моста, работающего в режиме маршрутизации со стороны отправителя.

4) Состояние моста (открытое или блокировка).

С.4.4.1.2.3 Выходы

Отсутствуют

С.5 Протокол диспетчера

Этот вопрос подлежит дальнейшему изучению и его решение будет зависеть от последующих изменений в стандарте IEEE Std 802.1B.

Дополнительно к протоколу диспетчера, определенному в 7.2, в таблицах С.6 и С.7 указаны операции диспетчера моста, специфичные для моста ПМСО.

С.5.1 Операции

В таблицах С. 6 и С.7 определено преобразование операций диспетчера моста ПМСО для логического объекта диспетчера моста ПМСО (см. С.4.3) и логического объекта базы данных пары портов моста ПМСО (см. С.4.4).

Таблица С.6 — Преобразование операций логического объекта диспетчера моста ПМСО в операции протокола административного управления

Операция диспетчера	Операция протокола	ИД действия	Код идентификатора	Тип идентификатора	Тип значения
Установка предельного значения ДМ	Установка	—	srtBridgeEntity	PortRdLimId	RdLimit
Чтение режима СДК	Получение	—	srtBridgeEntity	LFModeId	LFMode
Установка режима СДК	Установка	—	srtBridgeEntity	LFModeId	LFMode
Чтение порта	Получение	—	srtBridgeEntity	PortNumber	PortInfo
Установка ИД ЛВС	Установка	—	srtBridgeEntity	PortLanId	LanId
Установка самого большого размера СБДУД	Установка	—	srtBridgeEntity	PortLargest MSDUSizeId	Largest MSDUSize

Таблица С.7 — Преобразование операций с базой данных пары портов моста ПМСО в операции протокола административного управления

Операция диспетчера	Операция протокола	ИД действия	Код идентификатора	Тип идентификатора	Тип значения
Чтение размера базы данных пары портов моста ПМСО	Получение	—	strBPPDatabase	SrtBPPDatabaseSize	SrtSize
Установка записи в базе данных пары портов моста ПМСО	Установка	—	strBPPDatabase	strBPPEntry	SrtBPP Param
Чтение записи в базе данных пары портов моста ПМСО	Получение	—	strBPPDatabase	strBPPEntry	SrtBPP Info

С.5.2 Кодирование

Дополнительно к кодированию параметров, определенных в 7.3, вводится следующее кодирование.

```
- - ********************************************************************************************************
- - * ДОПОЛНЕНИЯ К ПЕРЕЧНЮ идентификаторов параметров
- - * (коды определителя типа идентификатора и значений
- - * параметров)
- - ********************************************************************************************************
- - Если функция маршрутизации со стороны отправителя реализована, в логическом объекте диспетчера
- - моста условно присутствуют следующие параметры.

IEEE802-1Bridge-Lme-ParameterID : : = SEQUENCE {
    idCode INTEGER,
    idtypeQualifier CHOICE          {...,

    portRdLimId             [15] IMPLICIT PortRdLimId,
    portNumber              [16] IMPLICIT PortNumber,
    potrLanId               [17] IMPLICIT PortLanId,
    portLargestMSDUSizeId   [18] IMPLICIT PortLargestMSDUSizeId,
    lFModeId                [19] IMPLICIT LFModeId,
    srtBPPDatabaseSizeId    [20] IMPLICIT SrtBPPDatabaseSizeID,
    srtBPPEntry             [21] IMPLICIT SrtBPPEntry }

IEEE802-1BridgeLme-ParameterValue : : = CHOICE { . . . ,

    rdLimit                 [17] IMPLICIT RdLimit ,
    portInfo                [18] IMPLICIT PortInfo,
    lanId                   [19] IMPLICIT LanId,
    largestMSDUSize         [20] IMPLICIT LargestMSDUSize,
    lFMode                  [21] IMPLICIT LFMode,
    srtSize                 [22] IMPLICIT SrtSize,
    srtBPPInfo              [23] IMPLICIT SrtBPPInfo,
    srtBPPParam             [24] IMPLICIT SrtBPPParam }
- - ********************************************************************************************************
- - * Конец ДОПОЛНЕНИЙ К ПЕРЕЧНЮ идентификаторов параметров
- - ********************************************************************************************************

- - ********************************************************************************************************
- - * ДОПОЛНЕНИЯ ДЛЯ логического объекта диспетчера моста
- - ********************************************************************************************************
```

```
bridgeEntity ATTRIBUTE  ...
: : = 1
   BridgeConfig : : = SEQUENCE {...,
      srtTransferCapacity        [4] IMPLICIT MaxMSDUSize }
     MaxMSDUSize : : = INTEGER  - - Определяет максимальный размер СБДУД в октетах. Это значение
                                - - должно быть установлено в 0 в мостах, которые не обеспечивают ре-
                                - - жим ПМСО.
- -  ********************************************************************************************************
- - * Конец ДОПОЛНЕНИЙ ДЛЯ логического объекта диспетчера моста
- -  ********************************************************************************************************

- -  ********************************************************************************************************
- - * ДОПОЛНЕНИЯ ДЛЯ процесса продвижения
- -  ********************************************************************************************************
- - В дополнение к счетчикам, определенным в 7.3 основной части настоящего стандарта, для моста ПМСО
- - предусмотрены следующие счетчики порта.
forwardProc ATTRIBUTE ...
: : = 2
PortCounters : : = SEQUENCE (
      invalidRi                [8]  IMPLICIT Counter64,
      dupLout                  [9]  IMPLICIT Counter64,
      lanIdMismatch            [10] IMPLICIT Counter64,
      dupLanIdOrTreeError      [11] IMPLICIT Counter64,
      steRdLimitExceeded       [12] IMPLICIT Counter64,
      areRdLmtExceeded         [13] IMPLICIT Counter64,
      srfsFozwarded            [14] IMPLICIT Counter64,
      steFramesForwarded       [15] IMPLICIT Counter64,
      areFramesForwarded       [16] IMPLICIT Counter64 )
- -  ********************************************************************************************************
- - * Конец ДОПОЛНЕНИЙ ДЛЯ процесса продвижения
- -  ********************************************************************************************************

- -  ********************************************************************************************************
- - * ДОПОЛНЕНИЯ ДЛЯ логического объекта протокола моста ПМСО
- -  ********************************************************************************************************
srtBridgeEntity ATTRIBUTE
    IDTYPES       [15] PortRDLimId
                  [16] PortNumber
                  [17] PortLanId
                  [18] PortLargestMSDUSizeId
                  [19] LFModeId
    VALUETYPES    [17] RdLimit
                  [18] PortInfo
                  [19] LanId
                  [20] LargestMSDUSize
                  [21] LFMode
: : = 5
- - Доступ к логическому объекту диспетчера моста может осуществляться следующими способами.
- - Операция «установка», определяющая idCode (5) и ТипИд PortRdLimId, устанавливает конкретное пре-
- - дельное число ДМ.
- - Операция «получение», определяющая idCode (5) и ТипИд PortNumber, выдает значение информации
- - порта.
- - Операция «установка», определяющая idCode (5) и ТипИд PortLanId, устанавливает идентификатор ЛВС
- - для определенного порта.
```

```
- - Операция «установка», определяющая idCode (5) и ТипИд PortLargestMSDUSizeId, устанавливает мак-
- - симальный размер для порта.
- - Операция «установка», определяющая idCode (5) и ТипИд LFModeId, устанавливает режим СДК моста.
- - Операция «получение», определяющая idCode (5) и ТипИд LFModeId, выдает режим СДК моста.
PortRdLimId : : = INTEGER  {
                        SteLimit (0),
                        AreLimit (1)  }
PortNumber : : = INTEGER  - - для идентификации порта принимает значения от 1 и выше
PortLanId    : : = INTEGER  - - для идентификации ИД ЛВС принимает значения от 1 и выше
PortLargestMSDUId : : = INTEGER  - - принимает значения от 1 и выше
PortInfo : : = SEQUENCE {
                    srtPortType          [0] IMPLICIT SrtPortType,
                    lanId                [1] IMPLICIT LanId,
                    largestMSDUSize      [2] IMPLICIT LargestMSDUSize,
                    SteRdLimit           [3] IMPLICIT RdLimit,
                    areRdLimit           [4] IMPLICIT RdLimit  }
- - П р и м е ч а н и е — Самый большой размер СБДУД используется при определении установки битов СДК
- - в кадрах анализаторах.
SrtPortType : : = INTEGER  {
                  TB (0),
                  SRT (1),    }
LanId : : = INTEGER  - - значения от 1 до 4095.
LargestMSDUSize : : = INTEGER  - - это значение указывает число октетов;
                                 - - может превышать 65 535 октетов.
RdLimit : : = INTEGER            - - значения от 0 до 14.
LFModeId : : = NULL
LFMode    : : = INTEGER  {
                      Base (0),
                      Extended (1)  }
- -  ************************************************************************************************
- - * ОПРЕДЕЛЕНИЕ логического объекта база данных ППМ ПМСО
- -  ************************************************************************************************
- - Логический объект базы данных пары портов моста ПМСО
srtBPPDatabase    ATTRIBUTE
     IDTYPES        [20] SrtBPPDatabaseSizeId
                    [21] SrtBPPEntry
     VALUETYPES  [23] SrtSize
                    [24] SrtBPPInfo
                    [25] SrtBPPParam
: : = 6
- - Доступ к записи в базе данных пары портов моста ПМСО может осуществляться следующими способами.
- - Операция «получение», определяющая idCode (6) и ТипИд SrtBPPDatabaseSize, выдает значение размера
- - базы данных ПМСО.
- - Операция «установка», определяющая idCode (6) и ТипИд SrtBPPEntry, устанавливает параметры пары
- - портов моста ПМСО для данной записи.
- - Операция «получение», определяющая idCode (6) и ТипИд SrtBPPEntry, выдает значение информации
- - пары портов моста ПМСО для данной записи.
SrtBPPDatabaseSizeId : : = NULL
SrtBPPEntry : : = SEQUENCE {
                    bppLowPortNum      [0] IMPLICIT PortNumber,
                    bppHighPortNum     [1] IMPLICIT PortNumber  }
```

```
SrtSize : : = INTEGER - - указывает количество записей в базе данных ППМ

SrtBPPInfo  : : =  SEQUENCE {
                   bppLowPortNum      [0] IMPLICIT PortNumber,
                   bppHighPortNum     [1] IMPLICIT PortNumber,
                   srtBPPParam        [2] IMPLICIT SrtBPPParam  }

SrtBPPParam : : = SEQUENCE  {
                   bppBridgeNum       [0] IMPLICIT BPPBridgeNum,
                   bppState           [1] IMPLICIT BPPState  }

DPPBridgeNum : : = INTEGER  - - значения от 0  до 15; указывает номер моста, назначенный  для пары
                              - - портов.

BPPState : : = INTEGER  {
                   disabled (0),
                   enabled (1)  }

END
```

ПРИЛОЖЕНИЕ D*
(обязательное)

Форма ЗСРП для операций моста при прозрачной маршрутизации со стороны отправителя

D.1 **Введение**

Поставщик реализации, претендующей на соответствие приложению С настоящего стандарта, должен заполнить приводимую ниже форму заявки о соответствии реализации протоколу (ЗСРП) и представить информацию, необходимую для идентификации как поставщика, так и реализации.

D.2 **Сокращения и специальные символы. Символы факультативного статуса**

О — обязательно

Ф — факультативно

Фп — факультативно, но, по крайней мере, одна из групп факультативных возможностей отмеченных одним и тем же номером *n*, обязательна

З — запрещено

позиция — указываемый статус или ответ применим только в том случае, если ЗСРП констатирует, что элемент, идентифицированный данной позицией, обеспечивается

D.3 **Инструкции по заполнению копии формы ЗСРП**

Форма ЗСРП представляет собой вопросник фиксированного формата. Она позволяет поставщику обеспечить дополнительную информацию двух видов. При наличии такой информации она должна быть представлена в виде позиций, помеченных О.i или Д.i, с целью взаимных ссылок, где i — любой недвусмысленный идентификатор позиции (например обычный номер); никаких других ограничений на ее формат и представление не налагается.

Заполненная форма ЗСРП — это заявка о соответствии реализации протоколу для рассматриваемой реализации.

Ответы на вопросы приводятся в крайней правой колонке либо простой пометкой ответа из ограниченного выбора ответов (обычно «Да» или «Нет»), либо путем записи значения, набора или диапазона значений.

Позиции особой информации обусловлены определенными ответами в вопроснике; они указываются символом взаимной ссылки «О.____» и подлежат заполнению. Такие позиции имеют место, например, в тех случаях, когда ответ указывает, что возможность, определенная как обязательная, не реализована; наличие позиции особой информации должно иметь соответствующее обоснование.

---

* Для форм ЗСРП установлено свободное авторское право. Пользователи настоящего стандарта могут свободно воспроизводить форму ЗСРП из данного приложения с тем, чтобы ее можно было использовать в необходимых целях, и могут в дальнейшем опубликовать заполненную ЗСРП.

Форма ЗСРП позволяет поставщику дать дополнительную информацию, облегчающую интерпретацию ЗСРП. Не предполагается наличие большого объема такой информации и ЗСРП может считаться заполненной и без нее. Примерами могут служить описания способов, с помощью которых реализация (отдельная) может быть приспособлена для работы в различных функциональных средах и конфигурациях.

Ссылки на позиции дополнительной информации могут быть даны после любого ответа в вопроснике и могут быть включены в позиции особой информации. Некоторые ответы в вопроснике также требуют дополнительной информации в тех случаях, когда необходимо пояснить, каким образом реализуется возможность, классифицируемая как факультативная.

Примечание — В случае, когда реализацию можно построить несколькими способами, в отдельной ЗСРП можно описать все соответствующие конфигурации. Однако поставщик может предпочесть разработку нескольких ЗСРП, каждая из которых охватывает некоторое подмножество возможных конфигураций реализации, если это облегчает и упрощает представление информации.

D.4 **Идентификация требований**

Информация в каждой из колонок «Функциональная возможность» в форме ЗСРП служит только для наименования требований к соответствию и облегчения их идентификации в настоящем стандарте. Эта информация не заменяет и не дополняет положений, приведенных в основной части настоящего стандарта.

D.5 **Соответствие стандартам по УДС и УЛЗ**

Позиция	Факультативная возможность	Статус	Ссылки	Обеспечение
1a	Соответствует ли реализованная технология управления доступом к среде необходимым стандартам по УДС?	Кольцо с маркерным методом доступа О ВОРИПД О	ГОСТ Р ИСО/МЭК 8802—5 ИСО 9314—2	Да Нет: О.__ Да Нет: О.__
1b	Соответствует ли реализованное управление логическим звеном стандарту по УЛЗ?	О	ГОСТ 28907	Да Нет: О.__
1c	Соответствует ли реализация стандарту, определяющему мосты на подуровне УЛЗ?	О	С.1.4	Да Нет: О.__

D.6 **Ретрансляция и фильтрация кадров**

Позиция	Факультативная возможность	Статус	Ссылки	Обеспечение
2a	Сохраняется ли порядок ретранслируемых кадров заданного приоритета пользователя?	О	С.2.3.1	Да Нет: О.__
2b	Кадры АПД, АВМ и КМСО предоставляются логическому объекту УДС для передачи только один раз?	О	С.2.3.2	Да Нет: О.__
2c	Устанавливает ли мост поле самого длинного кадра в соответствии с С.3.3.2?	О	С.2.3.2	Да Нет: О.__
2d	Обрабатываются ли кадры, полученные без информации маршрутизации (НМСО), в соответствии с настоящим стандартом?	О	С.3.5	Да Нет: О.__
2e	Обрабатываются ли полученные кадры, маршрутизируемые по конкретному маршруту (КМСО), в соответствии с С.3.7.1?	О	С.3.7.1	Да Нет: О.__

Окончание раздела D.6

Позиция	Факультативная возможность	Статус	Ссылки	Обеспечение
2f	Обрабатываются ли полученные кадры «анализатор всех маршрутов» (АВМ) в соответствии с С.3.7.2?	О	С.3.7.2	Да Нет: О.__
2g	Реализуется ли факультативная возможность 1 пункта С.3.7.2, позволяющая отфильтровывать кадры, прошедшие ранее через мост?	Ф	С.3.7.2	Да Нет
2h	Реализуется ли факультативная возможность 2 пункта С.3.7.2, позволяющая отфильтровывать кадры, прошедшие ранее через мост?	Ф	С.3.7.2	Да Нет
2i	Обрабатываются ли полученные кадры «анализатор покрывающего дерева» в соответствии с С.3.7.3?	О	С.3.7.3	Да Нет: О.__
2j	Продвигаются ли кадры анализаторы (АПД, АВМ) длиной 4 в соответствии с С.3.7.2 и С.3.7.3?	Ф	С.3.7.2 С.3.7.3	Да Нет

D.7 **Номера мостов и ИД ЛВС**

Позиция	Факультативная возможность	Статус	Ссылки	Обеспечение
3a	Может ли быть сформирован в мосту ИД подключенной ЛВС?	О	С.3.8.1	Да Нет: О.__
3b	Может ли быть присвоен мосту номер?	О	С.3.8.2	Нет Да: О.__
3c	Может ли диспетчер моста использовать проверку продублированного номера моста для обнаружения параллельных мостов, которым присвоен один и тот же номер?	Ф	С.3.7.4	Нет Да

D.8 **Диспетчер моста**

Этот вопрос подлежит дальнейшему изучению.

Позиция	Факультативная возможность	Статус	Ссылки	Обеспечение
4a	Операции диспетчера моста	Ф	С.4	Да Нет
4b	Чтение режима СДК	О	С.4.3.1.3	Нет Да: О.__
4c	Установка режима СДК	О	С.4.3.1.3	Нет Да: О.__
4d	Чтение параметров порта	О	С.4.3.2.1	Нет Да: О.__
4e	Установка предельного числа ДМ АВМ	О	С.4.3.2.4	Нет Да: О.__

Окончание раздела D.8

Позиция	Факультативная возможность	Статус	Ссылки	Обеспечение
4f	Определение максимального значения предельного числа ДМ АВМ для данной реализации	О	С.4.3.2.4	Д:___
4g	Установка предельного числа ДМ АПД	О	С.4.3.2.4	Нет Да: О.__
4h	Определение максимального значения предельного числа ДМ АПД для данной реализации	О	С.4.3.2.4	Д:___
4i	Установка ИД ЛВС	О	С.4.3.2.2	Нет Да: О.__
4j	Установка самого большого размера СБДУД	О	С.4.3.2.3	Нет Да: О.__
4k	Чтение размера базы данных пары портов моста ПМСО	О	С.4.4.1.1	Нет Да: О.__
4l	Чтение логического объекта базы данных пары портов моста ПМСО	О	С.4.4.1.2	Нет Да: О.__
4m	Установка логического объекта базы данных пары портов моста ПМСО	О	С.4.4.1.3	Нет Да: О.__

ПРИЛОЖЕНИЕ Е
(справочное)

Стандарты IEEE и ANSI

Настоящий стандарт содержит ссылки, которые имеют справочный характер, на следующие стандарты:

ANSI X3.159—89 Американские национальные стандарты по информационным системам. Языки программирования. Язык Си

IEEE Std 802—90 Стандарты IEEE по локальным и региональным вычислительным сетям. Общее описание и архитектура (ANSI)

IEEE Std 802.1B—92 Стандарты IEEE по локальным и региональным вычислительным сетям. Административное управление ЛВС/РВС (ANSI)

УДК 681.324:006.354 ОКС 35.110 П85 ОКСТУ 4002

Ключевые слова: обработка данных, обмен информацией, передача данных, взаимосвязь сетей, взаимосвязь открытых систем, локальные вычислительные сети, процедуры управления, спецификация

Редактор *В. П. Огурцов*
Технический редактор *Н. С. Гришанова*
Корректор *Е. Ю. Митрофанова*
Компьютерная верстка *В. Н. Романовой*

Изд. лиц. № 021007 от 10.08.95. Сдано в набор 20.10.99. Подписано в печать 09.12.99. Усл. печ. л. 16,28. Уч.-изд. л. 16,10.
Тираж 245 экз. С 4059. Зак. 2526

ИПК Издательство стандартов,107076, Москва, Колодезный пер., 14.
Набрано в Калужской типографии стандартов на ПЭВМ.
Калужская типография стандартов,248021, Калуга, ул. Московская, 256.
ПЛР № 040138

gost.e-ecolog.ru