



МЕЖДУНАРОДНЫЙ СОЮЗ ЭЛЕКТРОСВЯЗИ

МСЭ-Т

СЕКТОР СТАНДАРТИЗАЦИИ
ЭЛЕКТРОСВЯЗИ МСЭ

H.245

(07/2003)

СЕРИЯ H: АУДИОВИЗУАЛЬНЫЕ И
МУЛЬТИМЕДИЙНЫЕ СИСТЕМЫ

Инфраструктура аудиовизуальных служб –
Процедуры связи

**Управляющий протокол для
мультимедийной связи**

Рекомендация МСЭ-Т H.245

РЕКОМЕНДАЦИИ МСЭ-Т СЕРИИ Н
АУДИОВИЗУАЛЬНЫЕ И МУЛЬТИМЕДИЙНЫЕ СИСТЕМЫ

ХАРАКТЕРИСТИКИ ВИДЕОТЕЛЕФОННЫХ СИСТЕМ	Н.100–Н.199
ИНФРАСТРУКТУРА АУДИОВИЗУАЛЬНЫХ СЛУЖБ	
Общие положения	Н.200–Н.219
Мультиплексирование и синхронизация при передаче	Н.220–Н.229
Системные аспекты	Н.230–Н.239
Процедуры связи	Н.240–Н.259
Кодирование движущихся видеоизображений	Н.260–Н.279
Сопутствующие системные аспекты	Н.280–Н.299
Системы и оконечное оборудование для аудиовизуальных служб	Н.300–Н.349
Архитектура служб каталогов для аудиовизуальных и мультимедийных служб	Н.350–Н.359
Качество архитектуры обслуживания для аудиовизуальных и мультимедийных служб	Н.360–Н.369
Дополнительные услуги для мультимедийных служб	Н.450–Н.499
ПРОЦЕДУРЫ МОБИЛЬНОСТИ И СОВМЕСТНОЙ РАБОТЫ	
Обзор мобильности и совместной работы, определений, протоколов и процедур	Н.500–Н.509
Мобильность для мультимедийных систем и служб серии Н	Н.510–Н.519
Приложения и службы мобильной мультимедийной совместной работы	Н.520–Н.529
Безопасность для мобильных мультимедийных систем и служб	Н.530–Н.539
Безопасность для приложений и служб мобильной мультимедийной совместной работы	Н.540–Н.549
Процедуры мобильного взаимодействия	Н.550–Н.559
Процедуры взаимодействия мобильной мультимедийной совместной работы	Н.560–Н.569
ШИРОКОПОЛОСНЫЕ И МУЛЬТИМЕДИЙНЫЕ TRIPLE-PLAY УСЛУГИ	
Предоставление широкополосных мультимедийных услуг по VDSL	Н.610–Н.619

Для получения более подробной информации просьба обращаться к перечню Рекомендаций МСЭ-Т.

Рекомендация МСЭ-Т Н.245

Управляющий протокол для мультимедийной связи

Резюме

В настоящей Рекомендации определены синтаксис и семантика информационных сообщений терминалов, а также процедур использования этих сообщений для согласования внутри полосы в начале и во время связи. Эти сообщения охватывают возможности приема и передачи, информацию о предпочитаемом приемной стороной режиме, о сигнализации по логическому каналу, а также об управлении и индикации. Определены процедуры сигнализации с подтверждением приема, обеспечивающие надежную передачу аудиовизуальных сигналов и данных.

Продукты с заявленной совместимостью с версией 10 Рекомендации Н.245 должны удовлетворять всем обязательным требованиям настоящей Рекомендации. Продукты версии 10 можно идентифицировать по сообщениям набора возможностей терминала Н.245 TerminalCapabilitySet, содержащими значение идентификатора протокола protocolIdentifier {itu-t (0) recommendation (0) h (8) 245 version (0) 10}.

Источник

Рекомендация МСЭ-Т Н.245 утверждена 14 июля 2003 года 16-й Исследовательской комиссией МСЭ-Т (2001–2004 гг.) в соответствии с процедурой, изложенной в Рекомендации МСЭ-Т А.8.

ПРЕДИСЛОВИЕ

Международный союз электросвязи (МСЭ) является специализированным учреждением Организации Объединенных Наций в области электросвязи. Сектор стандартизации электросвязи МСЭ (МСЭ-Т) – постоянный орган МСЭ. МСЭ-Т отвечает за изучение технических, эксплуатационных и тарифных вопросов и за выпуск Рекомендаций по ним с целью стандартизации электросвязи на всемирной основе.

На Всемирной ассамблее по стандартизации электросвязи (ВАСЭ), которая проводится каждые четыре года, определяются темы для изучения Исследовательскими комиссиями МСЭ-Т, которые, в свою очередь, вырабатывают Рекомендации по этим темам.

Утверждение Рекомендаций МСЭ-Т осуществляется в соответствии с процедурой, изложенной в Резолюции 1 ВАСЭ.

В некоторых областях информационных технологий, которые входят в компетенцию МСЭ-Т, необходимые стандарты разрабатываются на основе сотрудничества с ИСО и МЭК.

ПРИМЕЧАНИЕ

В настоящей Рекомендации термин "администрация" используется для краткости и обозначает как администрацию электросвязи, так и признанную эксплуатационную организацию.

Соблюдение положений данной Рекомендации носит добровольный характер. Однако в Рекомендации могут содержаться определенные обязательные положения (например, для обеспечения возможности взаимодействия или применимости), и соблюдение положений данной Рекомендации достигается в случае выполнения всех этих обязательных положений. Для выражения необходимости выполнения требований используется синтаксис долженствования и соответствующие слова (такие, как "должен" и т.п.), а также их отрицательные эквиваленты. Использование этих слов не предполагает, что соблюдение положений данной Рекомендации является обязательным для какой-либо из сторон.

ПРАВА ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

МСЭ обращает внимание на вероятность того, что практическое применение или реализация этой Рекомендации может включать использование заявленного права интеллектуальной собственности. МСЭ не занимает какую бы то ни было позицию относительно подтверждения, обоснованности или применимости заявленных прав интеллектуальной собственности, независимо от того, отстаиваются ли они членами МСЭ или другими сторонами вне процесса подготовки Рекомендации.

На момент утверждения настоящей Рекомендации МСЭ получил извещение об интеллектуальной собственности, защищенной патентами, которые могут потребоваться для выполнения этой Рекомендации. Однако те, кто будет применять Рекомендацию, должны иметь в виду, что это может не отражать самую последнюю информацию, и поэтому им настоятельно рекомендуется обращаться к патентной базе данных БСЭ.

© ITU 2005

Все права сохранены. Никакая часть данной публикации не может быть воспроизведена с помощью каких-либо средств без письменного разрешения МСЭ.

СОДЕРЖАНИЕ

	Стр.
1 Сфера применения	1
2 Ссылки	1
3 Определения	5
4 Сокращения	6
5 Введение.....	7
5.1 Определение отношения "ведущий-ведомый".....	8
5.2 Обмен возможностями	8
5.3 Процедуры сигнализации логического канала	9
5.4 Запрос принимающего терминала о закрытии логического канала	9
5.5 Изменение данных таблицы мультиплексирования Н.223.....	9
5.6 Запрос режима передачи аудиовизуальной информации и данных	9
5.7 Определение задержки из-за подтверждения приема	10
5.8 Циклы обслуживания	10
5.9 Команды и индикация	10
Приложение А – Сообщения: Синтаксис.....	10
Приложение В – Сообщения: семантические определения	75
В.1 Сообщения, определяющие отношение "ведущий-ведомый"	76
В.2 Сообщения о возможностях терминалов.....	76
В.3 Сообщения для передачи сигнализации логического канала.....	106
В.4 Сигнальные сообщения таблицы мультиплексирования.....	115
В.5 Сигнальные сообщения повторной передачи таблицы мультиплексирования.....	116
В.6 Сообщения с запросом режима	117
В.7 Сообщения о задержке из-за подтверждения приема	121
В.8 Сообщения цикла обслуживания	121
В.9 Сообщения о режиме связи.....	122
В.10 Сообщения с запросом конференции и ответом на нее	123
В.11 Многоканальные сообщения	126
В.12 Сообщения об изменении скорости передачи по логическому каналу.....	128
В.13 Команды	129
В.14 Индикация	135
В.15 Универсальные сообщения.....	142
Приложение С – Процедуры	143
С.1 Введение	143
С.2 Процедуры определения отношения "ведущий-ведомый"	145
С.3 Процедуры обмена возможностями.....	156
С.4 Процедуры сигнализации однонаправленного логического канала.....	164
С.5 Процедуры сигнализации двунаправленного логического канала	177
С.6 Процедуры запроса о закрытии логического канала.....	192

	Стр.
C.7	Процедуры таблицы мультиплексирования H.223 198
C.8	Процедуры запроса элемента таблицы мультиплексирования 209
C.9	Процедуры запроса режима 216
C.10	Процедуры определения задержки из-за подтверждения приема 226
C.11	Процедуры цикла обслуживания 230
Приложение D – Присвоение идентификаторов объектов 239	
Приложение E – Определение возможностей ИСО/МЭК 14496-2 242	
Приложение F – Определения, относящиеся к возможности "Управление скоростью передачи данных логического канала" 244	
Приложение G – Определения, относящиеся к возможности ИСО/МЭК 14496-1 245	
G.1	Идентификатор возможности 246
G.2	Параметры возможностей, используемые для согласования возможностей и сигнализации логического канала 246
G.3	Параметры возможностей, используемые только при сигнализации логического канала 248
Приложение H – Определения возможностей ИСО/МЭК 14496-3 248	
Приложение I – Определения, относящиеся к возможности "Адаптивная мультискоростная передача данных в стандарте GSM" 252	
I.1	Определение сигнализации режима и заполнения битами для обеспечения выравнивания по октетам 254
Приложение J – Определения, относящиеся к речевому кодексу ACELP в стандарте TDMA ... 263	
Приложение K – Определения, относящиеся к речевому кодексу US1 в стандарте TDMA 264	
Приложение L – Определения, относящиеся к речевому кодексу EVRC в стандарте CDMA ... 265	
Приложение M – Определения ИСО/МЭК 13818-7 и МСЭ-Р BS.1196 266	
Добавление I – Обзор синтаксиса ASN.1 267	
I.1	Введение в язык ASN.1 267
I.2	Основные типы данных ASN.1 268
I.3	Агрегированные типы данных 269
I.4	Тип данных "идентификатор объекта" 270
Добавление II – Примеры процедур H.245 271	
II.1	Введение 271
II.2	Объект сигнализации определения отношения "ведущий-ведомый" 272
II.3	Объект сигнализации обмен возможностями 276
II.4	Объект сигнализации логический канал 277
II.5	Объект сигнализации закрытие логического канала 279
II.6	Объект сигнализации элемент таблицы мультиплексирования 281
II.7	Объект сигнализации запрос режима 282
II.8	Объект сигнализации задержка из-за двухстороннего приема 284
II.9	Объект сигнализации двунаправленный логический канал 285
Добавление III – Перечень таймеров и счетчиков процедур 288	
III.1	Таймеры 288
III.2	Счетчики 289

	Стр.
Добавление IV – Процедура расширения Рекомендации Н.245	289
Добавление V – Процедура replacementFor	291
Добавление VI – Примеры установок структуры возможностей Н.263.....	292
VI.1 Примеры установки параметров усовершенствованного уровня Н.245	292
VI.2 Примеры установки параметров обратного видеоканала Н.245	294
Добавление VII – Процедура и шаблон для определения новых возможностей в качестве универсальных возможностей Н.245	297
VII.1 Процедура.....	298
VII.2 Шаблон	298
VII.3 Пример шаблона – Н.261	300
Добавление VIII – Перечень универсальных возможностей и универсальных сообщений, определенных в других Рекомендациях и стандартах помимо настоящей Рекомендации	301
Добавление IX – Использование языка АСН.1 в настоящей рекомендации	303
IX.1 Сопровождение данных тегами.....	303
IX.2 Типы.....	303
IX.3 Ограничения и диапазоны.....	303
IX.4 Расширяемость.....	303

Рекомендация МСЭ-Т Н.245

Управляющий протокол для мультимедийной связи

1 Сфера применения

В настоящей Рекомендации определены синтаксис и семантика информационных сообщений терминалов, а также процедур использования этих сообщений для согласования внутри полосы в начале и во время связи. Эти сообщения охватывают возможности приема и передачи, информацию о предпочитаемом приемной стороной режиме, о сигнализации по логическому каналу, а также об управлении и индикации. Определены процедуры сигнализации с подтверждением приема, обеспечивающие надежную передачу аудиовизуальных сигналов и данных.

Настоящая Рекомендация охватывает широкий диапазон возможных приложений, в том числе хранение и поиск, службы передачи сообщений и распределения, а также диалоговые службы. Она применима к мультимедийным системам, которые используют мультиплексирование, определенное в Рекомендациях МСЭ-Т Н.222.0, Н.223 и Н.225.0 (но не ограничивается ими). В этих различных системах используется одни и те же синтаксис и семантика, следовательно, они побитово совместимы. Некоторые из этих процедур применимы ко всем системам, тогда как другие более специфичны и применимы только к конкретным системам.

Различные системы, в которых используется настоящая Рекомендация, могут быть основаны на различных транспортных протоколах. Однако Рекомендация предназначена для использования на надежном транспортном уровне, то есть на таком транспортном уровне, который обеспечивает гарантированную доставку правильных данных.

ПРИМЕЧАНИЕ. – Не следует путать эти Рекомендации с Рекомендациями на систему управления Т.120, сеанс которой устанавливается внутри потока данных, и функциональные возможности которого отличаются от возможностей, описанных в настоящей Рекомендации, поскольку поток по Рекомендации Н.245 и поток данных по Рекомендации Т.120 дополняют друг друга.

2 Ссылки

Указанные ниже Рекомендации МСЭ-Т и другие источники содержат положения, которые путем ссылки на них в данном тексте составляют положения настоящей Рекомендации. На момент публикации указанные издания были действующими. Все Рекомендации и другие источники могут подвергаться пересмотру; поэтому всем пользователям данной Рекомендации предлагается изучить возможность применения последнего издания Рекомендаций и других источников, перечисленных ниже. Список действующих в настоящее время Рекомендаций МСЭ-Т регулярно публикуется. Ссылка на документ в данной Рекомендации не придает ему как отдельному документу статус Рекомендации.

- [1] ITU-T Recommendation E.164 (1997), *The international public telecommunication numbering plan*.
- [2] ITU-T Recommendation G.711 (1988), *Pulse code modulation (PCM) of voice frequencies*.
- [3] ITU-T Recommendation G.722 (1988), *7 kHz audio-coding within 64 kbit/s*.
- [4] ITU-T Recommendation G.723.1 (1996), *Dual rate speech coder for multimedia communication transmitting at 5.3 and 6.3 kbit/s*.
- [5] ITU-T Recommendation G.728 (1992), *Coding of speech at 16 kbit/s using low-delay code excited linear prediction*.
- [6] ITU-T Recommendation G.729 (1996), *Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)*.
- [7] ITU-T Recommendation H.221 (1999), *Frame structure for a 64 to 1920 kbit/s channel in audiovisual teleservices*.
- [8] ITU-T Recommendation H.222.0 (2000) | ISO/IEC 13818-1:2000, *Information technology – Generic coding of moving pictures and associated audio information: Systems*.

- [9] ITU-T Recommendation H.222.1 (1996), *Multimedia multiplex and synchronization for audiovisual communication in ATM environments.*
- [10] ITU-T Recommendation H.223 (1996), *Multiplexing protocol for low bit rate multimedia communication.*
- [11] ITU-T Recommendation H.224 (2000), *A real time control protocol for simplex applications using the H.221 LSD/HSD/MLP channels.*
- [12] ITU-T Recommendation H.225.0 (2003), *Call signalling protocols and media stream packetization for packet-based multimedia communications systems.*
- [13] ITU-T Recommendation H.230 (1999), *Frame-synchronous control and indication signals for audiovisual systems.*
- [14] ITU-T Recommendation H.233 (1995), *Confidentiality system for audiovisual services.*
- [15] ITU-T Recommendation H.234 (1994), *Encryption and key management and authentication system for audiovisual services.*
- [16] ITU-T Recommendation H.235 (2000), *Security and encryption for H-Series (H.323 and other H.245-based) multimedia terminals.*
- [17] ITU-T Recommendation H.243 (2000), *Procedures for establishing communication between three or more audiovisual terminals using digital channels up to 1920 kbits.*
- [18] ITU-T Recommendation H.261 (1993), *Video codec for audiovisual services at $p \times 64$ kbit/s.*
- [19] ITU-T Recommendation H.262 (2000) | ISO/IEC 13818-2:2000, *Information technology – Generic coding of moving pictures and associated audio: Video.*
- [20] ITU-T Recommendation H.263 (1998), *Video coding for low bit rate communication.*
- [21] ITU-T Recommendation H.310 (1998), *Broadband audiovisual communication systems and terminals.*
- [22] ITU-T Recommendation H.320 (1999), *Narrow-band visual telephone systems and terminal equipment.*
- [23] ITU-T Recommendation H.323 (2003), *Packet-based multimedia communications systems.*
- [24] ITU-T Recommendation H.324 (2002), *Terminal for low bit-rate multimedia communication.*
- [25] ITU-T Recommendation I.363.x series, *B-ISDN ATM adaptation layer (AAL) specification.*
- [26] ITU-T Recommendation Q.2931 (1995), *Digital subscriber signalling system No. 2 – User-Network Interface (UNI) layer 3 specification for basic call/connection control.*
- [27] ITU-T Recommendation T.30 (2003), *Procedures for document facsimile transmission in the general switched telephone network.*
- [28] ITU-T Recommendation T.35 (2000), *Procedure for the allocation of ITU-T defined codes for non-standard facilities.*
- [29] ITU-T Recommendation T.38 (2003), *Procedures for real-time Group 3 facsimile communication over IP networks.*
- [30] ITU-T Recommendation T.51 (1992), *Latin based coded character sets for telematic services.*
- [31] ITU-T Recommendation T.84 (1996) | ISO/IEC 10918-3:1997, *Information technology – Digital compression and coding of continuous-tone still images: Extensions.*

- [32] ITU-T Recommendation T.120 (1996), *Data protocols for multimedia conferencing*.
- [33] ITU-T Recommendation T.123 (1999), *Network-specific data protocol stacks for multimedia conferencing*.
- [34] ITU-T Recommendation T.140 (1998), *Protocol for multimedia application text conversation*.
- [35] ITU-T Recommendation T.434 (1999), *Binary file transfer format for the telematic services*.
- [36] ITU-T Recommendation V.14 (1993), *Transmission of start-stop characters over synchronous bearer channels*.
- [37] ITU-T Recommendation V.34 (1998), *A modem operating at data signalling rates of up to 33 600 bit/s for use on the general switched telephone network and on leased point-to-point 2-wire telephone-type circuits*.
- [38] ITU-T Recommendation V.42 (2002), *Error-correcting procedures for DCEs using asynchronous-to-synchronous conversion*.
- [39] ITU-T Recommendation V.140 (1998), *Procedures for establishing communication between two multiprotocol audiovisual terminals using digital channels at a multiple of 64 or 56 kbit/s*.
- [40] ITU-T Recommendation X.680 (2002) | ISO/IEC 8824-1:2002, *Information technology – Abstract Syntax Notation One (ASN.1) – Specification of basic notation*.
- [41] ITU-T Recommendation X.681 (2002) | ISO/IEC 8824-2:2002, *Information technology – Abstract Syntax Notation One (ASN.1) – Information object specification*.
- [42] ITU-T Recommendation X.691 (2002) | ISO/IEC 8825-2:2002, *Information technology – ASN.1 encoding rules – Specification of Packed Encoding Rules (PER)*.
- [43] ISO/IEC 13239:2002, *Information technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures*.
- [44] ISO/IEC 11172-2:1993, *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s – Part 2: Video*.
- [45] ISO/IEC 11172-3:1993, *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s – Part 3: Audio*.
- [46] ISO/IEC 13818-3:1998, *Information technology – Generic coding of moving pictures and associated audio information – Part 3: Audio*.
- [47] ISO/IEC 13818-6:1998, *Information technology – Generic coding of moving pictures and associated audio information – Part 6: Extensions for DSM-CC*.
- [48] ISO/IEC 14496-1:1999, *Information technology – Coding of audio-visual objects – Part 1: Systems*.
- [49] ISO/IEC 14496-2:1999, *Information technology – Coding of audio-visual objects – Part 2: Visual*.
- [50] ISO/IEC 14496-3:1999, *Information technology – Coding of audio-visual objects – Part 3: Audio*.
- [51] ISO/IEC 14496-3/Amd.1:2000, *Information technology – Coding of audio-visual objects – Part 3: Audio – Amendment 1: Audio extensions*.
- [52] ISO/IEC TR 9577:1999, *Information technology – Protocol identification in the network layer*.

- [53] ETSI ETS 300 961 (GSM 06.10), *Full rate speech transcoding.*
- [54] ETSI ETS 300 969 (GSM 06.20), *Half rate speech transcoding.*
- [55] ETSI ETS 300 726 (GSM 06.60), *Enhanced Full Rate (EFR) speech transcoding.*
- [56] ETSI ETS 300 963 (GSM 06.12), *Comfort noise aspect for full rate speech traffic channels.*
- [57] ETSI ETS 300 971 (GSM 06.22), *Comfort noise aspects for half rate speech traffic channels.*
- [58] ETSI ETS 300 728 (GSM 06.62), *Comfort noise aspects for Enhanced Full Rate (EFR) speech traffic channels.*
- [59] ETSI ETS 300 964 (GSM 06.31), *Discontinuous Transmission (DTX) for full rate speech traffic channels.*
- [60] ETSI ETS 300 972 (GSM 06.41), *Discontinuous transmission (DTX) for half rate speech traffic channels.*
- [61] ETSI ETS 300 729 (GSM 06.81), *Discontinuous Transmission (DTX) for Enhanced Full Rate (EFR) speech traffic channels.*
- [62] ETSI ETS 300 962 (GSM 06.11), *Substitution and muting of lost frames for full rate speech traffic channels.*
- [63] ETSI ETS 300 970 (GSM 06.21), *Substitution and muting of lost frames for half rate speech traffic channels.*
- [64] ETSI ETS 300 727 (GSM 06.61), *Substitution and muting of lost frames for Enhanced Full Rate (EFR) speech traffic channels.*
- [65] ETSI ETS 300 965 (GSM 06.32), *Voice Activity Detector (VAD) for full rate speech traffic channels.*
- [66] ETSI ETS 300 973 (GSM 06.42), *Voice Activity Detector (VAD) for half rate speech traffic channels.*
- [67] ETSI ETS 300 730 (GSM 06.82), *Voice activity detection for enhanced full rate speech traffic channels.*
- [68] ETSI ETS 300 724 (GSM 06.53), *ANSI-C code for the GSM Enhanced Full Rate Speech (EFR) speech codec.*
- [69] ETSI EN 301 712 GSM 06.73, *ANSI-C code for the AMR speech codec.*
- [70] ETSI EN 301 704 GSM 06.90, *Adaptive Multi-Rate (AMR) speech transcoding.*
- [71] ETSI EN 301 705 GSM 06.91, *Substitution and muting of lost frames for Adaptive Multi Rate (AMR) speech traffic channels.*
- [72] ETSI EN 301 706 GSM 06.92, *Comfort noise aspects for Adaptive Multi-Rate (AMR) speech traffic channels.*
- [73] ETSI EN 301 708 GSM 06.94, *Voice Activity Detection (VAD) for Adaptive Multi-Rate (AMR) speech traffic channels.*
- [74] RCR STD-27H, *Personal Digital Cellular Telecommunication System RCR Standard.*
- [75] TIA/EIA – 136-Rev.A, Part 410, *TDMA Cellular/PCS – Radio Interface, Enhanced Full Rate Voice Codec (ACELP). Formerly IS-641. TIA published standard, 1998.*
- [76] TIA/EIA/IS 641-A (1998), *TDMA Cellular/PCS – Radio Interface, US1 Full Rate Voice Codec.*

- [77] ITU-T Recommendation H.239 (2003), *Role management and additional media channels for H.300-series terminals*.
- [78] ITU-T Recommendation H.241 (2003), *Extended video procedures and control signals for H.300-series terminals*.

3 Определения

В настоящей Рекомендации даны определения следующих терминов:

- 3.1 двунаправленный логический канал:** Двунаправленный логический канал состоит из пары связанных друг с другом трактов передачи между двумя терминалами, по одному в каждом направлении передачи.
- 3.2 возможность:** Терминал обладает некоторой конкретной возможностью, если он может кодировать и передавать или принимать и декодировать этот конкретный сигнал.
- 3.3 канал:** Канал – это однонаправленная линия связи между двумя конечными точками.
- 3.4 команда:** Команда – это сообщение, которое требует выполнить действие, но не обязательно требует ответа в явном виде.
- 3.5 элементарный поток:** Элементарный поток – это общий термин для обозначения кодированного видеосигнала, кодированного аудиосигнала или любого другого кодированного потока битов.
- 3.6 элемент:** Слово "элемент" используется для обозначения элементов наборов или таблиц, таких как набор возможностей и таблицы мультиплексирования.
- 3.7 прямая [передача]:** Под прямой передачей понимается передача от терминала, запрашивающего другой терминал о предоставлении двунаправленного логического канала.
- 3.8 внутрисполосный:** Внутрисполосными называют сообщения, которые передаются внутри того канала или логического канала, к которому они относятся.
- 3.9 принимающий:** Принимающий сигнал сигнализации объект не может инициировать процедуру, но он отвечает на сообщения удаленного объекта сигнализации и на примитивы собственного пользователя.
- 3.10 индикация:** Индикация – это сообщение, которое содержит информацию, но не требует действия или ответа.
- 3.11 логический канал:** Логический канал – это однонаправленный тракт или двунаправленный тракт передачи информации.
- 3.12 номер логического канала:** Номер логического канала – это номер, который идентифицирует один определенный логический канал.
- 3.13 сигнализация по логическому каналу:** Сигнализация по логическому каналу – это набор процедур, которые используются для открывания и закрытия логических каналов.
- 3.14 ведущий терминал:** Ведущий терминал – это терминал, который определяется как ведущий терминал согласно процедуре определения отношения "ведущий-ведомый", указанной в настоящей Рекомендации, или же согласно некоторой другой процедуре.
- 3.15 тип среды:** Тип среды – это отдельная форма, в которой информация представляется пользователю, или же данные, представляющие эту информацию. Примерами типов среды являются видеосигналы, аудиосигналы и текст.
- 3.16 режим:** Режим – это набор элементарных потоков, которые терминал передает, намеревается передать или хотел бы принимать.
- 3.17 мультимедийная связь:** Под мультимедийной связью понимают одновременную передачу и/или прием сигналов двух или более типов среды.
- 3.18 нестандартный:** Не соответствующий национальному или международному стандарту, указанному в настоящей Рекомендации.
- 3.19 передающий:** Передающий сигналы сигнализации объект является тем объектом, который инициализирует процедуру.

3.20 многоточечный: Многоточечным называется одновременное соединение трех или более терминалов для связи между несколькими объектами с помощью узлов управления многосторонней связью (MCU) (мостов), которые осуществляют централизованное направление потоков информации.

3.21 запрос: Запрос – это сообщение, которое вызывает действие другого терминала и требует от него немедленный ответ.

3.22 ответ: Ответ – это сообщение, являющееся ответом на запрос.

3.23 обратная [передача]: Под "обратной" понимается передача, идущая от терминала, принимающего запрос на предоставление двунаправленного логического канала, к терминалу, совершающему этот запрос.

3.24 сеанс: Сеанс – это период, в течение которого осуществляется связь между двумя терминалами; эта связь может быть диалоговой или недиалоговой (например, поиск в базе данных).

3.25 ведомый терминал: Ведомый терминал – это терминал, который определяется как ведомый терминал согласно процедуре определения отношения "ведущий-ведомый", определенной в настоящей Рекомендации или же согласно некоторой другой процедуре.

3.26 поддержка: Возможность работать в заданном режиме; однако требование обеспечения поддержки некоторого режима не означает, что режим должен фактически использоваться постоянно; если другие режимы не запрещены, они могут применяться по взаимному согласованию.

3.27 терминал: Терминал – это любая конечная точка; он может быть терминалом пользователя или некоторой другой системой связи, например узлом управления многосторонней связью (MCU) или информационным сервером.

3.28 идентификатор TSAP: Часть информации, применяемая для мультиплексирования нескольких транспортных соединений одного типа в одном объекте H.323 со всеми транспортными соединениями, совместно использующими один и тот же адрес локальной сети (например, номер порта в среде TCP/UDP/IP). Идентификаторы TSAP могут (заранее) присваиваться некоторым международным органом или же могут выделяться динамически при установлении соединения. Динамически присвоенные идентификаторы TSAP являются временными, то есть их значения действуют только в течение одного вызова.

3.29 однонаправленный логический канал: Однонаправленный логический канал – это тракт для передачи одиночного элементарного потока от одного терминала к другому терминалу.

4 Сокращения

В настоящей Рекомендации используются следующие сокращения:

AAL	Уровень адаптации ATM
AL1,2,3	1, 2 и 3 уровни адаптации H.223
ACH.1	Абстрактно-синтаксическая нотация версии 1
ATM	Асинхронный способ передачи
B-LCSE	Объект сигнализации двунаправленного логического канала
CESE	Объект сигнализации обмена возможностями
CIF	Общий формат изображения (см. Рекомендации МСЭ-Т H.261 и H.263)
CLCSE	Объект сигнализации закрытия логического канала
CPCS	Общая часть подуровня конвергенции (5 уровня адаптации ATM)
DSM-CC	Цифровая запоминающая среда/Передача команд и управление
DTMF	Двухтональный многочастотный набор
GOB	Группа блоков (видеоизображения: см. Рекомендации МСЭ-Т H.261 и H.263)

KTCOPI	Коммутируемая телефонная сеть общего пользования
HDLC	Управление звеном данных верхнего уровня
HRD	Гипотетический эталонный декодер (см. Рекомендации МСЭ-Т Н.261 и Н.263)
IV	Вектор инициализации (используется для шифрования: см. Рекомендации МСЭ-Т Н.233 и Н.234)
LAPM	Протокол доступа модемов к каналу
LCSE	Объект сигнализации логического канала
MC	Объект многоточечного управления Н.323
MCU	Блок адресного управления
MLSE	Объект сигнализации цикла обслуживания
MPI	Минимальный интервал между изображениями
MRSE	Объект сигнализации запроса режима
MSDSE	Объект сигнализации определения отношения "ведущий-ведомый"
MTSE	Объект сигнализации таблицы мультиплексирования
PCR	Программный тактовый опорный сигнал (см. Рекомендации МСЭ-Т Н.222.0 ИСО/МЭК 13818-1)
PID	Идентификатор пакета (см. Рекомендацию МСЭ-Т Н.222.0 ИСО/МЭК 13818-1)
QCIF	Четвертичный общий формат изображения
RMESE	Объект сигнализации запроса элемента таблицы мультиплексирования
RTCP	Протокол управления транспортным потоком в реальном времени
RTDSE	Объект сигнализации задержки из-за подтверждения приема
RTP	Транспортный протокол реального времени
SDL	Язык спецификации и описания
СБД	Сервисный блок данных
SE	Сообщение об обмене сеансами (используется для шифрования, см. Рекомендации МСЭ-Т Н.233 и Н.234)
SQCIF	Формат Sub QCIF (Подмножество общего формата изображения)
STD	Системный целевой декодер (см. Рекомендацию МСЭ-Т Н.222.0 ИСО/МЭК 13818-1)
VC	Виртуальный канал АТМ

5 Введение

В настоящей Рекомендации описано несколько различных служб, причем предполагается, что некоторые из них будут применимы ко всем терминалам, работающим в соответствии с настоящим документом, а другие службы носят более специальный характер и определены для конкретных терминалов. Определены процедуры, обеспечивающие: обмен возможностями передачи аудиовизуальной информации и данных; запрос на передачу в некотором конкретном режиме обмена аудиовизуальной информацией и данными; управление логическими каналами, используемыми для транспортировки аудиовизуальной информации и данных; определение того, какой терминал является ведущим, а какой – ведомым (это нужно для управления логическими каналами); передачу различных сигналов управления и индикации; управление скоростью передачи данных по отдельным логическим каналам и по всему мультиплексному каналу; а также для измерения задержки из-за подтверждения при передаче от одного терминала к другому и обратно. Более подробное объяснение этих процедур приведено ниже.

После общего введения представлены разделы, в которых детально описаны синтаксис и семантика сообщений, а также процедуры. Синтаксис определен посредством нотации АСН.1 [40], а семантика

определяет значение синтаксических элементов и устанавливает синтаксические ограничения, которые не определены в синтаксисе АСН.1. В разделе, посвященном процедурам, определены протоколы, которые используют сообщения, определенные в других разделах.

Несмотря на то, что не все сообщения и процедуры, определенные в настоящей Рекомендации, будут применяться для всех терминалов, здесь такие ограничения не указаны. Эти ограничения содержатся в рекомендациях, использующих настоящую Рекомендацию.

По определению настоящая Рекомендация не зависит от используемого транспортного механизма, однако предполагается, что она применяется для надежного транспортного уровня, то есть транспортного уровня, который обеспечивает гарантированную доставку правильных данных.

5.1 Определение отношения "ведущий-ведомый"

Конфликты могут возникнуть в том случае, если два участвующие в установлении вызова терминала одновременно инициируют аналогичные события, при этом только одно подобное событие возможно или желательно, например, если ресурсов достаточно только для наступления одного экземпляра такого события. Для разрешения таких конфликтов один терминал должен играть роль ведущего терминала, а другой терминал должен играть роль ведомого терминала. Правила устанавливают, как ведущий и ведомый терминалы должны реагировать при конфликтах.

Процедура определения отношения "ведущий-ведомый" позволяет терминалам при вызове установить, какой терминал является ведущим, а какой – ведомым. Статус терминала может быть переопределен в любой момент в течение вызова; однако, терминал может инициировать процесс определения отношения "ведущий-ведомый" только в том случае, если никакая процедура, зависящая от результата этого процесса, не является локально активной.

5.2 Обмен возможностями

Процедуры обмена возможностями должны гарантировать, что будут передаваться только такие мультимедийные сигналы, которые могут приниматься и должным образом обрабатываться принимающим терминалом. Для этого требуется, чтобы возможности приема и декодирования каждого терминала были известны другому терминалу. Необязательно, чтобы терминал понимал или сохранял все поступающие возможности; те возможности, которые не были поняты, и возможности, которые не могут использоваться, должны игнорироваться, и не должно считаться, что возникла какая-либо ошибка. Если принимается возможность, которая содержит не понятое терминалом расширения, то эта возможность должна приниматься таким образом, как если бы она не содержала эти расширения.

Все возможности приема и декодирования различных сигналов терминалом сообщаются другому терминалу путем передачи его набора возможностей.

Возможности приема описывают способность терминала принимать и обрабатывать входящие потоки информации. Отправители должны ограничивать содержание передаваемой ими информации содержанием, для которого получатель указал возможность приема. Отсутствие возможности приема означает, что данный терминал не может осуществлять прием (что он является только передатчиком).

Возможности передачи описывают способность терминала передавать потоки информации. Возможности передачи используются для предложения получателю возможность выбора различных режимов работы, чтобы получатель мог запросить тот режим, в котором он предпочитает вести прием. Отсутствие возможности передачи указывает на то, что данный терминал не позволяет получателю выбирать предпочтительные режимы (но при этом сохраняется возможность любой передачи в пределах возможностей получателя).

Эти наборы возможностей позволяют одновременно передавать нескольких потоков с данным типом среды. Например, терминал может объявить о том, что он может одновременно принимать (или передавать) два независимых потока видеосигналов H.262 и два независимых потока аудиосигналов G.722. Сообщения о возможностях были определены таким образом, чтобы позволить терминалу указывать, что он не имеет фиксированных возможностей, но что они зависят от того, какие другие режимы используются одновременно. Например, можно заявить, что видеосигнал с более высоким разрешением может декодироваться при использовании более простого алгоритма для аудиосигнала, или что могут декодироваться либо две последовательности видеосигналов с низким разрешением или одна последовательность видеосигналов с высоким разрешением. Также можно указать компромиссы между возможностями передачи и приема.

Нестандартные возможности и управляющие сообщения можно определять с помощью структуры NonStandardParameter (нестандартный параметр). Отметим, что хотя значение нестандартных сообщений определяется отдельными организациями, оборудование любого производителя может передать любое нестандартное сообщение, если известно его значение.

Терминалы могут в любой момент времени повторно передавать наборы возможностей.

5.3 Процедуры сигнализации по логическому каналу

Протокол подтверждения предназначен для открытия и закрытия логических каналов, по которым передается аудиовизуальная информация и данные. Цель этих процедур – обеспечить способность терминала принимать и декодировать данные, которые будут передаваться по логическому каналу в то время, когда логический канал открыт, а не только тогда, когда по нему передаются первые данные, а также убедиться в том, что принимающий терминал способен получать и декодировать данные, которые будут передаваться по логическому каналу до начала передачи. Сообщение OpenLogicalChannel ("открыть логический канал") включает в себя описание передаваемых данных, например H.262 MP@ML при 6 Мбит/с. Логические каналы следует открывать только при наличии достаточных возможностей для одновременного приема данных по всем открытым логическим каналам.

Частично этот протокол посвящен открытию двунаправленных каналов. Для того чтобы избежать конфликтов, которые могут возникнуть, когда два терминала инициируют подобные события в одно и то же время, один терминал определяется как ведущий, а другой – как ведомый. Определен протокол, устанавливающий, какой именно терминал является ведущим, а какой – ведомым. Однако в системах, работающих на основе настоящей Рекомендации, для определения ведущего и ведомого терминалов может использоваться как процедура, определенная в настоящей Рекомендации, так и другое средство.

5.4 Запрос принимающего терминала о закрытии логического канала

Логический канал открывается и закрывается с передающей стороны. Определен механизм, который позволяет принимающему терминалу запрашивать закрытие входящего логического канала. Передающий терминал может принимать или отклонять запрос о закрытии логического канала. Например, терминал может использовать эти процедуры для запроса закрытия входящего логического канала, который по какой-либо причине не удается декодировать. Возможно также применение этих процедур, когда запрос о закрытии двунаправленного логического канала делает тот терминал, который не открывал этот канал.

5.5 Изменение элемента таблицы мультиплексирования H.223

Таблица мультиплексирования H.223 ставит в соответствие каждому окtetу в сообщении H.223 MUX конкретный номер логического канала. Таблица мультиплексирования H.223 может включать до 15 элементов. Предусмотрен механизм, который позволяет передающему терминалу задавать новые элементы таблицы мультиплексирования H.223 и передавать их получателю. Принимающий терминал также может запросить повторную передачу элемента таблицы мультиплексирования.

5.6 Запрос режима передачи аудиовизуальной информации и данных

После завершения протокола обмена возможностями оба терминала будут знать возможности передачи и приема друг друга, определенные дескрипторами возможностей, которыми они обменялись. Терминал не обязан заявлять все свои возможности; он должен объявить только о тех возможностях, которые он считает желательными.

Терминал может указывать свои возможности, связанные с передачей. Терминал, который получает возможности передачи от удаленного терминала, может запросить, чтобы передача ему велась в определенном режиме. Если терминал не передает возможности передачи, то это означает, что этот терминал не хочет, чтобы его режим передачи регулировался удаленным терминалом.

5.7 Определение задержки из-за подтверждения приема

В некоторых случаях полезно знать задержку из-за подтверждения приема, возникающую между передающим терминалом и принимающим терминалом. Предусмотрен механизм измерения этой задержки из-за подтверждения приема. Такой механизм также может использоваться как средство проверки того, работает ли удаленный терминал.

5.8 Циклы обслуживания

Определены процедуры создания циклов обслуживания. Можно задать цикл для отдельного логического канала или как цифровой цикл, или как цикл декодирования, а также цикл всего мультиплексного канала.

5.9 Команды и индикация

Предусмотрены команды и индикация, предназначенные для различных целей: их примерами могут служить сигналы активности/неактивности видео/аудио для информирования пользователя, а также запрос на быстрое обновление соединений при коммутации источника в многоточечных приложениях. Ни команды, ни индикация не вызывают ответных сообщений от удаленного терминала. Команды заставляют удаленный терминал совершить действие, тогда как индикация только сообщает информацию и не требует никакого действия.

Определена команда, позволяющая управлять скоростью передачи в логических каналах и во всем мультиплексном канале с удаленного терминала. Она используется в различных целях: межсетевого взаимодействия с терминалами, использующих мультиплексные каналы, у которых предусмотрено только конечное количество скоростей передачи; обеспечение многоточечных приложений, в которых необходимо согласовывать скорости передачи различных источников; а также управление потоком данных в перегруженных сетях.

Приложение А

Сообщения: Синтаксис

В этом приложении определяется синтаксис сообщений на основе нотации, определенной в АСН.1 [40]. Сообщения должны кодироваться для передачи с применением определенных в [42] правил пакетного кодирования, используя базовый тип выравнивания. Первый бит каждого передаваемого октета является самым старшим битом октета, как определено в Рекомендации МСЭ-Т Х.691 | ИСО/МЭК 8825-2.

```
MULTIMEDIA-SYSTEM-CONTROL DEFINITIONS AUTOMATIC TAGS ::=  
BEGIN
```

```
-- Экспорт всех символов
```

```
-- =====  
-- Сообщения верхнего уровня  
-- =====
```

```
MultimediaSystemControlMessage ::=CHOICE  
{  
    request RequestMessage,  
    response ResponseMessage,  
    command CommandMessage,  
    indication IndicationMessage,  
    ...  
}
```

```
-- Сообщение RequestMessage приводит к действию и требует немедленного ответа
```

```
RequestMessage ::=CHOICE  
{  
    nonStandard NonStandardMessage,
```

```

masterSlaveDetermination      MasterSlaveDetermination,

terminalCapabilitySet         TerminalCapabilitySet,

openLogicalChannel
closeLogicalChannel          OpenLogicalChannel,
                              CloseLogicalChannel,

requestChannelClose          RequestChannelClose,

multiplexEntrySend           MultiplexEntrySend,

requestMultiplexEntry        RequestMultiplexEntry,

requestMode                  RequestMode,

roundTripDelayRequest        RoundTripDelayRequest,

maintenanceLoopRequest      MaintenanceLoopRequest,

...,
communicationModeRequest     CommunicationModeRequest,
conferenceRequest            ConferenceRequest,

multilinkRequest             MultilinkRequest,
logicalChannelRateRequest    LogicalChannelRateRequest,
genericRequest               GenericMessage
}

-- Сообщение ResponseMessage является ответом на сообщение RequestMessage

ResponseMessage              ::=CHOICE
{
    nonStandard               NonStandardMessage,

    masterSlaveDeterminationAck MasterSlaveDeterminationAck,
    masterSlaveDeterminationReject MasterSlaveDeterminationReject,

    terminalCapabilitySetAck    TerminalCapabilitySetAck,
    terminalCapabilitySetReject TerminalCapabilitySetReject,

    openLogicalChannelAck      OpenLogicalChannelAck,
    openLogicalChannelReject   OpenLogicalChannelReject,
    closeLogicalChannelAck     CloseLogicalChannelAck,

    requestChannelCloseAck     RequestChannelCloseAck,
    requestChannelCloseReject  RequestChannelCloseReject,

    multiplexEntrySendAck      MultiplexEntrySendAck,
    multiplexEntrySendReject   MultiplexEntrySendReject,

    requestMultiplexEntryAck   RequestMultiplexEntryAck,
    requestMultiplexEntryReject RequestMultiplexEntryReject,

    requestModeAck            RequestModeAck,
    requestModeReject         RequestModeReject,

    roundTripDelayResponse     RoundTripDelayResponse,

    maintenanceLoopAck        MaintenanceLoopAck,
    maintenanceLoopReject     MaintenanceLoopReject,

    ...,
    communicationModeResponse  CommunicationModeResponse,

```

```

conferenceResponse      ConferenceResponse,
multilinkResponse       MultilinkResponse,
logicalChannelRateAcknowledge LogicalChannelRateAcknowledge,
logicalChannelRateReject LogicalChannelRateReject,
genericResponse         GenericMessage
}

-- Сообщение CommandMessage требует действия, но не требует ответа в явном виде

```

```

CommandMessage          ::=CHOICE
{
  nonStandard           NonStandardMessage,
  maintenanceLoopOffCommand MaintenanceLoopOffCommand,
  sendTerminalCapabilitySet SendTerminalCapabilitySet,
  encryptionCommand     EncryptionCommand,
  flowControlCommand    FlowControlCommand,
  endSessionCommand     EndSessionCommand,
  miscellaneousCommand  MiscellaneousCommand,
  ...,
  communicationModeCommand CommunicationModeCommand,
  conferenceCommand     ConferenceCommand,
  h223MultiplexReconfiguration H223MultiplexReconfiguration,
  newATMVCCCommand     NewATMVCCCommand,
  mobileMultilinkReconfigurationCommand MobileMultilinkReconfigurationCommand,
  genericCommand        GenericMessage
}

```

```

-- Сообщение IndicationMessage является информационным и не требует действия или
-- ответа

```

```

IndicationMessage      ::=CHOICE
{
  nonStandard          NonStandardMessage,
  functionNotUnderstood FunctionNotUnderstood,
  masterSlaveDeterminationRelease MasterSlaveDeterminationRelease,
  terminalCapabilitySetRelease TerminalCapabilitySetRelease,
  openLogicalChannelConfirm OpenLogicalChannelConfirm,
  requestChannelCloseRelease RequestChannelCloseRelease,
  multiplexEntrySendRelease MultiplexEntrySendRelease,
  requestMultiplexEntryRelease RequestMultiplexEntryRelease,
  requestModeRelease  RequestModeRelease,
}

```

```

miscellaneousIndication      MiscellaneousIndication,
jitterIndication             JitterIndication,
h223SkewIndication           H223SkewIndication,
newATMVCIndication           NewATMVCIndication,
userInput                    UserInputIndication,
...,
h2250MaximumSkewIndication   H2250MaximumSkewIndication,
mcLocationIndication         MCLocationIndication,
conferenceIndication          ConferenceIndication,
vendorIdentification          VendorIdentification,
functionNotSupported          FunctionNotSupported,
multilinkIndication           MultilinkIndication,
logicalChannelRateRelease     LogicalChannelRateRelease,
flowControlIndication         FlowControlIndication,
mobileMultilinkReconfigurationIndication MobileMultilinkReconfigurationIndication,
genericIndication             GenericMessage
}

-- SequenceNumber определяется здесь, как если бы он использовался в нескольких
-- сообщениях
SequenceNumber                ::=INTEGER (0..255)

-- =====
-- Определения универсального сообщения Generic Message
-- =====

GenericMessage                ::=SEQUENCE
{
    messageIdentifier          CapabilityIdentifier,
    subMessageIdentifier        INTEGER(0..127) OPTIONAL,

    messageContent              SEQUENCE OF GenericParameter OPTIONAL,
    ...
}

-- =====
-- Определения нестандартных сообщений
-- =====

NonStandardMessage            ::=SEQUENCE
{
    nonStandardData             NonStandardParameter,
    ...
}

NonStandardParameter          ::=SEQUENCE
{
    nonStandardIdentifier        NonStandardIdentifier,
    data                         OCTET STRING
}

```

```

NonStandardIdentifier                ::=CHOICE
{
    object                            OBJECT IDENTIFIER,
    h221NonStandard                   SEQUENCE
    {
        t35CountryCode                INTEGER (0..255),      -- страна по
                                                -- Дополнению А/Т.35
        t35Extension                   INTEGER (0..255),
                                                -- назначается в масштабе всей страны,
                                                -- кроме двоичного t35CountryCode
                                                -- 1111 1111, и в этом случае он должен
                                                -- содержать код страны согласно
                                                -- Приложению В/Т.35
        manufacturerCode               INTEGER (0..65535)   -- назначается в
                                                -- масштабе всей страны
    }
}

-- =====
-- Определение отношения "Ведущий-ведомый"
-- =====

MasterSlaveDetermination             ::=SEQUENCE
{
    terminalType                       INTEGER (0..255),
    statusDeterminationNumber          INTEGER (0..16777215),
    ...
}

MasterSlaveDeterminationAck          ::=SEQUENCE
{
    decision                           CHOICE
    {
        master                         NULL,
        slave                           NULL
    },
    ...
}

MasterSlaveDeterminationReject       ::=SEQUENCE
{
    cause                              CHOICE
    {
        identicalNumbers                NULL,
        ...
    },
    ...
}

MasterSlaveDeterminationRelease      ::=SEQUENCE
{
    ...
}

-- =====
-- Определения обмена возможностями
-- =====

TerminalCapabilitySet                ::=SEQUENCE
{
    sequenceNumber                     SequenceNumber,

```

```

protocolIdentifier          OBJECT IDENTIFIER,
                            -- должно быть установлено значение
                            -- {itu-t (0) recommendation (0) h (8) 245
                            -- version (0) 10}

multiplexCapability        MultiplexCapability OPTIONAL,

capabilityTable            SET SIZE (1..256) OF CapabilityTableEntry OPTIONAL,

capabilityDescriptors      SET SIZE (1..256) OF CapabilityDescriptor OPTIONAL,
...
}

CapabilityTableEntry       ::=SEQUENCE
{
    capabilityTableEntryNumber  CapabilityTableEntryNumber,
    capability                  Capability OPTIONAL
}

CapabilityDescriptor       ::=SEQUENCE
{
    capabilityDescriptorNumber  CapabilityDescriptorNumber,
    simultaneousCapabilities    SET SIZE (1..256) OF AlternativeCapabilitySet OPTIONAL
}

AlternativeCapabilitySet   ::=SEQUENCE SIZE (1..256) OF CapabilityTableEntryNumber

CapabilityTableEntryNumber ::=INTEGER (1..65535)

CapabilityDescriptorNumber ::=INTEGER (0..255)

TerminalCapabilitySetAck   ::=SEQUENCE
{
    sequenceNumber            SequenceNumber,
    ...
}

TerminalCapabilitySetReject ::=SEQUENCE
{
    sequenceNumber            SequenceNumber,
    cause                     CHOICE
    {
        unspecified          NULL,
        undefinedTableEntryUsed NULL,
        descriptorCapacityExceeded NULL,
        tableEntryCapacityExceeded CHOICE
        {
            highestEntryNumberProcessed CapabilityTableEntryNumber,
            noneProcessed              NULL
        },
        ...
    },
    ...
}

TerminalCapabilitySetRelease ::=SEQUENCE
{
    ...
}

```

```
-- =====
-- Определения обмена возможностями: описание возможности верхнего уровня
-- =====
```

```
Capability ::=CHOICE
{
    nonStandard NonStandardParameter,

    receiveVideoCapability VideoCapability,
    transmitVideoCapability VideoCapability,
    receiveAndTransmitVideoCapability VideoCapability,

    receiveAudioCapability AudioCapability,
    transmitAudioCapability AudioCapability,
    receiveAndTransmitAudioCapability AudioCapability,

    receiveDataApplicationCapability DataApplicationCapability,
    transmitDataApplicationCapability DataApplicationCapability,
    receiveAndTransmitDataApplicationCapability DataApplicationCapability,

    h233EncryptionTransmitCapability BOOLEAN,
    h233EncryptionReceiveCapability SEQUENCE
    {
        h233IVResponseTime INTEGER (0..255), -- единицы измерения
        -- - миллисекунды
    },
    ...,
    conferenceCapability ConferenceCapability,
    h235SecurityCapability H235SecurityCapability,
    maxPendingReplacementFor INTEGER (0..255),
    receiveUserInputCapability UserInputCapability,
    transmitUserInputCapability UserInputCapability,
    receiveAndTransmitUserInputCapability UserInputCapability,

    genericControlCapability GenericCapability,
    receiveMultiplexedStreamCapability MultiplexedStreamCapability,
    transmitMultiplexedStreamCapability MultiplexedStreamCapability,
    receiveAndTransmitMultiplexedStreamCapability MultiplexedStreamCapability,
    receiveRTPAudioTelephonyEventCapability AudioTelephonyEventCapability,
    receiveRTPAudioToneCapability AudioToneCapability,
    fecCapability FECCapability,
    multiplePayloadStreamCapability MultiplePayloadStreamCapability
}

```

```
H235SecurityCapability ::=SEQUENCE
{
    encryptionAuthenticationAndIntegrity EncryptionAuthenticationAndIntegrity,

    mediaCapability CapabilityTableEntryNumber,
    -- ПРИМЕЧАНИЕ. - Возможность mediaCapability должна ссылаться на
    -- Таблицу
    -- элементов данных возможностей, которая действительно содержит
    -- характеристики AudioCapability, VideoCapability,
    -- DataApplicationCapability для приема, передачи или приема и
    -- передачи, или сходные характеристики, которые обозначаются только
    -- параметром NonStandardParameter

    ...
}

```

```
-- =====
-- Определения обмена возможностями: Возможности мультиплексирования
-- =====
```

```
MultiplexCapability ::=CHOICE
{
    nonStandard NonStandardParameter,
    h222Capability H222Capability,
}

```



```

h223Capability          H223Capability,
v76Capability          V76Capability,
...
h2250Capability        H2250Capability,

genericMultiplexCapability  GenericCapability
}

H222Capability          ::= SEQUENCE
{
    numberOfVCs          INTEGER (1..256),
    vcCapability          SET OF VCCapability,
    ...
}

VCCapability           ::=SEQUENCE
{
    aall                 SEQUENCE
    {
        nullClockRecovery          BOOLEAN,
        srtsClockRecovery          BOOLEAN,
        adaptiveClockRecovery      BOOLEAN,
        nullErrorCorrection        BOOLEAN,
        longInterleaver            BOOLEAN,
        shortInterleaver          BOOLEAN,
        errorCorrectionOnly        BOOLEAN,
        structuredDataTransfer     BOOLEAN,
        partiallyFilledCells      BOOLEAN,
        ...
    } OPTIONAL,
    aal5                 SEQUENCE
    {
        forwardMaximumSDUSize      INTEGER (0..65535),  -- единицы измерения
                                                -- - октеты
        backwardMaximumSDUSize     INTEGER (0..65535),  -- единицы измерения
                                                -- - октеты

        ...
    } OPTIONAL,
    transportStream      BOOLEAN,
    programStream        BOOLEAN,
    availableBitRates    SEQUENCE
    {
        type              CHOICE
        {
            singleBitRate  INTEGER (1..65535),  -- единицы измерения
                                                -- - 64 кбит/с

            rangeOfBitRates SEQUENCE
            {
                lowerBitRate  INTEGER (1..65535),  -- единицы измерения
                                                -- - 64 кбит/с
                higherBitRate  INTEGER (1..65535)  -- единицы измерения
                                                -- - 64 кбит/с
            }
        },
        ...
    },
    ...
    aallViaGateway       SEQUENCE
    {
        gatewayAddress      SET SIZE(1..256) OF Q2931Address,
        nullClockRecovery  BOOLEAN,
        srtsClockRecovery  BOOLEAN,
        adaptiveClockRecovery  BOOLEAN,
        nullErrorCorrection  BOOLEAN,
        longInterleaver    BOOLEAN,
        shortInterleaver   BOOLEAN,
        errorCorrectionOnly  BOOLEAN,
    }
}

```

```

        structuredDataTransfer      BOOLEAN,
        partiallyFilledCells      BOOLEAN,
        ...
    } OPTIONAL
}

H223Capability ::=SEQUENCE
{
    transportWithI-frames          BOOLEAN,          -- транспорт H.245
                                         -- I-кадра

    videoWithAL1                  BOOLEAN,
    videoWithAL2                  BOOLEAN,
    videoWithAL3                  BOOLEAN,
    audioWithAL1                  BOOLEAN,
    audioWithAL2                  BOOLEAN,
    audioWithAL3                  BOOLEAN,
    dataWithAL1                   BOOLEAN,
    dataWithAL2                   BOOLEAN,
    dataWithAL3                   BOOLEAN,

    maximumAL2SDUSize             INTEGER (0..65535), -- единицы измерения
                                         -- - октеты
    maximumAL3SDUSize             INTEGER (0..65535), -- единицы измерения
                                         -- - октеты
    maximumDelayJitter            INTEGER (0..1023),  -- единицы измерения
                                         -- - миллисекунды

    h223MultiplexTableCapability  CHOICE
    {
        basic                      NULL,
        enhanced                    SEQUENCE
        {
            maximumNestingDepth    INTEGER (1..15),
            maximumElementListSize INTEGER (2..255),
            maximumSubElementListSize INTEGER (2..255),
            ...
        }
    },
    ...,
    maxMUXPDUSizeCapability        BOOLEAN,
    nsrpSupport                    BOOLEAN,
    mobileOperationTransmitCapability SEQUENCE
    {
        modeChangeCapability        BOOLEAN,
        h223AnnexA                  BOOLEAN,
        h223AnnexADoubleFlag        BOOLEAN,
        h223AnnexB                  BOOLEAN,
        h223AnnexBwithHeader        BOOLEAN,
        ...
    } OPTIONAL,
    h223AnnexCCapability           H223AnnexCCapability OPTIONAL,

    bitRate                        INTEGER (1..19200) OPTIONAL, -- единицы
                                         -- измерения -
                                         -- 100 бит/с

    mobileMultilinkFrameCapability SEQUENCE
    {
        maximumSampleSize           INTEGER (1..255), -- единицы измерения
                                         -- - октеты
        maximumPayloadLength        INTEGER (1..65025), -- единицы измерения
                                         -- - октеты
        ...
    } OPTIONAL
}

```

```

H223AnnexCCapability ::= SEQUENCE
{
    videoWithAL1M          BOOLEAN,
    videoWithAL2M          BOOLEAN,
    videoWithAL3M          BOOLEAN,
    audioWithAL1M          BOOLEAN,
    audioWithAL2M          BOOLEAN,
    audioWithAL3M          BOOLEAN,
    dataWithAL1M           BOOLEAN,
    dataWithAL2M           BOOLEAN,
    dataWithAL3M           BOOLEAN,
    alpduInterleaving      BOOLEAN,

    maximumAL1MPDUSize     INTEGER (0..65535), -- единицы измерения
                                                -- - октеты
    maximumAL2MSDUSize     INTEGER (0..65535), -- единицы измерения
                                                -- - октеты
    maximumAL3MSDUSize     INTEGER (0..65535), -- единицы измерения
                                                -- - октеты

    ...,
    rsCodeCapability       BOOLEAN OPTIONAL
}

V76Capability ::= SEQUENCE
{
    suspendResumeCapabilitywAddress BOOLEAN,
    suspendResumeCapabilitywoAddress BOOLEAN,
    rejCapability           BOOLEAN,
    sREJCapability         BOOLEAN,
    mREJCapability         BOOLEAN,
    crc8bitCapability      BOOLEAN,
    crc16bitCapability     BOOLEAN,
    crc32bitCapability     BOOLEAN,
    uihCapability          BOOLEAN,
    numOfDLCS              INTEGER (2..8191),
    twoOctetAddressFieldCapability BOOLEAN,
    loopBackTestCapability BOOLEAN,
    n401Capability         INTEGER (1..4095),
    maxWindowSizeCapability INTEGER (1..127),
    v75Capability          V75Capability,
    ...
}

V75Capability ::= SEQUENCE
{
    audioHeader            BOOLEAN,
    ...
}

H2250Capability ::= SEQUENCE
{
    maximumAudioDelayJitter INTEGER (0..1023), -- единицы измерения
                                                -- - миллисекунды

    receiveMultipointCapability MultipointCapability,
    transmitMultipointCapability MultipointCapability,
    receiveAndTransmitMultipointCapability MultipointCapability,
    mcCapability           SEQUENCE
    {
        centralizedConferenceMC    BOOLEAN,
        decentralizedConferenceMC  BOOLEAN,
        ...
    },
    rtcpVideoControlCapability    BOOLEAN, -- FIR и NACK
    mediaPacketizationCapability  MediaPacketizationCapability,
    ...,
    transportCapability           TransportCapability OPTIONAL,
    redundancyEncodingCapability  SEQUENCE SIZE (1..256) OF RedundancyEncodingCapability OPTIONAL,
}

```

```

        logicalChannelSwitchingCapability    BOOLEAN,
        t120DynamicPortCapability          BOOLEAN
    }

MediaPacketizationCapability                ::=SEQUENCE
{
    h261aVideoPacketization                BOOLEAN,
    ...,
    rtpPayloadType                          SEQUENCE SIZE(1..256) OF RTPPayloadType OPTIONAL
}

RSVPPParameters                            ::=SEQUENCE
{
    qosMode QOSMode OPTIONAL,
    tokenRate                               INTEGER (1..4294967295) OPTIONAL,
                                           -- скорость в байтах/с
    bucketSize                              INTEGER (1..4294967295) OPTIONAL,
                                           -- размер в байтах
    peakRate                                INTEGER (1..4294967295) OPTIONAL,
                                           -- пиковая ширина полосы в байтах/с
    minPoliced                              INTEGER (1..4294967295) OPTIONAL,
                                           --
    maxPktSize                              INTEGER (1..4294967295) OPTIONAL,
                                           -- размер в байтах
    ...
}

QOSMode                                    ::=CHOICE
{
    guaranteedQOS                           NULL,
    controlledLoad                           NULL,
    ...
}

ATMPParameters                             ::=SEQUENCE
{
    maxNTUSize                              INTEGER (0..65535), -- единицы измерения
                                           -- - октеты
    atmUBR                                  BOOLEAN,          -- неопределенная
                                           -- скорость передачи
    atmrtVBR                                BOOLEAN,          -- изменяющаяся в
                                           -- реальном времени
    atmmrtVBR                               BOOLEAN,          -- скорость передачи
                                           -- изменяющаяся не в
                                           -- реальном времени
    atmABR                                  BOOLEAN,          -- доступная скорость
                                           -- передачи
    atmCBR                                  BOOLEAN,          -- постоянная скорость
                                           -- передачи
    ...
}

QOSCapability                              ::=SEQUENCE
{
    nonStandardData                          NonStandardParameter OPTIONAL,
    rsvpParameters                           RSVPPParameters OPTIONAL,
    atmParameters                             ATMPParameters OPTIONAL,
    ...
}

MediaTransportType                          ::=CHOICE
{
    ip-UDP                                   NULL,
    ip-TCP                                   NULL,
    atm-AAL5-UNIDIR                          NULL, -- виртуальные каналы используются как однонаправленные
    atm-AAL5-BIDIR                           NULL, -- виртуальные каналы используются как двунаправленные
    ...,
}

```

```

    atm-AAL5-compressed          SEQUENCE
    {
        variable-delta          BOOLEAN,
        ...
    }
}

MediaChannelCapability          ::=SEQUENCE
{
    mediaTransport              MediaTransportType OPTIONAL,
    ...
}

TransportCapability             ::=SEQUENCE
{
    nonStandard                 NonStandardParameter OPTIONAL,
    qosCapabilities              SEQUENCE SIZE(1..256) OF QosCapability OPTIONAL,
    mediaChannelCapabilities     SEQUENCE SIZE(1..256) OF MediaChannelCapability OPTIONAL,
    ...
}

RedundancyEncodingCapability   ::=SEQUENCE
{
    redundancyEncodingMethod    RedundancyEncodingMethod,
    primaryEncoding              CapabilityTableEntryNumber,
    secondaryEncoding           SEQUENCE SIZE(1..256) OF CapabilityTableEntryNumber OPTIONAL,
    ...
}

RedundancyEncodingMethod       ::=CHOICE
{
    nonStandard                 NonStandardParameter,
    rtpAudioRedundancyEncoding  NULL,
    ...,
    rtpH263VideoRedundancyEncoding RTPH263VideoRedundancyEncoding
}

RTPH263VideoRedundancyEncoding ::= SEQUENCE
{
    numberOfThreads              INTEGER (1..16),
    framesBetweenSyncPoints      INTEGER (1..256),
    frameToThreadMapping         CHOICE
    {
        roundrobin               NULL,
        custom                    SEQUENCE SIZE(1..256) OF
        RTPH263VideoRedundancyFrameMapping,
        -- пустая SEQUENCE для согласования возможностей
        -- содержание имеет значение только для
        -- OpenLogicalChannel
        ...
    },
    containedThreads              SEQUENCE SIZE(1..256) OF INTEGER (0..15) OPTIONAL,
    -- используется только для открытия логических
    -- каналов
    ...
}

RTPH263VideoRedundancyFrameMapping ::= SEQUENCE
{
    threadNumber                 INTEGER (0..15),
    frameSequence                SEQUENCE SIZE(1..256) OF INTEGER (0..255),
    ...
}

```

```

MultipointCapability ::=SEQUENCE
{
    multicastCapability          BOOLEAN,
    multiUniCastConference      BOOLEAN,
    mediaDistributionCapability  SEQUENCE OF MediaDistributionCapability,
    ...
}

MediaDistributionCapability ::=SEQUENCE
{
    centralizedControl          BOOLEAN,
    distributedControl          BOOLEAN,
    centralizedAudio            BOOLEAN,
    distributedAudio            BOOLEAN,
    centralizedVideo            BOOLEAN,
    distributedVideo            BOOLEAN,
    centralizedData             SEQUENCE OF DataApplicationCapability OPTIONAL,
    distributedData             SEQUENCE OF DataApplicationCapability OPTIONAL,
    ...
}
-- для дальнейшего изучения
-- в Рекомендации МСЭ-Т Н.323

-- =====
-- Определения обмена возможностями: Возможности видео
-- =====

VideoCapability ::=CHOICE
{
    nonStandard                 NonStandardParameter,
    h261VideoCapability         H261VideoCapability,
    h262VideoCapability         H262VideoCapability,
    h263VideoCapability         H263VideoCapability,
    is11172VideoCapability      IS11172VideoCapability,
    ...,
    genericVideoCapability      GenericCapability,
    extendedVideoCapability     ExtendedVideoCapability
}

ExtendedVideoCapability ::= SEQUENCE
{
    videoCapability             SEQUENCE OF VideoCapability,
    videoCapabilityExtension    SEQUENCE OF GenericCapability OPTIONAL,
    ...
}

H261VideoCapability ::=SEQUENCE
{
    qcifMPI                     INTEGER (1..4) OPTIONAL, -- единица измерения
    cifMPI                       INTEGER (1..4) OPTIONAL, -- единица измерения
    temporalSpatialTradeOffCapability  BOOLEAN,
    maxBitRate                   INTEGER (1..19200), -- единица измерения
    stillImageTransmission      BOOLEAN, -- Приложение D/Н.261
    ...,
    videoBadMBsCap              BOOLEAN
}

```

```

H262VideoCapability ::=SEQUENCE
{
    profileAndLevel-SPatML          BOOLEAN,
    profileAndLevel-MPatLL          BOOLEAN,
    profileAndLevel-MPatML          BOOLEAN,
    profileAndLevel-MPatH-14        BOOLEAN,
    profileAndLevel-MPatHL          BOOLEAN,
    profileAndLevel-SNRatLL         BOOLEAN,
    profileAndLevel-SNRatML         BOOLEAN,
    profileAndLevel-SpatialatH-14  BOOLEAN,
    profileAndLevel-HPatML          BOOLEAN,
    profileAndLevel-HPatH-14        BOOLEAN,
    profileAndLevel-HPatHL          BOOLEAN,
    videoBitRate                    INTEGER (0.. 1073741823) OPTIONAL, -- единица измерения - 400 бит/с
    vbvBufferSize                   INTEGER (0.. 262143) OPTIONAL, -- единица измерения - 16384 бита
    samplesPerLine                   INTEGER (0..16383) OPTIONAL, -- единица измерения - отсчет/
    linesPerFrame                   INTEGER (0..16383) OPTIONAL, -- единица измерения - строк/кадр
    framesPerSecond                 INTEGER (0..15) OPTIONAL, -- код скорости кадров
    luminanceSampleRate             INTEGER (0..4294967295) OPTIONAL, -- единица измерения - отсчет/с
    . . . ,
    videoBadMBsCap                  BOOLEAN
}

H263VideoCapability ::=SEQUENCE
{
    sqcifMPI                        INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    qcifMPI                         INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    cifMPI                          INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    cif4MPI                        INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    cif16MPI                       INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    maxBitRate                      INTEGER (1..192400), -- единица измерения - 100 бит/с
    unrestrictedVector              BOOLEAN,
    arithmeticCoding                BOOLEAN,
    advancedPrediction              BOOLEAN,
    pbFrames                        BOOLEAN,
    temporalSpatialTradeOffCapability BOOLEAN,
    hrd-B                           INTEGER (0..524287) OPTIONAL, -- единица измерения - 128 битов
    bppMaxKb                        INTEGER (0..65535) OPTIONAL, -- единица измерения - 1024 бита
    . . . ,
    slowSqcifMPI                   INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    slowQcifMPI                    INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    slowCifMPI                     INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    slowCif4MPI                   INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    slowCif16MPI                  INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    errorCompensation              BOOLEAN,
    enhancementLayerInfo           EnhancementLayerInfo OPTIONAL,
    h263Options                    H263Options OPTIONAL
}

EnhancementLayerInfo ::=SEQUENCE
{
    baseBitRateConstrained          BOOLEAN,
    snrEnhancement                 SET SIZE(1..14) OF EnhancementOptions OPTIONAL,
    spatialEnhancement             SET SIZE(1..14) OF EnhancementOptions OPTIONAL,
    bPictureEnhancement            SET SIZE(1..14) OF BEnhancementParameters OPTIONAL,
    . . .
}

BEnhancementParameters ::=SEQUENCE
{
    enhancementOptions             EnhancementOptions,
    numberOfBPictures              INTEGER (1..64),
    . . .
}

```

```

EnhancementOptions ::=SEQUENCE
{
    sqcifMPI          INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    qcifMPI           INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    cifMPI            INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    cif4MPI           INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    cif16MPI          INTEGER (1..32) OPTIONAL, -- единица измерения - 1/29,97 Гц
    maxBitRate        INTEGER (1..192400),      -- единица измерения 100 бит/с
    unrestrictedVector    BOOLEAN,
    arithmeticCoding    BOOLEAN,
    temporalSpatialTradeOffCapability    BOOLEAN,
    slowSqcifMPI       INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    slowQcifMPI        INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    slowCifMPI         INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    slowCif4MPI        INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    slowCif16MPI       INTEGER (1..3600) OPTIONAL, -- единица измерения - секунд/кадр
    errorCompensation    BOOLEAN,
    h263Options         H263Options OPTIONAL,
    ...
}

H263Options ::= SEQUENCE
{
    advancedIntraCodingMode    BOOLEAN,
    deblockingFilterMode       BOOLEAN,
    improvedPBFramesMode       BOOLEAN,

    unlimitedMotionVectors     BOOLEAN,

    fullPictureFreeze          BOOLEAN,
    partialPictureFreezeAndRelease    BOOLEAN,
    resizingPartPicFreezeAndRelease    BOOLEAN,
    fullPictureSnapshot        BOOLEAN,
    partialPictureSnapshot     BOOLEAN,
    videoSegmentTagging        BOOLEAN,
    progressiveRefinement      BOOLEAN,

    dynamicPictureResizingByFour    BOOLEAN,
    dynamicPictureResizingSixteenthPel    BOOLEAN,
    dynamicWarpingHalfPel          BOOLEAN,
    dynamicWarpingSixteenthPel     BOOLEAN,

    independentSegmentDecoding     BOOLEAN,

    slicesInOrder-NonRect          BOOLEAN,
    slicesInOrder-Rect            BOOLEAN,
    slicesNoOrder-NonRect         BOOLEAN,
    slicesNoOrder-Rect           BOOLEAN,

    alternateInterVLCMode         BOOLEAN,
    modifiedQuantizationMode      BOOLEAN,
    reducedResolutionUpdate       BOOLEAN,

    transparencyParameters        TransparencyParameters OPTIONAL,
    separateVideoBackChannel      BOOLEAN,
    refPictureSelection           RefPictureSelection OPTIONAL,
    customPictureClockFrequency   SET SIZE (1..16) OF CustomPictureClockFrequency OPTIONAL,
    customPictureFormat           SET SIZE (1..16) OF CustomPictureFormat OPTIONAL,
    modeCombos                    SET SIZE (1..16) OF H263VideoModeCombos OPTIONAL,
    ...,
    videoBadMBsCap                BOOLEAN,
    h263Version3Options           H263Version3Options
}

```



```

TransparencyParameters ::= SEQUENCE
{
    presentationOrder      INTEGER(1..256),
    offset-x                INTEGER(-262144..262143), -- 1/8 пикселя
    offset-y                INTEGER(-262144..262143), -- 1/8 пикселя
    scale-x                 INTEGER(1..255),
    scale-y                 INTEGER(1..255),
    ...
}

RefPictureSelection ::= SEQUENCE
{
    additionalPictureMemory SEQUENCE
    {
        sqcifAdditionalPictureMemory  INTEGER(1..256) OPTIONAL, -- единица измерения - кадр
        qcifAdditionalPictureMemory   INTEGER(1..256) OPTIONAL, -- единица измерения - кадр
        cifAdditionalPictureMemory    INTEGER(1..256) OPTIONAL, -- единица измерения - кадр
        cif4AdditionalPictureMemory   INTEGER(1..256) OPTIONAL, -- единица измерения - кадр
        cif16AdditionalPictureMemory  INTEGER(1..256) OPTIONAL, -- единица измерения - кадр
        bigCpfAdditionalPictureMemory INTEGER(1..256) OPTIONAL, -- единица измерения - кадр
        ...
    } OPTIONAL,
    videoMux                BOOLEAN,
    videoBackChannelSend   CHOICE
    {
        none                NULL,
        ackMessageOnly     NULL,
        nackMessageOnly    NULL,
        ackOrNackMessageOnly NULL,
        ackAndNackMessage  NULL,
        ...
    },
    ...,
    enhancedReferencePicSelect SEQUENCE
    {
        subPictureRemovalParameters SEQUENCE
        {
            mpuHorizMBS      INTEGER(1..128),
            mpuVertMBS      INTEGER(1..72),
            mpuTotalNumber  INTEGER(1..65536),
            ...
        } OPTIONAL,
        ...
    }
}

CustomPictureClockFrequency ::= SEQUENCE
{
    clockConversionCode      INTEGER(1000..1001),
    clockDivisor             INTEGER(1..127),
    sqcifMPI                 INTEGER(1..2048) OPTIONAL,
    qcifMPI                  INTEGER(1..2048) OPTIONAL,
    cifMPI                   INTEGER(1..2048) OPTIONAL,
    cif4MPI                  INTEGER(1..2048) OPTIONAL,
    cif16MPI                 INTEGER(1..2048) OPTIONAL,
    ...
}

CustomPictureFormat ::= SEQUENCE
{
    maxCustomPictureWidth    INTEGER(1..2048), -- единица измерения -4 пикселя
    maxCustomPictureHeight  INTEGER(1..2048), -- единица измерения -4 пикселя
    minCustomPictureWidth   INTEGER(1..2048), -- единица измерения -4 пикселя
    minCustomPictureHeight  INTEGER(1..2048), -- единица измерения -4 пикселя
}

```

```

mPI                               SEQUENCE
{
  standardMPI                      INTEGER (1..31) OPTIONAL,
  customPCF                         SET SIZE (1..16) OF SEQUENCE
  {
    clockConversionCode            INTEGER (1000..1001),
    clockDivisor                   INTEGER (1..127),
    customMPI                      INTEGER (1..2048),
    ...
  } OPTIONAL,
  ...
},

pixelAspectInformation            CHOICE
{
  anyPixelAspectRatio             BOOLEAN,
  pixelAspectCode                 SET SIZE (1..14) OF INTEGER(1..14),
  extendedPAR                     SET SIZE (1..256) OF SEQUENCE
  {
    width                          INTEGER(1..255),
    height                         INTEGER(1..255),
    ...
  },
  ...
},
...
}

H263VideoModeCombos              ::= SEQUENCE
{
  h263VideoUncoupledModes         H263ModeComboFlags,
  h263VideoCoupledModes          SET SIZE (1..16) OF H263ModeComboFlags,
  ...
}

H263ModeComboFlags               ::= SEQUENCE
{
  unrestrictedVector              BOOLEAN,
  arithmeticCoding                BOOLEAN,
  advancedPrediction              BOOLEAN,
  pbFrames                        BOOLEAN,
  advancedIntraCodingMode         BOOLEAN,
  deblockingFilterMode            BOOLEAN,
  unlimitedMotionVectors          BOOLEAN,
  slicesInOrder-NonRect           BOOLEAN,
  slicesInOrder-Rect              BOOLEAN,
  slicesNoOrder-NonRect           BOOLEAN,
  slicesNoOrder-Rect              BOOLEAN,
  improvedPBframesMode            BOOLEAN,
  referencePicSelect              BOOLEAN,
  dynamicPictureResizingByFour    BOOLEAN,
  dynamicPictureResizingSixteenthPel  BOOLEAN,
  dynamicWarpingHalfPel           BOOLEAN,
  dynamicWarpingSixteenthPel      BOOLEAN,
  reducedResolutionUpdate         BOOLEAN,
  independentSegmentDecoding      BOOLEAN,
  alternateInterVLCMode           BOOLEAN,
  modifiedQuantizationMode        BOOLEAN,
  ...,
  enhancedReferencePicSelect      BOOLEAN,
  h263Version3Options             H263Version3Options}

```

```

H263Version3Options                               ::=SEQUENCE
{
    dataPartitionedSlices                          BOOLEAN,
    fixedPointIDCT0                                BOOLEAN,
    interlacedFields                                BOOLEAN,
    currentPictureHeaderRepetition                 BOOLEAN,
    previousPictureHeaderRepetition                BOOLEAN,
    nextPictureHeaderRepetition                    BOOLEAN,
    pictureNumber                                   BOOLEAN,
    spareReferencePictures                          BOOLEAN,
    ...
}

IS11172VideoCapability                             ::=SEQUENCE
{
    constrainedBitstream                            BOOLEAN,
    videoBitRate                                    INTEGER (0.. 1073741823) OPTIONAL, -- единица измерения - 400 бит/с
    vbvBufferSize                                  INTEGER (0.. 262143) OPTIONAL, -- единица измерения - 16384 бита
    samplesPerLine                                  INTEGER (0..16383) OPTIONAL, -- единица измерения - отсчет/
    linesPerFrame                                   INTEGER (0..16383) OPTIONAL, -- единица измерения - строк/
    pictureRate INTEGER (0..15) OPTIONAL,           кадр
    luminanceSampleRate                             INTEGER (0..4294967295) OPTIONAL, -- единица измерения - отсчет/с
    ... ,
    videoBadMBsCap                                  BOOLEAN
}

-- =====
-- Определения обмена возможностями: Возможности аудио
-- =====

-- Для мультиплексной передачи H.222 целые означают размер буфера STD в единицах
-- измерения по 256 октетов
-- Для мультиплексной передачи H.223 целые означают максимальное число
-- аудиокадров на AL-SDU
-- Для мультиплексной передачи H.225.0 целые означают максимальное число
-- аудиокадров на пакет

AudioCapability                                   ::=CHOICE
{
    nonStandard                                     NonStandardParameter,
    g711Alaw64k                                     INTEGER (1..256),
    g711Alaw56k                                     INTEGER (1..256),
    g711Ulaw64k                                     INTEGER (1..256),
    g711Ulaw56k                                     INTEGER (1..256),

    g722-64k                                        INTEGER (1..256),
    g722-56k                                        INTEGER (1..256),
    g722-48k                                        INTEGER (1..256),

    g7231                                           SEQUENCE
    {
        maxAl-sduAudioFrames                       INTEGER (1..256),
        silenceSuppression                          BOOLEAN
    },

    g728                                              INTEGER (1..256),
    g729                                              INTEGER (1..256),
    g729AnnexA                                       INTEGER (1..256),
    is11172AudioCapability                           IS11172AudioCapability,
    is13818AudioCapability                           IS13818AudioCapability,
    ... ,
    g729wAnnexB                                       INTEGER (1..256),
    g729AnnexAwAnnexB                               INTEGER (1..256),
    g7231AnnexCCapability                           G7231AnnexCCapability,

```

```

gsmFullRate          GSMAudioCapability,
gsmHalfRate          GSMAudioCapability,
gsmEnhancedFullRate GSMAudioCapability,
genericAudioCapability,
g729Extensions      G729Extensions,
vbd                  VBDCapability,
audioTelephonyEvent NoPTAudioTelephonyEventCapability,
audioTone            NoPTAudioToneCapability
}

G729Extensions ::= SEQUENCE
{
    audioUnit          INTEGER (1..256) OPTIONAL,
    annexA             BOOLEAN,
    annexB             BOOLEAN,
    annexD             BOOLEAN,
    annexE             BOOLEAN,
    annexF             BOOLEAN,
    annexG             BOOLEAN,
    annexH             BOOLEAN,
    ...
}

G7231AnnexCCapability ::= SEQUENCE
{
    maxAl-sduAudioFrames INTEGER (1..256),
    silenceSuppression   BOOLEAN,
    g723AnnexCAudioMode SEQUENCE
    {
        highRateMode0    INTEGER (27..78), -- единицы измерения - октеты
        highRateMode1    INTEGER (27..78), -- единицы измерения - октеты
        lowRateMode0     INTEGER (23..66), -- единицы измерения - октеты
        lowRateMode1     INTEGER (23..66), -- единицы измерения - октеты
        sidMode0         INTEGER (6..17),  -- единицы измерения - октеты
        sidMode1         INTEGER (6..17),  -- единицы измерения - октеты
        ...
    } OPTIONAL,
    ...
}

IS11172AudioCapability ::= SEQUENCE
{
    audioLayer1         BOOLEAN,
    audioLayer2         BOOLEAN,
    audioLayer3         BOOLEAN,

    audioSampling32k    BOOLEAN,
    audioSampling44k1   BOOLEAN,
    audioSampling48k    BOOLEAN,

    singleChannel       BOOLEAN,
    twoChannels         BOOLEAN,

    bitRate             INTEGER (1..448), -- единицы измерения - кбит/с
    ...
}

IS13818AudioCapability ::= SEQUENCE
{
    audioLayer1         BOOLEAN,
    audioLayer2         BOOLEAN,
    audioLayer3         BOOLEAN,

```

```

audioSampling16k          BOOLEAN,
audioSampling22k05       BOOLEAN,
audioSampling24k         BOOLEAN,
audioSampling32k         BOOLEAN,
audioSampling44k1       BOOLEAN,
audioSampling48k         BOOLEAN,

singleChannel            BOOLEAN,
twoChannels              BOOLEAN,
threeChannels2-1        BOOLEAN,
threeChannels3-0        BOOLEAN,
fourChannels2-0-2-0     BOOLEAN,
fourChannels2-2         BOOLEAN,
fourChannels3-1         BOOLEAN,
fiveChannels3-0-2-0     BOOLEAN,
fiveChannels3-2         BOOLEAN,

lowFrequencyEnhancement  BOOLEAN,

multilingual             BOOLEAN,

bitRate                  INTEGER (1..1130), -- единицы измерения - кбит/с
...
}

GSMAudioCapability ::= SEQUENCE
{
    audioUnitSize      INTEGER (1..256),
    comfortNoise       BOOLEAN,
    scrambled           BOOLEAN,
    ...
}

VBDCapability ::= SEQUENCE
{
    type                AudioCapability, -- не должен быть равен "vbd"
    ...
}

-- =====
-- Определения обмена возможностями: Возможности передачи данных
-- =====

DataApplicationCapability ::= SEQUENCE
{
    application        CHOICE
    {
        nonStandard    NonStandardParameter,
        t120            DataProtocolCapability,
        dsm-cc         DataProtocolCapability,
        userData       DataProtocolCapability,
        t84SEQUENCE    {
            t84Protocol DataProtocolCapability,
            t84Profile  T84Profile
        },
        t434           DataProtocolCapability,
        h224           DataProtocolCapability,
        nlpid         SEQUENCE
        {
            nlpidProtocol DataProtocolCapability,
            nlpidData     OCTET STRING
        },
    }
}

```

```

        dsvdControl                NULL,
        h222DataPartitioning       DataProtocolCapability,
        ...,
        t30fax                     DataProtocolCapability,
        t140                       DataProtocolCapability,
        t38fax                     SEQUENCE
        {
            t38FaxProtocol          DataProtocolCapability,
            t38FaxProfile           T38FaxProfile
        },
        genericDataCapability       GenericCapability
    },
    maxBitRate                     INTEGER (0..4294967295), -- единицы измерения - 100 бит/с
    ...
}

DataProtocolCapability             ::=CHOICE
{
    nonStandard                    NonStandardParameter,
    v14buffered                   NULL,
    v42lapm                        NULL, -- может согласовываться с V.42 bis
    hdlcFrameTunnelling           NULL,
    h310SeparateVCStack           NULL,
    h310SingleVCStack             NULL,
    transparent                    NULL,
    ...,
    segmentationAndReassembly     NULL,
    hdlcFrameTunnelingwSAR        NULL,
    v120                           NULL, -- как в H.230
    separateLANStack              NULL,
    v76wCompression              CHOICE
    {
        transmitCompression        CompressionType,
        receiveCompression         CompressionType,
        transmitAndReceiveCompression CompressionType,
        ...
    },
    tcp                            NULL,
    udp                            NULL
}

CompressionType                   ::=CHOICE
{
    v42bis                        V42bis,
    ...
}

V42bis                            ::=SEQUENCE
{
    numberOfCodewords              INTEGER (1..65536),
    maximumStringLength            INTEGER (1..256),
    ...
}

T84Profile                         ::=CHOICE
{
    t84Unrestricted               NULL,
    t84Restricted                 SEQUENCE
    {
        qcif                      BOOLEAN,
        cif                       BOOLEAN,
        ccir601Seq                BOOLEAN,
        ccir601Prog               BOOLEAN,
        hdtvSeq                   BOOLEAN,
    }
}

```

```

        hdtvProg                                BOOLEAN,

        g3FacsMH200x100                        BOOLEAN,
        g3FacsMH200x200                        BOOLEAN,
        g4FacsMMR200x100                       BOOLEAN,
        g4FacsMMR200x200                       BOOLEAN,
        jbig200x200Seq                          BOOLEAN,
        jbig200x200Prog                         BOOLEAN,
        jbig300x300Seq                          BOOLEAN,
        jbig300x300Prog                         BOOLEAN,

        digPhotoLow                            BOOLEAN,
        digPhotoMedSeq                          BOOLEAN,
        digPhotoMedProg                         BOOLEAN,
        digPhotoHighSeq                         BOOLEAN,
        digPhotoHighProg                       BOOLEAN,

        ...
    }
}

T38FaxProfile ::= SEQUENCE
{
    fillBitRemoval                            BOOLEAN,
    transcodingJBIG                           BOOLEAN,
    transcodingMMR                             BOOLEAN,
    ...,
    version                                    INTEGER (0..255),
    -- По умолчанию версия 0 относится к
    -- Т.38 (1998)

    t38FaxRateManagement                       T38FaxRateManagement,
    -- По умолчанию управление скоростью
    -- передачи данных определяется выбором
    -- DataProtocolCapability

    t38FaxUdpOptions                           T38FaxUdpOptions OPTIONAL,
    -- Для UDP по умолчанию используется
    -- t38UDPRedundancy

    t38FaxTcpOptions                           T38FaxTcpOptions OPTIONAL
}

T38FaxRateManagement ::= CHOICE
{
    localTCF                                  NULL,
    transferredTCF                             NULL,
    ...
}

T38FaxUdpOptions ::= SEQUENCE
{
    t38FaxMaxBuffer                            INTEGER OPTIONAL,
    t38FaxMaxDatagram                           INTEGER OPTIONAL,
    t38FaxUdpEC                                 CHOICE
    {
        t38UDPFEC                               NULL,
        t38UDPRedundancy                         NULL,
        ...
    }
}

T38FaxTcpOptions ::= SEQUENCE
{
    t38TCPBidirectionalMode                     BOOLEAN,
    ...
}

```

```

-- =====
-- Определения возможностей шифрования
-- =====

EncryptionAuthenticationAndIntegrity ::=SEQUENCE
{
    encryptionCapability           EncryptionCapability OPTIONAL,
    authenticationCapability       AuthenticationCapability OPTIONAL,
    integrityCapability             IntegrityCapability OPTIONAL,
    ...
}

EncryptionCapability               ::=SEQUENCE SIZE(1..256) OF MediaEncryptionAlgorithm

MediaEncryptionAlgorithm           ::=CHOICE
{
    nonStandard                    NonStandardParameter,
    algorithm                       OBJECT IDENTIFIER, -- многие определены
                                                    -- в ISO/IEC 9979
    ...
}

AuthenticationCapability           ::=SEQUENCE
{
    nonStandard                    NonStandardParameter OPTIONAL,
    ...,
    antiSpamAlgorithm              OBJECT IDENTIFIER OPTIONAL
}

IntegrityCapability                ::=SEQUENCE
{
    nonStandard                    NonStandardParameter OPTIONAL,
    ...
}

-- =====
-- Определения обмена возможностями: Ввод данных пользователем
-- =====

UserInputCapability                ::= CHOICE
{
    nonStandard                    SEQUENCE SIZE(1..16) OF NonStandardParameter,
    basicString                     NULL, -- алфавитно-цифровой
    iA5String                       NULL, -- алфавитно-цифровой
    generalString                   NULL, -- алфавитно-цифровой
    dtmf                            NULL, -- поддерживает dtmf, используя
                                                    -- signal и signalUpdate
    hookflash                       NULL, -- поддерживает hookflash,
                                                    -- используя signal
    ...,
    extendedAlphanumeric            NULL,
    encryptedBasicString            NULL, -- зашифрованная базовая строка в
                                                    -- encryptedAlphanumeric
    encryptedIA5String              NULL, -- зашифрованная строка IA5 в
                                                    -- encryptedSignalType
    encryptedGeneralString          NULL, -- зашифрованная общая строка в
                                                    -- extendedAlphanumeric.encryptedalphanumeric
    securedTMF                      NULL -- безопасный DTMF, используя
                                                    -- encryptedSignalType
}

-- =====
-- Определения обмена возможностями: Конференция
-- =====

```



```

ConferenceCapability ::=SEQUENCE
{
    nonStandardData          SEQUENCE OF NonStandardParameter OPTIONAL,
    chairControlCapability   BOOLEAN,
    ...,
    videoIndicateMixingCapability  BOOLEAN,
    multipointVisualizationCapability  BOOLEAN OPTIONAL      -- так же как H.230 MVC
}

```

```

-- =====
-- Определения обмена возможностями: Универсальная возможность
-- =====

```

```

GenericCapability ::=SEQUENCE
{
    capabilityIdentifier      CapabilityIdentifier,

    maxBitRate               INTEGER (0..4294967295) OPTIONAL,
                            -- единица измерения 100 бит/с
    collapsing                SEQUENCE OF GenericParameter OPTIONAL,
    nonCollapsing            SEQUENCE OF GenericParameter OPTIONAL,
    nonCollapsingRaw         OCTET STRING OPTIONAL,
                            -- Обычно содержит возможность описания
                            -- кодирования данных PER ASN.1
    transport                DataProtocolCapability OPTIONAL,
    ...
}

```

```

CapabilityIdentifier ::=CHOICE
{
    standard                 OBJECT IDENTIFIER,
                            -- например, { itu-t (0) recommendation (0) h (8) 267
                            -- version (0) 2 subIdentifier (0) }
    h221NonStandard         NonStandardParameter,
    uuid                    OCTET STRING ( SIZE (16) ),
    domainBased             IA5String ( SIZE (1..64) ),
    ...
}

```

```

-- ПРИМЕЧАНИЕ. - Диапазоны значений параметров выбраны так, чтобы заголовок
-- GenericParameter, стандартная часть идентификатора параметра,
-- результат кодирования данного выбора и заголовок значения параметра
-- умещались в 2 октета.

```

```

GenericParameter ::=SEQUENCE
{
    parameterIdentifier      ParameterIdentifier,
    parameterValue          ParameterValue,
    supersedes              SEQUENCE OF ParameterIdentifier OPTIONAL,
    ...
}

```

```

ParameterIdentifier ::=CHOICE
{
    standard                 INTEGER (0..127), -- Назначено
                            -- спецификацией
                            -- ВОЗМОЖНОСТИ
    h221NonStandard         NonStandardParameter, -- Примечание.
                            -- NonStandardIdentifier
                            -- является недостаточным
                            -- в этом случае
    uuid                    OCTET STRING ( SIZE (16) ), -- Для нестан-
                            -- дартных
    domainBased             IA5String ( SIZE (1..64) ),
    ...
}

```

```

}

ParameterValue ::=CHOICE
{
    logical          NULL,          -- Доступен, только
                                -- если все объекты
                                -- включают эту опцию

    booleanArray     INTEGER (0..255), -- массив 8 логических
                                -- типов

    unsignedMin      INTEGER (0..65535), -- Искать минимальное
                                -- общее значение

    unsignedMax      INTEGER (0..65535), -- Искать максимальное
                                -- общее значение

    unsigned32Min    INTEGER (0..4294967295), -- Искать минимальное
                                -- общее значение

    unsigned32Max    INTEGER (0..4294967295), -- Искать максимальное
                                -- общее значение

    octetString      OCTET STRING,    -- несвертываемая
                                -- строка октетов

    genericParameter SEQUENCE OF GenericParameter,
    ...
}

-- =====
-- Определения обмена возможностями: Возможность мультиплексированного потока
-- =====

MultiplexedStreamCapability ::=SEQUENCE
{
    multiplexFormat      MultiplexFormat,
    controlOnMuxStream   BOOLEAN,
    capabilityOnMuxStream SET SIZE (1..256) OF
AlternativeCapabilitySet OPTIONAL,
    ...
}

MultiplexFormat ::= CHOICE
{
    nonStandard          NonStandardParameter,
    h222Capability       H222Capability,
    h223Capability       H223Capability,
    ...
}

-- =====
-- Определения обмена возможностями: AudioTelephonyEventCapability и
-- AudioToneCapability
-- =====

AudioTelephonyEventCapability ::=SEQUENCE
{
    dynamicRTPPayloadType INTEGER(96..127),
    audioTelephoneEvent   GeneralString, -- Как в <перечне значений>
                                -- в 3.9/RFC 2833
    ...
}

AudioToneCapability ::=SEQUENCE
{
    dynamicRTPPayloadType INTEGER(96..127),
    ...
}

```

```

-- Следующие определения - такие, как выше, но без поля Payload Type.

NoPTAudioTelephonyEventCapability ::=SEQUENCE
{
    audioTelephoneEvent      GeneralString, -- Как в <перечне значений>
                                -- в 3.9/RFC 2833
    ...
}

NoPTAudioToneCapability ::=SEQUENCE
{
    ...
}

-----
-- Определения обмена возможностями: MultiplePayloadStreamCapability
-----

MultiplePayloadStreamCapability ::=SEQUENCE
{
    capabilities              SET SIZE(1..256) OF AlternativeCapabilitySet,
    ...
}

-----
-- Определения обмена возможностями: FECCapability
-----

FECCapability ::=CHOICE
{
    rfc2733                    SEQUENCE
    {
        redundancyEncoding     BOOLEAN,
        separateStream          SEQUENCE
        {
            separatePort        BOOLEAN,
            samePort             BOOLEAN,
            ...
        },
        ...
    },
    ...
}

-----
-- Определения сигнализации в логическом канале
-----

-- "Прямая" используется для ссылки на передачу в прямом направлении от
-- терминала, формирующего исходный запрос в логическом канале, к другому
-- терминалу, а "обратная" используется для ссылки на передачу в обратном
-- направлении в случае запроса двунаправленного канала.

OpenLogicalChannel ::=SEQUENCE
{
    forwardLogicalChannelNumber LogicalChannelNumber,

    forwardLogicalChannelParameters SEQUENCE
    {
        portNumber              INTEGER (0..65535) OPTIONAL,
        dataType                 DataType,
        multiplexParameters     CHOICE
        {

```

```

        h222LogicalChannelParameters      H222LogicalChannelParameters,
        h223LogicalChannelParameters      H223LogicalChannelParameters,
        v76LogicalChannelParameters       V76LogicalChannelParameters,
        ...,
        h2250LogicalChannelParameters     H2250LogicalChannelParameters,
        none                               NULL -- для использования в отдельном
                                           -- стеке, когда multiplexParameters
                                           -- не требуются или не являются
                                           -- соответствующими
    },
    ...,
    forwardLogicalChannelDependency       LogicalChannelNumber OPTIONAL,
                                           -- также используются для ссылки на
                                           -- первичный логический канал в случае
                                           -- использования избыточного кодирования
                                           -- видеосигнала
    replacementFor                       LogicalChannelNumber OPTIONAL
},

-- Используется, чтобы определить обратный канал для запроса на открытие
-- двунаправленного канала

reverseLogicalChannelParameters SEQUENCE
{
    dataType                             DataType,
    multiplexParameters                   CHOICE
    {
        -- параметры H.222 никогда не присутствуют в обратном направлении
        h223LogicalChannelParameters     H223LogicalChannelParameters,
        v76LogicalChannelParameters       V76LogicalChannelParameters,
        ...,
        h2250LogicalChannelParameters     H2250LogicalChannelParameters
    } OPTIONAL, -- не присутствует для H.222
    ...,
    reverseLogicalChannelDependency       LogicalChannelNumber OPTIONAL,
        -- также используются для ссылки на первичный логический канал в
        -- случае использования избыточного кодирования видеосигнала
    replacementFor                       LogicalChannelNumber OPTIONAL
} OPTIONAL, -- Не присутствует в запросах однонаправленного
-- канала

...,
separateStack                           NetworkAccessParameters OPTIONAL,
    -- для ответчика на Open при организации стека
encryptionSync                           EncryptionSync OPTIONAL -- используется только
-- ведущим терминалом
}

LogicalChannelNumber ::= INTEGER (1..65535)

NetworkAccessParameters ::= SEQUENCE
{
    distribution CHOICE
    {
        unicast NULL,
        multicast NULL, -- Для дальнейшего изучения в T.120
        ...
    } OPTIONAL,
    networkAddress CHOICE
    {
        q2931Address Q2931Address,
        e164Address IA5String(SIZE(1..128)) (FROM ("0123456789#*")),

```

localAreaAddress	TransportAddress,
...	
},	
associateConference	BOOLEAN,
externalReference	OCTET STRING(SIZE(1..255)) OPTIONAL,
...	
t120SetupProcedure	CHOICE
{	
originateCall	NULL,
waitForCall	NULL,
issueQuery	NULL,
...	
} OPTIONAL	
}	
Q2931Address	::=SEQUENCE
{	
address	CHOICE
{	
internationalNumber	NumericString(SIZE(1..16)),
nsapAddress	OCTET STRING(SIZE(1..20)),
...	
},	
subaddress	OCTET STRING(SIZE(1..20)) OPTIONAL,
...	
}	
V75Parameters	::= SEQUENCE
{	
audioHeaderPresent	BOOLEAN,
...	
}	
DataType	::=CHOICE
{	
nonStandard	NonStandardParameter,
nullData	NULL,
videoData	VideoCapability,
audioData	AudioCapability,
data	DataApplicationCapability,
encryptionData	EncryptionMode,
...	
h235Control	NonStandardParameter,
h235Media	H235Media,
multiplexedStream	MultiplexedStreamParameter,
redundancyEncoding	RedundancyEncoding,
multiplePayloadStream	MultiplePayloadStream,
fec	FECDATA
}	
H235Media	::=SEQUENCE
{	
encryptionAuthenticationAndIntegrity	EncryptionAuthenticationAndIntegrity,
mediaType	CHOICE
{	
nonStandard	NonStandardParameter,
videoData	VideoCapability,
audioData	AudioCapability,
data	DataApplicationCapability,
...	

```

        redundancyEncoding      RedundancyEncoding,
        multiplePayloadStream    MultiplePayloadStream,
        fecFECData
    },
    ...
}

MultiplexedStreamParameter ::=SEQUENCE
{
    multiplexFormat              MultiplexFormat,
    controlOnMuxStream          BOOLEAN,
    ...
}

H222LogicalChannelParameters ::=SEQUENCE
{
    resourceID                   INTEGER (0..65535),
    subChannelID                 INTEGER (0..8191),
    pcr-pid                       INTEGER (0..8191) OPTIONAL,
    programDescriptors           OCTET STRING OPTIONAL,
    streamDescriptors            OCTET STRING OPTIONAL,
    ...
}

H223LogicalChannelParameters ::=SEQUENCE
{
    adaptationLayerType          CHOICE
    {
        nonStandard              NonStandardParameter,
        allFramed                 NULL,
        allNotFramed             NULL,
        al2WithoutSequenceNumbers NULL,
        al2WithSequenceNumbers   NULL,
        al3                       SEQUENCE
        {
            controlFieldOctets   INTEGER (0..2),
            sendBufferSize        INTEGER (0..16777215) -- единицы измерения - октеты
        },
        ...,
        allM                      H223AL1MParameters,
        al2M                      H223AL2MParameters,
        al3M                      H223AL3MParameters
    },

    segmentableFlag             BOOLEAN,
    ...
}

H223AL1MParameters ::=SEQUENCE
{
    transferMode                 CHOICE
    {
        framed                   NULL,
        unframed                 NULL,
        ...
    },
    headerFEC                   CHOICE
    {
        sebch16-7                NULL,
        golay24-12               NULL,
        ...
    },
}

```

```

    crcLength                               CHOICE
    {
        crc4bit                             NULL,
        crc12bit                            NULL,
        crc20bit                            NULL,
        crc28bit                            NULL,
        ...,
        crc8bit                             NULL,
        crc16bit                            NULL,
        crc32bit                            NULL,
        crcNotUsed                          NULL
    },

    rcpcCodeRate                            INTEGER (8..32),

    arqType                                 CHOICE
    {
        noArq                               NULL,
        typeIArq                            H223AnnexCArqParameters,
        typeIIArq                           H223AnnexCArqParameters,
        ...
    },
    alpduInterleaving                       BOOLEAN,
    alsduSplitting                          BOOLEAN,
    ...,
    rsCodeCorrection                        INTEGER (0..127) OPTIONAL
}

H223AL2MParameters ::=SEQUENCE
{
    headerFEC                               CHOICE
    {
        sebch16-5                           NULL,
        golay24-12                          NULL,
        ...
    },
    alpduInterleaving                       BOOLEAN,
    ...
}

H223AL3MParameters ::=SEQUENCE
{
    headerFormat                            CHOICE
    {
        sebch16-7                           NULL,
        golay24-12                          NULL,
        ...
    },
    crcLength                               CHOICE
    {
        crc4bit                             NULL,
        crc12bit                            NULL,
        crc20bit                            NULL,
        crc28bit                            NULL,
        ...,
        crc8bit                             NULL,
        crc16bit                            NULL,
        crc32bit                            NULL,
        crcNotUsed                          NULL
    },

```

```

rcpcCodeRate                INTEGER (8..32),

argType                      CHOICE
{
    noArq                    NULL,
    typeIArq                 H223AnnexCArqParameters,
    typeIIArq                H223AnnexCArqParameters,
    ...
},

alpduInterleaving           BOOLEAN,
...,
rsCodeCorrection            INTEGER (0..127) OPTIONAL
}

H223AnnexCArqParameters     ::=SEQUENCE
{
    numberOfRetransmissions  CHOICE
    {
        finite               INTEGER (0..16),
        infinite             NULL,
        ...
    },
    sendBufferSize           INTEGER (0..16777215), -- единицы измерения - октеты
    ...
}

V76LogicalChannelParameters ::=SEQUENCE
{
    hdlcParameters           V76HDLParameters,
    suspendResume            CHOICE
    {
        noSuspendResume     NULL,
        suspendResumewAddress NULL,
        suspendResumewoAddress NULL,
        ...
    },
    uIH                      BOOLEAN,
    mode                      CHOICE
    {
        eRM                  SEQUENCE
        {
            windowSize       INTEGER (1..127) ,
            recovery          CHOICE
            {
                rej           NULL,
                sREJ          NULL,
                mSREJ         NULL,
                ...
            },
            ...
        },
        uNERM                NULL,
        ...
    },
    v75Parameters            V75Parameters,
    ...
}

```



```

V76HDLParameters ::=SEQUENCE
{
    crcLength          CRCLength,
    n401               INTEGER (1..4095),
    loopbackTestProcedure
    ...
}

CRCLength ::=CHOICE
{
    crc8bit           NULL,
    crc16bit          NULL,
    crc32bit          NULL,
    ...
}

H2250LogicalChannelParameters ::=SEQUENCE
{
    nonStandard        SEQUENCE OF NonStandardParameter OPTIONAL,
    sessionID          INTEGER (0..255),
    associatedSessionID
    mediaChannel        TransportAddress OPTIONAL,
    mediaGuaranteedDelivery
    mediaControlChannel TransportAddress OPTIONAL, -- обратный
    ...                -- канал RTCP
    mediaControlGuaranteedDelivery
    silenceSuppression
    destination        BOOLEAN OPTIONAL,
    ...                BOOLEAN OPTIONAL,
    ...                TerminalLabel OPTIONAL,
    dynamicRTPPayloadType
    mediaPacketization
    {
        h261aVideoPacketization
        ...
        rtpPayloadType
    } OPTIONAL,
    ...
    transportCapability
    redundancyEncoding
    source             TransportCapability OPTIONAL,
    ...                RedundancyEncoding OPTIONAL,
    ...                TerminalLabel OPTIONAL
}

RTPPayloadType ::= SEQUENCE
{
    payloadDescriptor CHOICE
    {
        nonStandardIdentifier
        rfc-number      NonStandardParameter,
        oid             INTEGER (1..32768, ...),
        ...            OBJECT IDENTIFIER,
    },
    payloadType        INTEGER (0..127) OPTIONAL,
    ...
}

RedundancyEncoding ::=SEQUENCE
{
    redundancyEncodingMethod
    secondaryEncoding  RedundancyEncodingMethod,
    ...               DataType OPTIONAL, -- ЗАВИСИТ ОТ МЕТОДА
    ...
}

```

-- Приведенная ниже последовательность может использоваться вместо приведенного
 -- выше поля secondaryEncoding

```

rtpRedundancyEncoding          SEQUENCE
{
    primary                    RedundancyEncodingElement OPTIONAL,
                                -- Присутствует, если redundancyEncoding
                                -- выбрано в качестве dataType
                                -- в OpenLogicalChannel или как
                                -- часть MultiplePayloadStream
    secondary                  SEQUENCE OF RedundancyEncodingElement OPTIONAL,
    ...
} OPTIONAL
}

RedundancyEncodingElement      ::=SEQUENCE
{
    dataType                   DataType,
    payloadType                INTEGER(0..127) OPTIONAL,
    ...
}

MultiplePayloadStream          ::=SEQUENCE
{
    elements                   SEQUENCE OF MultiplePayloadStreamElement,
    ...
}

MultiplePayloadStreamElement   ::=SEQUENCE
{
    dataType                   DataType,
    payloadType                INTEGER(0..127) OPTIONAL,
    ...
}

FECData                        ::=CHOICE
{
    rfc2733                    SEQUENCE
    {
        mode                   CHOICE
        {
            redundancyEncoding  NULL,
            separateStream      CHOICE
            {
                differentPort    SEQUENCE
                {
                    protectedSessionID  INTEGER(1..255),
                    protectedPayloadType  INTEGER(0..127) OPTIONAL,
                    ...
                },
                samePort          SEQUENCE
                {
                    protectedPayloadType  INTEGER(0..127),
                    ...
                },
                ...
            },
            ...
        },
        ...
    }
}

```

```

TransportAddress ::=CHOICE
{
    unicastAddress UnicastAddress,
    multicastAddress MulticastAddress,
    ...
}

UnicastAddress ::=CHOICE
{
    iPAddress SEQUENCE
    {
        network OCTET STRING (SIZE(4)),
        tsapIdentifier INTEGER(0..65535),
        ...
    },
    iPXAddress SEQUENCE
    {
        node OCTET STRING (SIZE(6)),
        netnum OCTET STRING (SIZE(4)),
        tsapIdentifier OCTET STRING (SIZE(2)),
        ...
    },
    iP6Address SEQUENCE
    {
        network OCTET STRING (SIZE(16)),
        tsapIdentifier INTEGER(0..65535),
        ...
    },
    netBios OCTET STRING (SIZE(16)),
    iPSourceRouteAddress SEQUENCE
    {
        routing CHOICE
        {
            strict NULL,
            loose NULL
        },
        network OCTET STRING (SIZE(4)),
        tsapIdentifier INTEGER(0..65535),
        route SEQUENCE OF OCTET STRING (SIZE(4)),
        ...
    },
    ...,
    nsap OCTET STRING (SIZE(1..20)),
    nonStandardAddress NonStandardParameter
}

MulticastAddress ::=CHOICE
{
    iPAddress SEQUENCE
    {
        network OCTET STRING (SIZE(4)),
        tsapIdentifier INTEGER(0..65535),
        ...
    },
    iP6Address SEQUENCE
    {
        network OCTET STRING (SIZE(16)),
        tsapIdentifier INTEGER(0..65535),
        ...
    },
    ...,
    nsap OCTET STRING (SIZE(1..20)),
    nonStandardAddress NonStandardParameter
}

```

```

EncryptionSync ::=SEQUENCE
-- используется для поддержки нового ключа
-- и точки синхронизации
{
    nonStandardParameter OPTIONAL,
    synchFlag INTEGER(0..255) , -- для Н.324 может
-- потребоваться
-- БОЛЬШОЙ диапазон
-- и т. д.
-- должен быть Dynamic
-- Payload# для Н.323
    h235Key OCTET STRING (SIZE(1..65535)),
-- кодированное
-- значение Н.235
    escrowentry SEQUENCE SIZE(1..256) OF EscrowData OPTIONAL,
    ...
}

EscrowData ::=SEQUENCE
{
    escrowID OBJECT IDENTIFIER,
    escrowValue BIT STRING (SIZE(1..65535)),
    ...
}

OpenLogicalChannelAck ::=SEQUENCE
{
    forwardLogicalChannelNumber LogicalChannelNumber,

    reverseLogicalChannelParameters SEQUENCE
    {
        reverseLogicalChannelNumber LogicalChannelNumber,
        portNumber INTEGER (0..65535) OPTIONAL,
        multiplexParameters CHOICE
        {
            h222LogicalChannelParameters H222LogicalChannelParameters,
-- параметры Н.223 никогда не присутствуют в обратном направлении
            ...,
            h2250LogicalChannelParameters H2250LogicalChannelParameters
        }
    } OPTIONAL, -- не присутствует для Н.223
    ...,
    replacementFor LogicalChannelNumber OPTIONAL
} OPTIONAL, -- не присутствует для запроса
-- однонаправленного канала
...,
separateStack NetworkAccessParameters OPTIONAL,
-- для запросчика Open при организации
-- стека
forwardMultiplexAckParameters CHOICE
{
-- параметры Н.222 никогда не присутствуют в Ask
-- параметры Н.223 никогда не присутствуют в Ask
-- параметры V.76 никогда не присутствуют в Ask
    h2250LogicalChannelAckParameters H2250LogicalChannelAckParameters,
    ...
} OPTIONAL,
encryptionSync EncryptionSync OPTIONAL -- используется только
-- ведущим терминалом
}

OpenLogicalChannelReject ::=SEQUENCE
{
    forwardLogicalChannelNumber LogicalChannelNumber,
    cause CHOICE
    {
        unspecified NULL,

```

```

        unsuitableReverseParameters  NULL,
        dataTypeNotSupported          NULL,
        dataTypeNotAvailable          NULL,
        unknownDataType               NULL,
        dataTypeALCombinationNotSupported  NULL,
        ...,
        multicastChannelNotAllowed     NULL,
        insufficientBandwidth          NULL,
        separateStackEstablishmentFailed  NULL,
        invalidSessionID               NULL,
        masterSlaveConflict            NULL,
        waitForCommunicationMode       NULL,
        invalidDependentChannel        NULL,
        replacementForRejected         NULL
    },
    ...
}

OpenLogicalChannelConfirm ::=SEQUENCE
{
    forwardLogicalChannelNumber LogicalChannelNumber,
    ...
}

H2250LogicalChannelAckParameters ::=SEQUENCE
{
    nonStandard SEQUENCE OF NonStandardParameter OPTIONAL,
    sessionID INTEGER(1..255) OPTIONAL,
    mediaChannel TransportAddress OPTIONAL,
    mediaControlChannel TransportAddress OPTIONAL, -- канал прямой
                                                    -- передачи RTSP
    dynamicRTTPayloadType INTEGER(96..127) OPTIONAL, -- используется
                                                    -- только ведущим
                                                    -- терминалом
                                                    -- или MC
    ...,
    flowControlToZero BOOLEAN,
    portNumber INTEGER (0..65535) OPTIONAL
}

CloseLogicalChannel ::=SEQUENCE
{
    forwardLogicalChannelNumber LogicalChannelNumber,
    source CHOICE
    {
        user NULL,
        lcse NULL
    },
    ...,
    reason CHOICE
    {
        unknown NULL,
        reopen NULL,
        reservationFailure NULL,
        ...
    }
}

CloseLogicalChannelAck ::=SEQUENCE
{
    forwardLogicalChannelNumber LogicalChannelNumber,
    ...
}

```

```

RequestChannelClose                               ::=SEQUENCE
{
    forwardLogicalChannelNumber                   LogicalChannelNumber,
    ...,
    qosCapability                                 QOSCapability OPTIONAL,
    reason                                         CHOICE
    {
        unknown                                   NULL,
        normal                                    NULL,
        reopen                                    NULL,
        reservationFailure                        NULL,
        ...
    }
}

RequestChannelCloseAck                             ::=SEQUENCE
{
    forwardLogicalChannelNumber                   LogicalChannelNumber,
    ...
}

RequestChannelCloseReject                         ::=SEQUENCE
{
    forwardLogicalChannelNumber                   LogicalChannelNumber,
    cause                                         CHOICE
    {
        unspecified                               NULL,
        ...
    },
    ...
}

RequestChannelCloseRelease                         ::=SEQUENCE
{
    forwardLogicalChannelNumber                   LogicalChannelNumber,
    ...
}

-- =====
-- Определения таблицы мультиплексирования Н.223
-- =====

MultiplexEntrySend                                ::=SEQUENCE
{
    sequenceNumber                               SequenceNumber,
    multiplexEntryDescriptors                     SET SIZE (1..15) OF MultiplexEntryDescriptor,
    ...
}

MultiplexEntryDescriptor                          ::=SEQUENCE
{
    multiplexTableEntryNumber                     MultiplexTableEntryNumber,
    elementList                                   SEQUENCE SIZE (1..256) OF MultiplexElement OPTIONAL
}

MultiplexElement                                  ::=SEQUENCE
{
    type                                           CHOICE
    {
        logicalChannelNumber                       INTEGER(0..65535),
        subElementList                             SEQUENCE SIZE (2..255) OF MultiplexElement
    },
    repeatCount                                    CHOICE

```

```

        {
            finite                INTEGER (1..65535),  -- повторение типа
            untilClosingFlag      NULL                -- используется для
                                                -- последнего элемента
        }
    }

MultiplexTableEntryNumber ::=INTEGER (1..15)

MultiplexEntrySendAck ::=SEQUENCE
{
    sequenceNumber              SequenceNumber,
    multiplexTableEntryNumber    SET SIZE (1..15) OF
                                MultiplexTableEntryNumber,
    ...
}

MultiplexEntrySendReject ::=SEQUENCE
{
    sequenceNumber              SequenceNumber,
    rejectionDescriptions       SET SIZE (1..15) OF
                                MultiplexEntryRejectionDescriptions,
    ...
}

MultiplexEntryRejectionDescriptions ::=SEQUENCE
{
    multiplexTableEntryNumber    MultiplexTableEntryNumber,
    cause                        CHOICE
    {
        unspecifiedCause         NULL,
        descriptorTooComplex     NULL,
        ...
    },
    ...
}

MultiplexEntrySendRelease ::=SEQUENCE
{
    multiplexTableEntryNumber    SET SIZE (1..15) OF
                                MultiplexTableEntryNumber,
    ...
}

RequestMultiplexEntry ::=SEQUENCE
{
    entryNumbers                 SET SIZE (1..15) OF
                                MultiplexTableEntryNumber,
    ...
}

RequestMultiplexEntryAck ::=SEQUENCE
{
    entryNumbers                 SET SIZE (1..15) OF
                                MultiplexTableEntryNumber,
    ...
}

RequestMultiplexEntryReject ::=SEQUENCE
{
    entryNumbers                 SET SIZE (1..15) OF
                                MultiplexTableEntryNumber,
    rejectionDescriptions        SET SIZE (1..15) OF
                                RequestMultiplexEntryRejectionDescriptions,
    ...
}

```

```

}

RequestMultiplexEntryRejectionDescriptions ::=SEQUENCE
{
    multiplexTableEntryNumber MultiplexTableEntryNumber,
    cause CHOICE
    {
        unspecifiedCause NULL,
        ...
    },
    ...
}

RequestMultiplexEntryRelease ::=SEQUENCE
{
    entryNumbers SET SIZE (1..15) OF
    MultiplexTableEntryNumber,
    ...
}

-- =====
-- Определения режима запроса
-- =====

-- Режим запроса - это перечень режимов (в порядке предпочтения), которые
-- терминал хотел бы передавать.

RequestMode ::=SEQUENCE
{
    sequenceNumber SequenceNumber,
    requestedModes SEQUENCE SIZE (1..256) OF ModeDescription,
    ...
}

RequestModeAck ::=SEQUENCE
{
    sequenceNumber SequenceNumber,
    response CHOICE
    {
        willTransmitMostPreferredMode NULL,
        willTransmitLessPreferredMode NULL,
        ...
    },
    ...
}

RequestModeReject ::=SEQUENCE
{
    sequenceNumber SequenceNumber,
    cause CHOICE
    {
        modeUnavailable NULL,
        multipointConstraint NULL,
        requestDenied NULL,
        ...
    },
    ...
}

RequestModeRelease ::=SEQUENCE
{
    ...
}

```



```

-- =====
-- Определения режима запроса: Описание режима
-- =====

```

```

ModeDescription ::= SET SIZE (1..256) OF ModeElement

ModeElementType ::= CHOICE
{
    nonStandard          NonStandardParameter,
    videoMode           VideoMode,
    audioMode           AudioMode,
    dataMode            DataMode,
    encryptionMode     EncryptionMode,
    ...,
    h235Mode            H235Mode,
    multiplexedStreamMode MultiplexedStreamParameter,
    redundancyEncodingDTMode RedundancyEncodingDTMode,
    multiplePayloadStreamMode MultiplePayloadStreamMode,
    fecMode             FECMode
}

ModeElement ::= SEQUENCE
{
    type                ModeElementType,

    h223ModeParameters H223ModeParameters OPTIONAL,
    ...,
    v76ModeParameters  V76ModeParameters OPTIONAL,
    h2250ModeParameters H2250ModeParameters OPTIONAL,
    genericModeParameters GenericCapability OPTIONAL,
    multiplexedStreamModeParameters MultiplexedStreamModeParameters OPTIONAL,
    logicalChannelNumber LogicalChannelNumber OPTIONAL
}

H235Mode ::= SEQUENCE
{
    encryptionAuthenticationAndIntegrity EncryptionAuthenticationAndIntegrity,

    mediaMode          CHOICE
    {
        nonStandard    NonStandardParameter,
        videoMode      VideoMode,
        audioMode      AudioMode,
        dataMode       DataMode,
        ...
    },
    ...
}

MultiplexedStreamModeParameters ::= SEQUENCE
{
    logicalChannelNumber LogicalChannelNumber,
    ...
}

RedundancyEncodingDTMode ::= SEQUENCE
{
    redundancyEncodingMethod RedundancyEncodingMethod,
    primary                  RedundancyEncodingDTModeElement,
    secondary                SEQUENCE OF RedundancyEncodingDTModeElement,
    ...
}

```

```

RedundancyEncodingDTModeElement ::=SEQUENCE
{
    type CHOICE
    {
        nonStandard NonStandardParameter,
        videoMode VideoMode,
        audioMode AudioMode,
        dataMode DataMode,
        encryptionMode EncryptionMode,
        h235Mode H235Mode,
        ...
    },
    ...
}

MultiplePayloadStreamMode ::=SEQUENCE
{
    elements SEQUENCE OF MultiplePayloadStreamElementMode,
    ...
}

MultiplePayloadStreamElementMode ::=SEQUENCE
{
    type ModeElementType,
    ...
}

FECMode ::=CHOICE
{
    rfc2733Mode SEQUENCE
    {
        mode CHOICE
        {
            redundancyEncoding NULL,
            separateStream CHOICE
            {
                differentPort SEQUENCE
                {
                    protectedSessionID INTEGER(1..255),
                    protectedPayloadType INTEGER(0..127) OPTIONAL,
                    ...
                },
                samePort SEQUENCE
                {
                    protectedType ModeElementType,
                    ...
                },
                ...
            },
            ...
        },
        ...
    },
    ...
}

H223ModeParameters ::=SEQUENCE
{
    adaptationLayerType CHOICE
    {
        nonStandard NonStandardParameter,
        allFramed NULL,
        allNotFramed NULL,
        al2WithoutSequenceNumbers NULL,

```

```

        al2WithSequenceNumbers      NULL,
        al3                          SEQUENCE
        {
            controlFieldOctets      INTEGER(0..2),
            sendBufferSize           INTEGER(0..16777215) -- единицы измерения - октеты
        },
        ...,
        al1M                          H223AL1MParameters,
        al2M                          H223AL2MParameters,
        al3M                          H223AL3MParameters
    },

    segmentableFlag                  BOOLEAN,
    ...
}

V76ModeParameters                  ::=CHOICE
{
    suspendResumewAddress           NULL,
    suspendResumewoAddress          NULL,
    ...
}

H2250ModeParameters                ::=SEQUENCE
{
    redundancyEncodingMode          RedundancyEncodingMode OPTIONAL,
    ...
}

RedundancyEncodingMode             ::=SEQUENCE
{
    redundancyEncodingMethod        RedundancyEncodingMethod,
    secondaryEncoding              CHOICE
    {
        nonStandard                 NonStandardParameter,
        audioData                   AudioMode,
        ...
    } OPTIONAL,
    ...
}

-- =====
-- Определения режима запроса: Видеорежимы
-- =====

VideoMode                           ::=CHOICE
{
    nonStandard                     NonStandardParameter,
    h261VideoMode                   H261VideoMode,
    h262VideoMode                   H262VideoMode,
    h263VideoMode                   H263VideoMode,
    is11172VideoMode               IS11172VideoMode,
    ...,
    genericVideoMode                GenericCapability
}

H261VideoMode                       ::=SEQUENCE
{
    resolution                       CHOICE
    {
        qcif                         NULL,
        cif                           NULL
    },

```

```

    bitRate                INTEGER (1..19200), -- единица измерения - 100 бит/с
    stillImageTransmission BOOLEAN,
    ...
}

H262VideoMode            ::=SEQUENCE
{
    profileAndLevel        CHOICE
    {
        profileAndLevel-SPatML    NULL,
        profileAndLevel-MPatLL    NULL,
        profileAndLevel-MPatML    NULL,
        profileAndLevel-MPath-14  NULL,
        profileAndLevel-MPathHL   NULL,
        profileAndLevel-SNRatLL   NULL,
        profileAndLevel-SNRatML   NULL,
        profileAndLevel-SpatialH-14 NULL,
        profileAndLevel-HPatML    NULL,
        profileAndLevel-HPath-14  NULL,
        profileAndLevel-HPatHL    NULL,
        ...
    },
    videoBitRate            INTEGER(0..1073741823) OPTIONAL, -- единица измерения - 400 бит/с
    vbvBufferSize          INTEGER(0..262143) OPTIONAL, -- единица измерения 16384 бита
    samplesPerLine         INTEGER(0..16383) OPTIONAL, -- единица измерения отсчет/строка
    linesPerFrame          INTEGER(0..16383) OPTIONAL, -- единица измерения строка/кадр
    framesPerSecond        INTEGER(0..15) OPTIONAL, -- код частоты кадров
    luminanceSampleRate    INTEGER(0..4294967295) OPTIONAL, -- единица измерения отсчет/с
    ...
}

H263VideoMode            ::=SEQUENCE
{
    resolution              CHOICE
    {
        sqcif                NULL,
        qcif                  NULL,
        cif                    NULL,
        cif4                   NULL,
        cif16                  NULL,
        ...,
        custom                 NULL
    },
    bitRate                 INTEGER (1..19200), -- единица измерения - 100 бит/с
    unrestrictedVector      BOOLEAN,
    arithmeticCoding        BOOLEAN,
    advancedPrediction      BOOLEAN,
    pbFrames                 BOOLEAN,
    ...,
    errorCompensation        BOOLEAN,
    enhancementLayerInfo    EnhancementLayerInfo OPTIONAL,
    h263Options              H263Options OPTIONAL
}

IS11172VideoMode        ::=SEQUENCE
{
    constrainedBitstream    BOOLEAN,
    videoBitRate            INTEGER(0..1073741823) OPTIONAL, -- единица
                                                                -- измерения
                                                                -- 400 бит/с
    vbvBufferSize           INTEGER(0..262143) OPTIONAL, -- единица
                                                                -- измерения
                                                                -- 16384 бита
    samplesPerLine          INTEGER(0..16383) OPTIONAL, -- единица
                                                                -- измерения
                                                                -- отсчет/строка
}

```

```

linesPerFrame                INTEGER(0..16383) OPTIONAL,    -- единица
                                -- измерения
                                -- строка/кадр
pictureRate                   INTEGER(0..15) OPTIONAL,
luminanceSampleRate          INTEGER(0..4294967295) OPTIONAL, -- единица
                                -- измерения
                                -- отсчет/с
...
}

```

```

-- =====
-- Определения режима запроса: Аудиорежимы
-- =====

```

```

AudioMode                    ::=CHOICE
{
    nonStandard                NonStandardParameter,
    g711Alaw64k                NULL,
    g711Alaw56k                NULL,
    g711Ulaw64k                NULL,
    g711Ulaw56k                NULL,

    g722-64k                   NULL,
    g722-56k                   NULL,
    g722-48k                   NULL,

    g728                       NULL,
    g729                       NULL,
    g729AnnexA                 NULL,

    g7231                       CHOICE
    {
        noSilenceSuppressionLowRate  NULL,
        noSilenceSuppressionHighRate NULL,
        silenceSuppressionLowRate    NULL,
        silenceSuppressionHighRate   NULL
    },

    is11172AudioMode           IS11172AudioMode,
    is13818AudioMode           IS13818AudioMode,

    ...,
    g729wAnnexB                INTEGER(1..256),
    g729AnnexAwAnnexB          INTEGER(1..256),
    g7231AnnexCMode            G7231AnnexCMode,
    gsmFullRate                 GSMAudioCapability,
    gsmHalfRate                 GSMAudioCapability,
    gsmEnhancedFullRate        GSMAudioCapability,
    genericAudioMode            GenericCapability,
    g729Extensions              G729Extensions,
    vbd                          VBDMode
}

```

```

IS11172AudioMode             ::=SEQUENCE
{
    audioLayer                  CHOICE
    {
        audioLayer1            NULL,
        audioLayer2            NULL,
        audioLayer3            NULL
    },

    audioSampling               CHOICE
    {
        audioSampling32k       NULL,
        audioSampling44k1      NULL,

```

```

        audioSampling48k                NULL
    },
    multichannelType                    CHOICE
    {
        singleChannel                   NULL,
        twoChannelStereo                NULL,
        twoChannelDual                  NULL
    },
    bitRate                             INTEGER (1..448), -- единица измерения - кбит/с
    ...
}

IS13818AudioMode                      ::= SEQUENCE
{
    audioLayer                          CHOICE
    {
        audioLayer1                    NULL,
        audioLayer2                    NULL,
        audioLayer3                    NULL
    },
    audioSampling                       CHOICE
    {
        audioSampling16k               NULL,
        audioSampling22k05             NULL,
        audioSampling24k               NULL,
        audioSampling32k               NULL,
        audioSampling44k1              NULL,
        audioSampling48k               NULL
    },
    multichannelType                    CHOICE
    {
        singleChannel                  NULL,
        twoChannelStereo               NULL,
        twoChannelDual                 NULL,
        threeChannels2-1               NULL,
        threeChannels3-0               NULL,
        fourChannels2-0-2-0            NULL,
        fourChannels2-2                 NULL,
        fourChannels3-1                 NULL,
        fiveChannels3-0-2-0            NULL,
        fiveChannels3-2                 NULL
    },
    lowFrequencyEnhancement            BOOLEAN,
    multilingual                        BOOLEAN,
    bitRate                             INTEGER (1..1130), -- единица измерения - кбит/с
    ...
}

G7231AnnexCMode                      ::= SEQUENCE
{
    maxAl-sduAudioFrames               INTEGER (1..256),
    silenceSuppression                  BOOLEAN,
    g723AnnexCAudioMode                SEQUENCE
    {
        highRateMode0                  INTEGER (27..78), -- единица измерения - октеты
        highRateMode1                  INTEGER (27..78), -- единица измерения - октеты
        lowRateMode0                   INTEGER (23..66), -- единица измерения - октеты
    }
}

```

```

        lowRateModel      INTEGER (23..66), -- единица измерения - октеты
        sidMode0          INTEGER (6..17),  -- единица измерения - октеты
        sidModel          INTEGER (6..17),  -- единица измерения - октеты
        ...
    },
    ...
}

```

```

VBDMoDe                ::=SEQUENCE
{
    type                 AudioMoDe,      -- не должен быть равен "vbd"
    ...
}

```

```

-- =====
-- Определения режима запроса: Режимы передачи данных
-- =====

```

```

DataMoDe                ::=SEQUENCE
{
    application          CHOICE
    {
        nonStandard     NonStandardParameter,
        t120             DataProtocolCapability,
        dsm-cc           DataProtocolCapability,
        userData         DataProtocolCapability,
        t84              DataProtocolCapability,
        t434             DataProtocolCapability,
        h224             DataProtocolCapability,
        nlpid            SEQUENCE
        {
            nlpidProtocol DataProtocolCapability,
            nlpidData      OCTET STRING
        },
        dsvdControl      NULL,
        h222DataPartitioning DataProtocolCapability,
        ...,
        t30fax           DataProtocolCapability,
        t140             DataProtocolCapability,
        t38fax           SEQUENCE
        {
            t38FaxProtocol DataProtocolCapability,
            t38FaxProfile   T38FaxProfile
        },
        genericDataMode  GenericCapability
    },
    bitRate              INTEGER (0..4294967295), -- единица измерения
                                                -- - 100 бит/с
    ...
}

```

```

-- =====
-- Определения режима запроса: Режимы шифрования
-- =====

```

```

EncryptionMoDe         ::=CHOICE
{
    nonStandard          NonStandardParameter,
    h233Encryption       NULL,
    ...
}

```

-- =====
-- *Определения задержки из-за подтверждения сигнала*
-- =====

```
RoundTripDelayRequest ::=SEQUENCE
{
    sequenceNumber      SequenceNumber,
    ...
}
```

```
RoundTripDelayResponse ::=SEQUENCE
{
    sequenceNumber      SequenceNumber,
    ...
}
```

-- =====
-- *Определения цикла обслуживания*
-- =====

```
MaintenanceLoopRequest ::=SEQUENCE
{
    type                CHOICE
    {
        systemLoop      NULL,
        mediaLoop        LogicalChannelNumber,
        logicalChannelLoop LogicalChannelNumber,
        ...
    },
    ...
}
```

```
MaintenanceLoopAck ::=SEQUENCE
{
    type                CHOICE
    {
        systemLoop      NULL,
        mediaLoop        LogicalChannelNumber,
        logicalChannelLoop LogicalChannelNumber,
        ...
    },
    ...
}
```

```
MaintenanceLoopReject ::=SEQUENCE
{
    type                CHOICE
    {
        systemLoop      NULL,
        mediaLoop        LogicalChannelNumber,
        logicalChannelLoop LogicalChannelNumber,
        ...
    },
    cause                CHOICE
    {
        canNotPerformLoop NULL,
        ...
    },
    ...
}
```



```

MaintenanceLoopOffCommand      ::=SEQUENCE
{
    ...
}

-- =====
-- Определения режима связи
-- =====

CommunicationModeCommand      ::=SEQUENCE
{
    communicationModeTable    SET SIZE(1..256) OF CommunicationModeTableEntry,
    ...
}

CommunicationModeRequest      ::=SEQUENCE
{
    ...
}

CommunicationModeResponse     ::=CHOICE
{
    communicationModeTable    SET SIZE(1..256) OF CommunicationModeTableEntry,
    ...
}

CommunicationModeTableEntry   ::=SEQUENCE
{
    nonStandard                SEQUENCE OF NonStandardParameter OPTIONAL,
    sessionID                  INTEGER(1..255),
    associatedSessionID        INTEGER(1..255) OPTIONAL,

    terminalLabel              TerminalLabel OPTIONAL, -- если
                                                                -- отсутствует,
                                                                -- то это
                                                                -- относится ко
                                                                -- всем участникам
                                                                -- конференции

    sessionDescription         BMPString (SIZE(1..128)) ,
                                                                -- основной ISO/IEC 10646-1 (Unicode)
    dataType                   CHOICE
    {
        videoData              VideoCapability,
        audioData              AudioCapability,
        data                    DataApplicationCapability,
        ...
    },
    mediaChannel                TransportAddress OPTIONAL,
    mediaGuaranteedDelivery     BOOLEAN OPTIONAL,
    mediaControlChannel        TransportAddress OPTIONAL,
                                                                -- обратный канал RTCP
    mediaControlGuaranteedDelivery BOOLEAN OPTIONAL,
    ...
    redundancyEncoding         RedundancyEncoding OPTIONAL,
    sessionDependency           INTEGER (1..255) OPTIONAL,
    destination                 TerminalLabel OPTIONAL
}

-- =====
-- Определения запроса конференции
-- =====

```

```

ConferenceRequest                               ::=CHOICE
{
    terminalListRequest                          NULL,          -- так же как H.230 TCU (term->MC)
    makeMeChair                                  NULL,          -- так же как H.230 CCA (term->MC)
    cancelMakeMeChair                            NULL,          -- так же как H.230 CIS (term->MC)
    dropTerminal                                 TerminalLabel, -- так же как H.230 CCD (term->MC)
    requestTerminalID                            TerminalLabel, -- так же как TCP (term->MC)
    enterH243Password                            NULL,          -- так же как H.230 TCS1 (MC->term)
    enterH243TerminalID                          NULL,          -- так же как H.230 TCS2/TCI
                                                    -- (MC->term)
    enterH243ConferenceID                        NULL,          -- так же как H.230 TCS3 (MC->term)
    ...,
    enterExtensionAddress                        NULL,          -- так же как H.230 TCS4 (GW->term)
    requestChairTokenOwner                       NULL,          -- так же как H.230 TCA (term->MC)
    requestTerminalCertificate                   SEQUENCE
    {
        terminalLabel                            TerminalLabel OPTIONAL,
        certSelectionCriteria                   CertSelectionCriteria OPTIONAL,
        sRandom                                  INTEGER (1..4294967295) OPTIONAL,
                                                    -- является выбором запросчика
        ...
    },
    broadcastMyLogicalChannel                    LogicalChannelNumber, -- аналогично H.230 MCV
    makeTerminalBroadcaster                      TerminalLabel,      -- аналогично H.230 VCB
    sendThisSource                              TerminalLabel,      -- аналогично H.230 VCS
    requestAllTerminalIDs                       NULL,
    remoteMCRequest                             RemoteMCRequest
}

CertSelectionCriteria                            ::=SEQUENCE SIZE (1..16) OF Criteria

Criteria                                         ::=SEQUENCE
{
    field                                         OBJECT IDENTIFIER, -- может включать
                                                    -- тип сертификата
    value                                         OCTET STRING (SIZE(1..65535)),
    ...
}

TerminalLabel                                   ::=SEQUENCE
{
    mcuNumber                                    McuNumber,
    terminalNumber                               TerminalNumber,
    ...
}

McuNumber                                        ::=INTEGER(0..192)
TerminalNumber                                  ::=INTEGER(0..192)

```

```

-- =====
-- Определения ответа на запрос конференции
-- =====

```

```

ConferenceResponse                               ::=CHOICE
{
    mCTerminalIDResponse                         SEQUENCE          -- ответ на TCP
                                                    -- (так же как TIP)
    {
        terminalLabel                            TerminalLabel,
        ...
    }
}

```

```

    terminalID          TerminalID,
    ...
},

terminalIDResponse    SEQUENCE          -- ответ на TCS2 или TCI
{                    -- так же как IIS
    terminalLabel      TerminalLabel,    -- (term->MC)
    terminalID         TerminalID,
    ...
},

conferenceIDResponse SEQUENCE          -- ответ на TCS3
{                    -- так же как IIS
    terminalLabel      TerminalLabel,    -- (term->MC)
    conferenceID      ConferenceID,
    ...
},

passwordResponse     SEQUENCE          -- ответ на TCS1
{                    -- так же как IIS
    terminalLabel      TerminalLabel,    -- (term->MC)
    password          Password,
    ...
},

terminalListResponse SET SIZE (1..256) OF TerminalLabel,

videoCommandReject   NULL,            -- так же как H.230 VCR
terminalDropReject   NULL,            -- так же как H.230 CIR

makeMeChairResponse  CHOICE          -- так же как H.230 CCR
{
    grantedChairToken NULL,            -- так же как H.230 CIT
    deniedChairToken  NULL,            -- так же как H.230 CCR
    ...
},
...,
extensionAddressResponse SEQUENCE    -- ответ на TCS4
{
    extensionAddress  TerminalID,      -- так же как IIS (term->GW)
    ...
},
chairTokenOwnerResponse SEQUENCE    -- ответ на TCA (так же как TIR)
-- передается только MC
{
    terminalLabel      TerminalLabel,
    terminalID         TerminalID,
    ...
},
terminalCertificateResponse SEQUENCE
{
    terminalLabel      TerminalLabel OPTIONAL,
    certificateResponse OCTET STRING (SIZE(1..65535)) OPTIONAL,
    ...
},
broadcastMyLogicalChannelResponse CHOICE
{
    grantedBroadcastMyLogicalChannel NULL,    -- аналогично H.230 MVA
    deniedBroadcastMyLogicalChannel NULL,    -- аналогично H.230 MVR
    ...
},
makeTerminalBroadcasterResponse CHOICE
{
    grantedMakeTerminalBroadcaster NULL,

```

```

        deniedMakeTerminalBroadcaster    NULL,
        ...
    },
    sendThisSourceResponse                CHOICE
    {
        grantedSendThisSource            NULL,
        deniedSendThisSource             NULL,
        ...
    },
    requestAllTerminalIDsResponse        RequestAllTerminalIDsResponse,
    remoteMCResponse                     RemoteMCResponse
}

```

```

TerminalID                               ::=OCTET STRING (SIZE(1..128)) - как в H.230
ConferenceID                             ::=OCTET STRING (SIZE(1..32))
Password                                 ::=OCTET STRING (SIZE(1..32))

```

```

RequestAllTerminalIDsResponse            ::=SEQUENCE
{
    terminalInformation                  SEQUENCE OF TerminalInformation,
    ...
}

```

```

TerminalInformation                      ::=SEQUENCE
{
    terminalLabel                       TerminalLabel,
    terminalID                           TerminalID,
    ...
}

```

```

-- =====
-- Определения удаленного запроса MC
-- =====

```

```

RemoteMCRequest                          ::=CHOICE
{
    masterActivate                       NULL,
    slaveActivate                        NULL,
    deactivate                            NULL,
    ...
}

```

```

RemoteMCResponse                          ::=CHOICE
{
    accept                                NULL,
    reject                                CHOICE
    {
        unspecified                       NULL,
        functionNotSupported              NULL,
        ...
    },
    ...
}

```

```

-- =====
-- Определения многоточечного соединения
-- =====

```

```

MultilinkRequest                          ::=CHOICE
{
    nonStandard                          NonStandardMessage,

    callInformation                      SEQUENCE
    {

```

```

        maxNumberOfAdditionalConnections    INTEGER (1..65535),
        ...
    },
    addConnection                          SEQUENCE
    {
        sequenceNumber                     SequenceNumber, -- уникальный
                                                -- идентификатор запроса
        dialingInformation                  DialingInformation,
        ...
    },
    removeConnection                       SEQUENCE
    {
        connectionIdentifier                ConnectionIdentifier,
        ...
    },
    maximumHeaderInterval                  SEQUENCE
    {
        requestType                        CHOICE
        {
            currentIntervalInformation      NULL,
            requestedInterval                INTEGER (0..65535), -- максимальный
                                                                    -- интервал
                                                                    -- заголовка,
                                                                    -- миллисекунды
            ...
        },
        ...
    },
    ...
}

MultilinkResponse ::= CHOICE
{
    nonStandard                            NonStandardMessage,

    callInformation                        SEQUENCE
    {
        dialingInformation                  DialingInformation,
        callAssociationNumber                INTEGER (0..4294967295),
        ...
    },

    addConnection                          SEQUENCE
    {
        sequenceNumber                     SequenceNumber, -- Равен значению в запросе
        responseCode                        CHOICE
        {
            accepted                        NULL,
            rejected                         CHOICE
            {
                connectionsNotAvailable     NULL, -- по любой технической причине
                userRejected                 NULL,
                ...
            },
            ...
        },
        ...
    },
    ...
}

```

```

removeConnection          SEQUENCE
{
  connectionIdentifier    ConnectionIdentifier,
  ...,
},

maximumHeaderInterval    SEQUENCE
{
  currentInterval         INTEGER (0..65535), -- максимальный
  ...,                   -- интервал
  ...,                   -- заголовка,
  ...,                   -- миллисекунды
},
...
}

MultilinkIndication      ::= CHOICE
{
  nonStandard             NonStandardMessage,
  crcDesired              SEQUENCE
  {
    ...,
  },
  excessiveError          SEQUENCE
  {
    connectionIdentifier  ConnectionIdentifier,
    ...,
  },
  ...
}

DialingInformation       ::= CHOICE
{
  nonStandard             NonStandardMessage,
  differential            SET SIZE (1..65535) OF DialingInformationNumber,
  ...,                   -- перечень чисел для всех дополнительных
  ...,                   -- каналов; только младшие разряды,
  ...,                   -- отличающиеся от номера начального
  ...,                   -- канала
  infoNotAvailable        INTEGER (1..65535), -- максимальное
  ...,                   -- количество каналов
  ...,                   -- дополнительных
  ...,                   -- каналов
},

DialingInformationNumber ::= SEQUENCE
{
  networkAddress          NumericString (SIZE (0..40)),
  subAddress              IA5String (SIZE (1..40)) OPTIONAL,
  networkType             SET SIZE (1..255) OF DialingInformationNetworkType,
  ...
}

DialingInformationNetworkType ::= CHOICE
{
  nonStandard             NonStandardMessage,
  n-isdn                  NULL,
  gstn                    NULL,
  ...,
  mobile                  NULL
}

```

```

ConnectionIdentifier ::= SEQUENCE
{
    channelTag          INTEGER (0..4294967295), -- из H.MULTILINK
    sequenceNumber     INTEGER (0..4294967295), -- из H.MULTILINK
    ...
}

-- =====
-- Определения изменения скорости передачи логического канала
-- =====

MaximumBitRate ::= INTEGER (0.. 4294967295) -- единица измерения
-- - 100 бит/с

LogicalChannelRateRequest ::= SEQUENCE
{
    sequenceNumber      SequenceNumber,
    logicalChannelNumber LogicalChannelNumber,
    maximumBitRate     MaximumBitRate,
    ...
}

LogicalChannelRateAcknowledge ::= SEQUENCE
{
    sequenceNumber      SequenceNumber,
    logicalChannelNumber LogicalChannelNumber,
    maximumBitRate     MaximumBitRate,
    ...
}

LogicalChannelRateReject ::= SEQUENCE
{
    sequenceNumber      SequenceNumber,
    logicalChannelNumber LogicalChannelNumber,
    rejectReason       LogicalChannelRateRejectReason,
    currentMaximumBitRate MaximumBitRate OPTIONAL,
    ...
}

LogicalChannelRateRejectReason ::= CHOICE
{
    undefinedReason    NULL,
    insufficientResources NULL,
    ...
}

LogicalChannelRateRelease ::= SEQUENCE
{
    ...
}

-- =====
-- Определения сообщения Command Message
-- =====

-- =====
-- Сообщения Command Message: Послать набор возможностей терминала
-- =====

SendTerminalCapabilitySet ::= CHOICE
{

```

```

specificRequest          SEQUENCE
{
    multiplexCapability    BOOLEAN,

    capabilityTableEntryNumbers SET SIZE (1..65535) OF
                                CapabilityTableEntryNumber OPTIONAL,

    capabilityDescriptorNumbers SET SIZE (1..256) OF
                                CapabilityDescriptorNumber OPTIONAL,
    ...
},
genericRequest          NULL,
...
}

```

```

-- =====
-- Сообщение Command Message: Шифрование
-- =====

```

```

EncryptionCommand      ::=CHOICE
{
    encryptionSE        OCTET STRING,          -- как в H.233, но
                                                -- без защиты от
                                                -- ошибок
                                                -- требует новый IV

    encryptionIVRequest NULL,
    encryptionAlgorithmID SEQUENCE
    {
        h233AlgorithmIdentifier SequenceNumber,
        associatedAlgorithm      NonStandardParameter
    },
    ...
}

```

```

-- =====
-- Сообщение Command Message: Контроль потока
-- =====

```

```

FlowControlCommand     ::=SEQUENCE
{
    scope CHOICE
    {
        logicalChannelNumber LogicalChannelNumber,
        resourceID            INTEGER (0..65535),
        wholeMultiplex        NULL
    },
    restriction          CHOICE
    {
        maximumBitRate       INTEGER (0..16777215), -- единица измерения
                                                                    -- - 100 бит/с
        noRestriction         NULL
    },
    ...
}

```

```

-- =====
-- Сообщение Command Message: Изменение или завершение сеанса
-- =====

```

```

EndSessionCommand     ::=CHOICE
{
    nonStandard          NonStandardParameter,

    disconnect          NULL,

    gstnOptions         CHOICE
    {

```



```

        telephonyMode          NULL,
        v8bis                   NULL,
        v34DSVD                 NULL,
        v34DuplexFAX           NULL,
        v34H324                 NULL,
        ...
    },
    ...,
    isdnOptions                 CHOICE
    {
        telephonyMode          NULL,
        v140                   NULL,
        terminalOnHold          NULL,
        ...
    }
}

-- =====
-- Сообщение Command Message: Команды проведения конференции
-- =====

ConferenceCommand             ::=CHOICE
{
    broadcastMyLogicalChannel  LogicalChannelNumber, -- аналогично H.230 MCV
    cancelBroadcastMyLogicalChannel LogicalChannelNumber, -- аналогично
                                                                    -- H.230 Cancel-MCV

    makeTerminalBroadcaster    TerminalLabel,          -- аналогично H.230 VCB
    cancelMakeTerminalBroadcaster NULL,                -- аналогично H.230
                                                                    -- Cancel-VCB

    sendThisSource             TerminalLabel,          -- аналогично H.230 VCS
    cancelSendThisSource       NULL,                  -- аналогично H.230
                                                                    -- cancel VCS

    dropConference             NULL,                    -- аналогично H.230 CCK
    ...,
    substituteConferenceIDCommand SubstituteConferenceIDCommand
}

SubstituteConferenceIDCommand ::=SEQUENCE
{
    conferenceIdentifier       OCTET STRING (SIZE(16)),
    ...
}

-- =====
-- Сообщение Command Message: Прочие команды типа H.230
-- =====

EncryptionUpdateDirection    ::= CHOICE
{
    masterToSlave             NULL,
    slaveToMaster             NULL,
    ...
}

MiscellaneousCommand          ::=SEQUENCE
{
    logicalChannelNumber      LogicalChannelNumber,
    type                       CHOICE
    {

```

```

equaliseDelay          NULL,          -- так же как H.230 ACE
zeroDelay              NULL,          -- так же как H.230 ACZ
multipointModeCommand NULL,
cancelMultipointModeCommand NULL,
videoFreezePicture    NULL,
videoFastUpdatePicture NULL,

videoFastUpdateGOB    SEQUENCE
{
    firstGOB           INTEGER (0..17),
    numberOfGOBs       INTEGER (1..18)
},

videoTemporalSpatialTradeOff  INTEGER (0..31), -- определяет значение обмена

videoSendSyncEveryGOB  NULL,
videoSendSyncEveryGOBCancel  NULL,

...,
videoFastUpdateMB      SEQUENCE
{
    firstGOB           INTEGER (0..255) OPTIONAL,
    firstMB            INTEGER (1..8192) OPTIONAL,
    numberOfMBs        INTEGER (1..8192),
    ...
},
maxH223MUXPDUsize     INTEGER(1..65535), -- единица измерения - октеты
encryptionUpdate       EncryptionSync,
encryptionUpdateRequest EncryptionUpdateRequest,
switchReceiveMediaOff  NULL,
switchReceiveMediaOn   NULL,

progressiveRefinementStart SEQUENCE
{
    repeatCount        CHOICE
    {
        doOneProgression          NULL,
        doContinuousProgressions  NULL,
        doOneIndependentProgression NULL,
        doContinuousIndependentProgressions NULL,
        ...
    },
    ...
},
progressiveRefinementAbortOne          NULL,
progressiveRefinementAbortContinuous   NULL,

videoBadMBs          SEQUENCE
{
    firstMB           INTEGER (1..9216),
    numberOfMBs       INTEGER (1..9216),
    temporalReference INTEGER (0..1023),
    ...
},
lostPicture          SEQUENCE OF PictureReference,
lostPartialPicture   SEQUENCE
{
    pictureReference  PictureReference,
    firstMB           INTEGER (1..9216),
    numberOfMBs       INTEGER (1..9216),
    ...
},
recoveryReferencePicture SEQUENCE OF PictureReference,
encryptionUpdateCommand SEQUENCE -- для подтвержденного обновления ключа в H.235V3

```

```

    {
        encryptionSync          EncryptionSync,
        multiplePayloadStream   MultiplePayloadStream OPTIONAL,
        ...
    },
    encryptionUpdateAck        SEQUENCE
    {
        synchFlag              INTEGER (0..255),
        ...
    }
},
...
direction                    EncryptionUpdateDirection OPTIONAL
}

KeyProtectionMethod          ::=SEQUENCE -- определяет способ защиты
                                -- нового ключа
{
    secureChannel              BOOLEAN,
    sharedSecret               BOOLEAN,
    certProtectedKey           BOOLEAN,
    ...
}

EncryptionUpdateRequest      ::=SEQUENCE
{
    keyProtectionMethod        KeyProtectionMethod OPTIONAL,
    ...,
    synchFlag                  INTEGER (0..255) OPTIONAL
}

PictureReference             ::=CHOICE
{
    pictureNumber              INTEGER (0..1023),
    longTermPictureIndex       INTEGER (0..255),
    ...
}

-----
-- Сообщение Command Message: Реконфигурация мультиплексирования H.223
-- =====

H223MultiplexReconfiguration ::=CHOICE
{
    h223ModeChange            CHOICE
    {
        toLevel0              NULL,
        toLevel1              NULL,
        toLevel2              NULL,
        toLevel2withOptionalHeader NULL,
        ...
    },

    h223AnnexADoubleFlag      CHOICE
    {
        start                  NULL,
        stop                   NULL,
        ...
    },

    ...
}

```

```

-- =====
-- Сообщение Command Message: Команда нового виртуального канала ATM
-- =====

```

```

NewATMVCCCommand ::=SEQUENCE
{
    resourceID          INTEGER(0..65535),
    bitRate             INTEGER(1..65535), -- единица измерения - 64 кбит/с
    bitRateLockedToPCRClock  BOOLEAN,
    bitRateLockedToNetworkClock  BOOLEAN,
    aal                CHOICE
    {
        aal1          SEQUENCE
        {
            clockRecovery  CHOICE
            {
                nullClockRecovery  NULL,
                srtsClockRecovery  NULL,
                adaptiveClockRecovery  NULL,
                ...
            },
            errorCorrection  CHOICE
            {
                nullErrorCorrection  NULL,
                longInterleaver      NULL,
                shortInterleaver     NULL,
                errorCorrectionOnly  NULL,
                ...
            },
            structuredDataTransfer  BOOLEAN,
            partiallyFilledCells    BOOLEAN,
            ...
        },
        aal5          SEQUENCE
        {
            forwardMaximumSDUSize  INTEGER (0..65535), -- единица измерения - октеты
            backwardMaximumSDUSize  INTEGER (0..65535), -- единица измерения - октеты
            ...
        },
        ...
    },
    multiplex          CHOICE
    {
        noMultiplex      NULL,
        transportStream  NULL,
        programStream    NULL,
        ...
    },
    reverseParameters SEQUENCE
    {
        bitRate          INTEGER(1..65535), -- единица измерения - 64 кбит/с
        bitRateLockedToPCRClock  BOOLEAN,
        bitRateLockedToNetworkClock  BOOLEAN,
        multiplex          CHOICE
        {
            noMultiplex      NULL,
            transportStream  NULL,
            programStream    NULL,
            ...
        },
        ...
    },
    ...
}

```

```

-- =====
-- Сообщение Command Message: Команда реконфигурации мобильной многоточечной
-- СВЯЗИ
-- =====

MobileMultilinkReconfigurationCommand ::=SEQUENCE
{
    sampleSize                INTEGER (1..255),
    samplesPerFrame           INTEGER (1..255),
    status                     CHOICE
    {
        synchronized          NULL,
        reconfiguration        NULL,
        ...
    },
    ...
}

-- =====
-- Определения сообщений индикации
-- =====

-- =====
-- Сообщение индикации: Функция не понята
-- =====

-- Данное сообщение используется для возврата запроса, ответа или команды,
-- которые не понята

FunctionNotUnderstood          ::=CHOICE
{
    request                   RequestMessage,
    response                   ResponseMessage,
    command                    CommandMessage
}

-- =====
-- Сообщение индикации: Функция не поддерживается
-- =====

-- Данное сообщение используется для возврата запроса, ответа или команды,
-- которые не распознаны

FunctionNotSupported          ::=SEQUENCE
{
    cause                     CHOICE
    {
        syntaxError           NULL,
        semanticError          NULL,
        unknownFunction        NULL,
        ...
    },
    returnedFunction           OCTET STRING OPTIONAL,
    ...
}

-- =====
-- Сообщение индикации: Конференция
-- =====

ConferenceIndication          ::=CHOICE
{
    sbeNumber                 INTEGER (0..9), -- так же как H.230 SBE Number
}

```

```

terminalNumberAssign      TerminalLabel, -- так же как H.230 TIA
terminalJoinedConference  TerminalLabel, -- так же как H.230 TIN
terminalLeftConference   TerminalLabel, -- так же как H.230 TID
seenByAtLeastOneOther    NULL,          -- так же как H.230 MIV
cancelSeenByAtLeastOneOther NULL,          -- так же как H.230
                                                                -- ОТМЕНИТЬ MIV

seenByAll                 NULL,          -- аналогично H.230 MIV
cancelSeenByAll           NULL,          -- аналогично H.230 MIV

terminalYouAreSeeing      TerminalLabel, -- так же как H.230 VIN

requestForFloor           NULL,          -- так же как H.230 TIF

...,
withdrawChairToken       NULL,          -- так же как H.230 CCR MC-> chair
floorRequested           TerminalLabel, -- так же как H.230 TIF MC-> chair
terminalYouAreSeeingInSubPictureNumber TerminalYouAreSeeingInSubPictureNumber,
videoIndicateCompose     VideoIndicateCompose
}

TerminalYouAreSeeingInSubPictureNumber ::= SEQUENCE
{
    terminalNumber      TerminalNumber,
    subPictureNumber   INTEGER (0..255),
    ...
}

VideoIndicateCompose ::= SEQUENCE
{
    compositionNumber  INTEGER (0..255),
    ...
}

-- =====
-- Сообщение индикации: Прочая индикация типа H.230
-- =====

MiscellaneousIndication ::=SEQUENCE
{
    logicalChannelNumber LogicalChannelNumber,
    type                  CHOICE
    {
        logicalChannelActive  NULL,          -- так же как H.230 AIA и VIA
        logicalChannelInactive NULL,          -- так же как H.230 AIM и VIS

        multipointConference  NULL,
        cancelMultipointConference NULL,

        multipointZeroComm    NULL,          -- так же как H.230 MIZ
        cancelMultipointZeroComm NULL,       -- так же как H.230 cancel MIZ

        multipointSecondaryStatus NULL,      -- так же как H.230 MIS
        cancelMultipointSecondaryStatus NULL, -- так же как H.230 cancel MIS

        videoIndicateReadyToActivate NULL,   -- так же как H.230 VIR

        videoTemporalSpatialTradeOff INTEGER (0..31), -- указывает текущее
                                                                -- соотношение

        ...,

```

```

        videoNotDecodedMBs          SEQUENCE
        {
            firstMB                  INTEGER (1..8192),
            numberOfMBs              INTEGER (1..8192),
            temporalReference        INTEGER (0..255),
            ...
        },
        transportCapability          TransportCapability
    },
    ...
}

-- =====
-- Сообщение индикации: Индикация дрожания
-- =====

JitterIndication                    ::=SEQUENCE
{
    scope CHOICE
    {
        logicalChannelNumber        LogicalChannelNumber,
        resourceID                  INTEGER (0..65535),
        wholeMultiplex              NULL
    },
    estimatedReceivedJitterMantissa INTEGER (0..3),
    estimatedReceivedJitterExponent INTEGER (0..7),
    skippedFrameCount              INTEGER (0..15) OPTIONAL,
    additionalDecoderBuffer         INTEGER (0..262143) OPTIONAL,
    -- 262143 - это 2^18 - 1
    ...
}

-- =====
-- Сообщение индикации: Расфазировка логического канала H.223
-- =====

H223SkewIndication                  ::=SEQUENCE
{
    logicalChannelNumber1           LogicalChannelNumber,
    logicalChannelNumber2           LogicalChannelNumber,
    skew INTEGER (0..4095),         -- единица измерения - миллисекунда
    ...
}

-- =====
-- Сообщение индикации: Максимальная расфазировка логического канала H.225.0
-- =====

H2250MaximumSkewIndication          ::=SEQUENCE
{
    logicalChannelNumber1           LogicalChannelNumber,
    logicalChannelNumber2           LogicalChannelNumber,
    maximumSkew                    INTEGER (0..4095), -- единица измерения
    -- - миллисекунда
    ...
}

-- =====
-- Сообщение индикации: Указание местонахождения MC
-- =====

MCLocationIndication                ::=SEQUENCE
{

```

```

    signalAddress          TransportAddress, -- это адрес сигнализации
                                -- вызова Н.323
                                -- объекта, который
                                -- содержит МС
    ...
}

-- =====
-- Сообщение индикации: Идентификация производителя
-- =====

VendorIdentification      ::=SEQUENCE
{
    vendor                  NonStandardIdentifier,
    productNumber          OCTET STRING (SIZE(1..256)) OPTIONAL,
                                -- для производителя
    versionNumber          OCTET STRING (SIZE(1..256)) OPTIONAL,
                                -- для номера продукта
    ...
}

-- =====
-- Сообщение индикации: Индикация нового виртуального канала АТМ
-- =====

NewATMVCIndication      ::=SEQUENCE
{
    resourceID              INTEGER(0..65535),
    bitRate                 INTEGER(1..65535),    -- единица измерения
                                                -- - 64 кбит/с
    bitRateLockedToPCRClock  BOOLEAN,
    bitRateLockedToNetworkClock  BOOLEAN,
    aal                     CHOICE
    {
        aall                 SEQUENCE
        {
            clockRecovery    CHOICE
            {
                nullClockRecovery  NULL,
                srtsClockRecovery  NULL,
                adaptiveClockRecovery  NULL,
                ...
            },
            errorCorrection    CHOICE
            {
                nullErrorCorrection  NULL,
                longInterleaver     NULL,
                shortInterleaver    NULL,
                errorCorrectionOnly  NULL,
                ...
            },
            structuredDataTransfer  BOOLEAN,
            partiallyFilledCells    BOOLEAN,
            ...
        },
        aal5                 SEQUENCE
        {
            forwardMaximumSDUSize  INTEGER (0..65535),    -- единицы измерения
                                                                -- - октеты
            backwardMaximumSDUSize  INTEGER (0..65535),    -- единицы измерения
                                                                -- - октеты
            ...
        },
        ...
    },
    ...
},
...
},
...
},
...
}

```



```

multiplex                                CHOICE
{
    noMultiplex                          NULL,
    transportStream                      NULL,
    programStream                        NULL,
    ...
},
...,
reverseParameters                        SEQUENCE
{
    bitrate                              INTEGER(1..65535), -- единица измерения - 64 кбит/с
    bitrateLockedToPCRClock              BOOLEAN,
    bitrateLockedToNetworkClock          BOOLEAN,
    multiplex                             CHOICE
    {
        noMultiplex                      NULL,
        transportStream                  NULL,
        programStream                    NULL,
        ...
    },
    ...
}
}

-- =====
-- Сообщение индикации: Ввод данных пользователем
-- =====

IV8 ::= OCTET STRING (SIZE(8))
-- начальное значение
-- 64-битовых блочных шифров

IV16 ::= OCTET STRING (SIZE(16))
-- начальное значение
-- 128-битовых блочных шифров

Params ::= SEQUENCE
{
    iv8 IV8 OPTIONAL, -- 8-октетовый вектор инициализации
    iv16 IV16 OPTIONAL, -- 16-октетовый вектор инициализации
    iv OCTET STRING OPTIONAL, -- вектор инициализации
    -- произвольной длины
    ...
}

UserInputIndication ::= CHOICE
{
    nonStandard alphanumeric NonStandardParameter,
    ...,
    userInputSupportIndication CHOICE
    {
        nonStandard basicString NonStandardParameter,
        basicString NULL, -- указывает на небезопасную базовую строку
        iA5String NULL, -- указывает на небезопасную строку iA5
        generalString NULL, -- указывает на небезопасную общую строку
        ...,
        encryptedBasicString NULL, -- указывает на зашифрованную базовую строку
        encryptedIA5String NULL, -- указывает на зашифрованную строку IA5
        encryptedGeneralString NULL -- указывает на зашифрованную общую строку
    },
    signal SEQUENCE
    {

```

```

    signalType          IA5String (SIZE (1) ^ FROM ("0123456789#*ABCD!")),
                        -- содержит фиктивный "!" при
                        -- использовании encryptedSignalType
    duration            INTEGER (1..65535) OPTIONAL,
                        -- миллисекунды
    rtp
    {
        timestamp      INTEGER (0..4294967295) OPTIONAL,
        expirationTime INTEGER (0..4294967295) OPTIONAL,
        logicalChannelNumber
        LogicalChannelNumber,
        ...
    } OPTIONAL,
    ...,
    rtpPayloadIndication
    paramS              NULL OPTIONAL,
                        Params OPTIONAL, -- любые параметры
                        -- времени выполнения
    encryptedSignalType
    OCTET STRING (SIZE(1)) OPTIONAL
                        -- зашифрованный тип сигнала (signalType)
},
signalUpdate          SEQUENCE
{
    duration            INTEGER (1..65535), -- миллисекунды
    rtp
    {
        logicalChannelNumber
        LogicalChannelNumber,
        ...
    } OPTIONAL,
    ...
},
extendedAlphanumeric SEQUENCE
{
    alphanumeric       GeneralString, -- содержит пустую строку,
                                    -- если используется
                                    -- encryptedAlphanumeric
    rtpPayloadIndication
    NULL OPTIONAL,
    ...,
    encryptedAlphanumeric
    SEQUENCE
    {
        algorithmOID   OBJECT IDENTIFIER,
        paramS          Params OPTIONAL, -- любые параметры времени выполнения
        encrypted      OCTET STRING, -- зашифрованная общая строка
        ...
    } OPTIONAL
},
encryptedAlphanumeric SEQUENCE
{
    algorithmOID       OBJECT IDENTIFIER,
    paramS              Params OPTIONAL, -- любые параметры времени выполнения
    encrypted          OCTET STRING, -- зашифрованная базовая строка
    ...
}
}

```

```

-- =====
-- Сообщение индикации: Контроль потока
-- =====

```

```

FlowControlIndication ::=SEQUENCE
{
    scope              CHOICE
    {
        logicalChannelNumber
        LogicalChannelNumber,
        resourceID     INTEGER (0..65535),
        wholeMultiplex
        NULL
    },
}

```

```

restriction                                CHOICE
{
    maximumBitRate                         INTEGER (0..16777215), -- единица
                                           -- измерения
                                           -- - 100 бит/с
    noRestriction                           NULL
},
...
}

-- =====
-- Сообщение индикации: Индикация реконфигурации мобильной связи
-- =====

MobileMultilinkReconfigurationIndication ::=SEQUENCE
{
    sampleSize                             INTEGER (1..255),
    samplesPerFrame                         INTEGER (1..255),
    ...
}

END

```

Приложение В

Сообщения: семантические определения

B.0 В этом приложении вводятся семантические определения и ограничения, накладываемые на элементы синтаксиса, определенные в предыдущем приложении.

B.0.1 MultimediaSystemControlMessage: Выбор типов сообщений. Сообщения, определенные в настоящей Рекомендации, классифицируются как запросы, ответы, команды и индикация.

B.0.2 RequestMessage: Сообщение с запросом вызывает действие удаленного терминала и требует от него немедленного ответа. Для передачи нестандартных запросов можно использовать сообщение nonStandard ("нестандартное").

B.0.3 ResponseMessage: Сообщение с ответом – это ответ на сообщение с запросом. Для передачи нестандартных ответов можно использовать сообщение nonStandard.

B.0.4 CommandMessage: Сообщение с командой требует выполнения действия, но не требует ответа в явном виде. Для передачи нестандартных команд можно использовать сообщение nonStandard.

B.0.5 IndicationMessage: Индикация содержит информацию, которая не требует действия или ответа. Для передачи нестандартной индикации можно использовать сообщение nonStandard.

B.0.6 NonStandardParameter ("нестандартный параметр"): Может применяться для индикации нестандартного параметра. Состоит из наименования и фактических параметров, которые кодируются как строка октетов.

B.0.7 NonStandardIdentifier ("нестандартный идентификатор"): используется для обозначения типа нестандартного параметра. Является либо идентификатором объекта, либо идентификатором типа H.221, представляющим собой строку октетов, состоящую точно из четырех перечисленных ниже октетов. Код страны состоит из двух октетов, первый из которых описан в Приложении A/T.35. Второй октет присваивается внутри страны, за исключением случая, когда первый октет равен 1111 1111 (при этом второй октет должен содержать код страны согласно Приложению B/T.35). Код изготовителя терминала состоит из двух октетов, которые присваивают внутри страны. Коды изготовителя – это коды, определенные для использования в Рекомендации МСЭ-Т H.320 [22]. Нестандартные идентификаторы H.245 могут быть или типа "объект", или типа "h221NonStandard", что выбирается по усмотрению изготовителя, определяющего нестандартное сообщение, поскольку идентификаторы объектов OBJECT IDENTIFIER и сообщения типа h221NonStandard поступают из

непересекающихся областей и их нельзя путать. Однако поскольку сообщения h221NonStandard также используются в Рекомендации МСЭ-Т Н.320, эти сообщения приходят из той же самой области, что и сообщения Н.320, и они должны иметь то же самое значение.

В.1 Сообщения, определяющие отношение "ведущий-ведомый"

Этот набор сообщений используется протоколом для определения того, какой терминал является ведущим терминалом, а какой – ведомым.

В.1.1 Определение отношения "ведущий-ведомый"

Это сообщение передается от MSDSE равноправному MSDSE.

terminalType – это число, которое определяет различные типы терминалов, такие как терминалы, MCU и шлюзы. Описание назначения значений типам терминалов выходит за рамки настоящей Рекомендации.

statusDeterminationNumber – это случайное число в диапазоне $0 \dots 2^{24} - 1$.

В.1.2 Подтверждение определения отношения "ведущий-ведомый"

Это сообщение применяется для подтверждения того, что данный терминал является ведущим терминалом или ведомым терминалом, как указано в решении. Если тип решения – это "ведущий" (master), то терминал, принимающий это сообщение, является ведущим, а если тип решения – это "ведомый" (slave), то этот терминал является ведомым.

В.1.3 Отклонение определения отношения "ведущий-ведомый"

Это сообщение используется для того, чтобы отказаться от сообщения MasterSlaveDetermination. Если причина имеет тип identicalNumbers ("идентичные числа"), это означает, что отказ вызван равенством случайных чисел и совпадением типов терминалов.

В.1.4 Сброс установленного отношения "ведущий-ведомый"

Это сообщение передается в случае тайм-аута.

В.2 Сообщения возможностей терминала

Этот набор сообщений предназначен для безопасного обмена возможностями между двумя терминалами.

В.2.1 Обзор

Передающий терминал присваивает каждому отдельному режиму, в котором этот терминал может работать, номер в capabilityTable ("таблице возможностей"). Например, каждому из режимов передачи аудиосигнала G.723.1, аудиосигнала G.728 и видеосигнала CIF Н.263 могут быть назначены отдельные номера.

Эти номера возможностей сгруппируются в структуры AlternativeCapabilitySet ("альтернативный набор возможностей"). Каждая структура AlternativeCapabilitySet указывает, что данный терминал может работать точно в одном режиме, задаваемом в наборе возможностей. Например, перечень AlternativeCapabilitySet {G.711, G.723.1, G.728} означает, что терминал может функционировать в любом из этих режимов передачи аудиосигнала, но не больше чем в одном режиме.

Такие структуры AlternativeCapabilitySet группируются в структуры simultaneousCapabilities ("одновременных возможностей"). Каждая структура simultaneousCapabilities указывает набор режимов, которые терминал может использовать одновременно. Например, структура simultaneousCapabilities, содержащая две структуры AlternativeCapabilitySet {H.261, H.263} и {G.711, G.723.1, G.728}, означает, что терминал может использовать любой из видеокодеков одновременно с любым из аудиокодеков. Набор simultaneousCapabilities {{H.261}, {H.261, H.263}, {G.711, G.723.1, G.728}} означает, что терминал может использовать два видеоканала и один аудиоканал одновременно: один видеоканал согласно H.261, другой видеоканал или в соответствии с H.261, или в соответствии с H.263 и один аудиоканал – согласно G.711, G.723.1 или G.728.

ПРИМЕЧАНИЕ. – Реальные возможности, записанные в capabilityTable, часто более сложны, чем показанные здесь возможности. Например, каждая из возможностей H.263 указывает детали, в том числе способность поддержки различных форматов изображения при заданных минимальных интервалах между изображениями и возможность использования дополнительных режимов кодирования.

Все возможности терминала описываются набором структур CapabilityDescriptor ("дескриптор возможностей"), каждая из которых представляет собой отдельную структуру simultaneousCapabilities и capabilityDescriptorNumber ("номер дескриптора возможности"). Передавая более одной структуры CapabilityDescriptor, терминал может сообщать о зависимостях между рабочими режимами путем описания различных наборов режимов, которые он может использовать одновременно. Например, терминал может передать две структуры CapabilityDescriptor, одну структуру {{H.261, H.263}, {G.711, G.723.1, G.728}}, как в предыдущем примере, и другую структуру {{H.262}, {G.711}}, это означает, что терминал также может использовать видеокodeк H.262, но только при использовании простого аудиокодека G.711.

Терминалы могут динамически добавлять возможности в течение сеанса связи, формируя дополнительные структуры CapabilityDescriptor, или удалять возможности, передавая исправленные структуры CapabilityDescriptor. Все терминалы должны передавать по меньшей мере одну структуру CapabilityDescriptor.

В.2.2 Набор возможностей терминала

Это сообщение содержит информацию относительно возможностей передачи и приема терминала. Оно также указывает используемую версию данной Рекомендации. Это сообщение передается передающим CESE равноправному принимающему CESE.

sequenceNumber ("последовательный номер") используется для обозначения экземпляров TerminalCapabilitySet ("набора возможностей терминала") для обеспечения возможности идентификации соответствующего ответа.

protocolIdentifier ("идентификатор протокола") используется для указания используемой версии настоящей Рекомендации. В Приложении D перечислены идентификаторы объектов, определенных для использования настоящей Рекомендацией.

multiplexCapability ("возможность мультиплексирования") задает возможности, связанные с мультиплексированием и адаптацией сети. Терминал должен включать multiplexCapability в первый передаваемый TerminalCapabilitySet.

V75Capability указывает возможности управляющего объекта V.75. audioHeader ("заголовок аудиосигнала") сообщает возможность заголовка аудиосигнала V.75.

В.2.2.1 Таблица возможностей

Таблица возможностей представляет собой пронумерованный перечень возможностей. Терминал должен иметь возможность поддерживать все, что перечислено в его таблице возможностей, но он не обязан одновременно выполнять более одной из них.

TerminalCapabilitySet может содержать ноль или более CapabilityTableEntry ("элементов таблицы возможностей"). Первоначально никакие элементы таблицы не определены. При получении элемента CapabilityTableEntry он заменяет ранее полученный CapabilityTableEntry с тем же самым номером CapabilityTableEntryNumber ("номером элемента таблицы возможностей"). CapabilityTableEntry без возможности может использоваться для удаления ранее полученного CapabilityTableEntry с тем же самым CapabilityTableEntryNumber.

В.2.2.2 Дескрипторы возможностей

CapabilityDescriptors ("дескрипторы возможностей") предназначены для указания возможности передачи и приема терминала. Каждый дескриптор CapabilityDescriptor содержит независимое утверждение о возможностях терминала.

Номер capabilityDescriptorNumber используется для нумерации дескрипторов CapabilityDescriptors. Если терминал предпочитает определенный режим для передачи или приема и желает выразить его при передаче своих возможностей, то он может сделать это, присвоив CapabilityDescriptors, относящимся к его предпочтительному режиму или режимам, малые значения capabilityDescriptorNumber.

simultaneousCapabilities – это набор AlternativeCapabilitySet. Он используется для перечисления возможностей, которые могут поддерживаться терминалом одновременно.

AlternativeCapabilitySet – это последовательность номеров CapabilityTableEntryNumbers. В наборе AlternativeCapabilitySet должны присутствовать только те элементы CapabilityTableEntry, которые были определены, хотя можно определить элементы CapabilityTableEntry и обращаться к ним в рамках одного и того же TerminalCapabilitySet. Если терминал имеет предпочтительный режим

передачи или приема и он желает выразить его при передаче своих возможностей, то он может сделать это, перечислив элементы в наборах `AlternativeCapabilitySets` в порядке снижения предпочтения.

Терминал должен быть в состоянии одновременно поддерживать любую одну возможность из каждого набора `AlternativeCapabilitySet`, указанного в наборе `simultaneousCapabilities`.

По меньшей мере у одного дескриптора возможностей должна быть следующая структура: должен быть по меньшей мере один набор `AlternativeCapabilitySet`, содержащий только возможности единственного типа среды для каждого типа среды, который может поддерживать терминал. Это должно обеспечить возможность выбора удаленным терминалом способа передачи, который включает в себя по меньшей мере один экземпляр для каждого типа среды, поддерживаемого приемником.

ПРИМЕЧАНИЕ 1. – Повторение возможности в наборе `AlternativeCapabilitySet` является избыточным и не передает дополнительной информации, тогда как повторение возможности в различных наборах `AlternativeCapabilitySets` в одном и том же `CapabilityDescriptor` указывает на возможность дополнительной одновременной поддержки экземпляров данной возможности.

ПРИМЕЧАНИЕ 2. – Терминалы, который не могут изменять распределение ресурсов, могут полностью указывать свои возможности, используя всего один дескриптор `CapabilityDescriptor`.

В.2.2.3 Возможность

Варианты `receiveVideoCapability` ("возможность приема видеосигнала"), `receiveAudioCapability` ("возможность приема аудиосигнала"), `receiveDataApplicationCapability` ("возможность приема для случая передачи данных"), `receiveUserInputCapability` ("возможность приема данных пользователя") и `receiveMultiplexedStreamCapability` ("возможность приема мультимплексированного потока") устанавливают возможность приема соответственно для `VideoCapability`, `AudioCapability`, `DataApplicationCapability`, `UserInputCapability` и `MultiplexedStreamCapability`.

Варианты `transmitVideoCapability` ("возможность передачи видеосигнала"), `transmitAudioCapability` ("возможность передачи аудиосигнала"), `transmitDataApplicationCapability` ("возможность передачи для случая передачи данных"), `transmitUserInputCapability` ("возможность передачи данных пользователя") и `transmitMultiplexedStreamCapability` ("возможность передачи мультимплексированного потока") устанавливают возможность передачи соответственно для `VideoCapability`, `AudioCapability`, `DataApplicationCapability`, `UserInputCapability` и `MultiplexedStreamCapability`.

Варианты `receiveAndTransmitVideoCapability` ("возможность приема и передачи видеосигнала"), `receiveAndTransmitAudioCapability` ("возможность приема и передачи аудиосигнала"), `receiveAndTransmitDataApplicationCapability` ("возможность приема и передачи для случая передачи данных"), `receiveAndTransmitUserInputCapability` ("возможность приема и передачи данных пользователя") и `receiveAndTransmitMultiplexedStreamCapability` ("возможность приема и передачи мультимплексированного потока") устанавливают возможность симметричного приема и передачи соответственно для `VideoCapability`, `AudioCapability`, `DataApplicationCapability` и `UserInputCapability` и `MultiplexedStreamCapability`. Эти кодовые точки могут быть полезны для указания того, что возможности приема и передачи не являются независимыми.

Поясним это на примере: терминал, который объявляет `{{Rx-G.723.1, Rx-G.729}, {Tx-G.723.1, Tx-G.729}}`, не показывает симметричное ограничение, поэтому он может принимать G.723.1 при передаче G.729, тогда как терминал, который заявляет `{{RxAndTx-G.723.1, RxAndTx-G.729}}`, действительно указывает симметричное ограничение, и поэтому он не может принимать G.723.1 при передаче G.729.

Если булева структура `h233EncryptionTransmitCapability` ("возможность передачи с шифрованием") принимает значение "истина", она свидетельствует о том, что терминал поддерживает шифрование в соответствии с Рекомендациями МСЭ-Т Н.233 [14] и Н.234 [15].

`h233IVResponseTime` ("время ответа") измеряется в миллисекундах, оно показывает минимальное время, которое передающая сторона должна переждать по требованию получателя после завершения передачи сообщения IV ("вектор инициализации") до начала использования нового вектора инициализации. Средства передачи вектора инициализации не определены в настоящей Рекомендации.

`ConferenceCapability` показывает различные возможности конференц-связи.

Возможность `multipointVisualizationCapability` (аналогичная MVC Н.230) включается в набор возможностей MCU или терминала, чтобы показать, что он должен правильно формировать или обрабатывать `conferenceResponse.broadcastMyLogicalChannel.grantedBroadcastMyLogicalChannel` (аналогично MVA Н.230) и

`conferenceResponse.broadcastMyLogicalChannel.deniedBroadcastMyLogicalChannel` (аналогично MVR Н.230) в ответ на `conferenceRequest.broadcastMyLogicalChannel` (аналогично Н.230 MCV).

h235SecurityCapability ("возможность защиты") показывает, какие возможности терминал поддерживает согласно Рекомендации МСЭ-Т Н.235 [16]. Поле mediaCapability ("характеристика среды") должно содержать ссылку на элемент таблицы возможностей, которые действительно содержат возможности передачи, приема или приема и передачи AudioCapability, VideoCapability, DataApplicationCapability или аналогичную возможность, указанную только с помощью NonStandardParameter.

Параметр EncryptionAuthenticationAndIntegrity ("аутентификация и целостность шифрования") показывает, какие возможности шифрования, аутентификации и целостности поддерживаются для передаваемой mediaCapability. Возможность mediaCapability определяет поддерживаемые алгоритмы передачи данных, аудиосигналов или видеосигналов, а также поддерживаемые методы распределения (например, прием, передача или прием и передача). Параметр maxPendingReplacementFor указывает максимальное число операций в открытых логических каналах, которые могут одновременно находиться в состоянии REPLACEMENT PENDING ("задержка замены"). Состояние REPLACEMENT PENDING наступает в том случае, когда логический канал был установлен с использованием параметра replacementFor, но заменяемый логический канал еще не был закрыт.

genericControlCapability ("возможность универсального управления") показывает универсальные возможности управления.

В.2.2.4 Возможности мультиплексирования

Возможности MultiplexCapability показывают возможности, относящиеся к мультиплексированию и адаптации сети. Терминал должен передавать MultiplexCapability в первом передаваемом наборе TerminalCapabilitySet. Если не определено иное, то эти возможности относятся к приему.

H222Capability: Показывает возможности мультиплексирования и адаптации сети, которые относятся к системе мультиплексирования, определенной в Рекомендации МСЭ-Т Н.222.1 [9].

Параметр numberOfVCs ("число внутренних каналов") указывает, сколько одновременных виртуальных каналов АТМ (VC) может поддерживать данный терминал. Сюда включаются все виртуальные каналы, которые обеспечивают транспортировку данных Н.245, Т.120, DSM-СС или каких-либо других данных, и все виртуальные каналы, передающие аудиовизуальную информацию. К ним не относится виртуальный канал, используемый для сигнализации Q.2931 [26].

vcCapability ("возможность виртуального канала") – это набор элементов, количество которых равно значению параметра numberOfVCs, он указывает существующие возможности для каждого доступного виртуального канала.

Последовательность aal1, если она имеется, задает возможность уровня адаптации 1 АТМ и его поддерживаемые опции, определенные в Рекомендации МСЭ-Т I.363 [25]. Кодовые точки определены в таблице В.1.

Таблица В.1/Н.245 –Кодовые точки уровня адаптации 1 АТМ

Кодовая точка АСН.1	Семантическое значение кодовой точки
nullClockRecovery	Метод восстановления тактовой частоты нулевого источника: передача по синхронному каналу.
srtsClockRecovery	Метод восстановления тактовой частоты источника синхронной остаточной временной метки.
adaptiveClockRecovery	Метод восстановления тактовой частоты адаптивного источника тактовой частоты.
nullErrorCorrection	Коррекция ошибок не поддерживается.
longInterleaver	Поддерживается метод прямого исправления ошибок для транспорта сигналов, чувствительного к потерям.
shortInterleaver	Поддерживается метод прямого исправления ошибок для транспорта сигналов, чувствительного к задержкам.
errorCorrectionOnly	Поддерживается метод прямого исправления ошибок без интерливинга.
structuredDataTransfer	Поддерживается передача структурированных данных.
partiallyFilledCells	Поддерживаются частично заполненные ячейки.

Последовательность aal5, если она имеется, показывает возможность уровня 5 адаптации ATM и указывает его поддерживаемые опции, определенные в Рекомендации МСЭ-Т I.363 [25]. Параметры forwardMaximumSDUSize и backwardMaximumSDUSize задают максимальный размер CPCS-SDU в прямом и обратном направлениях, измеряемый в октетах. Должны присутствовать последовательности aal1 или aal5 или обе эти последовательности.

Если булевы параметры transportStream и programStream принимают значение "истина", то они показывают возможность поддержки системы мультиплексирования Transport Stream (Транспортный поток) и Program Stream (Программный поток), соответственно [8].

availableBitRates показывает возможности скорости передачи данных для виртуального канала. Это последовательность различных скоростей передачи данных, которые могут поддерживаться, измеряемые в единицах по 64 кбит/с. Скорости передачи данных перечислены в порядке уменьшения, то есть максимальная поддерживаемая скорость передачи данных приводится первой в списке. Поддерживаемые скорости передачи могут задаваться как отдельные величины посредством поля singleBitRate ("одиночная скорость передачи") или как rangeOfBitRates ("диапазон скоростей передачи") от lowerBitRate ("минимальная скорость передачи") до higherBitRate ("максимальная скорость передачи"), показывая, что поддерживаются все значения между этим нижним пределом и верхним пределом, включая сами данные пределы. Указанные скорости передачи данных измеряются на AAL-SAP.

Последовательность aal1ViaGateway, если она имеется, показывает возможность уровня 1 адаптации ATM, поддерживаемую шлюзами преобразования AAL1/5. Используются те же кодовые точки, что и в случае последовательности aal1. Последовательность Q2931Address показывает один или несколько наборов номеров участника Q.2931 и субадреса участника.

H223Capability: Показывает возможности, определенные для системы мультиплексирования H.223 [10].

Если булев параметр transportWithI-frames принимает значение "истина", он показывает, что терминал может передавать и принимать сообщения управления каналом, используя информационные кадры (I-кадры) процедуры доступа модема к каналу LAPM в соответствии с определением этого протокола в Рекомендации МСЭ-Т V.42 [38].

Если булевы параметры videoWithAL1, videoWithAL2, videoWithAL3, audioWithAL1, audioWithAL2, audioWithAL3, dataWithAL1, dataWithAL2 и dataWithAL3 принимают значение "истина", они показывают возможность приема указанного типа среды (видеосигналов, аудиосигналов или данных), используя заданный уровень адаптации (AL1, AL2 или AL3).

Целые числа maximumAL2SDUSize и maximumAL3SDUSize определяют максимальное количество октетов в каждом SDU (сервисном блоке данных), которые данный терминал может принимать при использовании уровней адаптации типа 2 и 3, соответственно.

maximumDelayJitter показывает максимальный полный размах дрожания при мультиплексировании, который может вызвать передатчик. Оно измеряется в миллисекундах. Дрожание при мультиплексировании определяется как разность между временем доставки первого октета аудиокадра при передаче в мультиплексированном потоке и временем, когда он был бы доставлен при его передаче при постоянной скорости передачи без мультиплексирования.

h223MultiplexTableCapability: Показывает возможность терминалов принимать и обрабатывать элементы таблицы мультиплексирования.

basic ("базовые") указывает, что система мультиплексирования может принимать только основные MultiplexEntryDescriptors ("дескрипторы элементов мультиплексирования"), определенные в Рекомендации МСЭ-Т H.223 [8].

enhanced ("расширенные") указывает, что система мультиплексирования может принимать улучшенные дескрипторы MultiplexEntryDescriptors с дополнительными параметрами, которые определены ниже.

Параметр maximumNestingDepth ("максимальная глубина вложения") задает максимальную глубину вложения рекурсивно вызываемых полей subElementList. Дескрипторы MultiplexEntryDescriptors, которые не используют поле subElementList, должны рассматриваться как имеющие нулевую глубину вложения.

Параметр maximumElementListSize ("максимальный размер списка элементов") задает максимальное число полей в последовательности ACH.1 SEQUENCE.

Параметр maximumSubElementListSize ("максимальный размер списка подэлементов") указывает максимальное количество подэлементов в subElementList ("списке подэлементов").

Если булев параметр maxMUXPDUSizeCapability принимает значение "истина", он показывает, что передатчик может ограничить размер передаваемых блоков H.223 MUX-PDU. Если этот параметр входит в возможность приема, он не имеет никакого смысла.

Если булев параметр `nsrpSupport` принимает значение "истина", он свидетельствует о поддержке режима NSRP, определенного в Приложении А/Н.324.

MobileOperationTransmitCapability: Показывает возможность передачи уровней мультиплексирования, описанных в Приложении А/Н.223 и Приложении В/Н.223.

Если булев параметр `h223AnnexA` принимает значение "истина", он показывает, что терминал может передавать MUX-PDU согласно определению в Приложении А/Н.223.

Если булев параметр `h223AnnexADoubleFlag` принимает значение "истина", он показывает, что терминал может передавать блоки MUX-PDU в соответствии с определением Приложения А/Н.223 при дополнительном режиме двойного флага.

Если булев параметр `h223AnnexB` принимает значение "истина", он показывает, что терминал может передавать блоки MUX-PDU согласно определению Приложения В/Н.223.

Если булев параметр `h223AnnexBwithOptionalHeaderField` принимает значение "истина", он показывает, что терминал может передавать MUX-PDU в соответствии с определением Приложения В/Н.223 с дополнительным полем заголовка.

h223AnnexCCapability: Показывает возможность приема и обработки протокольных блоков AL-PDU согласно описанию, приведенному в Приложении С/Н.223 при соблюдении следующего условия.

Если булевы параметры `videoWithAL1M`, `videoWithAL2M`, `videoWithAL3M`, `audioWithAL1M`, `audioWithAL2M`, `audioWithAL3M`, `dataWithAL1M`, `dataWithAL2M` и `dataWithAL3M` принимают значение "истина", они показывают возможность приема данного типа среды (видеосигнал, аудиосигнал или данные), используя указанный уровень адаптации (AL1M, AL2M или AL3M).

Если параметр `alpdInterleaving` принимает значение "истина", он показывает возможность приема и обработки протокольных блоков данных AL-PDU, для которых применяется интерливинг.

Целый параметр `maximumA1MPDUSize` показывает максимальное количество октетов в каждом блоке данных протокола (PDU), которое терминал может принимать при использовании уровня адаптации AL1M.

Целые параметры `maximumA12MSDUSize` и `maximumA13MSDUSize` показывают максимальное количество октетов в каждом SDU, которые терминал может принимать при использовании уровня адаптации AL2M и AL3M, соответственно.

Если параметр `rsCodeCapability` принимает значение "истина", он показывает возможность приема AL-PDU, для которых указано кодирование Рида-Соломона.

Если присутствует параметр `bitRate` ("скорость передачи данных"), он показывает скорость передачи потока битов на выходе системы мультиплексирования Н.223.

Если присутствует параметр `mobileMultilinkFrameCapability`, он показывает возможность приема и обработки мобильных многоканальных кадров с заданными параметрами `maximumSampleSize` и `maximumPayloadLength`. Параметр `maximumSampleSize` ("максимальный размер отчета") задает максимальное количество октетов в каждом отчете, которое может обработать данный терминал. Параметр `maximumPayloadLength` ("максимальная длина полезной нагрузки") задает максимальную длину кадров в октетах, которую может обработать данный терминал.

V76Capability: Показывает возможности, относящиеся к системе мультиплексирования V.76.

Возможность `suspendResumeCapabilitywAddress` показывает способность поддерживать приостановку/возобновление V.76 с полем адреса. Возможность `suspendResumeCapabilitywoAddress` показывает способность поддерживать приостановку/возобновление V.76 без поля адреса.

`rejCapability` показывает, что функция контроля ошибок мультиплексирования V.76 может выполнять отбраковку.

`sREJCapability` показывает, что функция контроля ошибок мультиплексирования может выполнять селективную отбраковку.

`mREJCapability` указывает о возможности функции контроля ошибок мультиплексирования выполнять множественную селективную отбраковку.

Возможность `crc8bitCapability` показывает способность системы мультиплексирования использовать 8-битовый циклический избыточный код (CRC).

Возможность `crc16bitCapability` показывает способность системы мультиплексирования использовать 16-битовый CRC.

Возможность `src32bitCapability` показывает способность системы мультиплексирования использовать 32-битовый CRC.

`uuhCapability` показывает поддержку кадров UИH V.76.

`numOfDLCs` показывает количество каналов передачи данных (DLC), которые может поддерживать система мультиплексирования V.76.

`twoOctetAddressFieldCapability` показывает, что система мультиплексирования V.76 может поддерживать поле адреса, состоящее из двух октетов.

`loopBackTestCapability` показывает поддержку замкнутого цикла в соответствии с Рекомендацией МСЭ-Т V.76. `n401Capability` определяет максимальное значение N401, описанное в Рекомендации МСЭ-Т V.76. `maxWindowSizeCapability` задает максимальный размер окна, поддерживаемый системой мультиплексирования V.76.

H2250Capability: Показывает возможности, относящиеся к уровню формирования транспортных пакетов в среде H.225.0.

`maximumAudioDelayJitter` показывает максимальную амплитуду вносимого передатчиком дрожания при доставке аудиопакетов на транспортный уровень. Она измеряется в миллисекундах.

`receiveMultipointCapability` показывает возможности приема терминала при многосторонней конференц-связи.

`transmitMultipointCapability` показывает возможности передачи терминала при многосторонней конференц-связи.

`receiveAndTransmitMultipointCapability` показывает возможности приема и передачи терминала при многосторонней конференц-связи.

`mcCapability` показывает, что терминал способен играть роль объекта многоточечного управления (MC) при конференц-связи с централизованным или распределенным управлением.

`rtcpVideoControlCapability` показывает возможность обработки терминалом сообщения с полным запросом интрасети (FIR) RTCP и сообщений негативного подтверждения (NACK) RTCP.

`MediaPacketizationCapability` показывает, какие дополнительные схемы формирования транспортных пакетов поддерживаются данной конечной точкой.

`h261aVideoPacketization` показывает, что используется альтернативный вариант формата полезной нагрузки RTP H261, описанный в Рекомендации МСЭ-Т H.225.0.

`rtpPayloadType` показывает следующие схемы формирования пакетов с полезной нагрузкой RTP, поддерживаемые конечной точкой.

Дескриптор `payloadDescriptor` определяет семантику, связанную с типом полезной нагрузки (`payloadType`): если выбран номер RFC (`rfc-number`), то он указывает на официальный документ IETF, в котором определен формат полезной нагрузки. (Здесь не следует упоминать устаревшие документы серии RFC). Если выбран компонент `oid` (идентификатор объекта), то он определяет формат полезной нагрузки, представляющей собой часть принятой МСЭ Рекомендации или международный стандарт, утвержденный Международной организацией по стандартизации и зарегистрированный в соответствующем документе с данным идентификатором объекта. Это в равной мере относится и к обмену возможностями и к открытию логических каналов. Дескриптор `payloadDescriptor` должен заполняться так, как показано ниже:

- 1) Если Рекомендация МСЭ-Т H.225.0 определяет `oid` или номер `rfc` для применения кодеком, то должны выполняться положения Рекомендации МСЭ-Т H.225.0.
- 2) Если `oid` для кодека описан в Рекомендации МСЭ-Т по кодеку, то должен использоваться этот `oid`.
- 3) В ином случае, если кодек определен в Рекомендации МСЭ-Т (без `oid` в явном виде), то должен применяться компонент `oid`, при этом он должен соответствовать номеру Рекомендации МСЭ-Т в следующем виде: `{itu-t (0) recommendation (0) <символ> (<символ>) <номер>}`. Например, для Рекомендации МСЭ-Т G.711 следует использовать `oid {itu-t (0) recommendation (0) g (7) 711}`.
- 4) Если имеется документ RFC, определяющий формирование пакетов кодека, должен использоваться компонент с номером `rfc`.

- 5) В ином случае должен использоваться компонент `nonStandardIdentifier` ("нестандартный идентификатор").

Дополнительные сведения, относящиеся к определению типа полезной нагрузки (необязательные режимы, версии, скорости передачи данных и т. д., если они имеются), должны находиться в структурах `DataType` в `OpenLogicalChannel`. Декодер H.245 должен распознавать упомянутый выше `oid` (`oids`) в дополнение к любому определенному номеру `rfs` для кодека.

`payloadType` ("тип полезной нагрузки") может быть включен для того, чтобы показать, какой тип полезной нагрузки связан с данным форматом. Если он используется при обмене возможностями, то в качестве `payloadType` должен быть установлен статически присваиваемый тип полезной нагрузки в том и только в том случае, если он существует для данного формата полезной нагрузки. В противном случае `payloadType` должен пропускаться. При использовании совместно с `OpenLogicalChannel` `payloadType` должен показывать значение типа полезной нагрузки RTP (статический или динамический), который должен использоваться независимо от статически назначенного типа полезной нагрузки. Обратите внимание на то, что если значение типа полезной нагрузки находится в диапазоне 96..127, то идентичное значение типа полезной нагрузки должно также помещаться в `h2250LogicalChannelParameters.dynamicRTPPayloadType`.

Возможность `TransportCapability` показывает дополнительные транспортные возможности, такие как качество обслуживания и медийный тип канала.

`redundancyEncodingCapability` ("возможность избыточного кодирования") показывает, какие режимы кодирования избыточности поддерживаются (если таковые имеются). Для каждого элемента возможности параметр `redundancyEncodingMethod` ("режим избыточного кодирования") определяет тип кодирования, который должен определяться, первичное кодирование и виды вторичного кодирования, которые поддерживаются для данного первичного кодирования. Выбор схем кодирования зависит от выбранного режима. Параметр `rtpAudioRedundancyEncoding` относится к избыточному кодированию аудиосигнала; если этим режимом является выбранный `redundancyEncodingMethod` ("метод избыточного кодирования"), то действительны только `CapabilityEntryNumbers` ("номера элементов характеристики"), относящиеся к кодированию аудиосигнала. Параметр `rtpH263VideoRedundancyEncoding` показывает, что возможно избыточное кодирование видеосигнала в соответствии с Приложением N H.263+ или что логический канал должен быть открыт с использованием избыточного кодирования видеосигнала. Ниже приведены следующие дополнительные параметры:

Параметр `numberOfThreads` ("число потоков сообщений") показывает максимальное число потоков сообщений ("нитей"), которые может поддерживать отправитель/получатель во время обмена возможностями; он содержит фактическое число потоков сообщений для конкретного потока при открытии логического канала.

Параметр `framesBetweenSyncPoints` ("кадры между точками синхронизации") определяет максимальное количество видеок кадров, которые могут передаваться (сумма по всем цепочкам задач) между двумя точками синхронизации всех потоков сообщений во время обмена возможностями; он задает фактическое количество кадров для конкретного потока `OpenLogicalChannel`.

Параметр `frameToThreadMapping` определяет, какие режимы поддерживаются отправителем/получателем при обмене возможностями и какой режим должен использоваться при открытии логического канала: "round-robin" ("карусель") означает, что кадры назначаются потокам сообщений по кольцевому списку, причем первый кадр после точки синхронизации назначается потоку сообщений 0, второй – потоку сообщений 1, и т. д. "Заказной" формат позволяет определять произвольное соответствие между кадрами и потоками сообщений; при обмене возможностями поддержка заказного формата обозначается путем выбора этого компонента и кодирования произвольной (возможно пустой) последовательности SEQUENCE. Поддержка заказных форматов подразумевает поддержку "карусельных" соответствий.

Параметр `containedThreads` ("объединенные потоки сообщений") относится только к тем командам, которые открывают логические каналы: этот параметр показывает, какие из данных потоков сообщений передаются в открываемом логическом канале. Логический канал может содержать любое число потоков сообщений (от 1 до 15 включительно); однако два логических канала не должны определяться так, чтобы они содержали один и тот же поток сообщений.

В случае кодирования `rtpH263VideoRedundancyEncoding` параметр `secondaryEncoding` ("вторичное кодирование") не должен присутствовать; это относится также к структурам `H2250ModeParameters` и `RedundancyEncoding ASN.1`, определенным в Рекомендации МСЭ-Т H.245.

При открытии логического канала для избыточного кодирования видеосигнала первым должен открываться логический канал, содержащий поток сообщений 0, и на этот логический канал должны делать ссылки все остальные логические каналы, используя параметр `forwardLogicalChannelDependency` в команде `OpenLogicalChannel`.

Возможность `LogicalChannelSwitchingCapability` ("возможность переключения логических каналов") показывает способность получателя переключать предоставляемый поток (например, логический канал), используя команды включения и выключения `switchReceiveMedia` ("переключение принимаемой среды").

Возможность `t120DynamicPortCapability` показывает, что конечная точка может размещать вызов T.120 [32] по динамическому транспортному адресу, а не по стандартному хорошо известному адресу порта, как определено в Рекомендации МСЭ-Т T.123 [33].

MultipointCapability ("многоточечные возможности"): Показывает возможности терминала, относящиеся к работе в многоточечном режиме.

Возможность `multicastCapability` ("многоадресная возможность") свидетельствует о том, что терминал может обеспечить многоадресную передачу аудио или видеотрафика.

Возможность `multiUniCastConference` показывает, что терминал может участвовать в конференциях, организованных по принципу "связь с множеством адресов и одиночным адресом" (`multiUniCast`).

MediaDistributionCapability ("возможности распределения среды"): показывает способность терминала в плане передачи и приема среды в многосторонней конференции. Параметры `Centralized Control` ("централизованное управление") и `Audio` ("звук") должны иметь значение "истина" для терминалов стандарта H.323. Если поддерживается видеосигнал, то параметр `Centralized Video` должен иметь значение "истина". Если поддерживается рекомендация T.120, то должна присутствовать возможность `Centralized Data T.120 Data Application Capability`.

Централизованное и распределенное управление, передача аудио и видеосигналов свидетельствуют о способности терминала участвовать в конференции с такими типами распределения среды. При централизованной и распределенной передаче данных терминал может участвовать в конференции с этими типами распределения среды при конкретном протоколе передачи данных (`Data Application Protocol`). Возможность `MediaDistributionCapability` – это последовательность, позволяющая определить одновременные возможности (например, централизованная передача аудиосигнала при распределенной передаче видеосигнала или централизованная передача видеосигнала при распределенной передаче аудиосигнала, или конкретные возможности передачи данных в Протоколе доступа к данным).

Возможность `QOSCapabilities` отображает возможности, относящиеся к качеству обслуживания, такие как параметры `RSVPParameters` и `ATMParameters`.

Возможность `mediaChannelCapabilities` показывает, по какому транспорту может передаваться канал среды. `IP-UDP` означает, что конечная точка поддерживает транспортировку канала среды на уровне сети IP и на транспортном уровне UDP. В случае `IP-TCP` конечная точка поддерживает транспортировку канала среды на уровне IP сети и на уровне транспорта TCP. `atm-AAL5-UNIDIR` означает, что конечная точка поддерживает транспортировку канала среды по однонаправленному виртуальному каналу ATM AAL5. `atm-AAL5-BIDIR` означает, что конечная точка поддерживает транспортировку канала среды по двунаправленному виртуальному каналу ATM AAL5.

`RSVPParameters` содержит информацию конкретных параметров о протоколе RSVP.

`ATMParameters` включает в себя информацию конкретных параметров о виртуальном канале ATM.

`QosMode` показывает, обеспечивает ли режим гарантируемое качество обслуживания или же предоставляет режим регулируемой нагрузки, при котором не устанавливается ограничение сверху на сквозную задержку.

genericMultiplexCapability: Показывает универсальные возможности системы мультиплексирования.

В.2.2.5 Возможности видеосигнала

Здесь показаны возможности видеосигнала. Если в одной возможности `VideoCapability` задано нескольких отдельных возможностей, то это не означает, что они выполняются одновременно. Возможности одновременной обработки могут обозначаться появлением `VideoCapability` в различных наборах `AlternativeCapabilitySets` в рамках одного дескриптора `CapabilityDescriptor`.

ExtendedVideoCapability: задает возможности видеосигнала с помощью последовательности соответствующих `GenericCapability`.

`videoCapability` показывает последовательность альтернативных возможностей видеосигнала; любая из возможностей `VideoCapability` может использоваться с указанным расширением `videoCapabilityExtension`.

Если присутствует расширение `videoCapabilityExtension`, то оно показывает последовательность структур `GenericCapability`, соответствующих `videoCapability`.

Последовательность структур VideoCapability не должна содержать ExtendedVideoCapability.

Если возможность ExtendedVideoCapability.videoCapability используется в сообщении OpenLogicalChannel, то она должна содержать точно одну возможность VideoCapability.

H261VideoCapability: задает возможности H.261 [18].

Если присутствует параметр qcifMPI, то он определяет минимальный интервал изображения в единицах по 1/29,97 для кодирования и/или декодирования изображения QCIF, а если он отсутствует, то не указано никакой возможности изображения QCIF.

Если присутствует параметр cifMPI, то он определяет минимальный интервал изображения в единицах по 1/29,97 для кодирования и/или декодирования изображения CIF, а если он отсутствует, то не указано никакой возможности изображения CIF.

Если булев параметр temporalSpatialTradeOffCapability принимает значение "истина", он показывает, что кодер может изменять свой выбор между временным и пространственным разрешением в соответствии с командами удаленного терминала. Если он входит в возможность приема, то он не несет никакого значения.

Параметр maxBitRate показывает максимальную скорость передачи данных в единицах по 100 бит/с, с которой отправитель может передавать видеосигнал или получатель может его принимать.

Параметр stillImageTransmission задает возможность передачи неподвижных изображений в соответствии с Приложением D/H.261.

Если параметр videoBadMBsCap принимает значение "истина", он показывает возможность кодера принимать или возможность декодера передавать команду videoBadMBs. Если этот параметр входит в возможности передачи, то он указывает на способность кодера обрабатывать команды videoBadMBs и предпринимать соответствующие корректирующие действия, направленные на восстановление качества видеосигнала. Если же этот параметр входит в возможности приема, то он свидетельствует о возможности декодера передавать соответствующую индикацию videoBadMB.

H262VideoCapability: Показывает возможности H.262 [19].

Список булевских параметров показывает возможность обработки для конкретных профилей и уровней: значение "истина" означает, что такая операция возможна, в то время как значение "ложь" свидетельствует о невозможности соответствующей операции. Кодер должен создавать потоки битов в соответствии со спецификациями профиля и уровня, для которых он указал данную возможность, но при этом должны соблюдаться ограничения, накладываемые необязательными полями (см. ниже). Декодер должен уметь принимать все потоки битов, соответствующие профилю и уровню, для которых он указал данную возможность, при условии, что не нарушаются ограничения, устанавливаемые необязательными полями. Необязательные поля – это целые числа, соответствующие единицы измерений определены в таблице В.2.

Параметр videoBadMBsCap применяется в возможности H262VideoCapability таким же образом, как он используется в возможности H261VideoCapability.

Таблица В.2/Н.245 – Единицы измерения кодовых точек H.262

Кодовая точка АСН.1	Единицы измерения для указанного параметра
videoBitRate	400 бит/с
vbvBufferSize	16 384 бита
samplesPerLine	отсчетов на строку
linesPerFrame	Строк в кадре
framesPerSecond	Индекс frame_rate_code в таблице 6-4/Н.262
luminanceSampleRate	отсчетов в секунду

H263VideoCapability: Показывает возможности H.263 [20].

Если присутствует параметр sqcifMPI, то он задает минимальный интервал изображения в единицах по 1/29,97 для кодирования и/или декодирования изображений SQCIF, а если он отсутствует, то не указано никакой возможности изображения SQCIF.

Если присутствует параметр `qcifMPI`, то он задает минимальный интервал изображения в единицах $1/29,97$ для кодирования и/или декодирования изображений QCIF, а если он отсутствует, то не указано никакой возможности изображения QCIF.

Если присутствует параметр `cifMPI`, то он задает минимальный интервал изображения в единицах $1/29,97$ для кодирования и/или декодирования изображений CIF, а если он отсутствует, то не указано никакой возможности изображения CIF.

Если присутствует параметр `cif4MPI`, то он задает минимальный интервал изображения в единицах $1/29,97$ для кодирования и/или декодирования кадров 4CIF, а если он отсутствует, то не указано никакой возможности изображения 4CIF.

Если присутствует параметр `cif16MPI`, то он задает минимальный интервал изображения в единицах $1/29,97$ для кодирования и/или декодирования кадров 16CIF, а если он отсутствует, то не указано никакой возможности изображения 16CIF.

Параметр `maxBitRate` показывает максимальную скорость передачи данных, определяемую в единицах 100 бит/с, с которой передатчик может передавать видеосигнал или приемник может принимать видеосигнал.

Если булевы параметры `unrestrictedVector` (Приложение D/H.263), `arithmeticCoding` (Приложение E/H.263), `advancedPrediction` (Приложение F/H.263) и `pbFrames` (Приложение G/H.263) принимают значение "истина", то они показывают на возможность передачи и/или приема в этих дополнительных режимах, определенных в Приложениях к Рекомендации МСЭ-Т H.263.

Если булев параметр `temporalSpatialTradeOffCapability` ("возможность соотношения между временным и пространственным разрешением") принимает значение "истина", он показывает, что данный кодер может изменять соотношение между разрешением по времени и пространственным разрешением в соответствии с командами от удаленного терминала. Если он входит в состав возможностей приема, то он не имеет никакого смысла.

Если присутствует целый параметр `hrd-B`, он указывает параметр `B HRD` в Приложении В/H.263; он измеряется в единицах, равных 128 битам. Если он отсутствует, то применяется значение по умолчанию, определенное в Приложении В/H.263. Это возможность приемника, которая не имеет никакого значения в наборах возможностей передачи.

Если присутствует целый параметр `bppMaxKb`, он задает максимальное число битов на одно закодированное изображение, которое получатель может принимать и правильно декодировать; оно измеряется в единицах, равных 1024 битам. Если он отсутствует, то применяется значение, устанавливаемое по умолчанию, определенное в Рекомендации H.263. Это возможность приемника, которая не имеет смысла в наборах возможностей передачи.

Описанные ниже возможности предназначены для использования в определенных системах с очень низкой скоростью передачи кадров, таких как системы наблюдения.

Если присутствует параметр `slowSqcifMPI`, то он задает минимальный интервал изображения в единицах, соответствующих секундам на кадр для кодирования и/или декодирования изображений SQCIF. Если он отсутствует и отсутствует параметр `sqcifMPI`, то не задано никакой возможности изображений SQCIF. Если параметр `sqcifMPI` присутствует, то не должно быть `slowSqcifMPI`.

Если присутствует параметр `slowQcifMPI`, то он задает минимальный интервал изображения в единицах, соответствующих секундам на кадр для кодирования и/или декодирования изображений QCIF. Если он отсутствует и отсутствует параметр `qcifMPI`, то не задано никакой возможности изображений QCIF. Если параметр `qcifMPI` присутствует, то не должно быть `slowQcifMPI`.

Если присутствует параметр `slowCifMPI`, то он задает минимальный интервал изображения в единицах, соответствующих секундам на кадр для кодирования и/или декодирования изображений CIF. Если он отсутствует и отсутствует параметр `cifMPI`, то не задано никакой возможности изображений CIF. Если параметр `cifMPI` присутствует, то не должно быть `slowCifMPI`.

Если присутствует параметр `slowCif4MPI`, то он задает минимальный интервал изображения в единицах, соответствующих секундам на кадр для кодирования и/или декодирования изображений 4CIF. Если он отсутствует и отсутствует параметр `cif4MPI`, то не задано никакой возможности изображений 4CIF. Если параметр `cif4MPI` присутствует, то не должно быть `slowCif4MPI`.

Если присутствует параметр `slowCif16MPI`, то он задает минимальный интервал изображения в единицах, соответствующих секундам на кадр для кодирования и/или декодирования изображений 16CIF. Если он отсутствует и отсутствует параметр `cif16MPI`, то не задано никакой возможности изображений 16CIF. Если параметр `cif16MPI` присутствует, то не должно быть `slowCif16MPI`.

Значения MPI ("минимальный интервал изображения") применимы в тех случаях, когда используются все необязательные режимы, для которых указана возможность, а также когда используется любая их комбинация. Терминал может передать возможность для меньшего MPI, когда

некоторые опции не используются, передавая другую VideoCapability, включающую этот меньший MPI и указывающую на сокращенный набор опций.

Если булев параметр errorCompensation ("компенсация ошибки") принимает значение "истина", то он показывает возможность передачи и/или приема информации обратной связи для компенсации ошибок в соответствии с описанием в Добавлении I/H.263. Когда этот параметр является частью возможностей передачи, он свидетельствует о способности кодера обрабатывать показания videoNotDecodedMBs и компенсировать ошибки. Если этот параметр является частью возможности приема, он указывает способность декодера распознавать ошибочные MB, обрабатывать их как некодированные и передавать соответствующую индикацию videoNotDecodedMBs.

Если присутствует параметр enhancementLayerInfo, то он свидетельствует о способности кодера передавать или же о способности декодера принимать битовые потоки при дополнительном режиме масштабируемости (Приложение O/H.263). enhancementLayerInfo – это последовательность, задающая параметры конфигурации режима масштабируемости.

Если присутствует параметр H263Options, то он показывает возможность дополнительных режимов согласно Рекомендации МСЭ-Т H.263.

EnhancementLayerInfo: Показывает возможность режима масштабируемости согласно Рекомендации МСЭ-Т H.263.

baseBitRateConstrained показывает, ограничивается ли базовый уровень таким образом, чтобы не превышалась максимальная скорость передачи данных для возможности видеосигнала за вычетом суммы максимальных скоростей передачи по каждой из усовершенствованных опций.

Если присутствует параметр snrEnhancement, он указывает наличие возможности уровня улучшения отношения сигнал/шум. Размер набора равен количеству уровней snrEnhancement, которые терминал может поддерживать внутри одного логического канала.

Если присутствует параметр spatialEnhancement, он свидетельствует о наличии возможности улучшения пространственного уровня. Поток битов улучшенного уровня содержит изображение, которое в два раза больше по ширине или по высоте, или по тому и другому, чем изображение того уровня, на который имеется ссылка. Для того чтобы терминал мог обеспечивать пространственное улучшение по одному измерению (ширине или высоте), он также должен указать наличие возможности поддерживать соответствующий заказной формат изображения, требуемый улучшенным уровнем. Размер набора равен количеству уровней spatialEnhancement, которые терминал может поддержать внутри одного логического канала.

Если имеется bPictureEnhancement, то он свидетельствует о наличии возможности улучшенного уровня В-изображений. Размер набора соответствует количеству уровней bPictureEnhancement, которые терминал может поддерживать внутри одного логического канала.

Опции EnhancementOptions внутри последовательности bPictureEnhancement показывают, какие дополнительные опции может передавать кодер или принимать декодер в В-изображениях.

Параметр numberOfBPictures показывает максимальное количество В-изображений, которые терминал может поддерживать между последовательными парами опорных эталонных кадров, используемых при предсказании В-изображений. Например, если этот параметр равен 2, то два В-изображения могут передаваться между каждой парой Р-изображений или между другими опорными изображениями.

EnhancementOptions: Показывает возможности уровня улучшения масштабируемости.

Параметры в EnhancementOptions имеют те же самые семантические определения, что и параметры с такими же именами в возможности H263VideoCapability.

H263Options: Показывает возможность дополнительных необязательных режимов Рекомендации МСЭ-Т H.263.

Если параметр advancedIntraCodingMode принимает значение "истина", он указывает на возможность передавать или принимать в режиме усовершенствованного кодирования INTRA, описанном в Приложении I/H.263.

Если параметр deblockingFilterMode принимает значение "истина", он указывает на возможность передавать или принимать в режиме распаковки блоков, описанном в Приложении J/H.263.

Если параметр improvedPBFramesMode принимает значение "истина", он указывает возможность передавать или принимать в режиме улучшенных РВ кадров, определенном в Приложении M/H.263.

Если параметр `unlimitedMotionVectors` принимает значение "истина", он указывает на способность кодера или декодера поддерживать неограниченный диапазон вектора движения, если также указан `Unrestricted Motion Vector Mode` ("режим неограниченного вектора движения"), соответствующий Приложению D/H.263). `unlimitedMotionVectors` должен принимать значение "ложь", если параметр `unrestrictedVector` имеет значение "ложь" в той же самой возможности `H263VideoCapability` или режиме `H263VideoMode`.

Если параметр `fullPictureFreeze` принимает значение "истина", он показывает возможность кодера передавать или возможность декодера принимать команды полного стоп-кадра (`Full Picture Freeze`), описанные в Приложении L/H.263.

Если параметр `partialPictureFreezeAndRelease` принимает значение "истина", он показывает возможность кодера передавать или возможность декодера принимать команды частичного стоп-кадра и его сброса (`Release`), описанные в Приложении L/H.263.

Если параметр `resizingPartPicFreezeAndRelease` принимает значение "истина", он показывает возможность кодера передавать или возможность декодера принимать команды частичного стоп-кадра с изменением размеров и сброса, описанные в Приложении L/H.263.

Если параметр `fullPictureSnapshot` принимает значение "истина", он показывает возможность кодера передавать или возможность декодера принимать моментальные копии полноэкранного изображения содержимого видеосигнала в соответствии с Приложением L/H.263.

Если параметр `partialPictureSnapshot` принимает значение "истина", он показывает возможность кодера передавать или возможность декодера принимать моментальные копии части изображения содержимого видеосигнала в соответствии с Приложением L/H.263.

Если параметр `videoSegmentTagging` принимает значение "истина", он показывает, что кодер может передавать или что декодер может принимать теги видеосегмента (`Video Segment tagging`) содержимого видеосигнала в соответствии с описанием в Приложении L/H.263.

Если параметр `progressiveRefinement` принимает значение "истина", он показывает возможность кодера передавать или возможность декодера принимать теги прогрессивного уточнения (`Progressive Refinement tagging`) в соответствии с описанием в Приложении L/H.263. Кроме того, когда он принимает значение "истина", кодер должен отвечать на различные команды прогрессивного уточнения `doOneProgression`, `doContinuousProgressions`, `doOneIndependentProgression`, `doContinuousIndependentProgressions`, `progressiveRefinementAbortOne` и `progressiveRefinementAbortContinuous`. Кроме того, кодер должен вставлять теги начала сегмента прогрессивного уточнения (`Progressive Refinement Segment Start Tags`) и теги конца сегмента прогрессивного уточнения (`Progressive Refinement Segment End Tags`), определенные в Дополнительной спецификации режима улучшения Приложения L/H.263.

ПРИМЕЧАНИЕ. – Кодер может передавать теги последовательного улучшения, а декодер может принимать их даже в случае отсутствия соответствующего требования в смешанной команде (`miscellaneous command`).

Если параметр `dynamicPictureResizingByFour` принимает значение "истина", он показывает, что кодер или декодер способен поддерживать подрежим с изменением размеров изображения в четыре раза (при клиппировании) режима неявной повторной дискретизации эталонного изображения, рекомендованного в Приложении P/H.263.

Если параметр `dynamicPictureResizingSixteenthPel` принимает значение "истина", он показывает, что кодер или декодер может поддерживать изменение размеров эталонного изображения до любой ширины и высоты посредством применения режима неявной повторной дискретизации эталонного изображения, рекомендованного Приложением P/H.263 (при клиппировании).

Если параметр `dynamicWarpingHalfPel` принимает значение "истина", он показывает, что кодер или декодер может поддерживать операцию произвольного изменения масштаба эталонного изображения до любой ширины и высоты посредством применения режима неявной повторной дискретизации эталонного изображения, рекомендованного в Приложении P/H.263 (при любом режиме заполнения) посредством деформирования с точностью в половину пикселя.

Если параметр `dynamicWarpingSixteenthPel` принимает значение "истина", он показывает, что кодер или декодер может поддерживать операцию произвольного изменения масштаба эталонного изображения до любой ширины и высоты посредством применения режима неявной повторной дискретизации эталонного изображения, рекомендованного в Приложении P/H.263 (при любом режиме заполнения) посредством деформирования с точностью в половину пикселя или в одну шестнадцатую пикселя.

Если параметр `DynamicPictureResizingSixteenthPel` принимает значение "истина", то параметр `DynamicPictureResizingByFour` должен иметь значение "истина". Если параметр `DynamicWarpingSixteenthPel` принимает значение "истина", то параметры `DynamicWarpingHalfPel`, `DynamicPictureResizingByFour` и `DynamicPictureResizingSixteenthPel` должны иметь значение "истина".

Объявление возможности `dynamicPictureResizingByFour` для данного размера изображения, который называется здесь собственным размером изображения, подразумевает поддержку не более двух других размеров изображения, которые в настоящем документе именуются производными размерами

изображения. Если определить собственный размер изображения как изображение с шириной W и высотой H , то у поддерживаемых производных размеров изображения ширина изображения должна составлять $W/2$, а высота – $H/2$, а также высота $W/4$ и ширина $H/4$ в соответствии со следующим ограничением: каждый размер производного изображения должен поддерживаться при условии, что ширина его изображения не меньше 128, а высота изображения не меньше 96 (128 и 96 – это ширина и высота изображения формата SQCIF). Производные размеры изображения должны поддерживаться с теми же самыми дополнительными режимами, MPI (минимальный интервал изображения) и тактовой частотой, которые поддерживаются при собственном размере изображения.

Если параметр `independentSegmentDecoding` принимает значение "истина", он свидетельствует о том, что кодер или декодер может поддерживать режим декодирования независимых сегментов, описанный в Приложении R/H.263.

Если параметр `slicesInOrder-NonRect` принимает значение "истина", он показывает, что кодер или декодер может поддерживать подрежим режима структурирования сегментов (Slice Structured Mode) (Приложение K/H.263), в котором сегменты передаются по порядку и содержат макроблоки в порядке сканирования изображения.

Если параметр `slicesInOrder-Rect` принимает значение "истина", он показывает возможность кодера или декодера поддерживать подрежим режима структурирования сегментов (Slice Structured Mode) (Приложение K/H.263), в котором сегменты передаются по порядку и сегмент занимает прямоугольный участок изображения.

Если параметр `slicesNoOrder-NonRect` принимает значение "истина", он показывает возможность кодера или декодера поддерживать подрежим режима структурирования сегментов (Slice Structured Mode) (Приложение K/H.263), при котором сегменты содержат макроблоки в порядке сканирования изображения и не обязательно передаются по порядку.

Если параметр `slicesNoOrder-Rect` принимает значение "истина", он показывает возможность кодера или декодера поддерживать подрежим режима структурирования сегментов (Приложение K/H.263), при котором сектор занимает прямоугольный участок изображения и они не обязательно передаются по порядку.

Если параметр `alternateInterVLCMode` принимает значение "истина", он показывает, что кодер или декодер может поддерживать Режим попеременного кодирования с переменной длиной поля записи (Alternate Inter VLC Mode), описанный в Приложении S/H.263.

Если параметр `modifiedQuantizationMode` принимает значение "истина", он показывает, что кодер или декодер может поддерживать Режим модифицированного квантования, описанный в Приложении T/H.263.

Если параметр `reducedResolutionUpdate` принимает значение "истина", он показывает, что кодер или декодер может поддерживать Режим сниженного разрешения, определенный в Приложении Q/H.263.

Возможность `videoBadMBsCap` используется в `H263VideoCapability` так же, как она используется в `H261VideoCapability`.

Если параметр `dataPartitionedSlices` принимает значение "истина", он показывает, что кодер или декодер может поддерживать режим Data Partitioned Slice (Сегмента с разделенными данными), определенный в Приложении V/H.263. Параметр `dataPartitionedSlices` должен иметь значение "ложь", если все параметры `slicesInOrder-NonRect`, `slicesInOrder-Rect`, `slicesNoOrder-NonRect` и `slicesNoOrder-Rect` имеют значение "ложь" в том же сообщении `H263Options`.

Если параметр `fixedPointIDCT0` принимает значение "истина", он показывает, что кодер или декодер может поддерживать эталонное IDCT 0, определенное в Приложении W/H.263.

Если параметр `interlacedFields` принимает значение "истина", он показывает, что кодер или декодер может поддерживать чересстрочное кодирование полей, соответствующее определению в Приложении W/H.263.

Если параметр `currentPictureHeaderRepetition` принимает значение "истина", он показывает, что кодер или декодер может поддерживать повторение заголовка текущего изображения, заданное в Приложении W/H.263.

Если параметр `previousPictureHeaderRepetition` принимает значение "истина", он показывает, что кодер или декодер может поддерживать повторение заголовка предыдущего изображения в соответствии с определением в Приложении W/H.263.

Если параметр `nextPictureHeaderRepetition` принимает значение "истина", он показывает, что кодер или декодер может поддерживать повторение заголовка следующего изображения (при наличии указания на надежный опорный сигнал с разрешением по времени или без него) согласно определению в Приложении W/H.263.

Если параметры `currentPictureHeaderRepetition`, `previousPictureHeaderRepetition`, и `nextPictureHeaderRepetition` принимают значение "истина" и составляют часть возможностей получателя, то они показывают, что декодер может восстанавливать заголовок изображения в случае его искажения или потери, заменяя искаженный или потерянный заголовок изображения заголовком, переданным согласно Приложению W/H.263.

Если параметр `pictureNumber` принимает значение "истина", он показывает, что кодер может передавать номера изображений в соответствии с Приложением W/H.263 или что декодер может обнаруживать потери эталонного изображения на основе переданных номеров изображений.

Если параметр `spareReferencePictures` принимает значение "истина", он свидетельствует о том, что кодер может поддерживать формирование индикации запасных эталонных изображений в соответствии с определением в Приложении W/H.263 или о том, что декодер может использовать запасное эталонное изображение, если у него нет реального фактического эталонного изображения.

TransparencyParameters: Показывает параметры, определяющие "прозрачный" уровень видеосигнала.

Параметр `presentationOrder` показывает разбиение на прозрачные слои видеосигнала. Во время обмена возможностями значение параметр `presentationOrder` должен принимать одно из следующих значений: 0, 1 или 2. Если он равен 0, это свидетельствует о том, что обеспечивается тип поддержки типа прозрачности "фон эталонного изображения" (RPB), соответствующий определению в Приложении L/H.263. Если этот параметр равен 1, то это значит, что может использоваться изображение с фоном, регулируемым извне, а если он принимает значение 2, то поток битов может задавать применение или прозрачности типа "Фон эталонного изображения" (PRB), или прозрачности типа "изображение с фоном, регулируемым извне". При открытом логическом канале значение INTEGER определяет порядок представления. Уровень с более высоким порядком представления должен располагаться над слоем с более низким порядком представления. Параметр `presentationOrder` можно рассматривать как ось, перпендикулярную экрану, с увеличением параметра по направлению к зрителю.

Параметры `offset-x` и `offset-y` показывают смещение пикселя (в 1/8 части пикселя) для прозрачного передаваемого уровня по отношению к базовому уровню в единицах относительно базового уровня. При использовании в сообщении возможностей они определяют возможность смещения расположения прозрачного уровня видеосигнала и должны иметь значения, ограниченные величинами 1, 2, 4 или 8, в единицах 1/8 пикселя: например, если значение равно 4, то задана возможность смещения прозрачного уровня приращениями по 1/2 пикселя.

Параметры `scale-x` (масштаб по x) и `scale-y` (масштаб по y) задают коэффициент масштабирования, который должен применяться в соответствующей системе координат x и y к переданному прозрачному уровню до разбиения видеосигнала на уровни, в единицах относительно базового уровня. В сообщении с возможностями они задают максимальный коэффициент масштабирования, который может применяться: 1 показывает, что изменение масштаба не поддерживается, 2 означает, что он может удваивать размер уровня или сохранять его без масштабирования, 3 указывает, что он может его удваивать, утраивать или сохранять его без масштабирования и т. д.

Если булев параметр **`separateVideoBackChannel`** принимает значение "истина", он показывает, что терминал может поддерживать режим отдельного логического канала: никакая другая возможность видео не должна быть указана в этой `H263VideoCapability`: никакие значения `MPI` не должны присутствовать, и все остальные флаги режимов и содержание не играют никакой роли и должны принимать значение "ложь" или отсутствовать. При передаче в сообщении запроса режима `separateVideoBackChannel = true` (истина) должен передаваться как единственная возможность видеосигнала в этом режиме `H263VideoMode` и указывает, что получатель хочет вести прием по каналу, содержащему только данные обратного канала, соответствующие определению H.263. Если этот параметр присутствует в сообщении `OpenLogicalChannel`, то это значит, что данный логический канал предназначен только для сообщений по обратному видеоканалу и что никакой другой поток битов видеосигнала H.263 не должен передаваться по этому логическому каналу.

`refPictureSelection`: Показывает возможность режима выбора эталонного изображения (Приложение N/H.263), а также (дополнительно) возможность режима улучшенного выбора эталонного изображения (Приложение U/H.263).

Если имеется параметр `additionalPictureMemory`, то он указывает на наличие дополнительного объема памяти помимо того объема, который может использоваться нормальным декодером, который не поддерживает режим выбора эталонного изображения. Его отсутствие означает, что у кодера другого терминала нет никакой информации относительно дополнительного объема памяти, которую может использовать декодер. Присутствие этого параметра в `H263VideoMode` свидетельствует о наличии дополнительного объема памяти на изображение, используемого для декодирования.

Параметр `sqcifAdditionalPictureMemory` показывает, что кодер может передавать или что декодер может принимать поток битов H.263, что требует от декодера дополнительной памяти для хранения указанного количества изображений размера SQCIF или меньшего размера как по горизонтали, так и по вертикали, если в параметре `customPictureFormat` задана поддержка заказного формата для таких изображений.

Параметр `qcifAdditionalPictureMemory` указывает, что кодер может передавать или декодер может принимать поток битов H.263, что требует от декодера дополнительной памяти для хранения указанного количества изображений размера QCIF или меньшего размера как по горизонтали, так и по вертикали, если в параметре `customPictureFormat` задана поддержка заказного формата для таких изображений. Количество памяти на изображение, определенное в параметре `qcifAdditionalPictureMemory`, не должно превышать количество изображений, указанных в параметре `sqcifAdditionalPictureMemory` (если таковой имеется).

Параметр `cifAdditionalPictureMemory` показывает, что кодер может передавать или декодер может принимать поток битов H.263, что требует от декодера дополнительной памяти для хранения указанного числа изображений размера CIF или меньшего размера как по горизонтали, так и по вертикали, если в параметре `customPictureFormat` задана поддержка заказного формата для таких изображений. Количество памяти на изображение, указанное в параметре `cifAdditionalPictureMemory`, не должно превышать количество изображений, указанное в параметре `sqcifAdditionalPictureMemory` или `qcifAdditionalPictureMemory` (если они имеются).

Параметр `cif4AdditionalPictureMemory` показывает, что кодер может передавать или декодер может принимать поток битов H.263, что требует от декодера дополнительной памяти для хранения указанного количества изображений размера 4CIF или меньшего размера как по горизонтали, так и по вертикали, если в параметре `customPictureFormat` задана поддержка заказного формата для таких изображений. Количество памяти на изображение, определенное в параметре `cif4AdditionalPictureMemory`, не должно превышать количество изображений, указанное в параметрах `sqcifAdditionalPictureMemory`, `qcifAdditionalPictureMemory` или `cifAdditionalPictureMemory` (если они имеются).

Параметр `cif16AdditionalPictureMemory` показывает, что кодер может передавать или что декодер может принимать поток битов H.263, что требует от декодера дополнительной памяти для хранения указанного количества изображений размера 16CIF или меньшего размера как по горизонтали, так и по вертикали, если в параметре `customPictureFormat` задана поддержка заказного формата для таких изображений. Количество памяти на изображение, определенное в параметре `cif16AdditionalPictureMemory`, не должно превышать количество изображений, указанное в параметрах `sqcifAdditionalPictureMemory`, `qcifAdditionalPictureMemory`, `cifAdditionalPictureMemory` или `cif4AdditionalPictureMemory` (если они имеются).

Параметр `bigCpfAdditionalPictureMemory` показывает, что кодер может передавать или что декодер может принимать поток битов H.263, что требует от декодера дополнительной памяти для хранения указанного количества изображений заказного формата размера, указанного в параметре `customPictureFormat`, который превышает формат кадра 16CIF по горизонтали или по вертикали. Объем памяти на изображение, определенное в параметре `bigCpfAdditionalPictureMemory`, не должно превышать количество изображений, указанное в параметрах `sqcifAdditionalPictureMemory`, `qcifAdditionalPictureMemory`, `cifAdditionalPictureMemory`, `cif4AdditionalPictureMemory` или `cif16AdditionalPictureMemory` (если они имеются).

Параметр `videoMux` во время процедуры обмена возможностями показывает, что терминал может поддерживать режим VideoMux, показанный в Приложении N/H.263. Когда он принимает значение "истина", кодер или декодер могут использовать поток битов видеосигнала, содержащий сообщения обратного видеоканала. Его присутствие в `H263VideoMode` показывает, что предпочтительными являются сообщения обратного видеоканала в режиме VideoMux. При использовании их в `H263VideoMode` параметры `videoMux` и `separateVideoBackChannel` не должны оба принимать значение "истина".

Параметр `videoBackChannelSend` показывает, какой тип сообщений обратного видеоканала поддерживается терминалом. Наличие его в `H263VideoMode` указывает на предпочтительный тип сообщений обратного канала.

Параметр `none` (ни один) показывает, что кодер не может передавать или декодер не может принимать поток битов H.263, содержащий запросы на возврат любых сообщений обратного канала.

Параметр `askMessageOnly` показывает, что кодер может передавать или что декодер может принимать поток битов H.263, содержащий запросы о возврате только тех сообщений обратного канала, которые являются подтверждениями.

Параметр `nackMessageOnly` показывает, что кодер может передавать или декодер может принимать поток битов H.263, содержащий запросы о возврате только тех сообщений обратного канала, которые являются отрицательными подтверждениями.

Параметр `askOrNackMessageOnly` показывает, что кодер может передавать или декодер может принимать поток битов H.263, содержащий запросы о возврате тех сообщений обратного канала, которые являются подтверждениями или отрицательными подтверждениями, но только одного для заданного потока битов видеосигнала.

Параметр `askAndNackMessage` показывает, что кодер может передавать или декодер может принимать поток битов H.263, содержащий запросы о возврате тех сообщений обратного канала, которые являются подтверждениями и отрицательными подтверждениями.

Если присутствует параметр `enhancedReferencePicSelect`, то он указывает на возможность кодера или декодера использовать улучшенный режим выбора эталонного изображения, описанный в Приложении U/H.263. Если кодер может работать в улучшенном режиме выбора эталонного изображения из Приложения U/H.263, он также должен быть в состоянии принимать три следующие разные командные сообщения: `lostPicture`, `lostPartialPicture` и `recoveryReferencePicture` и совершать необходимые действия для восстановления качества декодированных изображений на удаленном конце.

Если присутствует параметр `subPictureRemovalParameters`, то он указывает на возможность удаления фрагмента эталонного изображения согласно Приложению U/H.263.

Параметр `mpuHorizMBs` показывает размер минимальной единицы изображения по горизонтали в макроблоках для удаления фрагмента эталонного изображения согласно Приложению U/H.263.

Параметр `mpuVertMBs` показывает размер минимальной единицы изображения по вертикали в макроблоках для удаления фрагмента эталонного изображения согласно Приложению U/H.263.

Параметр `mpuTotalNumber` задает общий объем памяти буфера на много изображений при работе по удалению фрагмента в соответствии с Приложением U/H.263, указывая ее в единицах, соответствующих минимальному изображению.

CustomPictureClockFrequency: Показывает возможность поддержки тактовой частоты заказного изображения, если присутствует в качестве возможности, и параметры тактовой частоты синхронизации заказного изображения, если входит в `OpenLogicalChannel` и `RequestMode`.

Если при использовании в `OpenLogicalChannel`, если `customPictureClockFrequency` содержит более одного члена в своем наборе, то потоку битов видеосигнала разрешено переходить на различные тактовые частоты изображения (PCF) в пределах этого набора в одном потоке битов видеосигнала. Даже когда в этом наборе имеется только одна PCF, если любые значения MPI передаются для стандартной PCF на более высоких уровнях в одном сообщении (например, в одном сообщении `H263VideoCapability`), то в одном потоке битов возможен переход от стандартной PCF на заказную PCF. Если желательно указать, что PCF не следует менять внутри одного потока битов, то должны передаваться данные, относящиеся только к одной PCF (или только значения MPI для стандартной PCF, или только `customPictureClockFrequency`).

Параметр `clockConversionCode` задает код преобразования тактовой частоты, если согласно Рекомендации МСЭ-Т H.263 используется тактовая частота заказного изображения.

Параметр `clockDivisor` показывает естественное двоичное представление значения делителя тактовой частоты. Заказная тактовая частота изображения задается частотой $1\ 800\ 000 / (\text{делитель тактовой частоты} * \text{коэффициент преобразования тактовой частоты})$ Гц.

Если присутствует параметр `sqcifMPI`, то он указывает минимальный интервал изображения в единицах по $1 / (\text{тактовая частота заказного изображения})$ для кодирования и/или декодирования изображений SQCIF. Его отсутствие означает, что не указано никакой возможности для изображений SQCIF.

Если присутствует параметр `qcifMPI`, то он указывает минимальный интервал изображения в единицах по $1 / (\text{тактовая частота заказного изображения})$ для кодирования и/или декодирования изображений QCIF. Его отсутствие означает, что не указано никакой возможности для изображений QCIF.

Если присутствует параметр `cifMPI`, то он указывает минимальный интервал изображения в единицах по $1/(\text{тактовая частота заказного изображения})$ для кодирования и/или декодирования изображений CIF. Его отсутствие означает, что не указано никакой возможности для изображений CIF.

Если присутствует параметр `cif4MPI`, то он указывает минимальный интервал изображения в единицах по $1/(\text{тактовая частота заказного изображения})$ для кодирования и/или декодирования изображений 4CIF. Его отсутствие означает, что не указано никакой возможности для изображений 4CIF.

Если присутствует параметр `cif16MPI`, то он указывает минимальный интервал изображения в единицах по $1/(\text{тактовая частота заказного изображения})$ для кодирования и/или декодирования изображений 16CIF. Его отсутствие означает, что не указано никакой возможности для изображений 16CIF.

CustomPictureFormat: Показывает возможность поддержки формата заказного изображения, если присутствует в качестве возможности, и параметры формата заказного изображения, если входит в `OpenLogicalChannel` и `RequestMode`.

Параметры `maxCustomPictureWidth`, `maxCustomPictureHeight`, `minCustomPictureWidth`, `minCustomPictureHeight` указывают диапазон размеров изображения, которые может поддерживать кодер или декодер, в единицах по 4 пикселя, а также требуемый размер изображения в случае применения вместе с `RequestMode`.

Параметр `standardMPI` указывает минимальный интервал изображения в единицах по $1/29,97$, когда не используется тактовая частота заказного изображения.

Параметр `customPCF` указывает параметры для тактовой частоты заказного изображения при использовании совместно с форматом заказного изображения.

Параметр `clockConversionCode` указывает код преобразования тактовой частоты, когда согласно Рекомендации МСЭ-Т Н.263 используется тактовая частота заказного изображения.

Параметр `clockDivisor` показывает естественное двоичное представление величины делителя тактовой частоты. Тактовая частота заказного изображения равна $1\ 800\ 000/(\text{делитель тактовой частоты} * \text{коэффициент преобразования тактовой частоты})$ Гц.

Параметр `customMPI` указывает минимальный интервал изображения в единицах $1/(\text{тактовая частота заказного изображения})$ для кодирования и/или декодирования изображений в размере формата заказного изображения.

Параметр `pixelAspectInformation` указывает, что кодер или декодер может поддерживать различные коэффициенты сжатия пикселя, он задает требуемый коэффициент сжатия пикселя в случае использования вместе с `RequestMode`.

Параметр `pixelAspectCode` указывает возможность поддержки коэффициента сжатия пикселя, указанного в коде PAR, описанном в Рекомендации МСЭ-Т Н.263.

`extendedPAR: width` (ширина), `height` (высота) показывают возможность поддержки коэффициента сжатия пикселя, как указано в расширенном коде коэффициента сжатия пикселя (EPAR) согласно Рекомендации МСЭ-Т Н.263.

H263VideoModeCombos

Если присутствует `h263VideoModeCombos`, то он используется для индикации зависимостей между необязательными режимами Рекомендации МСЭ-Т Н.263. Не предполагается, что комбинации режимов, возможности которых указаны в `h263VideoModeCombos`, разрешены к использованию вместе с другими необязательными режимами, передаваемыми на более высоких уровнях в одних и тех же `H263Options` или `H263VideoCapability` или сообщении `H263VideoMode`, за исключением случая, отмеченного в четвертом абзаце этого пункта и в третьем абзаце следующего пункта. Другими словами, если булевы величины поддержки некоторого режима для тех же режимов, которые содержат булевы величины в `H263VideoModeCombos`, указаны на более высоких уровнях синтаксиса в `h263Mode` или `H263Capability`, то не предполагается, что эти режимы также применимы в несвязанных комбинациях с режимами, объявленными в `H263VideoModeCombos`.

`h263VideoUncoupledModes` указывает, какие необязательные режимы работы в соответствии с Н.263 могут включаться или выключаться независимо друг от друга любым синтаксически правильным образом для некоторого изображения и какие могут включаться и выключаться независимо от режимов, заданных в `h263VideoCoupledModes`, которые передаются в той же последовательности `H263VideoModeCombos`.

h263VideoCoupledModes показывает один или более наборов необязательных режимов работы H.263, которые могут вместе включаться или выключаться для изображения в потоке битов H.263, но для которых не подразумевается возможность независимого включения или выключения любого подмножества этих режимов. Любой набор режимов, которые заданы как связанные в сообщении h263VideoCoupledModes, может использоваться вместе с полным набором или любым подмножеством режимов, которые определены как несвязанные в сопровождающем сообщении h263VideoUncoupledModes внутри того же сообщения H263VideoModeCombos. В содержимом каждого сообщения H263ModeComboFlags из сообщения h263VideoCoupledModes по меньшей мере два булевых флага должны быть установлены равными "истине", и не должно быть набора флагов режимов, установленных равными "истине", поскольку это указывает на связанную комбинацию режимов, которая синтаксически не разрешена в том же самом изображении потока битов H.263.

Некоторые необязательные возможности Рекомендации МСЭ-Т H.263 не включены в H263ModeComboFlags, поскольку считается, что они вряд ли потребуют связи при реализации. В частности, к ним относятся возможности, заданные в Приложении L/H.263 (например, fullPictureFreeze, partialPictureFreezeAndRelease и resizingPartPicFreezeAndRelease), а также необязательные форматы изображений и необязательные тактовые частоты изображений. Если о поддержке каких-либо подобных возможностей сообщается на более высоком уровне в тех же H263Options или H263VideoCapability или в сообщении H263VideoMode, то эти возможности должны выполняться несвязанным образом при комбинациях режимов, сигналы о которых передаются в H263VideoModeCombos. Ниже приведен довольно сложный пример использования H263VideoModeCombos.

В этом примере H263VideoCapability указывает, что поддерживаются advancedPrediction и unrestrictedVector, а также (в том же сообщении H263VideoCapability) внутри сообщения H263Options показано, что поддерживается dynamicPictureResizingByFour и (в том же сообщении H263VideoCapability) в сообщении H263VideoModeCombos находится сообщение h263VideoUncoupledModes, которое свидетельствует о поддержке advancedIntraCodingMode несвязанным образом, а в сообщении h263VideoCoupledModes указывается, что связанным образом поддерживаются modifiedQuantizationMode и slicesInOrder-NonRect. Это далее означает, что поток битов видеосигнала может содержать (только) изображения со следующими комбинациями режимов: None, advancedPrediction, unrestrictedVector, dynamicPictureResizingByFour, advancedPrediction с unrestrictedVector, advancedPrediction с dynamicPictureResizingByFour, unrestrictedVector с dynamicPictureResizingByFour, advancedPrediction с unrestrictedVector с dynamicPictureResizingByFour, advancedIntraCodingMode, modifiedQuantizationMode с slicesInOrder-NonRect, а также advancedIntraCodingMode с modifiedQuantizationMode с slicesInOrder-NonRect.

H263ModeComboFlags

Отдельные параметры H263ModeComboFlags имеют то же значение, что и параметры с тем же самым именем в H263VideoCapability и H263Options.

Параметр unlimitedMotionVectors должен принимать значение "ложь", если unrestrictedVector принимает значение "ложь" в том же самом сообщении H263VideoUncoupledModes. Параметр unlimitedMotionVectors должен принимать значение "ложь", если unrestrictedVector принимает значение "ложь" в том же самом сообщении H263VideoCoupledModes и в сообщении H263VideoUncoupledModes в том же сообщении H263VideoModeCombos.

Если параметр referencePicSelect принимает значение "истина", то он указывает на способность кодера или декодера использовать режим выбора эталонного изображения согласно Рекомендации МСЭ-Т H.263. Если принять значение "истина", конкретные параметры, определяющие, как может использоваться режим выбора эталонного изображения, должны быть такими, как те, которые передаются в поле refPictureSelection того же сообщения H263Options. Параметр referencePicSelect не должен принимать значение "истина", если refPictureSelection не присутствует в том же сообщении H263Options.

Параметр enhancedReferencePicSelect должен принимать значение "ложь", если параметр referencePicSelect имеет значение "ложь" в том же сообщении H263VideoUncoupledModes. Параметр enhancedReferencePicSelect должен принимать значение "ложь", если параметр referencePicSelect имеет значение "ложь" в том же сообщении H263VideoCoupledModes и в сообщении H263VideoUncoupledModes в том же сообщении H263VideoModeCombos.

Параметр `dataPartitionedSlices` должен принимать значение "ложь", если все следующие параметры: `slicesInOrder-NonRect`, `slicesInOrder-Rect`, `slicesNoOrder-NonRect` и `slicesNoOrder-Rect` имеют значение "ложь" в том одном и том же сообщении `H263VideoUncoupledModes`. Параметр `dataPartitionedSlices` должен принимать значение "ложь", если все следующие параметры: `slicesInOrder-NonRect`, `slicesInOrder-Rect`, `slicesNoOrder-NonRect` и `slicesNoOrder-Rect` имеют значение "ложь" в том же сообщении `H263VideoCoupledModes` и в сообщении `H263VideoUncoupledModes` в том же сообщении `H263VideoModeCombos`.

IS11172 VideoCapability: Показывает возможности IS11172 [44].

`constrainedBitstream` показывает возможность потоков битов, в которой флаг `constrained_parameters` установлен равным "1": значение "истина" означает, что такая работа возможна, в то время как значение "ложь" свидетельствует о невозможности такой работы. Кодер должен создавать потоки битов в пределах ограничений, накладываемых необязательными полями (см. ниже). Декодер должен иметь возможность принимать все потоки битов в пределах ограничений, указываемых необязательными полями. Необязательные поля – это целые числа, а соответствующие единицы измерения определены в таблице В.3.

Возможность `videoBadMBsCap` используется в возможности `IS11172VideoCapability` тем же самым способом, как она используется в возможности `H261VideoCapability`.

Таблица В.3/Н.245 – Единицы измерения кодовых точек IS11172-2

Кодовая точка АСН.1	Единица измерения соответствующего параметра
<code>videoBitRate</code>	400 бит/с
<code>vbvBufferSize</code>	16 384 бита
<code>samplesPerLine</code>	отсчетов на строку
<code>linesPerFrame</code>	строк на кадр
<code>pictureRate</code>	см. раздел 2.4.3.2 IS11172-2
<code>luminanceSampleRate</code>	отсчетов в секунду

genericVideoCapability: показывает универсальные возможности видеосигнала.

В.2.2.6 Возможности аудиосигнала

Здесь показаны возможности аудиосигнала. Если в одной возможности `AudioCapability` дано нескольких возможностей, то это не означает, что они могут выполняться одновременно. На способность одновременного выполнения возможностей может указывать появление `AudioCapability` в различных `AlternativeCapabilitySets` в рамках одного дескриптора `CapabilityDescriptor`.

Способность передавать и/или принимать аудиосигнал согласно Рекомендациям серии G задается выбором целых чисел. Если используется система мультиплексирования Н.222.1, то эти числа относятся к размеру имеющегося буфера STD, показанному в единицах по 256 октетов. Если применяется система мультиплексирования Н.223, то эти числа обозначают максимальное количество аудиокадров на AL-SDU. Если используется система мультиплексирования Н.225.0, то эти числа указывают максимальное число аудиокадров на пакет: конечная точка должна поддерживать прием любого количества кадров в пакете до максимального числа, заданного в `AudioCapability`; кроме того, конечная точка не должна передавать больше кадров на пакет, чем она указывает в своей возможности передачи `AudioCapability`. В таблице В.4 приведено точное значение кодовых точек.

Таблица В.4/Н.245 – Кодовые точки аудиосигнала для Рекомендаций серии G

Кодовая точка АСН.1	Семантическое значение кодовой точки
g711Alaw64k	Аудиосигнал, соответствующий спецификации G.711 со скоростью передачи 64 кбит/с, преобразование по А-закону
g711Alaw56k	Аудиосигнал, соответствующий спецификации G.711 со скоростью передачи 56 кбит/с, преобразование по А-закону, усеченный до 7 бит
g711Ulaw64k	Аудиосигнал, соответствующий спецификации G.711 со скоростью передачи 64 кбит/с, преобразование по μ -закону
g711Ulaw56k	Аудиосигнал, соответствующий спецификации G.711 со скоростью передачи 56 кбит/с, преобразование по μ -закону, усеченный до 7 бит
g722-64k	Аудиосигнал, соответствующий спецификации G.722 с частотой 7 кГц со скоростью передачи 64 кбит/с
g722-56k	Аудиосигнал, соответствующий спецификации G.722 с частотой 7 кГц со скоростью передачи 56 кбит/с
g722-48k	Аудиосигнал, соответствующий спецификации G.722 с частотой 7 кГц со скоростью передачи 48 кбит/с
g7231	Аудиосигнал, соответствующий спецификации G.723.1 со скоростью передачи 5,3 или 6,3 кбит/с
g728	Аудиосигнал, соответствующий спецификации G.728 со скоростью передачи 16 кбит/с
g729	Аудиосигнал, соответствующий спецификации G.729 со скоростью передачи 8 кбит/с
g729AnnexA	Аудиосигнал, соответствующий Приложению A/G.729 со скоростью передачи 8 кбит/с
g729wAnnexB	Аудиосигнал, соответствующий спецификации G.729 со скоростью передачи 8 кбит/с с подавлением шума в паузе согласно Приложению В
g729AnnexAwAnnexB	Аудиосигнал, соответствующий Приложению A/G.729 со скоростью передачи 8 кбит/с подавлением шума в паузе согласно Приложению В
g7231AnnexCCapability	Аудиосигнал, соответствующий спецификации G.723.1 и Приложению C/G.723.1
gsmFullRate	Кодирование речи с полной скоростью (GSM 06.10)
gsmHalfRate	Кодирование речи с половинной скоростью (GSM 06.20)
gsmEnhancedFullRate	Усовершенствованное кодирование речи (EFR) с полной скоростью (EFR) (GSM 06.60)
g729Extensions	Расширения, соответствующие спецификации G.729

G7231: Указывает на возможность обработки аудиосигнала кодека, соответствующего спецификации G.723.1. `maxAl-sduAudioFrames` показывает максимальное количество аудиокадров на AL-SDU. Если булев параметр `silenceSupression` принимает значение "истина", он показывает возможность использования подавления паузы, определенное в Приложении A/G.723.1.

G7231AnnexCCapability: Указывает на возможность обработки аудиосигнала кодека, соответствующего спецификации G.723.1 и Приложения C к ней. `maxAl-sduAudioFrames` определяет максимальное количество кадров аудиосигнала на AL-SDU. Если булев параметр `silenceSupression` принимает значение "истина", он свидетельствует о возможности использования подавления паузы, определенного в Приложении A/G.723.1. `g723AnnexCAudioMode` не должен присутствовать, если возможность `G7231AnnexCCapability` включена в сообщение `TerminalCapabilitySet`, но он должен присутствовать, если `G7231AnnexCCapability` включена в сообщение `OpenLogicalChannel`. В полях `highRateMode0`, `highRateMode1`, `lowRateMode0`, `lowRateMode1`, `sidMode0` и `sidMode1` показано количество октетов на кадр для каждого режима передачи аудиосигнала и защиты от ошибок G.723.1 и Приложения C/G.723.1, которые будут использоваться в логическом канале.

IS11172AudioCapability: Показывает возможность обработки аудиосигнала, закодированного в соответствии со стандартом ИСО/МЭК 11172-3 [45].

Булевы параметры, имеющие значение "истина", показывают, что возможен конкретный режим работы, тогда как значение "ложь" означает, что он невозможен. Булевы параметры `audioLayer1`, `audioLayer2` и `audioLayer3` определяют, какие уровни кодирования аудиосигнала могут обрабатываться. Булевы параметры `audioSampling32k`, `audioSampling44k1` и `audioSampling48k` указывают, какие из частот дискретизации аудиосигнала 32 кГц, 44,1 кГц и 48 кГц, соответственно, могут обрабатываться. Булевы параметры `singleChannel` и `twoChannels` задают характеристику работы по одиночному каналу и по стерео/двойному каналу, соответственно. Целый параметр `bitRate` определяет максимально возможную скорость передачи аудиосигнала, он измеряется в кбит/с.

IS13818AudioCapability: Указывает на возможность обработки аудиосигнала, закодированного в соответствии со стандартом ИСО/МЭК 13818-3 [46].

Булевы переменные, имеющие значение "истина", показывают, что возможен конкретный режим работы, тогда как значение "ложь" означает, что он невозможен. Булевы переменные audioLayer1, audioLayer2 и audioLayer3 определяют, какие уровни кодирования аудиосигнала могут обрабатываться. Булевы параметры audioSampling16k, audioSampling22k05, audioSampling24k, audioSampling32k, audioSampling44k1 и audioSampling48k указывают, какие из частот дискретизации аудиосигнала 16 кГц, 22,05 кГц, 24 кГц, 32 кГц, 44,1 кГц и 48 кГц соответственно могут обрабатываться.

Булевы параметры, относящиеся к многоканальной работе, определяют возможность работы в конкретных режимах, как указано в таблице В.5.

Таблица В.5/Н.245 – Многоканальные кодовые точки ИСО/МЭК 13818-3

Кодовая точка АСН.1	Семантическое значение кодовой точки
singleChannel	Один канал, использующий конфигурацию 1/0. Одноканальный режим (как в стандарте ИСО/МЭК 11172-3)
twoChannels	Два канала, использующие конфигурацию 2/0. Стерео режим или двухканальный режим (как в стандарте ИСО/МЭК 11172-3)
threeChannels2-1	Три канала, использующие конфигурацию 2/1. Левый канал, правый канал и одиночный канал окружающего звука
threeChannels3-0	Три канала, использующие конфигурацию 3/0. Левый, центральный и правый каналы без канала окружающего звука
fourChannels2-0-2-0	Четыре канала, использующие конфигурацию 2/0 + 2/0. Левый и правый канал первой программы и левый и правый канал второй программы
fourChannels2-2	Четыре канала, использующие конфигурацию 2/2. Левый, правый, левый окружающий канал и правый окружающий канал
fourChannels3-1	Четыре канала, использующие конфигурацию 3/1. Левый, центральный, правый каналы и одиночный канал окружающего звука
fiveChannels3-0-2-0	Пять каналов, использующие конфигурацию 3/0 + 2/0. Левый, центральный и правый каналы первой программы и левый и правый канал второй программы
fiveChannels3-2	Пять каналов, использующие конфигурацию 3/2. Левый, центральный, правый, левый канал окружающего звука и правый канал окружающего звука.

Булевский параметр lowFrequencyEnhancement задает возможность улучшенного низкочастотного канала.

Если булевский параметр multilingual (многоязычный) принимает значение "истина", он свидетельствует о возможности поддержки до семи многоязычных каналов, а когда он принимает значение "ложь" – что ни один многоязычный канал не поддерживается.

Целый параметр bitRate показывает максимально возможную скорость передачи аудиосигнала, он измеряется в кбит/с.

GSMAudioCapability: Показывает возможности транскодирующих полноскоростных, полускоростных и улучшенных полноскоростных речевых аудиокодеков стандарта GSM. audioUnitSize показывает максимальное количество байтов, которые должны передаваться в каждом пакете, если параметр comfortNoise принимает значение "истина", то он задает способность поддержки обработки комфортного шума для канала передачи речи с полной скоростью, для канала передачи речи половинной скоростью или улучшенного для канала передачи речи с полной скоростью (GSM 06.12, GSM 06.22, GSM 06.62), а если параметр scrambled (скремблированный) принимает значение "истина", он показывает возможность поддержки побитового скремблирования для каналов передачи речи с полной скоростью, для каналов передачи речи с половинной скоростью или улучшенных каналов передачи речи с полной скоростью (GSM 06.10, GSM 06.20, GSM 06.60).

genericAudioCapability: Показывает универсальные возможности аудиосигнала.

g729Extensions: Показывает возможность дополнительных необязательных режимов, заданных в Рекомендации МСЭ-Т G.729. Эта кодовая точка не должна использоваться для индикации g729AnnexA, g729wAnnexB и g729AnnexAwAnnexB, для которых следует использовать g729AnnexA, g729AnnexB и g729AnnexAwAnnexB.

audioUnit:

- для системы мультиплексирования H.222 показывает размер буфера STD в единицах по 256 октетов;
- для системы мультиплексирования H.223 определяет максимальное количество аудиокадров на AL-SDU; а также
- для системы мультиплексирования H.225.0 определяет максимальное количество аудиокадров на пакет.

audioUnit должен присутствовать для Capability Exchange (обмена возможностями). Он может присутствовать для Mode Request (запроса режима).

Если параметр **annexA** принимает значение "истина", он показывает возможность передачи или приема аудиосигнала со скоростью 8 кбит/с в соответствии с Приложением A/G.729, а не в соответствии с основным текстом Рекомендации G.729.

Если параметр **annexB** принимает значение "истина", он свидетельствует о возможности подавления паузы согласно Приложению B/G.729.

Если параметр **annexD** принимает значение "истина", он указывает на возможность передачи или приема аудиосигнала, описанного в Приложении D/G.729, на скорости 6,4 кбит/с.

Если параметр **annexE** принимает значение "истина", он свидетельствует о возможности передавать или принимать аудиосигнал, соответствующий Приложению E/G.729, на скорости 11,8 кбит/с.

Если параметр **annexF** принимает значение "истина", он показывает возможность подавления паузы в соответствии с Приложением F/G.729.

Если параметр **annexG** принимает значение "истина", он показывает возможность подавления паузы согласно Приложению G/G.729.

Если параметр **annexH** принимает значение "истина", он показывает возможность перехода с работы на скорости 6,4 кбит/с (Приложение D/G.729) на скорость 11,8 кбит/с (Приложение E/G.729) и наоборот.

audioTelephonyEvent может включаться для индикации поддержки внутриволосных событий звуковой телефонии согласно документу RFC 2833. Поддерживаемые события должны быть описаны в audioTelephoneEvent, как описано в <списке значений> в разделе 3.9 документа RFC 2833. События с 0-го по 15-е (соответствующие цифрам DTMF 0-9, *, #, A, B, C, D) представляют собой все обязательные события.

audioTone может быть включен для индикации поддержки внутриволосных звуковых тонов согласно документу RFC 2833.

В.2.2.7 Возможности приложений передачи данных

Здесь показаны возможности обработки данных. Если в одной DataApplicationCapability ("возможность приложений передачи данных") задано нескольких возможностей, то это не означает, что они могут выполняться одновременно. На способность одновременной обработки возможностей может указывать появление DataApplicationCapability в различных наборах AlternativeCapabilitySets в рамках одного дескриптора CapabilityDescriptor.

Рекомендации, использующие настоящую Рекомендацию, могут накладывать ограничения на то, какие из этих режимов могут передаваться.

Для некоторых возможностей передачи данных требуются двунаправленные логические каналы, например для передачи протокола повторной передачи. Это требование включено в неявном виде в кодовые точки соответствующих возможностей.

DataApplicationCapability: Перечень приложений передачи данных и скоростей передачи. Каждое указанное приложение передачи данных должно поддерживаться одним или несколькими DataProtocolCapabilities ("возможностями протокола передачи данных").

maxBitRate определяет максимальную скорость передачи данных в единицах по 100 бит/с, с которой передатчик может передавать видеосигнал или получатель может принимать данные указанного приложения.

t120 свидетельствует о способности поддерживать протокол T.120 [32].

dsm-cc показывает возможность поддержки протокола DSM-CC [47].

userData ("данные пользователя") показывает возможность поддержки неопределенных данных пользователя из внешних портов передачи данных.

t84 показывает возможность поддержки передачи изображений типа T.84 [31] (JPEG, JBIG, факсимильные Группы 3/4).

t434 показывает возможность поддержки передачи телематических двоичных файлов T.434 [35].

h224 показывает возможность поддержки протокола H.224 [11] – симплексного протокола управления устройствами, работающими в реальном времени.

nlpid показывает возможность поддержки протокола сетевого уровня в соответствии с nlpidData, определенного в документе ИСО/МЭК TR 9577 [52]. Эти протоколы включают, помимо прочих, протокол маршрутизации в среде Интернет (IP) и протокол двухточечного соединения PPP IETF (PPP).

ПРИМЕЧАНИЕ. – Применение NLPID подробно описано в документе RFC 1490 "Multiprotocol Interconnect over Frame Relay" (Многопротокольное соединение поверх Frame Relay).

dsvdControl показывает, что терминал, работающий в цифровом стандарте одновременной передачи речи и данных (DSVD), может поддерживать канал управления с передачей вне основной полосы.

h222DataPartitioning показывает возможность поддержки модифицированного и ограниченного применения дробления данных H.262, определенного в Рекомендации МСЭ-Т H.222.1, при котором улучшенные данные передаются как канал данных, поддерживаемый посредством названной DataProtocolCapability ("возможности протокола данных").

t30fax: Эта кодовая точка указывает на возможность использовать аналоговый режим (G3V) Приложения C/T.30 в соответствии с определением в Рекомендации МСЭ-Т T.39 для режимов DSVF/MSVF.

t140: Эта кодовая точка показывает возможность использования протокола текстового диалога T.140, определенного в Рекомендации МСЭ-Т T.140.

t38fax: Эта кодовая точка указывает на протокол передачи данных, соответствующий Рекомендации МСЭ-Т T.38 [29].

Поля **version**, **t38FaxRateManagement**, **t38FaxUdpOptions** и **t38FaxTcpOptions** определены в Рекомендации МСЭ-Т T.38.

Если параметр fillBitRemoval принимает значение "истина", он означает, что шлюз/терминал может удалять и вставлять биты заполнения.

Если параметр transcodingJBIG принимает значение "истина", он указывает, что шлюз может осуществлять транскодирование в реальном времени между сжатием линии связи и JBIG для передачи по IP-сети.

Если параметр ranscodingMMG принимает значение "истина", он показывает, что шлюз может осуществлять транскодирование в реальном времени между сжатием линии связи и MMG для передачи по IP-сети.

genericDataCapability показывает универсальные возможности передачи данных. Если параметр maxBitRate включен в genericDataCapability, его значение должно быть таким же, как значение maxBitRate в DataApplicationCapability.

DataProtocolCapability: Содержит перечень протоколов передачи данных.

v14buffered показывает возможность поддержки указанного приложения передачи данных с помощью буферизированного V.14 [36].

v42larm показывает возможность поддержки указанного приложения передачи данных, используя протокол связи для модемов LAPM, определенный в Рекомендации МСЭ-Т V.42 [38].

hdlcFrameTunnelling показывает возможность поддержки указанного приложения передачи данных посредством туннелирования кадра HDLC. См. раздел 4.5.2 документа ИСО/МЭК 13239 [43].

h310SeparateVCStack показывает возможность поддержки указанного приложения передачи данных с помощью стека протоколов, определенного в Рекомендации МСЭ-Т H.310 для транспорта сообщений H.245 по отдельному виртуальному каналу АТМ к виртуальному каналу, используемому для аудиовизуальной связи.

h310SingleVCStack показывает возможность поддержки указанного приложения передачи данных с помощью стека протоколов, определенного в Рекомендации МСЭ-Т H.310 для транспорта сообщений H.245 по тому же виртуальному каналу АТМ, который используется для аудиовизуальной связи.

transparent (прозрачный) показывает возможность поддержки указанного приложения передачи данных с помощью "прозрачной" передачи данных.

v120: Применение v120 оставлено для дальнейшего изучения в Рекомендации МСЭ-Т Н.323.

separateLANStack показывает, что для транспортировки данных будет использоваться отдельный транспортный стек. Цель установления отдельного сетевого соединения для передачи данных обозначается параметром dataType ("тип данных") в OpenLogicalChannel, выраженным величинами h310SeparateVCStack или separateLANStack из DataProtocolCapability. Если выбранная возможность DataApplicationCapability представляет собой t120, этот выбор означает применение базового профиля T.123 для широкополосного ISDN (B-ISDN) и локальной сети (LAN), соответственно. Другие профили локальной сети могут выбираться с помощью характеристики non-Standard DataProtocolCapability ("характеристика нестандартного протокола передачи данных").

Если выбран параметр separateLANStack и в запросе OpenLogicalChannel присутствует separateStack, то получатель должен попытаться организовать указанный стек. В случае успешной попытки ответом будет подтверждение OpenLogicalChannelAck, в противном случае – отклонение OpenLogicalChannelReject с информацией о причине.

Если выбран параметр separateLANStack, а separateStack отсутствует в запросе OpenLogicalChannel, то получателю следует включить соответствующий separateStack в свой ответ OpenLogicalChannelAck. Затем получателю этого сообщения (первоначальной запрашивающей стороне) следует попытаться организовать указанный стек. В случае неудачи он выдаст сообщение CloseLogicalChannel.

Если выбран параметр separateLANStack и separateStack присутствует в запросе OpenLogicalChannel, то он может быть подменен параметром separateStack в ответе OpenLogicalChannelAck. Если первоначальная запрашивающая сторона не допускает подмены, она выдаст сообщение CloseLogicalChannel.

Если выбран параметр separateLANStack и separateStack отсутствует в запросе OpenLogicalChannel, а также он отсутствует в ответе OpenLogicalChannelAck, то первоначальная запрашивающая сторона может сделать вывод о том, что респондент не понимает эти расширения ACH.1 и ему следует сформировать сообщение CloseLogicalChannel для возврата ресурсов системе.

v76wCompression показывает возможность поддержки сжатия данных в канале передачи данных V.76.

tcp показывает возможность данного приложения поддерживать протокол управления передачей/Интернет-протокол (TCP/IP).

udp означает, что данное приложение может поддерживать протокол дейтаграмм пользователя (UDP).

T84Profile: Показывает типы профилей неподвижных изображений, которые могут быть поддержаны данным терминалом.

t84Unrestricted содержит указания на типы неподвижного изображения T.84, которые могут быть поддержаны данным терминалом: для определения того, может ли быть принято некоторое определенное изображение, следует использовать информацию на уровне T.84.

t84Restricted указывает тип неподвижного изображения T.84, который данный терминал может поддерживать.

qcif означает поддержку последовательного цветного изображения типа YCrCb с разрешением QCIF.

cif означает поддержку последовательного цветного изображения типа YCrCb с разрешением CIF.

ccir601Seq означает поддержку последовательного цветного изображения типа YCrCb с разрешением CCIR601.

ccir601Prog означает поддержку прогрессивного цветного изображения типа YCrCb с разрешением CCIR601.

hdtvSeq означает поддержку последовательного цветного изображения типа YCrCb с разрешением HDTV.

hdtvProg означает поддержку прогрессивного цветного изображения типа YCrCb с разрешением HDTV.

g3FacsMH200x100 свидетельствует о поддержке последовательного двухуровневого факсимильного изображения Группы 3, закодированного МН (модифицированным кодом Хаффмана) при нормальном разрешении (200 × 100 пикселей на дюйм).

g3FacsMH200x200 указывает на поддержку последовательного двухуровневого факсимильного изображения Группы 3, закодированного МН (модифицированным кодом Хаффмана) при высоком разрешении (200 × 200 пикселей на дюйм).

g4FacM200x100 показывает поддержку последовательного факсимильного двухуровневого изображения Группы 4, закодированного MMR (модифицированным усовершенствованным кодом Рида) при нормальном разрешении (200 × 100 пикселей на дюйм).

g4FacM200x200 указывает на поддержку последовательного факсимильного двухуровневого изображения Группы 4, закодированного MMR (модифицированным усовершенствованным кодом Рида) с высоким разрешением (200 × 200 пикселей на дюйм).

jb200x200Seq свидетельствует о поддержке последовательного двухуровневого изображения, кодированного в стандарте JBIG с разрешением 200 × 200 пикселей на дюйм.

jb200x200Prog свидетельствует о поддержке прогрессивного двухуровневого изображения, кодированного в стандарте JBIG с разрешением 200 × 200 пикселей на дюйм.

jb300x300Seq указывает на поддержку последовательного двухуровневого изображения, кодированного в стандарте JBIG с разрешением 300 × 300 пикселей на дюйм.

jb300x300Prog указывает на поддержку прогрессивного двухуровневого изображения, кодированного в стандарте JBIG с разрешением 300 × 300 пикселей на дюйм.

digPhotoLow свидетельствует о поддержке последовательного кодированного цветного изображения в стандарте JPEG с размером изображения до 720 × 576.

digPhotoMedSeq указывает на поддержку последовательного кодированного цветного изображения в стандарте JPEG с размером изображения до 1440 × 1152.

digPhotoMedProg указывает на поддержку прогрессивного кодированного цветного изображения в стандарте JPEG с размером изображения до 1440 × 1152.

digPhotoHighSeq свидетельствует о поддержке последовательного кодированного цветного изображения в стандарте JPEG с размером изображения до 2880 × 2304.

digPhotoHighProg свидетельствует о поддержке прогрессивного кодированного цветного изображения в стандарте JPEG с размером изображения до 2880 × 2304.

В.2.2.8 Возможности шифрования, аутентификации и целостности

Если присутствует EncryptionCapability ("возможности шифрования"), то она показывает возможности шифрования терминала для каждого типа среды, для которого такие возможности имеются. Сфера применения шифрования показывает, применяется ли данное шифрование по всему потоку битов, стандартным образом к части потока битов или нестандартным образом к части потока. Параметр algorithm (алгоритм) определяет алгоритм шифрования.

Если присутствует AuthenticationCapability ("возможность аутентификации"), то она показывает, что терминал поддерживает компоненты аутентификации, определенные в Рекомендации МСЭ-Т Н.235 [16]. antiSpamAlgorithm ("алгоритм противодействия спаму") задает метод и алгоритм, используемый для обеспечения контрмер против атак типа "затопление" и "отказ в обслуживании".

Если присутствует IntegrityCapability ("возможность проверки целостности"), то она означает, что терминал поддерживает компоненты целостности, определенные в Рекомендации МСЭ-Т Н.235 [16].

В.2.2.9 Возможности конференц-связи

ConferenceCapability ("возможности конференц-связи") задает такие характеристики организации конференции, как способность поддержки управления со стороны председателя (Chair Control), описанную в Рекомендации МСЭ-Т Н.243.

Возможность videoIndicateMixingCapability должна быть определена как передача сообщения, независимая от поставщиков (VIM) Н.230.

В.2.2.10 Возможности ввода данных пользователя

Возможность UserInputCapabilities ("возможность ввода данных пользователем") указывает, какие параметры в сообщении UserInputIndication поддерживаются терминалом. BasicString сообщает, что терминал поддерживает опцию basicString индикации userInputSupportIndication, iA5String означает, что терминал поддерживает опцию iA5String индикации userInputSupportIndication, а generalString указывает, что терминал поддерживает опцию generalString индикации userInputSupportIndication.

Dtmf показывает, что терминал поддерживает DTMF, используя компоненты сигнала и обновления сигнала сообщения userInputIndication. Hookflash означает, что терминал поддерживает сигнал отбоя с помощью компонентов сигнала и обновления сигнала сообщения userInputIndication.

Для безопасной сигнализации DTMF параметры UserInputCapabilities показывают, какие параметры в сообщении UserInputIndication закодированы.

encryptedBasicString показывает, что терминал поддерживает опцию encryptedalphanumeric индикации UserInputIndication.

encryptedIA5String показывает, что терминал поддерживает опцию encryptedSignalType индикации UserInputIndication.

encryptedGeneralString показывает, что терминал поддерживает опцию encryptedalphanumeric extendedAlphanumeric индикации UserInputIndication.

secureDTMF показывает, что терминал поддерживает тип encryptedSignalType в сигнале для безопасной передачи DTMF.

В.2.2.11 Универсальные возможности

Тип **GenericCapability** ("универсальная возможность") позволяет задавать новые возможности таким способом, чтобы не нужно было выпускать новую версию синтаксиса H.245. Благодаря этому универсальному средству определения возможностей такие сетевые устройства, как объекты многоточечного управления (МС) могут устанавливать самый высокий общий режим работы, не имея подробных сведений об используемых возможностях. Он позволяет определять описания возможностей, как основанные на Рекомендациях МСЭ-Т, так и на других стандартах (включая описания собственных возможностей). Описания возможностей на основе стандартов МСЭ-Т следует включать как дополнения в настоящую Рекомендацию H.245. Описания возможностей, не основанные на стандартах МСЭ-Т, можно публиковать в любой подходящей форме.

Поле **capabilityIdentifier** показывает тип определяемой возможности. Описания возможности на основе документов МСЭ-Т должны использовать **стандартный** идентификатор объекта OBJECT IDENTIFIER, тогда как описания на базе других стандартов и описания собственных возможностей должны использовать один соответствующий идентификатор из следующих идентификаторов объекта: **standard** (стандартный), **h221NonStandard**, **uuid** и **domainBased**.

Поле **subIdentifier** обозначает тип или набор параметров, относящихся к идентификатору **capabilityIdentifier**.

maxBitRate задает максимальную скорость, с которой может выполняться данная возможность при обмене возможностями, и фактическую скорость передачи данных, которая должна использоваться при передаче сигналов по открытому логическому каналу. Она должна присутствовать всякий раз, когда может быть указана значащая величина и когда она разрешена по спецификации описания конкретной возможности. Она определяется отдельно, чтобы посреднические узлы в тракте сигнализации могли знать используемую пропускную способность, не имея подробных сведений о каждой возможности.

Параметры возможности могут быть описаны в виде произвольной комбинации полей **collapsing**, **nonCollapsing**, и **nonCollapsingRaw**, вместе с **transport**, как определено в описании возможности.

В поле **collapsing** ("свертываемые") указаны возможности, которые описаны таким образом, чтобы МС мог объединять возможности нескольких конечных точек и создавать общий набор возможностей, используя простой набор правил без подробных сведений об отдельном кодеке.

В поле **nonCollapsing** возможности заданы с использованием того же синтаксиса, что в поле **collapsing**, но эти возможности не могут обрабатываться объектом многоточечного управления. В этом случае семантика **ParameterValue** меняется и указывает только величины, но не правила свертывания. Например, **unsignedMin** и **unsignedMax** имеют одну и ту же семантику и просто обозначают 16-битовый целый параметр.

Поле **nonCollapsingRaw** задает возможности посредством строки октетов (OCTET STRING). Обычно она состоит из закодированной PER структуры данных ASN.1. Обратите внимание на то, что объект многоточечного управления должен иметь специальные сведения о возможностях, описанных этим способом, чтобы воспользоваться ими.

В поле **transport** (транспорт) показаны транспортные параметры, относящиеся к описываемой возможности.

При задании описаний возможностей рекомендуется определить как можно больше таких параметров как **collapsing**, поскольку только определенные этим способом параметры гарантированно обрабатываются, а не просто пересылаются элементами сети.

Универсальные возможности **GenericCapabilities**, которые включают как **collapsing** (свертываемые), так и **nonCollapsing** (несвертываемые) последовательности, не должны включать структуры **GenericParameter** различных типов (**collapsing**, **nonCollapsing**), которые используют один идентификатор параметров **parameterIdentifier**.

ПРИМЕЧАНИЕ 1. – Такое многократное использование одного **parameterIdentifier** может привести к конфликту между значениями **parameterIdentifier**, если этот параметр автоматически передается системе, например системе H.320, которая не различает **collapsing** и **nonCollapsing** параметры.

Стандартному полю **parameterIdentifier** **GenericParameter** не должно назначаться значение 0.

ПРИМЕЧАНИЕ 2. – Назначение значения 0 может вызвать конфликт с автоматическим переводом в сигнализацию H.320, например согласно Приложению A/H.239 и Рекомендации МСЭ-Т H.241.

GenericParameter ("универсальный параметр") показывает один параметр возможности или одну группу параметров возможности.

parameterIdentifier ("идентификатор параметра") позволяет указывать значения стандартных параметров (то есть значения, определенные в описании возможности) и значения собственных параметров. Параметры, определенные в описании возможности, используют форму **standard**, которая идентифицирует параметр с помощью целого числа. Параметры, которые являются собственными расширениями, используют формы **h221NonStandard**, **uuid** или **domainBased**.

В поле **parameterValue** указана величина параметра. Присутствие параметра **logical** (логический) свидетельствует о том, что конечная точка поддерживает представляемую параметром опцию. Поле **booleanArray** содержит до восьми независимых булевых переменных. Поля **unsignedMin** и **unsignedMax** задают параметр, используя 16-битовое целое число без знака. Поля **unsigned32Min** и **unsigned32Max** задают параметр, используя 32-разрядное целое число без знака. В поле **octetString** параметр задается в виде строки октетов (OCTET STRING). В поле **genericParameter** указана последовательность параметров, которые сгруппированы вместе на данном уровне иерархии возможностей.

Для объединения описаний возможностей, полученных от нескольких конечных точек, в общее описание такой возможности, относительно которой у МС нет встроенных сведений, МС следует сначала проигнорировать любые параметры, не поддерживаемые всеми конечными точками, которые по решению МС являются кандидатами на использование конкретной возможности. Затем для каждого параметра конечных точек-кандидатов с одним и тем же идентификатором параметра **parameterIdentifier** МС должен:

- выполнить операцию "логическое И" в случае **booleanArray**;
- выбрать минимальное значение в случае **unsignedMin** или **unsigned32Min**; и
- выбрать максимальное значение в случае **unsignedMax** или **unsigned32Max**.

Поле **supersedes** (заменяет) позволяет описанию возможности содержать группу параметров, из которых при определении общего описания возможности следует выбрать только один параметр. Например, его можно использовать для видеокodeка, который поддерживает разрешения SQCIF, QCIF и CIF с различными минимальными интервалами изображений. Величина в **parameterIdentifier** относится к параметру того же уровня вложенности. Несколько полей **supersedes** включаются с параметром для того, чтобы дерево зависимостей параметров могло быть выражено так, как оно дано в описании возможности H.262. Каждый из параметров, определенных в поле **supersedes**, следует отбрасывать из общего описания возможности. Параметры, которые заменяют отброшенные параметры, также должны в свою очередь отбрасываться, и этот процесс должен повторяться до тех пор, пока не будут отброшены все замененные параметры.

Результатом этой операции является общее описание возможности.

ПРИМЕЧАНИЕ 3. – МС, имеющий встроенные сведения об описании некоторой конкретной возможности, может использовать свой собственный набор правил для создания общего описания возможности.

В.2.2.12 Возможности мультиплексированного потока

Возможность `multiplexedStreamCapability` ("возможность мультиплексированного потока") указывает на возможность поддержки мультиплексированного потока по одиночному логическому каналу.

`multiplexFormat` показывает поддерживаемый протокол мультиплексирования.

Если параметр `controlOnMuxStream` принимает значение "истина", он показывает поддержку сигнализации логического канала для мультиплексного потока при помощи канала управления, передаваемого по мультиплексному потоку. Если он принимает значение "ложь", то сигнализация логического канала для мультиплексного потока поддерживается с помощью этого канала управления H.245. Если `controlOnMuxStream` принимает значение "ложь", а `multiplexFormat` – это `H223Capability`, то должен быть открыт максимум один логический канал для мультиплексированного потока H.223. `controlOnMuxStream` должен принимать значение "ложь", если `MultiplexFormat` установлен равным `h223Capability`.

Если присутствует параметр `capabilityOnMuxStream`, он отображает набор возможностей мультиплексированного потока. Эти возможности указаны посредством набора `AlternativeCapabilitySet`. Этот набор `AlternativeCapabilitySet` не должен включать возможность `multiplexedStreamTransmission` ("передача мультиплексированного потока"). При отсутствии этого параметра должен производиться обмен набором возможностей с помощью канала управления, который передается посредством мультиплексированного потока после того, как открылся логический канал мультиплексированного потока.

В.2.2.13 Полезная нагрузка RTP для события аудиотелефонии и возможности звукового тона

Возможность `receiveRTPAudioTelephonyEventCapability` может быть включена для того, чтобы показать поддержку событий аудиотелефонии внутри полосы, соответствующих документу RFC 2833. Параметр `dynamicRTPPayloadType` показывает, какой динамический тип полезной нагрузки RTP должен использоваться для передачи этих событий. Поддерживаемые события должны быть описаны в `audioTelephoneEvent` согласно описанию в <list of value> ("списке значений") в разделе 3.9 документа RFC 2833. События 0-15 (соответствующие цифрам DTMF 0-9 и символам *, #, A, B, C, D) являются всеми обязательными событиями.

Возможность `receiveRTPAudioToneCapability` может быть включена для того, чтобы показать поддержку звуковых тонов внутри полосы, соответствующих документу RFC 2833. Параметр `dynamicRTPPayloadType` показывает, какой тип динамический полезной нагрузки RTP должен использоваться для передачи этих тонов.

В.2.2.14 Поток с несколькими полезными нагрузками

Поток с несколькими полезными нагрузками (MPS) содержит пакеты, представляющие один логический поток среды, то есть все пакеты представляют собой кодировку одного потока для указанных временных интервалов. Для обеспечения возможности идентификации и корреляции различных используемых кодировок все пакеты в одном MPS ДОЛЖНЫ передавать идентификаторы типа полезной нагрузки в одном и том же месте в пакете, и они ДОЛЖНЫ использовать отметки времени в одном формате, полученные от одного источника синхронизации (например, полезная нагрузка RTP должна использовать одни SSRC). В большинстве случаев эти пакеты представляют собой последовательные, непересекающиеся интервалы времени, и они просто выбирают различные кодировки для различных интервалов, но существуют случаи, в которых различные дополнительные кодировки представляют перекрывающиеся интервалы, например, если в середине интервала кодирования происходит событие, которое должно быть явно закодировано с использованием дополнительной кодировки. Например, это может иметь место, если тон DTMF обнаружен в середине интервала кодирования звука, и он должен передаваться с помощью телефонного события согласно документу RFC 2833. В этом случае отметка времени в пакете телефонного события должна соответствовать времени в середине интервала кодирования звука. Если представленное потоковое событие не имеет измеряемой продолжительности, то допускается использование пакетов с нулевой продолжительностью. Допускается также использование положений документа RFC 2198 для многократной передачи пакета, когда данные пакуются в пакет с другими типами полезной нагрузки и временными интервалами.

ПРИМЕЧАНИЕ. – Поскольку все пакеты должны представлять кодировки одного источника (адресата), в поток не допускается включать различные типы сред, например аудиосигналы и видеосигналы, но пакеты с данными, представляющие данные, полученные из медиапотока (например, цифры DTMF, обнаруженные в звуковом потоке) могут иметь альтернативное представление или кодировку.

В.2.2.15 Прямая коррекция ошибок

Конечная точка может объявлять способность прямой коррекции ошибок. При объявлении поддержки документа RFC 2733 конечная точка имеет возможность сообщить, что данные FEC могут передаваться в отдельном потоке или в том же самом потоке (используя избыточное кодирование) согласно документу RFC 2198. Эта возможность позволяет конечной точке показать (используя номер элемента таблицы возможностей), какие кодеки могут использоваться в потоке FEC.

Если конечная точка, передающая сообщение **OpenLogicalChannel**, желает использовать положения документа RFC 2198 для передачи данных FEC (и эта возможность поддерживается получателем), она должна использовать кодирование **Data Type redundancyEncoding**, включая, например, кодирование VBD как **primary** (первичное) кодирование, а **Data Type fec** – как **secondary** (вторичное) кодирование. Тип полезной нагрузки для пакетов RFC 2198 должен быть определен в поле **dynamicPayloadType** сообщения **OpenLogicalChannel**. Тип полезной нагрузки для первичного (**primary**) кодирования и данных FEC можно передать в поле **payloadType** полей **primary** и **secondary RedundancyEncodingElement**.

Если оконечная точка желает передавать данные FEC по отдельному потоку, она имеет две возможности: передавать на тот же порт, что и защищенные FEC данные, или на другой порт. При передаче на другой порт для потока FEC она должна явно использовать отдельное сообщение **OpenLogicalChannel**. Выбранный тип данных **data Type** должен быть равен **fec**, он не должен содержаться в поле **redundancyEncoding**. Необходимо выбрать порт **mode.separateStream.differentPort** и включить идентификатор сеанса защищенного потока и дополнительно тип полезной нагрузки защищенной среды в том случае, когда по подчиненному каналу передается несколько типов полезной нагрузки, такие как поток MPS. При передаче в отдельном потоке, но на тот же порт, что и защищенная среда, данные FEC должны передаваться как часть потока MPS. В этом случае одним элементом потока MPS должен быть **protected audio** (защищенный звук), и одним элементом должен быть **fec**. В этом случае следует выбрать **mode.separateStream.samePort** и передавать тип полезной нагрузки защищенного потока.

В.2.3 Подтверждение набора возможностей терминала

Это сообщение используется для подтверждения **TerminalCapabilitySet** ("набора возможностей приема") от равноправного объекта CESE.

Параметр **sequenceNumber** должен совпадать с **sequenceNumber** в наборе возможностей **TerminalCapabilitySet**, для которого это сообщение служит подтверждением.

В.2.4 Отклонение набора возможностей терминала

Это сообщение используется для отказа от **TerminalCapabilitySet**, поступившего от равноправного CESE.

Параметр **sequenceNumber** должен совпадать с **sequenceNumber** в наборе возможностей **TerminalCapabilitySet**, для которого это сообщение является отрицательным квитированием.

Причины передачи этого сообщения приведены в таблице В.6.

Таблица В.6/Н.245 – Причины для отказа от **TerminalCapabilitySet**

Кодовая точка АСН.1	Причина
Unspecified (не указана)	Причина отказа не указана.
undefinedTableEntryUsed	Дескриптор возможности содержит ссылку на неопределенный элемент таблицы capabilityTable .
descriptorCapacityExceeded	Терминал не смог записать всю информацию в TerminalCapabilitySet .
tableEntryCapacityExceeded	Терминал не смог записать больше элементов, чем указано в highestEntryNumberProcessed , или не смог записать никакие элементы.

В.2.5 Сброс набора возможностей терминала

Это сообщение передается в случае тайм-аута.

В.3 Сообщения сигнализации логического канала

Данный набор сообщений предназначен для сигнализации логического канала. Один и тот же набор сообщений используется для сигнализации однонаправленного и двунаправленного логического канала; однако некоторые параметры присутствуют только в случае сигнализации двунаправленного логического канала.

Термин "forward" (прямой) используется для описания передачи в направлении от терминала, совершающего первоначальный запрос логического канала на другой терминал, а термин "reverse" (обратный) используется для описания противоположного направления передачи в случае запроса двунаправленного канала.

В.3.1 Открыть логический канал

Это сообщение используется для попытки открыть однонаправленный логический канал между передающим LCSE и равноправным принимающим LCSE и для попытки открыть двунаправленный логический канал между передающим B-LCSE и равноправным принимающим B-LCSE.

forwardLogicalChannelNumber: Указывает номер логического канала для прямого логического канала, который должен быть открыт.

forwardLogicalChannelParameters: Включает параметры, относящиеся к логическому каналу в случае попытки открытия однонаправленного канала, и параметры, относящиеся к прямому логическому каналу в случае попытки открытия двунаправленного канала.

reverseLogicalChannelParameters: Включает параметры, относящиеся к обратному логическому каналу в случае попытки открытия двунаправленного канала. Его наличие указывает, что совершается запрос об организации двунаправленного логического канала с заданными параметрами, а его отсутствие означает, что сделан запрос об организации однонаправленного логического канала.

ПРИМЕЧАНИЕ. – Параметры H.222 не включены в reverseLogicalChannelParameters, поскольку их значения не известны терминалу, который инициирует запрос.

portNumber – это межпользовательский параметр, который может использоваться пользователем для назначения логическому каналу входного или выходного порта или номера канала на более высоком уровне.

dataType показывает данные, которые должны передаваться по логическому каналу.

Если при этом указан тип nullData, то логический канал не будет использоваться для транспорта данных элементарных потоков, а только для передачи информации уровня адаптации. Если видеосигнал должен передаваться только в одном направлении, но должен применяться протокол повторной передачи, такой как AL3, определенный в документе H.223, то необходим обратный канал для передачи запросов повторной передачи; он может также использоваться для описания логического канала, который содержит только значения PCR в случае транспортных потоков H.222.1 [9].

dataType для среды h235Media используется для определения кодирования логического канала; фактический используемый тип данных задается в H235Media наряду со спецификацией кодирования.

Терминалы, поддерживающие только однонаправленную работу (на передачу или прием) с типами среды для использования двунаправленных каналов, должны передавать возможности только для поддерживаемого направления работы. В обратном направлении должен применяться тип nullData, для которого никакие возможности не требуются. Терминалам, работающим только на передачу, следует передавать возможности передачи, но терминалам не следует предполагать, что отсутствие возможностей передачи подразумевает невозможность работы только на передачу.

separateStack показывает, что для передачи данных будет применяться отдельный транспортный стек, он предоставляет адрес, используемый при организации стека, который является адресом Q.2931, E.164 или транспортным адресом локальной сети.

networkAccessParameters определяет распределение, сетевой адрес, а также информацию для создания и присваивания, которая используется для separateStack.

distribution ("распределение") должно присутствовать, если networkAddress задан равным localAreaNetwork, он должен показывать, является ли networkAddress транспортным адресом одного устройства или групповым транспортным адресом.

networkAddress показывает адрес фактически используемого стека: адрес Q.2931, E.164 или транспортный адрес локальной сети.

associateConference показывает, является ли конференция с передачей данных новой конференцией (associateConference=FALSE) или это уже существующая конференция, которую следует ассоциировать с вызовом для передачи аудиосигнала/видеосигнала (associateConference=TRUE).

externalReference показывает информацию, которая может использоваться для предоставления дальнейших ассоциаций или информации относительно данного separateStack.

Если при этом используется тип VideoCapability, AudioCapability, то логический канал может использоваться для любого из вариантов, указанных каждой отдельной возможностью; он должен быть в состоянии переходить с одного варианта на другой и обратно только при помощи передачи сигналов внутри полосы логического канала; например, в случае видеосигнала H.261, если указаны и QCIF, и CIF, то должна поддерживаться возможность переходить с одного на другой при каждом изображении. В случае использования DataApplicationCapability может быть задан только один экземпляр этой возможности, поскольку отсутствует внутриволновая сигнализация, позволяющая осуществлять переход с одного варианта на другой.

Если используется encryptionData, то логический канал будет использоваться для транспорта информации шифрования в соответствии с определением.

Если используется multiplexedStream, то логический канал будет использоваться для транспорта аудиосигнала/видеосигнала/данных в виде мультиплексированного потока согласно определению. Поля MultiplexedStreamParameter имеют те же значения, что и поля с таким же названием в MultiplexedStreamCapability.

forwardLogicalChannelDependency показывает номер логического номера, от которого зависит прямой канал, который должен быть открыт.

reverseLogicalChannelDependency задает номер логического номера, от которого зависит обратный канал, который должен быть открыт.

Параметр replacementFor указывает, что логический канал, который должен быть открыт, будет заменой указанного существующего уже открытого логического канала. Этот параметр должен использоваться только для обозначения логических каналов, уже находящихся в состоянии ESTABLISHED ("организован"). По логическим каналам, открытым с использованием этого параметра, не должен передаваться никакой трафик данных до тех пор, пока не прекратится весь трафик по упомянутым организованным логическим каналам. В этом случае никогда не будет требоваться, чтобы медиатекодеры декодировали трафик данных от обоих логических каналов одновременно. После начала передачи трафика по новому организованному логическому каналу старый логический канал должен немедленно закрываться. Получатели могут подтверждать открытие логических каналов, используя механизм replacementFor с условием, что старый и новый логические каналы не должны использоваться одновременно, и поэтому не будут превышены возможности декодирования у получателя.

Поле encryptionSync должно использоваться ведущим терминалом для предоставления значения ключа шифрования и точки синхронизации, в которой этот ключ следует применять. Для H.323 флаг syncFlag должен устанавливаться равным номеру динамической полезной нагрузки RTP, который соответствует данному ключу.

H222LogicalChannelParameters: Используется для индикации параметров, относящихся именно для применения Рекомендацией МСЭ-Т H.222.1 [9]. Он должен присутствовать в параметрах прямого логического канала forwardLogicalChannelParameters и не должен присутствовать в параметрах обратного логического канала reverseLogicalChannelParameters.

resourceID указывает, по какому виртуальному каналу АТМ должен транспортироваться данный логический канал. В настоящей Рекомендации не определяются средства, с помощью которых этот параметр ассоциируется с виртуальным каналом АТМ. Если Рекомендация МСЭ-Т H.222.0 используется в Рекомендации МСЭ-Т H.323 как формат мультиплексированного потока, то этот параметр содержит логический номер канала мультиплексированного потока, в который должен мультиплексироваться данный логический канал.

subChannelID указывает, какой подканал H.222.1 используется для логического канала. Он должен быть равен PID в транспортном потоке и stream_id в программном потоке.

pcr-pid указывает PID, используемый для транспорта временных отметок программы при использовании транспортного потока. Он должен присутствовать, когда по виртуальному каналу АТМ передается транспортный поток, и должен отсутствовать, когда по виртуальному каналу АТМ передается программный поток.

Если присутствует необязательная строка октетов programDescriptors, то она содержит один или несколько дескрипторов, определенных в Рекомендациях МСЭ-Т Н.222.0 и Н.222.1, которые описывают ту программу, частью которой является информация, подлежащая передаче по логическому каналу.

Если присутствует необязательная строка октетов streamDescriptors, то она содержит один или несколько дескрипторов, заданных в Рекомендациях МСЭ-Т Н.222.0 и Н.222.1, которые описывают информацию, подлежащую передаче по логическому каналу.

H223LogicalChannelParameters: Используется для индикации параметров, относящихся только к применению Рекомендации МСЭ-Т Н.223 [10]. Он должен присутствовать в forwardLogicalChannelParameters и reverseLogicalChannelParameters.

adaptationLayerType указывает, какой уровень адаптации и какие опции будут применяться для логического канала. Используются следующие кодовые точки: nonStandard (нестандартный), al1Framed (кадрированный режим AL1), al1NotFramed (некадрированный режим AL1), al2WithoutSequenceNumbers (AL2 без порядковых номеров), al2WithSequenceNumbers (AL2 с порядковыми номерами) и al3 (AL3 с индикацией количества октетов управляющего поля, которые будут присутствовать, и размера пересылочного буфера B_s , который будет использоваться, при этом размер определяется в октетах), al1M (AL1M, определенный в Приложении С/Н.223 с заданными параметрами), al2M (AL2M, определенный в Приложении С/Н.223 с заданными параметрами) или al3M (AL3M, определенный в Приложении С/Н.223 с заданными параметрами).

Если параметр segmentableFlag принимает значение "истина", то он показывает, что канал определен как сегментируемый, а когда он принимает значение "ложь", он показывает, что канал определен как несегментируемый.

H223AL1MParameters: Используется для индикации параметров, определенных для применения уровня адаптации AL1M.

transferMode показывает, используется ли кадрированный или некадрированный режим.

headerFEC указывает, является ли прямая коррекция ошибок (FEC) SEBCH (16,7) или Golay (24,12).

Длина поля циклической суммы (CRC) для полезной нагрузки в битах задается посредством crcLength как 4, 8, 12, 16, 20, 28 или 32 битов или посредством crcNotUsed.

rsCodeRate задает кодовую скорость RCPC как 8/8, 8/9..., 8/32.

arqType указывает режим работы автоматического запроса ARQ: noARQ свидетельствует об отсутствии повторной передачи, typeIArq означает тип I ARQ, а typeIIArq означает тип II ARQ.

Если параметр TalpduInterleaving принимает значение "истина", то он свидетельствует об использовании интерливинга AL-PDU.

Если параметр alsduSplitting принимает значение "истина", то он указывает на использование режима разбиения AL-SDU.

rsCodeCorrection обозначает возможность коррекции кода RS в виде 0, 1, ..., 127 октетов. К каждому полю с переменной длиной AL-SDU и CRC добавляется фиксированное количество символов четности кода RS (октетов), соответствующее rsCodeCorrection. При использовании кодирования RS, typeIIArq и alpduInterleaving не поддерживаются.

H223AL2MParameters: Используется для индикации параметров, относящихся именно к применению уровня адаптации AL2M.

headerFEC показывает, является ли прямая коррекция ошибок (FEC) SEBCH (16,5) или Golay (24,12).

Если параметр alpduInterleaving принимает значение "истина", он свидетельствует об использовании интерливинга AL-PDU.

H223AL3MParameters: Используется для индикации параметров, относящихся именно к использованию уровня адаптации AL3M.

Имеет те же параметры, что и AL1MParameters, за исключением отсутствующих параметров transferMode и alsduSplitting.

H223AnnexCArqParameters

numberOfRetransmissions задает максимальное количество повторных передач, которые могут использоваться: finite (конечный) показывает конечный предел числа повторных передач, которые могут использоваться, в диапазоне от 0 до 16; а infinite (бесконечный) показывает, что нет ограничения на число повторных передач, которые могут использоваться. numberOfRetransmissions, равный конечной величине 0, показывает, что поле управления используется для режима разбиения, но что повторные передачи не используются.

sendBufferSize задает размер пересылочного буфера, который будет использоваться, его размер измеряется в октетах.

V76LogicalChannelParameters: Используется для индикации параметров, относящихся к применению именно Рекомендации МСЭ-Т V.76.

audioHeader используется для индикации использования заголовка аудиоинформации в логическом канале. Это допустимый параметр для каналов передачи аудиосигнала DataType.

suspendResume используется для индикации того, что в канале могут использоваться процедуры suspend/resume (приостановки/возобновления) для приостановки работы других логических каналов. Можно выбрать три варианта работы канала; но suspend resume (отсутствие приостановки/возобновления) в канале, приостановки/возобновления с использованием адреса, или приостановка/возобновление без адреса в соответствии с определением Рекомендации МСЭ-Т V.76. suspendResumewAddress показывает, что канал с приостановкой/возобновлением должен использовать поле адреса, заданное в Рекомендации МСЭ-Т V.76. suspendResumewoAddress означает, что канал с приостановкой/возобновлением не должен использовать поле адреса.

eRM свидетельствует о том, что логический канал должен выполнять процедуры исправления ошибок, определенные в Рекомендации МСЭ-Т V.76.

uNRM указывает, что логический канал должен использоваться в режиме без исправления ошибок, определенном в Рекомендации МСЭ-Т V.76.

Описание параметров n401, windowSize и loopbackTestProcedure приведено в разделе 12.2.1/V.42 и положениях указанной Рекомендации. Для решения задач, сформулированных в Рекомендации МСЭ-Т V.70, параметр n401 должен задаваться в октетах.

crcLength – это необязательный параметр, который указывает длину CRC, используемую в режиме исправления ошибок. Если этот параметр отсутствует, то должна использоваться задаваемая по умолчанию длина CRC. crc8bit свидетельствует об использовании 8-битовой CRC, crc16bit соответствует применению 16-битовой CRC, а crc32bit показывает использование 32-битовой CRC, определенной в Рекомендации МСЭ-Т V.76.

recovery (восстановление) – это необязательный параметр, который означает использование процедур исправления ошибок, определенных в Рекомендации МСЭ-Т V.76. Если этот параметр отсутствует, то должна применяться заданная по умолчанию процедура исправления ошибок. sREJ означает использование процедуры селективной отбраковки кадров, а mSREJ означает использование процедуры множественной селективной отбраковки кадров, определенной в Рекомендации МСЭ-Т V.76.

uH указывает на использование кадров UH по V.76.

rej означает использование процедуры отбраковки, описанной в Рекомендации МСЭ-Т V.76.

V75Parameters используется для индикации специального параметра, используемого в Рекомендации МСЭ-Т V.75. audioHeaderPresent указывает наличие заголовка аудиоинформации V.75.

H2250LogicalChannelParameters: Используется для индикации параметров, относящихся именно к применению Рекомендации МСЭ-Т H.225.0. Должен присутствовать в forwardLogicalChannelParameters и reverseLogicalChannelParameters.

sessionId – это уникальный идентификатор сеанса RTP или T.120 в конференции. Он используется отправителем для обозначения сеанса, к которому относится логический канал. Идентификатор сеанса может быть создан только ведущим терминалом. Условно считается, что имеются три основных сеанса. Первый основной сеанс с идентификатором сеанса 1 – это сеанс передачи аудиоинформации, второй основной сеанс с идентификатором сеанса 2 – это сеанс передачи видеоинформации, а третий основной сеанс с идентификатором сеанса 3 – это сеанс передачи данных. Ведомый объект может открыть дополнительный сеанс, предоставляя идентификатор сеанса 0 в сообщении openLogicalChannel. Ведущий объект создаст уникальный идентификатор сеанса и передаст его в сообщении openLogicalChannelAck.

Идентификатор `associatedSessionID` используется для того, чтобы ассоциировать один сеанс с другим. В типичном случае применения аудиосеанс ассоциируется с видеосеансом для того, чтобы указать, какие сеансы нужно обрабатывать при последующей синхронизации.

Параметр `mediaChannel` указывает транспортный адрес `transportAddress`, который должен использоваться для логического канала. Если транспорт адресован одному устройству, `mediaChannel` отсутствует в параметрах `forwardLogicalChannelParameters` сообщения `OpenLogicalChannel`, но он может присутствовать в `reverseLogicChannelParameters`. Если `transportAddress` является многоадресным, то ведущий терминал отвечает за создание группового транспортного адреса, и он должен включать это адрес в сообщение `OpenLogicalChannel`. Ведомый терминал, который желает открыть новый многоадресный канал, должен задавать нули в поле многоадресного `transportAddress`. Ведущий терминал создаст групповой адрес `transportAddress` и передаст его ведомому терминалу в сообщении `OpenLogicalChannelAck`. Обратите внимание на то, что МС будет использовать `communicationModeCommand` для определения детальной информации для всех сеансов RTP в конференции.

Параметр `mediaChannel` служит для описания транспортного адреса для логического канала. Адреса IPv4 и IPv6 должны кодироваться таким образом, чтобы самым старшим октетом адреса был первый октет в соответствующей строке октетов (OCTET STRING), например, адрес IPv4 класса B 130.1.2.97 должен иметь "130" закодированным в первом октете строки октетов, за которым идет "1", и так далее. В адресе IPv6 a148:2:3:4:a:b:c:d "a1" должен быть закодирован в первом октете, "48" – во втором, "00" – в третьем, "02" – в четвертом и так далее. Адреса IPX, узел, сетевой номер `netnum` и порт должны кодироваться так, чтобы самым старшим октетом каждого поля был первый октет в соответствующей строке октетов.

`mediaGuaranteedDelivery` показывает, следует или не следует выбирать базовый медиатранспорт для обеспечения или отсутствия обеспечения гарантируемой доставки данных.

`mediaControlChannel` показывает канал управления средой, по которому отправитель информации по открытому логическому каналу будет прослушивать сообщения управления средой для этого сеанса. Это поле присутствует только в том случае, когда требуется канал управления средой.

`mediaControlGuaranteedDelivery` показывает, следует или не следует выбирать базовый транспорт управления средой для обеспечения или отсутствия обеспечения гарантируемой передачи данных. Это поле присутствует только в том случае, когда требуется канал управления средой.

Параметр `silenceSuppression` используется для индикации того, перестает ли отправитель передавать пакеты во время паузы. Этот параметр должен включаться в сообщение `openLogicalChannel` для аудиоканала и должен опускаться для канала любого другого типа.

`destination` (адресат) показывает `terminalLabel` ("метку терминала") адресата, если такая метка терминала была присвоена.

`dynamicRTPPayloadType` задает значение динамической полезной нагрузки. Если это поле используется, то `RTPPayloadType.payloadType` и значение этого поля должны совпадать.

`mediaPacketization` показывает, какая дополнительная схема формирования пакетов среды используется.

`redundancyEncoding` означает, что для открываемого логического канала должен использоваться метод избыточного кодирования, указанный в настоящем параметре. Первичное кодирование определяется посредством параметра `dataType` ("тип данных") `forwardLogicalChannelParameters` или `reverseLogicalChannelParameters` соответственно. Тип избыточного кодирования, который должен использоваться в этом логическом канале, идентифицируется параметром `redundancyEncodingMethod`, вторичное кодирование задается параметром `secondaryEncoding`. Параметры `DataType` (аудиосигнал, видеосигнал и т. д.), выбранные для первичного и для вторичного кодирования, должны соответствовать друг другу и выбранному `redundancyEncodingMethod`. Параметр источника используется для определения номера терминала отправителя сообщения `OpenLogicalChannel`.

Открытие канала, защищенного избыточным кодированием согласно RFC 2198, достигается путем использования **`dataType.redundancyEncoding`**. Это поле позволяет передавать первичный тип данных и несколько **`secondary`** (вторичных) типов данных. Кроме того, оно позволяет использовать RFC 2198 с "несколькими потоками полезной нагрузки" и с прямой коррекцией ошибок.

При открытии логического канала тип полезной нагрузки RTP для пакетов RFC 2198 определяется полем **dynamicPayloadType** в **OpenLogicalChannel** или полем **payloadType** в структуре **multiplePayloadStreamElement**. Типы полезной нагрузки для первичного типа полезной нагрузки и вторичных типов полезной нагрузки определены в структуре **RedundancyEncodingElement** наряду с **DataType** первичных или вторичных данных.

Когда используются избыточное кодирование по RFC 2198, **redundancyEncodingMethod** должен быть установлен равным **rtpRedundancyEncoding**. Кроме того, при использовании документа RFC 2198 и загрузке последовательности **RedundancyEncoding** должна использоваться только последовательность **rtpRedundancyEncoding**. Поля **RedundancyEncoding.secondaryEncoding** и **RedundancyEncoding.rtpRedundancyEncoding** не должны использоваться одновременно.

Если определено кодирование для канала, передающего несколько полезных нагрузок, избыточное кодирование согласно RFC 2198 используется для сохранения фактически передаваемых типов полезной нагрузки. Тип полезной нагрузки Encapsulating (инкапсуляция) устанавливается равным значению, заданному в поле **syncFlag** элемента **encryptionSync**.

h235 Key: Используется для включения и определения метода, с помощью которого защищаются ключи сеанса конкретной среды при их передаче между двумя конечными точками. Это поле в кодированном виде представляет собой вложенную величину ASN.1, соответствующую описанию в Рекомендации МСЭ-Т Н.235.

EscrowData используется для определения типа и содержания любого используемого механизма условного депонирования ключей. Если разрешено кодирование среды, то при реализации могут потребоваться конкретные типы и содержание.

T120SetupProcedure показывает порядок организации конференции T.120. Для **originateCall** и **waitForCall** вызывающая сторона должна выводить название конференции T.120 в числовом виде из идентификатора канала H.323 CID (как описано в Рекомендации МСЭ-Т Н.323) и выдавать соответствующий протокольный блок данных PDU (если конечная точка является ведущим объектом, то она должна выдать запрос с приглашением, а если она является ведомым объектом, то она должна выдавать запрос на соединение). Для **issueQuery** вызывающая сторона должна сначала выдать запрос о поиске, а затем организовать конференцию T.120 в соответствии с содержанием ответа на запрос о поиске (как описано в Рекомендации МСЭ-Т Т.124).

В.3.2 Подтверждение запроса на открытие логического канала

Это сообщение используется для подтверждения получения запроса на подключение логического канала от равноправного LCSE или B-LCSE. В случае запроса об организации однонаправленного логического канала оно показывает подтверждение приема запроса об однонаправленном логическом канале. В случае запроса об организации двунаправленного логического канала это подтверждение запроса об организации двунаправленного логического канала; в нем указаны соответствующие параметры обратного канала.

forwardLogicalChannelNumber задает номер логического канала для открываемого прямого канала.

reverseLogicalChannelParameters присутствует только в том случае, если это ответ на запрос об организации двунаправленного канала.

reverseLogicalChannelNumber задает номер логического канала для обратного канала.

portNumber – это межпользовательский параметр, который может применяться пользователем в таких целях, как ассоциирование входного или выходного порта или номера канала на более высоком уровне с обратным логическим каналом.

multiplexParameters показывает специфические параметры системы мультиплексирования H.222, H.223 или H.225.0, которая используется для транспортировки обратного логического канала.

Параметр **FlowControlToZero** показывает, разрешено ли отправителю начинать передачу по логическому каналу. Если его значение установлено равным "истине", то он показывает, что отправителю не следует осуществлять передачу по логическому каналу до получения следующего сообщения **FlowControl**, относящегося к данному логическому каналу и разрешающему ему вести передачу. Если он установлен в "ложь" или отсутствует, то отправителю разрешено начинать передачу сразу же после организации канала.

Параметр **replacementFor** определяет, что логический канал, который должен быть открыт, будет заменой указанного существующего уже открытого логического канала. Этот параметр должен использоваться только для обозначения логических каналов, уже находящихся в состоянии

ESTABLISHED ("организован"). По логическим каналам, открытым с использованием этого параметра, не должен передаваться никакой трафик данных до тех пор, пока не прекратится вся передача трафика по упомянутому организованному логическому каналу. В таком случае никогда не будет требоваться, чтобы медиадекодеры декодировали трафик данных от обоих логических каналов одновременно. После начала передачи трафика по новому организованному логическому каналу старый логический канал должен немедленно закрываться. Получатели могут подтверждать открытие логических каналов, используя механизм replacementFor, при этом старый и новый логические каналы не должны использоваться одновременно, следовательно, не будут превышены возможности декодирования получателя.

separateStack указывает, что для транспортировки данных будет использоваться отдельный транспортный стек, этот параметр предоставляет адрес, используемый при организации стека, который является адресом Q.2931, E.164 или транспортным адресом локальной сети.

forwardMultiplexAckParameters показывает специфические параметры, определенные для системы мультиплексирования, H.222, H.223 или H.225.0, применяемой для транспорта прямого логического канала.

Поле encryptionSync должно использоваться ведущим терминалом для предоставления значения ключа шифрования и точки синхронизации, в которой этот ключ следует применять. Для выполнения требований Рекомендации МСЭ-Т H.323 поле syncFlag должно быть установлено равным номеру динамической полезной нагрузки RTP, который соответствует данному ключу.

H2250LogicalChannelAckParameters используются для индикации параметров, определенных для использования Рекомендацией МСЭ-Т H.225.0.

sessionID – это уникальный идентификатор сеанса RTP конференции, который может создаваться только ведущим объектом. Он создается и передается ведущим объектом, если ведомое устройство выражает желание создать новый сеанс, указав недействительный идентификатор сеанса 0 в сообщении openLogicalChannelAck.

Параметр mediaChannel указывает транспортный адрес transportAddress, который должен использоваться для логического канала. Он должен присутствовать в сообщении OpenLogicalChannelAck, когда транспорт адресован одному устройству, за исключением того случая, когда в запросе OpenLogicalChannel был указан обратный одноадресный канал mediaChannel. Если transportAddress является групповым адресом, то ведущий объект отвечает за создание группового транспортного адреса, и он должен включать этот адрес в сообщение OpenLogicalChannel. Ведомый объект, желающий открыть новый многоадресный канал, должен передать нули в поле многоадресного transportAddress. Ведущий объект создаст и передаст групповой адрес transportAddress в сообщении OpenLogicalChannelAck для ведомого объекта. Обратите внимание на то, что МС будет использовать команду communicationModeCommand для указания подробностей, относящихся ко всем сеансам RTP в ходе конференции.

Параметр mediaChannel используется для описания транспортного адреса для логического канала. Адреса IPv4 и IPv6 должны кодироваться таким образом, чтобы самым старшим октетом адреса был первый октет в соответствующей строке октетов OCTET STRING, например, адрес IPv4 класса В 130.1.2.97 должен иметь "130", закодированные в первом октете строки октетов, за которым следует "1" и так далее. В адресе IPv6 a148:2:3:4:a:b:c:d "a1" должен быть закодирован в первом октете, "48" – во втором, "00" – в третьем, "02" – в четвертом и так далее. Адрес IPX, узел, сетевой номер netnum и порт должны кодироваться таким образом, чтобы старший октет каждого поля был первым октетом в соответствующей строке октетов.

mediaControlChannel указывает канал управления средой, который отправитель openLogicalChannelAck будет прослушивать на предмет сообщений управления средой для данного сеанса. Это поле присутствует только в том случае, когда требуется канал управления средой.

dynamicRTPPayloadType указывает динамическое значение полезной нагрузки, которое используется в Рекомендации H.323 для альтернативной схемы H.225.0 схем формирования пакетов видеосигнала H.261. Это поле присутствует только в том случае, когда применяется динамическая полезная нагрузка RTP.

Поле portNumber используется в Приложении С/H.323, если принимающая конечная точка устанавливает, что В-HLI, заданный в поле portNumber в сообщении OpenLogicalChannel, является недопустимым, и указывает, что необходимо использовать альтернативное значение.

ПРИМЕЧАНИЕ. – Параметры H.223 не включены в reverseLogicalChannelParameters, поскольку их значения были указаны в сообщении с запросом OpenLogicalChannel.

В.3.3 Отклонение запроса об открытии логического канала

Это сообщение используется для отказа в запросе организации логического канала, полученного от равноправного LCSE или B-LCSE.

ПРИМЕЧАНИЕ. – В случае отказа в запросе организовать двунаправленный канал этот отказ распространяется как на прямой, так и на обратный канал. Невозможно принять к выполнению один запрос и отказать в другом.

forwardLogicalChannelNumber задает номер прямого логического канала, указанного в том запросе, в выполнении которого отказано.

Поле cause ("причина") показывает причину отказа в организации логического канала. Значения причин приведены в таблице В.7.

Таблица В.7/Н.245 – Причины отказа в организации логического канала OpenLogicalChannel

Кодовая точка АСН.1	Причина
unspecified (не указана)	Не дано никакой причины отказа.
unsuitableReverseParameters	Эта причина должна использоваться для отказа в запросе организации двунаправленного логического канала только тогда, когда единственная причина отказа состоит в том, что запрошенные reverseLogicalChannelParameters являются неправильными. За таким отказом должна немедленно следовать инициализация процедур, направленных на формирование аналогичного, но приемлемого двунаправленного логического канала.
dataTypeNotSupported	Терминал не имел возможности поддерживать тип данных dataType, указанный в OpenLogicalChannel.
dataTypeNotAvailable	Терминал не имел возможности поддерживать тип данных dataType, указанный в OpenLogicalChannel, одновременно с типами данных dataTypes уже открытых логических каналов.
unknownDataType	Терминал не понял тип данных dataType, указанный в OpenLogicalChannel.
dataTypeALCombinationNotSupported	Терминал не имел возможности поддерживать dataType, обозначенный в OpenLogicalChannel, одновременно с типом Adaptation Layer ("уровень адаптации"), указанным в H223LogicalChannelParameters.
multicastChannelNotAllowed	Не удалось открыть канал для многоадресной передачи.
insufficientBandwidth	Не удалось открыть канал, поскольку было отказано в разрешении использовать требуемую ширину полосы для данного логического канала.
separateStackEstablishmentFailed	Не удался запрос о передаче части данных вызова по отдельному стеку.
invalidSessionID	Попытка ведомого объекта установить идентификатор сеанса SessionID при открытии логического канала с ведущим объектом.
masterSlaveConflict	Попытка ведомого объекта открыть логический канал, в котором, как установил ведущий объект, может возникнуть конфликт (см. С.4.1.3 и С.5.1.3).
waitForCommunicationMode	Попытка открыть логический канал до того, как МС передал команду CommunicationModeCommand.
invalidDependentChannel	Попытка открыть логический канал с указанием зависимого канала, который не существует.
replacementForRejected	Логический канал такого типа, который пытаются открыть, не может быть открыт посредством параметра replacementFor. Отправитель может повторить попытку, сначала закрыв тот логический канал, который должен быть заменен, а затем открыв заменяющий его канал.

В.3.4 Подтверждение открытия логического канала

Это сообщение используется при передаче сигналов двунаправленного канала для того, чтобы сообщить принимающему объекту В-LCSE, что обратный канал открыт и может использоваться для передачи.

forwardLogicalChannelNumber указывает логический номер прямого канала, который был открыт.

В.3.5 Заккрытие логического канала

Это сообщение используется передающим LCSE или В-LCSE для закрытия логического канала, организованного между двумя равноправными LCSE или В-LCSE.

ПРИМЕЧАНИЕ. – В случае двунаправленного логического канала при этом закрываются как прямой, так и обратный каналы. Невозможно закрыть только один из них и не закрыть другой канал.

forwardLogicalChannelNumber указывает номер логического канала для прямого канала для того логического канала, который должен быть закрыт.

Стороны, которые могут вызвать закрытие логического канала, приведены в таблице В.8.

Таблица В.8/Н.245 – Источники закрытия логического канала

Кодовая точка АСН.1	Причина
user (пользователь)	Пользователь LCSE или В-LCSE является источником закрытия.
lcse	LCSE или В-LCSE является источником закрытия. Это может произойти в результате ошибки протокола.

reason (причина) показывает, почему закрывается канал. reservationFailure означает, что не удалось обеспечить резервирование QOS для данного канала, и поэтому он закрывается. геореп (повторное открытие) свидетельствует о том, что конечной точке следует закрыть этот канал, а затем повторно открыть его, используя процедуры OpenLogicalChannel. Это может произойти, например, если многоточечный вызов сведен к вызову "точка-точка" в результате отключения окончательных точек от конференции.

В.3.6 Подтверждение закрытия логического канала

Это сообщение используется для подтверждения закрытия организованного логического канала.

forwardLogicalChannelNumber указывает логический номер прямого канала для того логического канала, который закрывается.

В.3.7 Запрос о закрытии канала

Это сообщение используется передающим CLCSE для того, чтобы запросить закрытие организованного логического канала между двумя равноправными LCSE.

forwardLogicalChannelNumber указывает номер логического канала для прямого канала того логического канала, который требуется закрыть.

qosCapability служит для указания параметров QOS, которые использовались в этом канале.

reason (причина) показывает, почему возникает этот запрос о закрытии канала. reservationFailure означает, что не удалось осуществить резервирование качества услуг QOS для данного канала и что поэтому он закрывается. геореп (повторное открытие) свидетельствует о том, что конечной точке следует закрыть канал, а затем повторно открыть его, используя процедуры OpenLogicalChannel. Это может произойти, например, если многоточечный вызов сведен к вызову "точка-точка" в результате отключения конечных точек от конференции.

В.3.8 Подтверждение запроса о закрытии канала

Это сообщение используется принимающим CLCSE, чтобы показать, что организованный логический канал будет закрыт.

forwardLogicalChannelNumber указывает номер логического канала для прямого канала того логического канала, который требовалось закрыть.

В.3.9 Отклонение запроса о закрытии канала

Это сообщение используется принимающим CLCSE, чтобы показать, что организованный логический канал не будет закрыт.

forwardLogicalChannelNumber указывает номер логического канала для прямого канала, который требовалось закрыть.

В поле cause (причина) приводится причина отказа в запросе о закрытии логического канала. Единственное допустимое значение причины – unspecified ("не указана").

В.3.10 Отклонение запроса о закрытии канала

Это сообщение передается передающим CLCSE в случае тайм-аута.

forwardLogicalChannelNumber указывает номер логического канала для прямого канала того логического канала, который он потребовал закрыть.

В.4 Сообщения сигнализации таблицы мультиплексирования

Этот набор сообщений предназначен для обеспечения безопасной передачи элементов таблицы мультиплексирования H.223 от отправителя получателю.

В.4.1 Отправка элемента таблицы мультиплексирования

Это сообщение используется для передачи элементов таблицы мультиплексирования H.223 от отправителя получателю. Оно передается от передающего MTSE и равноправного принимающего MTSE.

sequenceNumber используется для обозначения копий MultiplexEntrySend, чтобы можно было идентифицировать соответствующий ответ.

MultiplexEntryDescriptors – это набор, включающий от 1 до 15 дескрипторов MultiplexEntryDescriptors.

MultiplexEntryDescriptor: Описывает один элемент таблицы мультиплексирования. Он включает номер MultiplexTableEntryNumber и список элементов MultiplexElements. Отсутствие списка элементов означает, что данный элемент таблицы мультиплексирования отключен.

MultiplexElement: Рекурсивная структура, которая описывает одиночный элемент и счетчик повторений (repeat count). Если ее тип равен logicalChannelNumber, то данный элемент означает одиночный слот из заданного логического канала, а счетчик повторений задает длину слота в октетах. Если же ее тип равен subElementList, то данный элемент обозначает последовательность вложенных MultiplexElements, а счетчик повторений показывает, сколько раз повторить последовательность. В любом случае, если поле repeatCount содержит untilClosingFlag, это означает, что данный элемент надо повторять неопределенное количество раз до флага закрытия MUX-PDU.

В каждом MultiplexEntryDescriptor счетчик repeatCount окончательного MultiplexElement в списке elementList должен быть установлен в состояние "untilClosingFlag" ("до флага закрытия"), а repeatCount всех остальных MultiplexElements в elementList должен быть установлен в состояние "finite" ("конечный"). Так обеспечивается определение всеми элементами таблицы мультиплексирования схемы последовательности мультиплексирования неопределенной длины, повторяющейся до появления флага закрытия MUX-PDU. MultiplexEntryDescriptor с отсутствующим полем elementList должен обозначать деактивированный элемент таблицы мультиплексирования.

Каждый запрос MultiplexEntrySend может содержать до 15 дескрипторов MultiplexEntryDescriptors, каждый из которых описывает единственный элемент таблицы мультиплексирования. Элементы таблицы мультиплексирования могут передаваться в любом порядке.

В.4.2 Подтверждение отправки элемента таблицы мультиплексирования

Это сообщение используется для подтверждения получения от равноправного MTSE одного или нескольких дескрипторов multiplexEntryDescriptors из MultiplexEntrySend.

Параметр sequenceNumber должен быть тем же, что и sequenceNumber в сообщении MultiplexEntrySend, для которого настоящее сообщение является подтверждением.

multiplexTableEntryNumber указывает, какие элементы таблицы мультиплексирования подтверждаются.

В.4.3 Отклонение отправки элемента таблицы мультиплексирования

Это сообщение используется для отказа от одного или нескольких дескрипторов multiplexEntryDescriptors из MultiplexEntrySend, переданных равноправным MTSE.

Параметр sequenceNumber должен быть тем же, что и sequenceNumber в MultiplexEntrySend, для которого настоящее сообщение служит отказом.

MultiplexEntryRejectionDescriptions определяет, какие элементы таблицы мультиплексирования отклоняются и почему. Причины отклонения приведены в таблице В.9.

Таблица В.9/Н.245 – Причины отклонения MultiplexEntrySend

Кодовая точка АСН.1	Причина
unspecified (не указана)	Причина отклонения не указана.
descriptorTooComplex	Дескриптор nMultiplexEntryDescriptor превысил возможности приемного терминала.

В.4.4 Отмена передачи элемента таблицы мультиплексирования

Это сообщение передается передающим MTSE в случае тайм-аута.

multiplexTableEntryNumber указывает, для каких элементов таблицы мультиплексирования наступил тайм-аут.

В.5 Сообщения сигнализации запроса повторной передачи таблицы мультиплексирования

Этот набор сообщений предназначен для обеспечения безопасного запроса повторной передачи одного или нескольких дескрипторов MultiplexEntryDescriptors от отправителя получателю.

В.5.1 Запрос повторной передачи элемента таблицы мультиплексирования

Это сообщение используется для запроса повторной передачи одного или нескольких дескрипторов MultiplexEntryDescriptors.

entryNumbers – это список MultiplexTableEntryNumbers ("номера элементов таблицы мультиплексирования") дескрипторов MultiplexEntryDescriptors, для которых запрошена передача.

В.5.2 Подтверждение запроса о передаче элемента таблицы мультиплексирования

Это сообщение используется принимающим RMESE для того, чтобы показать, где будет передаваться элемент таблицы мультиплексирования.

entryNumbers – это список MultiplexTableEntryNumbers дескрипторов MultiplexEntryDescriptors, которые будут передаваться.

В.5.3 Отклонение запроса о передаче элемента таблицы мультиплексирования

Это сообщение используется принимающим RMESE, чтобы показать, что не будет передаваться элемент таблицы мультиплексирования.

entryNumbers – это список номеров MultiplexTableEntryNumbers дескрипторов MultiplexEntryDescriptors, который не будет передаваться. Значения номеров MultiplexTableEntryNumber в entryNumbers должны соответствовать значениям MultiplexTableEntryNumber в rejectionDescriptions, в противном случае во время работы могут возникнуть ошибки.

RequestMultiplexEntryRejectionDescriptions ("описания отказа в запросе о передаче элемента таблицы мультиплексирования") показывает, какие элементы таблицы мультиплексирования отвергаются и почему. Причины отклонения приведены в таблице В.10.

Таблица В.10/Н.245 – Причины отклонения MultiplexEntrySend

Кодовая точка АСН.1	Причина
unspecified (не указана)	Причина отклонения не указана.

В.5.4 Сброс запроса о передаче элемента таблицы мультиплексирования

Это сообщение передается передающим RMESE в случае тайм-аута.

entryNumbers – это список номеров MultiplexTableEntryNumbers дескрипторов MultiplexEntryDescriptors, для которого наступил тайм-аут.

В.6 Сообщения с запросом режима

Данный набор сообщений используется приемным терминалом для того, чтобы запросить конкретные режимы передачи у передающего терминала.

В.6.1 Запрос режима

Это сообщение используется для запроса конкретных режимов передачи у передающего терминала. Это список режимов, в которых терминал хотел бы осуществлять прием, составленный в порядке предпочтения (начиная с наиболее предпочтительного режима). Каждый режим описан с помощью ModeDescription.

sequenceNumber используется для обозначения экземпляров RequestMode ("запрос режима"), чтобы можно было идентифицировать соответствующий ответ.

ModeDescription: Набор, состоящий из одного или нескольких ModeElements ("элементов режима").

ModeElement: Используется для описания элемента режима, то есть одной из составных частей полного описания режима. Он указывает запрашиваемый тип элементарного потока, а также (дополнительно) каким образом его требуется мультиплексировать.

type (тип) используется для указания типа запрашиваемого элементарного потока. Он выбирается из VideoMode, AudioMode, DataMode, EncryptionMode и H235Mode. H235Mode показывает, что запрашивается зашифрованная среда.

multiplexedStreamMode ("режим мультиплексированного потока") задает требуемый режим передачи мультиплексированного потока. Поля MultiplexedStream имеют то же значение, что и поля с тем же именем в MultiplexedStreamCapability.

h223ModeParameters: Используется для индикации параметров, определенных для использования в Рекомендации МСЭ-Т Н.223 [10].

adaptationLayerType указывает, какой уровень адаптации и какие опции запрошены для требуемого типа. Возможны следующие кодовые точки: nonStandard (нестандартный), allFramed (кадрированный режим AL1), allNotFramed (кадрированный режим AL1), al2WithoutSequenceNumbers (AL2 без порядковых номеров), al2WithSequenceNumbers (AL2 с порядковыми номерами) и al3 (AL3 с индикацией количества присутствующих октетов управляющего поля и размера используемого пересылочного буфера V_S , при этом размер задается в октетах), al1M (A11M, определенный в Приложении С/Н.223, с заданными параметрами), al2M (A12M, определенный в Приложении С/Н.223, с заданными параметрами) и al3M (A13M, определенный в Приложении С/Н.223, с заданными параметрами).

Если параметр segmentableFlag принимает значение "истина", то это означает, что требуется сегментируемое мультиплексирование, а если он принимает значение "ложь", это означает, что требуется несегментируемое мультиплексирование.

h2250ModeParameters содержит специальную информацию, которая должна использоваться согласно положениям Рекомендации МСЭ-Т Н.225.0 и Н.323.

Если присутствует параметр redundancyEncodingMode, он определяет, какой метод избыточного кодирования *redundancyEncodingMethod* должен применяться и какое вторичное кодирование *secondaryEncoding* должно использоваться как избыточное кодирование. Первичное кодирование определяется элементом *type*, который содержится в *ModeElement* ("элемент режима").

genericModeParameters указывает параметры универсального режима.

multiplexedStreamModeParameters ("параметры режима мультиплексирования потока") обозначает логический канал мультиплексированного потока, к которому относится данный запрос режима: логический канал идентифицируется посредством поля **logicalChannelNumber**.

Если присутствует параметр **logicalChannelNumber**, он показывает логический канал, для которого запрошен указанный режим. **logicalChannelNumber** должен использоваться только для определения открытого логического канала.

В.6.1.1 Режим видеoinформации

Используется для выбора видеорежимов.

H261VideoMode: Указывает запрошенное разрешение изображения (или QCIF, или CIF), скорость передачи данных в единицах по 100 бит/с и передачу неподвижного изображения.

H262VideoMode: Показывает требуемый профиль и уровень, а в необязательных полях, если они имеются, содержатся запрашиваемые значения заданных параметров. Необязательные поля – это целые числа, а единицы измерения определены в таблице В.2.

H263VideoMode: Показывает запрашиваемое разрешение изображения (SQCIF, QCIF, CIF, 4CIF и 16CIF или некоторый заказной формат изображения) и скорость передачи данных в единицах по 100 бит/с. При взаимодействии с конечной точкой, поддерживающей Рекомендацию МСЭ-Т Н.245 версии 8 или более раннюю версию, невозможно запросить только заказной формат изображения. Поэтому если при получении сообщения **RequestMode** от конечной точки, поддерживающей Рекомендацию МСЭ-Т Н.245 версии 8 или более ранней версии, **RequestMode** содержит заказной формат изображения, то следует рассматривать как запрашиваемое разрешение его, а не разрешение, указанное в поле разрешения **H263VideoMode**.

Если булевы параметры **unrestrictedVector**, **arithmeticCoding**, **advancedPrediction** и **pbFrames** принимают значение "истина", то они свидетельствуют о том, что требуется использовать те необязательные режимы, которые определены в Приложениях к Рекомендации МСЭ-Т Н.263.

Если булев параметр **errorCompensation** принимает значение "истина", он показывает, что кодер способен обработать указания **videoNotDecodedMBs** и компенсировать ошибки, как это описано в Приложении I/Н.263. Не требуется, чтобы кодер отвечал на указания **videoNotDecoded** ("видеoinформация не декодирована"). Представляется нецелесообразным, чтобы узел управления многосторонней связи (MCU) отвечал на все указания.

EnhancementOptions: Показывает параметры запрошенного уровня расширения масштабируемости.

H263Options: Показывает запрошенные необязательные режимы Рекомендации МСЭ-Т Н.263.

IS11172VideoMode: Показывает, что запрошен **constrainedBitstream** ("ограниченный поток битов"), а если присутствуют необязательные поля, то они отображают запрошенные значения заданных параметров. Необязательные поля – это целые числа, соответствующие единицы измерения определены в таблице В.3.

genericVideoMode указывает универсальные параметры видеорежима.

В.6.1.2 Режим аудиоинформации

Используется для выбора аудиорежимов.

Точное значение кодовых точек аудиоинформации серии G приведено в таблице В.4. Для аудиоинформации G.723.1 предусмотрены четыре варианта, позволяющие запрашивать любую из скоростей передачи информации (низкую скорость передачи 5,3 кбит/с или высокую скорость передачи 6,3 кбит/с) с использованием подавления паузы или без подавления паузы.

G7231AnnexCMode: Используется для запроса аудиоинформации, закодированной согласно Приложению C/G.723.1. **maxAl-sduAudioFrames** указывает требуемое максимальное количество аудиокадров на AL-SDU. Если булев параметр **silenceSupression** принимает значение "истина", то он запрашивает применение подавления паузы, определенное в Приложении A/G.723.1. Поля **g723AnnexCAudioMode**, **highRateMode0**, **highRateMode1**, **lowRateMode0**, **lowRateMode1**, **sidMode0** и **sidMode1** показывают требуемое количество октетов на кадр для каждого режима передачи аудиоинформации и режимов защиты от ошибок Рекомендации МСЭ-Т G.723.1 и Приложения C/G.723.1.

IS11172AudioMode: Используется для запроса аудиоинформации, закодированной согласно стандарту ИСО/МЭК 11172-3 [45].

audioLayer показывает запрошенный уровень кодирования: audioLayer1, audioLayer2 или audioLayer3.

audioSampling указывает запрошенную частоту дискретизации: audioSampling32k, audioSampling44k1 и audioSampling48k соответствуют частоте дискретизации аудиосигнала 32 кГц, 44,1 кГц и 48 кГц, соответственно.

multichannelType показывает, какой многоканальный режим запрошен: singleChannel, twoChannelStereo и twoChannelDual запрашивают работу по одиночному каналу, стереоканалу и двойному каналу соответственно.

bitRate обозначает запрошенную скорость передачи аудиоинформации, он измеряется в кбит/с.

IS13818AudioMode: Используется для запроса аудиоинформации, закодированной согласно стандарту ИСО/МЭК 13818-3 [46].

audioLayer показывает запрошенный уровень кодирования: audioLayer1, audioLayer2 или audioLayer3.

audioSampling указывает запрошенную частоту дискретизации: audioSampling16k, audioSampling22k05, audioSampling24k, audioSampling32k, audioSampling44k1 и audioSampling48k соответствуют частоте дискретизации аудиосигнала 16 кГц, 22,05 кГц, 24 кГц, 32 кГц, 44,1 кГц и 48 кГц соответственно.

multichannelType показывает, какой из многоканальных режимов, определенных в таблице В.11, запрошен.

Таблица В.11/Н.245 – Кодовые точки при многоканальном режиме, соответствующем ИСО/МЭК 13818-3

Кодовая точка АСН.1	Семантическое значение кода кодовой точки
singleChannel	Один канал, использующий конфигурацию 1/0. Одноканальный режим (как в ИСО/МЭК 11172-3)
twoChannelStereo	Два канала, использующие конфигурацию 2/0, режим стереоканала (как в ИСО/МЭК 11172-3)
twoChannelDual	Два канала, использующие конфигурацию 2/0, двухканальный режим (как в ИСО/МЭК 11172-3)
threeChannels2-1	Три канала, использующие конфигурацию 2/1. Левый канал, правый канал и одиночный канал окружающего звука
threeChannels3-0	Три канала, использующие конфигурацию 3/0. Левый, центральный и правый каналы без канала окружающего звука
fourChannels2-0-2-0	Четыре канала, использующие конфигурацию 2/0 + 2/0. Левый и правый каналы первой программы и левый и правый каналы второй программы
fourChannels2-2	Четыре канала, использующие конфигурацию 2/2. Левый, правый каналы, левый канал окружающего звука и правый канал окружающего звука
fourChannels3-1	Четыре канала, использующие конфигурацию 3/1. Левый, центральный, правый каналы и одиночный канал окружающего звука
fiveChannels3-0-2-0	Пять каналов, использующие конфигурацию 3/0 + 2/0. Левый, центральный и правый каналы первой программы и левый и правый каналы второй программы
fiveChannels3-2	Пять каналов, использующие конфигурацию 3/2. Левый, центральный, правый каналы, левый канал окружающего звука и правый канал окружающего звука

Если булев параметр lowFrequencyEnhancement принимает значение "истина", он запрашивает низкочастотный канал расширения.

Если булев параметр multilingual принимает значение "истина", он запрашивает до семи многоязычных каналов.

Параметр bitRate обозначает требуемую скорость передачи аудиосигнала, он измеряется в единицах кбит/с.

genericAudioMode указывает универсальные параметры аудиорежима.

В.6.1.3 Режим передачи данных

Используется для выбора приложения передачи данных и скорости передачи.

Параметр `bitRate` обозначает требуемую скорость передачи данных, он измеряется в единицах по 100 бит/с.

`t120` запрашивает использование протокола T.120 [32].

`dsm-ss` запрашивает использование протокола DSM-CC [47].

`userData` запрашивает использование неопределенных пользовательских данных из внешних портов передачи данных.

`t84` запрашивает использование Рекомендации МСЭ-Т T.84 [31] для передачи требуемых изображений (JPEG, JBIG, факсимильное изображение Группы 3/4).

`t434` запрашивает применение Рекомендации МСЭ-Т T.434 [35] для передачи телематических бинарных файлов.

`h224` запрашивает использование протокола управления симплексным устройством в реальном времени H.224 [11].

`nlpid` запрашивает применение указанного сетевого приложения передачи данных канального уровня.

`dsvdControl` запрашивает использование терминала DSVD для поддержки канала управления, работающего вне полосы.

`h222DataPartitioning` запрашивает применение модифицированного и ограниченного дробления данных согласно H.262, как определено в Рекомендации МСЭ-Т H.222.1, при котором улучшенные данные передаются как канал передачи данных, поддерживаемый описанной возможностью `DataProtocolCapability`.

`t30fax` запрашивает использование аналогового режима (G3V) из Приложения С/Т.30, соответствующего определению Рекомендации МСЭ-Т T.39, для режимов DSVF/MSVF.

`t140` запрашивает применение протокола текстового диалога T.140, определенного в Рекомендации МСЭ-Т T.140.

`t38fax` запрашивает применение Рекомендация МСЭ-Т T.38 [29].

genericDataMode: показывает параметры универсального режима передачи данных. Если `maxBitRate` включен в `genericDataMode`, его значение должно совпадать со значением `maxBitRate` в `DataMode`.

В.6.1.4 Режим шифрования

Представляет собой выбор режимов шифрования.

`h233Encryption` запрашивает применение шифрования в соответствии с Рекомендациями МСЭ-Т H.233 [14] и H.234 [15].

В.6.2 Подтверждение запроса режима

Это сообщение отправляется для подтверждения того, что передающий терминал намеревается осуществить передачу в одном из тех режимов, которые запросил принимающий терминал.

Параметр `sequenceNumber` должен быть таким же, как и `sequenceNumber` в `RequestMode`, для которого настоящее сообщение является подтверждением.

Поле `response` (ответ) показывает действие удаленного терминала. Возможные значения ответа приведены в таблице В.12.

Таблица В.12/Н.245 – Ответы с подтверждением запроса режима

Кодовая точка АСН.1	Ответ
<code>willTransmitMostPreferredMode</code>	Передающий терминал переключится на наиболее предпочтительный получателем режим.
<code>willTransmitLessPreferredMode</code>	Передающий терминал переключится на один из предпочтительных режимов получателя, но не на наиболее предпочтительный режим.

В.6.3 Отклонение запроса режима

Это сообщение передается для отказа в запросе приемного терминала.

Параметр `sequenceNumber` должен быть таким же, как и `sequenceNumber` в `RequestMode`, для которого данное сообщение является ответом.

В поле `cause` ("причина") указана причина отказа в предоставлении затребованного режима. Значения причины приведены в таблице В.13.

Таблица В.13/Н.245 – Отклоняющие ответы на запрос режима

Кодовая точка АСН.1	Ответ
<code>modeUnavailable</code>	Передающий терминал не будет менять свой режим передачи, поскольку требуемые режимы недоступны.
<code>multipointConstraint</code>	Передающий терминал не будет менять свой способ передачи из-за ограничения, обусловленного многоточечным соединением.
<code>requestDenied</code>	Передающий терминал не будет менять свой режим передачи.

В.6.4 Сброс запроса режима

Это сообщение используется передающим MRSE в случае тайм-аута.

В.7 Сообщения о задержке из-за подтверждения приема

Данный набор сообщений используется терминалом для определения времени задержки между двумя взаимодействующими терминалами из-за подтверждения приема. Кроме того, он дает возможность пользователю Н.245 определить, работает ли равноправный объект протокола Н.245.

В.7.1 Запрос задержки из-за подтверждения приема

Это сообщение передается передающим RTDSE на принимающий RTDSE.

`sequenceNumber` применяется для того, чтобы пометить экземпляры запроса `RoundTripDelayRequest` для идентификации соответствующего ответа.

В.7.2 Ответ на запрос задержки из-за подтверждения приема

Это сообщение передается принимающим RTDSE на передающий RTDSE.

Параметр `sequenceNumber` должен быть тем же, что и `sequenceNumber` в запросе `RoundTripDelayRequest`, для которого данное сообщение является ответом.

В.8 Сообщения цикла обслуживания

Этот набор сообщений используется терминалом для выполнения функций цикла обслуживания.

В.8.1 Запрос об организации цикла обслуживания

Это сообщение передается для запроса цикла особого типа. Для типов `mediaLoop` и `logicalChannelLoop` требуется цикл только одного логического канала, задаваемого `LogicalChannelNumber`, тогда как тип `systemLoop` относится ко всем логическим каналам. Точное определение этих типов зависит от системы и выходит за рамки настоящей Рекомендации.

В.8.2 Подтверждение запроса об организации цикла обслуживания

Это сообщение используется для подтверждения того, что терминал организует цикл согласно запросу.

В.8.3 Отклонение запроса об организации цикла обслуживания

Это сообщение используется, чтобы показать, что терминал не будет организовывать цикл в соответствии с запросом.

Терминал может использовать причину `canNotPerformLoop` для индикации того, что он не имеет возможности организовать запрошенный цикл.

В.8.4 Команда отключения цикла обслуживания

После получения этой команды терминал должен отключить все циклы и восстановить в нормальное состояние тракты аудиосигналов, видеосигналов и данных.

В.9 Сообщения о режиме взаимодействия

Этот набор сообщений используется объектом МС Н.323 для передачи режима взаимодействия конференции Н.323.

В.9.1 Команда определения режима взаимодействия

Команда **CommunicationModeCommand** ("команда определения режима взаимодействия") передается объектом МС Н.323 для определения режима связи для каждого типа среды – одноадресного или многоадресного. Эта команда может вызвать переключения с централизованной конференции на децентрализованную конференцию или наоборот, и, следовательно, она может привести к закрытию всех существующих логических каналов и открытию новых логических каналов.

Команда **CommunicationModeCommand** определяет все сеансы конференции. Для каждого сеанса устанавливаются следующие данные: идентификатор сеанса RTP, соответствующий идентификатор сеанса RTP (если он применим), метка терминала (если она применима), описание сеанса, тип данных сеансов (например, G.711), а также одиночный или групповой адрес для каналов передачи медиаинформации и каналов управления средой в зависимости от конфигурации конференции и типа конференции. В том случае, если должно использоваться избыточное кодирование, `communicationModeTableEntry` ("элемент таблицы режимов связи") также определяет метод избыточного кодирования `redundancyEncodingMethod`, а также формат вторичного кодирования.

Команда **CommunicationModeCommand** сообщает режимы передачи, которые конечные точки конференции должны использовать в ходе конференции. В этой команде не указываются режимы приема, поскольку они задаются командами **OpenLogicalChannel**, которые передаются из МС конечным точкам.

Предполагается, что команда **CommunicationModeCommand** определяет режимы конференции, и, следовательно, она передается после индикации **multipointConference** ("многоточечная конференция"), которой конечная точка уведомляется о том, что она должна выполнять команды МС. Если конечные точки получили индикацию **multipointConference**, то они должны ожидать команду **CommunicationModeCommand** прежде, чем открывать логические каналы.

Конечные точки, получающие команду **CommunicationModeCommand**, используют поле **terminalLabel** каждого элемента таблицы для того, чтобы определить, применим ли данный элемент для его собственной обработки. Элементы таблицы, которые не содержат **terminalLabel**, относятся ко всем конечным точкам конференции. Элементы таблицы, которые содержат **terminalLabels**, представляют собой команды для конкретных конечных точек, которые соответствуют **terminalLabel** в этом элементе. Например, если потоки аудиосигналов от всех конечных точек помещены на один групповой адрес (один сеанс), элемент таблицы для аудиорежима, медиа адреса и адреса управления средой не будут содержать **terminalLabel**. Если элемент таблицы подает конечной точке команду передать свою видеоинформацию по групповому адресу, МС включит **terminalLabel** этой конечной точки.

Параметр `SessionDependency` ("зависимость сеанса") установлен МС для указания на то, что некоторый сеанс зависит от другого сеанса в плане смыслового декодирования данных.

Поле **destination** ("назначение") в элементе `CommunicationModeTableEntry` ("элемент таблицы режима связи") указывает конечную точку, с которой передающей конечной точке следует открыть логический канал. Если в элементе `CommunicationModeTableEntry` имеется поле **destination**, то конечная точка должна использовать этот адресат в качестве поля **destination** `H2250LogicalChannelParameters` в сообщении `OpenLogicalChannel`.

Команда `CommunicationModeCommand` может использоваться, чтобы передать конечным точкам конференции (или двухточечного соединения) команду изменить режим (для чего с уже используемым `mediaChannel` указывается новый режим) или вести передачу на новый адрес (в этом случае указывается используемый в настоящее время режим, но дается новый `mediaChannel`). Аналогично конечная точка, которая получает команду `CommunicationModeCommand`, указывающую используемый в настоящее время режим без `mediaChannel`, должна закрыть соответствующий канал и попытаться снова открыть его, используя процедуры `OpenLogicalChannel`, в которых `OpenLogicalChannelAck` содержит адрес, по которому конечная точка будет посылать информацию.

В.9.2 Запрос режима взаимодействия

Этот запрос передается на МС для запроса режима взаимодействия текущей конференции.

В.9.3 Ответ на запрос режима взаимодействия

Этот запрос передается объектом многоточечного управления (МС) в ответ на запрос `CommunicationModeRequest` с указанием режима взаимодействия конференции.

В.10 Сообщения с запросом конференции и ответом на него

`TerminalID` ("идентификатор терминала"), который используется в сообщениях с запросом конференции и ответом на него, имеет длину, равную 128 октетам. При связи между терминалом Н.323 и терминалом Н.320 через шлюз Н.323 это поле будет усечено до 32 октетов.

В.10.1 Запрос списка терминалов

Данный запрос равносильен TCU Н.230, описанному в Рекомендации МСЭ-Т Н.243.

В.10.2 Ответ на запрос списка терминалов

Данный запрос равносильен последовательности `terminalNumbers`, описанной в Рекомендации МСЭ-Т Н.230.

В.10.3 Запрос "Сделай меня председателем" (Make Me Chair)

Данный запрос равносильен CCA, описанному в Рекомендации МСЭ-Т Н.230.

В.10.4 Отмена запроса "Сделай меня председателем"

Данный запрос равносильен CIS, описанному в Рекомендации МСЭ-Т Н.230.

В.10.5 Ответ на запрос "Сделай меня председателем"

Данный запрос равносильен CIT Н.230, если предоставлен маркер управления председателем, или CCR Н.230, если в маркере управления председателем отказано.

В.10.6 Запрос об удалении терминала

Данный запрос равносильен CCD, описанному в Рекомендации МСЭ-Т Н.230.

В.10.7 Отклонение запроса об удалении терминала

Данный ответ равносильен CIR, описанному в Рекомендации МСЭ-Т Н.230.

В.10.8 Запрос идентификатора терминала (RequestTerminal)

Данный запрос равносильен TCR, описанному в Рекомендации МСЭ-Т Н.230.

В.10.9 Ответ на запрос идентификатора терминала МС

Данный ответ равносильен TPR, описанному в Рекомендации МСЭ-Т Н.230.

В.10.10 Запрос ввода пароля Н.243

Данный запрос равносильен TCS1, описанному в Рекомендации МСЭ-Т Н.230.

В.10.11 Ответ на запрос о вводе пароля

Данный ответ равносильен IIS, описанному в Рекомендации МСЭ-Т Н.230.

В.10.12 Запрос ввода идентификатора терминала Н.243

Данный запрос равносильен TCS2/TCL, описанному в Рекомендации МСЭ-Т Н.230.

В.10.13 Ответ на запрос об идентификаторе терминала

Данный ответ равносильен IIS, описанному в Рекомендации МСЭ-Т Н.230.

В.10.14 Запрос ввода идентификатора конференции Н.243

Данный запрос равносильен TCS3, описанному в Рекомендации МСЭ-Т Н.230.

В.10.15 Ответ на запрос об идентификаторе конференции

Данный ответ равносильен IIS, описанному в Рекомендации МСЭ-Т Н.230.

В.10.16 Отказ в выполнении команды видеoinформации

Данный запрос равносильен VCR Н.230.

В.10.17 Запрос ввода адреса расширения

Данный запрос равносильен TCS4, как описано в Рекомендации МСЭ-Т Н.230.

В.10.18 Ответ на запрос о вводе адреса расширения

Этот ответ равносильен IIS, описанному в Рекомендации МСЭ-Т Н.230.

В.10.19 Запрос владельца маркера управления председателем

Данный запрос равносильен TCA, описанному в Рекомендации МСЭ-Т Н.230 для маркера управления председателем.

В.10.20 Ответ на запрос о владельце маркера управления председателем

Данный ответ равносильен TIR, описанному в Рекомендации МСЭ-Т Н.230 для маркера управления председателем.

В.10.21 Запрос сертификата терминала

Данный запрос выдается любой конечной точкой конференции на ее МС. Он позволяет конечной точке получить цифровой сертификат для пользователя конкретного терминала. Запрашивающий терминал может дополнительно включить свой собственный terminalCertificate ("сертификат терминала") и challengeString ("строку вызова"), которая зашифрована с помощью секретного ключа.

Параметр CertSelectionCriteria ("критерии выбора сертификата") определяет набор сертификатов, приемлемых для инициатора запроса. Респонденту (данному МС) следует попытаться соответствовать этим критериям. CertSelectionCriteria может присутствовать вместе с меткой терминала terminalLabel. В последнем случае МС может использовать критерии либо для выбора соответствующего сертификата из сертификатов, представленных указанным терминалом, либо может запросить у указанного терминала сертификат, соответствующий этим критериям, который он затем возвращает исходному инициировавшему запрос терминалу.

Этот ответ может вернуть цифровой сертификат и дополнительно подпись, ассоциированная с сертификатом следующим образом:

- Если источник terminalCertificateResponse ("ответа на запрос сертификата терминала") не имеет подходящего сертификата, то это сообщение может быть возвращено без сертификата (и, следовательно, без структуры certificateResponse ("ответ с сертификатом")).
- Если конечная точка запрашивает сертификат другой конечной точки в многосторонней конференции (обозначенный меткой terminalLabel), то отвечающий МС должен вернуть сертификат, ассоциированный с запрашиваемой конечной точкой (содержащийся в структуре certificateResponse).

- Должна присутствовать структура `certificateResponse`. В случае, если МС представляет сертификат инициатору запроса от имени другой конечной точки, должна существовать криптографическая связь между подписями и подписью МС. Это должно обеспечиваться одним из двух следующих способов.
 - Должен использоваться секретный ключ, который используется для защиты материала сеансового ключа, распределенного при самом последнем обмене.
 - Если не было обмена материалом ключа или если ключ не подходит для подписания, то источником секретного ключа должен быть сертификат, который использовался во время самой последней идентификации конечной точки МС.

В.10.22 Ответ на запрос сертификата терминала

В данном ответе возвращается цифровой сертификат и `responseString` ("строка ответа"), которая зашифрована посредством секретного ключа для конкретного терминала.

В.10.23 Запрос на осуществление широковещательной передачи своего логического канала

Этот запрос подобен MCV H.230, используемому согласно процедуре пункта 6.3.2.2/H.243, но он относится только к одиночному логическому каналу; ответом на него служит сообщение `broadcastMyLogicalChannelResponse`, которое подтверждает получение этого запроса. Обратите внимание на то, что если применима процедура MCV из пункта 6.3.2.1/H.243 (то есть когда на любом конце линии между терминалом и MCU или между MCU и MCU отсутствует характеристика `multipointVisualizationCapability` ("характеристика многоточечной визуализации")), вместо этого используется форма команды `conferenceCommand` запроса об осуществлении широковещательной передачи своего логического канала `BroadcastMyLogicalChannel`.

В.10.24 Ответ на запрос об осуществлении широковещательной передачи своего логического канала

Сообщает ответ с разрешением на выполнение запроса об осуществлении широковещательной передачи своего логического канала `BroadcastMyLogicalChannel` или с отказом от него.

В.10.25 Запрос "Сделать терминал широковещательным"

Такой запрос подобен VCB H.230, ответом на него служит сообщение `makeTerminalBroadcasterResponse`, которое подтверждает данный запрос.

В.10.26 Ответ на запрос "Сделать терминал широковещательным"

Сообщает ответ с разрешением на выполнение запроса `MakeTerminalBroadcaster` "Сделать терминал широковещательным" или с отказом в нем.

В.10.27 Запрос "Отправить этот источник"

Этот запрос подобен VCS H.230, ответом на него служит сообщение `SendThisSourceResponse` ("ответ на запрос об отправке этого источника"), которое подтверждает данный запрос.

В.10.28 Ответ на запрос "Отправить этот источник"

Сообщает ответ с разрешением на выполнение запроса `SendThisSource` ("Отправить этот источник") или с отказом в нем.

В.10.29 Запрос всех идентификаторов терминалов

Отправляется конечной точкой на МС конференции для получения всех меток терминалов и идентификаторов терминалов участников конференции.

В.10.30 Ответ на запрос всех идентификаторов терминалов

Ответ на запрос всех идентификаторов терминалов `RequestAllTerminalIDs`, содержащий список всех конечных точек конференции с меткой `terminalLabel` и идентификатором `terminalID`.

В.10.31 Запрос удаленного МС

Этот запрос отправляется активным МС другому МС для того, чтобы активизировать/деактивировать его. Запрос `RemoteMC`, в котором выбран вариант `masterActivate` ("активизация в качестве ведущего") или `slaveActivate` ("активизация в качестве ведомого"), может передаваться активным МС на неактивный МС для активизации его в качестве ведущего или ведомого объекта, соответственно, в каскадном включении. Запрос `RemoteMC`, в котором выбран вариант `deActivate` ("деактивировать"), может передаваться ведущим МС на уже активный ведомый МС для того, чтобы деактивировать его.

В.10.32 Ответ на запрос удаленного МС

Запрос удаленного МС RemoteMC Response показывает, что запрос RemoteMC принят или отклонен. Прием этого запроса определяется в соответствии со следующими критериями:

Choice (выбранный вариант) = activateSlave

Получатель неактивен, а отправитель данного запроса инициализировал этот вызов с помощью conferenceGoal ("цель конференции") в INVITE ("пригласить") в сообщении об организации вызова Setup H.225, или же получатель данного запроса инициировал этот вызов посредством conferenceGoal в JOIN ("присоединиться") в сообщении H.225.

Choice (выбранный вариант) = activateMaster

Получатель неактивен, а отправитель данного запроса инициировал этот вызов посредством conferenceGoal в CREATE ("создать") в сообщении Setup H.225.

Choice = deActivate

Получателем является активный МС.

Если указанные выше условия не удовлетворяются, то запрос следует отклонить, выбрав invalidConfiguration ("недействительная конфигурация").

Конечные точки, не поддерживающие каскадное соединение, выбирают вариант functionNotSupported ("функция не поддерживается").

В.11 Сообщения многоканальной связи

Сообщения multilinkRequest, multilinkResponse и multilinkIndication используются для поддержки агрегирования канала согласно Рекомендации МСЭ-Т H.226, которое определено в Приложении F/H.324. Эти сообщения обеспечивают добавление и отмену физических подключений, автоматический обмен сетевыми адресами (номерами телефонов), а также управление работой в соответствии с H.226.

В.11.1 Запрос callInformation и ответ на него

Запрос MultilinkRequest.callInformation используется иницирующей стороной согласно определению в Приложении F/H.324 для того, чтобы запросить информацию, необходимую для организации и ассоциирования дополнительных физических соединений. Максимальное количество дополнительных соединений, которые может установить отправитель, сообщается в параметре maxNumberOfAdditionalConnections ("максимальное число дополнительных соединений").

Сообщение MultilinkResponse.callInformation включает параметр DialingInformation, содержимое которого будет описано ниже, а также callAssociationNumber ("номер ассоциации вызова"). Параметр callAssociationNumber должен содержать 32-битовое равномерно распределенное случайное число. При любых последующих обменах callInformation внутри того же сеанса должен использоваться тот же самый callAssociationNumber.

В.11.2 Запрос addConnection и ответ на него

Респондент может использовать сообщение MultilinkRequest.addConnection ("многоканальный запрос о добавлении соединения") в соответствии с определением в Приложении F/H.324 для того, чтобы попросить инициатора добавить физические соединения. Структура DialingInformation должна указывать соединения, которые предстоит добавить. Параметр sequenceNumber должен наращиваться на 1 по модулю 256 для каждого нового отправленного сообщения MultilinkRequest.addConnection.

После приема этого сообщения инициатор должен ответить сообщением MultilinkResponse.addConnection ("ответ на многоканальный запрос о добавлении соединения"), указывая, что он либо намеревается добавить требуемые соединения, либо что он не предполагает этого делать, а также соответствующий код причины. Параметр sequenceNumber должен быть равен параметру sequenceNumber соответствующего сообщения MultilinkRequest.addConnection.

В.11.3 Запрос removeConnection Request и ответ на него

Инициатор или респондент может использовать сообщение MultilinkRequest.removeConnection ("многоканальный запрос об отмене соединения") в соответствии с определением в Приложении F/H.324 для запроса дальнего конца убрать канал из набора каналов H.226. Оно служит частью процедуры в Приложении F/H.324 при отмене физических соединений. Параметр

connectionIdentifier ("идентификатор соединения") должен идентифицировать тот канал, который предстоит убрать, используя нумерацию каналов, полученную посредством H.226 от терминала, принимающего сообщение MultilinkRequest.removeConnection.

Сообщение MultilinkResponse.removeConnection ("ответ на многоканальный запрос об отмене соединения") должно посылаться в качестве ответа после того, как канал был удален из набора каналов H.226, оно показывает, что данный канал более не используется (или никогда не использовался). Параметр connectionIdentifier должен совпадать со значением в соответствующем сообщении MultilinkRequest.removeConnection.

V.11.4 Запрос maximumHeaderInterval и ответ на него

Сообщение MultilinkRequest.maximumHeaderInterval ("многоканальный запрос максимального интервала между заголовками") может использоваться для запроса фактического максимального интервала между заголовками H.226, используемых удаленным терминалом без изменения (вариант currentIntervalInformation ("информация о текущем интервале")), или для запроса о том, чтобы вместо него применялось некоторое определенное значение (вариант requestedInterval, единица измерения – миллисекунда).

В ответе должно передаваться сообщение MultilinkResponse.maximumHeaderInterval ("ответ на многоканальный запрос максимального интервала между заголовками"). Если соответствующим запросом был запрос информации о текущей минимальной скорости передачи, то терминал должен в ответ сообщить значение, которое его передатчик в настоящий момент использует в качестве максимального интервала между заголовками (Maximum Header Interval). Если в соответствующем запросе указывалась конкретная минимальная скорость передачи, которую следовало использовать, то терминал должен попытаться выполнить этот запрос, изменив максимальный интервал между заголовками, используемый передатчиком. Независимо от того, изменяется ли максимальный интервал между заголовками, в ответе должно предоставляться новое используемое значение (которое может отличаться от запрошенного значения).

V.11.5 Многоканальная индикация

Терминал может передавать сообщение MultilinkIndication.crcDesired ("индикация желательности многоканальной отправки CRC") для индикации того, что он хочет, чтобы удаленный терминал передавал необязательные данные CRC H.226 во всех последующих наборах данных. Принимающий терминал может по желанию согласиться: никакое явное подтверждение или ответ в явном виде не требуются.

Сообщение MultilinkIndication.excessiveError ("индикация о чрезмерном уровне ошибок при многоканальной связи") может передаваться, чтобы указать удаленному терминалу о том, что по некоторому соединению принимается чрезмерно много ошибок. Средства, дающие терминалу возможность установить чрезмерную частоту ошибок или критерий определения того, что она является чрезмерной, определяются локально для конкретного терминала. Соединение задается с помощью параметра connectionIdentifier ("идентификатор соединения"). После получения этого сообщения терминал может принять решение о действиях, направленных на устранение описанного явления. Конкретные действия, направленные на исправление явления, не определены.

V.11.6 Сообщение DialingInformation

Сообщение типа DialingInformation ("информация о выборе") применяется для предоставления сведений о наборе (телефонных номеров) в явном виде, чтобы обеспечить возможность автоматической организации физических соединений. Выбор с указанием отличий дает перечень параметров DialingInformationNumber ("количество сведений о наборе"), по одному на каждое возможное дополнительное соединение. Длина этого перечня соответствует максимальному количеству доступных дополнительных соединений. Если такой информации нет, то выбирается infoNotAvailable ("нет информации"), в котором указывается только количество доступных дополнительных соединений.

V.11.7 Сообщение DialingInformationNumber

Сообщение типа DialingInformationNumber включает до трех подпараметров, в которых даны сведения о наборе номера для физического соединения, показывающего отличия от соответствующей информации для уже установленного исходного соединения.

Параметр networkAddress ("сетевой адрес") должен содержать самую младшую (самую правую) часть номера телефона для соединения вплоть до самого старшего разряда (включительно), отличающуюся от номера первоначально установленного соединения; он не должен включать более старшие цифры. Если номер соединения совпадает с номером исходного соединения, то параметр networkAddress должен состоять из строки нулевой длины (поскольку в телефонном номере нет отличающихся цифр).

ПРИМЕЧАНИЕ. – Метод указания различающихся цифр используется вместо полной строки цифр E.164, потому что первые несколько цифр набираемого номера могут меняться в зависимости от географического положения двух терминалов, например от того, расположены ли они в одном городе.

Если для вызова номера используется субадрес и субадрес данного соединения отличается от подадреса исходного соединения, то респондент должен включать подадрес полностью в дополнительном параметре subAddress ("субадрес").

Типы сетей, поддерживаемые для соединения (KTCOP, ISDN или оба типа), должны обозначаться посредством параметра networkType ("тип сети").

V.11.8 Сообщение DialingInformationNetworkType

Сообщение DialingInformationNetworkType ("тип сети вызова") определяет тип сети: сеть с коммутацией каналов, сеть типа n-isdn (N-ISDN), gsm (KTCOP) или сеть мобильной связи (Mobile).

V.11.9 Сообщение ConnectionIdentifier

Сообщение типа ConnectionIdentifier ("идентификатор соединения") используется для уникальной идентификации одиночного физического соединения в H.226, используя параметры channelTag ("тег канала") и sequenceNumber ("номер последовательности") из заголовка H.226. Если тег канала был совсем не указан в заголовке, то значением параметра channelTag должен быть ноль.

V.12 Сообщения об изменении скорости передачи по логическому каналу

Сообщения LogicalChannelRateRequest ("запрос об изменении скорости передачи по логическому каналу"), LogicalChannelRateAcknowledge ("подтверждение запроса об изменении скорости передачи по логическому каналу"), LogicalChannelRateReject ("отклонение запроса об изменении скорости передачи по логическому каналу") и LogicalChannelRateRelease ("сброс запроса об изменении скорости передачи по логическому каналу") используются для изменения скорости передачи по логическому каналу. Процедура применения этих сообщений заключается в том, что терминал может запросить нужную скорость передачи для конкретного логического канала, а удаленный терминал может подтвердить или отклонить этот запрос.

Эти сообщения позволяют обеспечить улучшенный уровень взаимодействия, предусматривая запрос нужной скорости передачи в отличие от принудительного использования максимальной скорости передачи, предписанной командой FlowControlCommand, и обеспечивая обратную связь с информацией о том, выполнен или отклонен соответствующий запрос.

V.12.1 Запрос об изменении скорости передачи по логическому каналу

Он используется терминалом для того, чтобы запросить изменение скорости передачи указанного логического канала, по которому к этому терминалу ведется передача.

Параметр sequenceNumber используется для того, чтобы пометить случаи отправки сообщения LogicalChannelRateRequest, что позволит идентифицировать соответствующий ответ.

Параметр logicalChannelNumber ("номер логического канала") указывает логический канал, к которому относится настоящий запрос об изменении скорости передачи данных.

Параметр maximumBitRate ("максимальная скорость передачи данных") определяет максимальную скорость передачи данных, требуемую для логического канала (единица ее измерения составляет 100 бит/с).

V.12.2 Подтверждение запроса об изменении скорости передачи по логическому каналу

Это сообщение отправляют для того, чтобы подтвердить запрос об изменении скорости передачи по логическому каналу.

Параметр sequenceNumber должен быть таким же, как и sequenceNumber в сообщении LogicalChannelRateRequest, для которого данное сообщение является ответом.

Параметр logicalChannelNumber указывает логический канал, к которому относится данный запрос об изменении скорости передачи данных.

Параметр maximumBitRate определяет максимальную скорость передачи данных по логическому каналу, которую подтверждает данный терминал (единица ее измерения составляет 100 бит/с).

В.12.3 Отклонение запроса об изменении скорости передачи по логическому каналу

Это сообщение отправляют для того, чтобы отклонить запрос об изменении скорости передачи по логическому каналу.

Параметр `sequenceNumber` должен быть таким же, как и `sequenceNumber` в сообщении `LogicalChannelRateRequest`, для которого данное сообщение является ответом.

Параметр `logicalChannelNumber` указывает логический канал, к которому относится настоящий запрос об изменении скорости передачи данных.

Параметр `rejectReason` ("причина отказа") сообщает причину, по которой запрос отклонен. На настоящий момент определены следующие причины: `undefined reason` ("причина не определена") и `insufficient resources` ("недостаточные ресурсы").

Параметр `currentMaximumBitRate` показывает максимальную скорость передачи данных по логическому каналу, с которой терминал намеревается вести передачу данных (единица ее измерения составляет 100 бит/с).

В.12.4 Сброс запроса об изменении скорости передачи по логическому каналу

Это сообщение передается в случае тайм-аута.

В.13 Команды

Командное сообщение требует действия, но не требует ответа в явном виде.

В.13.1 Команда "Передать набор возможностей терминала"

По команде `specificRequest` ("запрос особенностей") терминал на удаленном конце указывает свои возможности передачи и приема, для чего он посылает один или несколько наборов `TerminalCapabilitySets`, которые содержат требуемую информацию, как это определено ниже. Эта команда может передаваться в любое время для того, чтобы узнать от удаленного терминала его возможности, например после прерывания или в случае неопределенности по другой причине; однако эти сообщения не следует передавать повторно много раз без веского повода.

Терминал должен запрашивать передачу только тех `capabilityTableEntryNumbers` ("номеров элементов таблицы возможностей") и `capabilityDescriptorNumbers` ("номеров дескрипторов возможностей"), которые он получал ранее. Терминал должен игнорировать все запросы о передаче тех `capabilityTableEntryNumbers` и `capabilityDescriptorNumbers`, которые он ранее не передавал, и это не должно считаться ошибкой.

Если булев параметр `multiplexCapability` ("возможность мультиплексирования") принимает значение "истина", он запрашивает передачу `MultiplexCapability`.

`capabilityTableEntryNumbers` – это набор `CapabilityTableEntryNumbers`, указывающих те `CapabilityTableEntrys` ("элементы таблицы характеристик"), передачу которых запрашивает терминал.

`capabilityDescriptorNumbers` – это набор `capabilityDescriptorNumbers`, указывающих те `CapabilityDescriptors`, передачу которых запрашивает терминал.

Команда `genericRequest` ("универсальный запрос") заставляет удаленный терминал передать весь свой набор возможностей.

В.13.2 Команда "Шифрование"

По этой команде осуществляется обмен возможностями шифрования и передача вектора инициализации (IV), см. Рекомендации МСЭ-Т Н.233 [14] и Н.234 [15].

`encryptionSE` – это сообщение об обмене сеансами Н.233 (SE), за исключением того, что не должны использоваться биты защиты от ошибок, описанные в Рекомендации МСЭ-Т Н.233.

По команде `encryptionIVRequest` ("запрос вектора инициализации шифрования") блок шифрования удаленного конца передает новый вектор инициализации по логическому каналу, открытому для `encryptionData` ("данных шифрования").

`encryptionAlgorithmID` ("идентификатор алгоритма шифрования") сообщает получателю, что передающий терминал будет ассоциировать данное значение идентификатора `h233AlgorithmIdentifier` с нестандартным алгоритмом шифрования `associatedAlgorithm`.

В.13.3 Команда "Управление потоком данных"

Эта команда используется для определения верхнего предела скорости передачи данных или одиночного логического канала, или всей системы мультиплексирования. Терминал может отправлять эту команду для ограничения скорости передачи данных терминалом, находящимся на удаленном конце. Терминал, который принимает эту команду, должен выполнить ее.

Если используется тип области действия `logicalChannelNumber`, то этот предел относится к данному логическому каналу, если используется тип `resourceID` (идентификатор ресурса), то этот предел относится к данному виртуальному каналу АТМ, а если используется тип `wholeMultiplex` ("вся система мультиплексирования"), то этот предел относится ко всей системе мультиплексирования.

`maximumBitRate` измеряется в единицах по 100 бит/с, усредненных по неперекрывающимся последовательным периодам длительностью в одну секунду. Если он присутствует, указанный предел замещает собой любой существовавший ранее предел независимо от того, ниже он или выше. Если он отсутствует, то любые предыдущие ограничения скорости передачи данных для канала более не действуют.

В настоящей Рекомендации не определены точка, в которой применяется предел скорости передачи данных, и спецификация того, какие биты включаются в вычисление скорости передачи данных, но они должны быть определены в рекомендациях, использующих настоящую Рекомендацию.

Каждая передача этой команды влияет на конкретный логический канал или на всю систему мультиплексирования. Несколько таких команд могут действовать одновременно, ограничение по системе мультиплексирования не должно превышать число открытых логических каналов плюс один.

ПРИМЕЧАНИЕ. – Если возможная скорость передачи данных по логическому каналу ограничена конкретными значениями, например значением, заданным в G.723.1 для аудиоинформации, и пришел запрос на передачу со скоростью ниже, чем самая низкая скорость, с которой он нормально работает, то ответом должно быть прекращение передачи по этому логическому каналу.

В.13.4 Команда "Окончание сеанса"

Эта команда указывает на конец сеанса H.245. После передачи команды окончания сеанса `EndSessionCommand` терминал не должен больше отправлять сообщения, определенные в настоящей Рекомендации.

`disconnect` ("отключение") свидетельствует о том, что данное соединение будет отключено.

gstnOptions: Выбор вариантов, которые используются после окончания сеанса H.245, когда модем серии V используется по сети типа КТСОП.

Возможные варианты приведены в таблице В.14.

Таблица В.14/H.245 – Варианты работы после команды `EndSessionCommand` при использовании модема серии V в сети КТСОП

Кодовая точка АСН.1	Вариант
<code>telephonyMode</code> (режим телефонии)	Терминал должен инициализировать процедуры разъединения, определенные в Рекомендациях для модема серии V, за исключением того, что он не должен физически разъединять соединение по КТСОП.
<code>v8bis</code>	Терминал должен инициализировать процедуры разъединения, определенные в Рекомендации для модема серии V, и начать сеанс согласно Рекомендации V.8 bis.
<code>v34DSVD</code>	Терминал должен сохранять соединение модема по V.34, но он должен использовать его для поддержки V.70.
<code>v34DuplexFAX</code>	Терминал должен сохранять соединение модема по V.34, но он должен использовать его для поддержки FAX по T.30 [27].
<code>v34H324</code>	Терминал должен сохранять соединение модема по V.34, но он должен использовать его для поддержки Рекомендации МСЭ-Т H.324 [24].

isdnOptions: Выбор вариантов, которые будут использоваться после окончания сеанса H.245, когда терминал цифрового соединения используется в цифровой сети.

Возможные варианты приведены в таблице В.15.

Таблица В.15/Н.245 – Варианты работы после команды EndSessionCommand при использовании терминала цифровой связи в цифровой сети

Кодовая точка АСН.1	Вариант
telephonyMode	Терминал должен инициализировать процедуры разъединения, определенные в Рекомендации, управляющей связью по конкретному цифровому каналу, к которому подключен данный терминал, за исключением того, что эти процедуры не должны физически отключать цифровое соединение.
v140	Терминал должен инициализировать процедуры разъединения, определенные в Рекомендации, управляющей связью по конкретному цифровому каналу, к которому подключен данный терминал, и начать сеанс V.140 [39].
terminalOnHold	Терминал должен инициализировать процедуры "удержания терминала", определенные в Рекомендации, управляющей связью по конкретному цифровому каналу, к которому подключен данный терминал.

В.13.5 Разные команды

Здесь описаны разнообразные команды, некоторые из которых присутствуют в Рекомендациях МСЭ-Т Н.221 [7] и Н.230 [13].

logicalChannelNumber указывает логический номер канала, к которому относится данная команда. Он должен показать логический канал, открытый для видеоданных в том случае, если используется один из следующих типов: videoFreezePicture, videoFastUpdatePicture, videoFastUpdateGOB, videoTemporalSpatialTradeOff, videoSendSyncEveryGOB, videoFastUpdateMB, videoSendSyncEveryGOBCancel, lostPicture, lostPartialPicture и recoveryReferencePicture. Если применяется один из следующих типов: equaliseDelay, zeroDelay, multipointModeCommand или cancelMultipointModeCommand, в которых участвует несколько логических каналов, то logicalChannelNumber должен быть произвольным, но действительным номером LogicalChannelNumber (то есть находиться в диапазоне от 1 до 65535), а получатель должен игнорировать его значение.

Параметры equaliseDelay и zeroDelay должны иметь то же значение, что и команды ACE и ACZ, определенные в Рекомендации МСЭ-Т Н.230 [13].

По командам multipointModeCommand ("команда многоточечного режима") принимающий терминал должен выполнять все запросы requestMode, передаваемые MCU. Примером изменения режима является изменение кодирования аудиосигнала с G.711 на G.728.

Команда cancelMultipointModeCommand ("отмена многоточечного режима") отменяет ранее переданную команду multipointModeCommand.

По команде videoFreezePicture ("стоп-кадр видеоданных") декодер видеосигнала прекращает обновлять текущий видеокادر и далее отображает стоп-кадр до получения соответствующего сигнала управления сбросом стоп-кадра.

По команде videoFastUpdatePicture ("ускоренное обновление видеосигнала") видеокодер при первой возможности переходит в режим ускоренного обновления видеосигнала.

По команде videoFastUpdateGOB ("ускоренное обновление группы блоков видеосигнала") видеокодер на удаленном конце выполняет быстрое обновление одной или нескольких групп блоков. Первая группа блоков firstGOB указывает номер первой группы блоков, которая должна обновляться, а numberOfGOBs ("число групп блоков") показывает, сколько групп блоков предстоит обновить. Эта команда должна использоваться только с алгоритмами сжатия видеосигнала, которые определяют группы блоков, например Н.261 и Н.263. Нумерация групп блоков производится так, как в Н.263, даже если применяется Н.261. Первая группа блоков изображения – это группа блоков под номером 0, вторая группа блоков – это группа блоков под номером 1 и т. д. Сканирование групп блоков для интерпретации параметра numberOfGOBs должно производиться согласно соответствующему стандарту кодирования видеосигнала, поэтому вторая группа блоков в кадре CIF Н.261 находится справа от первой, хотя она находится ниже первой группы блоков в изображении QCIF Н.261 и в случае использования Н.263.

По команде videoTemporalSpatialTradeOff ("соотношение временного и пространственного разрешения") видеокодер на удаленном конце изменяет свое соотношение между временным и пространственным разрешением. Значение 0 вызывает высокое пространственное разрешение, а значение 31 передает команду установить высокую частоту кадров. Значения от 0 до 31 показывают

монотонный рост желательности повышения частоты кадров. Фактические значения не соответствуют точным значениям пространственного разрешения или частоты кадров.

По команде `videoSendSyncEveryGOB` ("отправка синхроимпульса для каждой группы блоков видеосигнала") видеокодер на удаленном конце начинает использовать синхроимпульс для каждой группы блоков согласно определению в Рекомендации МСЭ-Т Н.263 [20] до тех пор, пока не будет получена команда `videoSendSyncEveryGOBCancel` ("отмена отправки синхроимпульса для каждой группы блоков видеосигнала"), после чего видеокодер на удаленном конце может принять решения относительно частоты синхроимпульсов группы блоков. Эти команды должны применяться только для видеосигнала, кодированного согласно Рекомендации МСЭ-Т Н.263.

По команде `videoFastUpdateMB` ("ускоренное обновление макроблоков видеосигнала") видеокодер на удаленном конце выполняет ускоренное изменение одного или нескольких макроблоков. `firstGOB` указывает номер первой группы блоков, подлежащей обновлению, `firstMB` показывает номер первого макроблока, который будет изменен, а `numberOfMBs` ("число макроблоков") показывает количество макроблоков, которые должны обновляться. Эта команда должна использоваться только с алгоритмами сжатия видеосигнала, которые определяют макроблоки, например Н.261 и Н.263. Терминалы могут отвечать на эту команду, обновляя группу блоков, включающую запрошенные макроблоки. Нумерация групп блоков производится так, как в Н.263, даже если используется Рекомендация Н.261. Первая группа блоков изображения – это группа блоков под номером 0, вторая группа блоков – это группа блоков под номером 1 и т. д. Должны присутствовать или `firstGOB`, или `firstMB`, или оба параметра вместе. Если присутствует `firstGOB` и отсутствует `firstMB`, то первый подлежащий обновлению макроблок – это первый макроблок указанной группы блоков. Если присутствует как `firstGOB`, так и `firstMB`, то `firstMB` показан относительно начала данной `firstGOB` так, что первый макроблок указанной группы блоков считается макроблоком номер 1. Если `firstGOB` отсутствует, а `firstMB` присутствует, `firstMB` относится к верхней левой части изображения, причем верхний левый макроблок рассматривается как макроблок номер 1. Порядок сканирования макроблоков в остальной части группы блоков и далее до этой точки определяется как порядок сканирования соответствующего стандарта кодирования видеосигнала; при этом порядок сканирования с началом в третьей группе блоков в изображении CIF Н.261 начинается с макроблока номер 1, который является макроблоком в левом столбце четвертой строки изображения, и сканирование продолжается вниз через три строки группы блоков, пока не будет достигнут макроблок номер 33 в одиннадцатом столбце шестой строки, а затем происходит скачок по вертикали, чтобы начать сканирование следующей группы блоков, начинающейся в двенадцатом столбце четвертой строки.

По команде `maxH223MUXPDUsize` передатчик ограничивает размер H223 MUX-PDU, которые он передает, максимальным указанным количеством октетов.

`encryptionUpdate` ("обновление шифрования") и `EncryptionUpdateRequest` ("запрос об обновлении шифрования") используются для инициализации и распределения нового материала ключей, которые должны использоваться при шифровании указанных медиаканалов.

МС может использовать включение и выключение принимающей среды в качестве команды конечной точке перейти с одноадресного на многоадресный канал и наоборот, когда МС+МР микширует аудиосигнал. Если в этом случае поток МС включает аудиосигнал терминала, то МС+МР могут переключать конечную точку на одноадресный поток, который будет содержать специальный микшированный сигнал для терминала с вырезанным аудиосигналом.

`switchReceiveMediaOff` ("выключение среды приема") используется МС для индикации конечной точке, что определенный логический канал не должен использоваться для приема.

`switchReceiveMediaOn` ("выключение среды приема") используется МС для индикации конечной точке, что определенный логический канал следует использовать для приема.

По команде `doOneProgression` видеокодер начинает создавать прогрессивную улучшающую последовательность. В этом режиме кодер формирует видеоданные, состоящие из одного изображения, за которым идет последовательность из нуля или большего количества кадров повышения качества того же самого изображения. Кодер остается в этом режиме до тех пор, пока он или не решит, что достигнут приемлемый уровень точности воспроизведения, или не будет получена команда `progressiveRefinementAbortOne` ("прекращение прогрессивного улучшения"). Кроме того, кодер должен включать тег начала сегмента прогрессивного улучшения и тег конца сегмента прогрессивного улучшения для того, чтобы пометить начало и конец прогрессивного улучшения, как определено в Дополнительной спецификации информации улучшения (Приложение L/Н.263).

По команде `doContinuousProgressions` ("выполнение непрерывного прогрессивного улучшения") видеокодер начинает создавать последовательности прогрессивного улучшения. В этом режиме кодер формирует видеоданные, состоящие из одного изображения, за которым следует

последовательность из нуля или более кадров улучшения качества того же самого изображения. Когда кодер решает, что достигнут приемлемый уровень точности воспроизведения или же когда получена команда `progressiveRefinementAbortOne`, кодер прекращает улучшение текущей прогрессии и начинает следующее прогрессивное уточнение другого изображения. Последовательность прогрессивных улучшений продолжается до получения команды `progressiveRefinementAbortContinuous` ("прекращение непрерывного прогрессивного улучшения"). Кроме того, кодер должен включать теги начала сегмента прогрессивного улучшения и теги конца сегмента прогрессивного улучшения для того, чтобы отметить начало и конец каждого прогрессивного улучшения, как определено в Дополнительной спецификации информации улучшения (Приложение L/H.263).

По команде `doOneIndependentProgression` видеокодер начинает независимую прогрессивную улучшающую последовательность. В этом режиме кодер формирует видеоданные, состоящие из одного видеокadra, формируемого методом внутрикадрового кодирования, за которым идет последовательность, состоящая из нуля или более кадров улучшения качества того же самого изображения. Кодер остается в этом режиме до тех пор, пока или он не решит, что достигнут приемлемый уровень точности воспроизведения, или не будет получена команда `progressiveRefinementAbortOne`. Кроме того, кодер должен включать теги начала сегмента прогрессивного улучшения и теги конца сегмента прогрессивного улучшения для того, чтобы отметить начало и конец прогрессивного улучшения, как определено в Дополнительной спецификации информации улучшения (Приложение L/H.263).

По команде `doContinuousIndependentProgressions` видеокодер начинает создавать независимые прогрессивные улучшающие последовательности. В этом режиме кодер формирует видеоданные, состоящие из одного видеокadra, формируемого методом внутрикадрового кодирования, за которым идет последовательность из нуля или более кадров улучшения качества того же самого изображения. Когда кодер решает, что достигнут приемлемый уровень точности воспроизведения, или же когда получена команда `progressiveRefinementAbortOne`, кодер прекращает улучшать текущую прогрессию и начинает следующее независимое прогрессивное улучшение для другого изображения. Последовательность независимых прогрессивных улучшений продолжается до получения команды `progressiveRefinementAbortContinuous` ("прекращение непрерывного прогрессивного улучшения"). Кроме того, терминал должен вставить теги начала сегмента прогрессивного улучшения и теги конца сегмента прогрессивного улучшения для того, чтобы отметить начало и конец каждого независимого прогрессивного улучшения, как определено в Дополнительной спецификации информации улучшения (Приложение L/H.263).

По команде `progressiveRefinementAbortOne` видеокодер завершает выполнение `doOneProgression`, `doOneIndependentProgression` или текущего прогрессивного улучшения в последовательности прогрессивных улучшений или в `doContinuousProgressions`, или в `doContinuousIndependentProgressions`. По команде `progressiveRefinementAbortContinuous` видеокодер завершает выполнение или `doContinuousProgressions`, или `doContinuousIndependentProgressions`.

Команда `videoBadMBs` ("плохие макроблоки видеосигнала") заставляет видеокодер на удаленном конце предпринимать корректирующие меры в том случае, если набор макроблоков не был принят должным образом. Кодер должен использовать эту информацию, чтобы выполнить действия по восстановлению качества видеосигнала. В отличие от команды `videoNotDecodedMBs` команда `videoBadMBs` не содержит никакого конкретного определения того, как именно декодер интерпретировал указанный набор макроблоков. При ответе на эту команду кодеру следует проверить, что указанный набор макроблоков не используется для предсказания видеоизображения, идущих после получения кодером данной команды. Не определяются конкретные действия, которые должен выполнить кодер, но это могут быть любые уместные коррективные действия, например передача кадра, формируемого методом внутрикадрового кодирования. Эта команда не должна передаваться видеодекодером, если соответствующий кодер на удаленном конце не указал возможность `videoBadMBsCap` ("возможность плохих макроблоков видеосигнала"). Эта команда должна использоваться только с алгоритмами кодирования видеосигнала, которые определяют макроблоки, например H.261, H.262, IS 11172 и H.263. Нумерация макроблоков производится согласно порядку растрового сканирования внутри изображения, причем верхний левый макроблок кадра определяется как макроблок номер 1, а номер макроблока увеличивается сначала слева направо, а затем сверху вниз.

По команде `lostPicture` ("потерянное изображение") видеокодер на удаленном конце выполняет коррективные действия, вызванные потерей или искажением указанных изображений. На них указывает или краткосрочный номер изображения `pictureNumber` или долговременный индекс изображения `longTermPictureIndex`. Кодер, поддерживающий Приложение U/H.263 (Улучшенный выбор эталонного изображения с удалением фрагментов изображения или без него) и/или W.6.3.12/H.263 (Номер изображения), должен иметь возможность понимать такие сообщения и предпринимать коррективные меры.

Команда `lostPartialPicture` ("потеря части изображения") заставляет видеокодер на удаленном конце выполнить коррективные действия в том случае, если набор макроблоков не был получен должным образом. Она совпадает с командой `videoBadMBs`, за исключением того, что изображение обозначается или посредством кратковременного номера изображения `pictureNumber`, или посредством долгосрочного индекса изображения `longTermPictureIndex`. Кодер, поддерживающий требования Приложения U/H.263 (Улучшенный выбор эталонного изображения с удалением фрагмента изображения или без него) и/или W.6.3.12/H.263 (Номер изображения), должен быть в состоянии понять такое сообщение и предпринять коррективные меры.

По команде `recoveryReferencePicture` ("восстановление эталонного изображения") кодер на удаленном конце начинает использовать для предсказания только указанные изображения. Они обозначены кратковременным номером изображения `pictureNumber` или долгосрочным индексом изображения `longTermPictureIndex`. Кодер, поддерживающий требования Приложения U/H.263 (Улучшенный выбор эталонного изображения с удалением фрагмента изображения или без него) и/или W.6.3.12/H.263 (Номер изображения), должен быть в состоянии понять такое сообщение и предпринять коррективные меры. Эта команда может быть передана декодером, который считает, что указанные изображения получены и декодированы правильно, а другие (неуказанные) изображения были искажены при передаче.

Команда `encryptionUpdateCommand` должна использоваться в Рекомендации МСЭ-Т H.235 для улучшенной процедуры обновления распределения материала новых ключей для сеанса (см. B.2.6.2/H.235). Команда `multiplePayloadStream` должна использоваться, только если должен повторно шифроваться поток, содержащий несколько полезных нагрузок, в этом случае должен игнорироваться динамический тип полезной нагрузки в `EncryptionSync`.

Подтверждение `encryptionUpdateAck` должно использоваться в Рекомендации МСЭ-Т H.235 для процедуры улучшенного обновления ключей для того, чтобы позволить ведомому терминалу подтвердить прием нового материала ключей для сеанса по логическому каналу, которым владеет ведущий терминал (см. B.2.6.2/H.235).

`direction` ("направление") должно показывать направление логического канала (`masterToSlave` или `slaveToMaster`), в котором передается материал ключей (см. B.2.6.2/H.235).

В.13.6 Команда организации конференции

Запрос `BroadcastMyLogicalChannel` должен быть аналогичен MCV H.230, используемому согласно процедуре, описанной в пункте 6.3.2.1/H.243, но он должен относиться к только одиночному логическому каналу. Обратите внимание на то, что при использовании предпочтительной процедуры MCV из пункта 6.3.2.2/H.243 (то есть когда на обоих концах линии между терминалом и MCU или между MCU и MCU имеется возможность `multipointVisualizationCapability`), вместо нее используется форма `conferenceRequest` запроса об осуществлении широковещательной передачи своего логического канала `BroadcastMyLogicalChannel`.

Запрос `CancelBroadcastMyLogicalChannel` должен быть аналогичен запросу `Cancel-MCV` H.230, но он должен относиться только к одиночному логическому каналу.

Запрос `MakeTerminalBroadcaster` должен быть определен как VCB H.230.

Запрос `CancelMakeTerminalBroadcaster` должен быть определен как `Cancel-VCB`.

Запрос `SendThisSource` должен быть определен как VCS H.230.

Запрос `CancelSendThisSource` должен быть определен как `Cancel-VCS` H.230.

Запрос `DropConference` должен быть определен как CCK H.230.

Команда `Substitute CID` ("замена идентификатора конференции") позволяет активному МС изменить идентификатор конференции (CID), фактически перемещая получателя этой команды в другую конференцию. Получатель этой команды должен использовать заново назначенный CID во всех будущих сообщениях с передачей сигнализации вызова.

В.13.7 Реконфигурации системы мультиплексирования H.223

По команде `h223ModeChange` передатчик меняет уровень режима мультиплексирования, соответствующий описанию в Приложении C/H.324, на уровень 0, уровень 1, уровень 2 или уровень 2 с необязательным заголовком, определенные в Приложении B/H.223.

По команде `h223AnnexADoubleFlag` передатчик начинает или кончает использовать режим двойного флага по Приложению A/H.223.

В.13.8 Команда открытия нового виртуального канала АТМ

Используется как команда удаленному терминалу открыть виртуальный канал АТМ с заданными параметрами.

resourceID используется для идентификации виртуального канала АТМ. В настоящей Рекомендации не определены средства, посредством которых этот параметр ассоциируется с виртуальным каналом АТМ.

bitRate показывает скорость передачи данных в виртуальном канале, определенной на AAL-SAP; она измеряется в единицах, составляющих 64 кбит/с.

bitRateLockedToPCRClock показывает, что скорость передачи виртуального канала синхронизируется по тактовым импульсам, которые используются для формирования значений временной отметки H.222.0 (временной отметки программы или временной отметки системы).

bitRateLockedToNetworkClock показывает, что скорость передачи виртуального канала синхронизируется по тактовым импульсам локальной сети. При этом не гарантирует, что тактовая частота скорости передачи данных будет синхронизирована с локальной сетью у приемника, поскольку общие тактовые импульсы сети могут отсутствовать.

Последовательность aal показывает, какой уровень адаптации АТМ будет использоваться, а также определяет его параметры.

Последовательность aal1 показывает, какие из опций уровня адаптации 1 АТМ, определенные в Рекомендации МСЭ-Т I.363 [25], поддерживаются. Соответствующие кодовые точки определены в таблице В.1.

Последовательность aal5 показывает, какие опции уровня адаптации АТМ 5, определенные в Рекомендации МСЭ-Т I.363 [25], поддерживаются. Параметры forwardMaximumSDUSize и backwardMaximumSDUSize задают максимальный размер CPCS-SDU в прямом и обратном направлениях, они измеряются в октетах.

multiplex показывает тип системы мультиплексирования, который будет использоваться в виртуальном канале АТМ. Возможны следующие варианты: noMultiplex ("нет системы мультиплексирования H.222.0"), H.222.0 Transport Stream ("транспортный поток H.222.0") и H.222.0 Program Stream ("программный поток H.222.0").

В.13.9 Команда изменения конфигурации подвижной многоканальной связи

Используется как команда передатчику изменить многоканальную кадровую конфигурацию, описанную в Приложении Н/Н.324.

sampleSize показывает размер отсчета в октетах. Отсчет – это количество октетов, которые будут распределены по доступным физическим каналам.

samplesPerFrame задает длину многоканальной полезной нагрузки в отсчетах.

status отображает статус получателя, когда он отправляет это командное сообщение. Если его статус равен synchronized ("синхронизован"), то это означает, что получатель организовал кадровую синхронизацию и передает команду отправителю начать передачу сжатого кадра заголовка. Если его статус равен reconfiguration ("реконфигурация"), то это соответствует команде отправителю изменить размер отсчетов и/или длину кадра и начать передавать полный кадр заголовка.

В.14 Индикация

Индикация содержит информацию, которая не требует действия или ответа.

В.14.1 Функция не понята

Эта индикация используется для возврата запросов, ответов и команд, которые не поняты отправителем.

Если терминал получает запрос, ответ или команду, которые он не понимает либо потому, что он нестандартен, либо потому, что он определен в более поздней версии настоящей Рекомендации, то в ответ ему следует передать сообщение FunctionNotSupported ("функция не поддерживается") или FunctionNotUnderstood ("функция не понята").

ПРИМЕЧАНИЕ. – Сообщение FunctionNotUnderstood называлась FunctionNotSupported в версии 1 настоящей Рекомендации. Название этой функции было изменено, чтобы добавить более мощную команду FunctionNotSupported без нарушения обратной совместимости с синтаксисом версии 1.

В.14.2 Разные версии индикации

Здесь описаны разные возможности индикации, некоторые ее варианты присутствуют в Рекомендациях МСЭ-Т Н.221 [7] и Н.230 [13].

Индикация `logicalChannelNumber` отображает номер логического канала, к которому относится данная индикация. Он должен включать логический канал, открытый для передачи видеоданных в том случае, если используется тип `videoIndicateReadyToActivate` и `videoTemporalSpatialTradeOff`. Если используется один из следующих типов: `multipointConference`, `cancelMultipointConference`, `multipointZeroComm`, `cancelMultipointZeroComm`, `multipointSecondaryStatus` или `cancelMultipointSecondaryStatus`, в которых задействовано несколько логических каналов, то `logicalChannelNumber` должен быть произвольным, но действительным числом `LogicalChannelNumber` (то есть он должен находиться в диапазоне от 1 до 65535), а получатель должен игнорировать его значение.

Индикация `logicalChannelInactive` ("неактивный логический канал") свидетельствует о том, что содержимое логического канала не соответствует нормальному сигналу. Эта индикация аналогична AIM и VIS, определенным в Рекомендации МСЭ-Т Н.230.

Индикация `logicalChannelActive` ("неактивный логический канал") дополняет индикацию `logicalChannelInactive`. Она аналогична AIA и VIA, определенным в Рекомендации МСЭ-Т Н.230. `MultipointZeroComm`, `cancelMultipointZeroComm`, `multipointSecondaryStatus` и `cancelMultipointSecondaryStatus` должны иметь то же значение, что и соответственно MIZ, `cancelMIZ`, MIS и `cancelMIS`, определенные в Рекомендации МСЭ-Т Н.230.

Индикация `multipointConference` ("многоточечная конференция") указывает, что терминал подключен к многоточечной конференции Н.243, и предполагается, что терминал будет подчиняться симметризации скоростей передачи данных. Однако симметризация скоростей передачи данных производится с помощью сообщений `FlowControlCommand`. Обратите внимание на то, что индикация `multipointConference` имеет точно такое же значение, что и MCC в Рекомендации МСЭ-Т Н.230. Отметим, что подобно MCC `multipointConference` не требует симметрии режима.

Индикация `videoIndicateReadyToActivate` должна иметь то же значение, что и VIR, определенное в Рекомендации МСЭ-Т Н.230, то есть она передается терминалом, чей пользователь решил не отправлять изображение до тех пор, пока он также не получит изображения с другого конца.

Индикация `videoTemporalSpatialTradeOff` показывает видеodeкодеру на удаленном конце свое текущее соотношение между временным и пространственным разрешением. Значение 0 означает высокое пространственное разрешение, а значение 31 показывает высокую частоту кадров. Рост значения от 0 до 31 соответствует монотонному повышению частоты кадров. Фактические значения не соответствуют точным значениям пространственного разрешения или частоты кадров. Терминал, который указал возможность `temporalSpatialTradeOffCapability`, должен передавать эту индикацию всякий раз, когда он изменяет это свое соотношение и когда первоначально открывается логический видеоканал.

Индикация `videoNotDecodedMBs` сообщает видеodeкодеру на удаленном конце о том, что набор макроблоков был получен ошибочно и что все макроблоки в указанном наборе были интерпретированы как незакодированные. Данный кодер может использовать эту информацию для компенсации ошибок при передаче, как показано в Добавлении I/Н.263. `firstMB` задает номер первого макроблока, интерпретированного как незакодированный, а `numberOfMBs` задает количество макроблоков, которые интерпретированы как незакодированные. Нумерация макроблоков производится таким образом, чтобы макроблок в верхнем левом углу изображения считался макроблоком номер 1, а номер каждого макроблока увеличивался слева направо и затем сверху вниз в порядке растрового сканирования (таким, что если в изображении всего имеется N макроблоков, то нижний правый макроблок считается макроблоком номер N). Временной эталон изображения, содержащего не декодированные макроблоки, дан в `temporalReference`. Эта индикация должна использоваться только с алгоритмом сжатия видеосигнала Н.263.

В.14.3 Индикация дрожания

Она используется для указания величины дрожания логического канала по оценке приемного терминала. Эта информация может быть полезна для выбора скорости передачи данных и управления буфером в видеоканалах или для определения подходящей скорости передачи синхронизирующей информации и т. д. При этом видеodeкодер получит возможность использовать эту информацию для ограничения колебаний буфера видеodeкодера и скорости передачи данных видеосигнала, что поможет предотвратить опустошение или переполнение буфера декодера, учитывая возникающее

дрожание. Если кодер использует эту возможность, то это обеспечит правильную работу существующих схем буферов видеodeкодера независимо от амплитуды принимаемого дрожания, а также позволит правильно работать с минимальной задержкой.

Если тип области действия – это logicalChannelNumber, то такая информация относится к данному логическому каналу, если это тип resourceID ("идентификатор ресурса"), то информация относится к указанному виртуальному каналу АТМ, а при использовании типа wholeMultiplex ("вся система мультиплексирования") информация относится ко всей системе мультиплексирования.

estimatedReceivedJitterMantissa ("оценка мантиссы дрожания при приеме") и estimatedReceivedJitterExponent ("оценка экспоненты дрожания при приеме") сообщают величину дрожания, принятую терминалом, который отправил сообщение.

estimatedReceivedJitterMantissa указывает мантиссу оценки дрожания в соответствии с определением таблицы В.16.

Таблица В.16/Н.245 – Мантисса estimatedReceivedJitterMantissa в JitterIndication

estimatedReceivedJitterMantissa	Мантисса
0	1
1	2,5
2	5
3	7,5

estimatedReceivedJitterExponent указывает экспоненту оценки дрожания в соответствии с определением таблицы В.17.

Таблица В.17/Н.245 – Экспонента estimatedReceivedJitterExponent в JitterIndication

estimatedReceivedJitterExponent	Экспонента
0	Вне диапазона
1	1 мкс
2	10 мкс
3	100 мкс
4	1 мс
5	10 мс
6	100 мс
7	1 с

Оценку дрожания получают путем умножения мантиссы на экспоненту, если estimatedReceivedJitterExponent не равна нулю, в противном случае известно, что оценка превышает 7,5 секунд.

skippedFrameCount показывает, сколько кадров было пропущено декодером после получения последнего сообщения JitterIndication. Поскольку в случае реализации этого варианта максимальная величина, которая может быть закодирована, равна 15, то данная информация должна передаваться прежде, чем будет пропущено более 15 кадров.

ПРИМЕЧАНИЕ. – Поскольку кадры пропускаются в случае опустошения буфера декодера, дополнительное дрожание может вызвать опустошение буфера декодера более часто или менее часто, чем должны происходить пропуски кадров согласно ожиданиям кодера.

additionalDecoderBuffer задает дополнительный размер буфера видеodeкодера, на который превышает объем, требуемый указанным профилем и уровнем. Он определяется так же, как vbv_buffer_size в Рекомендации МСЭ-Т Н.262 [19].

В.14.4 Индикация расфазировки H.223

Она используется для указания терминалу на удаленном конце средней временной расфазировки между двумя логическими каналами.

logicalChannelNumber1 и logicalChannelNumber2 – это номера логических каналов для открытых логических каналов.

skew ("расфазировка") измеряется в миллисекундах, она показывает задержку, которую необходимо учитывать для данных, относящихся к logicalChannelNumber2, при измерении на выходе системы мультиплексирования для обеспечения синхронизации с logicalChannelNumber1 при измерении на выходе системы мультиплексирования. Расфазировка включает различия во времени дискретизации, задержку кодера и задержку буфера передатчика, она измеряется относительно момента передачи первого бита данных для заданной точки отсчета. Необходимая для синхронизации фактическая задержка зависит от конкретной реализации декодера, она выбирается для получателя на местном уровне.

В.14.5 Индикация нового виртуального канала ATM

Используется для указания параметров виртуального канала ATM, который данный терминал намеревается открыть.

resourceID служит для идентификации виртуального канала ATM. В настоящей Рекомендации не указаны средства, с помощью которых этот параметр ассоциируется с виртуальным каналом ATM.

bitRate показывает скорость передачи виртуального канала, определенную на AAL-SAP, она измеряется в единицах по 64 кбит/с.

bitRateLockedToPCRClock свидетельствует о том, что скорость передачи виртуального канала синхронизирована с часами, применяемыми для создания значений временной отметки H.222.0 (временной отметки программы или временной отметки системы).

bitRateLockedToNetworkClock означает, что скорость передачи виртуального канала синхронизирована с часами локальной сети. При этом не гарантируется, что метки времени скорости передачи будут синхронизованы по часам локальной сети у приемника, поскольку общие временные метки для разных сетей могут быть недоступны.

Последовательность aal указывает, какой уровень адаптации ATM будет применяться, а также задает его параметры.

Последовательность aal1 указывает, какие из опций уровня адаптации 1 ATM, определенные в Рекомендации МСЭ-Т I.363 [25], поддерживаются. Кодовые точки определены в таблице В.1.

Последовательность aal5 указывает, какие из опций уровня адаптации 5 ATM, определенные в Рекомендации МСЭ-Т I.363 [25], поддерживаются. Параметры forwardMaximumSDUSize и backwardMaximumSDUSize задают максимальный размер CPCS-SDU в прямом и обратном направлениях; они измеряются в октетах.

multiplex показывает тип системы мультиплексирования, который будет использоваться в виртуальном канале ATM. Возможными вариантами являются noMultiplex ("отсутствие системы мультиплексирования H.222.0"), H.222.0 Transport Stream ("транспортный поток H.222.0") и H.222.0 Program Stream ("программный поток H.222.0").

В.14.6 Ввод данных пользователем

Используется для сообщений о вводе данных пользователем.

alphanumeric ("алфавитно-цифровой") – это строка символов, закодированных согласно Рекомендации МСЭ-Т T.51 [30]. Ее можно использовать для ввода с вспомогательной клавиатуры, что равносильно двухтональному многочастотному набору (DTMF).

userInputSupportIndication ("индикация поддержки ввода данных пользователя"): Сообщает удаленному терминалу, какие типы GENERALSTRING поддерживает данный терминал.

ПРИМЕЧАНИЕ 1. – Предполагается, что большинство реализаций PER декодеров не будут в состоянии декодировать строки, отличные от строк IA5. Данная индикация должна использоваться для того, чтобы "предостеречь" удаленный терминал от попыток использовать необычные схемы кодирования с переменной длиной кодирования.

Если DTMF передается посредством RTP с помощью userInputIndication в алфавитно-цифровой форме, то должно осуществляться его кодирование в последовательность extendedAlphanumeric и должен быть включен флаг rtpPayloadIndication.

nonStandard ("нестандартный") – это NonStandardParameter, показывающий нестандартное использование сообщения индикации UserInput.

Если булев параметр basicString имеет значение "истина", он показывает, что поддерживаются символы 0-9, # и *.

Если булев параметр iA5String имеет значение "истина", он показывает, что поддерживается весь набор символов IA5String.

Если булев параметр generalString имеет значение "истина", он показывает, что поддерживается весь набор символов GeneralString.

Если булев параметр encryptedBasicString имеет значение "истина", то он свидетельствует о том, что поддерживается зашифрованная базовая строка.

Если булев параметр encryptedIA5String имеет значение "истина", то он свидетельствует о том, что поддерживается зашифрованная строку IA5.

Если булев параметр encryptedGeneralString имеет значение "истина", то он свидетельствует о том, что поддерживается зашифрованная строка символов GeneralString.

В пункте 8.7/Н.235 описываются процедуры для зашифрованного DTMF Н.245 и порядок использования полей encryptedAlphanumeric внутри UserInputIndication (= зашифрованная базовая строка), encryptedSignalType внутри сигнала (= зашифрованная строка IA5) и encryptedAlphanumeric внутри extendedAlphanumeric (= зашифрованная базовая строка).

Индикацию **signal** (сигнал) и индикацию **signalUpdate** можно использовать в тех случаях, когда желателен точный контроль над регулировкой DTMF или сигналом отбоя при передаче аудиосигнала по ассоциированному логическому каналу и когда требуется контроль или индикация длительности DTMF.

signal ("сигнал") показывает сигнальный элемент, который должен создаваться при передаче на шлюз КТСОП (коммутируемой телефонной сети общего пользования), который был обнаружен в аудиопотоке при передаче от шлюза КТСОП или который должен передаваться между другими комбинациями конечных точек. При приеме шлюзом к КТСОП **signal** заставляет этот шлюз ввести указанный сигнальный элемент в канал КТСОП; при приеме шлюзом к другому терминалу серии Н **signal** будет преобразовываться в соответствующее сообщение в протоколе подключенного терминала. Шлюзы выдают сообщения **signal** (и **signalUpdate**) для индикации обнаружения сигнальных элементов в аудиосигнале, принятом от оконечной точки КТСОП или путем преобразования соответствующих сообщений из других протоколов.

signalType устанавливается равным "!" (восклицательный знак) для индикации сигнала отбоя или в один из символов "0123456789*#ABCD", чтобы обозначить тон DTMF.

ПРИМЕЧАНИЕ 2. – Сигнал отбоя – это условия мгновенного отбоя (обычно длительностью полсекунды), который широко применяется для управления функциями подключенного коммутационного оборудования. Может оказаться невозможным для шлюза создать или обнаружить сигнал отбоя из-за характеристик канала КТСОП или из-за локальной конфигурации (чтобы предотвратить нежелательную активизацию функций подключенного оборудования). Следовательно, способность передавать или принимать индикацию сигнала отбоя объявляется отдельно в возможности **UserInputCapability**.

duration показывает общую длительность тона, если она известна, или исходную оценку длительности тона, если тон продолжается в момент передачи сообщения **signal**. Если индикация **duration** опущена, то получатель должен использовать соответствующее присваиваемое по умолчанию значение, основанное на локальной конфигурации и требованиях к сети. Индикация **duration** должна игнорироваться в случае индикации сигнала отбоя ("!").

signalUpdate исправляет оценку общей длительности или объявляет фактическую измеренную длительность обнаруженного или подлежащего формированию тона. Следует передавать эту индикацию таким образом, чтобы она прибывала задолго до того, как истечет срок действия оценки, отправленной ранее в **signal** или **signalUpdate**; в противном случае исправленная величина длительности будет игнорироваться, поскольку передача сигнала тона будет уже прекращена получателем. Обратите внимание на отсутствие необходимости передавать индикацию **signalUpdate**, если полная длительность была указана в индикации **signal**.

rtp содержит параметры, необходимые для настройки тонального сигнала или сигнала отбоя в соответствии с потоком RTP/UDP (Н.323). Этот элемент должен быть включен в **signalUpdate** только в том случае, если было выдано много сигнальных сообщений с указанием различных LogicalChannelNumbers и требуется обозначить, какой сигнал подлежит обновлению.

timestamp задает (относительно временной метки RTP первичного кодера по ассоциированному аудиоканалу) время, когда должны формироваться тональный сигнал или сигнал отбоя

(распределяться или вводиться в аудиопоток). Тон или сигнал отбоя не должны генерироваться до того, как прозвучит аудиосигнал с той же временной меткой; они должны формироваться как можно скорее после этого момента, но не позднее временной метки **expirationTime** ("истечение срока"). Отправитель индикации этого не должен устанавливать **timestamp** равным времени, которое находится "в будущем"; обычно **timestamp** устанавливаются равным временной метке аудиосигнала, который в настоящий момент передается или был передан самым последним по ассоциированному аудиоканалу. Если **timestamp** не указана, то сигнал должен передаваться или вводиться после получения.

expirationTime указывает (относительно временной метки RTP первичного кодера по ассоциированному аудиоканалу) время, после которого тональный сигнал или сигнал отбоя должны считаться "устаревшими" и должны отбрасываться получателем. Конечные точки, которые получают **signal** и смогут действовать в ответ на него до временной метки **expirationTime** по ассоциированному каналу, должны отбрасывать это сообщение. Если время **expirationTime** не указано отправителем, то сообщение может тем не менее отбрасываться из-за локальной конфигурации получателя.

logicalChannelNumber должен задавать LogicalChannelNumber ("номер логического канала") ассоциированного аудиоканала – контекст, в котором имеют смысл **timestamp** и **expirationTime**.

МС должен преобразовывать временные метки и номер логического канала из полученной индикации в правильный номер логического канала и временные метки для каждого выходного канала, когда он направляет индикацию на каждую принимающую конечную точку (временные метки могут изменяться, если аудиосигнал транскодируется или микшируется в MP). МС, принимающий индикацию после времени **expirationTime**, может отбрасывать это сообщение немедленно, не пересылая его; в противном случае МС должен пересылать все запросы немедленно, не ожидая наступления времени **timestamp**.

Конечные точки должны использовать индикацию **alphanumeric** ("буквенно-цифровую") для того, чтобы передать введенные пользователем DTMF данные, если не указана способность другой конечной точки принимать DTMF посредством возможности **UserInputCapability**.

Конечная точка, которая имеет возможность принимать указания DTMF с помощью **signal**, также должна иметь возможность принимать индикацию **alphanumeric** для обеспечения совместимости с более старыми терминалами. Индикация **alphanumeric** может интерпретироваться как последовательность, состоящая из одной или нескольких индикаций **signal** с опущенными элементами **duration**, **timestamp** и **expirationTime**, а символы, недопустимые в **signalType**, отбрасываются.

Если DTMF передается посредством RTP согласно положениям 10.5/Н.323 и в UserInputIndication в виде сигнала, то должен включаться флаг rtpPayloadIndication.

При обычном использовании шлюз, обнаруживающий DTMF в аудиопотоке из канала ТКСОП, будет передавать **signal** немедленно после обнаружения тона, используя относительно высокую оценку **duration** (продолжительности), и начинать измерение продолжительности тона. Когда передача тона заканчивается, передается индикация **signalUpdate**, показывающая общую измеренную длительность. Если передача тона не закончилась, но измеренная длительность приближается к предыдущей оценке (таким образом, что измеренная длительность может превысить эту оценку до возможного получения **signalUpdate**), то передается **signalUpdate**, увеличивая эту оценку. На усмотрение разработчика конкретной реализации оставляются частота отправки **signalUpdate**, исходная оценка длительности, передаваемая в **signal**, и также величина, на которую увеличиваются последующие оценки; однако следует соблюдать осторожность, чтобы не загружать сеть большим количеством сообщений **signalUpdate** и избежать преждевременного прекращения действия предыдущих оценок.

В типичных случаях для конечной точки, которая не является шлюзом, элемент **signal** содержит общую длительность тона, который должен создаваться шлюзом. Однако в некоторых приложениях может быть желательно обеспечить интерактивное управление передаваемой пользователю длительностью тона в реальном времени. В этом случае **signal** и **signalUpdate** должны использоваться способом, аналогичным описанным для шлюзов в предыдущем абзаце, когда **signal** выдается после активизации ввода данных пользователем (например, нажатия клавиши или немедленного управления), используя оценку длительности тона, а **signalUpdate** используется для передачи обновленных оценок, пока ввод продолжает активизироваться, и для указания общей длительности, когда ввод деактивируется.

В.14.7 Индикация организации конференции

sbeNumber должен быть определен как номер SBE H.230.

terminalNumberAssign должен быть определен как TIA H.230.

terminalJoinedConference должен быть определен как TIN H.230.

terminalLeftConference должен быть определен как TID H.230.

seenByAtLeastOneOther должен быть определен как MIV H.230.

cancelSeenByAtLeastOneOther должен быть определен как cancel-MIV H.230.

seenByAll должен быть определен как MIV H.230.

cancelSeenByAll должен быть определен как MIV H.230.

terminalYouAreSeeing должен быть определен как VIN H.230.

requestForFloor должен быть определен как TIF H.230, он будет передаваться от терминала в МС.

WithdrawChairToken должен быть определен как CCR H.230, он должен передаваться от МС владельцу маркера председателя.

FloorRequested должен определяться как TIF H.230 в случае передачи от МС владельцу маркера. Этот запрос включает TerminalLabel запрашивающего терминала.

terminalYouAreSeeingInSubPictureNumber должен определяться как H.230 VIN2. subPictureNumber определяется как N в соответствии с указанием на рисунках 2-4/H.243.

videoIndicateCompose должен быть определен как VIC H.230. compositionNumber определяется как M в таблице 4/H.243.

В.14.8 Максимальная расфазировка между логическими каналами H2250

Индикация H2250MaximumSkewIndication показывает максимальную расфазировку между логическими каналами.

Расфазировка измеряется в миллисекундах, она задает максимальное число миллисекунд, на которые задерживаются данные в logicalChannelNumber2 относительно данных в logicalChannelNumber1, доставляемых сетевому транспорту. Расфазировка измеряется относительно времени доставки до сетевого транспорта первых бит данных для заданной точки отсчета. Синхронизация губ, если она желательна, является локальным вопросом для получателя, она должна обеспечиваться путем использования временных меток.

В.14.9 Индикация местоположения МС

Эта индикация передается МС для того, чтобы сообщить другим терминалам адрес, который следует использовать для сигнализации на МС.

В.14.10 Индикация идентификатора изготовителя

Индикация vendorIdentification должна передаваться в начале каждого вызова для идентификации изготовителя, изделия и номера версии изделия.

В.14.11 Функция не поддерживается

Эта индикация используется для возврата отправителю непонятых запросов, ответов и команд.

Возвращается все сообщение с запросом RequestMessage, ResponseMessage или CommandMessage.

Если терминал получает запрос, ответ или команду, которую он не понимает или потому, что они являются нестандартными, или они были определены в более поздней версии настоящей Рекомендации, то он должен в ответ отправить сообщение FunctionNotSupported.

Если терминал получает запрос, ответ или команду с неправильным кодированием, то он должен установить в качестве причины значение `syntaxError` ("синтаксическая ошибка"). Если кодировка правильна, но закодированные значения семантически неверны, то он должен установить в качестве причины значение `semanticError` ("семантическая ошибка"). Если данное сообщение является непризнанным расширением сообщения `MultimediaSystemControlMessage`, `RequestMessage`, `ResponseMessage` или `CommandMessage`, то он должен установить значение причины равным `unknownFunction` ("неизвестная функция").

В каждом случае следует возвращать все сообщение `MultimediaSystemControlMessage` ("сообщение об управлении мультимедийной системой") в виде строки октетов в `returnedFunction` ("возврат функции").

Сообщение `FunctionNotSupported` не должно использоваться в каких-либо других случаях. В частности, индикация `FunctionNotSupported` не должна использоваться при наличии непризнанного расширения в других точках в синтаксисе: терминал должен отвечать на сообщение обычным образом, так, как в случае отсутствия расширения. Индикация `FunctionNotSupported` никогда не должна передаваться в ответ на полученную индикацию.

В.14.12 Индикация с информацией управления потоком данных

Эта индикация используется, чтобы сообщить удаленному терминалу, чтобы данный терминал изменил свою максимальную скорость передачи данных в ответ на входящую команду `FlowControlCommand` ("команда управления потоком") или в связи с тем, что терминал желает скорректировать свою исходящую скорость передачи. Такая индикация позволяет терминалу сигнализировать о любом изменении исходящей максимальной скорости передачи данных при соблюдении ограничений, обусловленных верхними пределами, которые определяются возможностями открытого логического канала и терминала.

Любому терминалу, получающему команду `FlowControlCommand` ("команда управления потоком"), следует отвечать выдачей индикации `FlowControl` ("индикация с информацией об управлении потоком"), чтобы сообщить новую установленную максимальную скорость передачи данных.

Поля сообщения `FlowControlIndication` имеют то же самое значение, что и поля с тем же названием в команде `FlowControlCommand`.

В.14.13 Индикация изменения мобильной многоканальной конфигурации

Применяется для оповещения получателя о том, что отправитель изменит размер отсчетов и/или число отсчетов на кадре в заголовке информационного кадра согласно описанию в Приложении Н/Н.324. Эту индикацию можно передавать в режиме полного заголовка, ее не следует передавать при использовании режима сжатого заголовка.

`sampleSize` показывает размер отсчета в октетах. Отсчет – это количество октетов, которые будут распределяться по имеющимся физическим каналам.

`samplesPerFrame` показывает длину многоканальной полезной нагрузки в отсчетах.

В.15 Универсальные сообщения

Тип **GenericMessage** разрешает определять новые элементы `RequestMessage`, `CommandMessage`, `ResponseMessage` и `IndicationMessage` таким образом, чтобы не требовалось издавать новую версию синтаксиса Н.245. Этот метод позволяет определять как стандартные, так и нестандартные сообщения.

ПРИМЕЧАНИЕ 1. – Структуры `GenericMessage`, определенные в настоящей Рекомендации, должны быть перечислены в приложениях к настоящей Рекомендации. Структуры `GenericMessage`, определенные в других Рекомендациях МСЭ-Т, должны упоминаться в дополнении к настоящей Рекомендации. Структуры `GenericMessage`, определенные вне документов МСЭ-Т, могут издаваться в любой подходящей форме.

Поле **messageIdentifier** показывает уникальный тип сообщения. Основанные на документах МСЭ-Т идентификаторы сообщения должны использовать стандартный `OBJECT IDENTIFIER` ("идентификатор объекта"), в то время как идентификаторы, основанные на других стандартах, и собственные идентификаторы сообщения должны использовать одно из следующего: `standard` ("стандартный"), `h221NonStandard`, `uuid` и `domainBased`, в зависимости от потребности.

Дополнительное поле **subMessageIdentifier** указывает под-сообщение, связанное с `messageIdentifier`.

Поле **messageContents** указывает параметры данного сообщения.

Чтобы избежать неоднозначности и проблем взаимодействия, идентификатор **standard ParameterIdentifier** со значением 0 не должен определяться для использования в поле `messageContents`.

ПРИМЕЧАНИЕ 2. – Некоторые Рекомендации определяют автоматические процедуры преобразования параметров GenericParameters из системы сигнализации H.245 в систему сигнализации кода BAS, используемую в Рекомендации МСЭ-Т H.320. Эти процедуры используют значение 0 вместо стандартного ParameterIdentifier в качестве специального сигнала, разграничивающего конец списка элементов GenericParameter.

Поле genericRequest – это универсальное сообщение GenericMessage, используемое для отправки универсального сообщения запроса RequestMessage.

Поле genericResponse – это универсальное сообщение, используемое для отправки универсального сообщения ответа ResponseMessage.

Поле genericCommand – это универсальное сообщение, используемое для отправки универсального сообщения команды CommandMessage.

Универсальное поле Indication – это универсальное сообщение, используемое для отправки универсального сообщения индикации IndicationMessage.

Приложение С

Процедуры

С.1 Введение

В этом приложении определяются универсальные процедуры управления мультимедийными системами, которые используют сообщения, определенные в настоящей Рекомендации. В Рекомендациях, использующих данную Рекомендацию, должно быть указано, какие из этих процедур можно применять, а также в них должны быть определены все особые требования.

В этом разделе приведено описание процедур, выполняющих следующие функции:

- определение отношения "ведущий-ведомый";
- обмен возможностями терминалов;
- сигнализация по однонаправленному логическому каналу;
- сигнализация по двунаправленному логическому каналу;
- запрос принимающего терминала о закрытии логического канала;
- модификация элементов таблицы мультиплексирования H.223;
- запрос элемента таблицы мультиплексирования;
- запрос режима передачи от получателя к отправителю;
- определение задержки из-за подтверждения приема;
- цикл обслуживания.

С.1.1 Метод спецификации

В этом приложении процедуры обычно определяются с помощью языка описания структур SDL. SDL позволяет задать процедуры в графическом виде и включать спецификации действий в исключительных ситуациях.

С.1.2 Связь между протокольным объектом и пользователем протокола

Взаимодействие с пользователем конкретной функции определяется через примитивы, передаваемые по интерфейсу между протокольным объектом и пользователем протокола. Примитивы предназначены для определения процедур протокола; не предполагается, что они будут задавать или ограничивать конкретную реализацию. С каждым примитивом может быть ассоциировано несколько параметров.

Для упрощения спецификации определены состояния протокола. Эти состояния носят концептуальный характер, они отражают общие условия протокольного объекта в последовательностях примитивов, которыми обмениваются протокольный объект и пользователь, а также обмен сообщениями между протокольным объектом и равноправным с ним объектом.

Для каждого протокольного объекта с помощью диаграммы переходов определяется разрешенная последовательность примитивов между пользователем и протокольным объектом. Разрешенная последовательность ограничивает действия пользователя и определяет возможные ответы от протокольного объекта.

Параметр примитива, описанный как нулевой (null), эквивалентен параметру, который не присутствует.

С.1.3 Связь между равноправными объектами

Информация протокола передается равноправному протокольному объекту в соответствующих сообщениях, определенных в Приложении А. Некоторые описанные протокольные объекты имеют ассоциированные с ними переменные состояния. Кроме того, с некоторыми описанными протокольными объектами ассоциируются соответствующие таймеры.

Таймер идентифицируется нотацией T_n , где n – номер. В диаграммах SDL установка таймера означает, что в таймер записано определенное значение и что этот таймер запускается. Переустановка таймера означает, что таймер остановлен и что сохраняется его значение в момент переустановки. Истечение таймера означает, что таймер работал в течение установленного времени и достиг нулевого значения.

Кроме того, протокольный объект может иметь ассоциированные параметры. Параметр определяется нотацией N_n , где n – номер.

Эти таймеры и счетчики перечислены в Добавлении III.

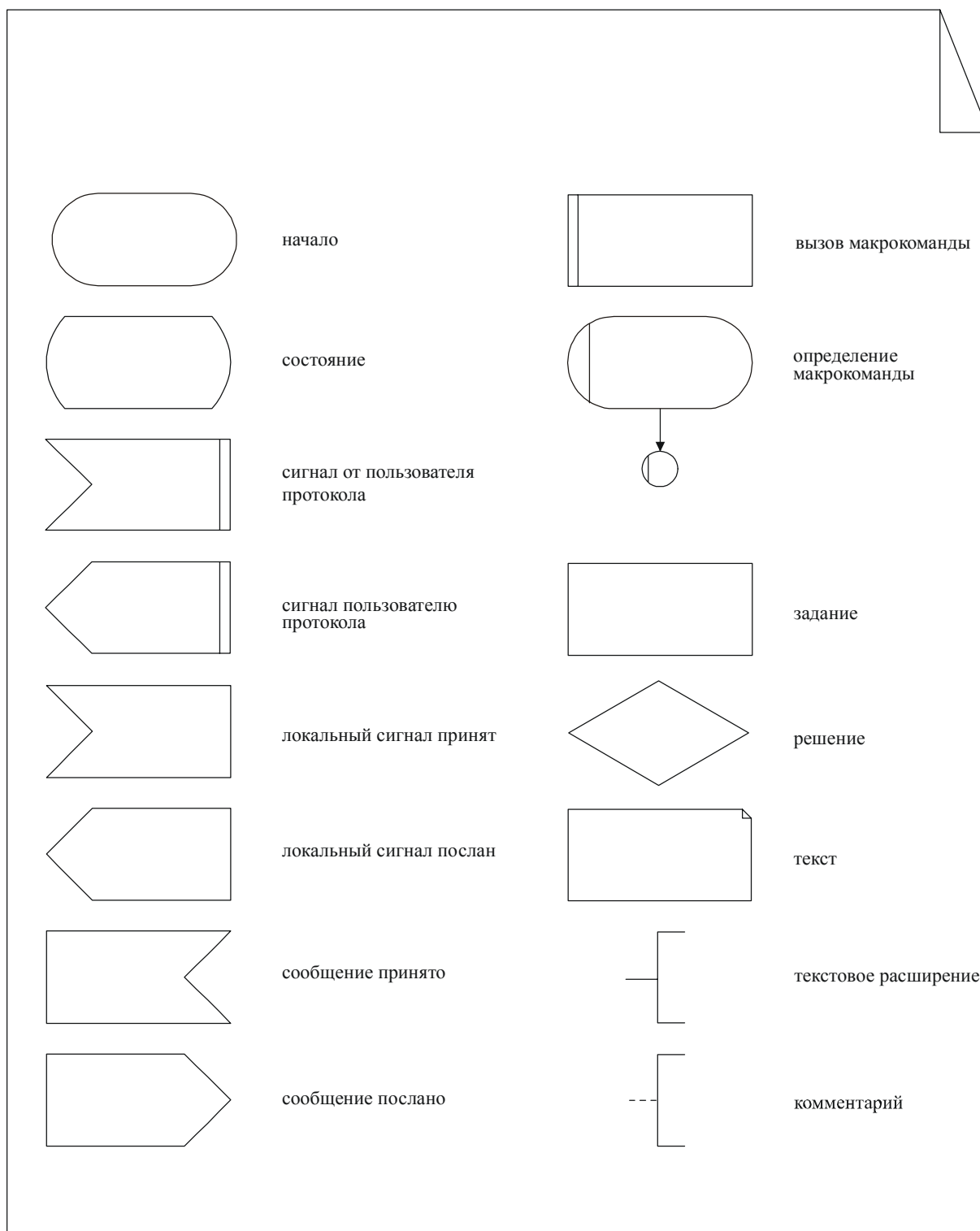
Некоторые протокольные объекты определяют примитив ошибки для сообщения управляющему объекту об условиях возникновения ошибки в протоколе.

С.1.4 Диаграммы языка описания структур SDL

В диаграммах SDL показаны действия, предпринимаемые в ответ на разрешенное взаимодействие с пользователем протокола и на прием сообщений от равноправного протокольного объекта. Примитивы, которые не разрешены для данного состояния, определенного диаграммами переходов между состояниями, не приводятся на диаграммах SDL. Однако на диаграммах SDL приводится описание ответов в случае получения неподходящих сообщений.

С.1.5 Основные обозначения языка описания структур SDL

Основные обозначения SDL показаны на рисунке С.1.



H.245_FC.1

Рисунок С.1/Н.245 – Основные обозначения языка описания структур SDL

С.2 Процедуры определения отношения "ведущий-ведомый"

С.2.1 Введение

Если два или более терминалов, участвующих в установлении вызова, одновременно инициируют похожие события при условии, что ресурсов достаточно только для наступления одного такого события (например, для открытия логических каналов), то возможно возникновение конфликтов. Для разрешения этих конфликтов один терминал может действовать как ведущий, а другой терминал (другие терминалы) могут действовать как ведомый терминал (терминалы). Описанные здесь

процедуры позволяют терминалам при организации вызова определить, какой терминал является ведущим, а какой терминалом (какие терминалы) является ведомым (ведомыми).

Описываемый здесь протокол называется ниже Объект сигнализации для определения отношения "ведущий-ведомый" (MSDSE). Для каждого терминала, участвующего в вызове, имеется один экземпляр MSDSE.

Любой терминал может инициализировать процесс определения отношения "ведущий-ведомый", передавая своему MSDSE примитив DETERMINE.request. Результат выполнения этой процедуры возвращается посредством примитивов DETERMINE.indication и DETERMINE.confirm. Несмотря на то, что примитив DETERMINE.indication указывает результат, он не означает, что этот результат известен удаленному терминалу. Примитив DETERMINE.confirm содержит результат и подтверждает, что он также известен удаленному терминалу. Терминал может инициировать процесс определения отношения "ведущий-ведомый" только в том случае, если нет никакой локально активной процедуры, которая зависит от его результата.

Терминал должен реагировать на процедуры, которые основаны на знании результата и инициируются удаленным терминалом в любое время после того, как результат определения статуса станет известен локальному терминалу. Это может произойти до получения локальным терминалом подтверждения того, что удаленный терминал также знает этот результат. Терминал не должен инициализировать процедуры, которые зависят от знания этого результата, пока он не получил подтверждения, что удаленный терминал также знает результат данного применения процедуры определения отношения "ведущий-ведомый".

В приведенном ниже тексте содержится общее описание действия этого протокола. В случае каких-либо расхождений между ним и официальной спецификацией протокола, которая приведена ниже, преимущественную силу имеет официальная спецификация.

С.2.1.1 Общее описание протокола – инициирование локальным пользователем

Процедура определения отношения "ведущий-ведомый" инициируется при выдаче пользователем MSDSE примитива DETERMINE.request. Сообщение MasterSlaveDetermination (определение отношения "ведущий-ведомый") передается равноправному объекту MSDSE, и запускается таймер T106. Если в ответ на сообщение MasterSlaveDetermination получено сообщение MasterSlaveDeterminationAck ("подтверждение определения отношения "ведущий-ведомый"), то таймер T106 останавливается и пользователь информируется посредством примитива DETERMINE.confirm о том, что процедура определения отношения "ведущий-ведомый" успешно завершена, и одноранговому MSDSE передается сообщение MasterSlaveDeterminationAck. Однако если в ответ на сообщение MasterSlaveDetermination получено сообщение MasterSlaveDeterminationReject, то формируется новый номер определения статуса, таймер T106 перезапускается и передается еще одно сообщение MasterSlaveDetermination. Если после передачи сообщения MasterSlaveDetermination N100 раз все еще не было получено сообщение MasterSlaveDeterminationAck, то таймер T106 останавливается, а пользователя информируют при помощи примитива REJECT.indication о том, что процедуре определения отношения "ведущий-ведомый" не удалось достичь результата.

Если заканчивается срок действия таймера T106, то пользователя MSDSE информируют об этом с помощью примитива REJECT.indication, а равноправному MSDSE передается сообщение MasterSlaveDeterminationRelease (сброс определения отношения "ведущий-ведомый").

С.2.1.2 Общее описание протокола – инициирование удаленным пользователем

Когда MSDSE получает сообщение MasterSlaveDetermination, инициализируется процедура определения статуса. Если процедура определения статуса возвращает определенный результат, то пользователь с помощью примитива DETERMINE.indication информируется о результате определения отношения "ведущий-ведомый", равноправному MSDSE передается сообщение MasterSlaveDeterminationAck и запускается таймер T106. Если сообщение MasterSlaveDeterminationAck получено в ответ на сообщение MasterSlaveDeterminationAck, то таймер T106 останавливается и посредством примитива DETERMINE.confirm пользователя извещают о том, что процедура определения отношения "ведущий-ведомый" успешно завершена.

Если заканчивается срок действия таймера T106, то пользователю MSDSE сообщается об этом посредством примитива REJECT.indication.

Однако если процедура определения состояния возвращает неопределенный результат, то равноправному MSDSE передается сообщение MasterSlaveDeterminationReject.

С.2.1.3 Общее описание протокола – одновременное инициирование

Когда сообщение MasterSlaveDetermination получено в MSDSE, который сам уже инициировал процедуру определения статуса и ожидает сообщение MasterSlaveDeterminationAck или MasterSlaveDeterminationReject, то иницируется процедура определения статуса. Если эта процедура определения статуса возвращает определенный результат, то MSDSE отвечает так, как если бы эта процедура была инициирована удаленным пользователем, и применяются описанные выше процедуры для такого случая.

Однако если процедура определения статуса возвращает неопределенный результат, то формируется новый номер определения статуса и MSDSE реагирует так, как если бы процедура снова инициировалась локальным пользователем MSDSE согласно приведенному выше описанию.

С.2.1.4 Процедура определения статуса

Следующая процедура используется для определения того, какой терминал является ведущим по значениям terminalType ("тип терминала") и statusDeterminationNumber ("номер определения статуса"). Сначала сравниваются значения terminalType, и терминал с большим номером типа терминала определяется как ведущий. Если номера типа терминалов одинаковы, то для определения ведущего терминала сравниваются значения statusDeterminationNumbers, используя арифметику по модулю.

Если значения полей terminalType у обоих терминалов равны, а разность между значениями полей statusDeterminationNumber по модулю 2^{24} равна 0 или 2^{23} , то получен неопределенный результат.

С.2.2 Связь между MSDSE и пользователем MSDSE

С.2.2.1 Примитивы, которыми обмениваются MSDSE и пользователь MSDSE

Связь между MSDSE и пользователем MSDSE осуществляется при помощи примитивов, показанных в таблице С.1.

Таблица С.1/Н.245 – Примитивы и параметры

Универсальное имя	тип			
	запрос	индикация	ответ	подтверждение
DETERMINE	– (Примечание 1)	TYPE	Не определен (Примечание 2)	TYPE
REJECT	Не определен	–	Не определен	Не определен
ERROR	Не определен	ERRCODE	Не определен	Не определен

ПРИМЕЧАНИЕ 1. – "-" означает отсутствие параметров.
ПРИМЕЧАНИЕ 2. – "Не определен" означает, что примитив не определен.

С.2.2.2 Определение примитивов

Ниже приведены определения этих примитивов:

а) Примитив DETERMINE используется для инициирования и возвращения результата процедуры определения отношения "ведущий-ведомый".

Примитив DETERMINE.request используется для инициирования процедуры определения отношения "ведущий-ведомый".

Примитив DETERMINE.indication используется для индикации результата процедуры определения отношения "ведущий-ведомый". Поскольку результат процедуры может быть неизвестен удаленному терминалу, данный терминал не должен инициировать никакие процедуры, которые основаны на знании этого результата, хотя он должен отвечать на любые процедуры, в основе которых лежит знание этого результата.

Примитив DETERMINE.confirm используется для индикации результата у процедуры определения отношения "ведущий-ведомый", а также того факта, что результат этой

процедуры известен обоим терминалам. Терминал может инициировать любые процедуры, в основе которых лежит знание этого результата, и должен отвечать на них.

- b) Примитив REJECT показывает, что процедура определения отношения "ведущий-ведомый" была неудачной.
- c) Примитив ERROR сообщает об ошибках MSDSE управляющему объекту.

С.2.2.3 Определение параметров

Ниже приведены определения параметров примитивов, показанных в таблице С.1.

- a) Параметр TYPE показывает статус терминала. Он принимает значения "MASTER" ("Ведущий") или "SLAVE" ("Ведомый").
- b) Значение ERRCODE показывает тип ошибки MSDSE. В таблице С.5 показаны значения, которые может принимать параметр ERRCODE.

С.2.2.4 Состояния MSDSE

Приведенные ниже состояния используются для определения разрешенной последовательности примитивов между MSDSE и пользователем MSDSE.

Состояние 0: IDLE

Процедура определения отношения "ведущий-ведомый" не была инициирована.

Состояние 1: OUTGOING AWAITING RESPONSE

Локальный пользователь MSDSE запросил процедуру определения отношения "ведущий-ведомый". Ожидается ответ от удаленного MSDSE.

Состояние 2: INCOMING AWAITING RESPONSE

Удаленный MSDSE инициировал процедуру определения отношения "ведущий-ведомый" в локальном MSDSE. Подтверждение было передано удаленному MSDSE, и ожидается ответ от удаленного MSDSE.

С.2.2.5 Диаграмма переходов между состояниями

Здесь определена разрешенная последовательность примитивов между MSDSE и пользователем MSDSE. Разрешенные последовательности показаны на рисунке С.2.

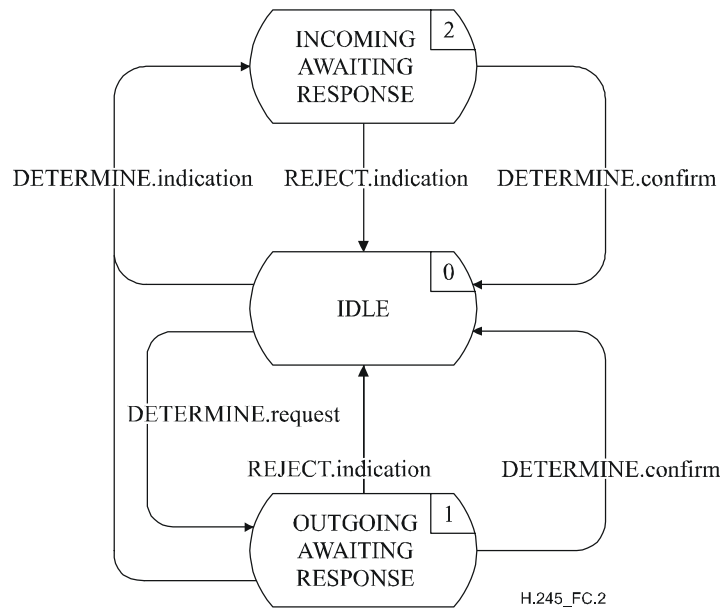


Рисунок С.2/Н.245 – Диаграмма перехода между состояниями для последовательности примитивов в MSDSE

С.2.3 Связь между равноправными объектами MSDSE

С.2.3.1 Сообщения MSDSE

В таблице С.2 приведены сообщения MSDSE и поля, определенные в Приложении А, которые относятся к протоколу MSDSE.

Таблица С.2/Н.245 – Названия и поля сообщений MSDSE

Функция	Сообщение	Поле
determination (определение)	MasterSlaveDetermination	terminalType statusDeterminationNumber
	MasterSlaveDeterminationAck	decision (решение)
	MasterSlaveDeterminationReject	cause (причина)
error recovery	MasterSlaveDeterminationRelease	–

С.2.3.2 Переменные состояния MSDSE

Определены следующие переменные состояния MSDSE:

sv_TT

Эта переменная состояния содержит номер типа терминала для данного терминала.

sv_SDNUM

Эта переменная состояния содержит номер определения статуса для данного терминала.

sv_STATUS

Эта переменная состояния используется для записи результата самой последней процедуры определения отношения "ведущий-ведомый". Она имеет значения "master" ("ведущий"), "slave" ("ведомый") и "indeterminate" ("неопределенный").

sv_NCOUNT

Эта переменная состояния используется для хранения результата количества сообщений MasterSlaveDetermination, которые были отправлены при сохранении состояния OUTGOING AWAITING RESPONSE.

С.2.3.3 Таймеры MSDSE

Для передающего MSDSE определен следующий таймер:

T106

Этот таймер используется во время состояния OUTGOING AWAITING RESPONSE и во время состояния INCOMING AWAITING RESPONSE. Он определяет максимальное разрешенное время без получения сообщения с подтверждением.

С.2.3.4 Счетчики MSDSE

Для MSDSE задан следующий параметр:

N100

Этот параметр указывает максимальное значение sv_NCOUNT.

С.2.4 Процедуры MSDSE

С.2.4.1 Введение

На рисунке С.3 показаны все примитивы MSDSE и их параметры, а также сообщения.

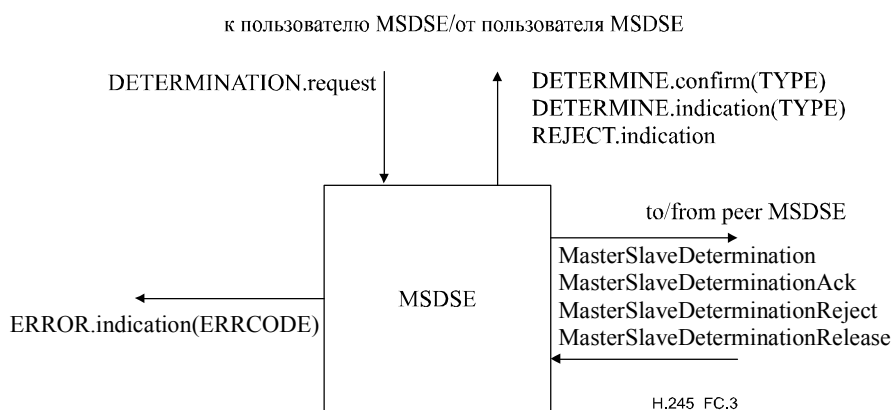


Рисунок С.3/Н.245 – Примитивы и сообщения в MSDSE

С.2.4.2 Принимаемые по умолчанию значения параметров примитивов

Если в диаграммах SDL в явном виде не установлены другие значения, то параметры индикации и примитивы квитирования имеют значения, приведенные в таблице С.3.

Таблица С.3/Н.245 – Принимаемые по умолчанию значения параметров примитивов

Примитив	Параметр	Значение по умолчанию
DETERMINE.confirm	TYPE (Тип)	MasterSlaveDeterminationAck.decision
DETERMINE.indication	TYPE	sv_STATUS

С.2.4.3 Принимаемые по умолчанию значения полей сообщений

Если в диаграммах SDL в явном виде не установлены другие значения, то поля сообщений принимают значения, показанные в таблице С.4.

Таблица С.4/Н.245 – Принимаемые по умолчанию значения полей сообщений

Сообщение	Поле	Значение по умолчанию
MasterSlaveDetermination	terminalType statusDeterminationNumber	sv_TT sv_SDNUM
MasterSlaveDeterminationAck	decision (решение)	Противоположно значению sv_STATUS т. е., if(sv_STATUS == master) decision = slave if(sv_STATUS == slave) decision = master
MasterSlaveDeterminationReject	cause (причина)	identicalNumbers

С.2.4.4 Значения параметров ERRCODE

В таблице С.5 приведены значения, которые может принимать параметр ERRCODE примитива ERROR.indication для MSDSE.

Таблица С.5/Н.245 – Значения параметров ERRCODE в MSDSE

Тип ошибки	Код ошибки	Условие ошибки	Состояние
нет ответа от удаленного MSDSE	A	Истек срок действия локального таймера T106	OUTGOING AWAITING RESPONSE INCOMING AWAITING RESPONSE
удаленный терминал не видит ответа от локального MSDSE	B	Истек срок действия удаленного таймера T106	OUTGOING AWAITING RESPONSE INCOMING AWAITING RESPONSE
несоответствующее сообщение	C	MasterSlaveDetermination	INCOMING AWAITING RESPONSE
	D	MasterSlaveDeterminationReject	INCOMING AWAITING RESPONSE
неверное значение поля	E	MasterSlaveDeterminationAck.decision != sv_STATUS	INCOMING AWAITING RESPONSE
максимальное количество повторных попыток	F	sv_NCOUNT == N100	OUTGOING AWAITING RESPONSE

С.2.4.5 Диаграммы языка описания структур (SDL)

Процедуры MSDSE представлены в форме SDL на рисунке С.4.

terminalTypeProcess является процессом, возвращающим номер, который идентифицирует различные типы терминалов, например терминалы, MCU и шлюзы.

randomNumber является процессом, который возвращает случайное число в диапазоне $0 \dots 2^{24} - 1$.

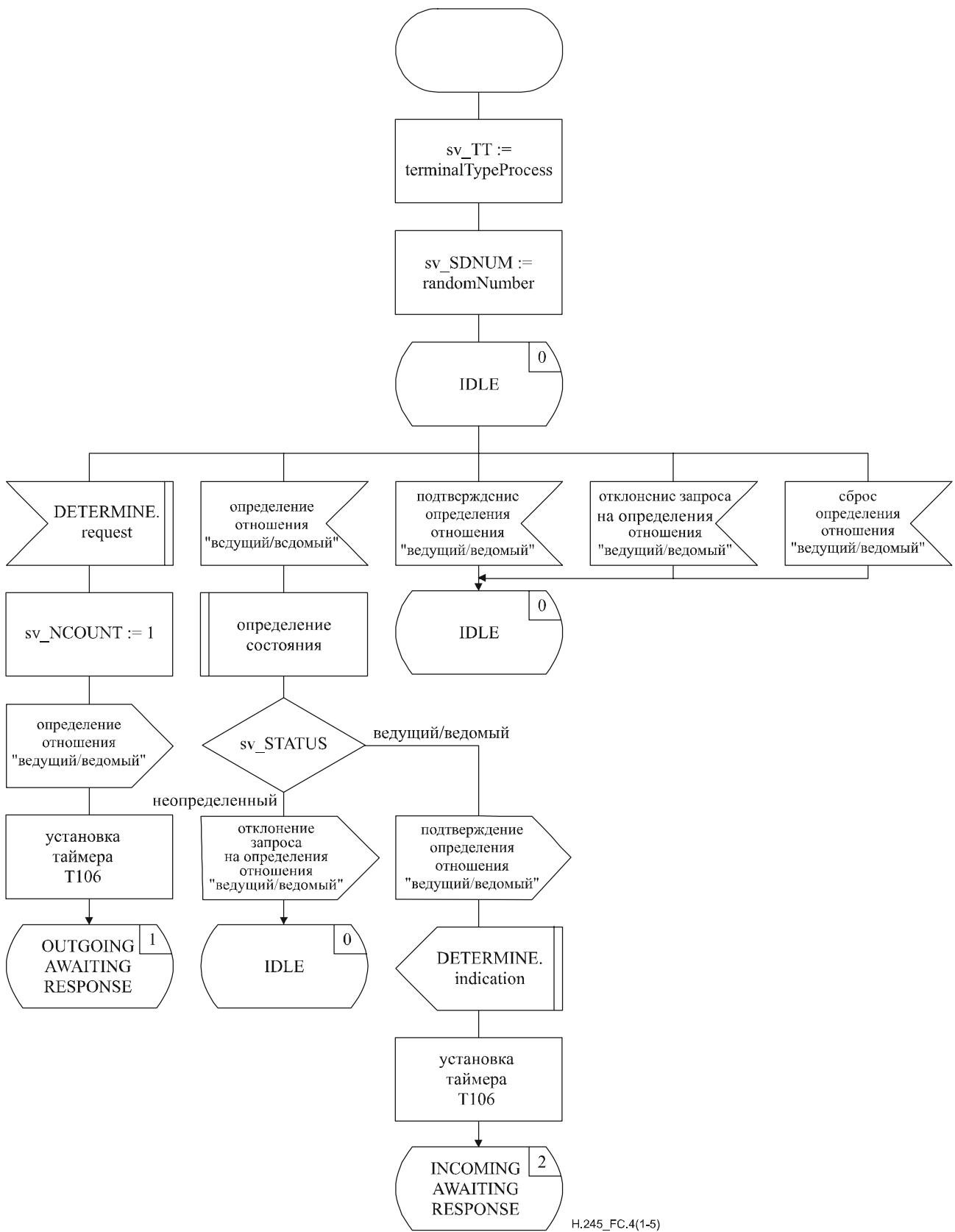


Рисунок С.4/Н.245 – Диаграмма SDL для MSDSE (лист 1 из 5)

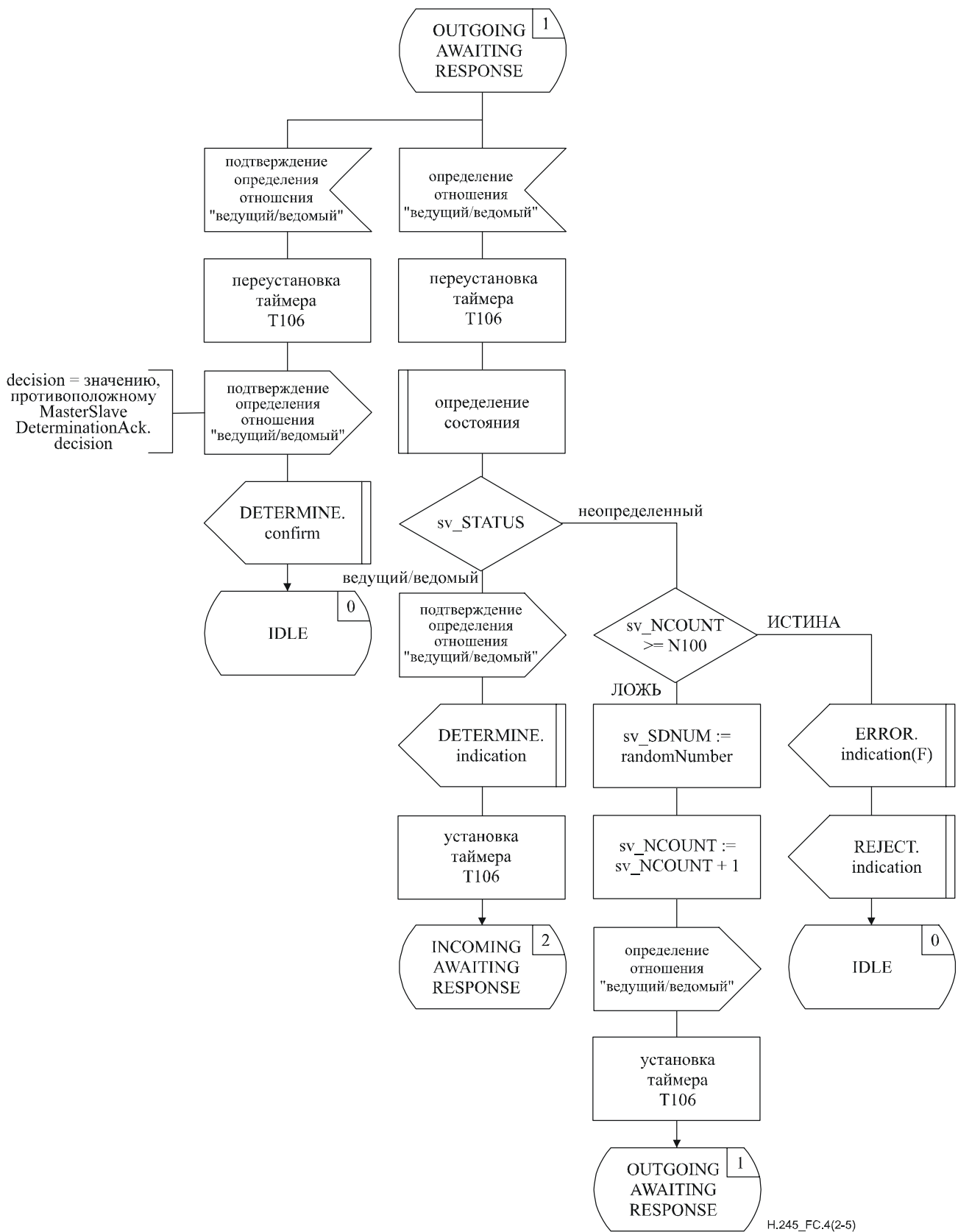


Рисунок С.4/Н.245 – Диаграмма SDL для MSDSE (лист 2 из 5)

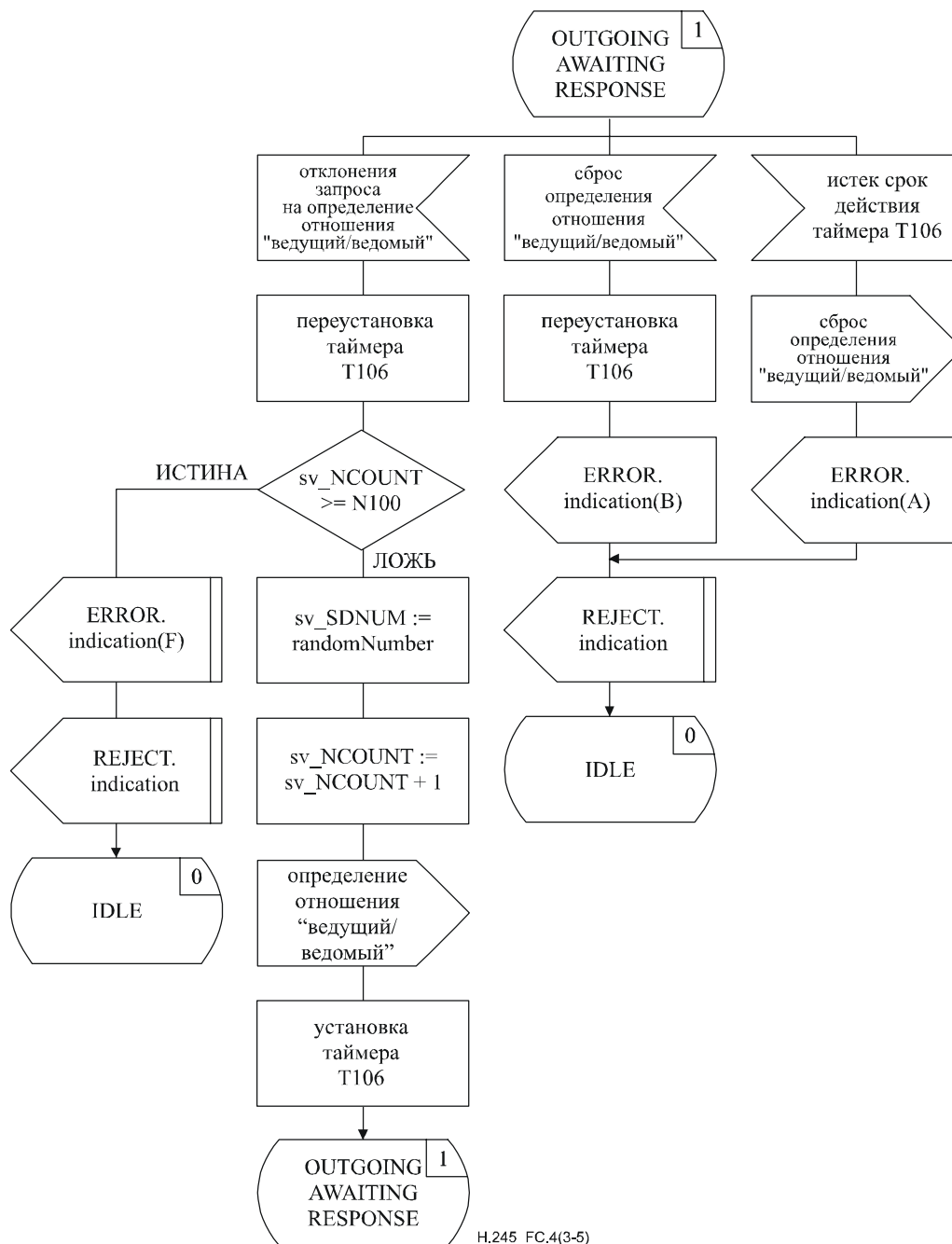


Рисунок С.4/Н.245 – Диаграмма SDL для MSDSE (лист 3 из 5)

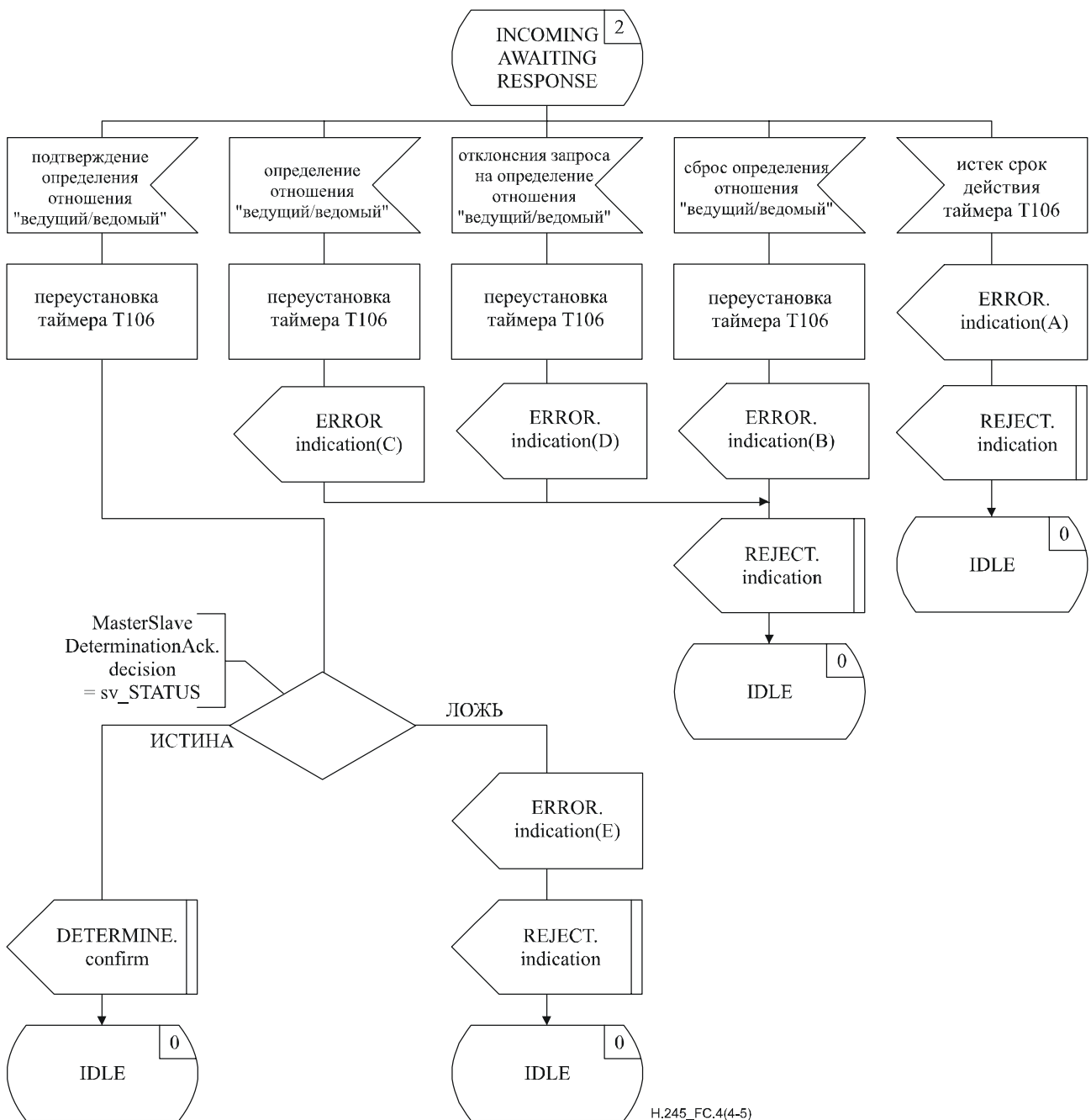
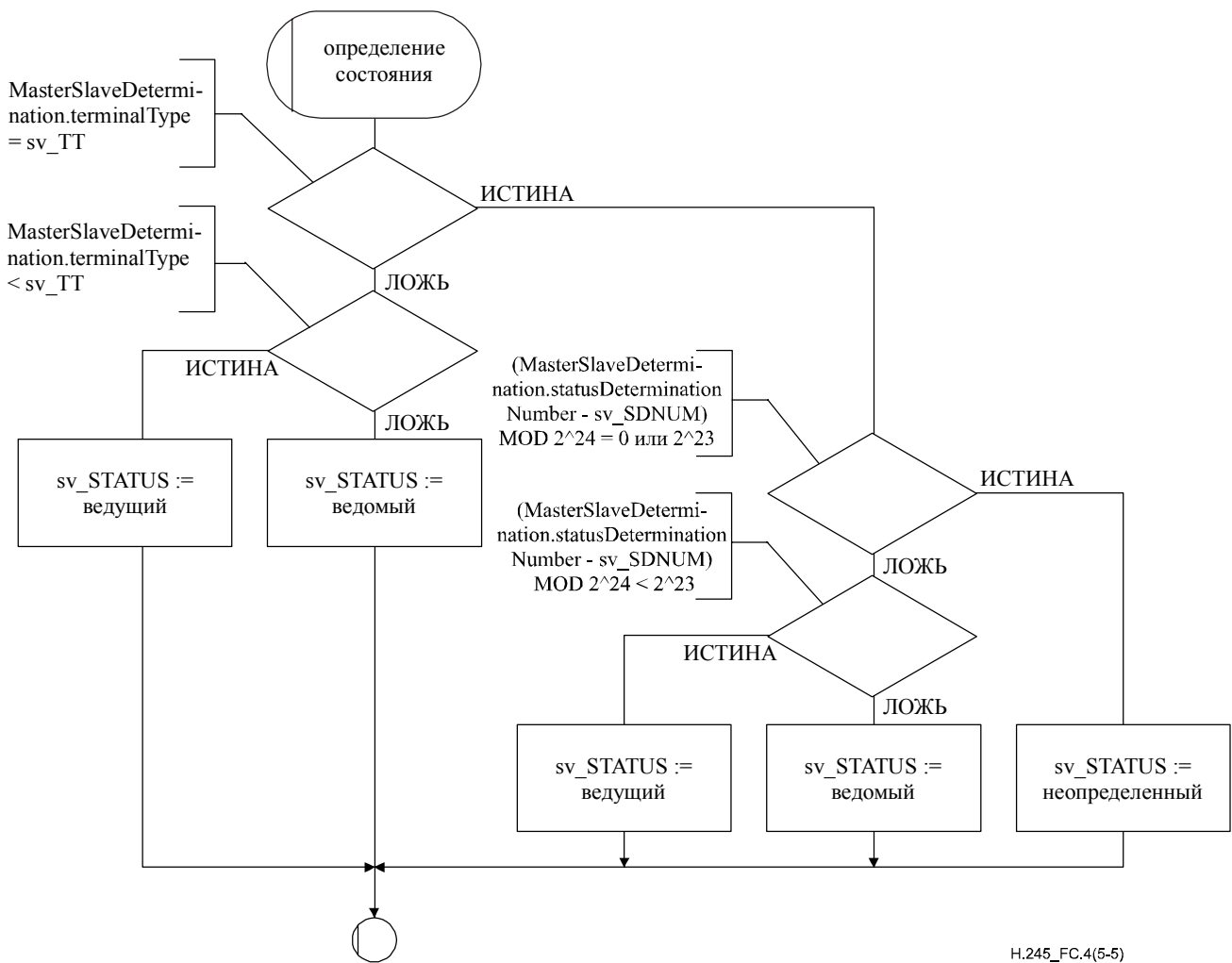


Рисунок С.4/Н.245 – Диаграмма SDL для MSDSE (лист 4 из 5)



H.245_FC.4(5-5)

Рисунок С.4/Н.245 – Диаграмма SDL для MSDSE (лист 5 из 5)

С.3 Процедуры обмена возможностями

С.3.1 Введение

Эти процедуры используются терминалами для передачи своих возможностей, они называются Объект сигнализации обмена возможностями (CESE). Процедуры определены с помощью примитивов и состояний в интерфейсе между CESE и пользователем CESE. Информация о протоколе передается равноправному CESE с помощью соответствующих сообщений, которые определены в Приложении А. Существуют передающий CESE и принимающий CESE. На каждом из передающих и принимающих концов существует один экземпляр CESE для каждого вызова.

Все терминалы, предназначенные для использования в приложениях связи "точка-точка" или те терминалы, которые подключены к MCU, должны иметь возможность идентифицировать TerminalCapabilitySet и его структуру, и такие значения его возможностей, которые обязательны для этих приложений; любые нераспознанные значения возможности должны игнорироваться, и предполагается, что при этом не формируется никакой ошибки.

Обмен возможностями может производиться в любой момент времени. При обмене возможностями могут передаваться как измененные, так и неизменные возможности. Не следует передавать неизменившиеся возможности без веской причины.

В приведенном ниже тексте содержится общее описание действия этого протокола. В случае каких-либо расхождений между ним и официальной спецификацией протокола, которая приведена ниже, преимущественную силу имеет официальная спецификация.

С.3.1.1 Общее описание протокола – передающий CESE

Обмен возможности инициируется, когда пользователь в передающем CESE формирует примитив TRANSFER.request. Сообщение TerminalCapabilitySet передается равноправному принимающему CESE, и запускается таймер T101. Если в ответ на сообщение TerminalCapabilitySet получено сообщение TerminalCapabilitySetAck, то таймер T101 останавливается, и пользователь с помощью примитива TRANSFER.confirm информируется о том, что обмен возможностями прошел успешно. Однако если в ответ на сообщение TerminalCapabilitySet было получено сообщение TerminalCapabilitySetReject, то таймер T101 останавливается и пользователь с помощью примитива REJECT.indication информируется о том, что равноправный пользователь CESE отказал в обмене возможностями.

Если срок действия таймера T101 истек, то передающий пользователь CESE информируется об этом с помощью примитива REJECT.indication, и передается сообщение TerminalCapabilitySetRelease.

С.3.1.2 Общее описание протокола – принимающий CESE

При получении сообщения TerminalCapabilitySet принимающим CESE пользователь с помощью примитива TRANSFER.indication информируется о запросе обмена возможностями. Принимающий пользователь CESE сообщает о приеме запроса обмена возможностями, формируя примитив TRANSFER.response, а равноправному передающему CESE передается сообщение TerminalCapabilitySetAck. Принимающий пользователь CESE сообщает об отклонении запроса обмена возможностями, формируя примитив REJECT.request, а равноправному передающему CESE передается сообщение TerminalCapabilitySetReject.

С.3.2 Связь между CESE и пользователем CESE

С.3.2.1 Обмен примитивами между CESE и пользователем CESE

Связь между CESE и пользователем CESE осуществляется с помощью примитивов, приведенных в таблице С.6.

Таблица С.6/Н.245 – Примитивы и параметры

Универсальное название	Тип			
	запрос	индикация	ответ	подтверждение
TRANSFER	PROTOID MUXCAP CAPTABLE CAPDESCRIPTORS	PROTOID MUXCAP CAPTABLE CAPDESCRIPTORS	– (Примечание 1)	–
REJECT	CAUSE	SOURCE CAUSE	не определен (Примечание 2)	не определен

ПРИМЕЧАНИЕ 1. – "-" означает отсутствие параметров.
ПРИМЕЧАНИЕ 2. – "не определен" означает, что этот примитив не определен.

С.3.2.2 Определение примитивов

Определения этих примитивов приведены ниже.

- Примитивы TRANSFER используются для передачи обмена возможностями.
- Примитивы REJECT используются для отклонения элемента дескриптора возможности и для прекращения текущей передачи возможности.

С.3.2.3 Определение параметров

Ниже даны определения параметров примитивов, приведенных в таблице С.6.

- a) Параметр PROTOID является параметром идентификатора протокола. Этот параметр отображается в поле protocolIdentifier сообщения TerminalCapabilitySet и прозрачно передается равноправному пользователю CESE. Этот параметр является обязательным.
- b) Параметр MUXCAP является параметром возможности мультиплексирования. Этот параметр отображается в поле multiplexCapability сообщения TerminalCapabilitySet и прозрачно передается равноправному пользователю CESE. Этот параметр является необязательным.
- c) Параметр CAPTABLE является параметром таблицы возможностей. Могут существовать один или несколько элементов таблицы возможностей, описываемых этим параметром. Этот параметр отображается в поле capabilityTable сообщения TerminalCapabilitySet, он прозрачно передается равноправному пользователю CESE. Этот параметр является необязательным.
- d) Параметр CAPDESCRIPTORS является параметром дескрипторов возможностей. Могут существовать один или несколько дескрипторов возможностей, описываемых этим параметром. Этот параметр отображается в поле capabilityDescriptors сообщения TerminalCapabilitySet, он прозрачно передается равноправному пользователю CESE. Этот параметр является необязательным.
- e) Параметр SOURCE указывает на источник примитива REJECT.indication. Параметр SOURCE имеет значение "USER" ("пользователь") или "PROTOCOL" ("протокол"). Последнее значение может быть следствием истечения таймера.
- f) Параметр CAUSE указывает на причину отклонения параметра CAPTABLE или CAPDESCRIPTORS. Параметр CAUSE не присутствует, если параметр SOURCE показывает "PROTOCOL".

С.3.2.4 Состояния CESE

Следующие состояния используются для определения разрешенной последовательности обмена примитивами между CESE и пользователем CESE.

Передающий CESE имеет следующие состояния:

Состояние 0: IDLE

CESE находится в состоянии ожидания.

Состояние 1: AWAITING RESPONSE

CESE ожидает ответа от удаленного CESE.

Принимающий CESE имеет следующие состояния:

Состояние 0: IDLE

CESE находится в состоянии ожидания.

Состояние 1: AWAITING RESPONSE

CESE ожидает ответа от пользователя CESE.

С.3.2.5 Диаграмма переходов между состояниями

В этом подразделе определяется разрешенная последовательность обмена примитивами между CESE и пользователем CESE. Разрешенная последовательность обмена примитивами связана с состояниями CESE так, как они видны с точки зрения пользователя CESE. Разрешенные последовательности обмена примитивами определены отдельно для передающего CESE и принимающего CESE, как это показано на рисунках С.5 и С.6, соответственно.

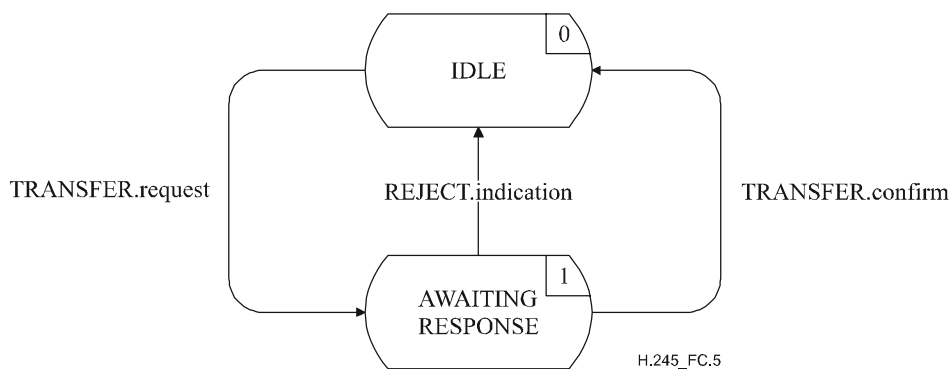


Рисунок С.5/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в передающем CESE

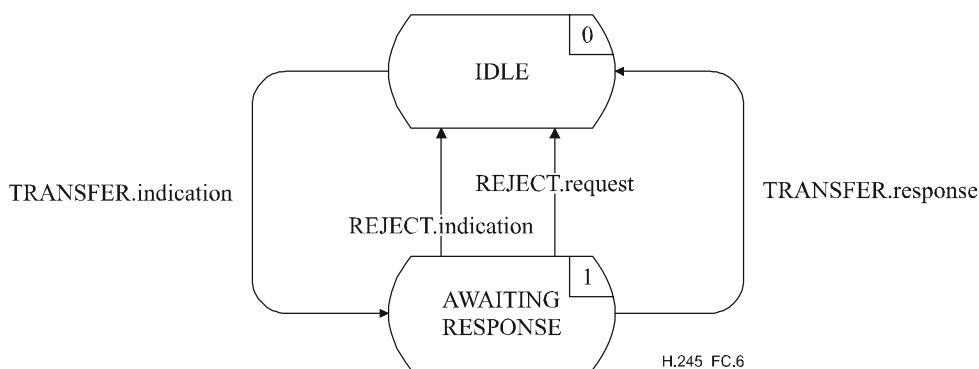


Рисунок С.6/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в принимающем CESE

С.3.3 Связь между равноправными CESE

С.3.3.1 Сообщения

В таблице С.7 показаны определенные в Приложении А сообщения и поля CESE, которые относятся к протоколу CESE.

Таблица С.7/Н.245 – Названия и поля сообщений CESE

Функция	Сообщение	Направление	Поле
transfer	TerminalCapabilitySet	O → I (Примечание)	sequenceNumber protocolIdentifier multiplexCapability capabilityTable capabilityDescriptors
	TerminalCapabilitySetAck	O ← I	sequenceNumber
reject	TerminalCapabilitySetReject	O ← I	sequenceNumber cause
reset	TerminalCapabilitySetRelease	O → I	–
ПРИМЕЧАНИЕ. – Направление: O – передающее, I – принимающее.			

С.3.3.2 Переменные состояния CESE

Для передающего CESE определена следующая переменная состояния:

out_SQ

Эта переменная состояния используется для указания самого последнего сообщения TerminalCapabilitySet. Она увеличивается на единицу и отображается в поле sequenceNumber сообщения TerminalCapabilitySet перед передачей сообщения TerminalCapabilitySet. При обработке out_SQ используется арифметика по модулю 256.

Для принимающего CESE определена следующая переменная состояния:

in_SQ

Эта переменная состояния используется для хранения значение поля sequenceNumber самого последнего полученного сообщения TerminalCapabilitySet. В сообщениях TerminalCapabilitySetAck и TerminalCapabilitySetReject поля sequenceNumber установлены равными значению in_SQ перед передачей равноправному CESE.

С.3.3.3 Таймеры CESE

Для передающего CESE определен следующий таймер:

T101

Этот таймер используется во время состояния AWAITING RESPONSE. Он определяет максимальное время, в течение которого могут не быть получены сообщения TerminalCapabilitySetAck или TerminalCapabilitySetReject.

С.3.4 Процедуры CESE

На рисунке С.7 показаны все примитивы CESE и их параметры, а также сообщения для передающего и принимающего CESE.

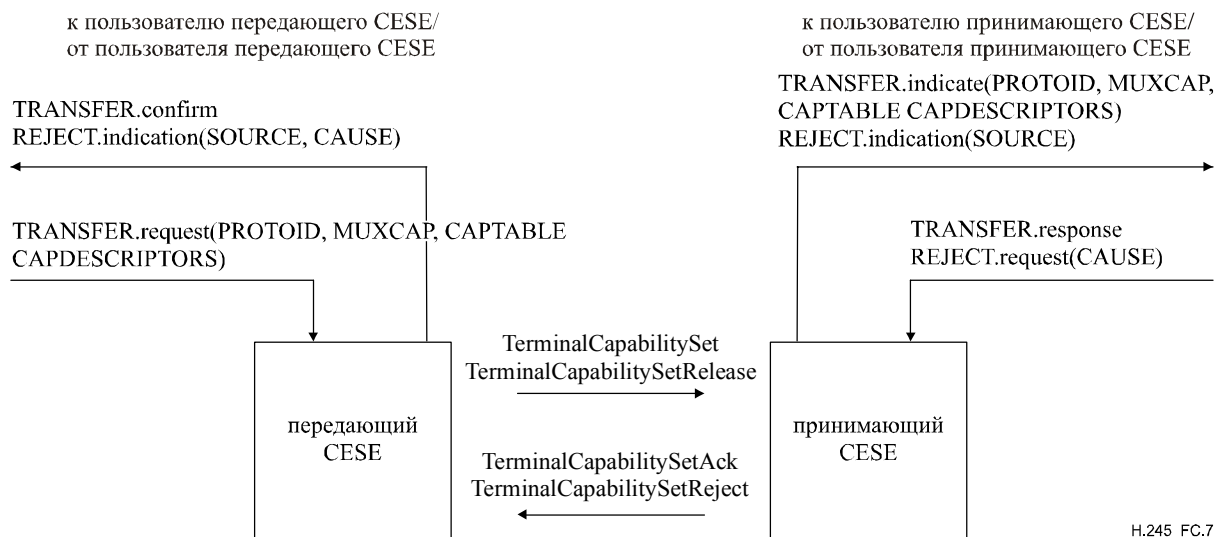


Рисунок С.7/Н.245 – Примитивы и сообщения в объекте сигнализации обмена возможностями

С.3.4.1 Принимаемые по умолчанию значения параметров примитивов

Если в диаграммах SDL в явном виде не установлены другие значения, то параметры индикации и примитивы квитирования принимают значения, приведенные в таблице С.8.

Таблица С.8/Н.245 – Принимаемые по умолчанию значения параметров примитивов

Примитив	Параметр	Значение по умолчанию
TRANSFER.indication	PROTOID	TerminalCapabilitySet.protocolIdentifier
	MUXCAP	TerminalCapabilitySet.multiplexCapability
	CAPTABLE CAPDESCRIPTORS	TerminalCapabilitySet.capabilityTable TerminalCapabilitySet.capabilityDescriptors
REJECT.indication	SOURCE CAUSE	USER null

С.3.4.2 Принимаемые по умолчанию значения полей сообщений

Если в диаграммах SDL в явном виде не установлены другие значения, то поля сообщения принимают значения, приведенные в таблице С.9.

Таблица С.9/Н.245 – Принимаемые по умолчанию значения полей сообщений

Сообщение	Поле	Значение по умолчанию (Примечание)
TerminalCapabilitySet	sequenceNumber	out_SQ
	protocolIdentifier	TRANSFER.request(PROTOID)
	multiplexCapability	TRANSFER.request(MUXCAP)
	capabilityTable	TRANSFER.request(CAPTABLE)
	capabilityDescriptors	TRANSFER.request(CAPDESCRIPTORS)
TerminalCapabilitySetAck	sequenceNumber	in_SQ
TerminalCapabilitySetReject	sequenceNumber	in_SQ
	cause	REJECT.request(CAUSE)
TerminalCapabilitySetRelease	–	–
ПРИМЕЧАНИЕ. – Поле сообщения не должно кодироваться, если соответствующий параметр примитива равен null (нулю), т.е. он не присутствует.		

С.3.4.3 Диаграммы языка описания структур (SDL)

Процедуры взаимодействия передающего CESE и принимающего CESE выражены в форме SDL на рисунках С.8 и С.9, соответственно.

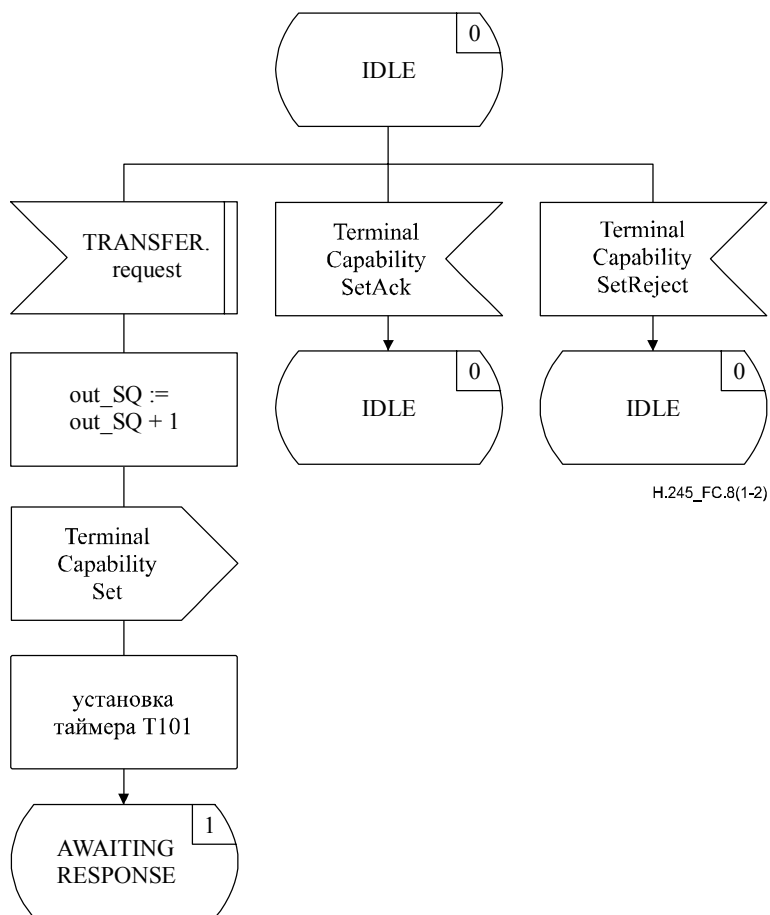
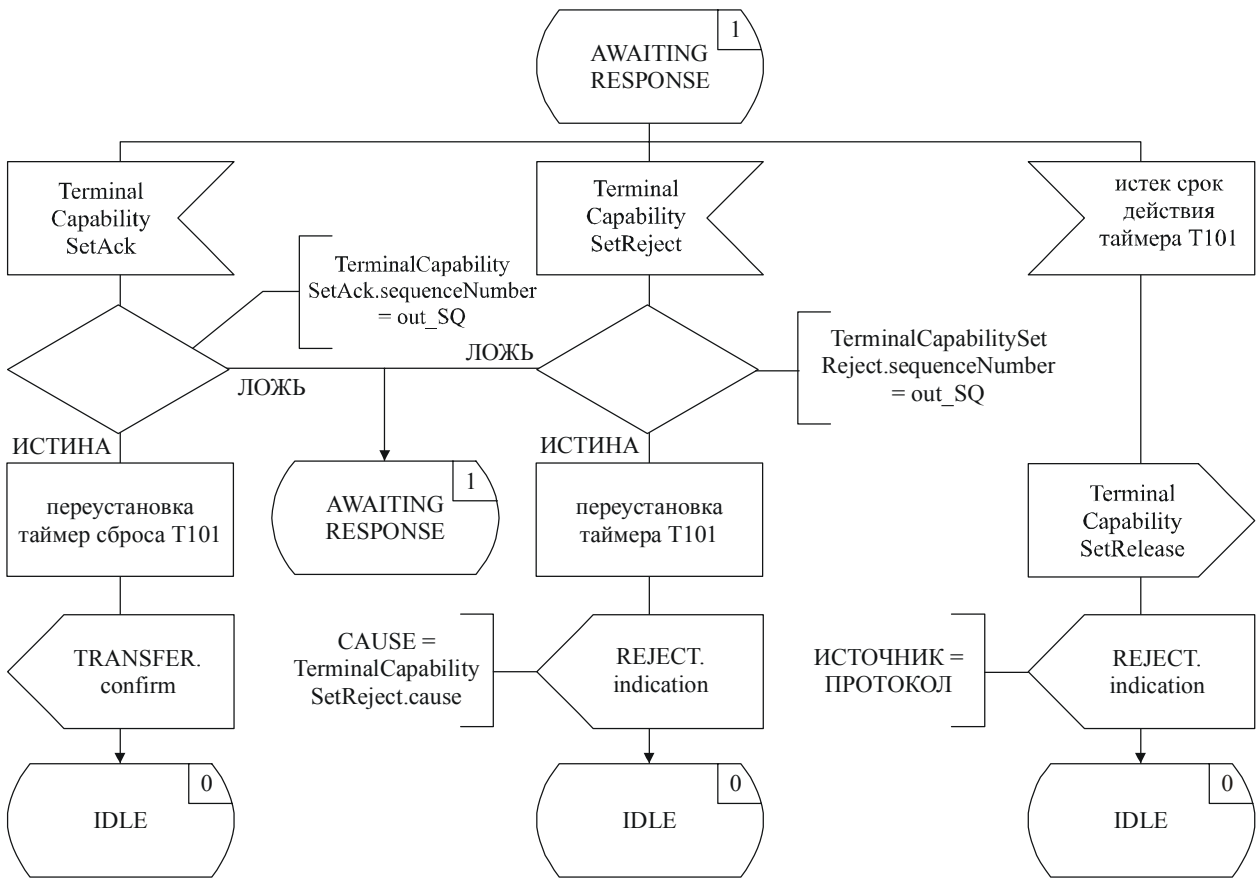
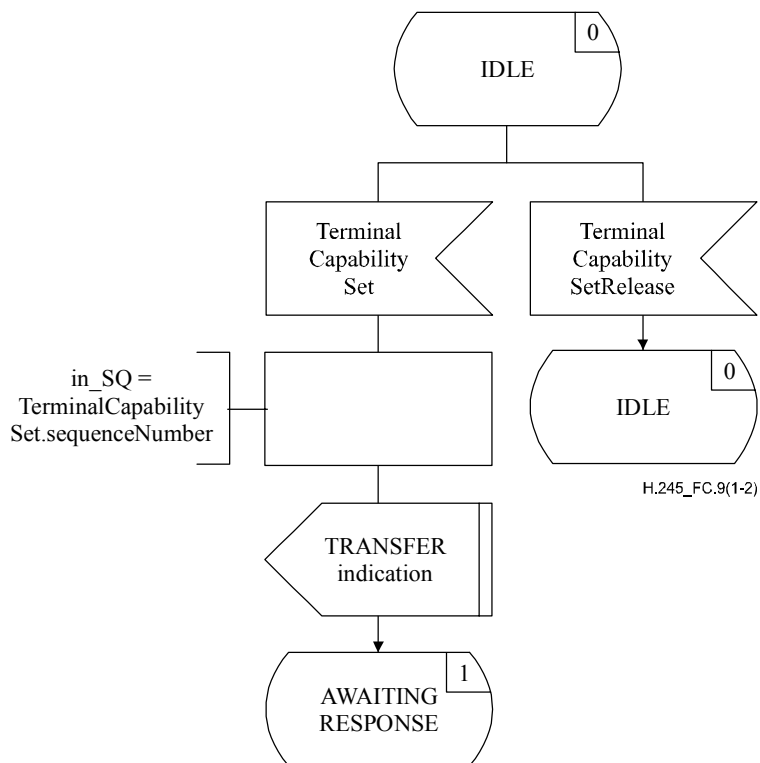


Рисунок С.8/Н.245 – Диаграмма SDL для передающего CESE (лист 1 из 2)



H.245_FC.8(2-2)

Рисунок С.8/Н.245 – Диаграмма SDL для передающего CESE (лист 2 из 2)



H.245_FC.9(1-2)

Рисунок С.9/Н.245 – Диаграмма SDL для принимающего CESE (лист 1 из 2)

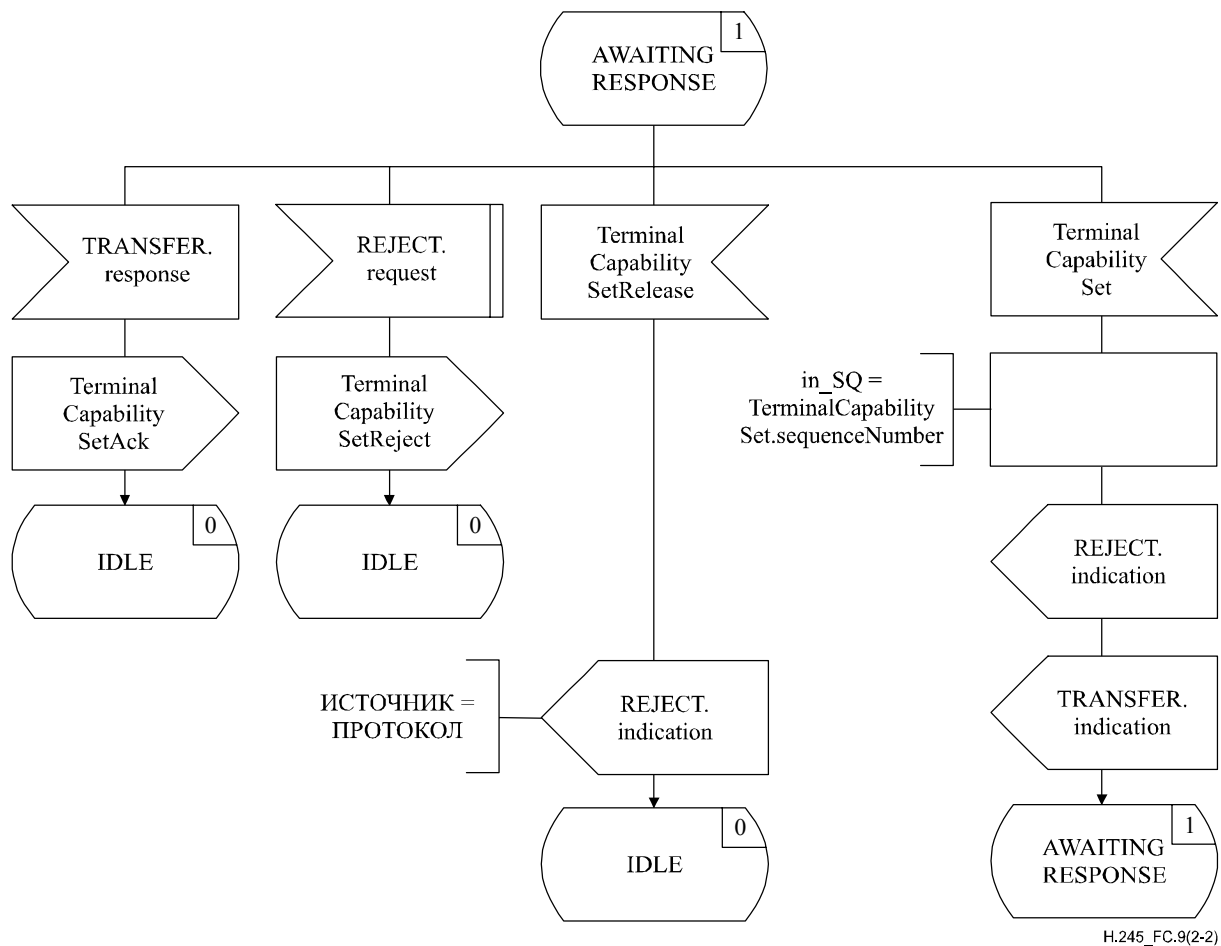


Рисунок С.9/Н.245 – Диаграмма SDL для принимающего CESE (лист 2 из 2)

С.4 Процедуры сигнализации однонаправленного логического канала

С.4.1 Введение

Определенный в этом разделе протокол обеспечивает надежное открытие и закрытие однонаправленных логических каналов, в протоколе используются процедуры с подтверждением приема.

Определенный в этом разделе протокол называется Объектом сигнализации логического канала (LCSE). Процедуры определены на основе примитивов в интерфейсе между LCSE и пользователем LCSE, а также состояний LCSE. Информация протокола передается равноправному LCSE через соответствующие сообщения, которые определены в Приложении А.

Существуют передающий LCSE и принимающий LCSE. На каждой из передающей и принимающей сторон существует одна копия LCSE для каждого однонаправленного логического канала. Не существует способа связи между принимающим LCSE и передающим LCSE на одной стороне иного, чем через примитивы к пользователю LCSE и от пользователя LCSE. Формируются сообщения о состояниях ошибки LCSE.

Данные должны передаваться только в том случае, когда логический канал находится в состоянии ESTABLISHED. Если данные получены по логическому каналу, который не находится в состоянии ESTABLISHED, то эти данные должны быть отброшены и не должно считаться, что произошла ошибка.

Переключение режимов должно производиться с помощью закрытия и открытия существующих логических каналов или путем открытия новых логических каналов.

ПРИМЕЧАНИЕ. – Некоторые Рекомендации, которые используют настоящую Рекомендацию, могут определять некоторые заданные по умолчанию логические каналы. Эти каналы должны рассматриваться как ESTABLISHED с самого начала связи и не должны открываться с использованием данных процедур. Однако

они могут быть закрыты данными процедурами и впоследствии они могут быть повторно открыты для той же или другой цели.

Терминал, который больше не в состоянии обрабатывать сигналы в логическом канале, должен предпринять соответствующие действия: они должны включать закрытие логического канала и передачу соответствующей (измененной) информации о возможности на удаленный терминал.

В приведенном ниже тексте содержится общее описание действия протокола LCSE. В случае каких-либо расхождений между ним и официальной спецификацией протокола преимущественную силу имеет официальная спецификация.

С.4.1.1 Общее описание протокола

Открытие логического канала инициируется передачей пользователем на передающем LCSE примитива ESTABLISH.request. Сообщение OpenLogicalChannel, содержащее параметры прямого логического канала, но не включающее параметры обратного логического канала, передается равноправному принимающему LCSE, а также запускается таймер T103. Если в ответ на сообщение OpenLogicalChannel получено сообщение OpenLogicalChannelAck, то таймер T103 останавливается и пользователь с помощью примитива ESTABLISH.confirm информируется о том, что логический канал был успешно открыт. С этого момента данный логический канал может использоваться для передачи информации пользователя. Однако если в ответ на сообщение OpenLogicalChannel получено сообщение OpenLogicalChannelReject, то таймер T103 останавливается и пользователь с помощью примитива RELEASE.indication информируется о том, что равноправный пользователь LCSE отказал в организации логического канала.

Если за этот период окончился срок действия таймера T103, то пользователь информируется об этом с помощью примитива RELEASE.indication, и в равноправный передающий LCSE посылается сообщение CloseLogicalChannel.

Успешно организованный логический канал может быть закрыт с помощью примитива RELEASE.request, который посылается пользователем в передающий LCSE. Равноправному передающему LCSE передается сообщение CloseLogicalChannel, и запускается таймер T103. Когда получено сообщение CloseLogicalChannelAck, таймер T103 останавливается и пользователь с помощью примитива RELEASE.confirm извещается о том, что логический канал был успешно закрыт.

Если за этот период окончился срок действия таймера T103, то пользователь информируется об этом с помощью примитива RELEASE.indication.

Перед тем, как в ответ на ранее отправленное сообщение OpenLogicalChannel будет получено любое из сообщений OpenLogicalChannelAck или OpenLogicalChannelReject, пользователь в передающем LCSE может закрыть логический канал, используя примитив RELEASE.request.

Перед тем, как в ответ на ранее отправленное сообщение CloseLogicalChannel будет получено сообщение CloseLogicalChannelAck, пользователь в передающем LCSE может организовать новый логический канал, передавая примитив ESTABLISH.request.

С.4.1.2 Общее описание протокола – принимающий LCSE

Когда в принимающем LCSE получено сообщение OpenLogicalChannel, пользователь информируется о получении запроса на открытие нового логического канала с помощью примитива ESTABLISH.indication. Пользователь принимающего LCSE сообщает о приеме запроса на организацию логического канала путем передачи примитива ESTABLISH.response, а равноправному передающему LCSE передается сообщение OpenLogicalChannelAck. С этого момента логический канал может использоваться для приема информации пользователя. Пользователь принимающего LCSE сообщает об отклонении запроса на организацию логического канала с помощью примитива RELEASE.request, а равноправному передающему LCSE передается сообщение OpenLogicalChannelReject.

Успешно организованный логический канал может быть закрыт, если принимающий LCSE получит сообщение CloseLogicalChannel. Пользователь принимающего LCSE информируется с помощью примитива RELEASE.indication, и равноправному передающему LCSE передается сообщение CloseLogicalChannelAck.

С.4.1.3 Разрешение конфликтов

При одновременной инициализации запросов на открытие логических каналов могут возникнуть конфликты. Факт возникновения конфликта можно установить из знания возможностей, которыми обменялись терминалы.

Терминалы должны быть способны определить, когда конфликта возник или может возникнуть, и они должны действовать следующим образом.

Перед тем, как логические каналы могут быть открыты, один из терминалов должен быть определен как ведущий терминал, а другой – как ведомый терминал. Определенный в С.2 протокол обеспечивает одно средство принятия такого решения. Ведущий терминал должен немедленно отклонять любой запрос от ведомого терминала, который идентифицируется как конфликтный запрос. Ведомый терминал может идентифицировать такие конфликты, но он должен отвечать на запрос ведущего терминала, зная, что его предыдущий запрос будет отвергнут.

ПРИМЕЧАНИЕ. – Такие конфликты могут возникать из-за ограниченных ресурсов терминала, например, если возможности передачи и приема зависят друг от друга (как в случае терминала, который может поддерживать несколько аудиоалгоритмов, но может декодировать только тот же алгоритм, который он использует для кодирования).

Если ведомая конечная точка имеет симметричные ограничения по возможностям, то рекомендуется использовать следующий порядок действий для сведения к минимуму вероятности открытия оконечными точками конфликтующих логических каналов. Когда ведущий и ведомый терминалы показали выбранные возможности приема для определенного типа среды, ведомый терминал должен попытаться открыть логический канал для наиболее предпочитаемой ведущим терминалом возможности, которая соответствует имеющейся у него самой возможности (при этом предпочтительность возможности определяется порядком, в котором ведущий терминал расположил свои возможности); а ведущий терминал должен попытаться открыть логический канал для своей наиболее предпочитаемой возможности, которая соответствует имеющейся у ведомого терминала возможности (при этом предпочтительность возможности определяется порядком, в котором он расположил свои возможности).

Например, если ведущий терминал объявил возможности G.723.1, G.729 и G.711, а ведомый терминал указал возможности G.711 и G.729 (при этом наиболее предпочитаемые возможности расположены первыми в обоих случаях), то и ведущий, и ведомый терминалы должны пытаться открыть логические каналы для G.729.

После того как запрос на открытие логического канала был отвергнут ведущим терминалом по причине, значение которой равно masterSlaveConflict, ведомый терминал отвечает за открытие бесконфликтного канала.

Если ведомые устройства определяют наличие конфликта, а ведущий терминал не отвергает конфликтующий открытый логический канал, ведомый терминал должен закрыть конфликтующий канал. В случае конфликтующих логических каналов при симметричных ограничениях возможности ведомый терминал должен открыть соответствующий логический канал, используя процедуру замены, и должным образом закрыть конфликтующий логический канал.

С.4.2 Связь между LCSE и пользователем LCSE

С.4.2.1 Обмен примитивами между LCSE и пользователем LCSE

Связь между LCSE и пользователем LCSE осуществляется с помощью примитивов, приведенных в таблице С.10.

Таблица С.10/Н.245 – Примитивы и параметры

Универсальное название	Тип			
	запрос	индикация	ответ	подтверждение
ESTABLISH	FORWARD_PARAM	FORWARD_PARAM	– (Примечание 1)	–
RELEASE	CAUSE	SOURCE CAUSE	не определен (Примечание 2)	–
ERROR	не определен	ERRCODE	не определен	не определен
ПРИМЕЧАНИЕ 1. – "-" означает отсутствие параметров.				
ПРИМЕЧАНИЕ 2. – "не определено" означает, что этот примитив не определен.				

С.4.2.2 Определение примитивов

Ниже приведены определения указанных примитивов:

- a) Примитивы типа ESTABLISH используются для организации логического канала для аудиовизуальной связи и передачи данных.
- b) Примитивы типа RELEASE используются для сброса логического канала.
- c) Примитивы типа ERROR передают ошибки LCSE в объект управления.

С.4.2.3 Определение параметров

Ниже даны определения параметров примитивов, приведенных в таблице С.10:

- a) Параметр FORWARD_PARAM определяет параметры, связанные с логическим каналом. Этот параметр преобразуется в поле forwardLogicalChannelParameters сообщения OpenLogicalChannel и прозрачно передается равноправному пользователю LCSE.
- b) Параметр SOURCE показывает пользователю LCSE источник сброса логического канала. Параметр SOURCE принимает значения "USER" или "LCSE", указывая на пользователя LCSE или на LCSE. Последнее значение может являться результатом ошибки протокола.
- c) Параметр CAUSE указывает причину, по которой равноправный пользователь LCSE отклонил запрос на организацию логического канала. Параметр CAUSE не присутствует, когда параметр SOURCE показывает "LCSE".
- d) Параметр ERRCODE показывает тип ошибки LCSE. В таблице С.14 приведены разрешенные значения параметра ERRCODE.

С.4.2.4 Состояния LCSE

Следующие состояния используются для определения разрешенной последовательности обмена примитивами между LCSE и пользователем LCSE и обмена сообщениями между равноправными LCSE. Состояния определены отдельно для каждого передающего LCSE и принимающего LCSE. Передающий LCSE имеет следующие состояния:

Состояние 0: RELEASED

Логический канал сброшен. Логический канал не должен использоваться для передачи исходящих данных.

Состояние 1: AWAITING ESTABLISHMENT

Передающий LCSE ожидает организации логического канала с равноправным принимающим LCSE. Логический канал не должен использоваться для передачи исходящих данных.

Состояние 2: ESTABLISHED

Установлено соединение по логическому каналу между равноправными LCSE. Логический канал может использоваться для передачи исходящих данных.

Состояние 3: AWAITING RELEASE

Передающий LCSE ожидает сброса логического канала с равноправным принимающим LCSE. Логический канал не должен использоваться для передачи исходящих данных.

Принимающий LCSE имеет следующие состояния:

Состояние 0: RELEASED

Логический канал сброшен. Логический канал не должен использоваться для приема входящих данных.

Состояние 1: AWAITING ESTABLISHMENT

Принимающий LCSE ожидает организации логического канала с равноправным передающим LCSE. Логический канал не должен использоваться для приема входящих данных.

Состояние 2: ESTABLISHED

Установлено соединение логического канала между равноправными LCSE. Логический канал может использоваться для приема входящих данных.

С.4.2.5 Диаграмма переходов между состояниями

В этом подразделе определяется разрешенная последовательность обмена примитивами между LCSE и пользователем LCSE. Разрешенная последовательность обмена примитивами связана с состояниями LCSE, которые видны пользователю LCSE. Разрешенные последовательности обмена примитивами определены отдельно для передающего LCSE и принимающего LCSE, они показаны на рисунках С.10 и С.11, соответственно.

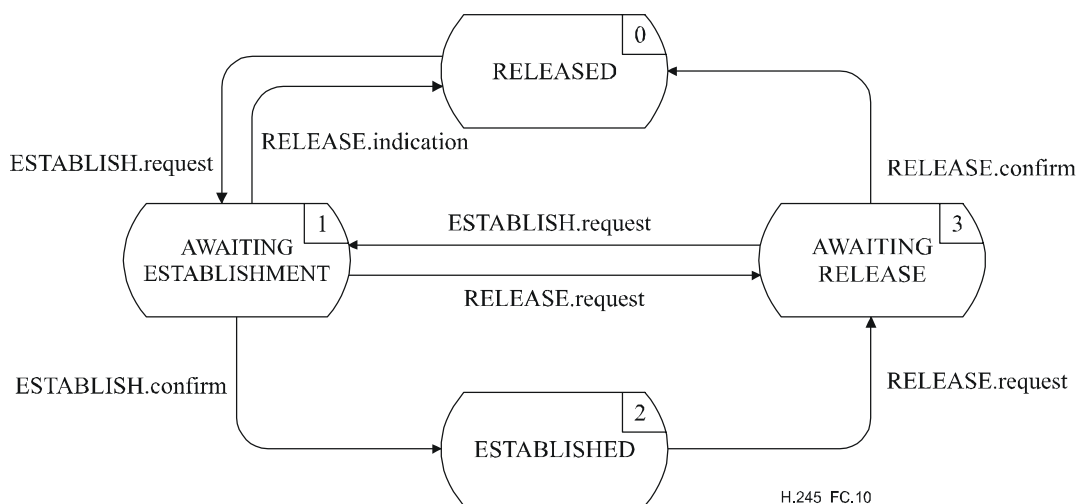


Рисунок С.10/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в передающем LCSE

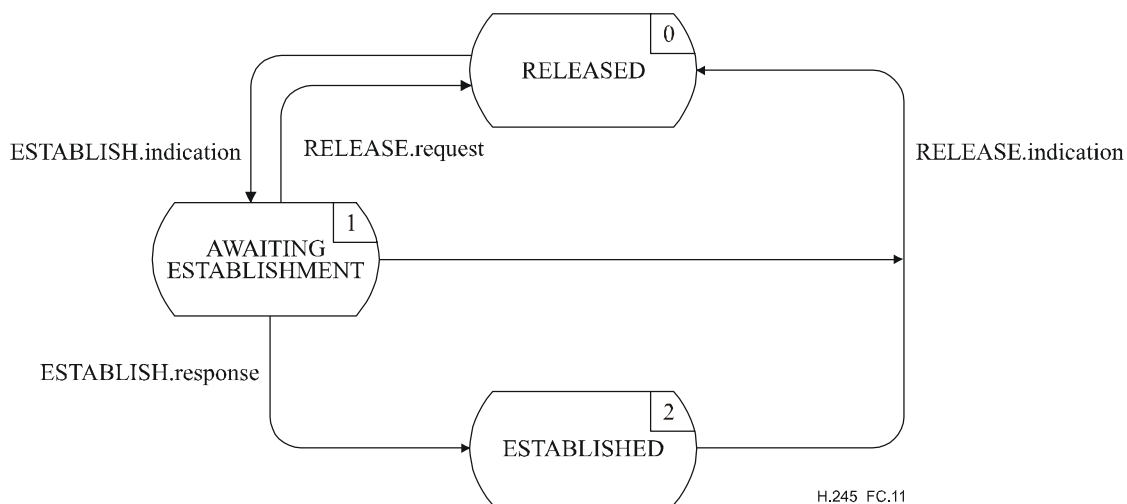


Рисунок С.11/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в принимающем LCSE

С.4.3 Связь между равноправными LCSE

С.4.3.1 Сообщения LCSE

В таблице С.11 приведены определенные в Приложении А сообщения и поля LCSE, которые используются в протоколе LCSE.

Таблица С.11/Н.245 – Названия и поля сообщений LCSE

Функция	Сообщение	Направление	Поле
establishment	OpenLogicalChannel	O → I (Примечание)	forwardLogicalChannelNumber forwardLogicalChannelParameters
	OpenLogicalChannelAck	O ← I	forwardLogicalChannelNumber
	OpenLogicalChannelReject	O ← I	forwardLogicalChannelNumber cause
release	CloseLogicalChannel	O → I	forwardLogicalChannelNumber source
	CloseLogicalChannelAck	O ← I	forwardLogicalChannelNumber
ПРИМЕЧАНИЕ. – Направление: O – передающее, I – принимающее.			

С.4.3.2 Переменные состояния LCSE

В передающем LCSE определена следующая фазовая переменная:

out_LCN

Эта переменная состояния устанавливает различие между передающими LCSE. Она инициализируется при инициализации передающего LCSE. Значение out_LCN используется для установки поля forwardLogicalChannelNumber сообщений LCSE, передаваемых передающим LCSE. Для сообщений LCSE, полученных в передающем LCSE, значение поля сообщения forwardLogicalChannelNumber идентично значению out_LCN.

В принимающем LCSE определена следующая фазовая переменная:

in_LCN

Эта переменная состояния устанавливает различие между принимающими LCSE. Она инициализируется при инициализации принимающего LCSE. Значение in_LCN используется для установки значения поля forwardLogicalChannelNumber сообщений LCSE, которые переданы из принимающего LCSE. Для сообщений LCSE, полученных принимающим LCSE, значение поля forwardLogicalChannelNumber сообщения идентично значению in_LCN.

С.4.3.3 Таймеры LCSE timers

Для передающего LCSE определен следующий таймер:

T103

Этот таймер используется при состояниях AWAITING ESTABLISHMENT и AWAITING RELEASE. Он определяет максимально возможное разрешенное время без получения сообщений OpenLogicalChannelAck, OpenLogicalChannelReject или CloseLogicalChannelAck.

С.4.4 Процедуры LCSE

С.4.4.1 Введение

На рисунке С.12 приведены примитивы и их параметры, а также сообщения для каждого передающего и принимающего LCSE.

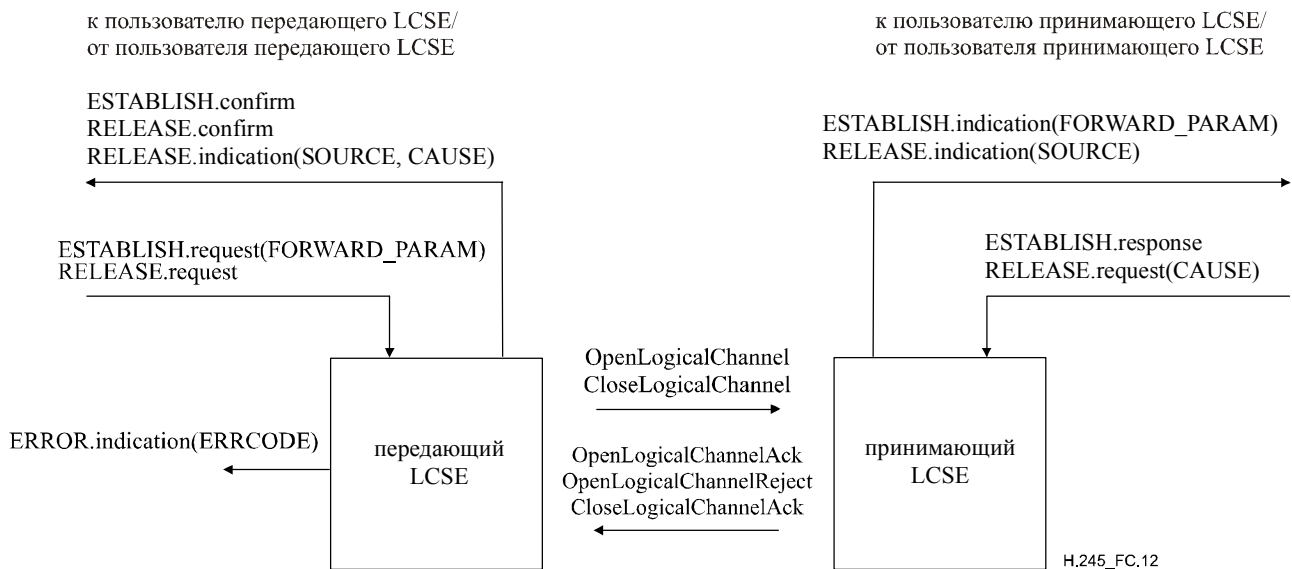


Рисунок С.12/Н.245 – Примитивы и сообщения в Объекте сигнализации логического канала

С.4.4.2 Принимаемые по умолчанию значения параметров примитивов

Параметры индикации и примитивы квитирования принимают значения, показанные в таблице С.12, если в диаграммах SDL в явном виде не установлены другие значения.

Таблица С.12/Н.245 – Принимаемые по умолчанию значения параметров примитивов

Примитив	Параметр	Значение по умолчанию (Примечание)
ESTABLISH.indication	FORWARD_PARAM	OpenLogicalChannel.forwardLogicalChannelParameters
RELEASE.indication	SOURCE CAUSE	CloseLogicalChannel.source null

ПРИМЕЧАНИЕ. – Параметр примитива должен кодироваться как null (ноль), если указанное поле сообщения не присутствует в данном сообщении.

С.4.4.3 Принимаемые по умолчанию значения полей

Поля сообщений принимают значения, показанные в таблице С.13, если в диаграммах SDL в явном виде не установлены иные значения.

Таблица С.13/Н.245 – Принимаемые по умолчанию значения полей

Сообщение	Поле	Значение по умолчанию (Примечание 1)
OpenLogicalChannel (Примечание 2)	forwardLogicalChannelNumber forwardLogicalChannelParameters	out_LCN ESTABLISH.request (FORWARD_PARAM)
OpenLogicalChannelAck	forwardLogicalChannelNumber	in_LCN
OpenLogicalChannelReject	forwardLogicalChannelNumber cause	in_LCN RELEASE.request (CAUSE)
CloseLogicalChannel	forwardLogicalChannelNumber source	out_LCN user
CloseLogicalChannelAck	forwardLogicalChannelNumber	in_LCN

ПРИМЕЧАНИЕ 1. – Поле сообщения не должно кодироваться, если соответствующий параметр равен null (нулю), т. е. он не присутствует.

ПРИМЕЧАНИЕ 2. – reverseLogicalChannelParameters не кодируется в процедурах сигнализации однонаправленного канала.

С.4.4.4 Значения параметра ERRCODE

Параметр ERRCODE примитива ERROR.indication показывает определенное условие возникновения ошибок. В таблице С.14 показаны значения, которые может принимать параметр ERRCODE в передающем LCSE. С принимающим LCSE не связано никакого примитива ERROR.indication.

Таблица С.14/Н.245 – Значения параметра ERRCODE в передающем LCSE

Тип ошибки	Код ошибки	Условие возникновения ошибки	Состояние
Несоответствующее сообщение	A	OpenLogicalChannelAck	RELEASED
	B	OpenLogicalChannelReject	RELEASED ESTABLISHED
	C	CloseLogicalChannelAck	ESTABLISHED
Нет ответа от равноправного LCSE	D	Истечение срока действия таймера T103	AWAITING ESTABLISHMENT AWAITING RELEASE

С.4.4.5 Диаграммы языка описания структур (SDL)

Процедуры передачи сообщений передающего LCSE и принимающего LCSE представлены в форме SDL на рисунках С.13 и С.14, соответственно.

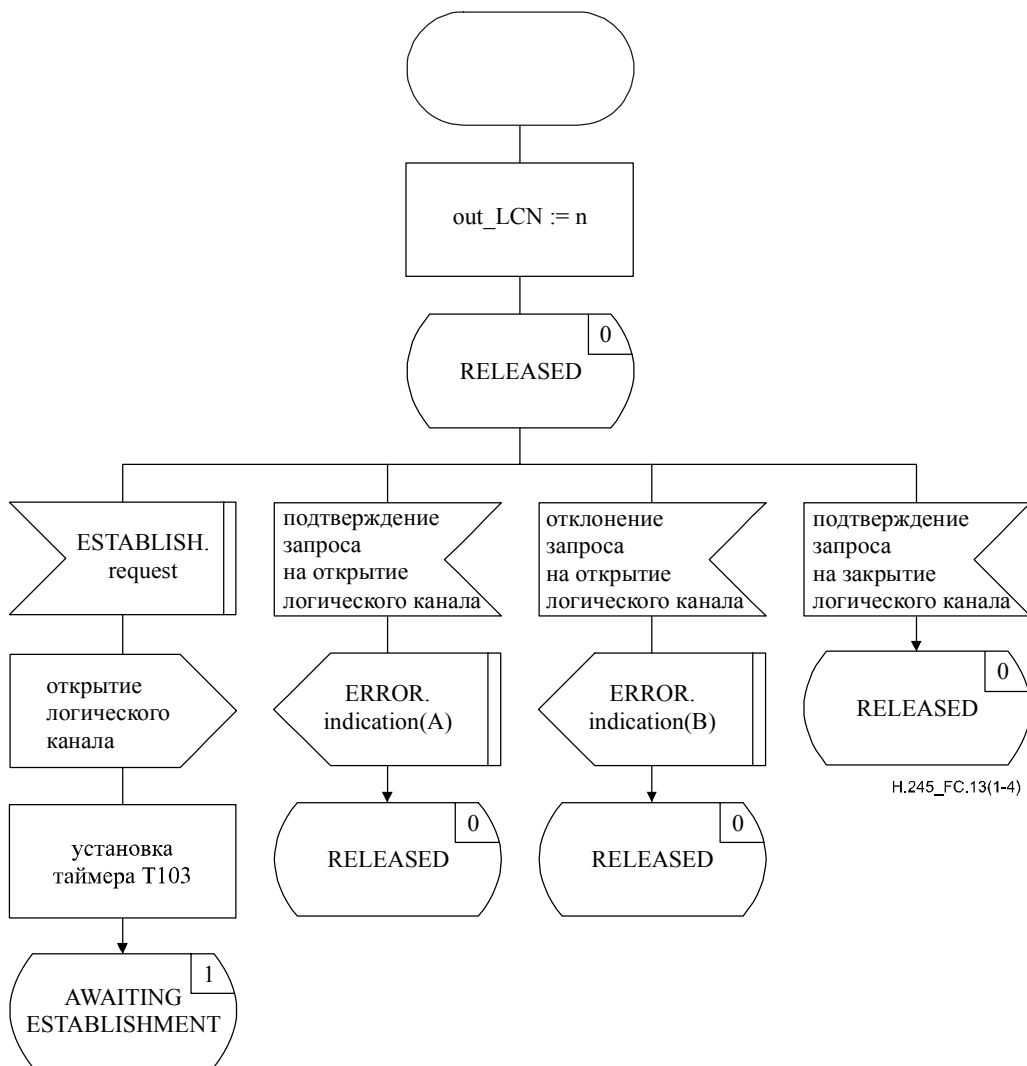
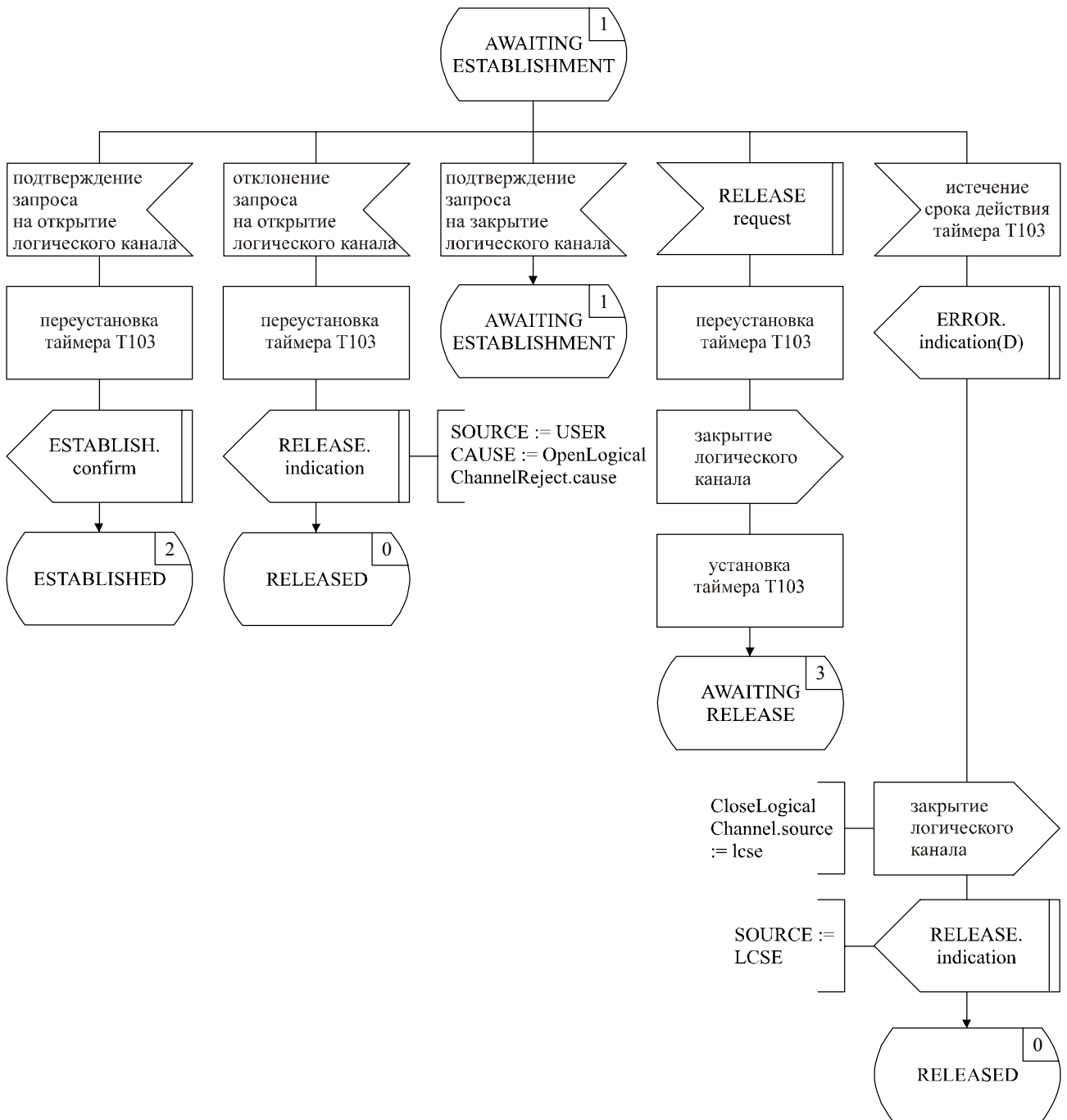


Рисунок С.13/Н.245 – Диаграмма SDL для передающего LCSE (лист 1 из 4)



H.245_C.13(2-4)

Рисунок С.13/Н.245 – Диаграмма SDL для передающего LCSE (лист 2 из 4)

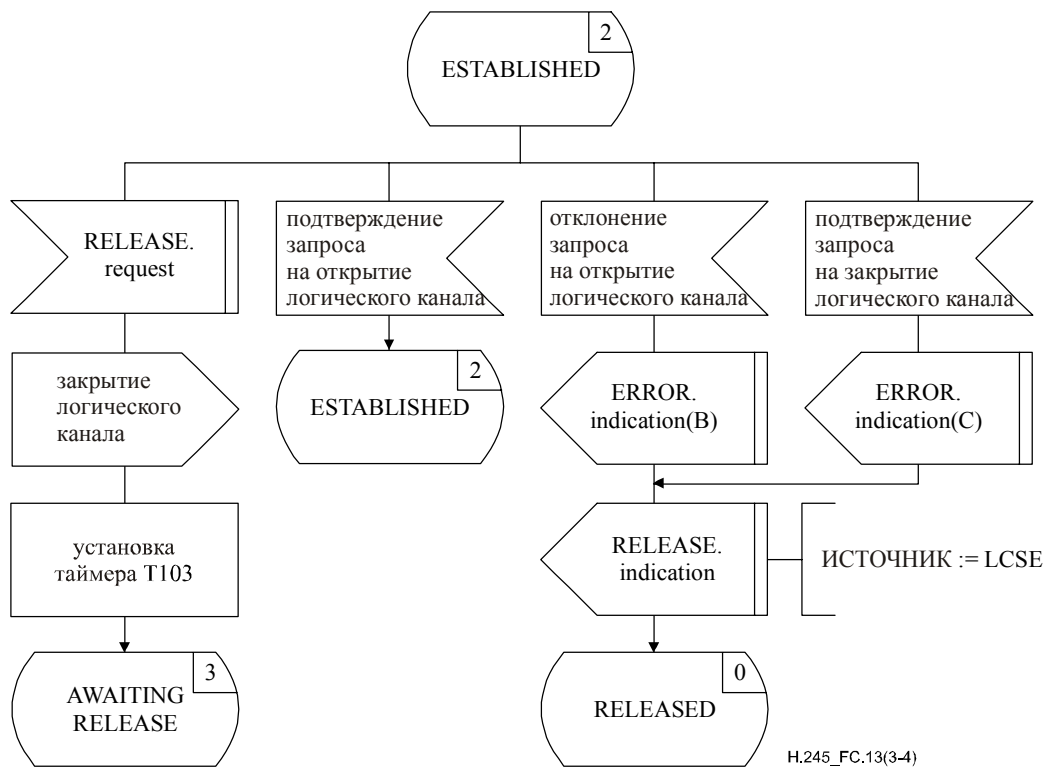


Рисунок С.13/Н.245 – Диаграмма SDL для передающего LCSE (лист 3 из 4)

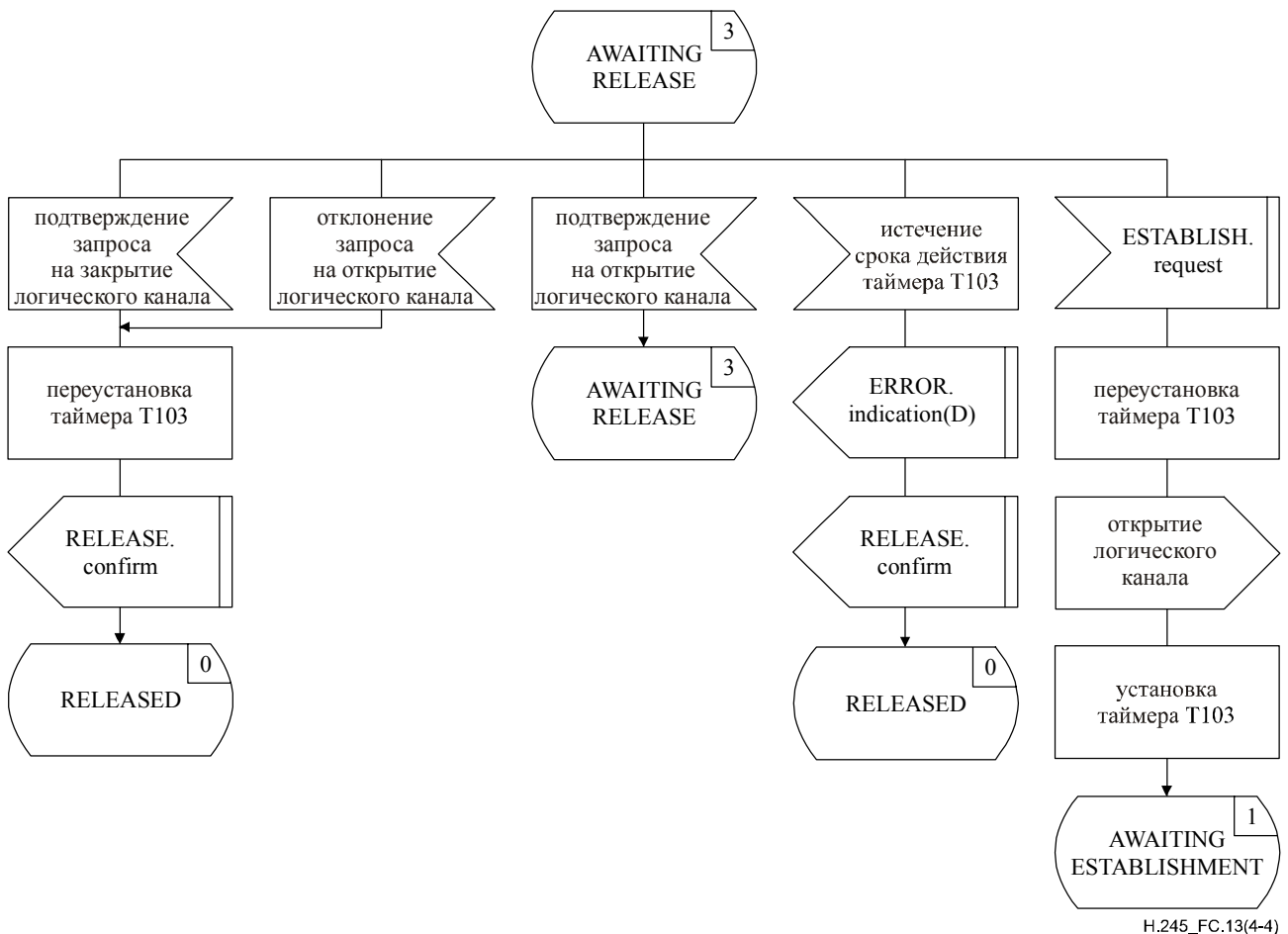


Рисунок С.13/Н.245 – Диаграмма SDL для передающего LCSE (лист 4 из 4)

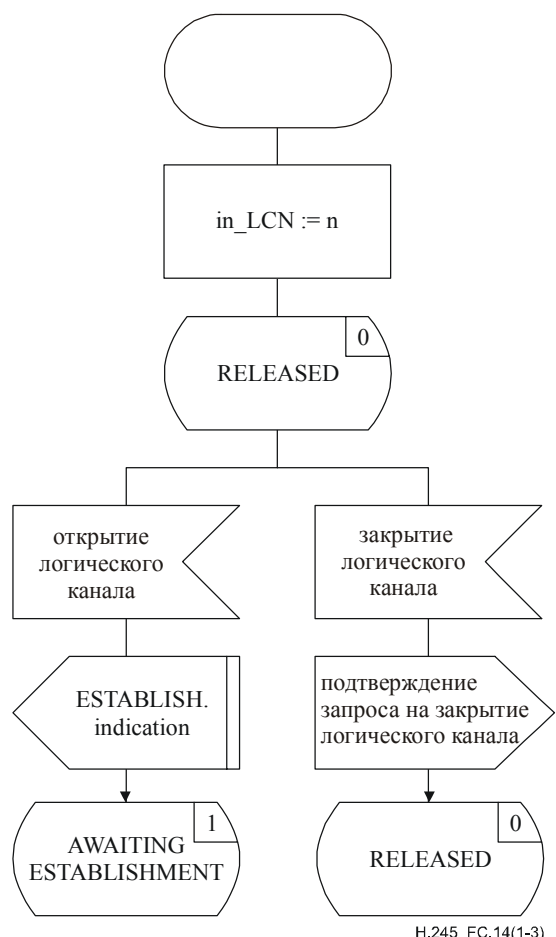


Рисунок С.14/Н.245 – Диаграмма SDL для принимающего LCSE (лист 1 из 3)

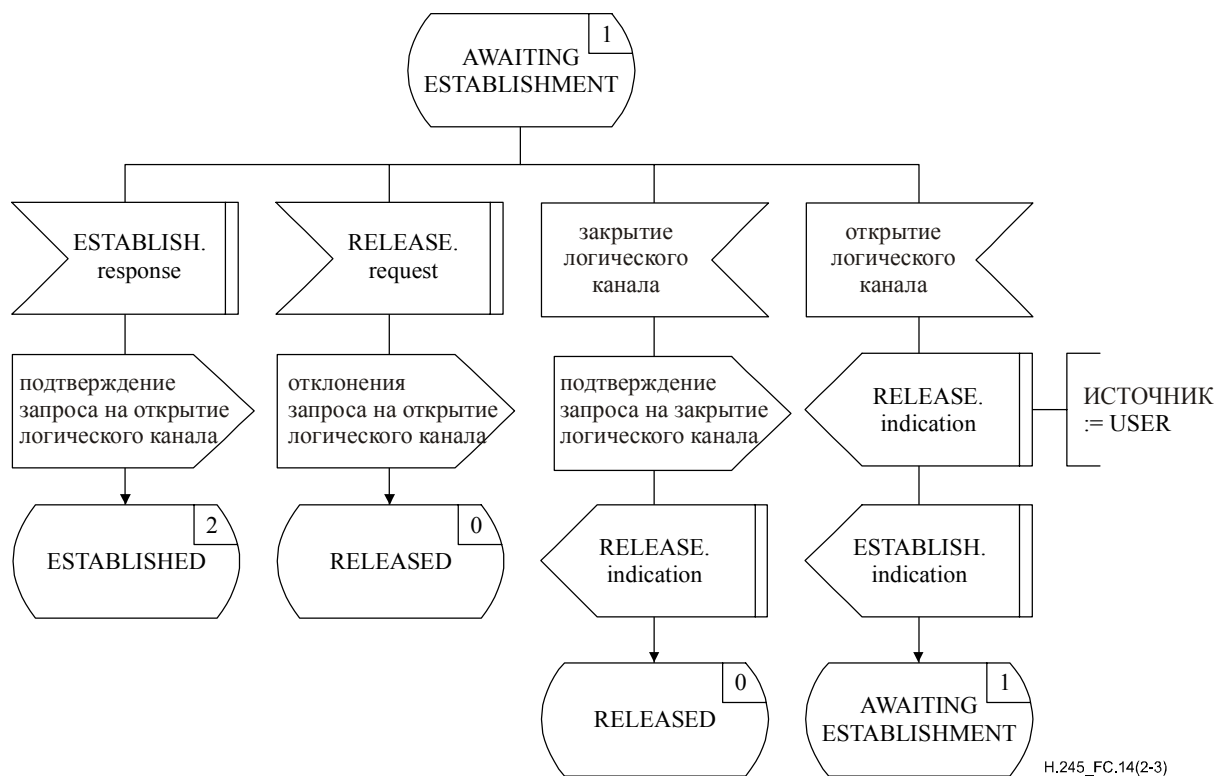


Рисунок С.14/Н.245 – Диаграмма SDL для принимающего LCSE (лист 2 из 3)

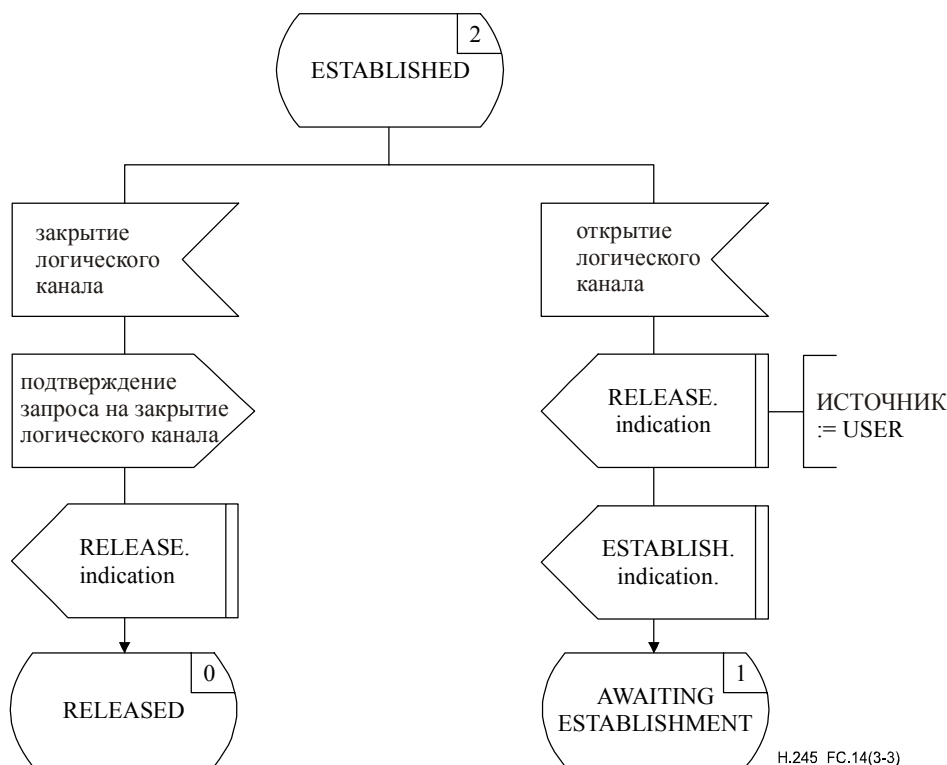


Рисунок С.14/Н.245 – Диаграмма SDL для принимающего LCSE (лист 3 из 3)

С.5 Процедуры сигнализации двунаправленного логического канала

С.5.1 Введение

Определяемый в этом разделе протокол обеспечивает надежное открытие и закрытие двунаправленных логических каналов, в протоколе используются процедуры с подтверждением приема.

Определяемый здесь протокол называется Объектом сигнализации двунаправленного логического канала (В-LCSE). Процедуры определены с помощью примитивов в интерфейсе между В-LCSE и пользователем В-LCSE, а также состояний В-LCSE. Информация протокола передается равноправному В-LCSE с помощью соответствующих сообщений, определенных в Приложении А.

Существует передающий В-LCSE и принимающий В-LCSE. На каждом передающем и принимающем конце имеется одна копия В-LCSE для каждого двунаправленного логического канала. Между принимающим В-LCSE и передающим В-LCSE на одном конце нет иного способа связи, кроме как с помощью примитивов, передаваемых пользователю В-LCSE и от него. Формируются сообщения об условиях возникновения ошибки.

Двунаправленный логический канал состоит из пары ассоциированных однонаправленных каналов. Под "прямой передачей" (передающая сторона) понимается передача от терминала, запрашивающего другой терминал о предоставлении двунаправленного логического канала, а под "обратной передачей" (принимающая сторона) имеется в виду передача в противоположном направлении.

Данные должны передаваться только в том случае, если двунаправленный логический канал находится в состоянии ESTABLISHED. Однако данные могут приниматься по прямому каналу, когда принимающий В-LCSE находится в состоянии AWAITING CONFIRMATION. Если данные получены в состоянии, отличном от состояния ESTABLISHED и состояния AWAITING CONFIRMATION, то эти данные должны быть отброшены, и это не должно восприниматься как ошибка.

Терминал может отклонить запрос на открытие двунаправленного логического канала только потому, что он не может поддерживать требуемые параметры обратного канала. В этом случае он должен отклонить запрос, указав в качестве причины `unsuitableReverseParameters` ("неподходящие параметры обратной передачи"), и немедленно инициировать процедуры организации двунаправленного логического канала, соответствующего запросу удаленного терминала; в этих процедурах параметры

обратной передачи идентичны параметрам прямой передачи неудавшегося запроса удаленного терминала, а параметры прямой передачи таковы, что данный терминал может их поддерживать, и известно, что удаленный терминал может поддержать их.

Переключение режимов должно осуществляться путем закрытия и открытия существующих логических каналов или путем открытия новых логических каналов.

ПРИМЕЧАНИЕ. – Некоторые Рекомендации, которые используют положения настоящей Рекомендации, могут определить некоторые устанавливаемые по умолчанию логические каналы. Эти каналы должны считаться ESTABLISHED с самого начала связи, они не должны открываться с помощью данных процедур. Однако они могут быть закрыты данными процедурами, и впоследствии они могут быть повторно открыты для той же или другой цели.

Терминал, который больше не в состоянии обрабатывать сигналы в логическом канале, должен предпринять соответствующие действия: они должны включать закрытие логического канала и передачу удаленному терминалу соответствующей (измененной) информации о возможностях.

В приведенном ниже тексте содержится общее описание действия протокола B-LCSE. В случае каких-либо расхождений между ним и официальной спецификацией протокола преимущественную силу имеет официальная спецификация.

C.5.1.1 Общее описание протокола

Открытие логического канала инициируется передачей пользователем в передающем B-LCSE примитива ESTABLISH.request. Сообщение OpenLogicalChannel, содержащее параметры как прямого, так и обратного логического канала, передается равноправному B-LCSE, и запускается таймер T103. Если в ответ на сообщение OpenLogicalChannel получено сообщение OpenLogicalChannelAck, то таймер T103 останавливается, равноправному принимающему B-LCSE передается сообщение OpenLogicalChannelConfirm и пользователь с помощью примитива ESTABLISH.confirm информируется о том, что логический канал был успешно открыт. Начиная с этого момента логический канал может использоваться для передачи и приема информации пользователя. Однако если в ответ на сообщение OpenLogicalChannel получено сообщение OpenLogicalChannelReject, то таймер T103 останавливается, и пользователь с помощью примитива RELEASE.indication извещается о том, что равноправный пользователь B-LCSE отказался от организации логического канала.

Если за этот период окончился срок действия таймера T103, то пользователь информируется с помощью примитива RELEASE.indication, а равноправному принимающему B-LCSE передается сообщение CloseLogicalChannel.

Успешно организованный логический канал может быть закрыт с помощью примитива RELEASE.request, который посылается пользователем в передающий B-LCSE. Равноправному принимающему B-LCSE передается сообщение CloseLogicalChannel, и запускается таймер T103. Когда получено сообщение CloseLogicalChannelAck, таймер T103 останавливается, и пользователь с помощью примитива RELEASE.confirm извещается о том, что логический канал был успешно закрыт.

Если за этот период окончился срок действия таймера T103, то пользователь информируется об этом с помощью примитива RELEASE.indication.

Перед тем как в ответ на ранее отправленное сообщение OpenLogicalChannel будет получено любое из сообщений OpenLogicalChannelAck или OpenLogicalChannelReject, пользователь в передающем B-LCSE может закрыть логический канал с помощью примитива RELEASE.request.

До того как в ответ на ранее отправленное сообщение CloseLogicalChannel будет получено любое из сообщений CloseLogicalChannelAck, пользователь в передающем B-LCSE может организовать новый логический канал, передавая примитив ESTABLISH.request.

C.5.1.2 Общее описание протокола – принимающий B-LCSE

Когда в принимающем B-LCSE получено сообщение OpenLogicalChannel, пользователь информируется о получении запроса на открытие нового логического канала с помощью примитива ESTABLISH.indication. Пользователь принимающего B-LCSE сообщает о получении запроса на организацию логического канала путем передачи примитива ESTABLISH.response, и равноправному

передающему B-LCSE посылается сообщение OpenLogicalChannelAck. Начиная с этого момента прямой канал двунаправленного логического канала можно использовать для приема информации пользователя. Пользователь принимающего B-LCSE информирует об отклонении запроса на организацию логического канала, отправляя примитив RELEASE.request, а на равноправный передающий B-LCSE передается сообщение OpenLogicalChannelReject.

Когда принимающий B-LCSE получает сообщение OpenLogicalChannelConfirm, пользователь с помощью примитива ESTABLISH.confirm извещается об организации двунаправленного логического канала. Начиная с этого момента можно использовать обратный канал двунаправленного логического канала для передачи информации пользователя.

Успешно организованный логический канал может быть закрыт, если принимающий B-LCSE получает сообщение CloseLogicalChannel. Пользователь принимающего B-LCSE информируется об этом с помощью примитива RELEASE.indication, а в равноправный передающий B-LCSE передается сообщение CloseLogicalChannelAck.

С.5.1.3 Разрешение конфликтов

При одновременном инициировании запросов на открытие логических каналов могут возникнуть конфликты. Наличие конфликта можно установить, если знать возможности, которыми обменялись терминалы. В других случаях оба терминала могут инициировать открытие двунаправленного логического канала с одной и той же целью, несмотря на то, что точные требуемые параметры могут быть различными и ресурсы обоих терминалов достаточны для обоих запросов. Терминалы должны быть в состоянии обнаруживать, когда возникают обе такие ситуации, и должны действовать следующим образом.

Перед тем как могут быть открыты логические каналы, один из терминалов должен быть определен как ведущий терминал, а другой – как ведомый терминал. Определенный в С.2 протокол обеспечивает одно средство для такого решения. Ведущий терминал должен немедленно отклонять любой запрос от ведомого терминала, который идентифицируется им как конфликтующий запрос. Ведомый терминал может идентифицировать такие конфликты, но он должен отвечать на запрос ведущего терминала, зная, что его предыдущий запрос будет отвергнут.

При конфликте второго описанного выше типа невозможно отличить случай, при котором фактически требуются два двунаправленных канала, от случая, когда требуется только один двунаправленный канал. Терминалы должны в своем ответе исходить из того, что нужен только один двунаправленный канал, но терминал может впоследствии повторить свой запрос, если это предположение было неверным.

Если ведомая конечная точка имеет симметричные ограничения по возможностям, то рекомендуется следующий порядок действий для сведения к минимуму вероятности открытия конечными точками конфликтующих логических каналов. Если ведущий и ведомый терминалы указали выбранные возможности приема для определенного типа среды, то ведомый терминал должен попытаться открыть логический канал для наиболее предпочтительной возможности ведущего терминала, которая соответствует имеющимся у него самого возможностям (при этом предпочтительность возможности определяется порядком, в котором ведущий терминал расположил свои возможности); а ведущий терминал должен попытаться открыть логический канал для наиболее предпочтительной для него возможности, которая соответствует имеющейся у ведомого терминала возможности (при этом предпочтительность возможности определяется порядком, в котором он расположил свои возможности).

Например, если ведущий терминал объявил возможности для G.723.1, G.729 и G.711, а ведомый терминал указал возможности для G.711 и G.729 (расположив наиболее предпочтительные возможности в обоих случаях первыми), то и ведущий терминал, и ведомый терминал должны пытаться открыть логические каналы для G.729.

После того как запрос на открытие логического канала был отвергнут ведущим терминалом по причине, значение которой равно masterSlaveConflict, ведомый терминал отвечает за открытие бесконфликтного канала.

Если ведомый терминал обнаруживает наличие конфликта, а ведущий терминал не отвергает конфликтующий открытый логический канал, то ведомый терминал должен закрыть конфликтующий канал. В случае конфликтующих логических каналов при симметричных ограничениях возможности ведомый терминал должен открыть соответствующий логический канал, используя процедуру замены, и должным образом закрыть конфликтующий логический канал.

С.5.2 Связь между В-LCSE и пользователем В-LCSE

С.5.2.1 Обмен примитивами между В-LCSE и пользователем В-LCSE

Связь между В-LCSE и пользователем В-LCSE осуществляется с помощью примитивов, приведенных в таблице С.15.

Таблица С.15/Н.245 – Примитивы и параметры

Универсальное название	Тип			
	запрос	индикация	ответ	подтверждение
ESTABLISH	FORWARD_PARAM REVERSE_PARAM	FORWARD_PARAM REVERSE_PARAM	REVERSE_DATA	REVERSE_DATA
RELEASE	CAUSE	SOURCE CAUSE	не определен (Примечание 2)	– (Примечание 1)
ERROR	не определен	ERRCODE	не определен	не определен
ПРИМЕЧАНИЕ 1. – "-" означает отсутствие параметров.				
ПРИМЕЧАНИЕ 2. – "не определен" означает, что этот примитив не определен.				

С.5.2.2 Определение примитивов

Ниже приведены определения указанных примитивов:

- Примитивы типа ESTABLISH используются для организации логического канала для аудиовизуальной связи и передачи данных.
- Примитивы типа RELEASE используются для сброса логического канала.
- Примитивы типа ERROR сообщают объекту управления об ошибках В-LCSE.

С.5.2.3 Определение параметров

Ниже даны определения параметров примитивов, приведенных в таблице С.15:

- Параметр FORWARD_PARAM определяет параметры, относящиеся к прямому каналу, то есть каналу от терминала, содержащего передающий В-LCSE, к терминалу, содержащему принимающий В-LCSE. Этот параметр отображается в поле forwardLogicalChannelParameters сообщения OpenLogicalChannel и прозрачно передается равноправному пользователю LCSE.
- Параметр REVERSE_PARAM определяет параметры, относящиеся к обратному каналу, то есть каналу от терминала, содержащего принимающий В-LCSE, к терминалу, содержащему передающий В-LCSE. Этот параметр отображается в поле reverseLogicalChannelParameters сообщения OpenLogicalChannel и прозрачно передается равноправному пользователю LCSE.
- Параметр REVERSE_DATA определяет некоторые параметры, относящиеся к обратному каналу, то есть каналу от терминала, содержащего принимающий В-LCSE, к терминалу, содержащему передающий В-LCSE. Этот параметр отображается в поле reverseLogicalChannelParameters сообщения OpenLogicalChannelAck и прозрачно передается равноправному пользователю В-LCSE.
- Параметр SOURCE показывает пользователю В-LCSE источник сброса логического канала. Параметр SOURCE имеет значение "USER" ("пользователь") или "В-LCSE", указывая или на пользователя В-LCSE, или на В-LCSE. Последнее значение может появиться как результат ошибки протокола.

- e) Параметр CAUSE показывает причину, по которой равноправный пользователь В-LCSE отклонил запрос на организацию логического канала. Параметр CAUSE отсутствует, когда параметр SOURCE имеет значение "В-LCSE".
- f) Параметр ERRCODE показывает тип ошибки В-LCSE. В таблице С.19 приведены разрешенные значения параметра ERRCODE.

С.5.2.4 Состояния В-LCSE

Описанные ниже состояния задают разрешенную последовательность обмена примитивами между В-LCSE и пользователем В-LCSE, а также обмен сообщениями между равноправными В-LCSE. Состояния определяются отдельно для передающего В-LCSE и принимающего В-LCSE. Передающий В-LCSE имеет следующие состояния:

Состояние 0: RELEASED

Логический канал сброшен. Этот логический канал не должен использоваться для передачи или приема данных.

Состояние 1: AWAITING ESTABLISHMENT

Передающий В-LCSE ожидает организации логического канала с равноправным принимающим В-LCSE. Логический канал не должен использоваться для передачи или приема данных.

Состояние 2: ESTABLISHED

Установлено соединение логического канала между равноправными В-LCSE. Этот логический канал может использоваться для передачи и приема данных.

Состояние 3: AWAITING RELEASE

Передающий В-LCSE ожидает сброса логического канала, установленного с равноправным принимающим В-LCSE. Логический канал не должен использоваться для передачи данных, но может продолжаться прием данных.

Принимающий В-LCSE имеет следующие состояния:

Состояние 0: RELEASED

Логический канал сброшен. Логический канал не должен использоваться для приема или передачи данных.

Состояние 1: AWAITING ESTABLISHMENT

Принимающий В-LCSE ожидает организации логического канала с равноправным передающим В-LCSE. Этот логический канал не должен использоваться для приема или передачи данных.

Состояние 2: AWAITING CONFIRMATION

Принимающий В-LCSE ожидает квитирования, подтверждающего организацию логического канала с равноправным передающим В-LCSE. Логический канал не должен использоваться для передачи данных, но принимать по нему данные можно.

Состояние 3: ESTABLISHED

Установлено соединение логического канала между равноправными В-LCSE. Этот логический канал может использоваться для приема и передачи данных.

С.5.2.5 Диаграмма переходов между состояниями

Здесь определяется разрешенная последовательность обмена примитивами между В-LCSE и пользователем В-LCSE. Разрешенная последовательность примитивов связана с состояниями В-LCSE, которые видит пользователь В-LCSE. Разрешенные последовательности определяются отдельно для каждого передающего В-LCSE и принимающего В-LCSE, как показано на рисунках С.15 и С.16 соответственно.

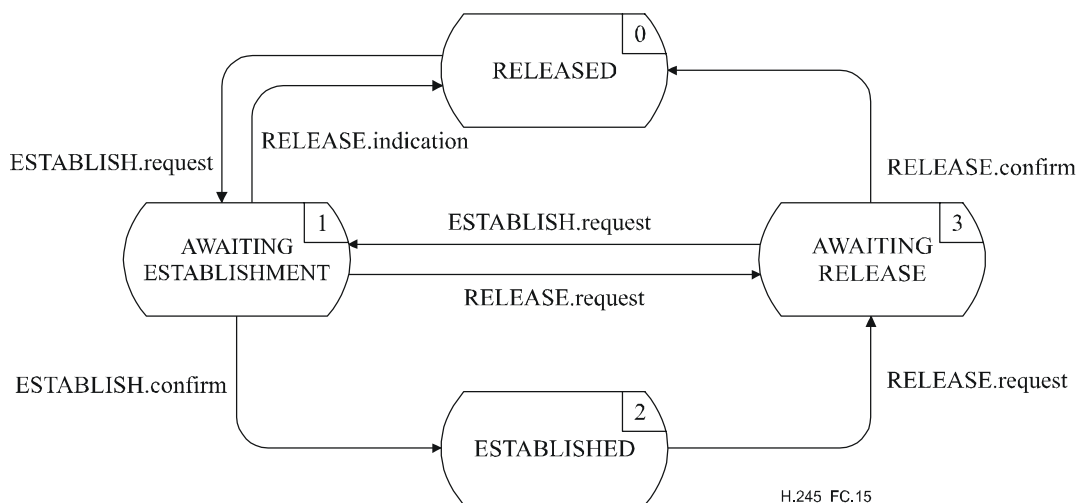


Рисунок С.15/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в передающем В-LCSE

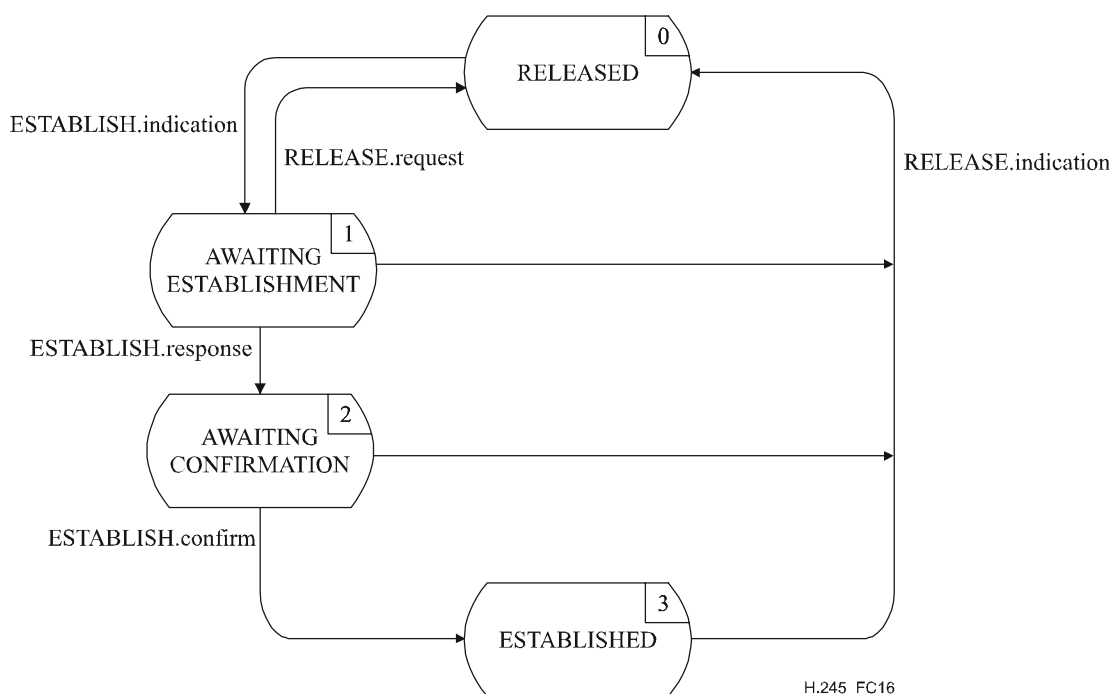


Рисунок С.16/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в принимающем В-LCSE

С.5.3 Связь между равноправными В-LCSE

С.5.3.1 Сообщения В-LCSE

В таблице С.16 показаны определенные в Приложении А сообщения и поля В-LCSE, которые относятся к протоколу В-LCSE.

Таблица С.16/Н.245 – Названия и поля сообщений В-LCSE

Функция	Сообщение	Направление	Поле
establishment	OpenLogicalChannel	O → I (Примечание)	forwardLogicalChannelNumber forwardLogicalChannelParameters reverseLogicalChannelParameters
	OpenLogicalChannelAck	O ← I	forwardLogicalChannelNumber reverseLogicalChannelParameters
	OpenLogicalChannelReject	O ← I	forwardLogicalChannelNumber cause
	OpenLogicalChannelConfirm	O → I	forwardLogicalChannelNumber
release	CloseLogicalChannel	O → I	forwardLogicalChannelNumber source
	CloseLogicalChannelAck	O ← I	forwardLogicalChannelNumber
ПРИМЕЧАНИЕ. – Направление: O – исходящее, I – входящее.			

С.5.3.2 Переменные состояния В-LCSE

В передающем В-LCSE определена следующая переменная состояния:

out_LCN

Эта переменная состояния устанавливает различие между передающими В-LCSE. Она инициализируется при инициализации передающего В-LCSE. Значение out_LCN используется для установки значения поля forwardLogicalChannelNumber сообщений В-LCSE, передаваемых передающим В-LCSE. Для сообщений В-LCSE, полученных в передающем В-LCSE, значение поля forwardLogicalChannelNumber в сообщении идентично значению out_LCN.

Для принимающего В-LCSE определена следующая переменная состояния:

in_LCN

Эта переменная состояния устанавливает различие между принимающими В-LCSE. Она инициализируется при инициализации принимающего В-LCSE. Значение in_LCN используется для установки значения поля forwardLogicalChannelNumber сообщений В-LCSE, переданных принимающим В-LCSE. Для сообщений В-LCSE, полученных принимающим В-LCSE, значение поля forwardLogicalChannelNumber в сообщении идентично значению in_LCN.

С.5.3.3 Таймеры В-LCSE

Для передающего и принимающего В-LCSE определен приведенный ниже таймер:

T103

В передающем В-LCSE этот таймер используется при состояниях AWAITING RELEASE и AWAITING ESTABLISHMENT. Он определяет максимально возможный период без получения сообщения OpenLogicalChannelAck, OpenLogicalChannelReject или CloseLogicalChannelAck.

В принимающем В-LCSE этот таймер используется при состоянии AWAITING CONFIRMATION. Он определяет максимально возможный период без получения сообщения OpenLogicalChannelConfirm.

С.5.4 Процедуры В-LCSE

С.5.4.1 Введение

На рисунке С.17 показаны все примитивы и их параметры, а также сообщения для передающего и принимающего В-LCSE.

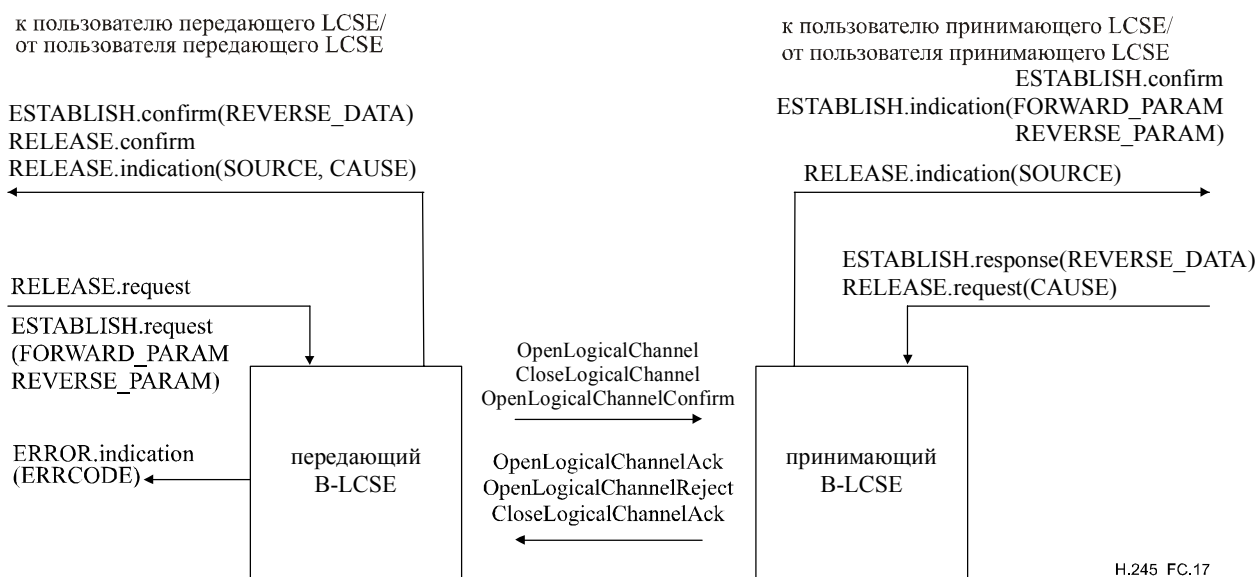


Рисунок С.17/Н.245 – Примитивы и сообщения в объекте сигнализации двунаправленного логического канала

С.5.4.2 Принимаемые по умолчанию значения параметров примитивов

Параметры примитивов индикации и квитирования принимают значения, показанные в таблице С.17, если в диаграммах SDL в явном виде не задано иное.

Таблица С.17/Н.245 – Принимаемые по умолчанию значения параметров примитивов

Примитив	Параметр	Значение, используемое по умолчанию (Примечание)
ESTABLISH.indication	FORWARD_PARAM	OpenLogicalChannel.forwardLogicalChannelParameters
	REVERSE_PARAM	OpenLogicalChannel.reverseLogicalChannelParameters
ESTABLISH.confirm	REVERSE_DATA	OpenLogicalChannelAck.reverseLogicalChannelParameters
RELEASE.indication	SOURCE	CloseLogicalChannel.source
	CAUSE	Null
ПРИМЕЧАНИЕ. – Параметр примитива должен кодироваться как null (ноль), если указанное поле сообщения не присутствует в данном сообщении.		

С.5.4.3 Принимаемые по умолчанию значения полей сообщений

Поля сообщений принимают значения, показанные в таблице С.18, если в диаграммах SDL в явном виде не задано иное.

Таблица С.18/Н.245 – Принимаемые по умолчанию значения полей сообщений

Сообщение	Поле	Значение по умолчанию (Примечание)
OpenLogicalChannel	forwardLogicalChannelNumber forwardLogicalChannelParameters reverseLogicalChannelParameters	out_LCN ESTABLISH.request (FORWARD_PARAM) ESTABLISH.request (REVERSE_PARAM)
OpenLogicalChannelAck	forwardLogicalChannelNumber reverseLogicalChannelParameters	in_LCN ESTABLISH.response (REVERSE_DATA)
OpenLogicalChannelReject	forwardLogicalChannelNumber cause	in_LCN RELEASE.request(CAUSE)
OpenLogicalChannelConfirm	forwardLogicalChannelNumber	out_LCN
CloseLogicalChannel	forwardLogicalChannelNumber source	out_LCN user
CloseLogicalChannelAck	forwardLogicalChannelNumber	in_LCN
ПРИМЕЧАНИЕ. – Поле сообщения не должно кодироваться, если соответствующий параметр примитива равен null (нулю), т. е. он не присутствует.		

С.5.4.4 Значения параметра ERRCODE

Параметр ERRCODE примитива ERROR.indication показывает определенное условие возникновения ошибки. В таблице С.19 показаны значения, которые может принимать параметр ERRCODE в передающем В-LCSE, а в таблице С.20 показаны значения, которые может принимать параметр ERRCODE в принимающем В-LCSE.

Таблица С.19/Н.245 – Значения параметра ERRCODE в передающем В-LCSE

Тип ошибки	Код ошибки	Условие возникновения ошибки	Состояние
несоответствующее сообщение	A	OpenLogicalChannelAck	RELEASED
	B	OpenLogicalChannelReject	RELEASED ESTABLISHED
	C	CloseLogicalChannelAck	ESTABLISHED
нет ответа от равноправного В-LCSE	D	истечение срока действия таймера T103	AWAITING ESTABLISHMENT AWAITING RELEASE

Таблица С.20/Н.245 – Значения параметра ERRCODE в принимающем В-LCSE

Тип ошибки	Код ошибки	Условие возникновения ошибки	Состояние
несоответствующее сообщение	E	OpenLogicalChannelConfirm	AWAITING ESTABLISHMENT
нет ответа от равноправного В-LCSE	F	истечение срока действия таймера T103	AWAITING CONFIRMATION

С.5.4.5 Диаграммы языка описания структур (SDL)

Процедуры передачи сообщений передающего В-LCSE и принимающего В-LCSE представлены в форме SDL на рисунках С.18 и С.19, соответственно.

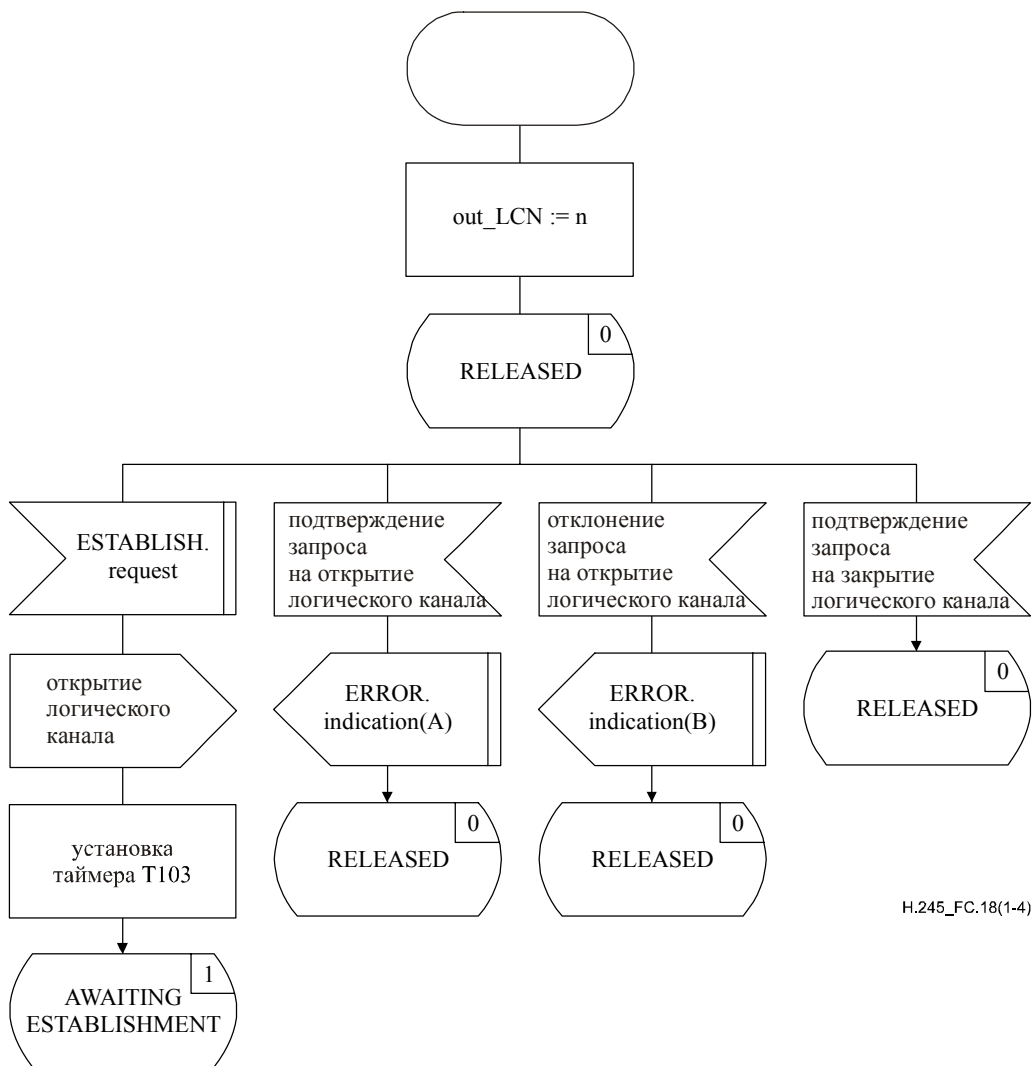


Рисунок С.18/Н.245 – Диаграммы SDL для передающего В-LCSE (лист 1 из 4)

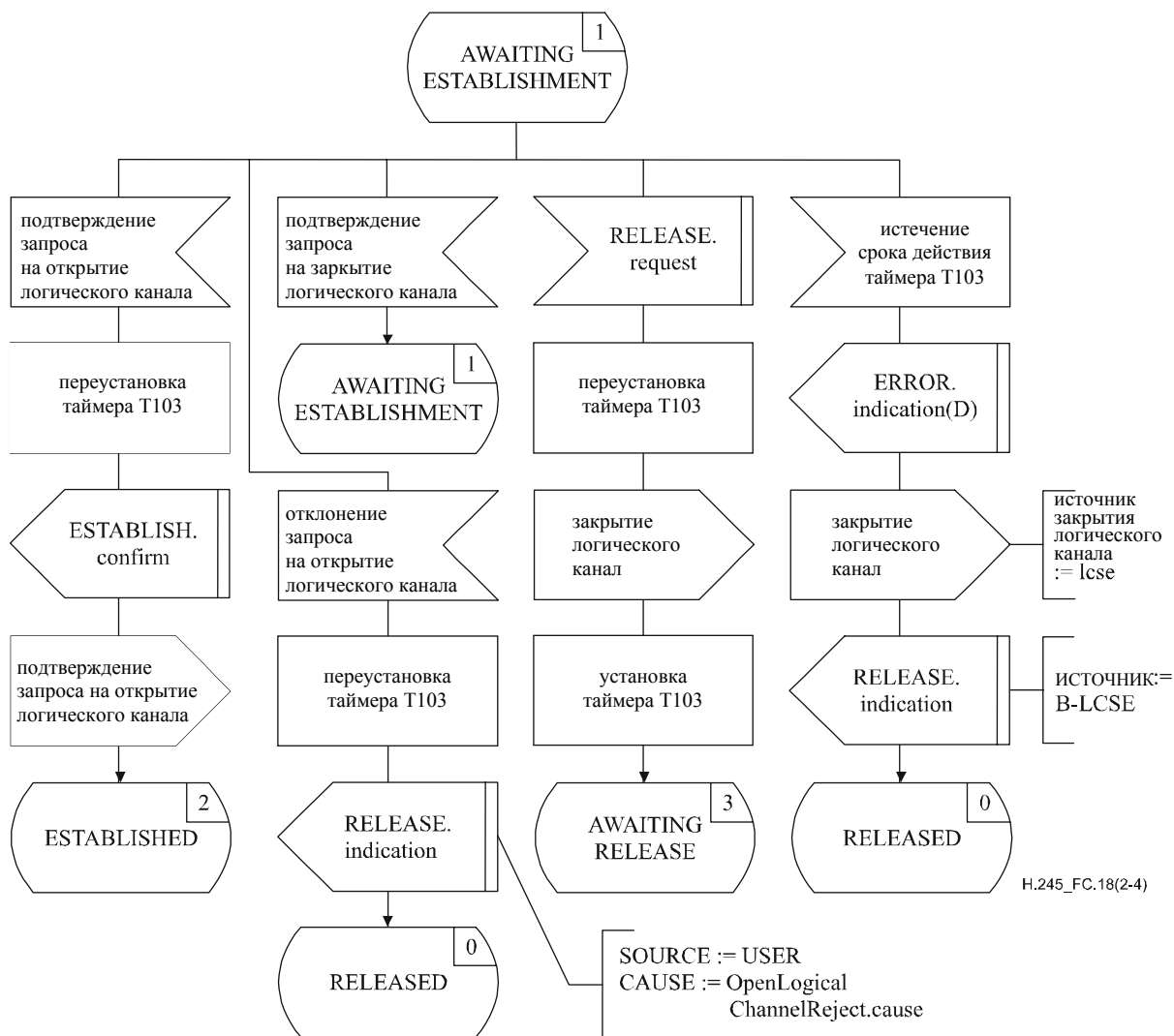


Рисунок С.18/Н.245 – Диаграммы SDL для передающего В-LCSE (лист 2 из 4)

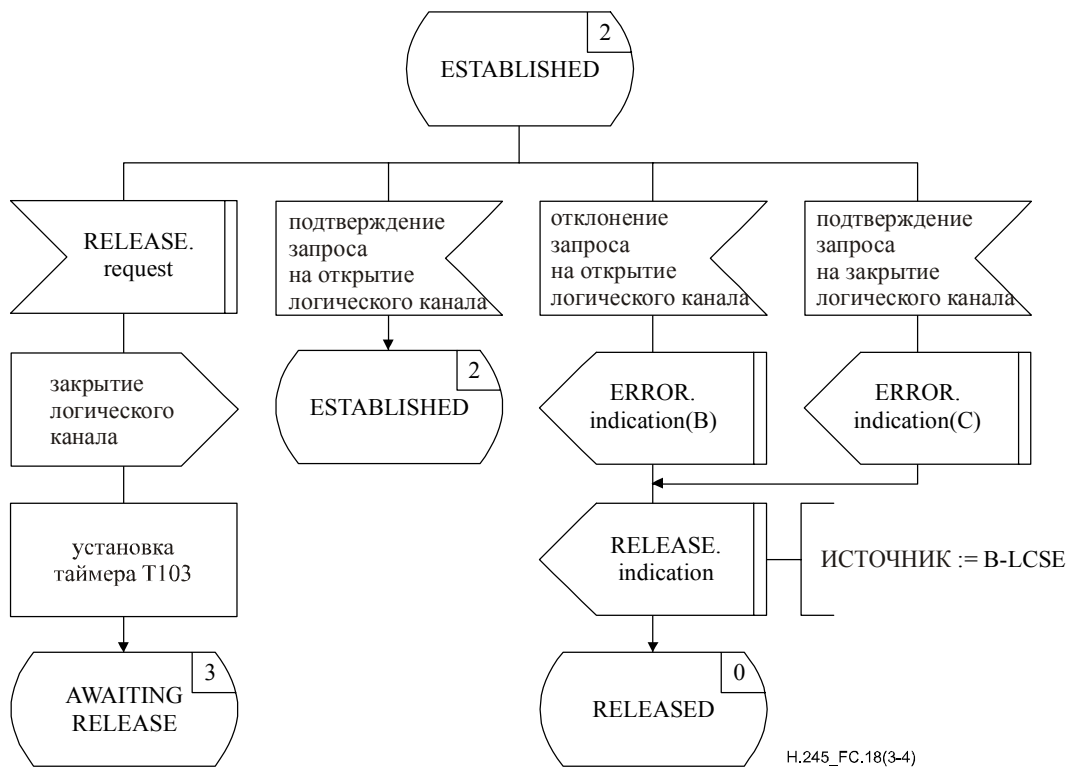


Рисунок С.18/Н.245 – Диаграммы SDL для передающего B-LCSE (лист 3 из 4)

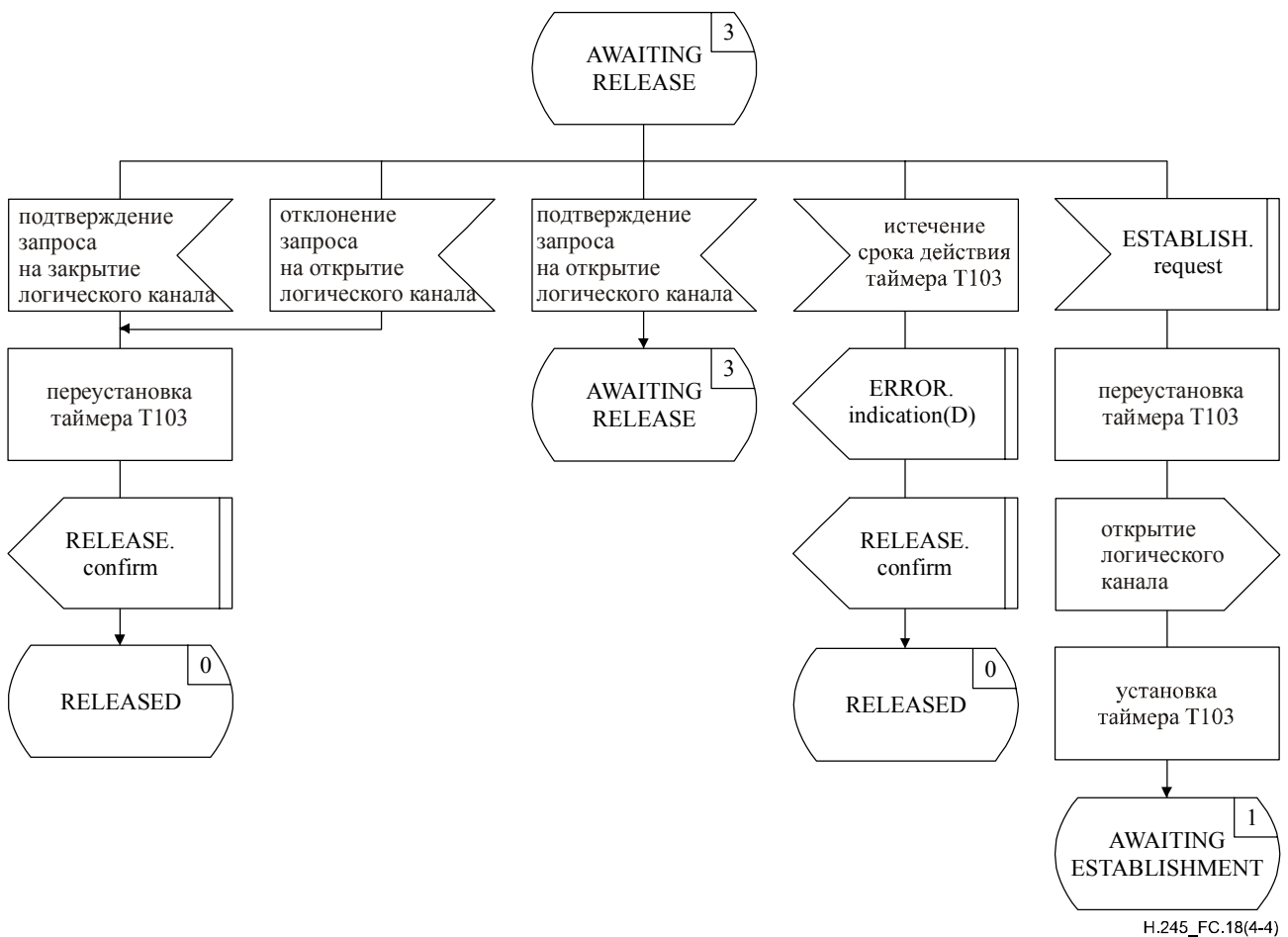


Рисунок С.18/Н.245 – Диаграммы SDL для передающего B-LCSE (лист 4 из 4)

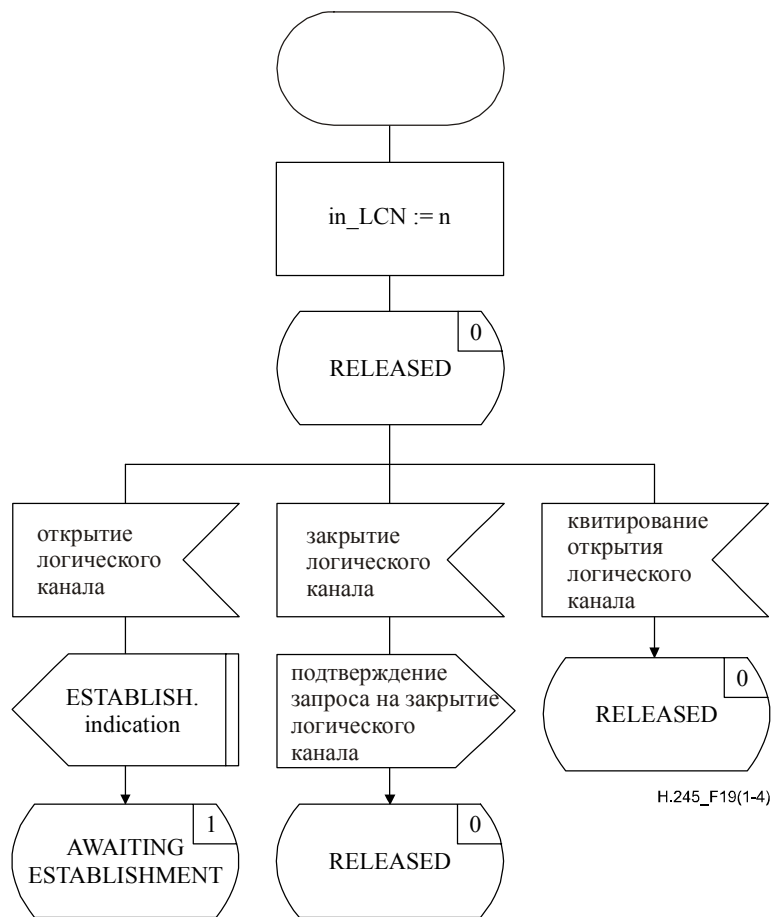


Рисунок С.19/Н.245 – Диаграммы SDL для принимающего В-LCSE (лист 1 из 4)

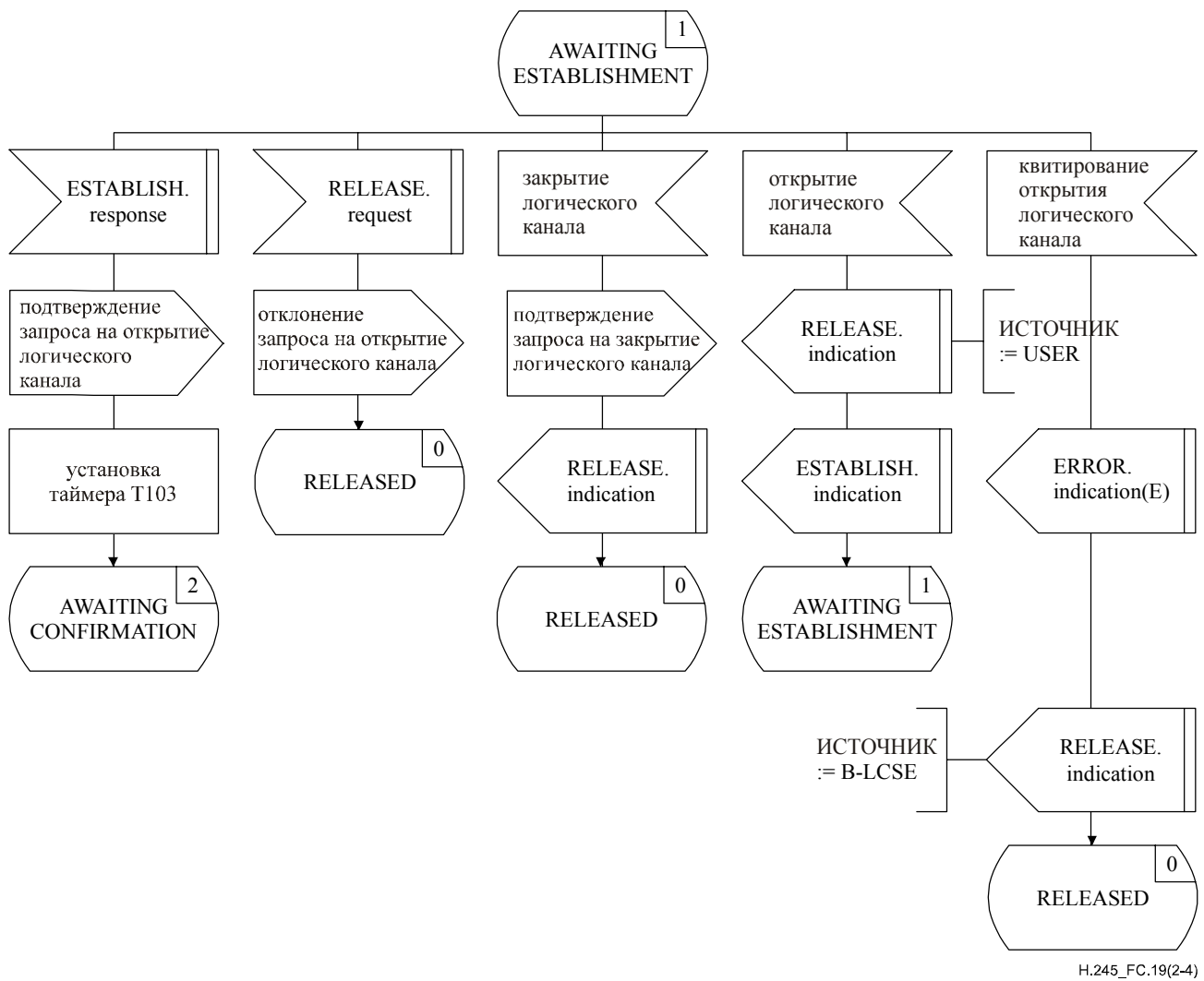


Рисунок С.19/Н.245 – Диаграммы SDL для принимающего B-LCSE (лист 2 из 4)

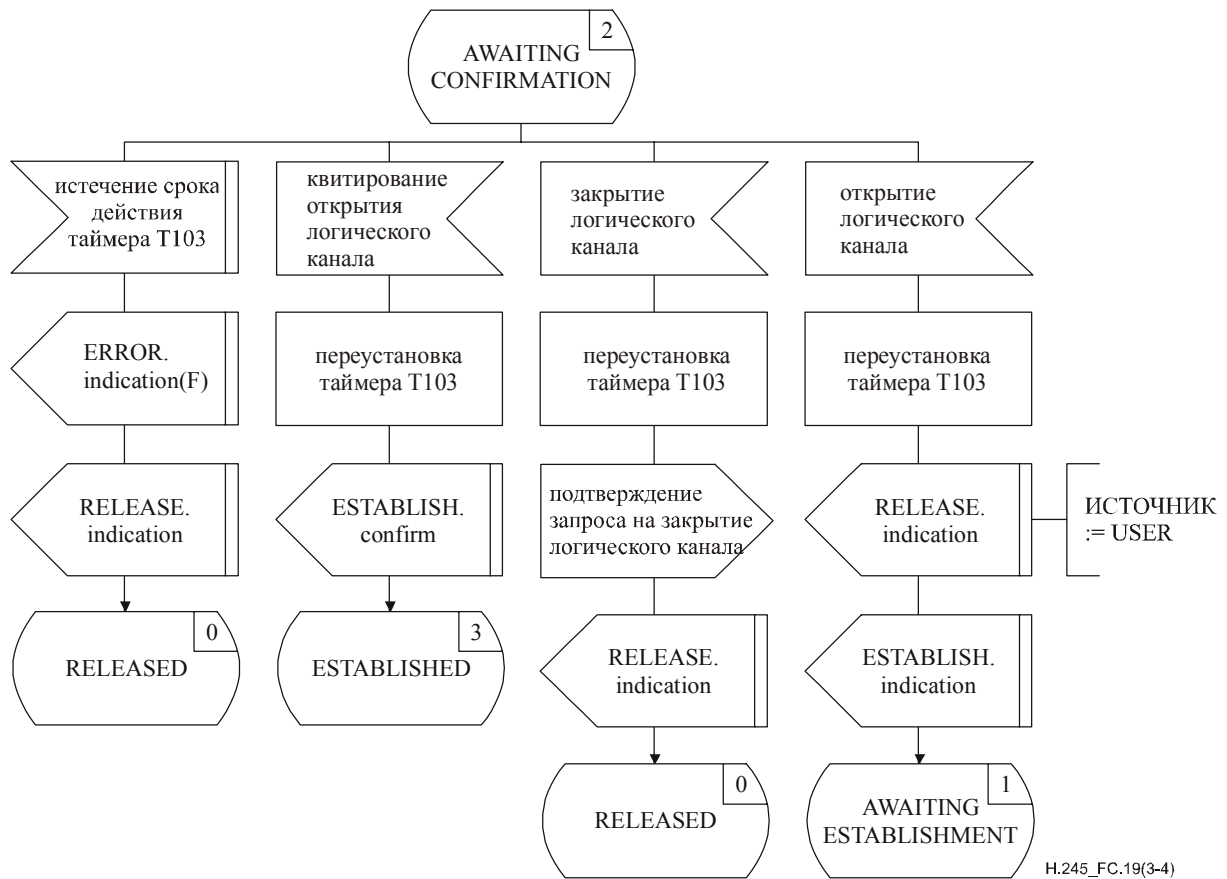


Рисунок С.19/Н.245 – Диаграммы SDL для принимающего В-LCSE (лист 3 из 4)

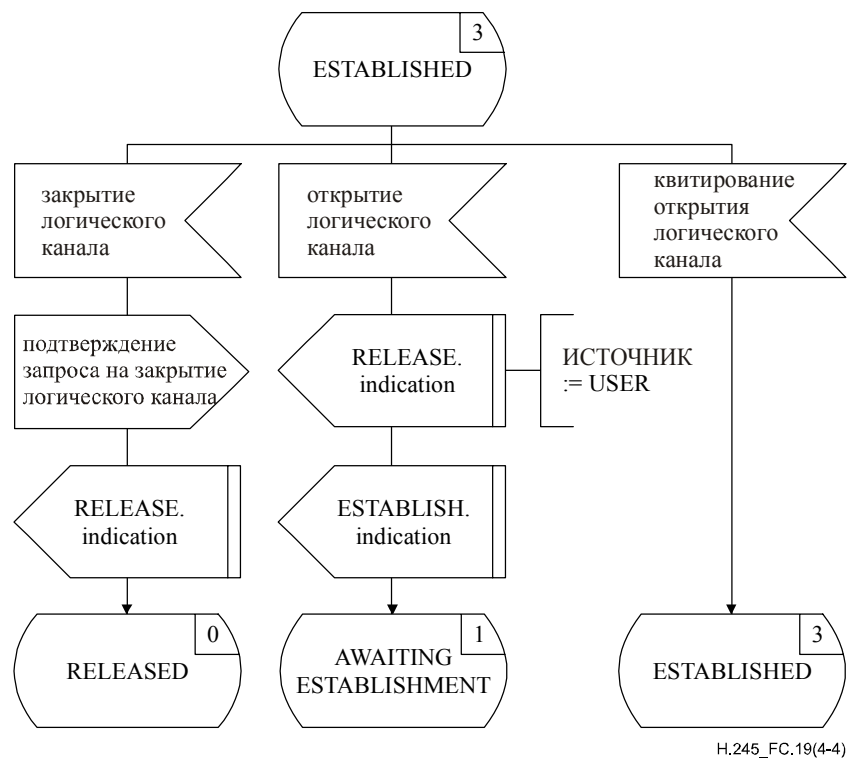


Рисунок С.19/Н.245 – Диаграммы SDL для принимающего В-LCSE (лист 4 из 4)

С.6 Процедуры закрытия логического канала

С.6.1 Введение

Эти процедуры используются терминалом для обращения к удаленному терминалу с запросом о закрытии логического канала. Обратите внимание на то, что они представляют собой только процедуры запроса о закрытии; фактическое закрытие логического канала производится с использованием процедур LCSE и B-LCSE. Эти процедуры называются далее процедурами объекта сигнализации закрытия логического канала (CLCSE). Процедуры определяются с помощью примитивов и состояний в интерфейсе между CLCSE и пользователем CLCSE. Информация протокола передается равноправному CLCSE с помощью соответствующих сообщений, которые определены в Приложении А. Существует передающий CLCSE и принимающий CLCSE. На каждом из передающих и принимающих концов существует одна копия CLCSE для каждого логического канала.

Если терминал не способен обработать поступающие сигналы, то он может использовать эти процедуры для запроса о закрытии соответствующих логических каналов.

Терминал, который отвечает на такой ответ положительно, то есть выдачей примитива CLOSE.response, должен инициализировать закрытие этого логического канала, передавая как можно быстрее примитив RELEASE.request соответствующему LCSE или B-LCSE.

В приведенном ниже тексте содержится общее описание действия данного протокола. В случае каких-либо расхождений между ним и официальной спецификацией протокола преимущественную силу имеет официальная спецификация.

С.6.1.1 Общее описание протокола – передающий CLCSE

Процедура запроса о закрытии логического канала инициализируется передачей пользователем в передающем CLCSE примитива CLOSE.request. Сообщение RequestChannelClose передается равноправному принимающему CLCSE, и запускается таймер T108. Если в ответ на сообщение RequestChannelClose получено сообщение RequestChannelCloseAck, то таймер T108 останавливается, и пользователь с помощью примитива CLOSE.confirm информируется о том, что процедура запроса о закрытии логического канала выполнена успешно. Однако если в ответ на сообщение RequestChannelClose получено сообщение RequestChannelCloseReject, то таймер T108 останавливается, и пользователь с помощью примитива REJECT.indication информируется о том, что равноправный пользователь CLCSE отказался закрывать логический канал.

Если заканчивается срок действия таймера T108, то пользователь передающего CLCSE информируется об этом с помощью примитива REJECT.indication, и передается сообщение RequestChannelCloseRelease.

С.6.1.2 Общее описание протокола – принимающий CLCSE

Если в принимающем CLCSE получено сообщение RequestChannelClose, то пользователь информируется о получении запроса на закрытие логического канала с помощью примитива CLOSE.indication. Пользователь принимающего CLCSE подтверждает получение запроса на закрытие логического канала путем выдачи примитива CLOSE.response, и равноправному передающему CLCSE передается сообщение RequestChannelCloseAck. Пользователь принимающего CLCSE сообщает об отклонении запроса на закрытие логического канала с помощью примитива REJECT.request, и равноправному передающему CLCSE передается сообщение RequestChannelCloseReject.

С.6.2 Связь между CLCSE и пользователем CLCSE

С.6.2.1 Обмен примитивами между CLCSE и пользователем CLCSE

Связь между CLCSE и пользователем CLCSE производится с помощью примитивов, приведенных в таблице С.21.

Таблица С.21/Н.245 – Примитивы и параметры

Универсальное название	Тип			
	запрос	индикация	ответ	Подтверждение
CLOSE	– (Примечание 1)	–	–	–
REJECT	CAUSE	SOURCE CAUSE	не определен (Примечание 2)	не определен
ПРИМЕЧАНИЕ 1. – "-" означает отсутствие параметров. ПРИМЕЧАНИЕ 2. – "не определен" означает, что примитив не определен.				

С.6.2.2 Определение примитивов

Ниже приведены определения указанных примитивов:

- а) Примитивы CLOSE используются для запроса о закрытии логического канала.
- б) Примитивы REJECT используются для того, чтобы отказать в закрытии логического канала.

С.6.2.3 Определение параметров

Ниже дано определение параметров примитивов, приведенных в таблице С.21:

- а) Параметр SOURCE показывает источник примитива REJECT.indication. Параметр SOURCE принимает значения "USER" ("пользователь") или "PROTOCOL" ("протокол"). Последнее значение может появиться в результате истечения срока действия таймера.
- б) Параметр CAUSE показывает причину отказа в закрытии логического канала. Параметр CAUSE не присутствует, если параметр SOURCE имеет значение "PROTOCOL".

С.6.2.4 Состояния CLCSE

Следующие состояния используются для определения разрешенной последовательности обмена примитивами между CLCSE и пользователем CLCSE.

Передающий CLCSE имеет следующие состояния:

Состояние 0: IDLE

CLCSE находится в состоянии ожидания.

Состояние 1: AWAITING RESPONSE

CLCSE ждет ответа от удаленного CLCSE.

Принимающий CLCSE имеет следующие состояния:

Состояние 0: IDLE

CLCSE находится в состоянии ожидания.

Состояние 1: AWAITING RESPONSE

CLCSE ожидает ответа от пользователя CLCSE.

С.6.2.5 Диаграмма переходов между состояниями

В этом подразделе определяется разрешенная последовательность обмена примитивами между CLCSE и пользователем CLCSE. Разрешенные последовательности обмена примитивами определены отдельно для каждого передающего CLCSE и принимающего CLCSE, как показано на рисунках С.20 и С.21, соответственно.

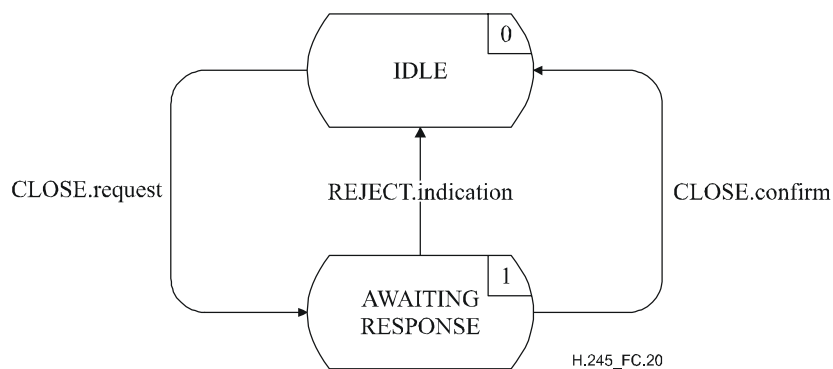


Рисунок С.20/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в передающем CLCSE

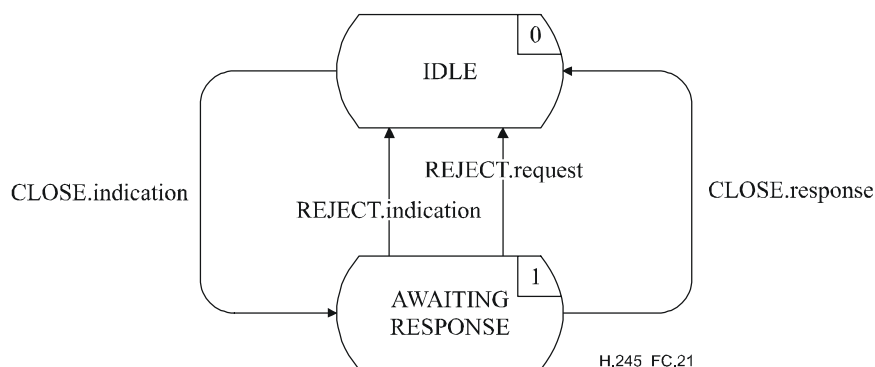


Рисунок С.21/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в принимающем CLCSE

С.6.3 Связь между одноранговыми CLCSE

С.6.3.1 Сообщения

В таблице С.22 показаны определенные в Приложении А сообщения CLCSE и поля, относящиеся к протоколу CLCSE.

Таблица С.22/Н.245 – Названия и поля сообщений CLCSE

Функция	Сообщение	Направление	Поле
transfer	RequestChannelClose	O → I (Примечание)	forwardLogicalChannelNumber
	RequestChannelCloseAck	O ← I	forwardLogicalChannelNumber
	RequestChannelCloseReject	O ← I	forwardLogicalChannelNumber
reset	RequestChannelCloseRelease	O → I	forwardLogicalChannelNumber
ПРИМЕЧАНИЕ. – Направление: O – исходящее, I – входящее.			

С.6.3.2 Переменные состояния CLCSE

Для передающего CLCSE определена следующая переменная состояния:

out_LCN

Эта переменная состояния устанавливает различие между передающими CLCSE. Она инициализируется при инициализации передающего CLCSE. Значение out_LCN используется для установки значения поля forwardLogicalChannelNumber сообщений CLCSE, которые отправлены передающим CLCSE. Для сообщений CLCSE, полученных в передающем CLCSE, значение поля forwardLogicalChannelNumber сообщения идентично значению out_LCN.

Для принимающего CLCSE определена следующая переменная состояния:

in_LCN

Эта переменная состояния устанавливает различие между принимающими CLCSE. Она инициализируется при инициализации принимающего CLCSE. Значение in_LCN используется в качестве значения поля forwardLogicalChannelNumber сообщений CLCSE, переданных принимающим CLCSE. Для сообщений CLCSE, полученных принимающим CLCSE, значение поля forwardLogicalChannelNumber сообщения идентично значению in_LCN.

С.6.3.3 Таймеры CLCSE

Для передающего CLCSE определен следующий таймер:

T108

Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимально возможный период без получения сообщения RequestChannelCloseAck или RequestChannelCloseReject.

С.6.4 Процедуры CLCSE

На рисунке С.22 показаны все примитивы CLCSE и их параметры, а также сообщения для каждого передающего и принимающего CLCSE.

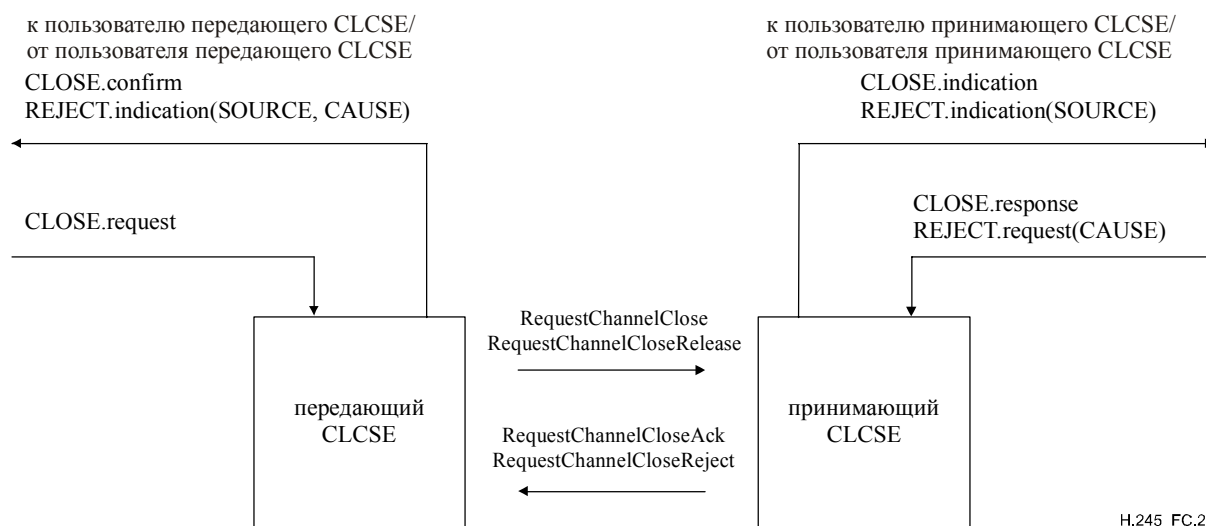


Рисунок С.22/Н.245 – Примитивы и сообщения в объекте сигнализации закрытия логического канала

С.6.4.1 Принимаемые по умолчанию значения параметров примитивов

Параметры примитивов индикации и подтверждения принимают значения, приведенные в таблице С.23, если в диаграммах SDL в явном виде не задано иное.

Таблица С.23/Н.245 – Принимаемые по умолчанию значения параметров примитивов

Примитив	Параметр	Значение по умолчанию
REJECT.indication	SOURCE	USER
	CAUSE	null

С.6.4.2 Принимаемые по умолчанию значения полей сообщений

Поля сообщений принимают значения, приведенные в таблице С.24, если в диаграммах SDL в явном виде не задано иное.

Таблица С.24/Н.245 – Принимаемые по умолчанию значения полей сообщений

Сообщение	Поле	Значение по умолчанию
RequestChannelClose	forwardLogicalChannelNumber	out_LCN
RequestChannelCloseAck	forwardLogicalChannelNumber	in_LCN
RequestChannelCloseReject	forwardLogicalChannelNumber cause	in_LCN REJECT.request(CAUSE)
RequestChannelCloseRelease	forwardLogicalChannelNumber	out_LCN

С.6.4.3 Диаграммы языка описания структур (SDL)

Процедуры передающего CLCSE и принимающего CLCSE представлены в форме SDL на рисунках С.23 и С.24, соответственно.

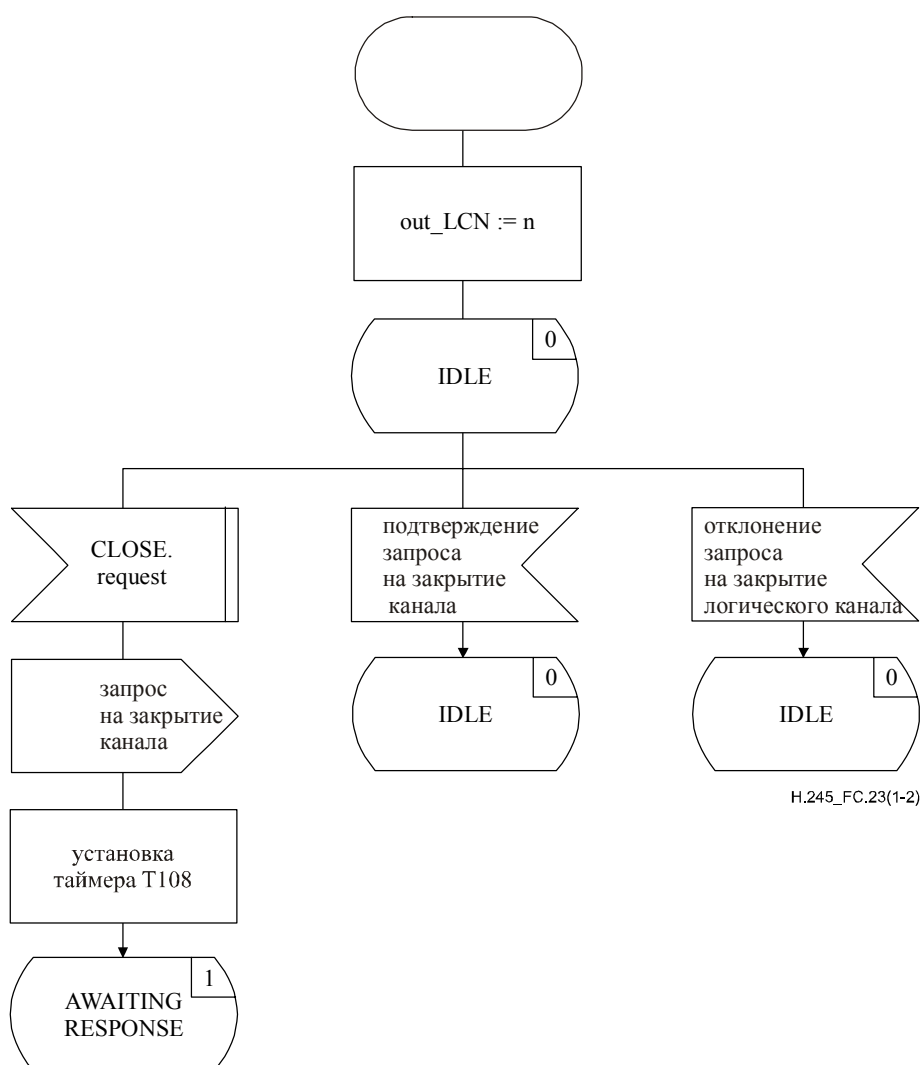


Рисунок С.23/Н.245 – Диаграмма SDL для передающего CLCSE (лист 1 из 2)

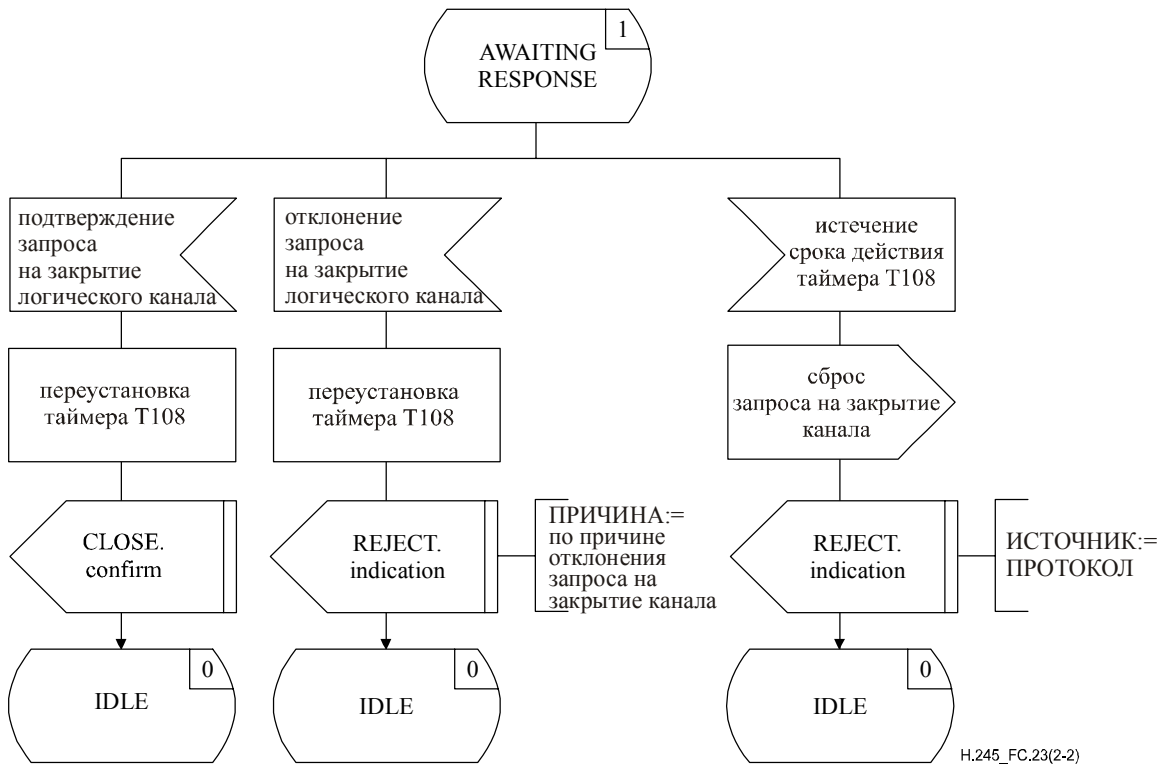


Рисунок С.23/Н.245 – Диаграмма SDL для передающего CLCSE (лист 2 из 2)

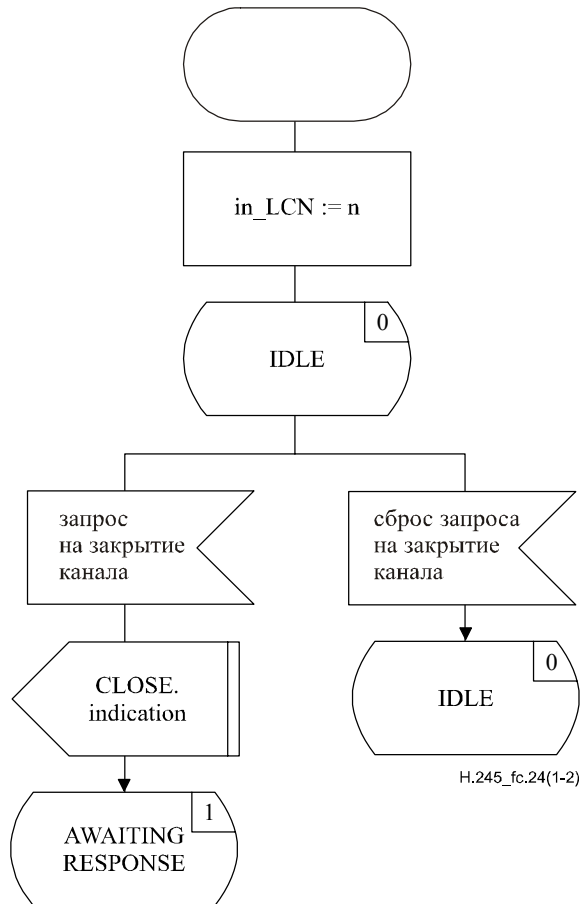


Рисунок С.24/Н.245 – Диаграмма SDL для принимающего CLCSE (лист 1 из 2)

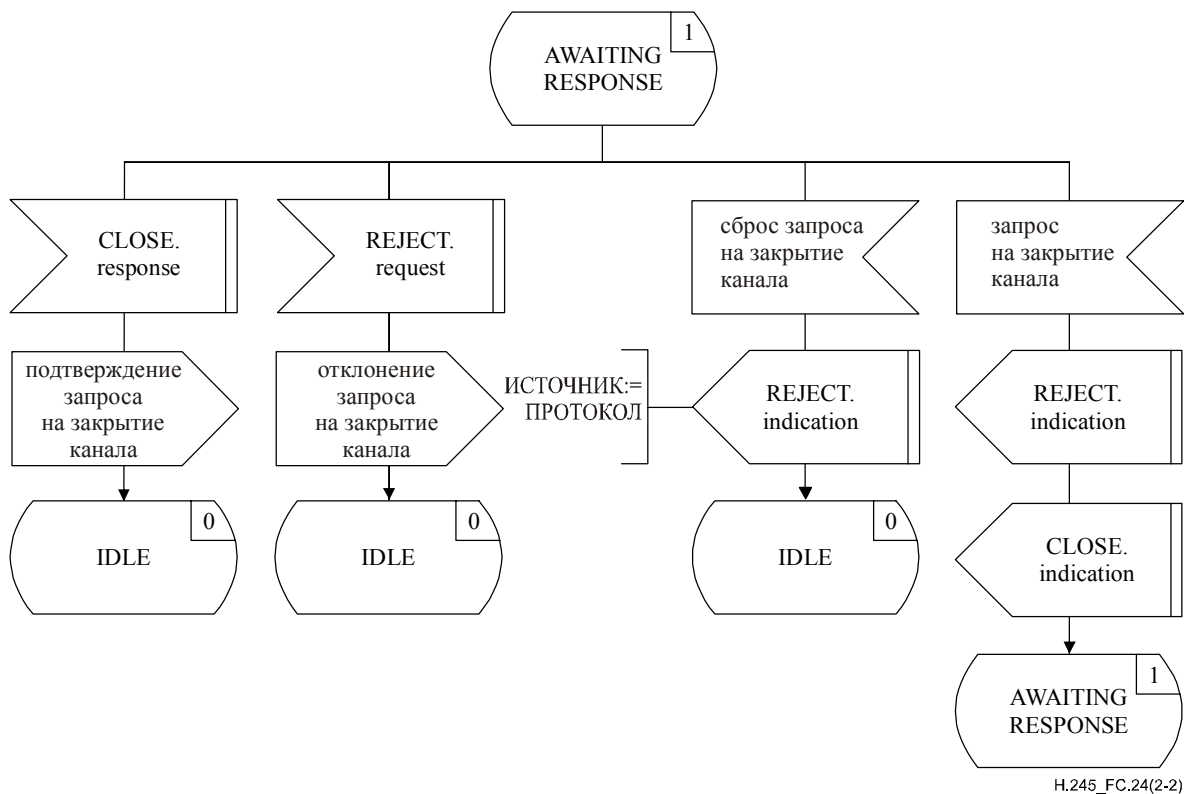


Рисунок С.24/Н.245 – Диаграмма SDL для принимающего CLCSE (лист 2 из 2)

С.7 Процедуры таблицы мультиплексирования Н.223

С.7.1 Введение

Таблица мультиплексирования служит для установления соответствия между каждым октетом MUX-PDU Н.223 [10] и номером конкретного логического канала. Таблица мультиплексирования Н.223 может содержать до 16 элементов с номерами от 0 до 15. Элементы таблицы от 1 до 15 должны передаваться от отправителей получателям в соответствии с определенными ниже процедурами.

Описанные здесь процедуры называются Объектом сигнализации о таблице мультиплексирования (MTSE). Процедуры определяются с помощью примитивов и состояний на интерфейсе между MTSE и пользователем MTSE. Информация протокола передается равноправному MTSE с помощью соответствующих сообщений, которые определены в Приложении А.

Существует передающий MTSE и принимающий MTSE. Для каждого элемента таблицы мультиплексирования существует один экземпляр MTSE.

Передающий терминал использует этот протокол для передачи на удаленный терминал одного или нескольких новых элементов таблицы мультиплексирования. Удаленный терминал может принять или отклонить новые элементы таблицы мультиплексирования. Если удаленный терминал принимает элемент таблицы мультиплексирования, то предыдущий элемент с данным номером элемента заменяется на новый элемент.

Отправитель может деактивировать элемент таблицы мультиплексирования, передав MultiplexEntryDescriptor без elementList. Отправителю запрещено когда-либо использовать деактивированный элемент таблицы мультиплекса. До передачи MultiplexEntrySend отправитель должен перестать использовать те элементы, которые им описаны. Он не должен повторно использовать эти элементы до тех пор, пока он не получит MultiplexEntrySendAck. Применение этой процедура обусловлено тем, что если использование данных элементов таблицы мультиплексирования не прекращается до отправки MultiplexEntrySend, то ошибки могут привести к неопределенности у получателя.

Отправитель должен прекратить использование деактивированных элементов до отправки сообщения MultiplexEntrySend, указывающего, что они были деактивированы. Деактивированные элементы могут применяться снова в любое время; для активирования такого элемента передается сообщение MultiplexEntrySend. Деактивация элементов, которые больше не нужны отправителю, может повысить вероятность обнаружения ошибок в поле кода мультиплексирования Multiplex Code H.223.

ПРИМЕЧАНИЕ. – В то время как некоторые элементы таблицы мультиплексирования обновляются, другие (активные) элементы могут по-прежнему использоваться. Кроме того, элемент таблицы мультиплексирования может удаляться с помощью того же сообщения MultiplexEntrySend, которое используется для изменения других элементов таблицы мультиплексирования.

В начале связи, если в соответствующей Рекомендации не указано иное, для передачи доступен только элемент таблицы 0, а элементы таблицы от 1 до 15 деактивированы.

В любое время может использоваться процедура запроса элемента таблицы (Request Multiplex Entry), вызывающая повторную передачу удаленным терминалом указанных элементов таблицы мультиплексирования, например после прерывания или неопределенности по другой причине.

В приведенном ниже тексте содержится общее описание действия данного протокола. В случае каких-либо расхождений между ним и официальной спецификацией протокола преимущественную силу имеет официальная спецификация.

С.7.1.1 Общее описание протокола – передающий MTSE

Процедура запроса о передаче элемента таблицы мультиплексирования инициируется, когда пользователь в передающем MTSE передает примитив TRANSFER.request. Сообщение MultiplexEntrySend передается на равноправный принимающий MTSE, и запускается таймер T104. Если в ответ на сообщение MultiplexEntrySend получено сообщение MultiplexEntrySendAck, то таймер T104 останавливается, и пользователь с помощью примитива TRANSFER.confirm извещается о том, что запрос о передаче элемента таблицы мультиплексирования прошел успешно. Однако если в ответ на сообщение MultiplexEntrySend получено сообщение MultiplexEntrySendReject, то таймер T104 останавливается, и пользователь с помощью примитива REJECT.indication извещается о том, что пользователь равноправного MTSE отклонил элемент таблицы мультиплексирования.

Если срок действия таймера T104 истек, то пользователь передающего MTSE извещается об этом с помощью примитива REJECT.indication, и передается сообщение MultiplexEntrySendRelease.

Принимаются только те сообщения MultiplexEntrySendAck и MultiplexEntrySendReject, которые передаются в ответ на самое последнее сообщение MultiplexEntrySend. Ответы на более ранние сообщения MultiplexEntrySend игнорируются.

Процедура запроса о передаче нового элемента таблицы мультиплексирования может быть инициирована при отправке пользователем в передающем MTSE примитива TRANSFER.request до того, как было получено сообщение MultiplexEntrySendAck или MultiplexEntrySendReject.

С.7.1.2 Общее описание протокола – принимающий MTSE

Если сообщение MultiplexEntrySend получено в принимающем MTSE, пользователь с помощью примитива TRANSFER.indication извещается о запросе на передачу элемента таблицы мультиплексирования. Пользователь принимающего MTSE подтверждает получение элемента таблицы мультиплексирования, формируя примитив TRANSFER.response, и на равноправный передающий MTSE передается сообщение MultiplexEntrySendAck. Пользователь принимающего MTSE сообщает об отклонении элемента таблицы мультиплексирования, формируя примитив REJECT.request, и на равноправный передающий MTSE отсылается сообщение MultiplexEntrySendReject.

Новое сообщение MultiplexEntrySend может быть получено до того, как пользователь принимающего MTSE ответит на более раннее сообщение MultiplexEntrySend. Пользователь принимающего MTSE информируется об этом с помощью примитива REJECT.indication, за которым следует примитив TRANSFER.indication, и пользователь принимающего MTSE отвечает на новый элемент таблицы мультиплексирования.

Если сообщение MultiplexEntrySendRelease получено до того, как пользователь принимающего MTSE ответил на более раннее сообщение MultiplexEntrySend, то пользователь принимающего MTSE информируется об этом с помощью REJECT.indication, и более ранний элемент таблицы мультиплексирования отбрасывается.

С.7.2 Связь между MTSE и пользователем MTSE

С.7.2.1 Обмен примитивами между MTSE и пользователем MTSE

Взаимодействие между MTSE и пользователем MTSE осуществляется с помощью примитивов, показанных в таблице С.25.

Таблица С.25/Н.245 – Примитивы и параметры

Универсальное название	Тип			
	запрос	индикация	ответ	подтверждение
TRANSFER	MUX-DESCRIPTOR	MUX-DESCRIPTOR	– (Примечание 1)	–
REJECT	CAUSE	SOURCE CAUSE	не определен (Примечание 2)	не определен

ПРИМЕЧАНИЕ 1. – "-" означает отсутствие параметров.
ПРИМЕЧАНИЕ 2. – "не определен" означает, что этот примитив не определен.

С.7.2.2 Определение примитивов

Ниже приведено определение указанных примитивов:

- Примитивы TRANSFER используются для передачи элементов таблицы мультиплексирования.
- Примитивы REJECT используются для отклонения элементов таблицы мультиплексирования и для прекращения передачи элемента таблицы мультиплексирования.

С.7.2.3 Определение параметров

Ниже дано определение параметров примитивов, приведенных в таблице С.25:

- Параметр MUX-DESCRIPTOR – это элемент таблицы мультиплексирования. Этот параметр отображается в поле MultiplexEntryDescriptor сообщения multiplexEntrySend, он прозрачно передается от пользователя MTSE в передающем MTSE пользователю MTSE в принимающем MTSE. К одному примитиву TRANSFER может относиться несколько дескрипторов MUX-DESCRIPTOR.
- Параметр SOURCE показывает источник примитива REJECT.indication. Параметр SOURCE принимает значение "USER" ("пользователь") или "PROTOCOL" ("протокол"). Последний случай может возникать в результате истечения срока действия таймера.
- Параметр CAUSE показывает причину отклонения элемента таблицы мультиплексирования. Параметр CAUSE не присутствует, если параметр SOURCE имеет значение "PROTOCOL".

С.7.2.4 Состояния MTSE

Описанные ниже состояния задают разрешенную последовательность обмена примитивами между MTSE и пользователем MTSE. Эти состояния определены отдельно для каждого передающего MTSE и принимающего MTSE. Передающий MTSE имеет следующие состояния:

Состояние 0: IDLE

Не происходит никакой передачи MTSE. Элемент таблицы мультиплексирования может использоваться отправителем.

Состояние 1: AWAITING RESPONSE

Пользователь MTSE запросил передачу элемента таблицы мультиплексирования, и ожидается ответ от равноправного MTSE. Данный элемент таблицы мультиплексирования не должен использоваться отправителем.

Принимающий MTSE имеет следующие состояния:

Состояние 0: IDLE

Не происходит никакой передачи MTSE. Элемент таблицы мультиплексирования может использоваться отправителем.

Состояние 1: AWAITING RESPONSE

Равноправный MTSE отправил элемент таблицы мультиплексирования, и ожидается ответ от пользователя MTSE. Данный элемент таблицы мультиплексирования не может использоваться отправителем.

С.7.2.5 Диаграмма переходов между состояниями

Здесь определяется разрешенная последовательность обмена примитивами между MTSE и пользователем MTSE. Разрешенные последовательности определены отдельно для каждого передающего MTSE и принимающего MTSE, они показаны на рисунках С.25 и С.26, соответственно.

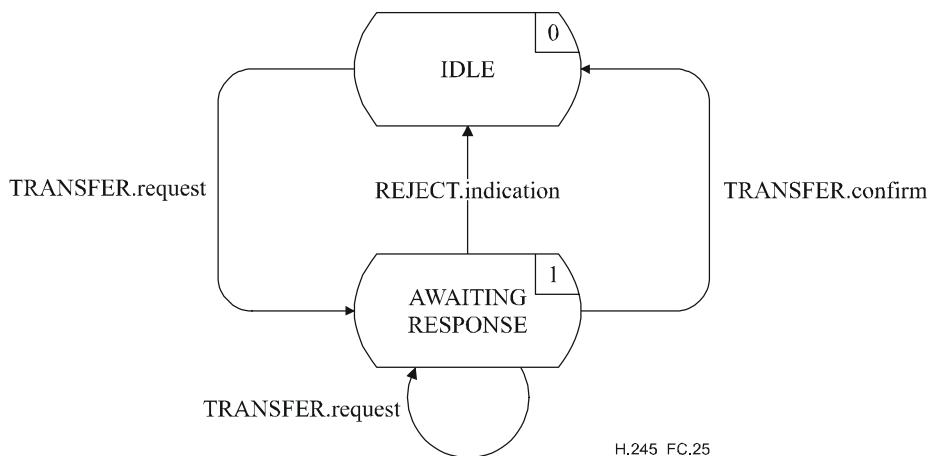


Рисунок С.25/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в передающем MTSE

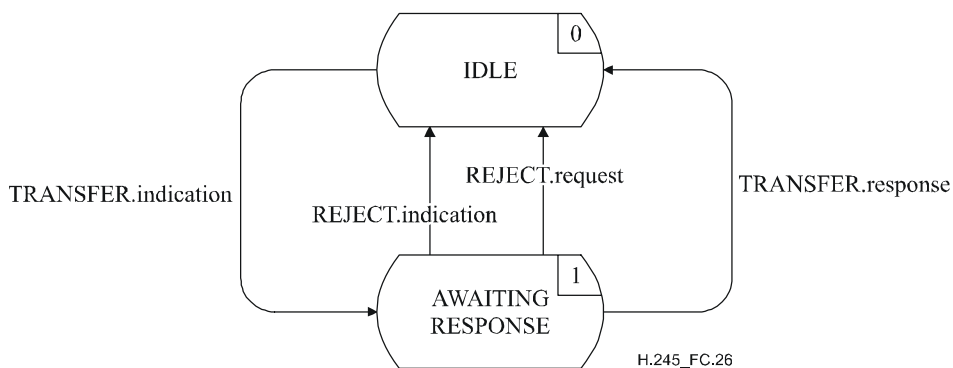


Рисунок С.26/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в принимающем MTSE

С.7.3 Связь между равноправными MTSE

С.7.3.1 Сообщения

В таблице С.26 приведены определенные в Приложении А сообщения и поля MTSE, относящиеся к протоколу MTSE.

Таблица С.26/Н.245 – Названия сообщений и полей MTSE

Функция	Сообщение	Направление	Поле
transfer	MultiplexEntrySend	O → I (Примечание)	sequenceNumber multiplexEntryDescriptors.multiplexTableEntryNumber multiplexEntryDescriptors.elementList
	MultiplexEntrySendAck	O ← I	sequenceNumber multiplexTableEntryNumber
reject	MultiplexEntrySendReject	O ← I	sequenceNumber multiplexTableEntryNumber rejectionDescriptions.cause
reset	MultiplexEntrySendRelease	O → I	multiplexTableEntryNumber
ПРИМЕЧАНИЕ 1. – Направление: O – исходящее, I – входящее.			

С.7.3.2 Переменные состояния MTSE

Для передающего MTSE определены следующие переменные состояния:

out_ENUM

Эта переменная состояния устанавливает различие между передающими MTSE. Она инициализируется при инициализации передающего MTSE. Значение out_ENUM используется в качестве значения поля multiplexTableEntryNumber сообщений MTSE, посланных передающим MTSE. Для сообщений MTSE, полученных в передающем MTSE, значение поля multiplexTableEntryNumber сообщения идентично значению out_ENUM.

out_SQ

Эта переменная состояния используется для указания сообщения MultiplexEntrySend, переданного самым последним. Она наращивается на единицу и отображается в поле sequenceNumber сообщения MultiplexEntrySend до передачи сообщения MultiplexEntrySend. К out_SQ применяется арифметика по модулю 256.

Для принимающего MTSE определены следующие переменные состояния:

in_ENUM

Эта переменная состояния устанавливает различие между принимающими MTSE. Она инициализируется при инициализации принимающего MTSE. Значение in_ENUM используется для установки значения поля multiplexTableEntryNumber сообщений MTSE, которые посланы принимающим MTSE. Для сообщений MTSE, полученных принимающим MTSE, значение поля multiplexTableEntryNumber сообщения идентично значению in_ENUM.

in_SQ

Эта переменная состояния используется для хранения значения поля sequenceNumber сообщения MultiplexEntrySend, полученного самым последним. Поля sequenceNumber сообщений MultiplexEntrySendAck и MultiplexEntrySendReject устанавливаются равными значению in_SQ до отправки этих сообщений равноправному MTSE.

С.7.3.3 Таймеры MTSE

Для передающего MTSE определен следующий таймер:

T104

Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимально возможный период без получения сообщения MultiplexEntrySendAck и MultiplexEntrySendReject.

С.7.4 Процедуры MTSE

С.7.4.1 Введение

На рисунке С.27 приведены все примитивы и их параметры, а также сообщения и соответствующие поля для каждого передающего и принимающего MTSE.

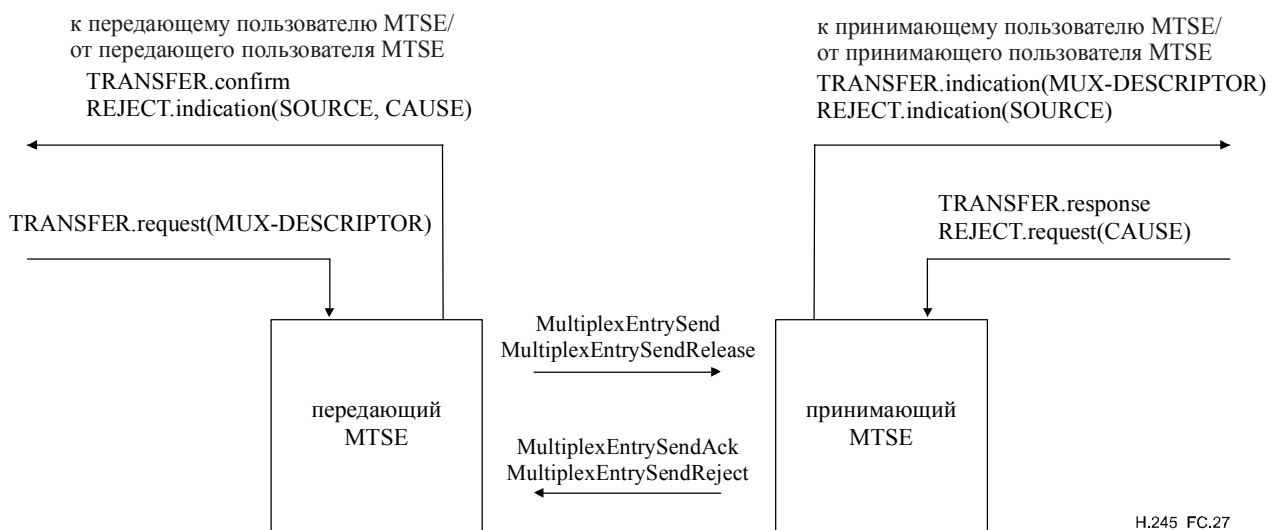


Рисунок С.27/Н.245 – Примитивы и сообщения в объекте сигнализации таблицы мультиплексирования

С.7.4.2 Принимаемые по умолчанию значения параметров примитивов

Параметры примитивов индикации и квитирования принимают значения, приведенные в таблице С.27, если в диаграммах SDL в явном виде не задано иное.

Таблица С.27/Н.245 – Принимаемые по умолчанию значения параметров примитивов

Примитив	Параметр	Значение по умолчанию
TRANSFER.indication	MUX-DESCRIPTOR	MultiplexEntrySend.multiplexEntryDescriptors.elementList
REJECT.indication	SOURCE	USER
	CAUSE	null

С.7.4.3 Принимаемые по умолчанию значения полей сообщений

Поля сообщения принимают значения, приведенные в Таблице С.28, если в диаграммах SDL в явном виде не задано иное.

Таблица С.28/Н.245 – Принимаемые по умолчанию значения полей сообщений

Сообщение	Поле	Значение по умолчанию (Примечание)
MultiplexEntrySend	sequenceNumber multiplexEntryDescriptors.multiplexTableEntryNumber multiplexEntryDescriptors.elementList	out_SQ out_ENUM TRANSFER.request(MUX-DESCRIPTOR)
MultiplexEntrySendAck	sequenceNumber multiplexTableEntryNumber	in_SQ in_ENUM
MultiplexEntrySendReject	sequenceNumber rejectionDescriptions.multiplexTableEntryNumber rejectionDescriptions.cause	in_SQ in_ENUM REJECT.request(CAUSE)
MultiplexEntrySendRelease	multiplexTableEntryNumber	out_ENUM
ПРИМЕЧАНИЕ. – Поле сообщения не должно кодироваться, если соответствующий параметр примитива равен null (нулю), т. е. он не присутствует.		

С.7.4.4 Диаграммы языка описания структур (SDL)

Процедуры для передающего MTSE и принимающего MTSE представлены в форме SDL на рисунках С.28 и С.29, соответственно.

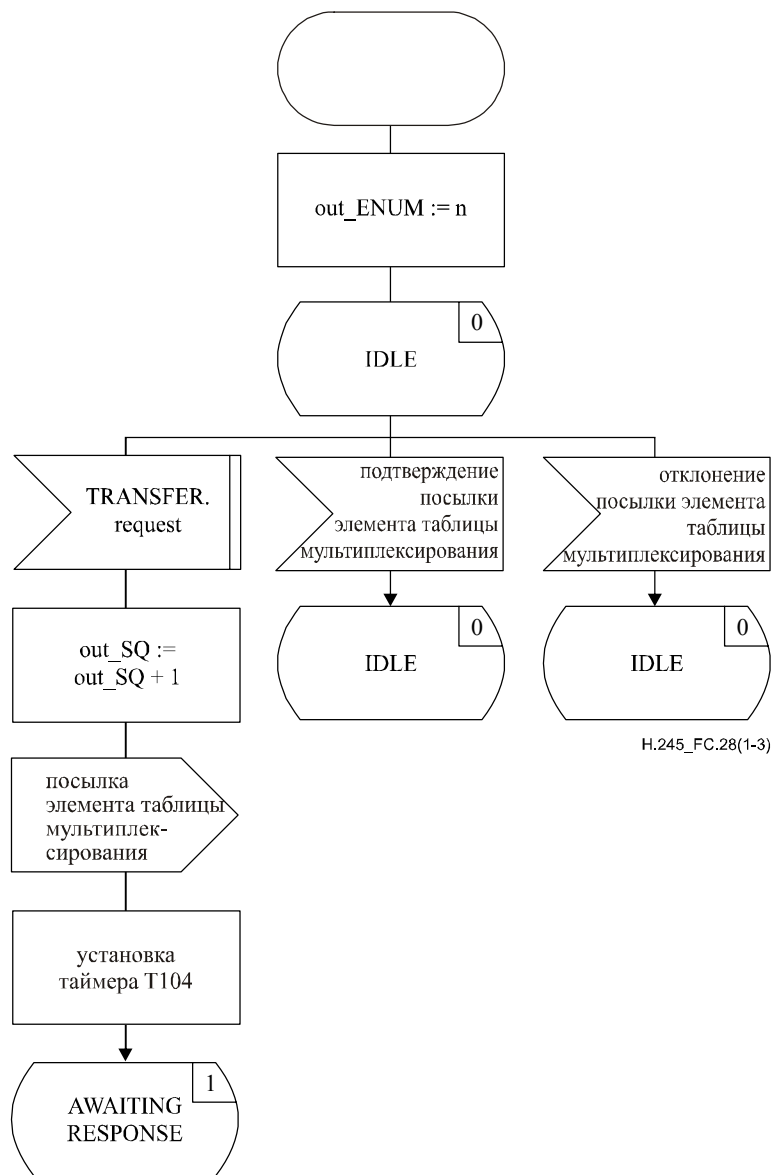
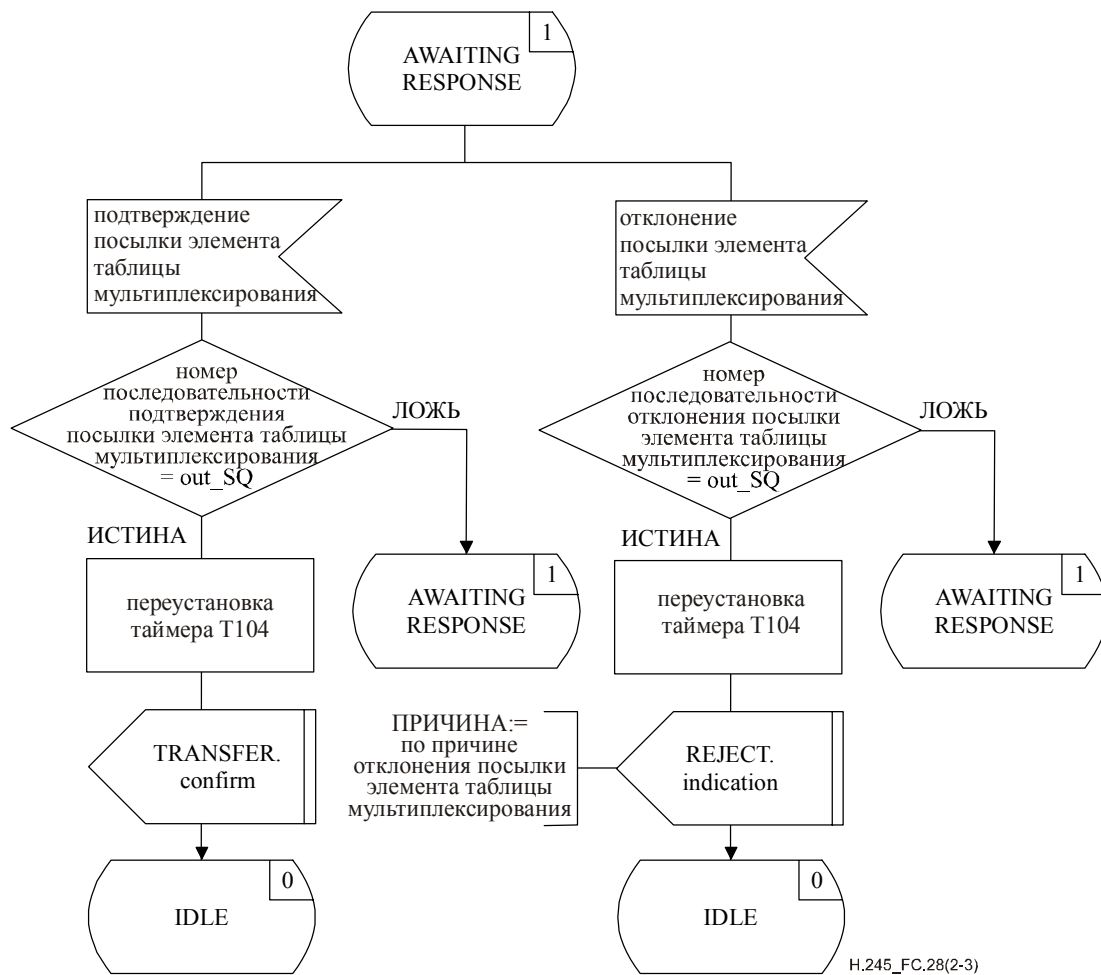


Рисунок С.28/Н.245 – Диаграммы SDL для передающего MTSE (лист 1 из 3)



H.245_FC.28(2-3)

Рисунок С.28/Н.245 – Диаграммы SDL для передающего MTSE (лист 2 из 3)

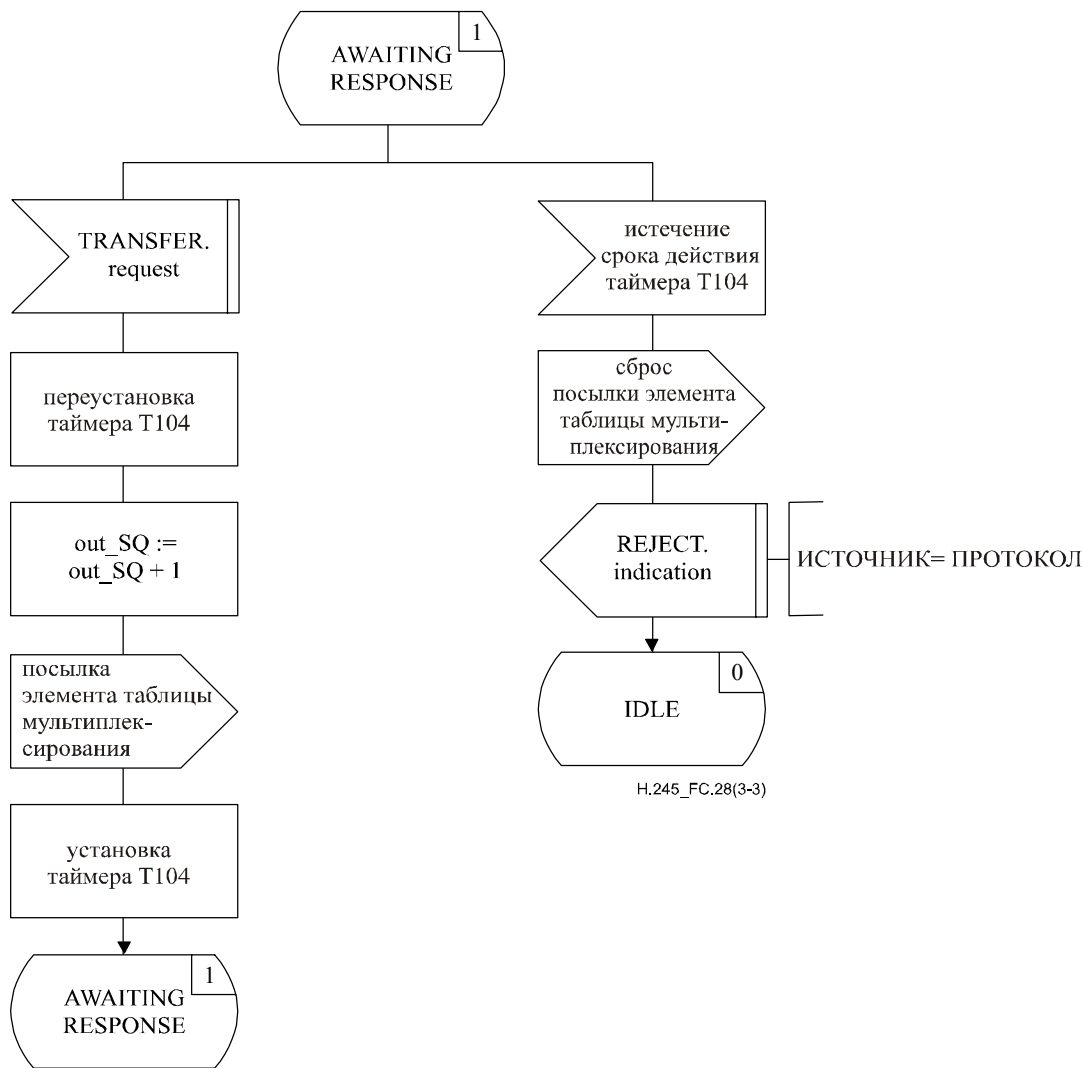


Рисунок С.28/Н.245 – Диаграммы SDL для передающего MTSE (лист 3 из 3)

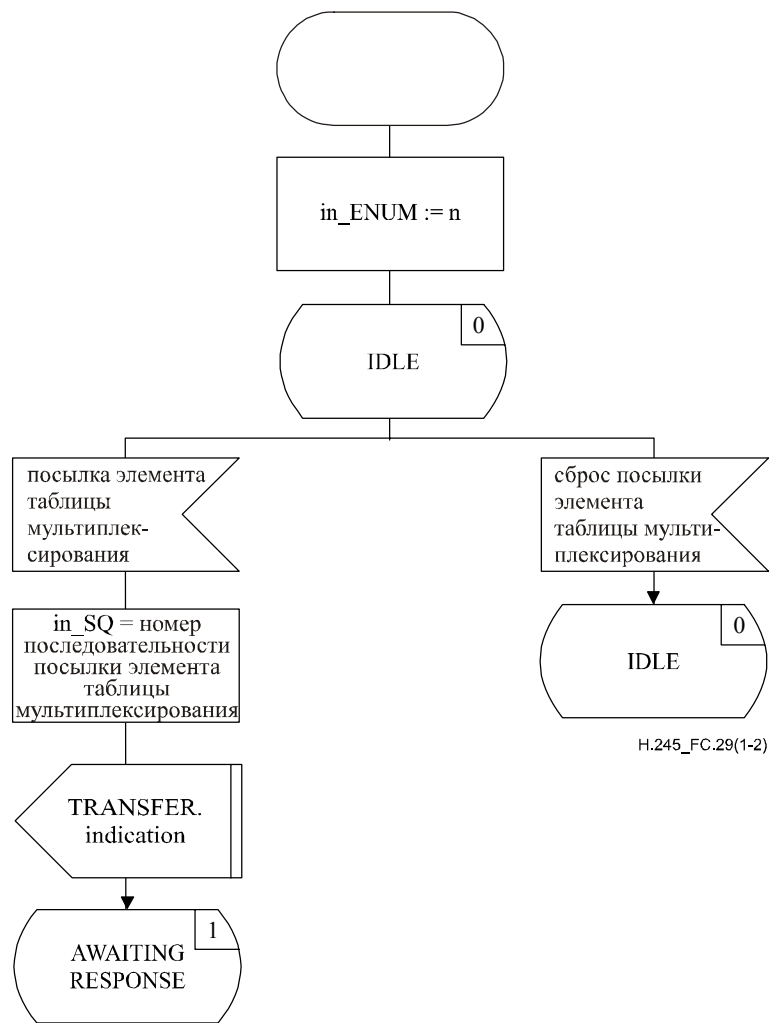


Рисунок С.29/Н.245 – Диаграммы SDL для принимающего MTSE (лист 1 из 2)

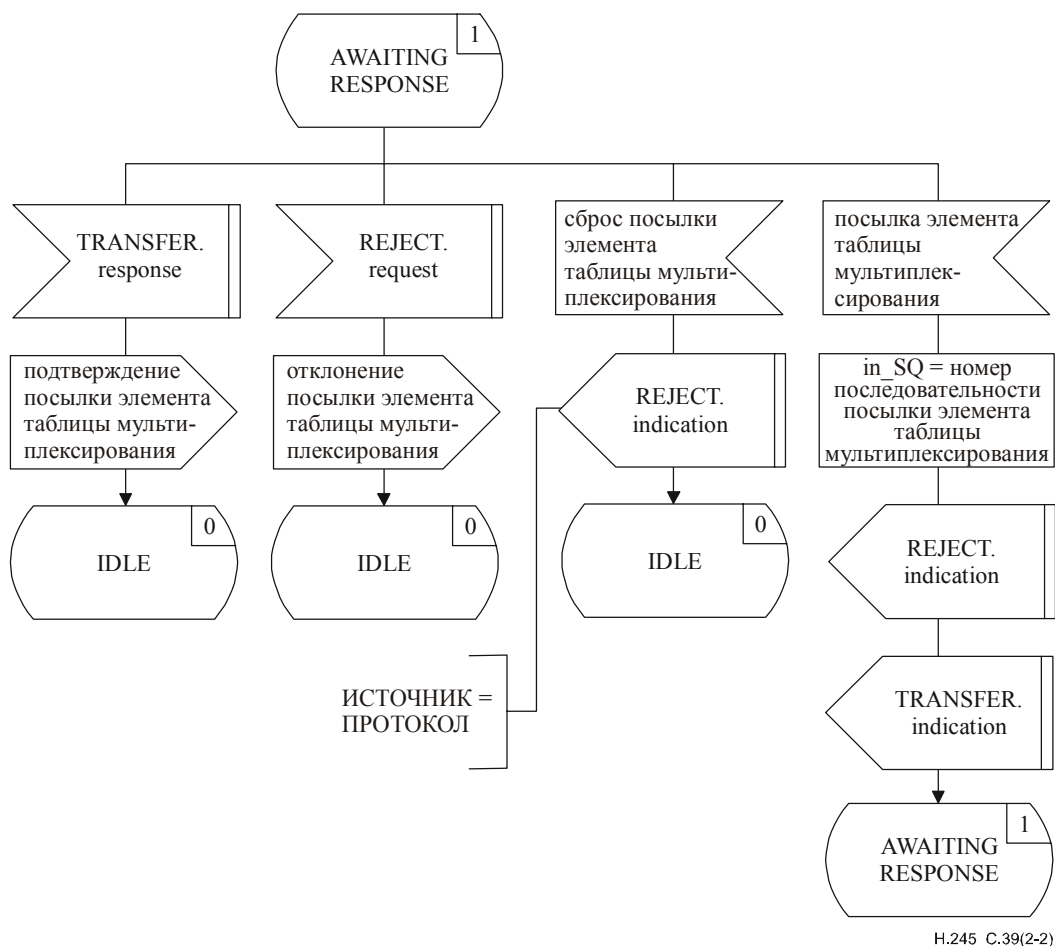


Рисунок С.29/Н.245 – Диаграммы SDL для принимающего MTSE (лист 2 из 2)

С.8 Процедуры запроса элемента таблицы мультиплексирования

С.8.1 Введение

Эти процедуры используются терминалом для запроса о повторной передаче одного или более MultiplexEntryDescriptors. Эти процедуры называются далее процедурами объекта сигнализации запроса элемента таблицы мультиплексирования (RMESE). Процедуры определяются с помощью примитивов и состояний на интерфейсе между RMESE и пользователем RMESE. Информация протокола передается равноправному RMESE с помощью соответствующих сообщений, которые определены в Приложении А. Существует передающий RMESE и принимающий RMESE. Для каждого элемента таблицы мультиплексирования имеется один экземпляр RMESE.

Терминал, который отвечает на такой ответ положительно, то есть формированием примитива SEND.response, должен инициализировать процедуры таблицы мультиплексирования, чтобы как можно скорее передать элемент таблицы мультиплексирования.

В приведенном ниже тексте содержится общее описание действия данного протокола. В случае каких-либо расхождений между ним и официальной спецификацией протокола преимущественную силу имеет официальная спецификация.

ПРИМЕЧАНИЕ. – Этот протокол был определен таким образом, чтобы для каждого элемента таблицы мультиплексирования существовал независимый RMESE, и синтаксис определен так, чтобы в одном сообщении могла передаваться информация, относящаяся к одному или более элементам таблицы мультиплексирования. Способ формирования сообщений зависит от конкретной реализации, например, терминал может отвечать на сообщение RequestMultiplexEntry, которое запрашивает три элемента, посредством посылки одного, двух или трех ответных сообщений.

С.8.1.1 Общее описание протокола – передающий RMESE

Процедура запроса элемента таблицы мультиплексирования инициируется передачей пользователем на передающем RMESE примитива SEND.request. Сообщение RequestMultiplexEntry передается равноправному принимающему RMESE, и запускается таймер T107. Если в ответ на сообщение RequestMultiplexEntry получено сообщение RequestMultiplexEntryAck, то таймер T107 останавливается, и пользователь с помощью примитива SEND.confirm извещается о том, что процедура запроса элемента таблицы мультиплексирования была выполнена успешно. Однако если в ответ на сообщение RequestMultiplexEntry получено сообщение RequestMultiplexEntryReject, то таймер T107 останавливается, и пользователь с помощью примитива REJECT.indication извещается о том, что пользователь равноправного RMESE отказал в посылке элемента таблицы мультиплексирования.

Если окончился срок действия таймера T107, то пользователь передающего RMESE информируется примитивом REJECT.indication, и передается сообщение RequestMultiplexEntryRelease.

С.8.1.2 Общее описание протокола – принимающий RMESE

Если принимающий RMESE получает сообщение RequestMultiplexEntry, пользователь информируется о получении запроса элемента таблицы мультиплексирования с помощью примитива SEND.indication. Пользователь принимающего RMESE подтверждает получение запроса элемента таблицы мультиплексирования путем формирования примитива SEND.response, и равноправному передающему RMESE передается сообщение RequestMultiplexEntryAck. Пользователь принимающего RMESE сообщает об отклонении запроса элемента таблицы мультиплексирования с помощью примитива REJECT.request, и равноправному передающему RMESE передается сообщение RequestMultiplexEntryReject.

С.8.2 Связь между RMESE и пользователем RMESE

С.8.2.1 Обмен примитивами между RMESE и пользователем RMESE

Связь между RMESE и пользователем RMESE осуществляется с помощью примитивов, приведенных в таблице С.29.

Таблица С.29/Н.245 – Примитивы и параметры

Универсальное название	Тип			
	запрос	индикация	ответ	подтверждение
SEND	– (Примечание 1)	–	–	–
REJECT	CAUSE	SOURCE CAUSE	не определен (Примечание 2)	не определен

ПРИМЕЧАНИЕ 1. – "-" означает отсутствие параметров.
ПРИМЕЧАНИЕ 2. – "не определен" означает, что этот примитив не определен.

С.8.2.2 Определение примитивов

Ниже дано определение указанных примитивов:

- Примитивы SEND используются для запроса о передаче элемента таблицы мультиплексирования.
- Примитивы REJECT используются для отклонения запроса о передаче элемента таблицы мультиплексирования.

С.8.2.3 Определение параметров

Ниже дано определение параметров примитивов, приведенных в таблице С.29:

- Параметр SOURCE показывает источник примитива REJECT.indication. Параметр SOURCE принимает значение "USER" ("пользователь") или "PROTOCOL" ("протокол"). Последнее значение может быть следствием истечения срока действия таймера.
- Параметр CAUSE показывает причину, по которой была отклонена посылка элемента таблицы мультиплексирования. Параметр CAUSE не присутствует, когда параметр SOURCE имеет значение "PROTOCOL".

С.8.2.4 Состояния RMESE

Описанные ниже состояния задают разрешенную последовательность обмена примитивами между RMESE и пользователем RMESE.

Передающий RMESE имеет следующие состояния:

Состояние 0: IDLE

RMESE находится в состоянии ожидания.

Состояние 1: AWAITING RESPONSE

RMESE ждет ответа от удаленного RMESE.

Принимающий RMESE имеет следующие состояния:

Состояние 0: IDLE

RMESE находится в состоянии ожидания.

Состояние 1: AWAITING RESPONSE

RMESE ожидает ответа от пользователя RMESE.

С.8.2.5 Диаграмма переходов между состояниями

Здесь определяется разрешенная последовательность обмена примитивами между RMESE и пользователем RMESE. Разрешенные последовательности обмена примитивами заданы отдельно для каждого передающего RMESE и принимающего RMESE, как показано на рисунках С.30 и С.31 соответственно.

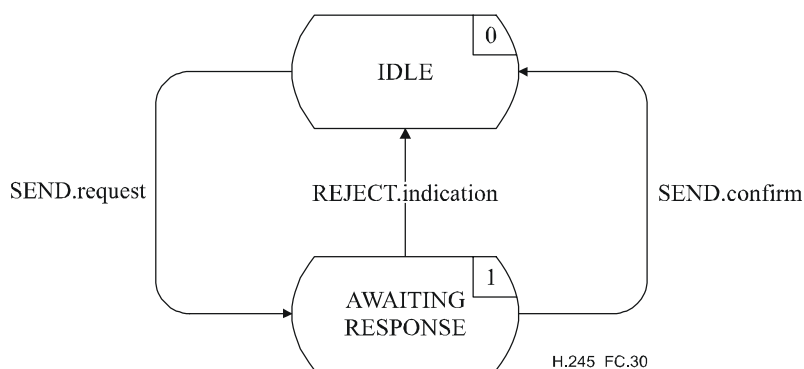


Рисунок С.30/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в передающем RMESE

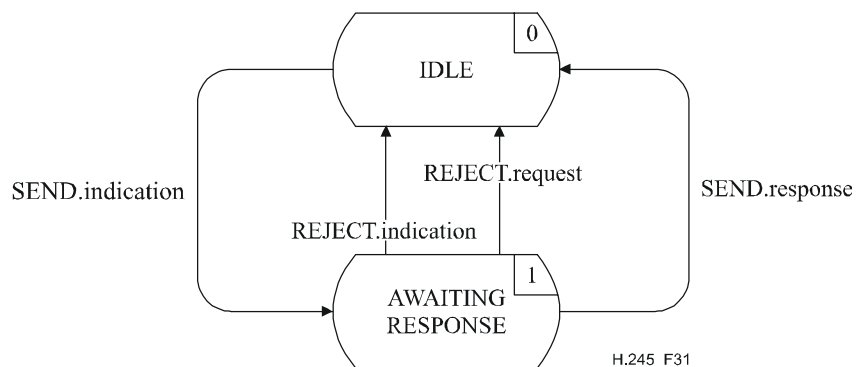


Рисунок С.31/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в принимающем RMESE

С.8.3 Связь между равноправными RMESE

С.8.3.1 Сообщения

В таблице С.30 приведены определенные в Приложении А сообщения и поля RMESE, относящиеся к протоколу RMESE.

Таблица С.30/Н.245 – Названия и поля сообщений RMESE

Функция	Сообщение	Направление	Поле
transfer	RequestMultiplexEntry	O → I (Примечание)	multiplexTableEntryNumber
	RequestMultiplexEntryAck	O ← I	multiplexTableEntryNumber
	RequestMultiplexEntryReject	O ← I	multiplexTableEntryNumber rejectionDescriptions.cause
reset	RequestMultiplexEntryRelease	O → I	

ПРИМЕЧАНИЕ. – Направление: O – исходящее, I – входящее.

С.8.3.2 Переменные состояний RMESE

Для передающего RMESE определена следующая переменная состояния:

out_ENUM

Эта переменная состояния устанавливает различие между передающими RMESE. Она инициализируется при инициализации передающего RMESE. Значение out_ENUM используется для установки значения поля multiplexTableEntryNumber сообщений RMESE, посланных передающим RMESE. Для сообщений RMESE, полученных в передающем RMESE, значение поля сообщения multiplexTableEntryNumber идентично значению out_ENUM.

Следующая переменная состояния определена в принимающем RMESE:

in_ENUM

Эта переменная состояния устанавливает различие между принимающими RMESE. Она инициализируется при инициализации принимающего RMESE. Значение in_ENUM используется для установки значения поля multiplexTableEntryNumber сообщений RMESE, переданных из принимающего RMESE. Для сообщений RMESE, полученных на принимающем RMESE, значение поля multiplexTableEntryNumber сообщения идентично значению in_ENUM.

С.8.3.3 Таймеры RMESE

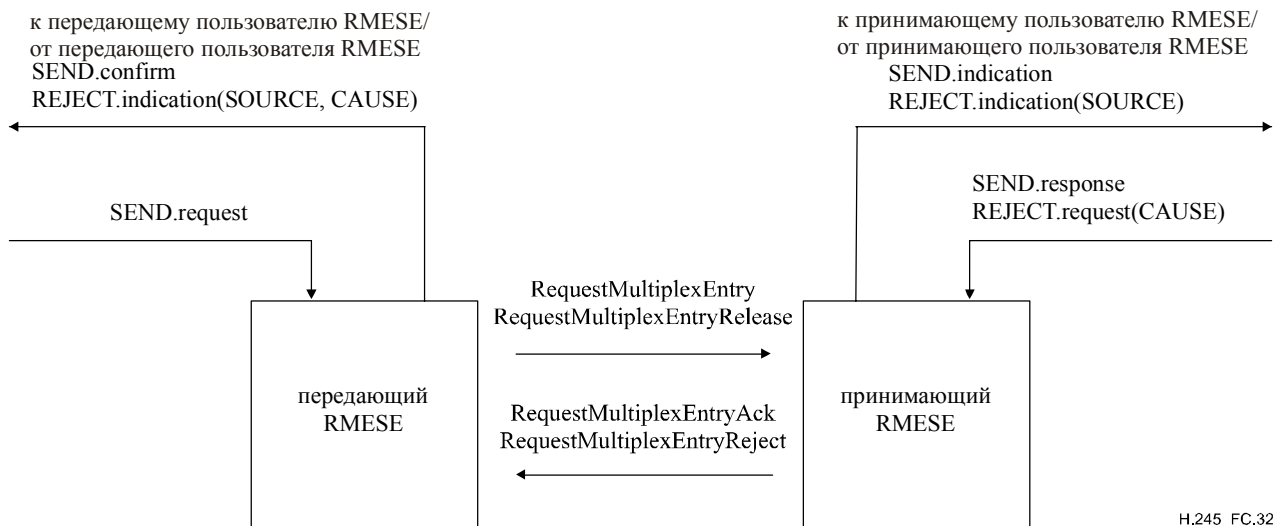
Для передающего RMESE определен приведенный ниже таймер:

T107

Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщения RequestMultiplexEntryAck или RequestMultiplexEntryReject.

С.8.4 Процедуры RMESE

На рисунке С.32 приведены все примитивы RMESE и их параметры, а также сообщения для каждого передающего и принимающего RMESE.



H.245_FC.32

Рисунок С.32/Н.245 – Примитивы и сообщения в запросе объекта сигнализации запроса элемента мультиплексирования

С.8.4.1 Принимаемые по умолчанию значения параметров примитивов

Параметры примитивов индикации и подтверждения принимают значения, показанные в таблице С.31, если в диаграммах SDL в явном виде не задано иное.

Таблица С.31/Н.245 – Принимаемые по умолчанию значения параметров примитивов

Примитив	Параметр	Значение по умолчанию
REJECT.indication	SOURCE CAUSE	USER null

С.8.4.2 Принимаемые по умолчанию значения полей сообщений

Поля сообщений принимают значения, показанные в таблице С.32, если в диаграммах SDL в явном виде не задано иное.

Таблица С.32/Н.245 – Принимаемые по умолчанию значения полей сообщений

Сообщение	Поле	Значение по умолчанию
RequestMultiplexEntry	multiplexTableEntryNumber	out_ENUM
RequestMultiplexEntryAck	multiplexTableEntryNumber	in_ENUM
RequestMultiplexEntryReject	multiplexTableEntryNumber Cause	in_ENUM REJECT.request(CAUSE)
RequestMultiplexEntryRelease	multiplexTableEntryNumber	out_ENUM

С.8.4.3 Диаграммы языка описания структур (SDL)

Процедуры передачи сообщений передающего RMESE и принимающего RMESE представлены в форме SDL на рисунках С.33 и С.34, соответственно.

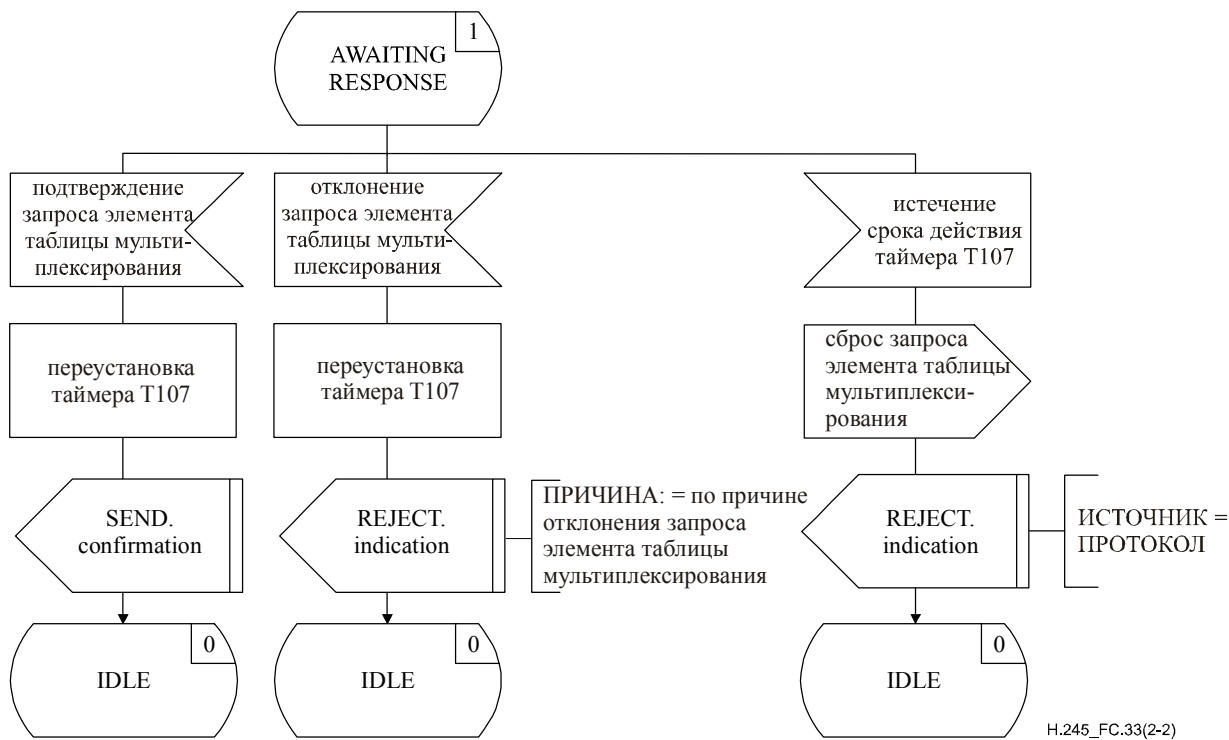


Рисунок С.33/Н.245 – Диаграмма SDL для передающего RMESE (лист 2 из 2)

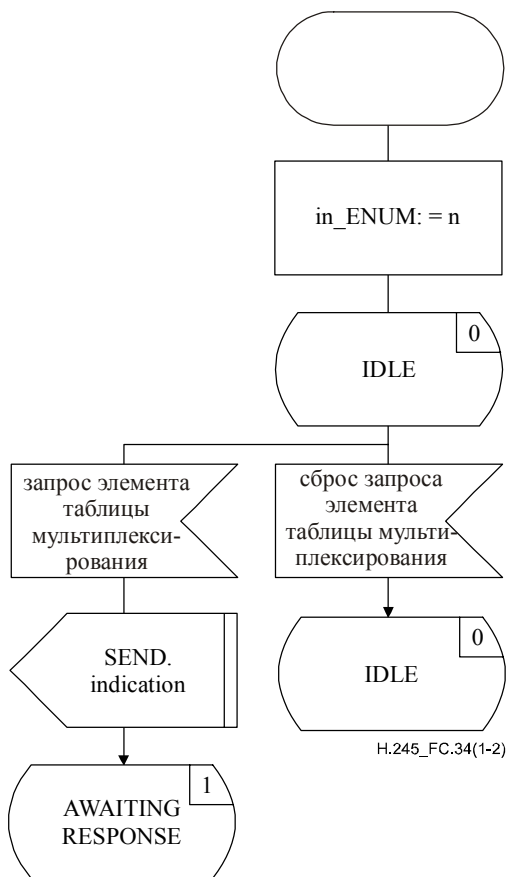


Рисунок С.34/Н.245 – Диаграмма SDL для принимающего RMESE (лист 1 из 2)

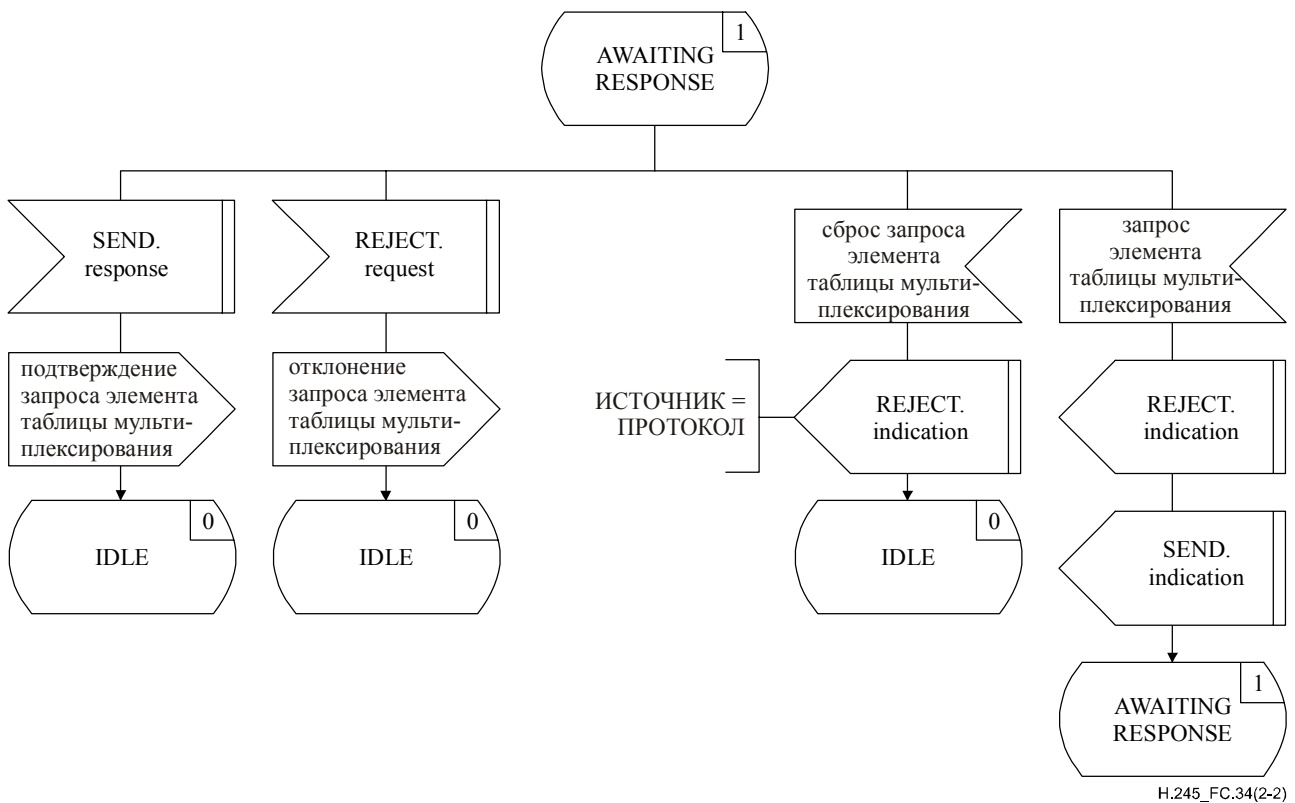


Рисунок С.34/Н.245 – Диаграмма SDL для принимающего RMESE (лист 2 из 2)

С.9 Процедуры запроса режима

С.9.1 Введение

Описанные ниже процедуры позволяют терминалу посылать запрос на удаленный терминал об использовании конкретного режима работы в направлении его передачи. Эти процедуры называются далее Объектом сигнализации запроса режима (MRSE). Процедуры определяются при помощи примитивов и состояний на интерфейсе между MRSE и пользователем MRSE. Информация протокола передается равноправному MRSE с помощью соответствующих сообщений, определенных в Приложении А. Имеются передающий MRSE и принимающий MRSE. На каждом передающем и принимающем конце существует один экземпляр MRSE.

Терминал, который отвечает на подобный запрос положительно, то есть формируя примитив TRANSFER.response, должен инициализировать процедуры сигнализации логического канала для скорейшего установления соответствующего способа передачи.

Если принимаемые с удаленного терминала и действительные на данный момент возможности содержат одну или более возможностей передачи, то терминал может выбрать предпочтительный для него режим передачи на него, выполнив процедуру запроса режима. Терминал, допустимые возможности которого на данный момент содержат одну или более возможностей передачи и который получает такой запрос, должен подчиниться этому запросу.

Запрос режима не должен посылаться на терминал, действительные на данный момент времени возможности которого не содержат возможности передачи, то есть терминал не желает и не должен дистанционно управляться. Однако если такой терминал действительно принимает запрос режима, он может подчиниться этому запросу.

Терминал, который получает команду multipointModeCommand, должен выполнять все полученные запросы режима до тех пор, пока действие этой команды не будет прекращено получением команды cancelMultipointModeCommand. Запрос режима может быть передан на терминал, разрешенные на данный момент возможности которого не содержат никаких возможностей передачи на момент, когда была выдана предыдущая команда multipointModeCommand.

В запросе режима могут указываться уже открытые каналы. Например, если канал для G.723.1 был открыт и терминал желал принимать дополнительный канал G.728, то он должен передать запрос режима, содержащий как канал G.723.1, так и G.728. Если бы запрос канала G.723.1 отсутствовал, то это означало бы, что канал G.723.1 больше не нужен.

Если присутствует параметр `logicalChannelNumber`, то запрос относится только к указанному логическому каналу, который должен быть в открытом состоянии и запрашивать изменение режима соответствующего логического канала на указанный режим.

ПРИМЕЧАНИЕ. – Если не присутствует параметр `logicalChannelNumber`, то описание режима запроса определяет полный режим. Например, если в текущий момент передается видео и получен запрос режима, который не включает никакой спецификации по видео, то этот запрос требует прекращения передачи видео.

В случае, если у нескольких получателей имеется один источник, он может быть не в состоянии отвечать ни на какие принятые сигналы, представляющие собой, например, запросы на передачу в конкретном режиме.

В приведенном ниже тексте содержится общее описание действия протокола MRSE. В случае каких-либо расхождений между ним и официальной спецификацией протокола преимущественную силу имеет официальная спецификация.

С.9.1.1 Общее описание протокола – передающий MRSE

Процедура запроса режима инициализируется, когда пользователь формирует примитив `TRANSFER.request` в передающем MRSE. Сообщение `RequestMode` передается равноправному принимающему MRSE, и запускается таймер T109. Если в ответ на сообщение `RequestMode` получено сообщение `RequestModeAck`, то таймер T109 останавливается, и пользователь с помощью примитива `TRANSFER.confirm` информируется об успешном запросе режима. Однако если в ответ на сообщение `RequestMode` получено сообщение `RequestModeReject`, то таймер T109 останавливается и пользователь с помощью примитива `REJECT.indication` информируется о том, что пользователь равноправного MRSE отказал в получении запроса режима.

Если окончился срок действия таймера T109, то пользователь передающего MRSE информируется об этом примитивом `REJECT.indication`, и передается сообщение `RequestModeRelease`.

Принимаются только сообщения `RequestModeAck` и `RequestModeReject`, которые служат ответом на самое последнее сообщение `RequestMode`. Игнорируются сообщения, переданные в ответ на предыдущие сообщения `RequestMode`.

Пользователь в передающем MRSE может инициировать новую процедуру запроса режима с помощью примитива `TRANSFER.request` до того, как было получено сообщение `RequestModeAck` или `RequestModeReject`.

С.9.1.2 Общее описание протокола – принимающий MRSE

Когда принимающий MRSE получил сообщение `RequestMode`, пользователь информируется о запросе режима с помощью примитива `TRANSFER.indication`. Пользователь принимающего MRSE подтверждает прием запроса режима путем выдачи примитива `TRANSFER.response`, а равноправному передающему MRSE передается сообщение `RequestModeAck`. Пользователь принимающего MRSE сообщает об отклонении запроса режима, выдавая примитив `REJECT.request`, а равноправному передающему MRSE посылается сообщение `RequestModeReject`.

Новое сообщение `RequestMode` может быть получено перед тем, как пользователь принимающего MRSE ответил на более раннее сообщение `RequestMode`. Пользователь принимающего MRSE информируется об этом с помощью примитива `REJECT.indication`, за которым следует примитив `TRANSFER.indication`, и пользователь принимающего MRSE отвечает на новый элемент таблицы мультимплексования.

Если сообщение `RequestModeRelease` получено до того, как пользователь принимающего MRSE ответил на более раннее сообщение `RequestMode`, то пользователь принимающего MRSE информируется об этом примитивом `REJECT.indication`, а более ранний запрос режима отвергается.

С.9.2 Связь между MRSE и пользователем MRSE

С.9.2.1 Обмен примитивами между MRSE и пользователем MRSE

Взаимодействие между MRSE и пользователем MRSE осуществляется с помощью примитивов, приведенных в таблице С.33.

Таблица С.33/Н.245 – Примитивы и параметры

Универсальное название	Тип			
	запрос	индикация	ответ	подтверждение
TRANSFER	MODE-ELEMENT	MODE-ELEMENT	MODE-PREF	MODE-PREF
REJECT	CAUSE	SOURCE CAUSE	не определен (Примечание)	не определен
ПРИМЕЧАНИЕ. – "не определен" означает, что этот примитив не определен.				

С.9.2.2 Определение примитивов

Ниже приведено определение этих примитивов:

- Примитивы TRANSFER используются для передачи запроса режима.
- Примитивы REJECT используются для отклонения запроса режима.

С.9.2.3 Определение параметров

Ниже даны определения параметров примитивов, приведенных в таблице С.33:

- Параметр MODE-ELEMENT определяет элемент режима. Этот параметр отображается в поле requestedModes сообщения RequestMode и прозрачно передается от пользователя передающего MRSE пользователю принимающего MRSE. Этот параметр является обязательным. Может существовать несколько элементов режима MODE-ELEMENTS, относящихся к примитивам TRANSFER.
- Параметр MODE-PREF информирует пользователя о том, будет ли использоваться запрошенный наиболее предпочтительный режим. Этот параметр отображается в поле Response (ответ) сообщения RequestModeAck, он прозрачно передается от пользователя принимающего RMSE пользователю передающего RMSE. Он имеет два значения: "MOST-PREFERRED" ("наиболее предпочтительный") и "LESS-PREFERRED" ("менее предпочтительный").
- Параметр SOURCE показывает источник примитива REJECT.indication. Параметр SOURCE имеет значение "USER" ("пользователь") или "PROTOCOL" ("протокол"). Последнее значение может появиться в результате окончания срока действия таймера.
- Параметр CAUSE показывает причину, по которой был отклонен запрос режима. Параметр CAUSE не присутствует, если параметр SOURCE имеет значение "PROTOCOL".

С.9.2.4 Состояния MRSE

Описанные ниже состояния задают разрешенную последовательность обмена примитивами между MRSE и пользователем MRSE. Передающий MRSE имеет следующие состояния:

Состояние 0: IDLE

MRSE находится в состоянии ожидания.

Состояние 1: AWAITING RESPONSE

MRSE ожидает ответа от удаленного MRSE.

Принимающий MRSE имеет следующие состояния:

Состояние 0: IDLE

MRSE находится в состоянии ожидания.

Состояние 1: AWAITING RESPONSE

MRSE ожидает ответа от пользователя MRSE.

С.9.2.5 Диаграмма переходов между состояниями

Здесь определена разрешенная последовательность обмена примитивами между MRSE и пользователем MRSE. Разрешенные последовательности обмена примитивами определены отдельно для каждого передающего MRSE и принимающего MRSE, как показано на рисунках С.35 и С.36 соответственно.

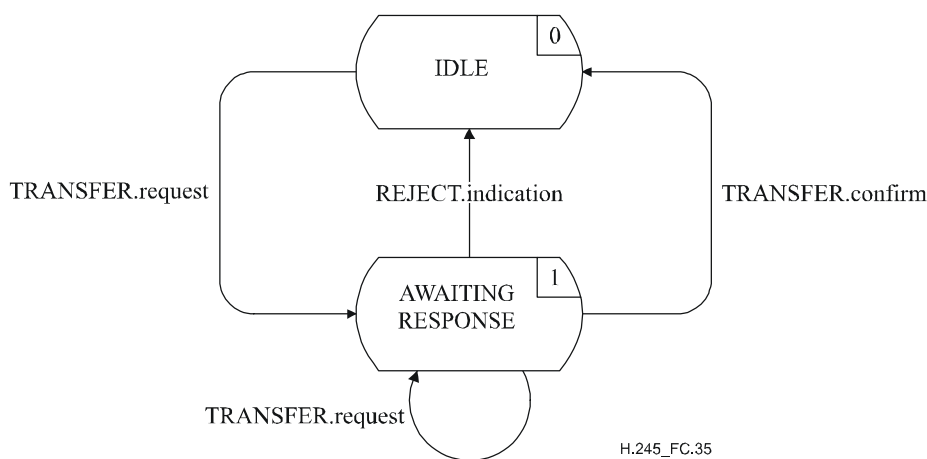


Рисунок С.35/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в передающем MRSE

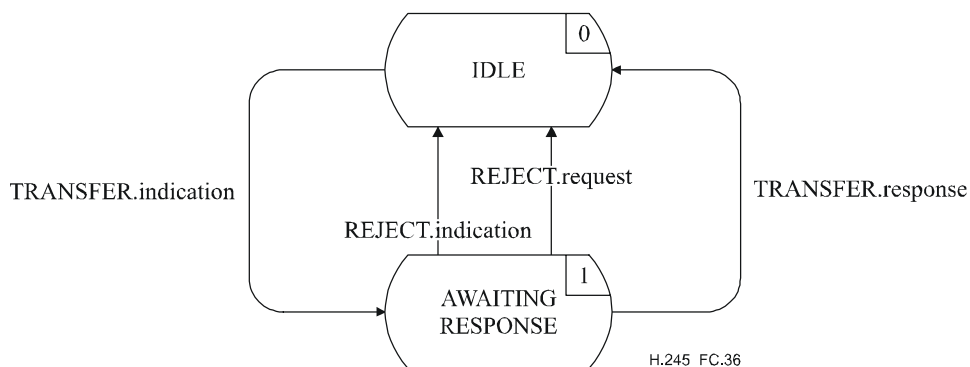


Рисунок С.36/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в принимающем MRSE

С.9.3 Связь между равноправными MRSE

С.9.3.1 Сообщения

В таблице С.34 приведены определенные в Приложении А сообщения и поля MRSE, которые относятся к протоколу MRSE.

Таблица С.34/Н.245 – Названия и поля сообщений MRSE

Функция	Сообщение	Направление	Поле
mode request	RequestMode	O → I (Примечание)	sequenceNumber requestedModes
	RequestModeAck	O ← I	sequenceNumber response
	RequestModeReject	O ← I	sequenceNumber cause
reset	RequestModeRelease	O → I	–
ПРИМЕЧАНИЕ. – Направление: O – исходящее, I – входящее.			

С.9.3.2 Переменные состояния MRSE

Для передающего MRSE определена следующая переменная состояния:

out_SQ

Эта переменная состояния показывает на самое последнее по времени сообщение RequestMode. Она увеличивается на единицу и отображается в поле sequenceNumber сообщения RequestMode перед передачей сообщения RequestMode. Для out_SQ используется арифметика по модулю 256.

Для принимающего MRSE определена следующая переменная состояния:

in_SQ

Эта переменная состояния используется для хранения значения поля sequenceNumber самого последнего по времени полученного сообщения RequestMode. В сообщениях RequestModeAck и RequestModeReject поля sequenceNumber устанавливаются равными значению in_SQ перед отправкой этих сообщений равноправному MRSE.

С.9.3.3 Таймеры MRSE

Для передающего MRSE определен следующий таймер:

T109

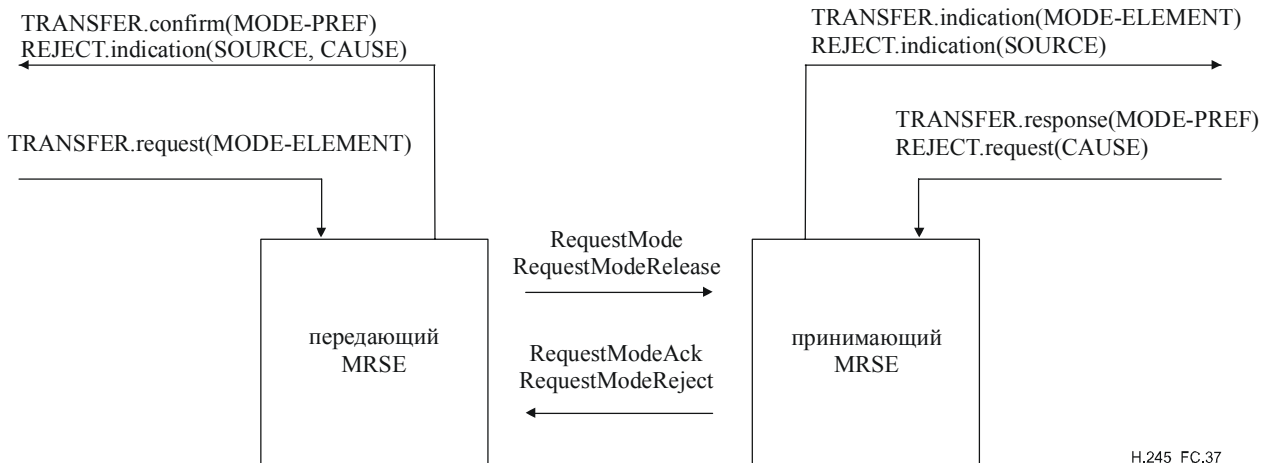
Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщения RequestModeAck или RequestModeReject.

С.9.4 Процедуры MRSE

На рисунке С.37 приведены все примитивы MRSE и их параметры, а также сообщения для каждого передающего и принимающего MRSE.

к передающему пользователю MRSE/
от передающего пользователя MRSE

к принимающему пользователю MRSE/
от принимающего пользователя MRSE



H.245_FC.37

Рисунок С.37/Н.245 – Примитивы и сообщения в объекте сигнализации запроса режима

С.9.4.1 Принимаемые по умолчанию значения переменных примитивов

Параметры примитивов индикации и подтверждения принимают значения, приведенные в таблице С.35, если в диаграммах SDL в явном виде не задано иное.

Таблица С.35/Н.245 – Принимаемые по умолчанию значения переменных примитивов

Примитив	Параметр	Значение по умолчанию
TRANSFER.indication	MODE-ELEMENT	RequestMode.requestedModes
TRANSFER.confirm	MODE-PREF	RequestModeAck.response
REJECT.indication	SOURCE	USER
	CAUSE	null

С.9.4.2 Принимаемые по умолчанию значения полей сообщений

Поля сообщения принимают значения, приведенные в таблице С.36, если в диаграммах SDL в явном виде не задано иное.

Таблица С.36/Н.245 – Принимаемые по умолчанию значения полей сообщений

Сообщение	Поле	Значение по умолчанию
RequestMode	sequenceNumber	out_SQ
	requestedModes	TRANSFER.request(MODE-ELEMENT)
RequestModeAck	sequenceNumber	in_SQ
	response	TRANSFER.response(MODE-PREF)
RequestModeReject	sequenceNumber	in_SQ
	cause	REJECT.request(CAUSE)
RequestModeRelease	—	—

С.9.4.3 Диаграммы языка описания структур (SDL)

Процедуры передачи сообщений передающего MRSE и принимающего MRSE представлены в форме SDL на рисунках С.38 и С.39, соответственно.

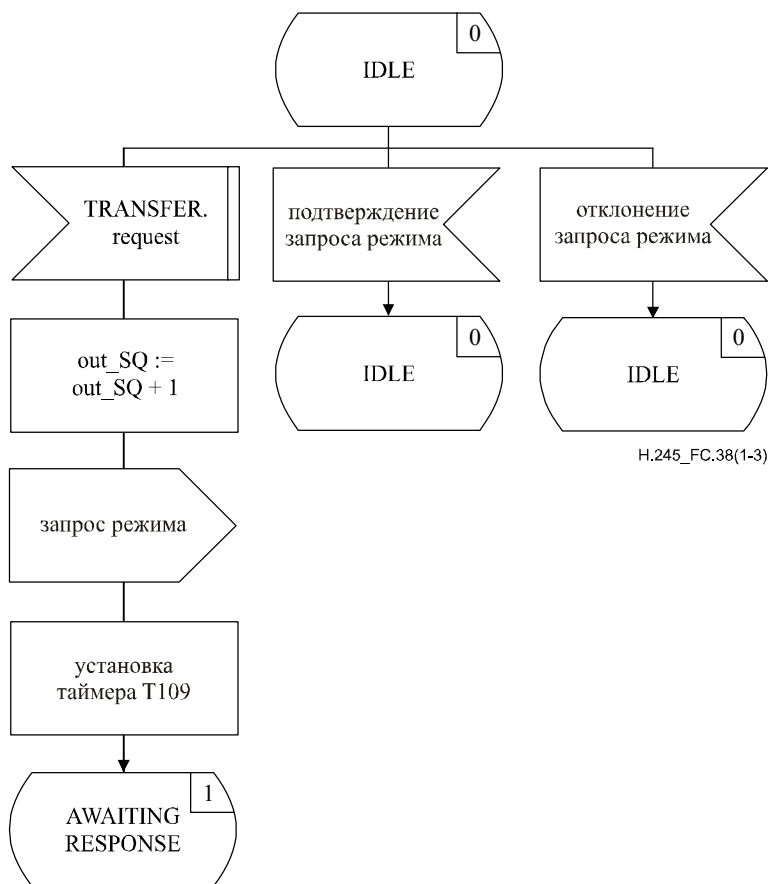
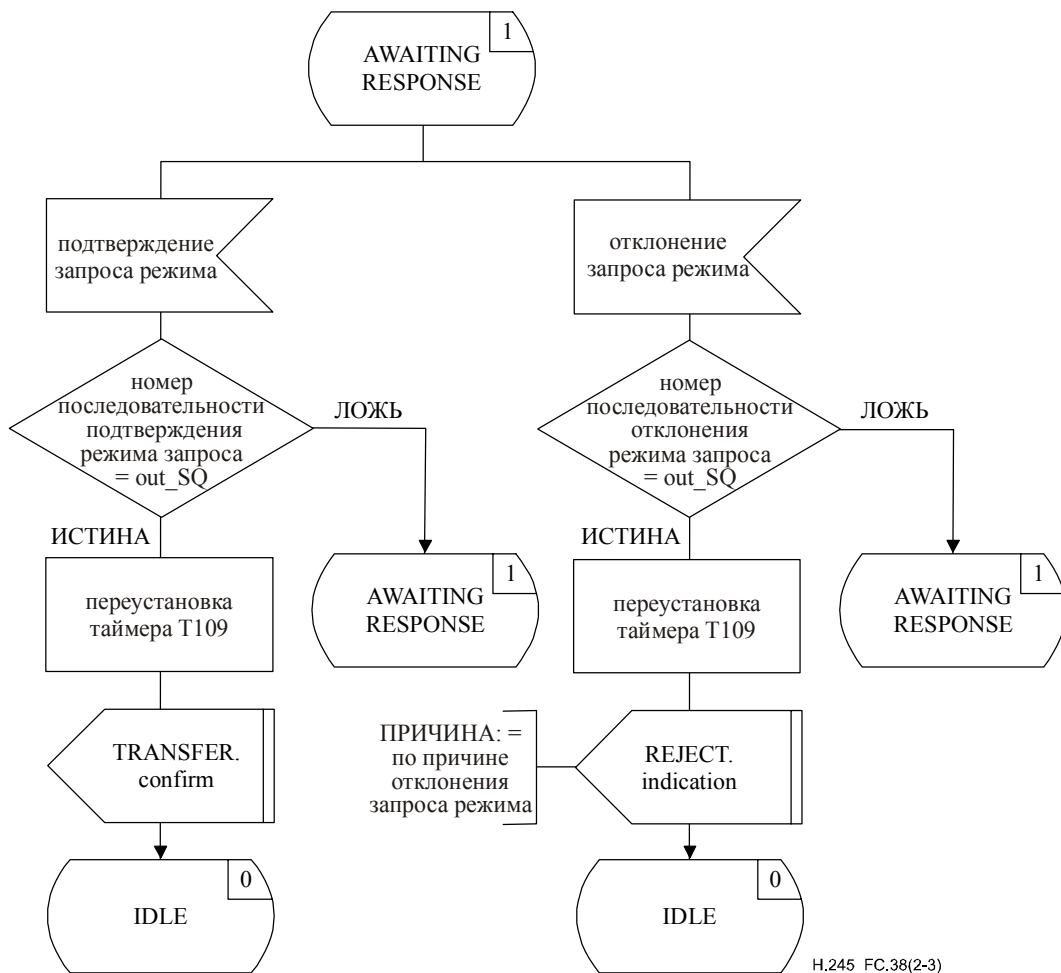


Рисунок С.38/Н.245 – Диаграмма SDL для передающего MRSE (лист 1 из 3)



H.245_FC.38(2-3)

Рисунок С.38/Н.245 – Диаграмма SDL для передающего MRSE (лист 2 из 3)

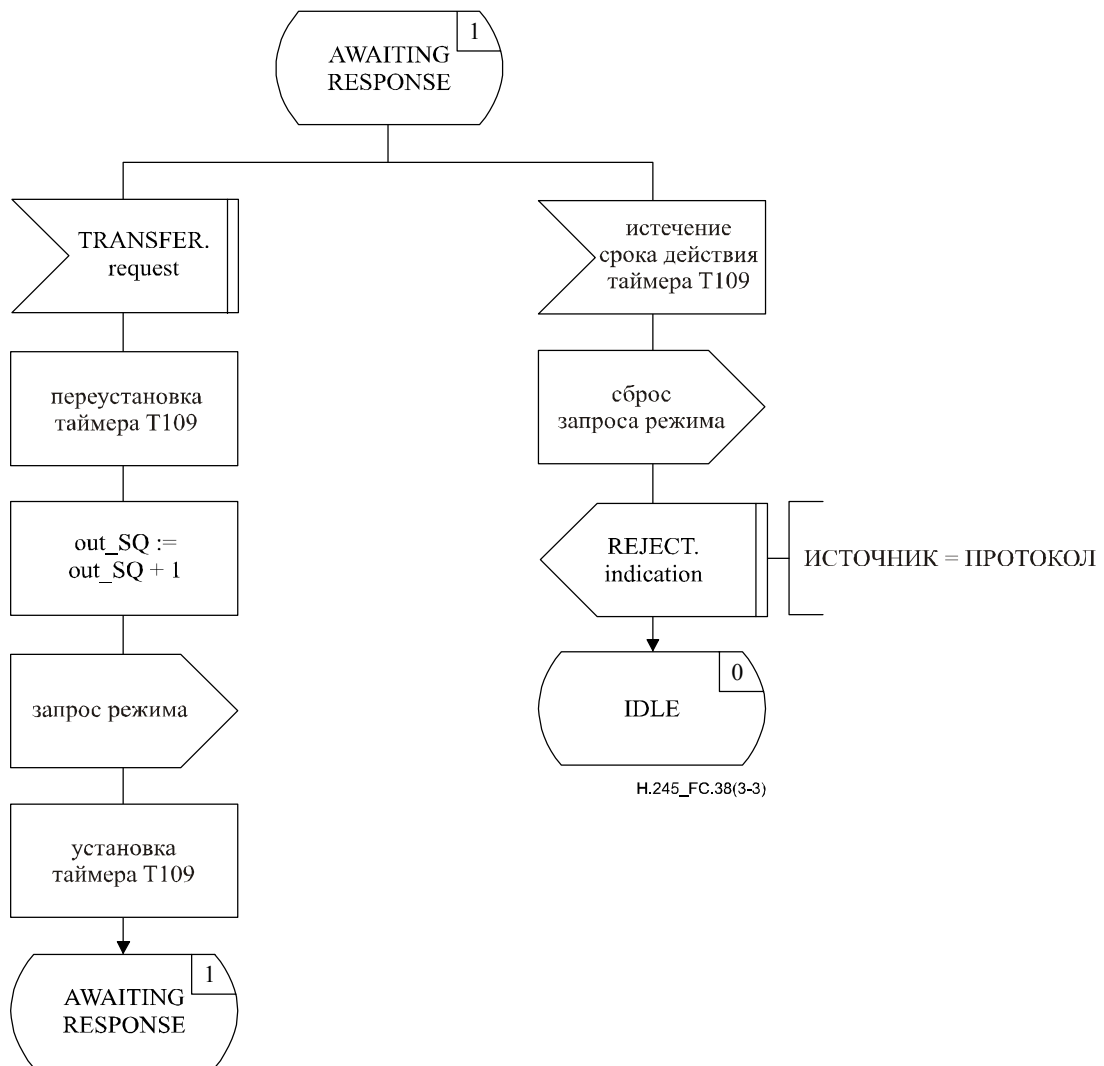


Рисунок С.38/Н.245 – Диаграмма SDL для передающего (лист 3 из 3)

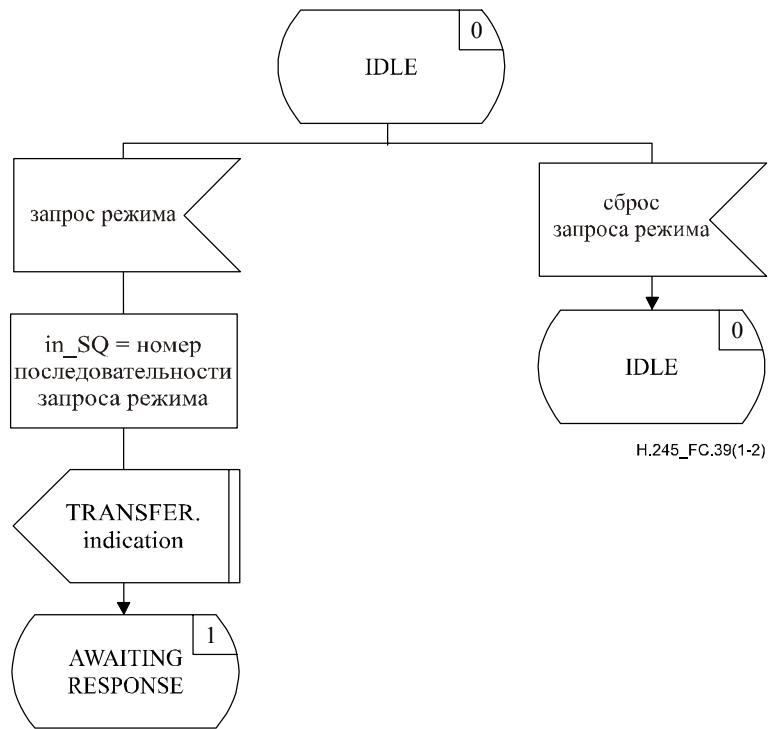


Рисунок С.39/Н.245 – Диаграмма SDL для принимающего MRSE (лист 1 из 2)

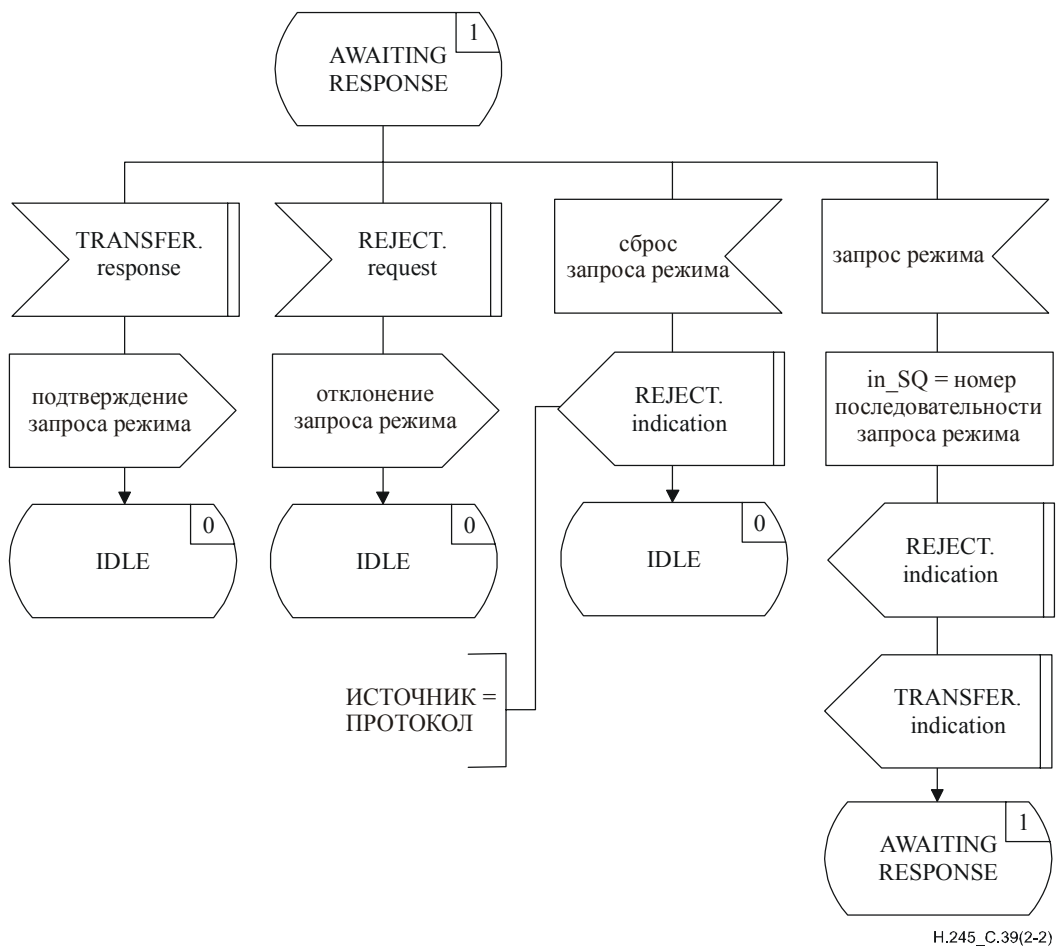


Рисунок С.39/Н.245 – Диаграмма SDL для принимающего MRSE (лист 2 из 2)

С.10 Процедуры двусторонней задержки из-за подтверждения приема

С.10.1 Введение

Описанные здесь процедуры позволяют определить задержку из-за подтверждения приема и между двумя соединенными терминалами. Кроме того, эта функция позволяет пользователю Н.245 определить, существует ли еще равноправный объект протокола Н.245.

Описываемая здесь функция называется далее Объектом сигнализации задержки из-за подтверждения приема (RTDSE). Процедуры определены с помощью примитивов и состояний на интерфейсе между RTDSE и пользователем RTDSE. В каждом терминале существует один экземпляр RTDSE. Любой терминал может произвести определение задержки из-за подтверждения приема.

В приведенном ниже тексте содержится общее описание действия протокола RTDSE. В случае каких-либо расхождений между ним и официальной спецификацией протокола преимущественную силу имеет официальная спецификация.

С.10.1.1 Общее описание протокола – RTDSE

Процедура определения задержки из-за подтверждения приема инициируется передачей пользователем RTDSE примитива TRANSFER.request. Сообщение RoundTripDelayRequest передается равноправному RTDSE, и запускается таймер T105. Если сообщение RoundTripDelayResponse получено в ответ на сообщение RoundTripDelayRequest, то таймер T105 останавливается и пользователь с помощью примитива TRANSFER.confirm информируется о задержке из-за подтверждения приема, величина которой равна значению таймера T105.

Если от равноправного RTDSE в любой момент времени получено сообщение RoundTripDelayRequest, то равноправному RTDSE немедленно посылается сообщение RoundTripDelayResponse.

Если окончился срок действия таймера T105, то пользователь RTDSE информируется об этом примитивом EXPIRY.indication.

Принимается только то сообщение RoundTripDelayResponse, которое является ответом на самое последнее по времени сообщение RoundTripDelayRequest. Игнорируются сообщения, которые переданы в ответ на более ранние сообщения RoundTripDelayRequest.

Новая процедура определения задержки из-за подтверждения приема может быть инициирована пользователем RTDSE с помощью примитива TRANSFER.request до получения сообщения RoundTripDelayResponse.

С.10.2 Связь между RTDSE и пользователем RTDSE

С.10.2.1 Обмен примитивами между RTDSE и пользователем RTDSE

Взаимодействие между RTDSE и пользователем RTDSE осуществляется с помощью примитивов, приведенных в таблице С.37. Эти примитивы используются для определения процедур RTDSE и не предназначены для конкретизации или ограничения реализации.

Таблица С.37/Н.245 – Примитивы и параметры

Универсальное название	Тип			
	запрос	индикация	ответ	подтверждение
TRANSFER	– (Примечание 1)	не определен (Примечание 2)	не определен	DELAY
EXPIRY	не определен	–	не определен	не определен

ПРИМЕЧАНИЕ 1. – "-" означает отсутствие параметров.
ПРИМЕЧАНИЕ 2. – "не определен" означает, что этот примитив не определен.

С.10.2.2 Определение примитивов

Ниже приведены определения этих примитивов:

- a) Примитив TRANSFER используется для запроса определения из-за подтверждения приема и сообщения его результатов.
- b) Примитив EXPIRY показывает, что от равноправного терминала не получено никакого ответа.

С.10.2.3 Определение параметра

Ниже приведено определение параметров примитивов, приведенных в таблице С.37:

- a) Параметр DELAY возвращает измеренную задержку из-за подтверждения приема.

С.10.2.4 Состояния RTDSE

Описанные ниже состояния используются для определения разрешенной последовательности обмена примитивами между RTDSE и пользователем RTDSE.

Состояние 0: IDLE

В данном состоянии передача RTDSE не выполняется.

Состояние 1: AWAITING RESPONSE

Пользователь RTDSE запросил измерение задержки из-за подтверждения приема. Ожидается ответ от равноправного RTDSE.

С.10.2.5 Диаграмма переходов между состояниями

Здесь определяется разрешенная последовательность обмена примитивами между RTDSE и пользователем RTDSE. Разрешенные последовательности показаны на рисунке С.40.

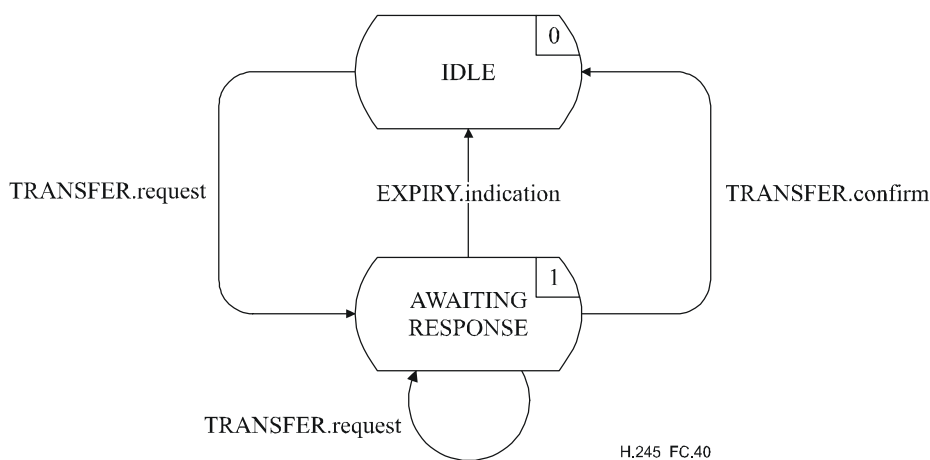


Рисунок С.40/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в RTDSE

С.10.3 Взаимодействие равноправных RTDSE

С.10.3.1 Сообщения

В таблице С.38 показаны определенные в Приложении А сообщения и поля RTDSE, относящиеся к протоколу RTDSE.

Таблица С.38/Н.245 – Названия и поля сообщений RTDSE

Функция	Сообщение	Поле
transfer	RoundTripDelayRequest	sequenceNumber
	RoundTripDelayResponse	sequenceNumber

С.10.3.2 Переменные состояния RTDSE

Для RTDSE определена следующая переменная состояния:

out_SQ

Эта переменная состояния используется для индикации самого последнего по времени сообщения RoundTripDelayRequest. Она увеличивается на единицу и отображается в поле sequenceNumber сообщения RoundTripDelayRequest перед передачей сообщения RoundTripDelayRequest. Для out_SQ используется арифметика по модулю 256.

С.10.3.3 Таймеры RTDSE

Для RTDSE определен следующий таймер:

T105

Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщения RoundTripDelayResponse.

С.10.4 Процедуры RTDSE

С.10.4.1 Введение

На рисунке С.41 показаны все примитивы RTDSE и их параметры, а также сообщения.

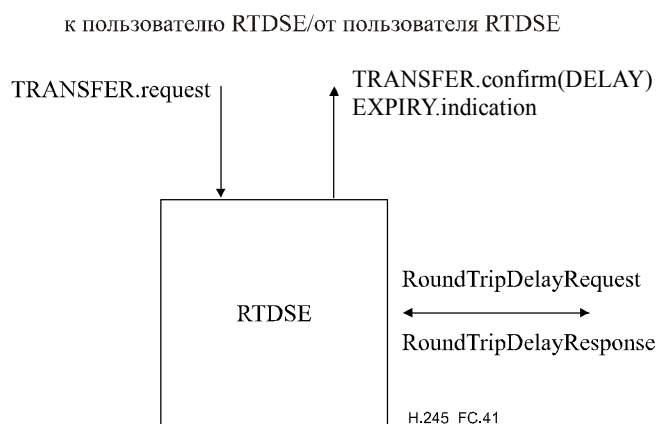


Рисунок С.41/Н.245 – Примитивы и сообщения в RTDSE

С.10.4.2 Принимаемые по умолчанию значения параметров примитивов

Параметры примитивов индикации и квитирования принимают значения, приведенные в таблице С.39, если в диаграммах SDL в явном виде не задано иное.

Таблица С.39/Н.245 – Принимаемые по умолчанию значения параметров примитивов

Примитив	Параметр	Значение по умолчанию
TRANSFER.confirm	DELAY	начальное значение таймера T105 минус значение таймера T105
EXPIRY.indication	–	–

ПРИМЕЧАНИЕ. – Таймеры определены таким образом, что они производят счет до нуля. Параметр DELAY показывает время, в течение которого таймер производит отсчет, таким образом его значение равно разности между начальным значением и оставшимся значением таймера.

С.10.4.3 Принимаемые по умолчанию значения полей сообщений

Поля сообщения принимают значения, показанные в таблице С.40, если в диаграммах SDL в явном виде не задано иное.

Таблица С.40/Н.245 – Принимаемые по умолчанию значения полей сообщений

Сообщение	Поле	Значение по умолчанию
RoundTripDelayRequest	sequenceNumber	out_SQ
RoundTripDelayResponse	sequenceNumber	RoundTripDelayRequest.sequenceNumber

С.10.4.4 Диаграммы языка описания структур (SDL)

Процедуры RTDSE представлены в форме SDL на рисунке С.42.

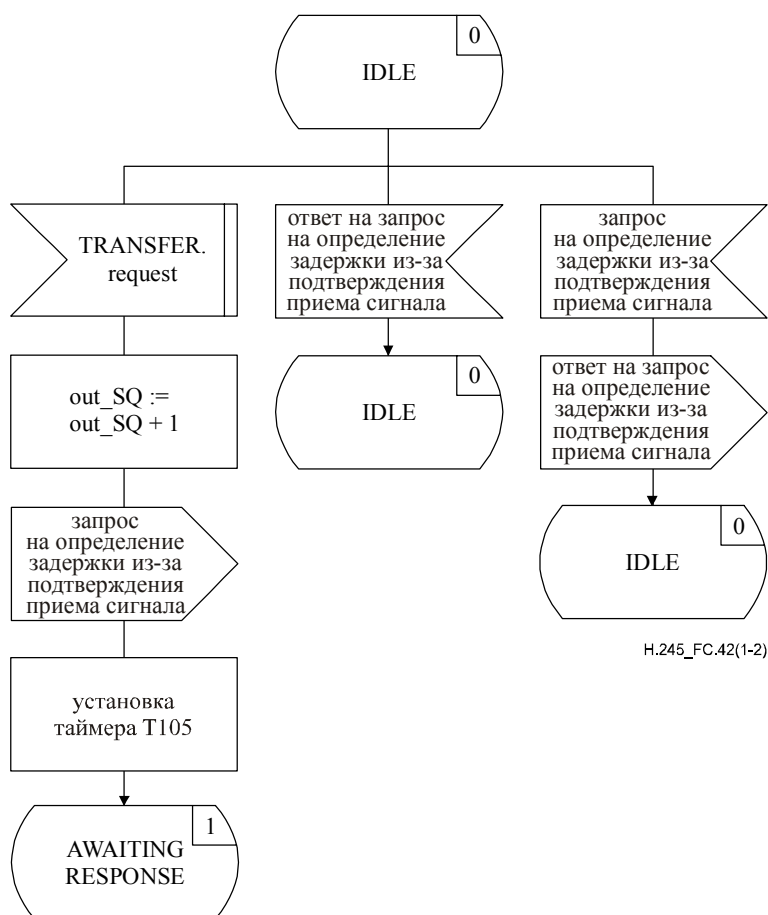


Рисунок С.42/Н.245 – Диаграмма SDL RTDSE (лист 1 из 2)

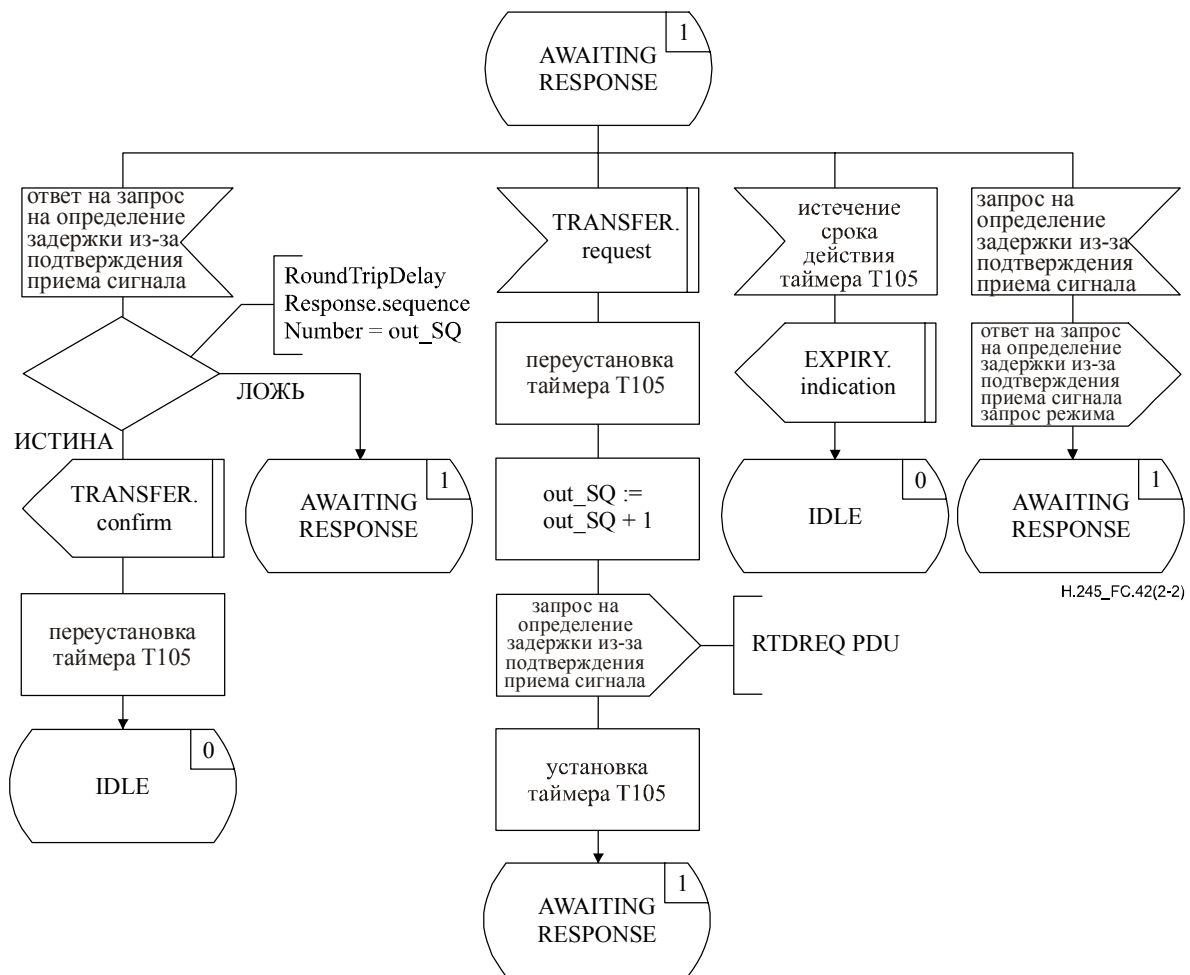


Рисунок С.42/Н.245 – Диаграмма SDL RTDSE (лист 2 из 2)

С.11 Процедуры цикла обслуживания

С.11.1 Введение

Описанный здесь протокол обеспечивает надежную работу циклов обслуживания с использованием процедур с подтверждением приема.

Описанный здесь протокол называется далее Объектом сигнализации цикла обслуживания (MLSE). Процедуры определяются с помощью примитивов на интерфейсе между MLSE и пользователем MLSE, а также состояний MLSE. Информация протокола передается равноправному MLSE с помощью соответствующих сообщений, которые определены в Приложении А.

Существует передающий MLSE и принимающий MLSE. На каждой из передающих и принимающих сторон существует один экземпляр MLSE для каждого двунаправленного логического канала, и имеется один экземпляр MLSE для системного цикла. Не существует иной связи между принимающим MLSE и передающим MLSE на одной стороне, чем обмен примитивами к пользователю MLSE и от него. Формируются сообщения об условиях возникновения ошибок MLSE.

Терминал, который содержит принимающий MLSE, должен передавать по циклу соответствующие данные только тогда, когда он находится в состоянии LOOPED, но не передавать ни в какое другое время. Терминал, который содержит передающий MLSE, должен быть в состоянии принимать передаваемые в цикле данные в любом состоянии, но в состоянии LOOPED он должен принимать только такие данные.

ПРИМЕЧАНИЕ. – Сообщение `MaintenanceLoopOffCommand` применимо ко всем MLSE. Оно всегда используется для прекращения всех циклов обслуживания.

В приведенном ниже тексте содержится общее описание действия протокола MLSE. В случае каких-либо расхождений между ним и официальной спецификацией протокола, которая приведена ниже, преимущественную силу имеет официальная спецификация.

С.11.1.1 Обзор протокола – передающий MLSE

Организация цикла обслуживания инициируется передачей пользователем в передающем MLSE примитива LOOP.request. Равноправному принимающему MLSE передается сообщение MaintenanceLoopRequest, и запускается таймер T102. Если в ответ на сообщение MaintenanceLoopRequest получено сообщение MaintenanceLoopAck, то таймер T102 останавливается, и пользователь с помощью примитива LOOP.confirm информируется о том, что цикл обслуживания был успешно организован. Однако если в ответ на сообщение MaintenanceLoopRequest получено сообщение MaintenanceLoopReject, то таймер T102 останавливается, и пользователь с помощью примитива RELEASE.indication информируется о том, что пользователь равноправного MLSE отказал в организации цикла для технического обслуживания.

Если в этот период заканчивается срок действия таймера T102, то пользователь информируется об этом с помощью примитива RELEASE.indication, и равноправному MLSE передается сообщение MaintenanceLoopOffCommand. Оно отменяет все циклы обслуживания, а не только тот цикл, который относится к определенному MLSE.

Успешно организованный цикл обслуживания может быть прекращен, когда пользователь в передающем MLSE формирует примитив RELEASE.request. Равноправному принимающему MLSE передается сообщение MaintenanceLoopOffCommand.

До того как в ответ на ранее отправленное сообщение MaintenanceLoopRequest будет получено сообщение MaintenanceLoopAck или MaintenanceLoopReject, пользователь в передающем MLSE может прекратить цикл обслуживания, используя примитив RELEASE.request.

С.11.1.2 Обзор протокола – принимающий MLSE

Если в принимающем MLSE получено сообщение MaintenanceLoopRequest, пользователь с помощью примитива LOOP.indication информируется о запросе на организацию цикла обслуживания. Пользователь принимающего MLSE подтверждает получение запроса на организацию цикла обслуживания, для чего он формирует примитив LOOP.response, и равноправному передающему MLSE передается сообщение MaintenanceLoopAck. С этого момента должен выполняться цикл обслуживания. Пользователь принимающего MLSE сообщает об отклонении запроса на организацию цикла обслуживания путем формирования примитива RELEASE.request, и равноправному передающему MLSE передается сообщение MaintenanceLoopReject.

Успешно организованный цикл обслуживания может быть прекращен, если в принимающем MLSE будет получено сообщение MaintenanceLoopOffCommand. Пользователь принимающего MLSE извещается об этом с помощью примитива RELEASE.indication.

С.11.2 Взаимодействие между MLSE и пользователем MLSE

С.11.2.1 Обмен примитивами между MLSE и пользователем MLSE

Взаимодействие между MLSE и пользователем MLSE осуществляется с помощью примитивов, приведенных в таблице С.41.

Таблица С.41/Н.245 – Примитивы и параметры

Универсальное название	Тип			
	запрос	индикация	ответ	подтверждение
LOOP	LOOP_TYPE	LOOP_TYPE	– (Примечание 1)	–
RELEASE	CAUSE	SOURCE CAUSE	не определен (Примечание 2)	не определен
ERROR	не определен	ERRCODE	не определен	не определен
ПРИМЕЧАНИЕ 1. – "-" означает отсутствие параметров. ПРИМЕЧАНИЕ 2. – "не определен" означает, что такой примитив не существует.				

С.11.2.2 Определение примитивов

Ниже дано определение указанных примитивов:

- a) Примитивы LOOP используются для организации цикла обслуживания.
- b) Примитивы RELEASE используются для прекращения цикла обслуживания.
- c) Примитив ERROR извещает объекту управления об ошибках MLSE.

С.11.2.3 Определение параметров

Ниже приведены определения параметров примитивов, приведенных в таблице С.41:

- a) Параметр LOOP_TYPE определяет параметры, относящиеся к циклу обслуживания. Он принимает значения "SYSTEM" ("система"), "MEDIA" ("среда") и "LOGICAL_CHANNEL" ("логический канал"). Этот параметр и номер логического канала определяют значение поля type ("тип") сообщения MaintenanceLoopRequest, которое затем прозрачно передается пользователю равноправного MLSE.
- b) Параметр SOURCE показывает пользователю MLSE источник сброса цикла обслуживания. Параметр SOURCE имеет значение "USER" ("пользователь") или "MLSE", указывая или на пользователя MLSE или на MLSE. Последнее значение может быть результатом ошибки протокола.
- c) Параметр CAUSE показывает причину, по которой пользователь равноправного MLSE отклонил запрос на организацию цикла обслуживания. Параметр CAUSE не присутствует, если параметр SOURCE имеет значение "MLSE".
- d) Параметр ERRCODE указывает тип ошибки MLSE. В таблице С.45 приведены допустимые значения параметра ERRCODE.

С.11.2.4 Состояния MLSE

Описанные ниже состояния задают разрешенную последовательность обмена примитивами между MLSE и пользователем MLSE, а также обмена сообщениями между равноправными MLSE. Эти состояния определены отдельно для каждого передающего MLSE и принимающего MLSE. Передающий MLSE имеет следующие состояния:

Состояние 0: NOT LOOPED

Нет цикла обслуживания.

Состояние 1: AWAITING RESPONSE

Передающий MLSE ожидает организации цикла обслуживания с равноправным принимающим MLSE.

Состояние 2: LOOPED

Организован цикл обслуживания равноправных MLSE. Все данные, получаемые по соответствующему каналу, должны передаваться в цикле.

Принимающий MLSE имеет следующие состояния:

Состояние 0: NOT LOOPED

Нет цикла обслуживания.

Состояние 1: AWAITING RESPONSE

Принимающий MLSE ожидает организации цикла обслуживания с равноправным передающим MLSE. Соответствующие данные не должны передаваться в цикле.

Состояние 2: LOOPED

Организован цикл обслуживания равноправных MLSE. Все данные, получаемые по соответствующему каналу, должны передаваться в цикле.

С.11.2.5 Диаграмма переходов между состояниями

Здесь определяется разрешенная последовательность обмена примитивами между MLSE и пользователем MLSE. Разрешенная последовательность обмена примитивами относится к состояниям MLSE, которые видны с точки зрения пользователя MLSE. Разрешенные последовательности обмена примитивами заданы отдельно для каждого передающего MLSE и принимающего MLSE, как это показано на рисунках С.43 и С.44, соответственно.

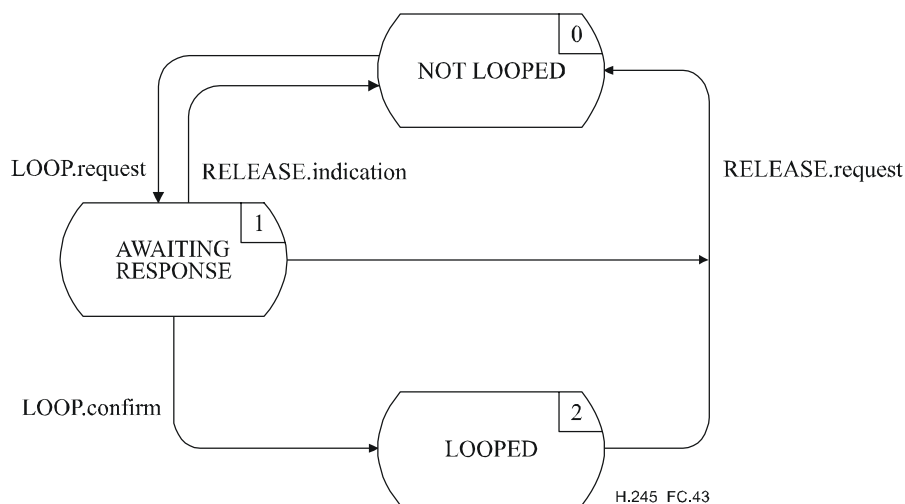


Рисунок С.43/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в передающем MLSE

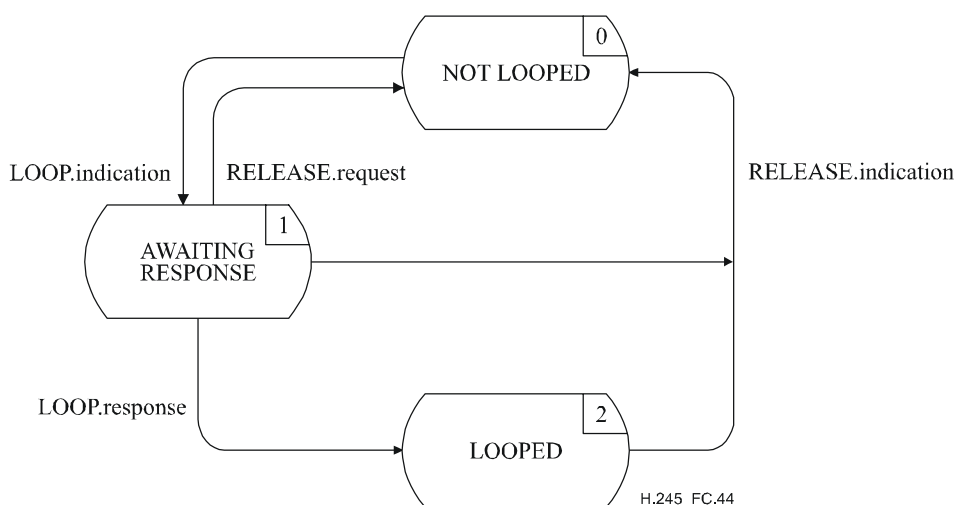


Рисунок С.44/Н.245 – Диаграмма переходов между состояниями для последовательности примитивов в принимающем MLSE

С.11.3 Взаимодействие между равноправными MLSE

С.11.3.1 Сообщения MLSE

В таблице С.42 приведены определенные в Приложении А сообщения и поля MLSE, относящиеся к протоколу MLSE.

Таблица С.42/Н.245 – Названия и поля сообщений MLSE

Функция	Сообщение	Направление	Поле
establish	MaintenanceLoopRequest	O → I (Примечание)	type
	MaintenanceLoopAck	O ← I	type
	MaintenanceLoopReject	O ← I	type cause
release	MaintenanceLoopOffCommand	O → I	–

ПРИМЕЧАНИЕ. – Направление: O – исходящее, I – входящее.

С.11.3.2 Переменные состояний MLSE

Для передающего MLSE определена следующая переменная состояния:

out_MLN

Эта переменная состояния устанавливает различие между передающими MLSE. Она инициализируется при инициализации передающего MLSE. Значение out_MLN используется для установки значения поля type (тип) сообщений MaintenanceLoopRequest, которые переданы передающим MLSE.

Для принимающего MLSE определена следующая переменная состояния:

in_MLN

Эта переменная состояния устанавливает различие между принимающими MLSE. Она инициализируется при инициализации входящего MLSE. В сообщениях MaintenanceLoopRequest, полученных в принимающем MLSE, значение поля type (тип) соответствует значению in_MLN.

in_TYPE

В этой переменной состояния хранится значение LOOP_TYPE при получении сообщения MaintenanceLoopRequest. Эта переменная состояния помогает установить значения поля type в сообщении MaintenanceLoopAck.

С.11.3.3 Таймеры MLSE

Для передающего MLSE определен следующий таймер:

T102

Этот таймер используется в состоянии AWAITING RESPONSE. Это определяет максимальный возможный период без получения сообщения MaintenanceLoopAck и MaintenanceLoopReject.

С.11.4 Процедуры MLSE

С.11.4.1 Введение

На рисунке С.45 показаны все примитивы и их параметры, а также сообщения для каждого передающего и принимающего MLSE.

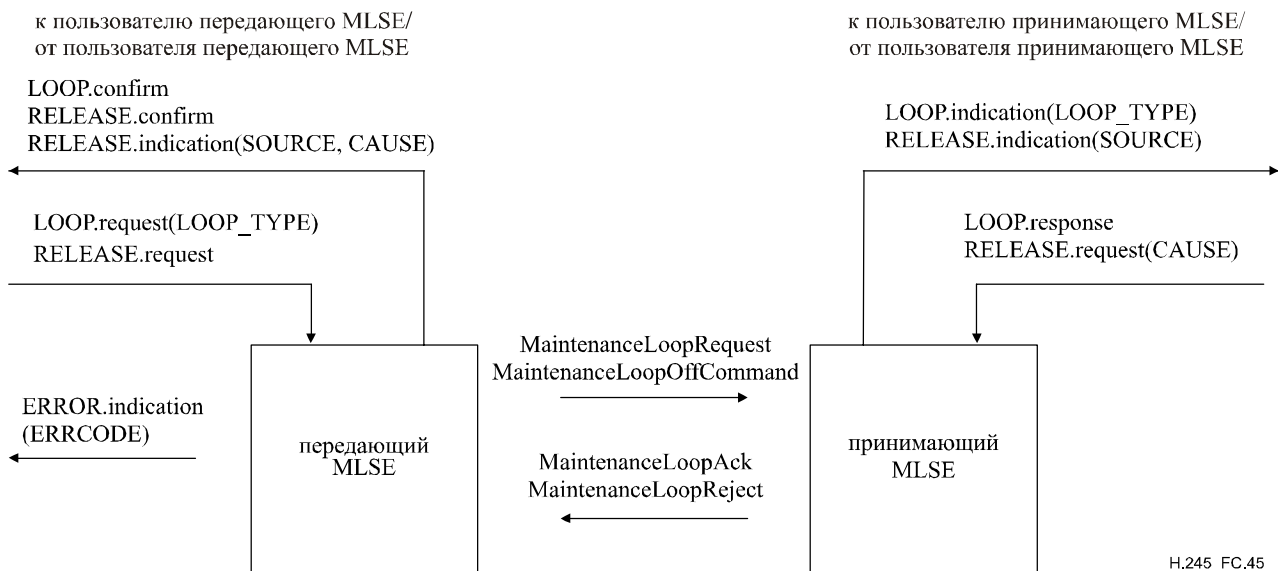


Рисунок С.45/Н.245 – Примитивы и сообщения в Объекте сигнализации цикла обслуживания

С.11.4.2 Принимаемые по умолчанию значения параметров примитивов

Параметры примитивов индикации и подтверждения принимают значения, приведенные в таблице С.43, если в диаграммах SDL не задано явно иное.

Таблица С.43/Н.245 – Принимаемые по умолчанию значения параметров примитивов

Примитив	Параметр	Значение по умолчанию (Примечание)
LOOP.indication	LOOP_TYPE	MaintenanceLoopRequest.type
RELEASE.indication	SOURCE CAUSE	USER MaintenanceLoopReject.cause

ПРИМЕЧАНИЕ. – Параметр примитива должен быть закодирован как null (нуль), если указанное поле сообщения не присутствует в сообщении.

С.11.4.3 Принимаемые по умолчанию значения полей сообщения

Поля сообщения принимают значения, приведенные в таблице С.44, если в диаграммах SDL не задано явно иное.

Таблица С.44/Н.245 – Принимаемые по умолчанию значения полей сообщений

Сообщение	Поле	Значение по умолчанию (Примечание 1)
MaintenanceLoopRequest	Type	LOOP.request(LOOP_TYPE) и out_MLN (Примечание 2)
MaintenanceLoopAck	type	in_LOOP и in_MLN (Примечание 3)
MaintenanceLoopReject	type cause	in_LOOP и in_MLN (Примечание 3) RELEASE.request(CAUSE)
MaintenanceLoopOffCommand	–	–

ПРИМЕЧАНИЕ 1. – Поле сообщения не должно кодироваться, если соответствующий параметр примитива равен null (нулю), т. е. он отсутствует.

ПРИМЕЧАНИЕ 2. – Значение поля type формируется на основе параметра LOOP_TYPE и номера логического канала.

ПРИМЕЧАНИЕ 3. – Значение поля type формируется на основе переменных состояния in_LOOP и in_MLN.

С.11.4.4 Значения параметра ERRCODE

Параметр ERRCODE примитива ERROR.indication указывает определенное условие возникновения ошибки. В таблице С.45 приведены значения, которые может принимать параметр ERRCODE в передающем MLSE. С принимающим MLSE не связано никакого примитива ERROR.indication.

Таблица С.45/Н.245 – Значения параметра ERRCODE в передающем MLSE

Тип ошибки	Код ошибки	Условия ошибки	Состояние
несоответствующее сообщение	A	MaintenanceLoopAck	LOOPED
нет ответа от равноправного MLSE	B	Истечение срока действия таймера T102	AWAITING RESPONSE

С.11.4.5 Диаграммы языка описания структур (SDL)

Процедуры передачи сообщений для передающего MLSE и принимающего MLSE представлены в форме SDL на рисунках С.46 и С.47, соответственно.

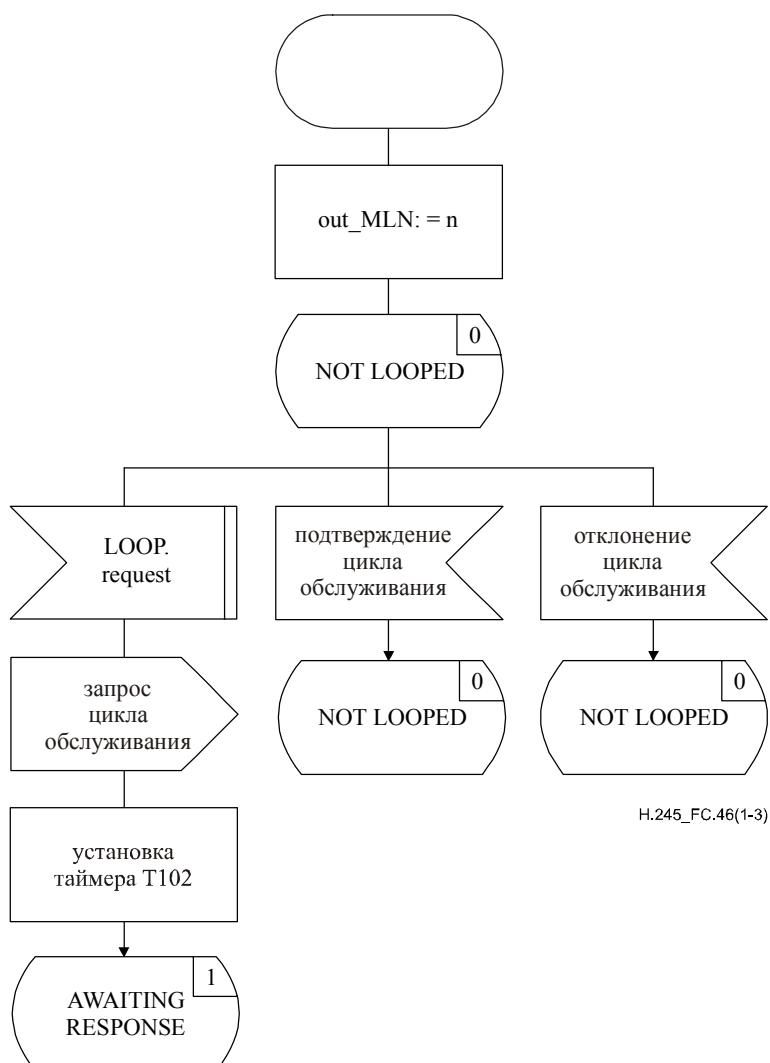


Рисунок С.46/Н.245 – Диаграмма SDL для передающего MLSE (лист 1 из 3)

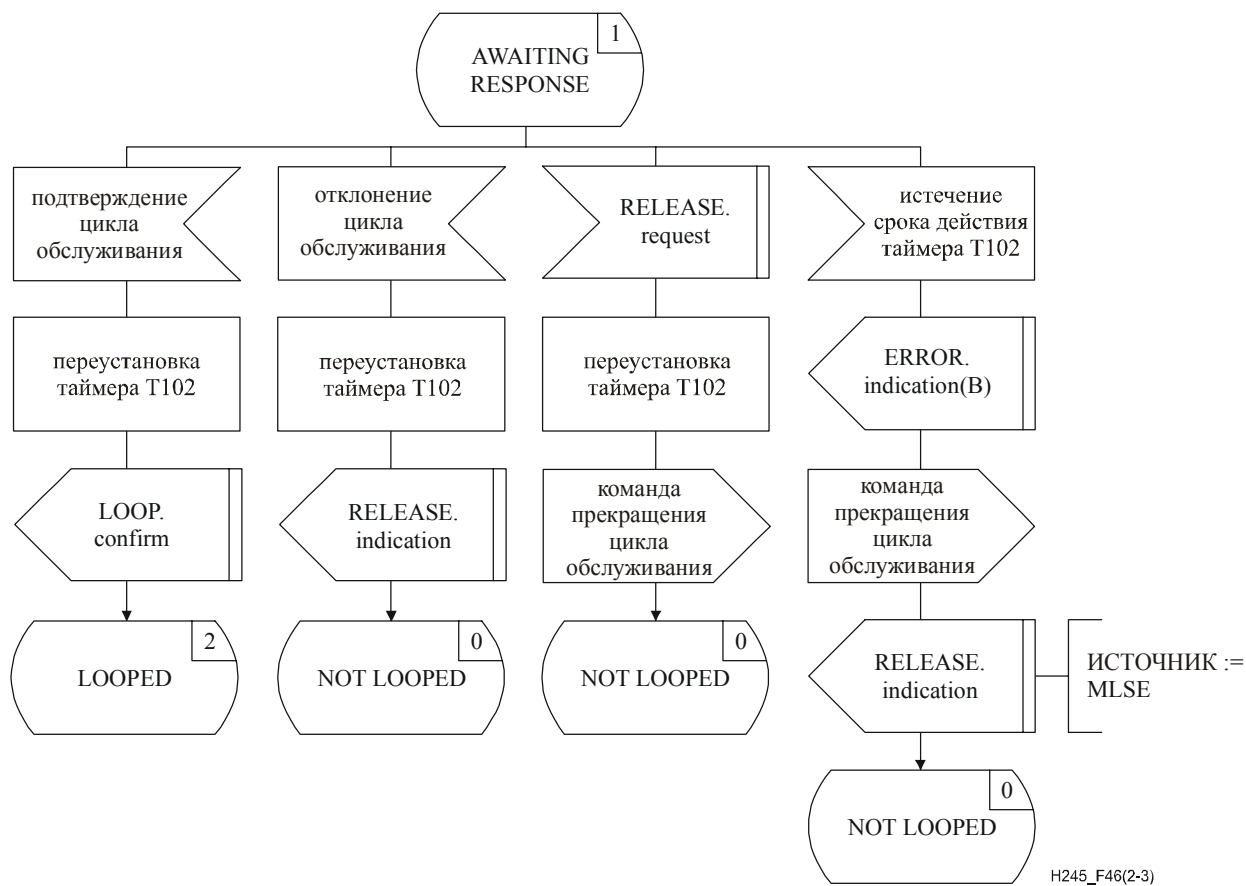


Рисунок С.46/Н.245 – Диаграмма SDL для передающего MLSE (лист 2 из 3)

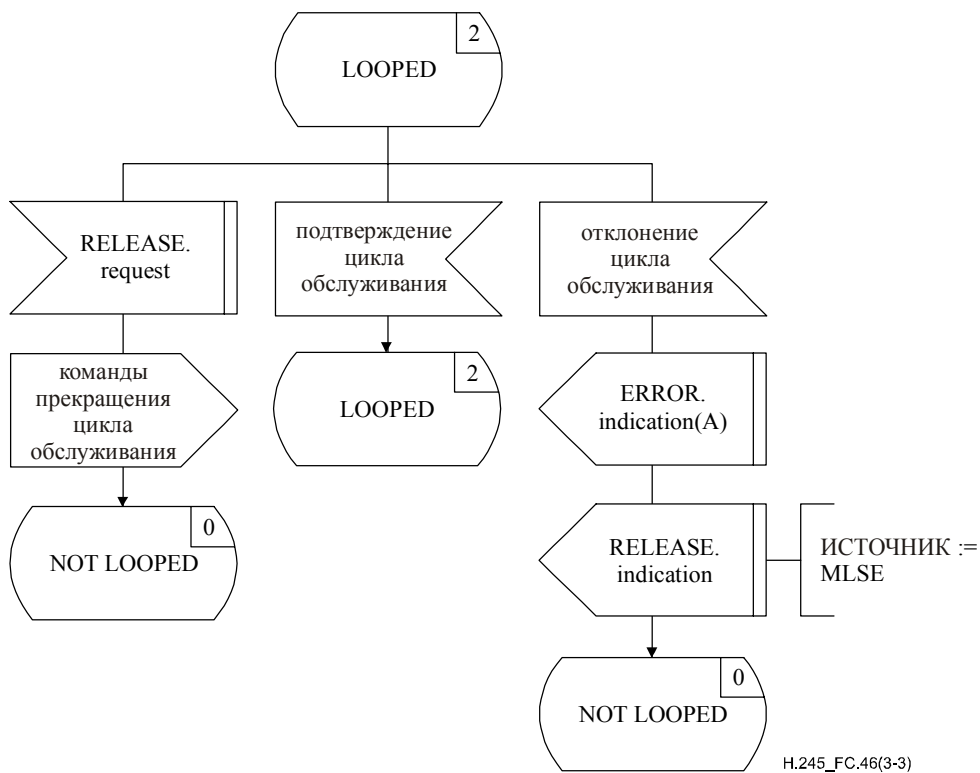


Рисунок С.46/Н.245 – Диаграмма SDL для передающего MLSE (лист 3 из 3)

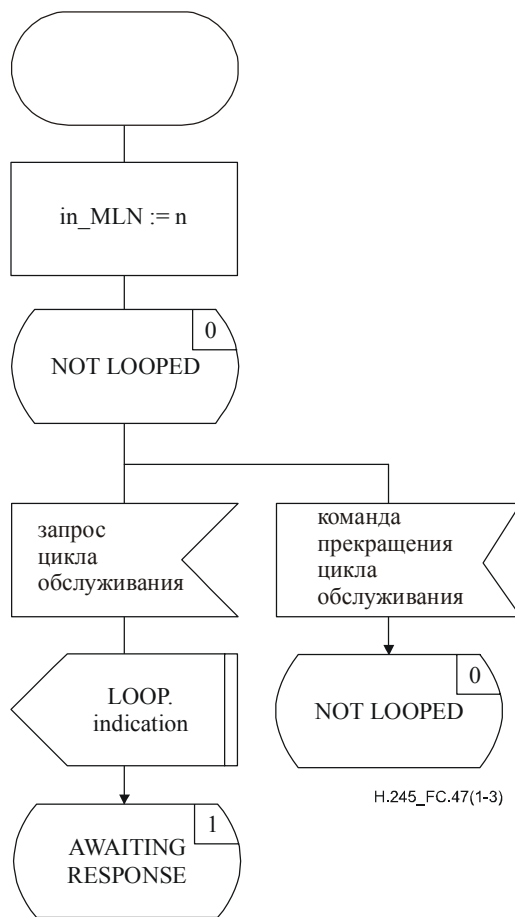


Рисунок С.47/Н.245 – Диаграмма SDL для принимающего MLSE (лист 1 из 3)

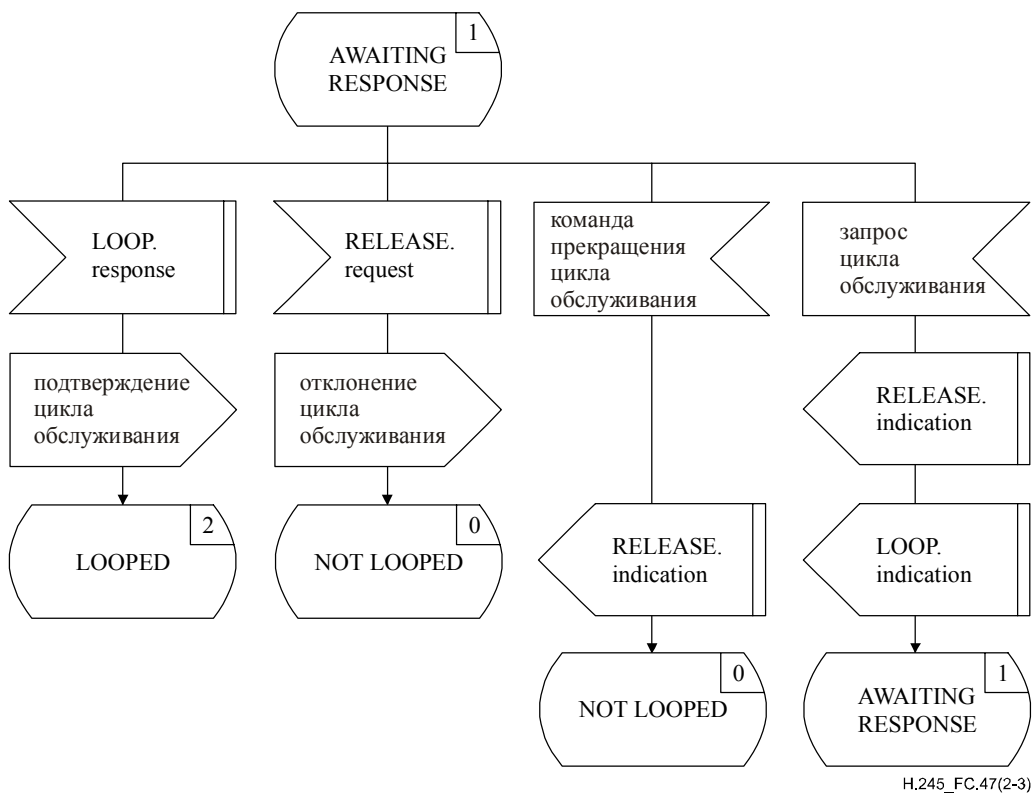


Рисунок С.47/Н.245 – Диаграмма SDL для принимающего MLSE (лист 2 из 3)

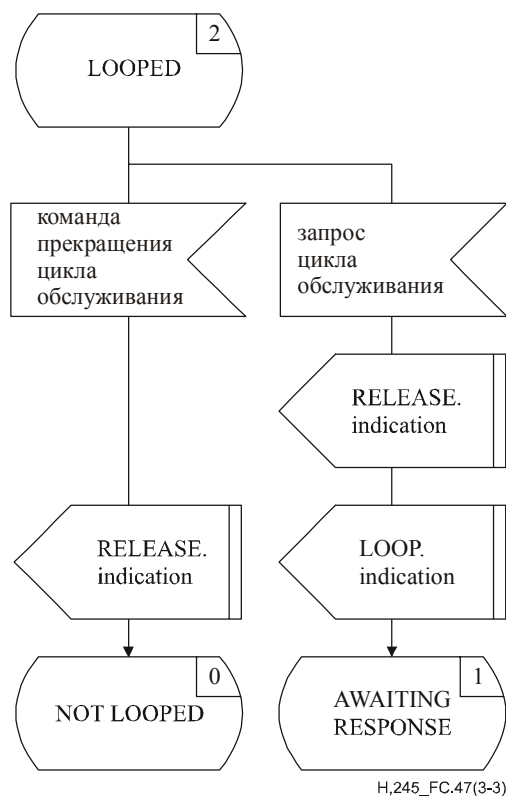


Рисунок С.47/Н.245 – Диаграмма SDL для принимающего MLSE (лист 3 из 3)

Приложение D

Присвоение идентификаторов объектов

В таблице D.1 приведен список назначения идентификаторов объектов, используемых в настоящей Рекомендации.

Таблица D.1/Н.245

Значение идентификатора объекта	Описание
{itu-t (0) recommendation (0) h (8) 245 version (0) 1}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. Он соответствует первой версии настоящей Рекомендации.
{itu-t (0) recommendation (0) h (8) 245 version (0) 2}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. В настоящее время определено десять стандартизованных версий. Он соответствует второй версии настоящей Рекомендации.

Таблица D.1/Н.245

Значение идентификатора объекта	Описание
{itu-t (0) recommendation (0) h (8) 245 version (0) 3}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. В настоящее время определено десять стандартизованных версий. Он соответствует третьей версии настоящей Рекомендации.
{itu-t (0) recommendation (0) h (8) 245 version (0) 4}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. В настоящее время определено десять стандартизованных версий. Он соответствует четвертой версии настоящей Рекомендации.
{itu-t (0) recommendation (0) h (8) 245 version (0) 5}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. В настоящее время определено десять стандартизованных версий. Он соответствует пятой версии настоящей Рекомендации.
{itu-t (0) recommendation (0) h (8) 245 version (0) 6}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. В настоящее время определено десять стандартизованных версий. Он соответствует шестой версии настоящей Рекомендации.
{itu-t (0) recommendation (0) h (8) 245 version (0) 7}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. В настоящее время определено десять стандартизованных версий. Он соответствует седьмой версии настоящей Рекомендации.
{itu-t (0) recommendation (0) h (8) 245 version (0) 8}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. В настоящее время определено десять стандартизованных версий. Он соответствует восьмой версии настоящей Рекомендации.
{itu-t (0) recommendation (0) h (8) 245 version (0) 9}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. В настоящее время определено десять стандартизованных версий. Он соответствует девятой версии настоящей Рекомендации.
{itu-t (0) recommendation (0) h (8) 245 version (0) 10}	Этот идентификатор объекта используется для указания версии данной Рекомендации, используемой в качестве управляющего протокола мультимедийной системы. В настоящее время определено десять стандартизованных версий. Он соответствует десятой версии настоящей Рекомендации.

Таблица D.1/Н.245

Значение идентификатора объекта	Описание
{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) video (0) ISO/IEC 14496-2 (0)}	Этот идентификатор объекта используется для обозначения универсальной возможности ИСО/МЭК 14496-2. Эта возможность определена в Приложении E.
{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) ISO/IEC 14496-3 (0)}	Этот идентификатор объекта используется для обозначения универсальной возможности ИСО/МЭК 14496-3. Эта возможность определена в Приложении H.
{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) amr (1)}	Этот идентификатор объекта используется для обозначения универсальной возможности адаптивного мультискоростного кодека речи GSM. Эта возможность определена в Приложении I.
{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) acelp (2)}	Этот идентификатор объекта используется для обозначения универсальной возможности речевого кодека ACELP, описанного в документе TIA/EIA/ANSI IS-136. Эта возможность определена в Приложении J.
{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) us1 (3)}	Этот идентификатор объекта используется для обозначения универсальной возможности кодека речи US1, описанного в документе TIA/EIA/ANSI IS-136. Эта возможность определена в Приложении K.
{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) is127evrc (4)}	Этот идентификатор объекта используется для обозначения универсальной возможности усовершенствованного кодека с переменной скоростью передачи данных, описанного в документе TIA/EIA IS-127. Эта возможность определена в Приложении L.
{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) data (2) ISO/IEC 14496-1 (0)}	Этот идентификатор объекта используется для обозначения универсальной возможности ИСО/МЭК 14496-1. Эта возможность определена в Приложении G.
{itu-t (0) recommendaton (0) h (8) 245 generic-capabilities (1) control (3) logical-channel-bit-rate-management (0)}	Этот идентификатор объекта используется для обозначения универсальной возможности управления скоростью передачи данных логического канала. Эта возможность определена в Приложении F.
{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) ISO/IEC 13818-7 (5)}	Этот идентификатор объекта используется для обозначения универсальной возможности ИСО/МЭК 13818-7. Эта возможность определена в Приложении M.
{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) itu-r bs.1196 (6)}	Этот идентификатор объекта используется для обозначения универсальной возможности ITU-R. BS.1196. Эта возможность определена в Приложении M.

Приложение Е

Определения возможности ИСО/МЭК 14496-2

В таблице Е.1 определяется идентификатор возможности для возможности ISO/IEC (ИСО/МЭК) 14496-2 [49]. Эти параметры должны быть включены только как **genericVideoCapability** в структуру **VideoCapability** и как **genericVideoMode** в структуру **VideoMode**. В таблицах с Е.2 по Е.6 определяются параметры, относящиеся к этой возможности.

Один экземпляр **profileAndLevel** ИСО/МЭК 14496-2 может поддерживать несколько визуальных объектов. Каждый такой визуальный объект передается как элементарный поток по своему собственному отдельному логическому каналу. Поскольку могут одновременно активно передаваться несколько визуальных сред ИСО/МЭК 14496-2 и поскольку каждая из них может формироваться из нескольких потоков объектов, необходимо иметь механизм, который показывает, какие потоки объектов объединены в одной визуальной среде ИСО/МЭК 14496-2. Такое объединение должно производиться с помощью механизма **forwardLogicalChannelDependency** в **OpenLogicalChannel** всякий раз, когда несколько визуальных объектов используются в одной визуальной среде ИСО/МЭК 14496-2. Все визуальные объекты, объединенные вместе в визуальную среду ИСО/МЭК 14496-2, должны иметь один и тот же профиль и уровень, для чего при открытии логических каналов указывается одно значение **profileAndLevel**. Если логический канал был открыт с обозначением зависимости от некоторого другого логического канала, а логический канал, зависимость от которого была обозначена, закрыт, то оставшиеся открытые логические каналы, которые были ранее сгруппированы вместе некоторой цепью связей **forwardLogicalChannelDependency**, должны оставаться логически сгруппированными в одну визуальную среду ИСО/МЭК 14496-2.

Таблица Е.1/Н.245 – Идентификатор возможности для возможности ИСО/МЭК 14496-2

Название возможности	ISO/IEC 14496-2
Класс возможности	Видеокодек
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) video (0) ISO/IEC 14496-2 (0)}
maxBitRate	Поле maxBitRate всегда должно быть включено.
nonCollapsingRaw	Это поле не должно быть включено.
transport	Это поле не должно быть включено.

Таблица Е.2/Н.245 – Параметр "Profile and Level" (профиль и уровень) возможности ИСО/МЭК 14496-2

Название параметра	profileAndLevel
Описание параметра	Это несвертываемый универсальный параметр. profileAndLevel обозначает возможность обработки конкретных профилей совместно с уровнем в соответствии с таблицей G.1 'Таблица FLC для указания профиля и уровня profile_and_level_indication' ИСО/МЭК 14496-2.
Значение идентификатора параметра	0
Статус параметра	Обязательный
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..255.
Заменяет	–

Таблица Е.3/Н.245 – Параметр "Object" (объект) возможности ИСО/МЭК 14496-2

Название параметра	object
Описание параметра	Это несвертываемый универсальный параметр. object показывает набор инструментов, которые должны использоваться декодером битового потока, содержащегося в логическом канале в соответствии с таблицей 6-10 'Таблица FLC для указания video_object_type' ИСО/МЭК 14496-2.
Значение идентификатора параметра	1
Статус параметра	Необязательный. Не должен присутствовать при обмене возможностями. Должен присутствовать при сигнализации логического канала (Logical Channel Signalling). Может присутствовать в запросе режима (Mode Request)
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..255.
Заменяет	–

Таблица Е.4/Н.245 – Параметр "Decoder Configuration Information" (Информация о конфигурации декодера) возможности ИСО/МЭК 14496-2

Название параметра	decoderConfigurationInformation
Описание параметра	Это несвертываемый универсальный параметр. decoderConfigurationInformation показывает, как сконфигурировать декодер для конкретного объекта (битового потока) (см. подраздел 6.2.1 'Start Codes' и подразделы с К.3.1 'VideoObject' по К.3.4 'FaceObject' ИСО/МЭК 14496-2).
Значение идентификатора параметра	2
Статус параметра	Необязательный. Не должен присутствовать при обмене возможностями и запросе режима. Может присутствовать при сигнализации логического канала.
Тип параметра	Строка октетов
Заменяет	–

Таблица Е.5/Н.245 – Параметр "Drawing order" (Порядок отображения) возможности ИСО/МЭК 14496-2

Название параметра	drawingOrder
Описание параметра	Это несвертываемый универсальный параметр. drawingOrder показывает порядок отображения визуального объекта внутри композиции (возможно наложенных друг на друга) визуальных объектов. Визуальный объект с самым низким значением параметра drawingOrder должен быть нарисован первым. Если визуальные объекты имеют одинаковое значение параметра drawingOrder, то первым должен отображаться объект, соответствующий логическому каналу с самым низким номером. Если параметр drawingOrder не присутствует во время передачи сигналов по логическому каналу, то предполагается, что его значение равно 32768.
Значение идентификатора параметра	3
Статус параметра	Необязательный. Не должен присутствовать при обмене возможностями и запросе режима. Может присутствовать при сигнализации логического канала.
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..65535.
Заменяет	–

Таблица Е.6/Н.245 – Параметр "Visual Back Channel Handle" (Дескриптор обратного визуального канала) возможности ИСО/МЭК 14496-2

Название параметра	visualBackChannelHandle
Описание параметра	Это свертываемый универсальный параметр. Наличие этого параметра показывает, что отправитель получает сообщения канала или что получатель передает сообщения обратного канала, как это предусмотрено в ИСО/МЭК 14496-2.
Значение идентификатора параметра	4
Статус параметра	Может присутствовать при обмене возможностями, сигнализации логического канала и запросе режима.
Тип параметра	Логический
Заменяет	–

Приложение F

Определения, относящиеся к возможности "Управление скоростью передачи данных в логическом канале"

В таблице F.1 определяется идентификатор возможности для управления скоростью передачи данных. Эти параметры, которые предоставляют информацию о том, какие сообщения для управления скоростью передачи данных поддерживаются указанным терминалом, должны быть включены только как **genericControlCapability** структуры **Capability**. В таблицах с F.2 по F.4 определяются относящиеся к данной возможности параметры.

Таблица F.1/Н.245 – Идентификатор возможности "управление скоростью передачи данных в логическом канале"

Название возможности	H.245 Logical Channel Bit-Rate Management
Класс возможности	Управляющая.
Тип идентификатора возможности	Стандартный.
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) control (3) logical-channel-bitrate-management (0)}
maxBitRate	Поле maxBitRate не должно быть включено.
nonCollapsingRaw	Это поле не должно быть включено.
transport	Это поле не должно быть включено.

Таблица F.2/Н.245 – Параметр "Flow Control Capability" (способность управления потоком данных) возможности "управление скоростью передачи данных"

Название параметра	Flow Control Capability
Описание параметра	Это свертываемый универсальный параметр. Наличие этого параметра показывает возможность поддержки сообщения FlowcontrolIndication.
Значение идентификатора параметра	0
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

**Таблица F.3/Н.245 – Параметр "Logical Channel Bit-Rate Change Capability"
(способность изменения скорости передачи данных логического канала)
возможности "управление скоростью передачи данных"**

Название параметра	Logical Channel Bit-Rate Change Capability
Описание параметра	Это свертываемый универсальный параметр. Присутствие этого параметра показывает возможность поддержки процедуры изменения скорости передачи данных в логическом канале, которая использует сообщения LogicalChannelRateRequest, LogicalChannelRateAcknowledge, LogicalChannelRateReject и LogicalChannelRateRelease.
Значение идентификатора параметра	1
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

**Таблица F.4/Н.245 – Параметр "RTCP frequency" (частота RTCP) возможности
"управление скоростью передачи данных"**

Название параметра	RTCP Frequency Capability
Описание параметра	Это свертываемый универсальный параметр. Этот параметр показывает частоту, с которой терминал может отправлять сообщения RTCP.
Значение идентификатора параметра	2
Статус параметра	Необязательный
Тип параметра	Unsigned32Min
Заменяет	–

Приложение G

Определения, относящиеся к возможности ИСО/МЭК 14496-1

В таблице G.1 определяется идентификатор возможности ИСО/МЭК 14496-1 [48]. В таблицах с G.2 по G.6 определяются параметры, относящиеся к возможности ИСО/МЭК 14496-1. Эти параметры должны быть включены только как **genericDataCapability** в структуры **DataCapability** и как **genericDataMode** структуры **DataMode**. При обмене возможностями параметры streamType и profileAndLevel должны быть определены, а параметр objectType может быть определен. При открытии логического канала (прямого или обратного) должны быть определены или ES_ID, или objectDescriptor.

Дальнейшая информация об использовании универсальной возможности ИСО/МЭК 14496-1 включена в Приложение F/Н.324.

G.1 Идентификатор возможности

Таблица G.1/Н.245 – Идентификатор возможности ИСО/МЭК 14496-1

Название возможности	ISO/IEC 14496-1
Класс возможности	Приложение передачи данных
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) data (2) ISO/IEC 14496-1 (0)}
maxBitRate	Это поле должно быть включено для обозначения максимальной скорости передачи данных одиночного потока данных ISO/IEC 14496.
nonCollapsingRaw	Это поле не должно быть включено.
transport	Это поле должно быть включено для обозначения протокола защиты от ошибок конкретного потока данных ИСО/МЭК 14496, передающегося в одном логическом канале.

G.2 Параметры возможности, используемые для согласования возможностей и сигнализации в логическом канале

Таблица G.2/Н.245 – Параметр streamType ("тип потока") данной возможности

Название параметра	streamType
Описание параметра	Это несвертываемый универсальный параметр. StreamType указывает тип потока ИСО/МЭК 14496, который упоминается особым образом в универсальной возможности ИСО/МЭК 14496-1 в соответствии с таблицей 9 ("Значения параметра streamType ") ИСО/МЭК 14496-1.
Значение идентификатора параметра	0
Статус параметра	Необязательный. Должен присутствовать при обмене возможностями. Не должен присутствовать при сигнализации логического канала и запросе режима.
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..255.
Заменяет	–

Таблица G.3/H.245 – Параметр profileAndLevel данной возможности

Название параметра	profileAndLevel
Описание параметра	<p>Это несвертываемый универсальный параметр.</p> <p>ProfileAndLevel показывает возможность обработки конкретных профилей совместно с уровнями в соответствии с</p> <ul style="list-style-type: none"> • таблицей 3 документа ИСО/МЭК 14496-1 (Значения параметра "ODProfileLevelIndication Values") для значения streamType = 0x01 • таблицей 4 документа ИСО/МЭК 14496-1 (Значения параметра "sceneProfileLevelIndication Values") для значения streamType = 0x03 • таблицей 5 документа ИСО/МЭК 14496-1 (Значения параметра "audioProfileLevelIndication Values") для значения streamType = 0x05 • таблицей 6 документа ИСО/МЭК 14496-1 (Значения параметра "visualProfileLevelIndication Values") для значения streamType = 0x04
Значение идентификатора параметра	1
Статус параметра	Необязательный. Должен присутствовать при обмене возможностями. Должен отсутствовать при сигнализации логического канала и запросе режима.
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..255.
Заменяет	–

Таблица G.4/H.245 – Параметр objectType данной возможности

Название параметра	objectType
Описание параметра	<p>Это несвертываемый универсальный параметр.</p> <p>objectType показывает набор инструментов, которые должны использоваться декодером битового потока, содержавшегося в логическом канале в соответствии с</p> <ul style="list-style-type: none"> • таблицей 8 документа ИСО/МЭК 14496-1 ("Значения параметра objectTypeIndication") для значения streamType = 0x04 или 0x05 • таблицей 7 документа ИСО/МЭК 14496-1 ("Значения параметра graphicsProfileLevelIndication") для значения streamType = 0x03 <p>Для всех остальных значений streamType параметр objectType не определен, и поэтому он не должен использоваться.</p>
Значение идентификатора параметра	2
Статус параметра	<p>Необязательный.</p> <p>Для значений параметра streamType = 0x04 или 0x05 не должен присутствовать при обмене возможностями, он должен присутствовать при сигнализации логического канала. Может присутствовать при запросе режима.</p> <p>Для значения параметра streamType = 0x03 должен присутствовать при обмене возможностями, он должен присутствовать при сигнализации логического канала. Может присутствовать при запросе режима.</p> <p>Не должен присутствовать для других значений параметра streamType.</p>
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..255.
Заменяет	–

G.3 Параметры возможности, используемые только для сигнализации логического канала

Таблица G.5/H.245 – Параметр **objectDescriptor** данной возможности

Название параметра	ObjectDescriptor
Описание параметра	Это несвертываемый универсальный параметр. ObjectDescriptor содержит строку октетов, которая обеспечивает всю необходимую информацию для конфигурирования декодера для конкретного битового потока в одном логическом канале (см. документ ИСО/МЭК 14496-1). Он должен содержать информацию только для одного элементарного потока.
Значение идентификатора параметра	3
Статус параметра	Необязательный. Не должен присутствовать при обмене возможностями и запросе режима. Может присутствовать при сигнализации логического канала.
Тип параметра	Строка октетов
Заменяет	–

Таблица G.6/H.245 – Параметр **ES_ID** данной возможности

Название параметра	ES_ID
Описание параметра	Это несвертываемый универсальный параметр. ES_ID показывает идентификатор элементарного потока, который содержится в одном конкретном логическом канале и по которому на него могут ссылаться другие потоки данных ИСО/МЭК 14496. Для InitialObjectDescriptor ES_ID должно присваиваться значение '0' (нулю).
Значение идентификатора параметра	4
Статус параметра	Необязательный. Должен отсутствовать при обмене возможностями. Может присутствовать при сигнализации логического канала. Должен присутствовать при запросе режима.
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..65535.
Заменяет	–

Приложение Н

Определения, относящиеся к возможности ИСО/МЭК 14496-3

В таблице Н.1 определяется идентификатор возможностей ИСО/МЭК 14496-3 [50] и ИСО/МЭК 14496-3/Amd.1 [51]. В таблицах с Н.2 по Н.11 задаются параметры, связанные с возможностями ИСО/МЭК 14496-3. Эти параметры должны включаться только как genericAudioCapability в структуре AudioCapability и genericAudioMode в структуре AudioMode. При обмене возможностями должны присутствовать параметры profileAndLevel, formatType и maxAl-sduAudioFrames, могут присутствовать параметры audioObjectType и maxAudioObjects, а все остальные параметры должны отсутствовать. Если параметр formatType показывает формат Transport Stream (транспортного потока) ИСО/МЭК 14496-3, то при обмене возможностями должен присутствовать параметр maxAudioObjects. При открытии логического канала (прямого или обратного) должны присутствовать параметры profileAndLevel, formatType и audioObjectType, а все остальные параметры могут быть определены. При запросе режима должны присутствовать параметры profileAndLevel и formatType, а параметр audioObjectType может быть определен.

Параметр profileAndLevel ИСО/МЭК 1446-3 и ИСО/МЭК 14496-3/Amd.1 может поддерживать несколько типов аудиообъектов. Аудиообъект должен передаваться как один из двух форматов

битового потока, которыми являются формат raw data (необработанных данных) и формат Transport Stream (транспортного потока) согласно ИСО/МЭК 14496-3. Параметр formatType показывает выбранный тип формата битового потока. В приложениях, использующих передачу с различными скоростями передачи данных или масштабируемую передачу данных, полезно предусмотреть возможность изменения структуры аудиообъектов в одном логическом канале. Это можно обеспечить с помощью формата MPEG-4/Audio, который позволяет изменять конфигурацию потока по кадрам. Для снижения избыточности при передаче конфигурации потока в каждом кадре при низкоскоростной передаче может использоваться формат необработанных данных.

Таблица Н.1/Н.245 – Идентификатор возможности ИСО/МЭК 14496-3

Название возможности	ISO/IEC 14496-3
Класс возможности	Аудиокодек
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) ISO/IEC 14496-3 (0)}
maxBitRate	Это поле всегда должно быть включено.
nonCollapsingRaw	Это поле не должно быть включено.
transport	Это поле не должно быть включено.

Таблица Н.2/Н.245 – Параметр "Profile and Level" ("профиль и уровень") возможности ИСО/МЭК 14496-3

Название параметра	profileAndLevel
Описание параметра	Это несвертываемый универсальный параметр. profileAndLevel показывает возможность обработки конкретных профилей совместно с уровнем в соответствии с документами ИСО/МЭК 14496-1 и ИСО/МЭК 14496-1/Amd.1.
Значение идентификатора параметра	0
Статус параметра	Обязательный
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..255.
Заменяет	–

Таблица Н.3/Н.245 – Параметр "formatType" ("тип формата") возможности ИСО/МЭК 14496-3

Название параметра	formatType
Описание параметра	Это несвертываемый универсальный параметр. FormatType показывает выбор типа формата битового потока аудиообъекта между форматом raw data (необработанных данных) и аудиоформатом следующим образом: <ul style="list-style-type: none"> • 0: формат необработанных данных (согласно документам ИСО/МЭК 14496-3 и ИСО/МЭК 14496-3/Amd.1) • 1: формат, который определен как малоизбыточный мультиплексированный транспортный формат для передачи звуковых данных MPEG-4 (LATM) в ИСО/МЭК 14496-3/Amd.1.
Значение идентификатора параметра	1
Статус параметра	Обязательный
Тип параметра	Логический
Заменяет	–

**Таблица Н.4/Н.245 – Параметр "maxAl-sduAudioFrames"
возможности ИСО/МЭК 14496-3**

Название параметра	maxal-sduFrames
Описание параметра	Это свертываемый универсальный параметр. Он определяет максимальное количество аудио кадров в AL-SDU
Значение идентификатора параметра	2
Статус параметра	Должен присутствовать при обмене возможностями и при сигнализации логического канала. Должен отсутствовать при запросе режима.
Тип параметра	UnsignedMin. Должен находиться в диапазоне 1..256.
Заменяет	–

Таблица Н.5/Н.245 – Параметр "audioObjectType" для возможности ИСО/МЭК 14496-3

Название параметра	audioObjectType
Описание параметра	Это несвертываемый универсальный параметр. audioObjectType показывает набор инструментов, которые должны использоваться декодером битового потока, содержащегося в логическом канале, в соответствии с документом ИСО/МЭК 14496-3/Amd.1. Этот параметр может использоваться для ограничения возможности в конкретном profileAndLevel (профиле и уровне) при обмене возможностями.
Значение идентификатора параметра	3
Статус параметра	Необязательный. Может присутствовать при обмене возможностями. Должен присутствовать при сигнализации логического канала. Может присутствовать при запросе режима.
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..31.
Заменяет	–

Таблица Н.6/Н.245 – Параметр "audioSpecificConfig" возможности ИСО/МЭК 14496-3

Название параметра	audioSpecificConfig
Описание параметра	Это несвертываемый универсальный параметр. audioSpecificConfig показывает порядок конфигурирования декодера для конкретного объекта (см. документ ИСО/МЭК 14496-3/Amd.1).
Значение идентификатора параметра	4
Статус параметра	Необязательный. Должен присутствовать при обмене возможностями и при запросе режима. Должен присутствовать при сигнализации логического канала, если параметр formatType равен 0 (формат необработанных данных). В противном случае он не должен присутствовать при сигнализации логического канала.
Тип параметра	Строка октетов
Заменяет	–

Таблица Н.7/Н.245 – Параметр "maxAudioObjects" возможности ИСО/МЭК 14496-3

Название параметра	maxAudioObjects
Описание параметра	Это свертываемый универсальный параметр. Этот параметр определяет максимальное количество мультиплексированных аудиообъектов в полезной аудионагрузке.
Значение идентификатора параметра	5
Статус параметра	Необязательный. Если formatType равен 0 (формат необработанных данных), то этот параметр не должен присутствовать при обмене возможностями и сигнализации логического канала. В противном случае этот параметр должен присутствовать при обмене возможностями и сигнализации логического канала. Он не должен присутствовать при запросе режима.
Тип параметра	UnsignedMin. Должен находиться в диапазоне 1..16.
Заменяет	–

Таблица Н.8/Н.245 – Параметр "muxConfigPresent" возможности ИСО/МЭК 14496-3

Название параметра	muxConfigPresent
Описание параметра	Это несвертываемый универсальный параметр. muxConfigPresent показывает, происходит ли мультиплексирование данных о конфигурации полезной аудионагрузки в полезную аудионагрузку в соответствии с документом ИСО/МЭК 14496-3/Amd.1: 0: данные о конфигурации полезной аудионагрузки (streamMuxConfig) не мультиплексируются в полезную аудионагрузку. 1: streamMuxConfig мультиплексируется в полезную аудионагрузку.
Значение идентификатора параметра	6
Статус параметра	Необязательный. Не должен присутствовать при обмене возможностями и запросе режима. Должен присутствовать при сигнализации логического канала, если параметр formatType равен 1 (формат LATM). Если параметр formatType не равен 1, он не должен присутствовать при сигнализации логического канала.
Тип параметра	Логический
Заменяет	–

Таблица Н.9/Н.245 – Параметр "EP_DataPresent" возможности ИСО/МЭК 14496-3

Название параметра	EP_DataPresent
Описание параметра	Это несвертываемый универсальный параметр. EP_DataPresent показывает, обладает ли полезная аудионагрузка устойчивостью по отношению к битовым ошибкам (не потерям пакетов) в соответствии с документами ИСО/МЭК (ИСО/МЭК 14496-3/Amd.1): 0: Полезная аудионагрузка не имеет устойчивости к ошибкам. 1: Полезная аудионагрузка устойчива к ошибкам. Данные о конфигурации инструмента ISO/IEC 14496-3/Amd.1 EP (ErrorProtection_SpecificConfig) может присутствовать для Logical Channel Signalling.
Значение идентификатора параметра	7
Статус параметра	Необязательный. Должен отсутствовать при обмене возможностями и запросе режима. Должен присутствовать при сигнализации логического канала, если formatType равен 1 (формат LATM). В противном случае должен отсутствовать при сигнализации логического канала.
Тип параметра	Логический
Заменяет	–

Таблица Н.10/Н.245 – Параметр "streamMuxConfig" возможности ИСО/МЭК 14496-3

Название параметра	streamMuxConfig
Описание параметра	Это несвертываемый универсальный параметр. streamMuxConfig показывает конфигурацию полезной аудионагрузки в соответствии с документом ISO/IEC 14496-3/Amd.1.
Значение идентификатора параметра	8
Статус параметра	Необязательный. Не должен присутствовать при обмене возможностями и запросе режима. Должен присутствовать при сигнализации логического канала, если formatType равен 1 (формат LATM). Если formatType не равен 1, то параметр streamMuxConfig не должен присутствовать при сигнализации логического канала.
Тип параметра	Строка октетов
Заменяет	–

Таблица Н.11/Н.245 – Параметр "ErrorProtection_SpecificConfig" возможности ИСО/МЭК 14496-3

Название параметра	ErrorProtection_SpecificConfig
Описание параметра	Это несвертываемый универсальный параметр. ErrorProtection_SpecificConfig показывает, как сконфигурировать инструмент ИСО/МЭК 14496-3/Amd.1 EP согласно описанию LATM EP_MuxElement() в ИСО/МЭК 14496-3/Amd.1.
Значение идентификатора параметра	9
Статус параметра	Необязательный. Не должен отсутствовать при обмене возможностями и запросе режима. Должен присутствовать при сигнализации логического канала, если параметр formatType равен 1 (формат LATM). Если этот параметр не равен 1, то параметр ErrorProtection_SpecificConfig не должен присутствовать при сигнализации логического канала.
Тип параметра	Строка октетов
Заменяет	–

Приложение I

Определения, относящиеся к возможности "Адаптивная мультискоростная передача данных стандарта GSM"

В таблице I.1 определяется идентификатор возможности "адаптивная мультискоростная передача данных стандарта GSM" (AMR). В таблицах с I.2 по I.7 определяются относящиеся к этой возможности параметры. Соответствующие спецификации представлены в документах [58], [69], [70], [71], [72], [73], [74] и [75].

В разделе I.1 определяется сигнализация режима и формирования пакетов кадров речи в структуру октетов.

Таблица I.1/Н.245 – Идентификатор возможности AMR GSM

Название возможности	AMR
Класс возможности	Аудиокодек
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) amr (1)}
maxBitRate	Должна составлять 122
NonCollapsingRaw	Это поле не должно быть включено
transport	Это поле не должно быть включено

Таблица I.2/Н.245 – Параметр "AMR GSM – maxAl-sduFrames" возможности AMR GSM

Название параметра	maxAl-sduFrames
Описание параметра	Это свертываемый универсальный параметр. Он определяет максимальное количество аудио кадров в AL-SDU
Значение идентификатора параметра	0
Статус параметра	Должен присутствовать при обмене возможностями и сигнализации логического канала. Должен отсутствовать для запроса режима.
Тип параметра	UnsignedMin
Заменяет	–

Таблица I.3/Н.245 – Параметр "bitRate" возможности AMR GSM

Название параметра	bitRate
Описание параметра	Это несвертываемый универсальный параметр. Он определяет скорость передачи данных AMR. Этот параметр должен использоваться только в запросах режима. 0 = 4,75, 1 = 5,15, 2 = 5,90, 3 = 6,70, 4 = 7,40, 5 = 7,95, 6 = 10,2, 7 = 12,2
Значение идентификатора параметра	1
Статус параметра	Необязательный
Тип параметра	UnsignedMin
Заменяет	–

Таблица I.4/Н.245 – Параметр "GSM AMR Comfort Noise" (комфортный шум при AMR GSM) возможности AMR GSM

Название параметра	gsmAmrComfortNoise
Описание параметра	Это свертываемый универсальный параметр. Он определяет, что в запросе режима GSM AMR должен использоваться параметр "комфортный шум GSM AMR". Этот параметр должен применяться только в запросах режима, но не при обмене возможностями, потому что эта возможность обязательна.
Значение идентификатора параметра	2
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Таблица I.5/Н.245 – Параметр "GSM EFR comfort noise" (комфортный шум при EFR GSM) возможности AMR GSM

Название параметра	gsmEfrComfortNoise
Описание параметра	Это свертываемый универсальный параметр. При обмене возможностями этот параметр определяет, присутствует ли возможность "комфортный шум GSM EFR". В запросе режима этот параметр показывает, что требуется возможность "комфортный шум GSM EFR".
Значение идентификатора параметра	3
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Таблица I.6/Н.245 – Параметр "IS-641 comfort noise" (комфортный шум IS-641) возможности AMR GSM

Название параметра	is-641ComfortNoise
Описание параметра	Это свертываемый универсальный параметр. При обмене возможностями этот параметр определяет, присутствует ли возможность "комфортный шум IS-641". В запросе режима этот параметр показывает, что требуется возможность "комфортный шум IS-641".
Значение идентификатора параметра	4
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Таблица I.7/Н.245 – Параметр "PDC EFR comfort noise" (комфортный шум PDC EFR) возможности AMR GSM

Название параметра	pdcEFRComfortNoise
Описание параметра	Это свертываемый универсальный параметр. При обмене возможностями этот параметр определяет, присутствует ли возможность "комфортный шум PDC EFR". PDC EFR является кодеком ACELP со скоростью передачи данных 6,7 кбит/с, описанным в разделе 5.4 документа [74]. В запросе режима этот параметр показывает, что требуется возможность "комфортный шум PDC EFR".
Значение идентификатора параметра	5
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

I.1 Определение сигнализации режима и заполнения битами для обеспечения выравнивания по октетам

Сигнализация режима AMR в системах подвижной связи частично основана на передаче сигналов, не заданных в спецификациях кодека речи AMR. Для обеспечения совместимости с системами подвижной связи в этом разделе определен режим сигнализации, необходимый для использования в режиме AMR в соответствии с Рекомендациями МСЭ-Т серии Н. Размер кадров речи в различных режимах кодека AMR не кратен восьми. По этой причине необходимо заполнение битами для обеспечения октетной структуры.

Обратите внимание на то, что в будущем содержание этого раздела может быть изменено для обеспечения связи с документацией ETSI или с другими соответствующими документами по стандартизации.

В таблице I.10 все режимы AMR отображены в определенных индексах режима $m_i(k)$. Индексы режима также зарезервированы для кадров подавления паузы, используемых в различных системах. Соответствующие форматы заданы в таблицах с I.11 по I.14. В таблице I.15 указан кадр отсутствия передачи.

Передаваемые кодером речи AMR биты $\{s(1),s(2),\dots,s(K_s)\}$ должны быть переставлены в соответствии с субъективным значением перед процедурой выравнивания октетов. В таблицах с I.16 по I.23 определен правильный порядок перестановки для режимов речевого кодека 12,2 кбит/с, 10,2 кбит/с, 7,95 кбит/с, 7,40 кбит/с, 6,70 кбит/с, 5,90 кбит/с, 5,15 кбит/с и 4,75 кбит/с, соответственно. В этих таблицах параметры речевого кодека пронумерованы в том порядке, в котором они поступают из соответствующего речевого кодера согласно GSM 06.90, а переставленным битам присваиваются обозначения $\{d(0),d(1),\dots,d(K_d - 1)\}$, они определены в порядке снижения важности. Индекс K_d относится к количеству битов, переданных речевым кодером (см. таблицу I.8).

Таблица I.8/Н.245 – Количество битов речи в различных режимах AMR

Режим кодека	Количество битов речи, доставляемых в одном блоке (K_d)
AMR12.2	244
AMR10.2	204
AMR7.95	159
AMR7.4	148
AMR6.7	134
AMR5.9	118
AMR5.15	103
AMR4.75	95

Алгоритм упорядочения записывается в псевдокоде в следующем виде:

$$\text{для } j = 0 \text{ до } k_d - 1$$

$$d(j) = s(\text{table}(j) + 1),$$

где $\text{table}(j)$ читается строка за строкой слева направо.

Поэтому структура октета $b_n(k)$ для каждого режима кодека AMR определяется следующим образом:

Количество битов заполнения: $K_s = 8 * N - K_d - K_i$, где K_i – число битов индекса режима

Октет[0]: $b_0(k) = m_i(k)$, для $k = 0, 1, 2, 3$ (индекс режима)
 $b_0(k) = d(k - 4)$, для $k = 4, 5, 6, 7$

Октет[m]: $b_m(k) = d(8 * m - 4 + k)$, для $k = 0, \dots, 7$ и $0 < m < N - 1$

Октет[N - 1]: $b_{N-1}(k) = d(8 * (N - 1) - 4 + k)$, для $k = 0, \dots, 7 - K_s$
 If $K_s > 0$
 $b_{N-1}(k) = UB$, для $k = 8 - K_s, \dots, 7$

Таблица I.9/Н.245 – Пример, показывающий отображение в режиме кодирования речи AMR со скоростью передачи данных 6,7 кбит/с

Октет	Структура октета							LSB
	MSB							
B_0	$d(3)$	$d(2)$	$d(1)$	$d(0)$	0	0	1	1
B_1	$d(11)$	$d(10)$	$d(9)$	$d(8)$	$d(7)$	$d(6)$	$d(5)$	$d(4)$
B_2	$d(12)$
b_{17}	UB	UB	UB	UB	UB	UB	$d(133)$	$d(132)$

Таблица I.10/Н.245 – Отображение режимов кодирования AMR, определенных в GSM 06.90, в биты индекса режима в передаваемых октетах

Mode_index (Индекс режима) (4 бита)	Название в GSM 06.90 и GSM 06.92
0 (Amr4-75k)	режим 4,75 кбит/с
1 (Amr5-15k)	режим 5,15 кбит/с
2 (Amr5-90k)	режим 5,90 кбит/с
3 (Amr6-70k)	режим 6,70 кбит/с (PDC-EFR)
4 (Amr7-40k)	режим 7,40 кбит/с (IS-641)
5 (Amr7-95k)	режим 7,95 кбит/с
6 (Amr10-2k)	режим 10,2 кбит/с
7 (Amr12-2k)	режим 12,2 кбит/с (GSM EFR)
8	кадр GsmAmr Comfort Noise (обязательный)
9	кадр Gsm-Efr Comfort Noise (необязательный)
10	кадр IS-641 Comfort Noise (необязательный)
11	кадр Pdc-Efr Comfort Noise (необязательный)
12-14	Для использования в будущем
15	Нет передачи

Таблица I.11/Н.245 – Отображение битов-дескрипторов комфортного шума из GSM 06.92 в октеты для индекса режима 8 (значения битов с s1 по s35 описаны в GSM 06.92)

Переданные октеты	MSB	Отображение битов						LSB
1	Индекс первого субвектора LSF s4	Индекс эталонного вектора LSF s3 s2 s1			Mode_Index mi(3) mi(2) mi(1) mi(0)			
2	Индекс второго субвектора LSF s12	Индекс первого субвектора LSF s11 s10 s9 s8 s7 s6 s5						
3		Индекс второго субвектора LSF s20 s19 s18 s17 s16 s15 s14 s13						
4		Индекс третьего субвектора LSF s28 s27 s26 s25 s24 s23 s22 s21						
5	бит типа SID t1	Энергия кадра s35 s34 s33 s32 s31 s30						Индекс третьего субвектора LSF s29
6		Биты заполнения UB UB UB UB			Speech_Mode_Indication smi(2) smi(1) smi(0)			

Для дескриптора подавления паузы в таблице I.11 требуются определения дополнительных битов-дескрипторов:

SID-type (t1) имеет значение {0=SID_FIRST, 1=SID_UPDATE}

Индикация режима речи (smi(0)-smi(2)) – это режим речи, соответствующий первым восьми элементам таблице индексов режима (Mode_Index).

Таблица I.12/Н.245 – Отображение битов-дескрипторов подавления паузы из GSM 06.60 (параметры также описаны в GSM 06.62) в октеты для индекса режима 9 (Значения битов с s1 по s91 описаны в GSM 06.60)

Переданные октеты	MSB	Отображение битов						LSB
1	Индекс первой субматрицы LSF s4 s3 s2 s1				Mode_Index mi(3) mi(2) mi(1) mi(0)			
2	Индекс второй субматрицы LSF s12 s11 s10 s9 s8					Индекс первой субматрицы LSF s7 s6 s5		
3	Индекс третьей субматрицы LSF s20 s19 s18 s17 s16					Индекс второй субматрицы LSF s15 s14 s13		
4	Индекс четвертой субматрицы LSF s28 s27 s26 s25				Знак третьей субматрицы LSF s24	Индекс третьей субматрицы LSF s23 s22 s21		
5	Индекс пятой субматрицы LSF s36 s35 s34 s33				Индекс четвертой субматрицы LSF s32 s31 s30 s29			
6	Биты заполнения	Фиксированное усиление кодовой книги					Индекс пятой субматрицы LSF	
	UB	s91	s90	s89	s88	s87	s38	s37

Таблица I.13/Н.245 – Отображение битов-дескрипторов подавления паузы из TIA IS-641-A в октеты для индекса режима 10 (Биты с cn0 по cn37 описаны в документе TIA IS-641-A)

Переданные октеты	MSB	Отображение битов						LSB
1	Индекс первого субвектора LSF cn3 cn2 cn1 cn0				Mode_Index mi(3) mi(2) mi(1) mi(0)			
2	Индекс второго субвектора LSF cn11 cn10 cn9 cn8				Индекс первого субвектора LSF cn7 cn6 cn5 cn4			
3	Индекс третьего субвектора LSF cn19 cn18 cn17			cn16	Индекс второго субвектора LSF cn15 cn14 cn13 cn12			
4	Усиление возбуждения по случайному закону cn27 cn26		Индекс третьего субвектора LSF cn25 cn24 cn23 cn22 cn21 cn20					

**Таблица I.13/Н.245 – Отображение битов-дескрипторов подавления паузы
из TIA IS-641-A в октеты для индекса режима 10
(Биты с cn0 по cn37 описаны в документе TIA IS-641-A)**

Переданные октеты	MSB	Отображение битов						LSB
5	Индекс первого параметра RESC cn35 cn34	Усиление возбуждения по случайному закону cn33 cn32 cn31 cn30 cn29 cn28						
6	Биты заполнения UB UB UB UB UB UB						Индекс второго параметра RESC cn37 cn36	

**Таблица I.14/Н.245 – Отображение битов-дескрипторов подавления паузы
из RCR STD-27H в октетах для индекса режима 11
(Биты с s1 по s35 описаны в документе RCR STD-27H)**

Переданные октеты	MSB	Отображение битов						LSB
1	Индекс первого субвектора LSF s4	Индекс эталонного вектора LSF s3 s2 s1			Mode_Index mi(3) mi(2) mi(1) mi(0)			
2	Индекс второго субвектора LSF s12	Индекс первого субвектора LSF s11 s10 s9 s8 s7 s6 s5						
3	Индекс второго субвектора LSF s20 s19 s18 s17 s16 s15 s14 s13							
4	Индекс третьего субвектора LSF s28 s27 s26 s25 s24 s23 s22 s21							
5	Тип SID t1	Энергия кадра s35 s34 s33 s32 s31 s30					Индекс третьего субвектора LSF s29	
6	Биты заполнения UB UB UB UB UB UB						Тип SID t2	

Определение дополнительных битов-дескрипторов, необходимых для PDC-EFR в таблице I.14:

Тип SID имеет значения {0=POST0, 1=POST1(SID_UPDATE), 2=PRE, 3=POST1_BAD}, где LSB SID_type равен t1, а MSB типа SID равен t2.

Таблица I.15/H.245 – Определение кадра "нет передачи" для индекса режима 15

Передаваемые октеты	MSB				Содержание кадра				LSB			
	Биты заполнения				Mode_Index							
1	UB	UB	UB	UB	mi(3)	mi(2)	mi(1)	mi(0)				

Таблица I.16/H.245 – Субъективное значение битов кодирования речи для AMR12.2

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	23	15	16	17	18
19	20	21	22	24	25	26	27	28	38
141	39	142	40	143	41	144	42	145	43
146	44	147	45	148	46	149	47	97	150
200	48	98	151	201	49	99	152	202	86
136	189	239	87	137	190	240	88	138	191
241	91	194	92	195	93	196	94	197	95
198	29	30	31	32	33	34	35	50	100
153	203	89	139	192	242	51	101	154	204
55	105	158	208	90	140	193	243	59	109
162	212	63	113	166	216	67	117	170	220
36	37	54	53	52	58	57	56	62	61
60	66	65	64	70	69	68	104	103	102
108	107	106	112	111	110	116	115	114	120
119	118	157	156	155	161	160	159	165	164
163	169	168	167	173	172	171	207	206	205
211	210	209	215	214	213	219	218	217	223
222	221	73	72	71	76	75	74	79	78
77	82	81	80	85	84	83	123	122	121
126	125	124	129	128	127	132	131	130	135
134	133	176	175	174	179	178	177	182	181
180	185	184	183	188	187	186	226	225	224
229	228	227	232	231	230	235	234	233	238
237	236	96	199						

Таблица I.17/Н.245 – Субъективное значение битов кодирования речи для AMR10.2

7	6	5	4	3	2	1	0	16	15
14	13	12	11	10	9	8	26	27	28
29	30	31	115	116	117	118	119	120	72
73	161	162	65	68	69	108	111	112	154
157	158	197	200	201	32	33	121	122	74
75	163	164	66	109	155	198	19	23	21
22	18	17	20	24	25	37	36	35	34
80	79	78	77	126	125	124	123	169	168
167	166	70	67	71	113	110	114	159	156
160	202	199	203	76	165	81	82	92	91
93	83	95	85	84	94	101	102	96	104
86	103	87	97	127	128	138	137	139	129
141	131	130	140	147	148	142	150	132	149
133	143	170	171	181	180	182	172	184	174
173	183	190	191	185	193	175	192	176	186
38	39	49	48	50	40	52	42	41	51
58	59	53	61	43	60	44	54	194	179
189	196	177	195	178	187	188	151	136	146
153	134	152	135	144	145	105	90	100	107
88	106	89	98	99	62	47	57	64	45
63	46	55	56						

Таблица I.18/Н.245 – Субъективное значение битов кодирования речи для AMR7.95

8	7	6	5	4	3	2	14	16	9
10	12	13	15	11	17	20	22	24	23
19	18	21	56	88	122	154	57	89	123
155	58	90	124	156	52	84	118	150	53
85	119	151	27	93	28	94	29	95	30
96	31	97	61	127	62	128	63	129	59
91	125	157	32	98	64	130	1	0	25
26	33	99	34	100	65	131	66	132	54
86	120	152	60	92	126	158	55	87	121
153	117	116	115	46	78	112	144	43	75
109	141	40	72	106	138	36	68	102	134
114	149	148	147	146	83	82	81	80	51
50	49	48	47	45	44	42	39	35	79
77	76	74	71	67	113	111	110	108	105
101	145	143	142	140	137	133	41	73	107
139	37	69	103	135	38	70	104	136	

Таблица I.19/Н.245 – Субъективное значение битов кодирования речи для AMR7.4

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	26	87	27
88	28	89	29	90	30	91	51	80	112
141	52	81	113	142	54	83	115	144	55
84	116	145	58	119	59	120	21	22	23
17	18	19	31	60	92	121	56	85	117
146	20	24	25	50	79	111	140	57	86
118	147	49	78	110	139	48	77	53	82
114	143	109	138	47	76	108	137	32	33
61	62	93	94	122	123	41	42	43	44
45	46	70	71	72	73	74	75	102	103
104	105	106	107	131	132	133	134	135	136
34	63	95	124	35	64	96	125	36	65
97	126	37	66	98	127	38	67	99	128
39	68	100	129	40	69	101	130		

Таблица I.20/Н.245 – Субъективное значение битов кодирования речи для AMR6.7

0	1	4	3	5	6	13	7	2	8
9	11	15	12	14	10	28	82	29	83
27	81	26	80	30	84	16	55	109	56
110	31	85	57	111	48	73	102	127	32
86	51	76	105	130	52	77	106	131	58
112	33	87	19	23	53	78	107	132	21
22	18	17	20	24	25	50	75	104	129
47	72	101	126	54	79	108	133	46	71
100	125	128	103	74	49	45	70	99	124
42	67	96	121	39	64	93	118	38	63
92	117	35	60	89	114	34	59	88	113
44	69	98	123	43	68	97	122	41	66
95	120	40	65	94	119	37	62	91	116
36	61	90	115						

Таблица I.21/Н.245 – Субъективное значение битов кодирования речи для AMR5.9

0	1	4	5	3	6	7	2	13	15
8	9	11	12	14	10	16	28	74	29
75	27	73	26	72	30	76	51	97	50
71	96	117	31	77	52	98	49	70	95
116	53	99	32	78	33	79	48	69	94
115	47	68	93	114	46	67	92	113	19
21	23	22	18	17	20	24	111	43	89
110	64	65	44	90	25	45	66	91	112
54	100	40	61	86	107	39	60	85	106
36	57	82	103	35	56	81	102	34	55
80	101	42	63	88	109	41	62	87	108
38	59	84	105	37	58	83	104		

Таблица I.22/Н.245 – Субъективное значение битов кодирования речи для AMR5.15

7	6	5	4	3	2	1	0	15	14
13	12	11	10	9	8	23	24	25	26
27	46	65	84	45	44	43	64	63	62
83	82	81	102	101	100	42	61	80	99
28	47	66	85	18	41	60	79	98	29
48	67	17	20	22	40	59	78	97	21
30	49	68	86	19	16	87	39	38	58
57	77	35	54	73	92	76	96	95	36
55	74	93	32	51	33	52	70	71	89
90	31	50	69	88	37	56	75	94	34
53	72	91							

Таблица I.23/Н.245 – Субъективное значение битов кодирования речи для AMR4.75

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	23	24	25	26
27	28	48	49	61	62	82	83	47	46
45	44	81	80	79	78	17	18	20	22
77	76	75	74	29	30	43	42	41	40
38	39	16	19	21	50	51	59	60	63
64	72	73	84	85	93	94	32	33	35
36	53	54	56	57	66	67	69	70	87
88	90	91	34	55	68	89	37	58	71
92	31	52	65	86					

Приложение J

Определения, относящиеся к кодексу звука ACELP стандарта TDMA (множественный доступ с временным разделением)

В таблице J.1 определен идентификатор возможности для кодека звука ACELP TIA/EIA 136 US1 [75]. В таблицах с J.2 по J.4 заданы относящиеся к возможности параметры. Этот кодек используется в базовых станциях и в мобильных телефонах систем сотовой связи стандарта TDMA и PCS. Технические спецификации этого кодека представлены в стандарте 136 TIA/EIA, часть 410. Этот стандарт опубликован Североамериканской ассоциацией промышленности средств связи (TIA) и утвержден Американским национальным институтом стандартов (ANSI).

Таблица J.1/Н.245 – Идентификатор возможности TIA/EIA 136 ACELP

Название возможности	TIA/EIA 136 ACELP Vocoder
Класс возможности	Аудиокодек
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) acelp (2)}
maxBitRate	Это поле должно быть включено.
NonCollapsingRaw	Это поле не должно быть включено.
transport	Это поле не должно быть включено.

Таблица J.2/Н.245 – Параметр "maxAl-sduFrames" возможности TIA/EIA 136 ACELP

Название параметра	TIA/EIA maxAl-sduFrames
Описание параметра	Это свертываемый универсальный параметр Он определяет максимальное количество аудиокадров в AL-SDU
Значение идентификатора параметра	0
Статус параметра	Должен присутствовать при обмене возможностями и сигнализации логического канала. Должен отсутствовать при запросе режима
Тип параметра	UnsignedMin
Заменяет	–

**Таблица J.3/Н.245 – Параметр возможности "Comfort Noise" (комфортный шум)
TIA/EIA 136 ACELP**

Название параметра	ComfortNoise
Описание параметра	Это свертываемый универсальный параметр. Он определяет использование комфортного шума TIA/EIA 136 (IS-641) в запросе режима. Этот параметр должен использоваться только при запросах режима, но не при обмене возможностями, потому что эта возможность является обязательной.
Значение идентификатора параметра	1
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Таблица J.4/H.245 – Параметр "Scrambled" (скремблировано) возможности TIA/EIA 136 ACELP

Название параметра	Scrambled
Описание параметра	Это свертываемый универсальный параметр. Он определяет использование шифрования при запросе режима. Этот параметр должен использоваться только при запросах режима, но не при обмене возможностями, потому что эта возможность является обязательной.
Значение идентификатора параметра	2
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Приложение К

Определения, относящиеся к речевому кодеку US1 стандарта TDMA

В таблице К.1 определен идентификатор возможности речевых кодеков TIA/EIA 136 US1 [76]. В таблицах с К.2 по К.4 заданы параметры, относящиеся к этой возможности. Этот кодек используется в базовых станциях и мобильных телефонах систем сотовой связи стандарта TDMA и PCS. Технические спецификации этого кодека приведены в стандарте TIA/EIA 136, часть 430. Этот стандарт опубликован Североамериканской ассоциацией промышленности средств связи (TIA) и утвержден Американским национальным институтом стандартов (ANSI).

Таблица К.1/H.245 – Идентификатор возможности TIA/EIA 136 US1

Название возможности	TIA/EIA 136 US1 Vocoder
Класс возможности	Аудиокодек
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) us1 (3)}
maxBitRate	Это поле должно быть включено.
NonCollapsingRaw	Это поле не должно быть включено.
transport	Это поле не должно быть включено.

Таблица К.2/H.245 – Параметр "maxAl-sduFrames" возможности TIA/EIA 136 US1

Название параметра	maxAl-sduFrames
Описание параметра	Это свертываемый универсальный параметр. Он определяет максимальное количество аудиокадров в AL-SDU.
Значение идентификатора параметра	0
Статус параметра	Должен присутствовать при обмене возможностями и сигнализации логического канала. Должен отсутствовать при запросе режима.
Тип параметра	UnsignedMin
Заменяет	–

Таблица К.3/Н.245 – Параметр "Comfort Noise" возможности TIA/EIA 136 US1

Название параметра	comfortNoise
Описание параметра	Это свертываемый универсальный параметр. Он определяет использование комфортного шума при запросе режима. Этот параметр должен использоваться только при запросах режима, но не при обмене возможностями, потому что эта возможность является обязательной.
Значение идентификатора параметра	1
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Таблица К.4/Н.245 – Параметр "Scrambled" (скремблировано) возможности TIA/EIA 136 US1

Название параметра	scrambled
Описание параметра	Это свертываемый универсальный параметр. Он определяет, что при запросе режима должно использоваться шифрование. Этот параметр должен использоваться только при запросах режима, но не при обмене возможностями, потому что эта возможность является обязательной.
Значение идентификатора параметра	2
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Приложение L

Определения, относящиеся к голосовому кодексу EVRC стандарта CDMA (множественный доступ с кодовым разделением каналов)

В таблице L.1 определен идентификатор возможности усовершенствованного кодека с переменной скоростью (EVRC) для TIA/EIA IS-127, который используется в базовых станциях и мобильных телефонах систем сотовой связи TIA/EIA IS-95, CDMA и PCS. Полное техническое описание и подробные спецификации данного кодека представлены в стандарте TIA/EIA IS-127, который опубликован Североамериканской ассоциацией промышленности средств связи (TIA). В таблицах с L.2 по L.4 определяются параметры, относящиеся к этой возможности.

Таблица L.1/Н.245 – Идентификатор возможности EVRC CDMA

Название возможности	TIA/EIA IS-127 CDMA EVRC
Класс возможности	Аудиокодек
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) is127evrc (4)}
maxBitRate	Это поле должно быть включено.
NonCollapsingRaw	Это поле не должно быть включено.
transport:	Это поле не должно быть включено.

Таблица L.2/Н.245 – Параметр maxAl-sduFrames возможности TIA/EIA IS-127 CDMA EVRC

Название параметра	maxAl-sduFrames
Описание параметра	Это свертываемый универсальный параметр. Он определяет максимальное количество аудиокадров в AL-SDU.
Значение идентификатора параметра	0
Статус параметра	Должен присутствовать при обмене возможностями и сигнализации логического канала. Должен отсутствовать при запросе режима.
Тип параметра	UnsignedMin
Заменяет	–

Таблица L.3/Н.245 – Параметр EVRC Bit-Rate возможности CDMA EVRC

Название параметра	EVRCRate
Описание параметра	Это несвертываемый универсальный параметр. Он определяет режим скорости передачи данных на выходе вокодера. Этот параметр должен использоваться при запросах режима: 1 = полная скорость; 2 = половинная скорость; 3 = одна восьмая часть скорости; 4 = режим останова (blanked mode).
Значение идентификатора параметра	1
Статус параметра	Необязательный
Тип параметра	UnsignedMin
Заменяет	–

Таблица L.4/Н.245 – Параметр "Scrambled" (скремблировано) возможности CDMA EVRC

Название параметра	Scrambled
Описание параметра	Это свертываемый универсальный параметр. Определяет, что при запросе режима должно использоваться скремблирование.
Значение идентификатора параметра	2
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Приложение М

Определения, относящиеся к ИСО/МЭК 13818-7 и МСЭ-Р BS.1196

Таблица М.1 определяет идентификатор возможности для ISO/IEC (ИСО/МЭК) 13818-7. Таблица М.2 определяет параметры, относящиеся к соответствующей возможности.

Таблица М.3 определяет идентификатор возможности для МСЭ-Р BS 1196. Отсутствуют параметры, соответствующие этой возможности.

Таблица М.1/Н.245 – Идентификатор возможности ИСО/МЭК 13818-7

Название возможности	ISO/IEC 13818-7
Класс возможности	Аудиокодек
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) ISO/IEC 13818-7 (5)}
maxBitRate	Это поле должно быть включено.
NonCollapsingRaw	Это поле не должно быть включено.
transport	Это поле не должно быть включено

Таблица М.2/Н.245 – Параметр "Profile and Level" (профиль и уровень) для возможности ISO/IEC 13818-7

Название параметра	profileAndLevel
Описание параметра	Это несвертываемый универсальный параметр. profileAndLevel показывает возможность обработки определенных профилей в сочетании с уровнем согласно стандарту ИСО/МЭК 13818-7.
Значение идентификатора параметра	0
Статус параметра	Обязательный
Тип параметра	unsignedMax. Должен находиться в диапазоне 0..255.
Заменяет	–

Таблица М.3/Н.245 – Идентификатор возможности ITU-R BS.1196

Название возможности	ITU-R BS.1196
Класс возможности	Аудиокодек
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) audio (1) itu-r bs.1196 (6)}
maxBitRate	Это поле должно быть включено.
NonCollapsingRaw	Это поле не должно быть включено.
transport	Это поле не должно быть включено.

Добавление I

Обзор синтаксиса языка АСН.1

I.1 Введение в язык АСН.1

Абстрактная синтаксическая нотация версии один (АСН.1) представляет собой язык спецификации данных. Первоначально она была стандартизирована в виде Рекомендации X.409 как часть Рекомендаций по электронной почте серии X.400. Свое развитие этот язык получил в Рекомендации X.208, а в самое последнее время – в виде Рекомендации X.680. Язык АСН.1 позволяет однозначно определять сложные структуры данных, в том числе структуры данных с полями переменной длины, необязательными полями и рекурсией.

В перечисленных выше Рекомендациях рассматривается только синтаксис и семантика языка АСН.1. Двоичному кодированию структур данных посвящены другие Рекомендации, из которых особое

значение имеют Рекомендация X.690 (базовые правила кодирования, или BER) и Рекомендация X.691 (правила пакетного кодирования, или PER). Базовые правила кодирования обеспечивают дешифровку данных в системах, в которые включена только общая информация о языке АСН.1, но где нет подробной спецификации процесса формирования данных. Иными словами, типы данных кодируются вместе со значениями данных. Правила пакетного кодирования значительно более эффективны, поскольку кодируются только значения данных и кодирование разработано с очень малой избыточностью. Этот метод можно использовать в тех случаях, когда как отправитель, так и получатель предполагают, что будет строго соблюдаться известная структура данных.

Выполнение положений настоящей Рекомендации основано на использовании правил пакетного кодирования. Поскольку обе стороны вызова знают, что сообщения будут соответствовать спецификации, данной в Рекомендации H.245, нет необходимости кодировать его определения в сообщениях. Для упрощения декодирования используется вариант PER с выравниванием. В этом варианте поля, требующие восемь или более битов, должны выравниваться на границах октетов и должны занимать целое число октетов. Выравнивание осуществляется путем дополнения данных нулями перед большими полями.

I.2 Базовые типы данных в АСН.1

Простейшим типом данных являются BOOLEAN (булевы) данные, которые представляют значения FALSE (Ложь) и TRUE (Истина). Они кодируются одним битом как 0 или 1, соответственно. Например, `segmentableFlag` BOOLEAN кодируется следующим образом:

Значение	Кодирование
FALSE (ложь)	0
TRUE (истина)	1

Самый фундаментальный тип данных – это INTEGER (Целое число), который представляет собой целые числа. Данные типа INTEGER могут быть не ограничены по величине:

`bitRate` INTEGER

либо они могут быть ограничены диапазоном значений, например:

`maximumA12SDUSize` INTEGER (0..65535)

Ограниченные целые числа кодируются различным образом в зависимости от размера диапазона. Предположим, что N – это количество целых чисел в данном диапазоне, то есть верхний предел минус нижний предел плюс один. В зависимости от N ограниченное целое будет кодироваться одним из следующих пяти способов:

N	Кодирование
1	Никаких битов не требуется
2-255	Невыровненное поле с числом битов от 1 до 8
256	Выворненное поле длиной 8 бит
257-65536	Выворненное поле длиной 16 бит
больше	Как минимальное количество выровненных октетов, которым предшествуют количества октетов, закодированных согласно приведенному выше кодированию

Во всех случаях фактически используемое число является значением, которое подлежит кодированию, минус нижний предел диапазона. В этих примерах "заполнение" представляет собой от нуля до семи нулевых битов, которые добавляются к закодированной величине так, чтобы следующее поле началось на границе восьми битов.

`firstGOV` INTEGER (0..17)

Значение	Кодирование
0	00000
3	00011

h233IVResponseTime **INTEGER (0..255)**

Значение	Кодирование
3	заполнение 00000011
254	заполнение 11111110

skew **INTEGER (0..4095)**

Значение	Кодирование
3	заполнение 00000000 00000011
4095	заполнение 00001111 11111111

Целые числа без ограничения (дополнение до 2), которые могут быть представлены 127 или меньшим количеством октетов, кодируются минимально необходимым количеством октетов. Количество октетов (длина) кодируется как выровненный октет, который предшествует самому числу. Например:

-1	заполнение 00000001 11111111
0	заполнение 00000001 00000000
128	заполнение 00000010 00000000 10000000
1000000	заполнение 00000011 00001111 01000010 01000000

АСН.1 поддерживает различные типы строковых данных. Они представляют собой списки битов, октетов или других коротких типов данных переменной длины. Обычно они кодируются так, что сначала указывается длина, а потом сами данные. Длина может кодироваться как неограниченное целое или ограниченное целое число, если задан размер строки SIZE. Например:

data **ОКТЕТ STRING**

Поскольку длина строки октетов не ограничена, она должна кодироваться как *полуограниченное целое число* (имеющее нижнюю границу, но не имеющее верхней границы). Сначала данные заполняются таким образом, чтобы кодирование было выровнено. Остальное кодирование осуществляется следующим образом:

Длина	Кодирование
0–127	Длина 8 бит, затем следуют данные
128–16К – 1	Длина 16 бит с набором MSB, затем следуют данные
16К – 32К – 1	11000001, 16К октетов данных, затем кодируются остальные данные
32К – 48К – 1	11000010, 32К октетов данных, затем кодируются остальные данные
48К – 64К – 1	11000011, 48К октетов данных, затем кодируются остальные данные
64К или больше	11000100, 64К октетов данных, затем кодируются остальные данные

Этот метод называется "фрагментацией". Обратите внимание на то, что если длина кратна 16 К, то представление данных заканчивается октетом с нулевым значением, который обозначает строку нулевой длины.

1.3 Агрегированные типы данных

АСН.1 включает несколько агрегированных или контейнерных типов данных, концепция которых аналогична типам данных "union", "struct" и "array" в языке программирования С. Это соответственно CHOICE (Выбор), SEQUENCE (Последовательность) и SEQUENCE OF. Во всех случаях они кодируются посредством нескольких битов, определяющих контейнер, за которыми следует закодированное обычным образом содержимое контейнера.

Тип данных CHOICE используется для выбора одной из групп типа данных. Например:

```
VideoCapability ::= CHOICE
{
    nonStandard NonStandardParameter,
    h261VideoCapability H261VideoCapability,
    h262VideoCapability H262VideoCapability,
    h263VideoCapability H263VideoCapability,
    is11172VideoCapability IS11172VideoCapability,
    ...
}
```

Каждой выбранной группе присваивается порядковый номер, начиная с нуля. Номер действительного выбора кодируется как ограниченное integer (целое число). За номером следует закодированное представление фактически выбранной группы или ничего, если выбран нуль (NULL). Если присутствует маркер расширения (как описано выше), то порядковому номеру предшествует бит, который равен нулю, если фактический выбор произведен из первоначального списка.

Тип данных SEQUENCE представляет собой просто сгруппированные несходные типы данных. Отдельные элементы этой последовательности могут быть OPTIONAL (Необязательными). Кодирование производится очень просто. Если присутствует маркер расширения, то первый бит указывает на наличие дополнительных элементов. За ним следует последовательность битов, по одному на каждый необязательный элемент, который показывает, есть ли такие данные. За ней следуют закодированные компоненты последовательности. Например:

```
H261VideoCapability ::= SEQUENCE
{
    qcifMPI INTEGER (1..4) OPTIONAL, -- ед. изм. 1/29,97 Гц
    cifMPI INTEGER (1..4) OPTIONAL, -- ед. изм. 1/29,97 Гц
    temporalSpatialTradeOffCapability BOOLEAN,
    ...
}
```

При кодировании один бит обозначает маркер расширения, два бита – необязательные поля, два бита уходит на каждое присутствующее необязательное поле, один бит – булевы данные, а затем данные любого расширения. Обратите внимание на то, что в этой последовательности не используются биты дополнения для выравнивания по октетам.

Типы данных SEQUENCE OF и SET OF описывают совокупность одинаковых компонентов (массив). Тип данных SEQUENCE OF показывает, что порядок элементов является существенным, в типе данных SET OF порядок элементов произвольный. Правила пакетного кодирования одинаковы для обоих типов данных.

Эти типы могут иметь ограничения по размеру SIZE или могут состоять из неограниченного количества элементов. Если количество элементов известно заранее и оно менее 64 К, то оно не кодируется. В противном случае фактическое количество компонентов кодируется как данные ограниченной или полуограниченной длины. За этим числом следуют данные в закодированном виде. Если длина составляет не менее 16 К и она закодирована, то перечень данных разбивается на фрагменты, подобные строке октетов. В этом случае разбиение на фрагменты производится после некоторого количества полей компонентов (16 К, 32 К и т. д.), а не после некоторого количества октетов.

1.4 Тип данных "идентификатор объекта"

Обычно в спецификации АСН.1 дается тип значения, поэтому необходимо кодировать и передавать только сами данные. Однако иногда желательно закодировать не только значение, но и тип данных. Например, **protocolIdentifier** содержит

```
protocolIdentifier OBJECT IDENTIFIER,
-- должно быть установлено значение
-- {itu-t (0) recommendation (0) h (8) 245 version (0) 1}
```

Все целые числа в фигурных скобках {} закодированы: как целые внутри круглых скобок (), так целые вне этих скобок. В данном примере подлежат кодированию целые числа 0, 0, 8, 245, 0, 1.

Кодирование осуществляется следующим образом: перед данными, закодированными согласно правилам BER (X.690), стоит длина этого кодирования в октетах. Длина кодируется как полуограниченное целое число (см. приведенный выше пример со строкой октетов). Ниже показано, как выполняется кодирование.

Первый октет показывает длину следующего за ним результата кодировки.

Первые два компонента идентификатора объекта объединяются вместе как $40 \times \text{первый компонент} + \text{второй компонент}$, в данном случае $40 \times 0 + 0 = 0$. Остальные компоненты кодируются обычным образом. Каждый компонент кодируется как последовательность октетов, причем первый бит каждого октета показывает, имеются ли следующие октеты. Таким образом:

0 → 0000 0000

8 → 0000 1000

в то время как 245, будучи больше 127, становится 1000 0001 0111 0101.

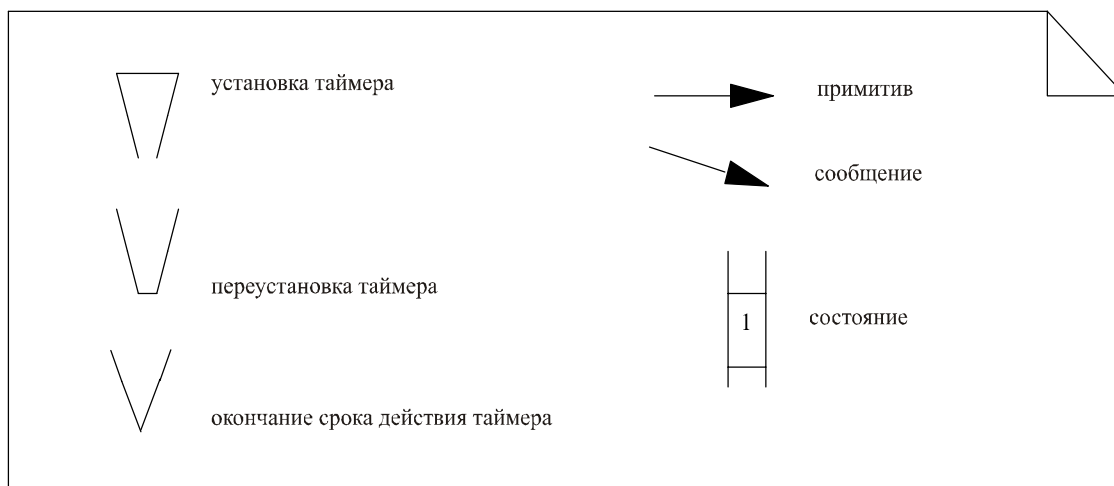
Таким образом, весь результат кодирования в шестнадцатеричном виде должен состоять из семи октетов 06000881 750001.

Добавление II

Примеры процедур H.245

II.1 Введение

В настоящем Добавлении приведены примеры процедур, определенных в Приложении С. На рисунке II.1-1 показаны обозначения, используемые в диаграммах настоящего Приложения.



H.245_FII.1-1

Рисунок II.1-1/H.245 – Обозначения на рисунках

II.2 Объект сигнализации определения отношения "ведущий-ведомый"

На рисунках с II.2-1 по II.2-10 сообщения представлены сокращенными названиями, которые приведены в таблице II.2-1.

Таблица II.2-1/Н.245 – Сокращенные названия при определении отношения "ведущий-ведомый"

Сообщение	Название в примерах
MasterSlaveDetermination	MSD
MasterSlaveDeterminationAck	MSDAck
MasterSlaveDeterminationReject	MSDReject
MasterSlaveDeterminationRelease	MSDRelease

На рисунках с II.2-1 по II.2-10 состояния IDLE, OUTGOING AWAITING RESPONSE и INCOMING AWAITING RESPONSE помечены цифрами "0", "1" и "2", соответственно.

На следующих рисунках значение параметра, относящееся к примитивам DETERMINE.indication и DETERMINE.confirm – это значение параметра TYPE ("тип"). Значение поля, относящееся к сообщению MasterSlaveDeterminationAck, совпадает со значением поля decision ("решение").

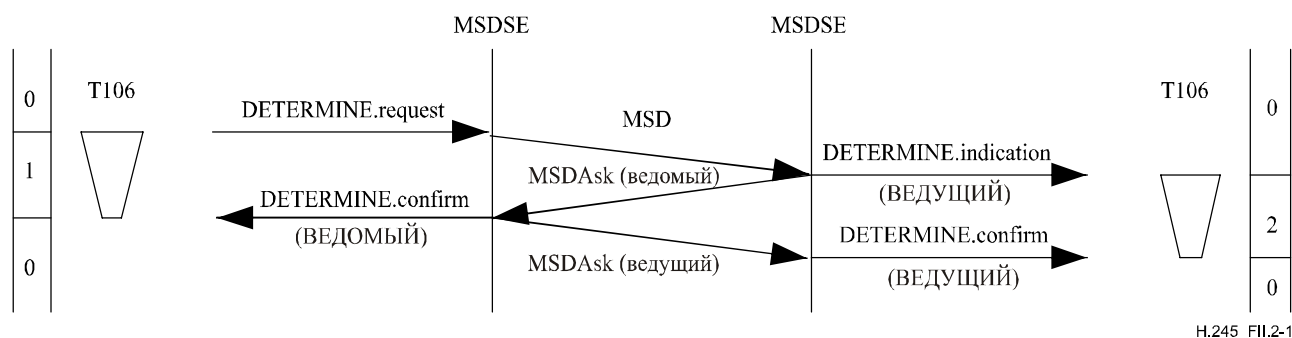


Рисунок II.2-1/Н.245 – Определение отношения "ведущий-ведомый" – Ведущий терминал на удаленном MSDSE

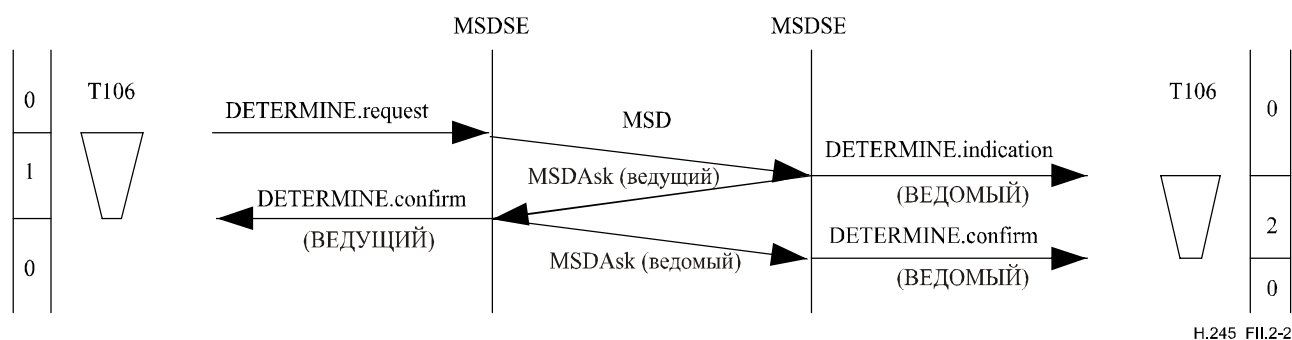


Рисунок II.2-2/Н.245 – Определение отношения "ведущий-ведомый" – Ведомый терминал на удаленном MSDSE

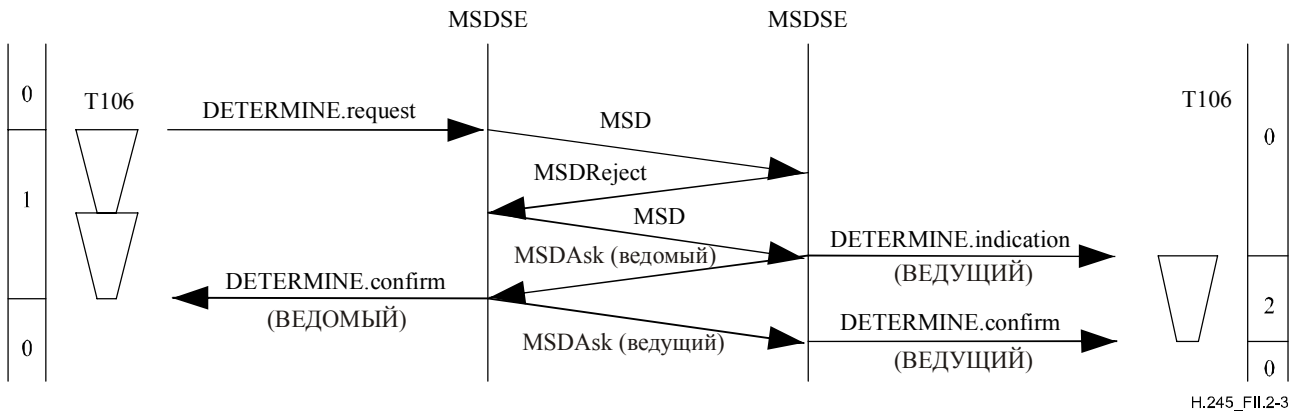


Рисунок П.2-3/Н.245 – Определение отношения "ведущий-ведомый" – Первая попытка привела к неопределенному результату. Вторая попытка была успешной

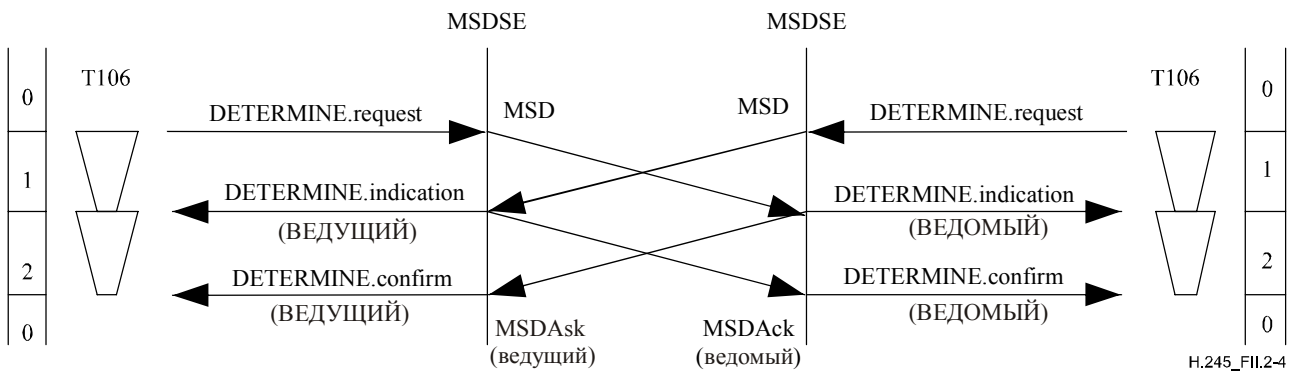


Рисунок П.2-4/Н.245 – Определение отношения "ведущий-ведомый" – Одновременное определение

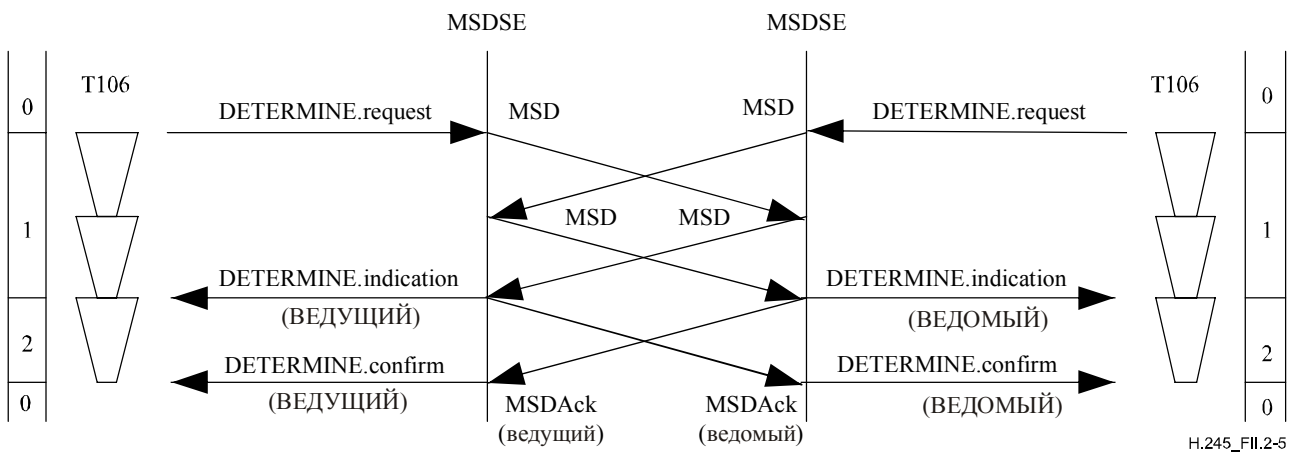


Рисунок П.2-5/Н.245 – Определение отношения "ведущий-ведомый" – Одновременное определение, но первая попытка принесла неопределенный результат

На рисунке П.2-6 показан случай, когда окончился срок действия локального таймера T106. Свой статус знает только терминал, находящийся справа. Терминал, находящийся справа, способен принимать новые команды, но не может ничего требовать у другого терминала, который полагается на знание результата определения статуса терминала. Терминал, находящийся слева, не может ни принимать, ни организовывать новые процедуры. Следует инициировать вторую процедуру определения статуса.

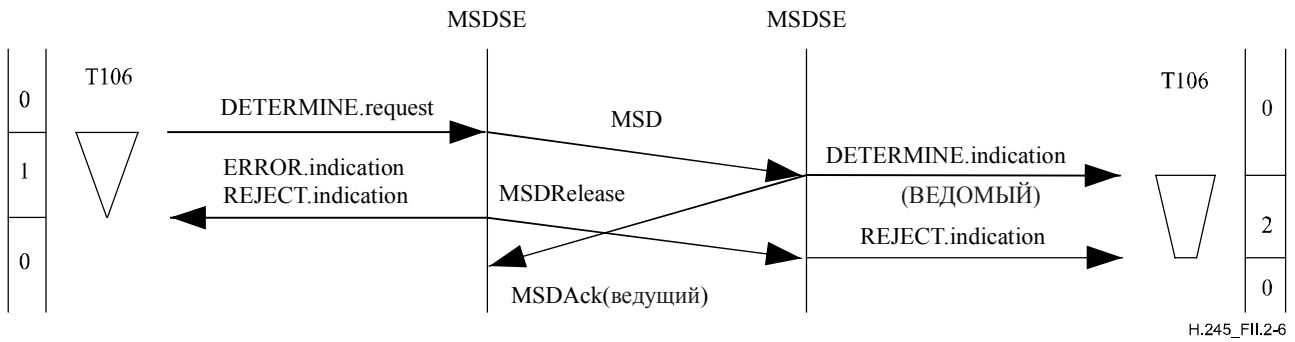


Рисунок П.2-6/Н.245 – Определение отношения "ведущий-ведомый" – Окончание срока действия локального таймера T106 в случае, когда ведомый терминал находится на удаленном конце

На рисунке П.2-7 показан случай, когда срок действия удаленного таймера T106 окончился при состоянии INCOMING AWAITING ACKNOWLEDGEMENT. Оба терминала знают свои статусы. Терминал, расположенный слева, может получать и посылать команды. Однако удаленный терминал не знает, готов ли локальный терминал к приему, и он не может выдавать команды, основанные на результате определения статуса. Следует инициировать вторую процедура определения статуса.

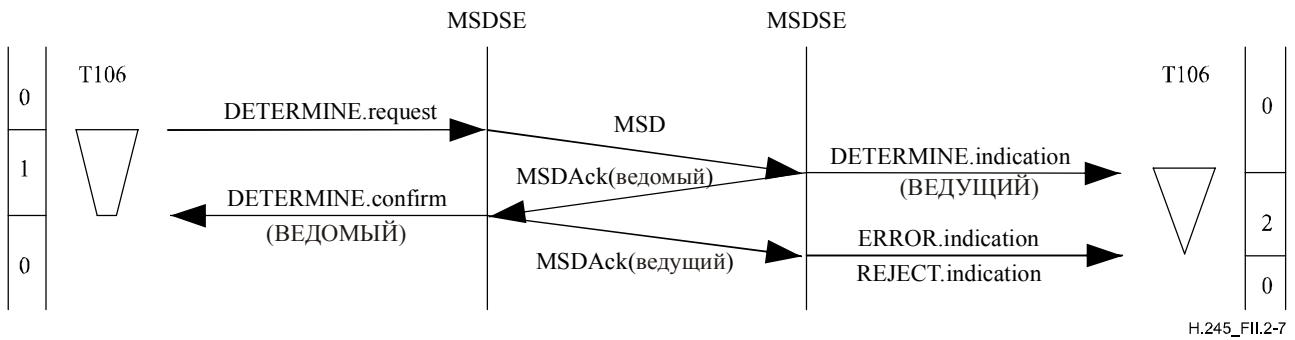


Рисунок П.2-7/Н.245 – Определение отношения "ведущий-ведомый" – Окончание срока действия удаленного таймера T106 в случае, когда ведущий терминал находится на удаленном конце

На рисунке П.2-8 показан случай, когда срок действия удаленного таймера T106 окончился при состоянии OUTGOING AWAITING ACKNOWLEDGEMENT во время процедуры одновременного определения. Оба терминала знают свой статус. Терминал, находящийся справа, может получать и посылать команды. Однако терминал, находящийся слева, не знает, готов ли другой терминал к приему, и поэтому он не может выдавать команды, основанные на результате определения статуса. Он может получать такие команды. Следует инициировать вторую процедуру определения статуса.

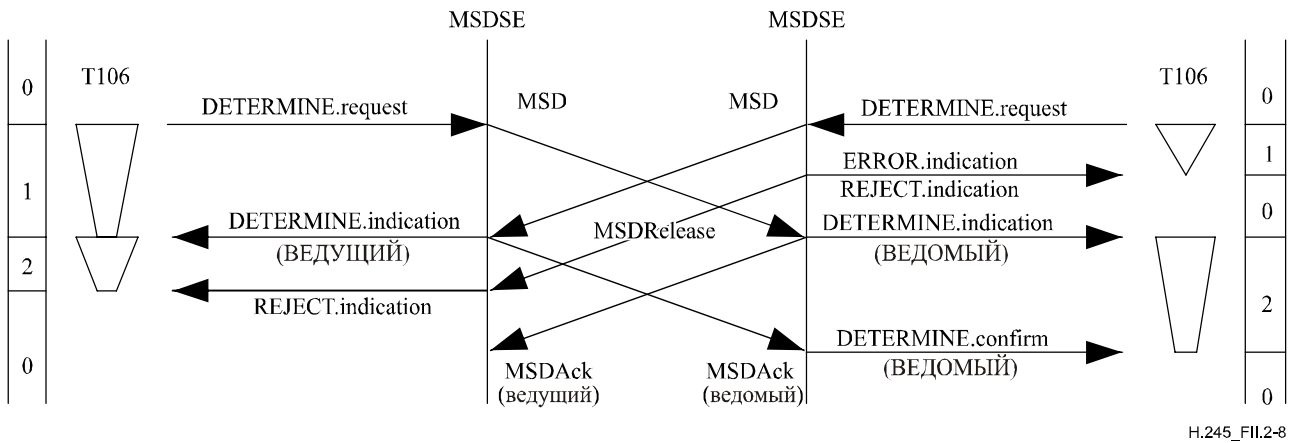


Рисунок П.2-8/Н.245 – Определение отношения "ведущий-ведомый" – Процедуры одновременного определения этого отношения при окончании срока действия таймера T106 в ведомом терминале

На рисунке П.2-9 показан случай, когда срок действия удаленного таймера T106 окончился в течение состояния INCOMING AWAITING ACKNOWLEDGEMENT во время процедуры одновременного определения. Оба терминала знают свой статус. Терминал, находящийся слева, может получать и посылать команды. Однако терминал, находящийся справа, не знает, готов ли другой терминал к приему, и поэтому он не может выдавать команды, основанные на результате определения статуса. Он может получать такие команды. Следует инициализировать вторую процедуру определения статуса.

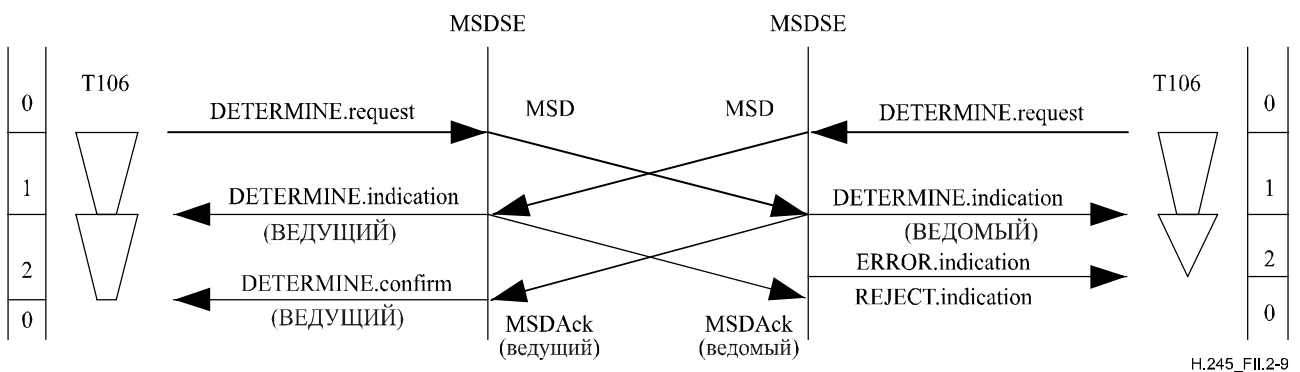
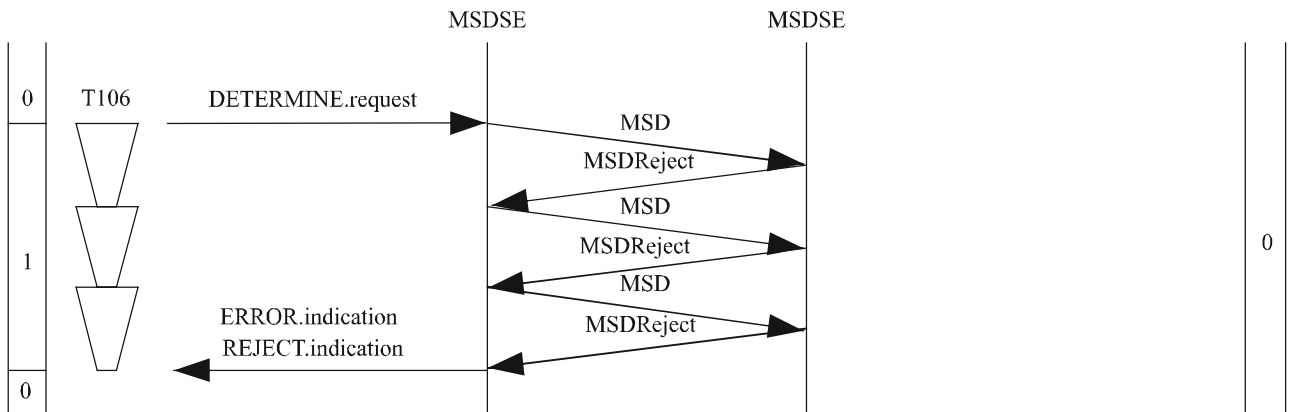


Рисунок П.2-9/Н.245 – Определение отношения "ведущий-ведомый" – Процедуры одновременного определения отношения при окончании срока действия таймера T106 во время события INCOMING AWAITING ACKNOWLEDGEMENT

На рисунке П.2-10 показан случай, когда неопределенный результат был получен N100 раз. В этом случае N100 = 3.

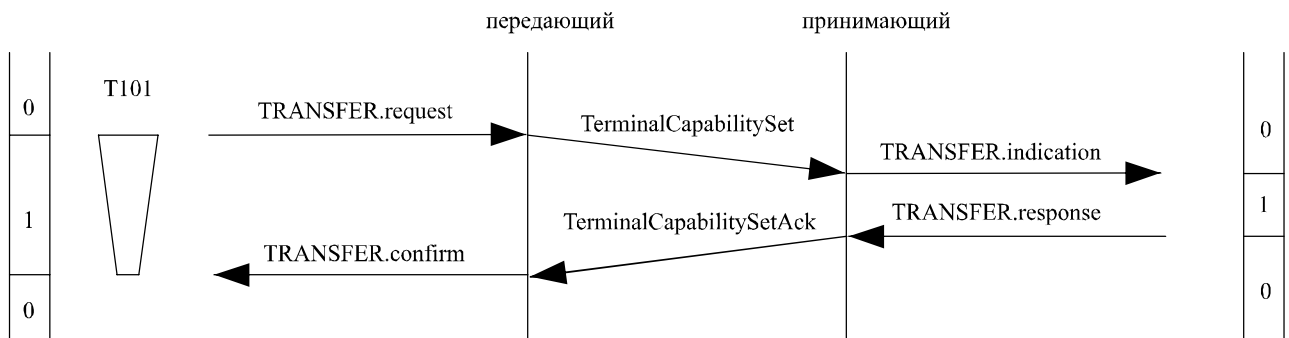


H.245_FII.2-10

Рисунок П.2-10/Н.245 – Определение отношения "ведущий-ведомый" – Неопределенный результат при N100 = 3

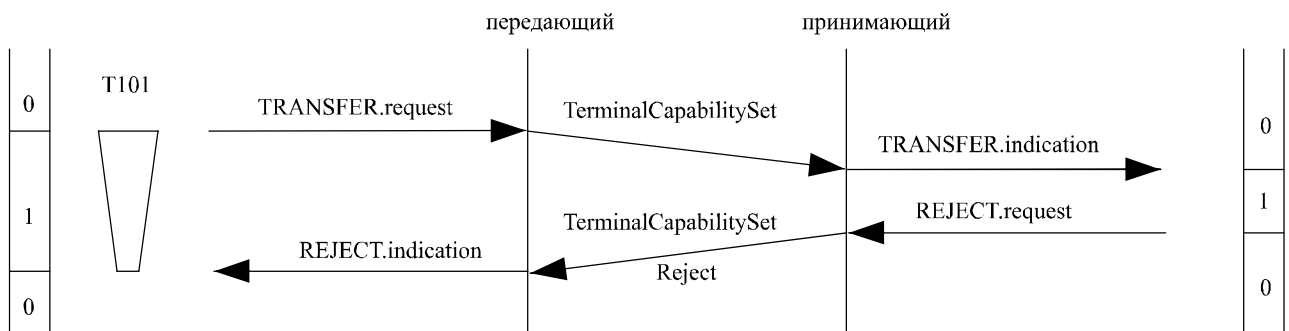
П.3 Объект сигнализации обмена возможностями

На рисунках с П.3-1 по П.3-4 показаны процедуры CESE. Состояния IDLE и AWAITING RESPONSE помечены цифрами "0" и "1", соответственно.



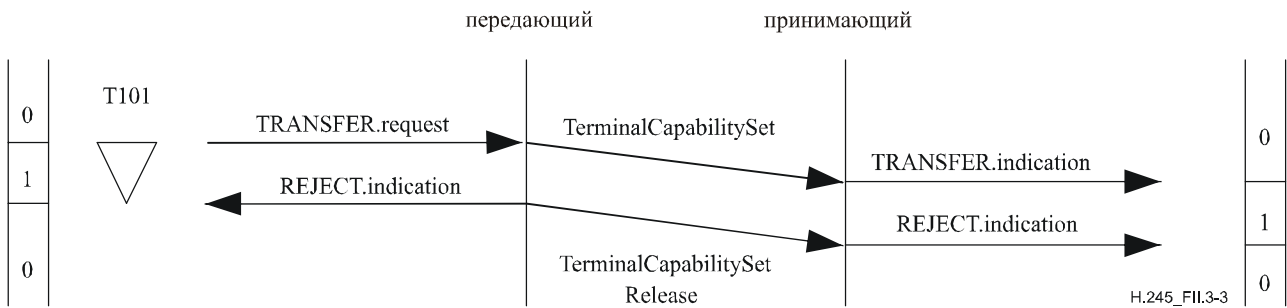
H.245_FII.3-1

Рисунок П.3-1/Н.245 – Обмен возможностями при подтверждении, полученном от пользователя равноправного принимающего CESE



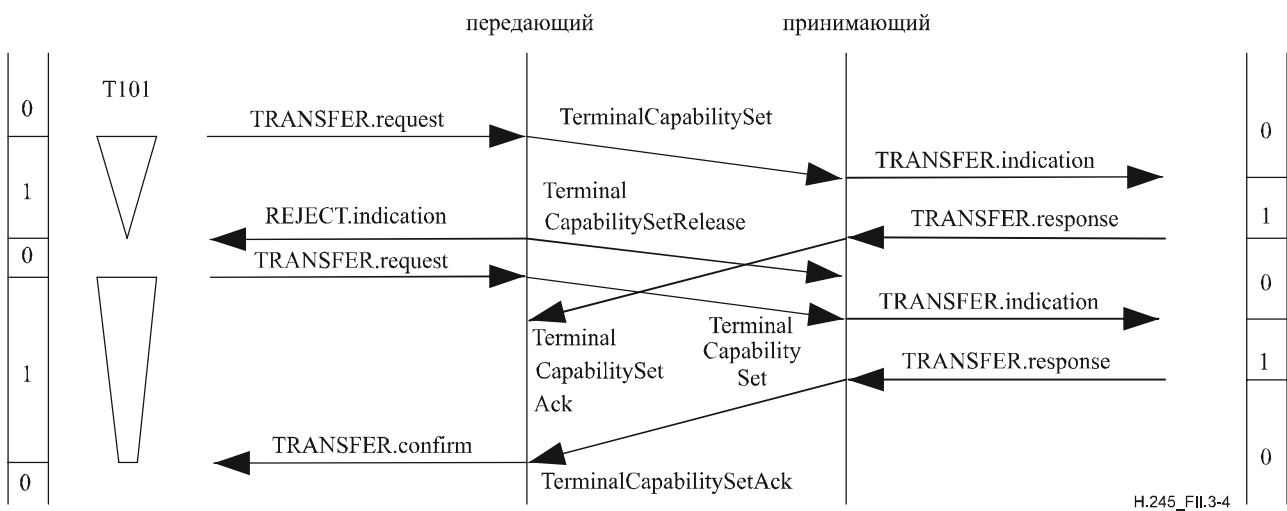
H.245_FII.3-2

Рисунок П.3-2/Н.245 – Обмен возможностями при отказе подтверждения со стороны пользователя равноправного принимающего CESE



ПРИМЕЧАНИЕ. – Сообщение TerminalCapabilitySetRelease приходит на принимающий CESE до ответа от пользователя принимающего CESE

Рисунок П.3-3/Н.245 – Обмен возможностями при окончании срока действия таймера T101



ПРИМЕЧАНИЕ. – Сообщение TerminalCapabilitySetRelease приходит на принимающий CESE после ответа от пользователя принимающего CESE. На передающем CESE Сообщение TerminalCapabilitySetAck, отправленное в ответ на первое сообщение TerminalCapabilitySet, игнорируется. Только вторая попытка обмена возможностями проходит успешно.

Рисунок П.3-4/Н.245 – Обмен возможностями при окончании срока действия таймера T101, после чего следует второй обмен возможностями

П.4 Объект сигнализации в логическом канале

На рисунках с П.4-1 по П.4-7 показаны процедуры LCSE. Состояния передающего LCSE RELEASED, AWAITING ESTABLISHMENT, ESTABLISHED и AWAITING RELEASE помечены цифрами "0", "1", "2" и "3" соответственно. Состояния принимающего LCSE RELEASED, AWAITING ESTABLISHMENT и ESTABLISHED помечены цифрами "0", "1" и "2", соответственно.

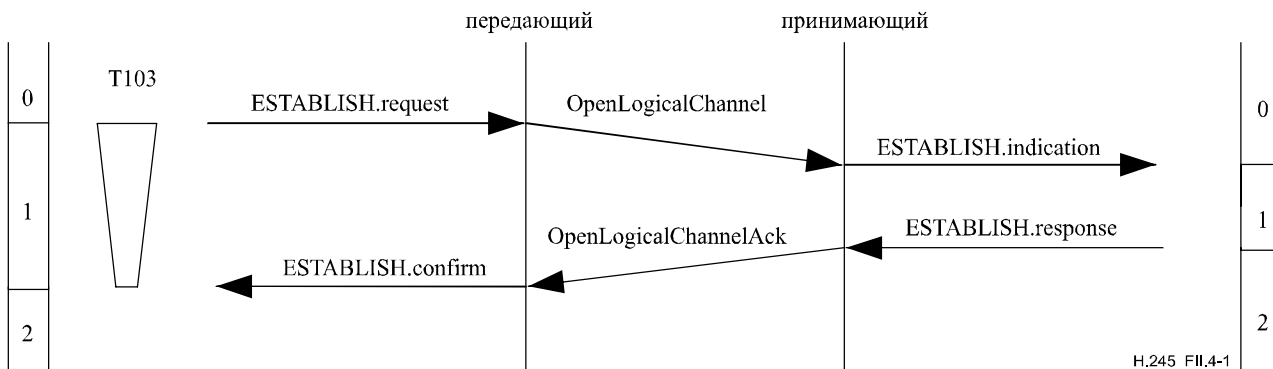


Рисунок П.4-1/Н.245 – Организация логического канала

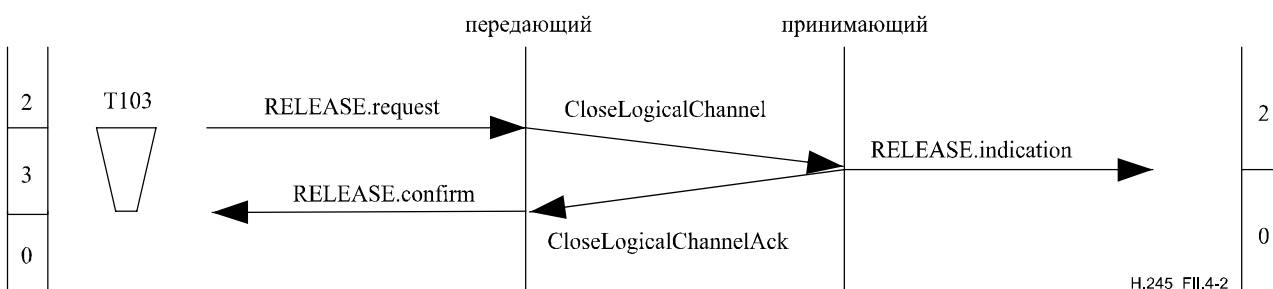


Рисунок П.4-2/Н.245 – Сброс логического канала

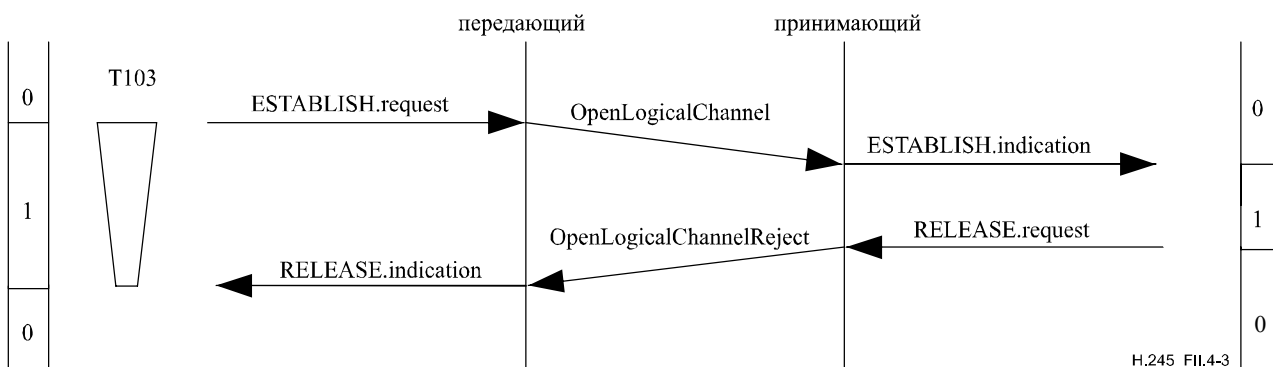


Рисунок П.4-3/Н.245 – Отказ пользователя равноправного LCSE организовать логический канал

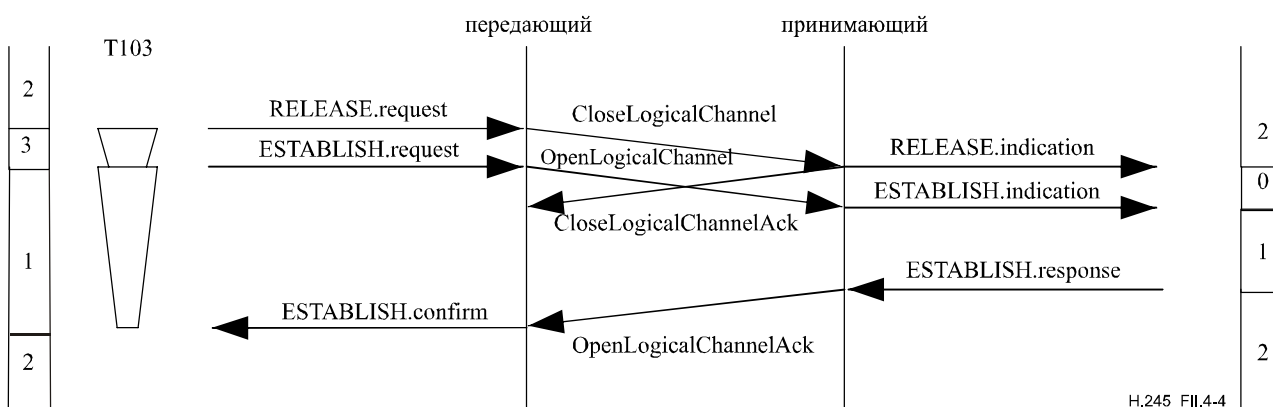


Рисунок П.4-4/Н.245 – Сброс логического канала, после чего следует его немедленная повторная организация

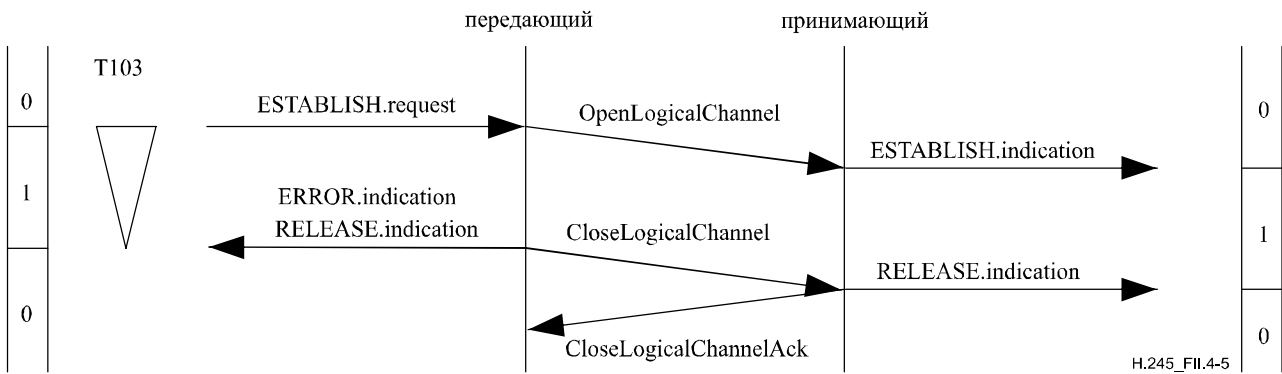
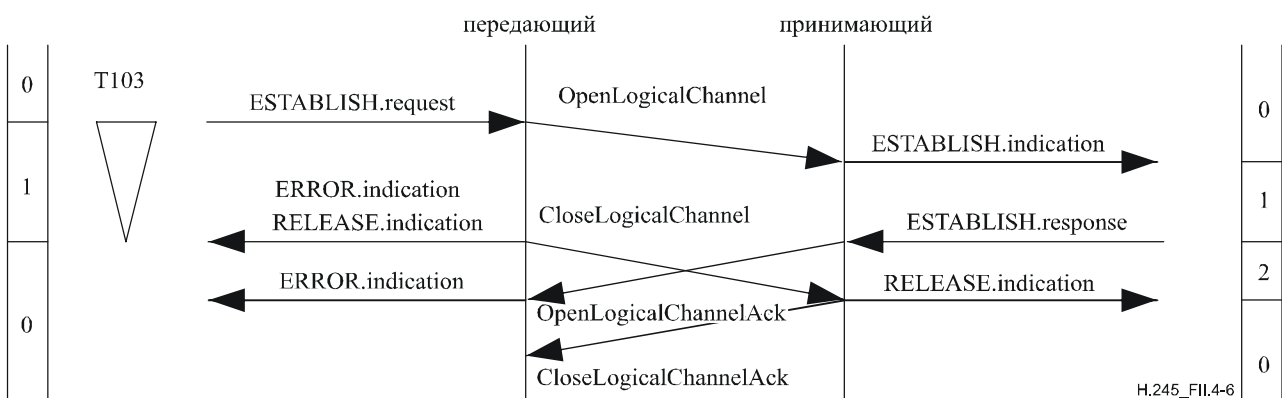


Рисунок П.4-5/Н.245 – Запрос на организацию логического канала при окончании срока действия таймера T103 из-за отсутствия своевременного ответа от пользователя равноправного принимающего LCSE



ПРИМЕЧАНИЕ. – Срок действия таймера T103 окончился после передачи сообщения OpenLogicalChannelAck на принимающем LCSE, но перед получением сообщения OpenLogicalChannelAck на передающем LCSE.

Рисунок П.4-6/Н.245 – Запрос на организацию логического канала при окончании срока действия таймера T103

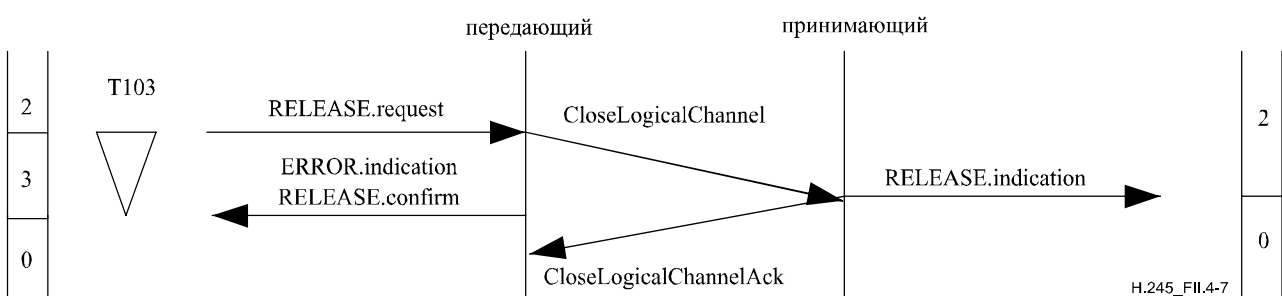


Рисунок П.4-7/Н.245 – Запрос на сброс логического канала при окончании срока действия таймера T1033

II.5 Объект сигнализации закрытия логического канала

На рисунках с II.5-1 по II.5-4 показаны процедуры CLCSE. Состояния IDLE и AWAITING RESPONSE помечены цифрами "0" и "1", соответственно.

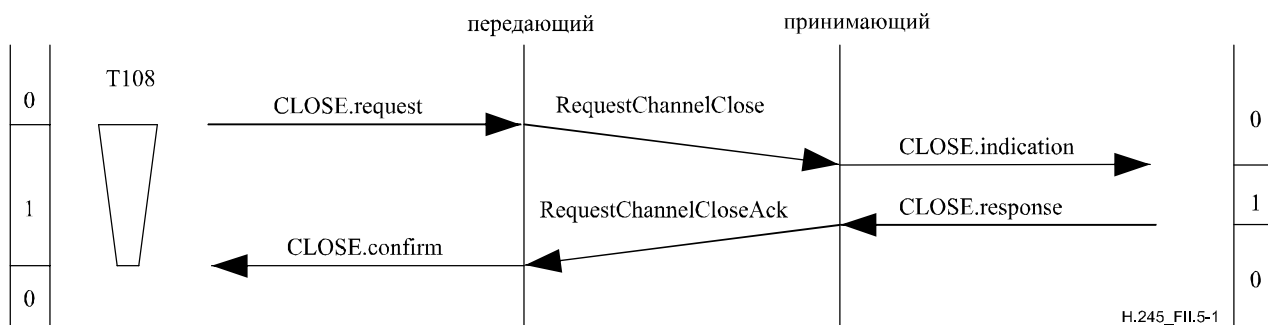


Рисунок П.5-1/Н.245 – Запрос на закрытие логического канала

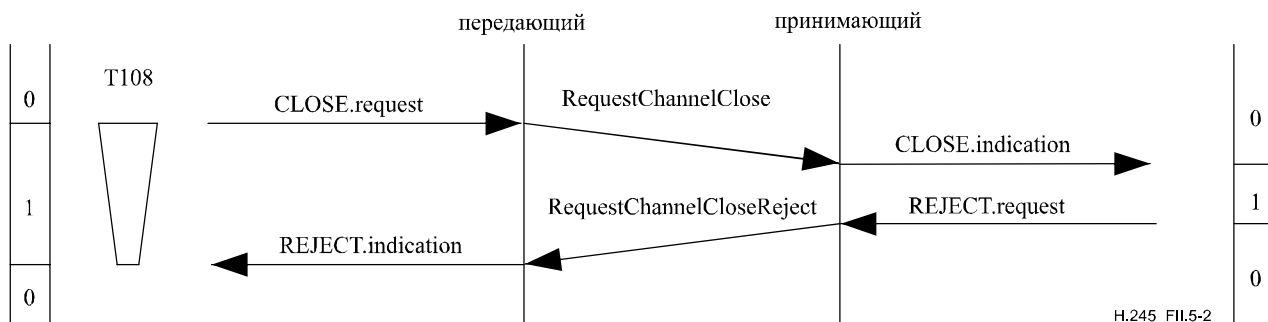
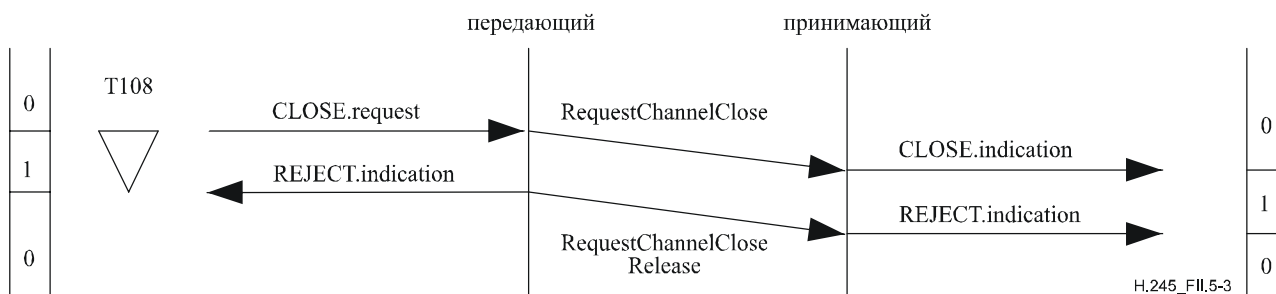
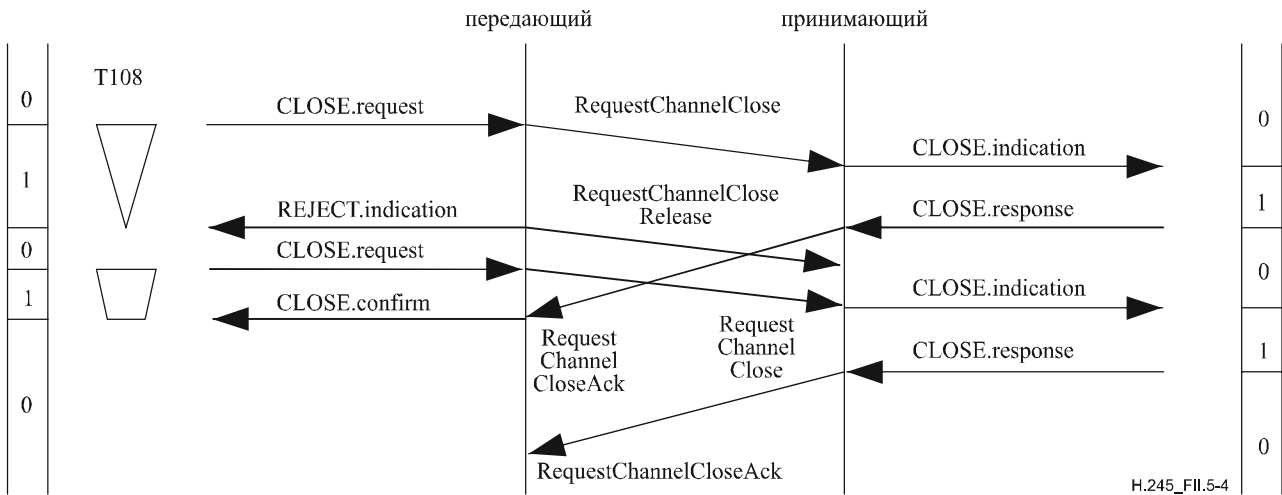


Рисунок П.5-2/Н.245 – Запрос на закрытие логического канала при отказе со стороны пользователя равноправного принимающего CLCSE



ПРИМЕЧАНИЕ. – Сообщение RequestChannelCloseRelease приходит на принимающий CLCSE до ответа от пользователя принимающего CLCSE.

Рисунок П.5-3/Н.245 – Запрос на закрытие логического канала при окончании срока действия таймера T108



ПРИМЕЧАНИЕ. – Запрос на закрытие канала подтверждается получением первого сообщения RequestChannelClose.

Рисунок II.5-4/Н.245 – Запрос на закрытие логического канала при окончании срока действия таймера T108, после чего следует второй запрос на закрытие логического канала

II.6 Объект сигнализации об элементах таблицы мультиплексирования

На рисунках с II.6-1 по II.6-5 показаны процедуры MTSE. Состояния IDLE и AWAITING RESPONSE помечены цифрами "0" и "1", соответственно.

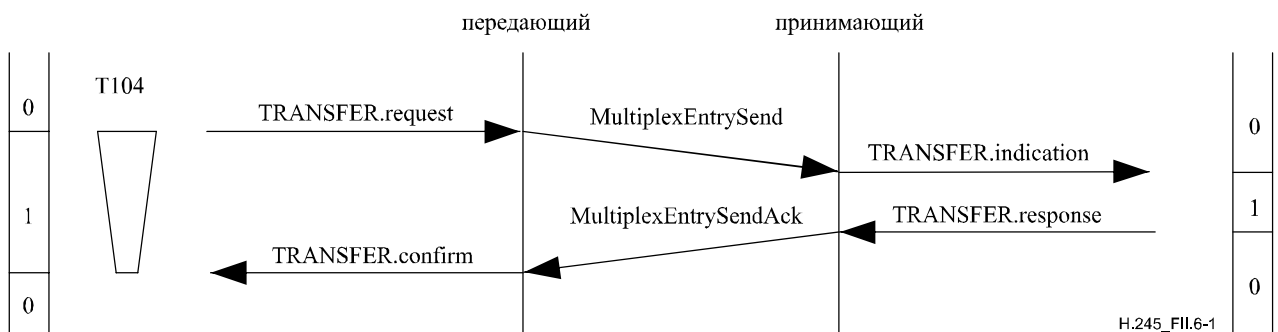


Рисунок II.6-1/Н.245 – Успешный запрос на передачу таблицы мультиплексирования

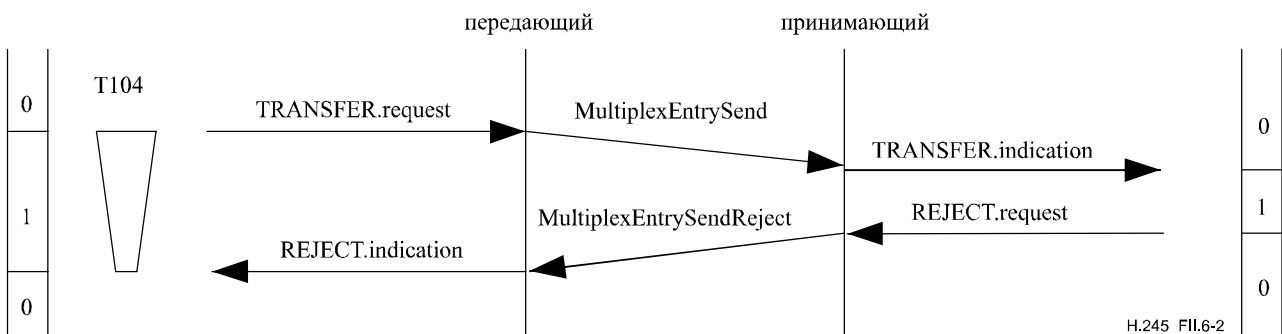
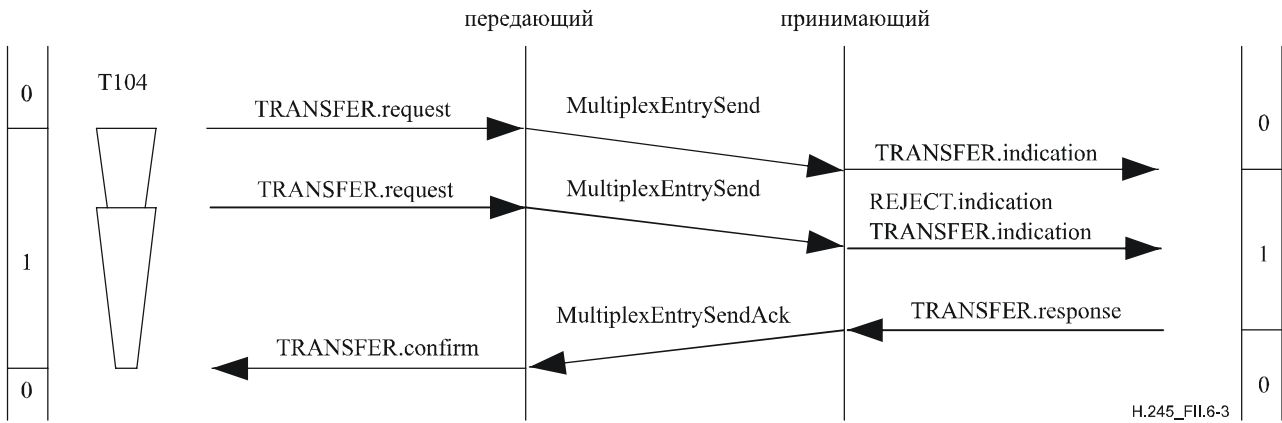


Рисунок II.6-2/Н.245 – Запрос на передачу таблицы мультиплексирования и отказ в нем со стороны пользователя равноправного MTSE



ПРИМЕЧАНИЕ. – Первый запрос был неудачным.

Рисунок II.6-3/Н.245 – Запрос на передачу таблицы мультиплексирования и второй запрос на передачу таблицы мультиплексирования до подтверждения первого запроса

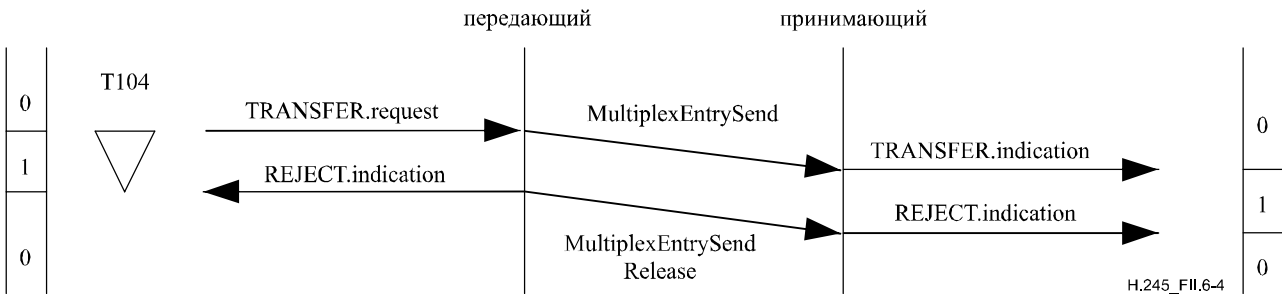
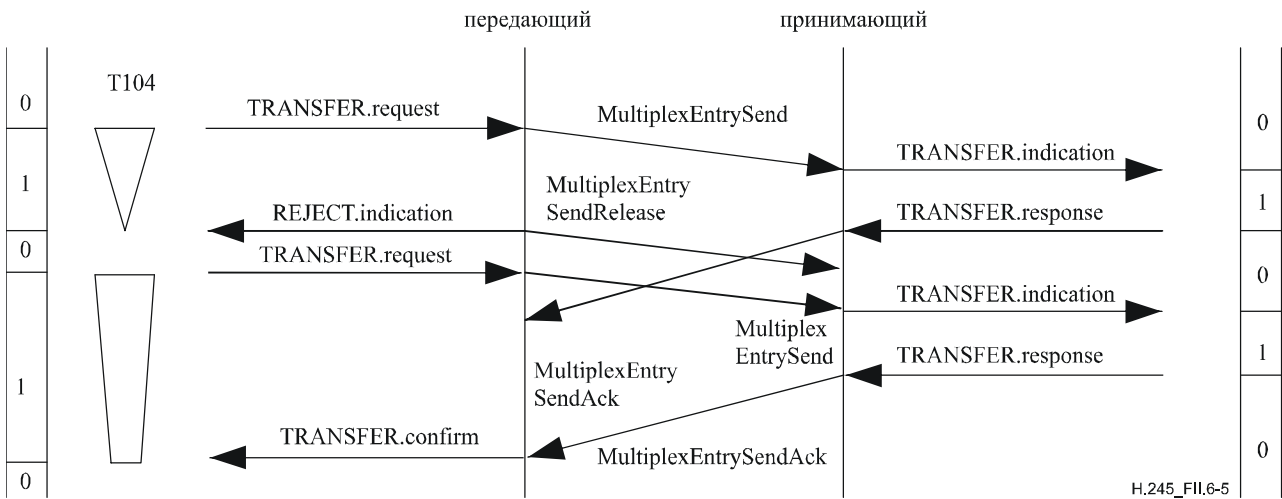


Рисунок II.6-4/Н.245 – Запрос на передачу таблицы мультиплексирования при окончании срока действия таймера T104



ПРИМЕЧАНИЕ. – Первое сообщение MultiplexEntrySendAsk игнорируется на передающем MTSE. Успешным был только второй запрос.

Рисунок II.6-5/Н.245 – Запрос на передачу таблицы мультиплексирования при окончании срока действия таймера T104, после чего происходит второй запрос на передачу таблицы мультиплексирования

II.7 Объект сигнализации запрос режима

На рисунках с II.7-1 по II.7-5 показан обмен между объектами MRSE. Состояния IDLE и AWAITING RESPONSE помечены цифрами "0" и "1", соответственно.

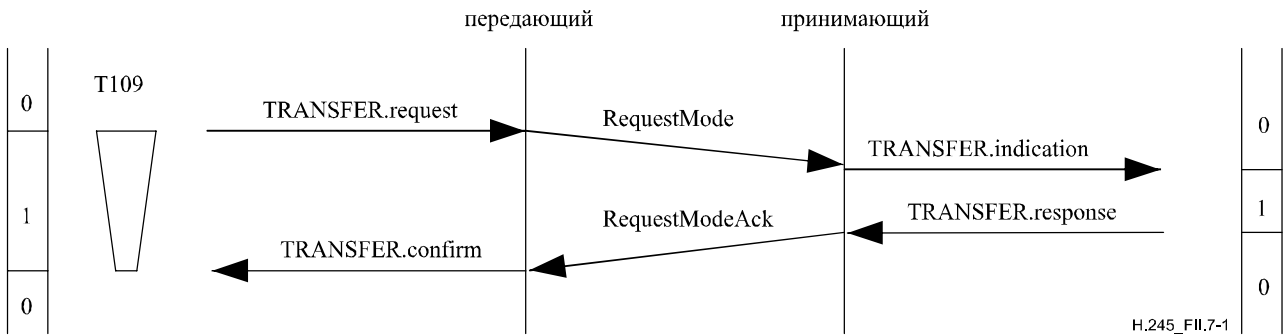


Рисунок II.7-1/Н.245 – Успешный запрос режима

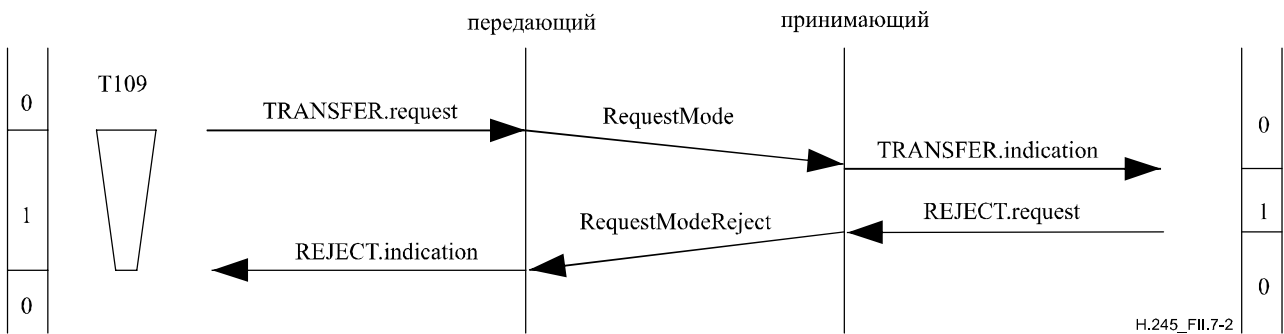
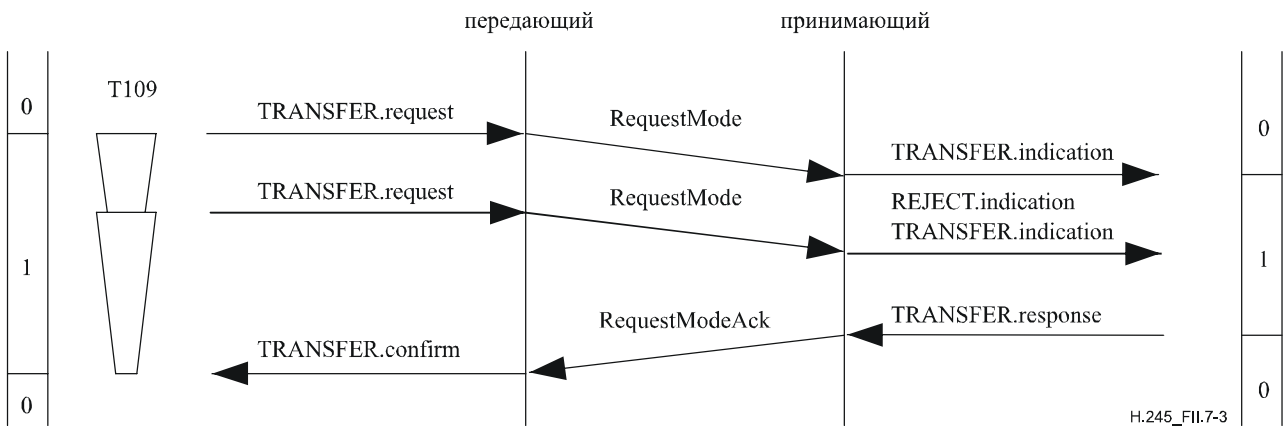


Рисунок II.7-2/Н.245 – Запрос режима и отказ пользователя равноправного MRSE выполнить его



ПРИМЕЧАНИЕ. – Первый запрос был неудачным.

Рисунок II.7-3/Н.245 – Запрос режима и второй запрос режима до подтверждения первого запроса

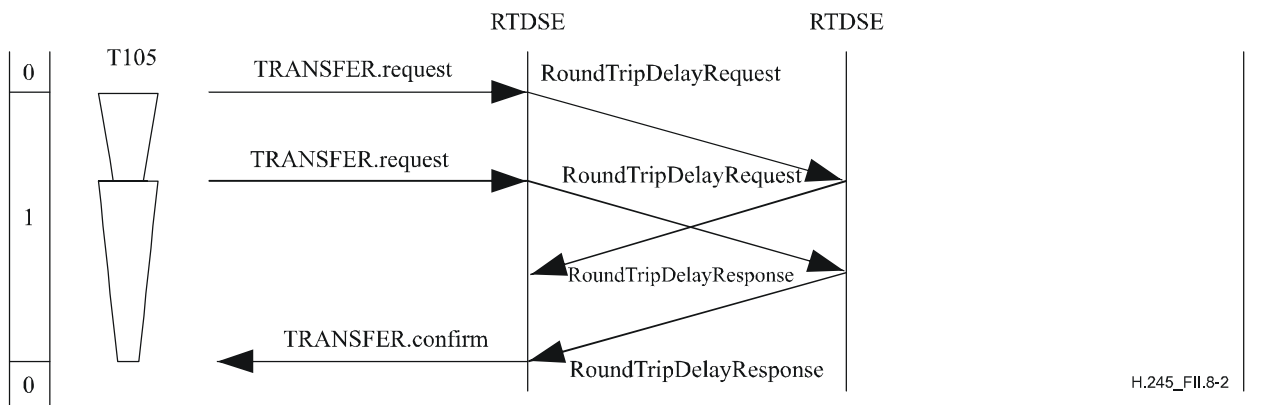


Рисунок П.8-2/Н.245 – Процедура определения задержки из-за подтверждения приема в случае, когда ожидает выполнения более ранняя неподтвержденная процедура определения задержки из-за подтверждения приема

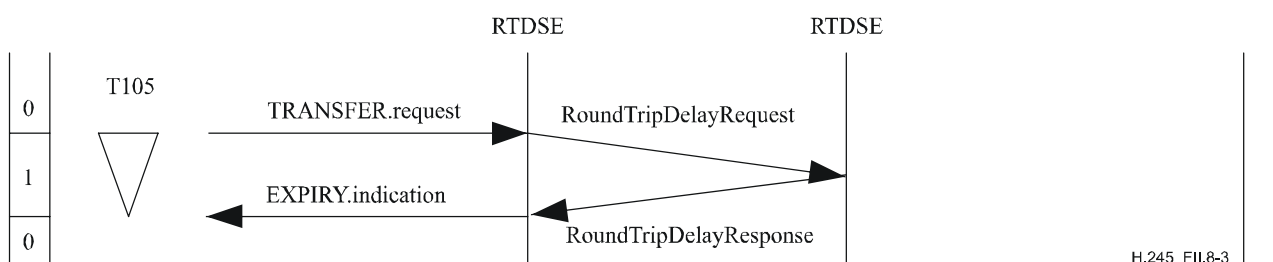
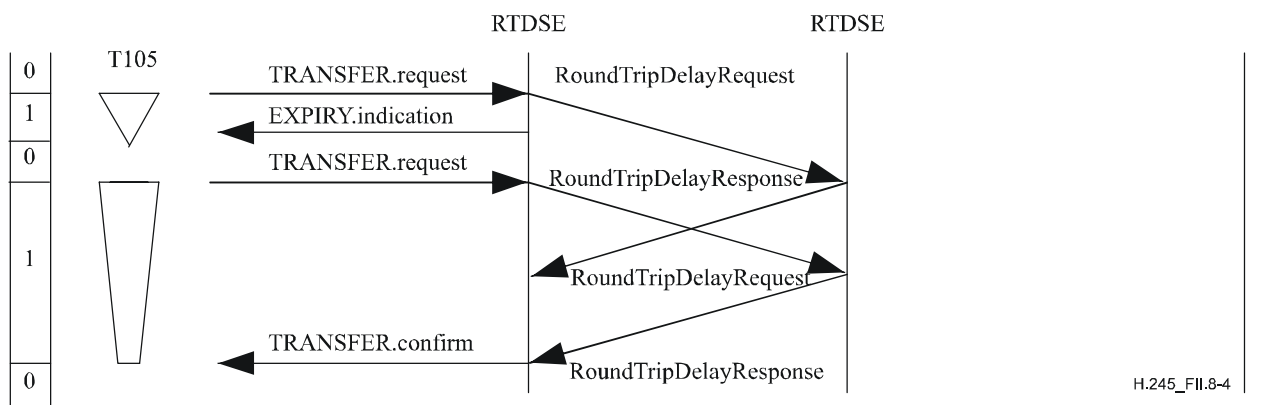


Рисунок П.8-3/Н.245 – Процедура определения задержки из-за подтверждения приема при окончании срока действия таймера T105



ПРИМЕЧАНИЕ. – Сообщение RoundTripDelayResponse от первой процедуры приходит во время выполнения второй процедуры и игнорируется.

Рисунок П.8-4/Н.245 – Процедура определения задержки из-за подтверждения приема при окончании срока действия таймера T105, после чего следует вторая процедура определения задержки из-за подтверждения приема

П.9 Объект сигнализации двунаправленного логического канал

На рисунках с П.9-1 по П.9-7 показаны процедуры B-LCSE. Состояния передающего терминала B-LCSE RELEASED, AWAITING ESTABLISHMENT, ESTABLISHED и AWAITING RELEASE помечены цифрами "0", "1", "2" и "3" соответственно. Состояния принимающего терминала B-LCSE RELEASED, AWAITING ESTABLISHMENT, AWAITING CONFIRMATION и ESTABLISHED помечены цифрами "0", "1", "2" и "3", соответственно.

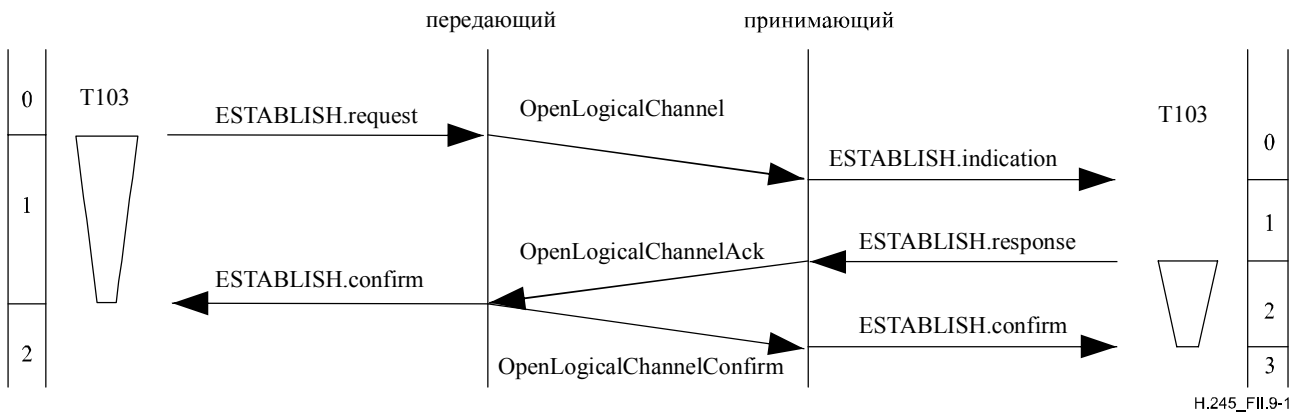


Рисунок П.9-1/Н.245 – Организация двунаправленного логического канала

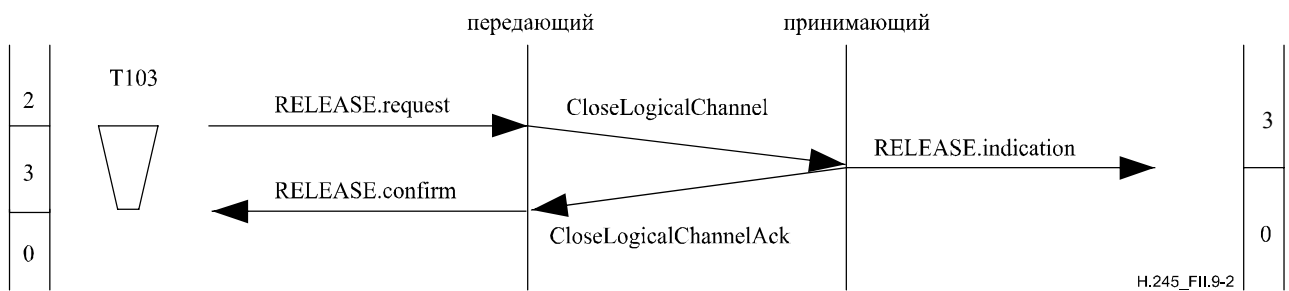


Рисунок П.9-2/Н.245 – Сброс двунаправленного логического канала

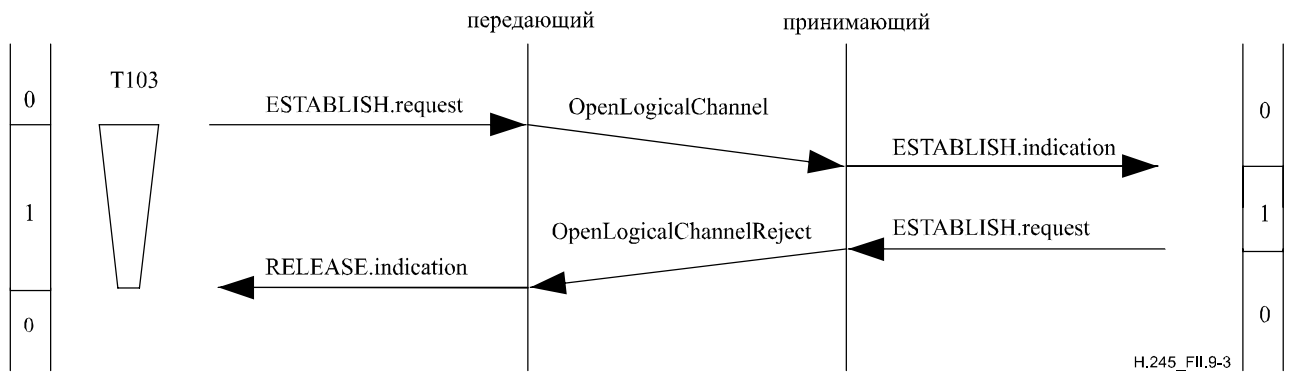


Рисунок П.9-3/Н.245 – Отказ в организации двунаправленного логического канала со стороны пользователя равноправного В-LCSE

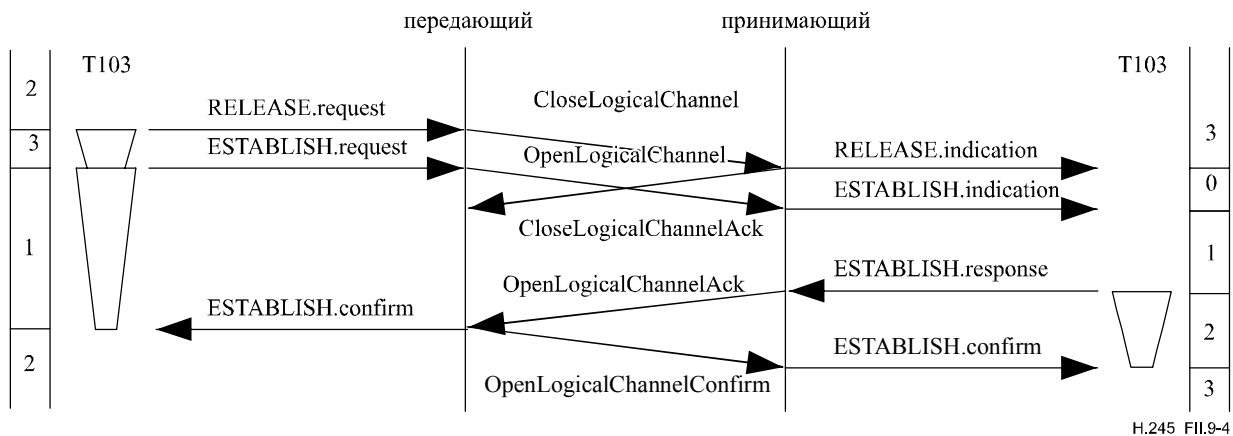


Рисунок П.9-4/Н.245 – Сброс двунаправленного логического канала, после чего следует его немедленная повторная организация

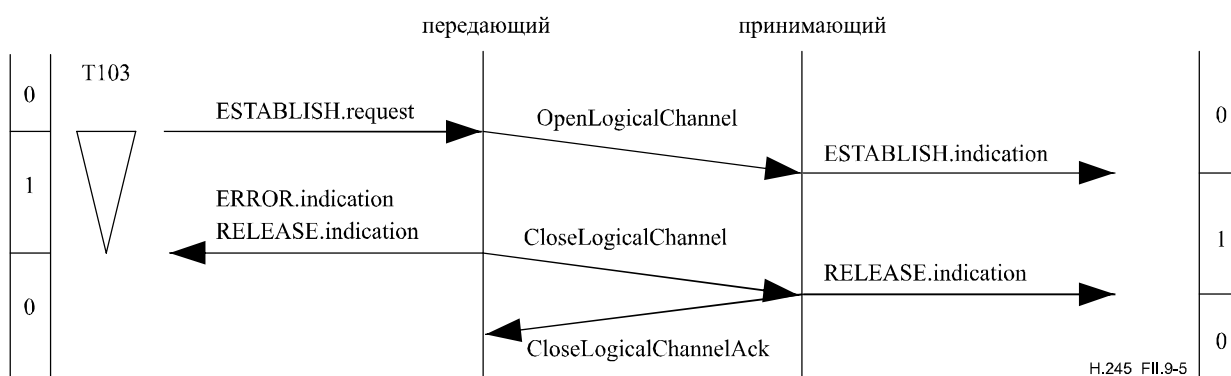
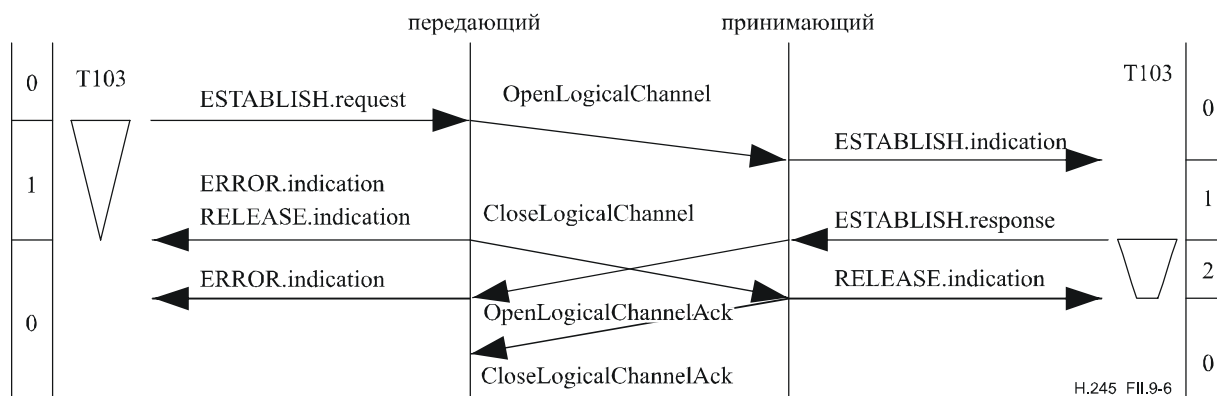


Рисунок П.9-5/Н.245 – Запрос на организацию двунаправленного логического канала при окончании срока действия таймера T103 из-за отсутствия своевременного ответа от пользователя равноправного принимающего В-LCSE



ПРИМЕЧАНИЕ. – Истек срок действия таймера T103 на передающей стороне после передачи сообщения OpenLogicalChannelAsk в принимающем В-LCSE, но до получения сообщения OpenLogicalChannelAck в передающем В-LCSE.

Рисунок П.9-6/Н.245 – Запрос на организацию двунаправленного логического канала при окончании срока действия таймера T103 на передающей стороне

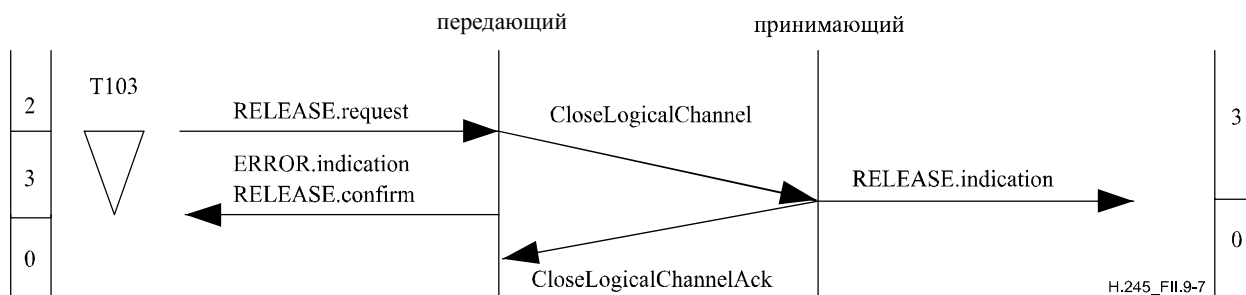


Рисунок П.9-7/Н.245 – Запрос на сброс двунаправленного логического канала при окончании срока действия таймера T103 на передающей стороне

Добавление III

Перечень таймеров и счетчиков процедур

В настоящем Добавлении приведен список таймеров и счетчиков, определенных в Приложении С.

В настоящей Рекомендации не определяются значения, загружаемые в эти таймеры. Эти значения могут быть определены в других Рекомендациях, таких как Рекомендации МСЭ-Т Н.310, Н.323 и Н.324.

III.1 Таймеры

В таблице III.1 приведены таймеры, определяемые в настоящей Рекомендации.

Таблица III.1/Н.245 – Таймеры процедур

Таймер	Процедура	Определение
T106	Определение отношения "ведущий-ведомый"	Этот таймер используется в состоянии OUTGOING AWAITING RESPONSE и во время состояния INCOMING AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщения с подтверждением.
T101	Обмен возможностями	Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщений TerminalCapabilitySetAck или TerminalCapabilitySetReject.
T103	Сигнализация в одностороннем и двустороннем логическом канале	Этот таймер используется в состояниях AWAITING RELEASE и AWAITING ESTABLISHMENT. Он определяет максимальный возможный период без получения сообщений OpenLogicalChannelAck, OpenLogicalChannelReject и CloseLogicalChannelAck.
T108	Закрытие логического канала	Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщений RequestChannelCloseAck или RequestChannelCloseReject.
T104	Таблица мультиплексирования Н.223	Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщений MultiplexEntrySendAck и MultiplexEntrySendReject.

Таблица III.1/Н.245 – Таймеры процедур

Таймер	Процедура	Определение
T109	Запрос режима	Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщений RequestModeAck или RequestModeReject.
T105	Задержка, связанная с подтверждением приема сигнала	Этот таймер используется в состоянии AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщения RoundTripDelayResponse.
T107	Запрос элемента таблицы мультиплексирования	Этот таймер используется в течение состояния AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщений RequestMultiplexEntryAck и RequestMultiplexEntryReject.
T102	Цикл обслуживания	Этот таймер используется в течение состояния AWAITING RESPONSE. Он определяет максимальный возможный период без получения сообщений MaintenanceLoopAck или MaintenanceLoopReject.

III.2 Счетчики

В таблице III.2 указаны счетчики, определяемые в настоящей Рекомендации.

Таблица III.2/Н.245 – Счетчики процедур

Таймер	Процедура	Определение
N100	Определение отношения "ведущий-ведомый"	Этот счетчик определяет максимальное количество раз, которое может передаваться сообщение MasterSlaveDetermination в течение состояния OUTGOING AWAITING RESPONSE.

Добавление IV

Процедура расширения Рекомендации Н.245

Настоящая Рекомендация является живым документом, который используется несколькими системными Рекомендациями, включая Рекомендации Н.310, Н.323, Н.324 и V.70, и весьма вероятно, что она будет расширяться с сохранением обратной совместимости на каждом заседании 16-й Исследовательской комиссии МСЭ-Т. В настоящем Добавлении разъясняется процедура, которую следует применять для добавления расширений к настоящей Рекомендации.

На настоящее время действует только один синтаксис Н.245. Никакая другая Рекомендация МСЭ-Т не должна включать иные варианты синтаксиса Н.245 в свои Рекомендации как нормативы.

Запросы на расширение настоящей Рекомендации следует направлять в виде документов с заголовком White Contribution или официального документа о связи в 16-ю Исследовательскую комиссию, причем следует передать копию такого документа докладчику и редактору документа Н.245 как можно раньше. В такой запрос следует включить:

- 1) функциональные требования к синтаксису, проект которых должен быть подготовлен редактором Н.245, или предлагаемый синтаксис, основанный на утвержденной на текущий момент версии настоящей Рекомендации; и
- 2) предлагаемую семантику для Приложения В; и
- 3) предлагаемые процедуры для Приложения С, если запрошены новые процедуры.

Все расширения настоящей Рекомендации должны обеспечивать обратную совместимость со всеми предыдущими версиями этой Рекомендации. Нельзя изменять существовавшие ранее синтаксис, семантику и процедуры. Нельзя изменять значение существовавшего ранее синтаксиса. А именно, при расширении возможностей H.245 нельзя изменять значение исходной возможности таким образом, чтобы терминалу, не понимающему данное расширение, пришлось бы модифицировать свою работу для того, чтобы использовать данную возможность без расширения. Все компоненты расширения ACH.1 следует ограничивать.

Запросы следует представлять как можно раньше, чтобы предусмотреть время для анализа расширений документа экспертами по Рекомендации H.245 в 16-й Исследовательской комиссии. Необходимо понимать, что возможны изменения точного требуемого синтаксиса по следующим причинам:

- 1) вследствие проверки правильности синтаксиса ACH.1;
- 2) из-за согласования с другими противоречащими запросами на расширение H.245;
- 3) в связи с необходимостью сохранения совместимости с более ранними версиями H.245;
- 4) в результате проведенного экспертами анализа размещения новых функций с точки зрения существующей структуры H.245;
- 5) в результате выявления противоречий с ранее существовавшим синтаксисом;
- 6) из-за неограниченных или неоднозначных компонентов ACH.1.

Следует избегать сокращений и акронимов, особенно в том случае, если какое-либо слово или фраза не сокращались или не были представлены как акроним в ранее существовавшем синтаксисе. Например, не следует сокращать слово "Parameters" как "Params". Если некоторое слово использовалось в существовавшем ранее синтаксисе, не вводите другое слово с тем же значением. Например, называйте компоненты агрегатного типа Entry, а не Item, поскольку слово Entry постоянно используется при их описании. Будьте последовательны.

Все компоненты ACH.1 должны быть ограниченными, и ниже показано, как ограничивать наиболее распространенные типы этих компонентов.

Для ограничения компонентов ACH.1 SET OF и SEQUENCE OF укажите или минимальное и максимальное значение или фиксированный размер. Если семантика компонента не задает очевидное максимальное значение, то выбирайте разумное (пусть произвольное) максимальное значение, например 256. Если компонент SET OF или SEQUENCE OF является OPTIONAL (необязательным), то установите отличное от нуля минимальное значение, исключением при этом будет ситуация, при которой имеется семантическое различие между случаями present-but-empty ("присутствует, но пусто") и not present ("не существует"), в этом случае следует описывать это семантическое различие. Если запрос на расширение содержит компоненты SET OF или SEQUENCE OF, которые не ограничены, то редактор может использовать SIZE (1.. 256) как устанавливаемое по умолчанию ограничение.

Для ограничения компонентов ACH.1 типа строки символов укажите размер как минимальный и максимальный размер или как фиксированный размер. Если запрос на расширение содержит компоненты типа "строка символов", которые не имеют ограничений, то редактор может использовать SIZE (0.. 255) как устанавливаемое по умолчанию ограничение.

Для ограничения компоненты типа INTEGER (целые) указывайте диапазон значений. Если семантика компонента не задает очевидного диапазона, выберите разумный (пусть произвольный) диапазон, максимальное значение которого выбирается из следующих значений:

255	$(2^8 - 1)$
65535	$(2^{16} - 1)$
16777215	$(2^{24} - 1)$
4294967295	$(2^{32} - 1)$

Если запрос на расширение содержит компоненты INTEGER, которые не ограничены, то редактор может использовать INTEGER (0.. 4294967295) как диапазон, применяемый по умолчанию.

Редактор документа H.245 проанализирует все запросы на расширение и предложит окончательный текст расширенных версий H.245 на утверждение 16-й Исследовательской комиссии. После утверждения Исследовательской комиссией каждой новой версии настоящей Рекомендации номер версии H.245 в **protocolIdentifier** будет увеличиваться на единицу для идентификации новой версии.

Обратите внимание на то, что 16-я Исследовательская комиссия намеривается принимать только согласованные расширения H.245, поступающие от редактора H.245.

Добавление V

Процедура `replacementFor`

Процедура H.245 `replacementFor` позволяет бесшовно заменить режимы одного кодека на режимы другого кодека без использования двух декодеров потоков медиаинформации. Эту процедуру можно использовать только в тех случаях, когда принимающий терминал указал на наличие у него возможности `maxPendingReplacementFor`.

Поскольку открытие и закрытие логических каналов H.245 не синхронизировано с содержанием медиаинформации, возможно пропадание медиаинформации между моментом закрытия логического канала и моментом открытия его замены. Параметр `replacementFor` позволяет избежать такого пропадания медиаинформации.

Пример

Предположим, что логический канал 723 открыт и передает аудиопоток G.723.1, и при этом желательно переключиться на G.711 (на логический канал 711), но получатель имеет возможность работать только с одним аудиоканалом. Отправитель может использовать процедуру `replacementFor` для бесшовного переключения режима следующим образом:

- 1) *Только для случая, когда H.323 использует RSVP (протокол резервирования ресурсов),* поскольку новый канал потребует большей пропускной способности (64 кбит/с), чем у существующего канала (6,4 кбит/с), отправитель и получатель резервируют большую пропускную способность RSVP.
- 2) Передатчик передает сообщение `OpenLogicalChannel` для нового логического канала 711, оно включает параметр `replacementFor`, ссылающийся на существующий логический канал 723.
По этому сообщению получатель узнает, что логический канал 711 является *заменой для* логического канала 723 и что данный логический канал 711 не будет передавать трафик одновременно с логическим каналом 723.
- 3) Продолжая декодировать аудиопоток G.723.1 от логического канала 723, получатель готовится к бесшовному переключению на декодирование G.711.
Такая подготовка может включать загрузку соответствующего программного обеспечения декодера.
После того как получатель завершил подготовку к приему аудиопотока G.711, он отвечает сообщением `OpenLogicalChannelAck` для логического канала 711. Для H.323 возвращаемые адреса медиаинформации и транспортные адреса управления медиаинформацией являются теми же, что и адреса, которые уже использовались для логического канала 723.
- 4) Отправитель прекращает передачу аудиопотока G.723 по логическому каналу 723 и бесшовно начинает передавать аудиопоток G.711 по логическому каналу 711.
- 5) Отправитель немедленно передает сообщение `CloseLogicalChannel` для логического канала 723, поскольку этот логический канал больше не передает никакой трафик и он более не нужен.
- 6) *Только в том случае, когда H.323 использует RSVP,* если новому каналу требуется меньшая пропускная способность, чем исходный канал, отправитель и получатель резервируют меньшую пропускную способность RSVP (не относится к данному примеру).

Во всех случаях функционирование LCSE и B-LCSE соответствует обычным процедурам. Параметр `replacementFor` просто информирует получателя об отложенном изменении режима и о том, что эти два логических канала не будут использоваться одновременно, и, следовательно, организация второго логического канала (в некоторых реализациях) может быть одобрена в тех случаях, в которых она была бы отвергнута (из-за отсутствия возможности принимать другой независимый логический канал).

Обратите внимание на то, что в некоторых случаях получатель может отклонить попытку открыть логический канал, используя механизм **replacementFor** (например, если получатель может принять механизм **replacementFor** для аудиоканалов, но не для видеоканалов). В таком случае отправителю следует повторить изменение режима без **replacementFor**, например, закрывая канал с последующим открытием нового канала, соглашаясь при этом с любым временным пропаданием медиаинформации.

Кроме того, обратите внимание на то, что в системах H.323 требуется, чтобы получатели повторно использовали существующие адреса медиаинформации и транспортные адреса управления медиаинформацией. Момент переключения на новый логический канал отмечается в заголовке RTP.

Добавление VI

Примеры задания структуры возможности H.263

В этом Добавлении приведено несколько примеров для разъяснения использования структуры возможности H.263.

VI.1 Примеры установки параметров усовершенствованного уровня H.245

В таблице VI.1 приведены следующие примеры настройки параметров усовершенствованного уровня.

Пример № 1: Соответствует простой базовой видеовозможности H.263 со скоростью 10 кадров/с, максимальной скоростью передачи данных 20 кбит/с без опций.

Пример № 2: Настоящие установки параметров соответствуют возможности потока по логическому каналу с пространственным усовершенствованным уровнем с разрешением QCIF со скоростью 10 кадров/с и максимальной скоростью передачи данных 5 кбит/с без других установленных опций.

Пример № 3: Настоящие установки параметров соответствуют возможности потока по логическому каналу с усовершенствованным уровнем отношения "сигнал-шум" (SNR) при разрешении SQCIF со скоростью 10 кадров/с и максимальной скоростью передачи данных 5 кбит/с без других установленных опций.

Пример № 4: Настоящие установки параметров соответствуют возможности потока по логическому каналу с тремя усовершенствованными уровнями. Два усовершенствованных уровня SNR (один с SQCIF, а другой с QCIF) со скоростью передачи данных 10 кадров/с и без установки других опций и один пространственный усовершенствованный уровень при разрешении CIF со скоростью 10 кадров/с и без установки других опций; все три уровня объединяются с максимальной скоростью передачи данных 15 кбит/с.

Пример № 5: Настоящие установки параметров соответствуют возможности потока по логическому каналу с тремя усовершенствованными уровнями и базовым уровнем с максимальной скоростью передачи данных 25 кбит/с. Базовый уровень имеет разрешение QCIF без опций. Кроме того, терминал способен иметь один усовершенствованный уровень SNR с разрешением QCIF со скоростью 10 кадров/с и без установки других опций, один усовершенствованный уровень SNR с разрешением CIF со скоростью 10 кадров/с и без установки других опций и пространственный усовершенствованный уровень с разрешением CIF со скоростью 10 кадров/с и без установки других опций.

**Таблица VI.1/Н.245 – Примеры установки параметров
усовершенствованного уровня Н.245**

Параметры H263Capability	Примеры									
	1	2	3	4			5			
sqcifMPI	3	NP	NP	NP			NP			
qcifMPI	NP	NP	NP	NP			3			
cifMPI	NP	NP	NP	NP			NP			
cif4MPI	NP	NP	NP	NP			NP			
cif16MPI	NP	NP	NP	NP			NP			
maxBitRate	200	50	50	150			250			
unrestrictedVector	F	F	F	F			F			
arithmeticCoding	F	F	F	F			F			
advancedPrediction	F	F	F	F			F			
pbFrames	F	F	F	F			F			
temporalSpatialTradeOffCap	F	F	F	F			F			
hrd-B	NP	NP	NP	NP			NP			
bppMaxKb	NP	NP	NP	NP			NP			
slowSqcifMPI	NP	NP	NP	NP			NP			
slowQcifMPI	NP	NP	NP	NP			NP			
slowCifMPI	NP	NP	NP	NP			NP			
slowCif4MPI	NP	NP	NP	NP			NP			
slowCif16MPI	NP	NP	NP	NP			NP			
errorCompensation	NP	NP	NP	NP			NP			
SET OF (EnhancementOptions ^{a)}) =	NP	NP	1	1	2		1	2		
snrEnhancement	sqcifMPI		3	3	NP		NP	NP		
	qcifMPI		NP	NP	3		3	NP		
	cifMPI		NP	NP	NP		NP	3		
	cif4MPI		NP	NP	NP		NP	NP		
	cif16MPI		NP	NP	NP		NP	NP		
	maxbitrate		50	50	50		50	50		
SET OF (EnhancementOptions ^{a)}) =	NP	1	NP	NP	NP	1	NP	NP	1	
spatialEnhancement	sqcifMPI	NP				NP			NP	
	qcifMPI	3				NP			NP	
	cifMPI	NP				3			3	
	cif4MPI	NP				NP			NP	
	cif16MPI	NP				NP			NP	
	maxbitrate	50				50			50	
SET OF (EnhancementOptions ^{a)}) =	NP	NP	NP	NP	NP	NP				
bframeEnhancement	sqcifMPI									
	qcifMPI									
	cifMPI									
	cif4MPI									
	cif16MPI									
	maxbitrate									
NP	Не присутствует									
T	True (истина)									
F	False (ложь)									
a)	Другие опции ниже "maxbitrate" в структуре EnhancementOptions не показаны									

VI.2 Примеры установок параметров обратного видеоканала H.245

В этом разделе приведены примеры установок параметров настройки H263Capability и H263Options для работы обратного видеоканала.

Пример 1: Режим отдельных логических каналов

В данном режиме для сообщений обратного видеоканала открывается дополнительный двунаправленный логический канал. Зависимость между прямым видеоканалом и обратным видеоканалом описывается параметрами **forwardLogicalChannelDependency** и **reverseLogicalChannelDependency** в сообщении OpenLogicalChannel.

Логический канал для сообщений обратного видеоканала должен устанавливаться только после организации прямого видеоканала. Если получено сообщение OpenLogicalChannel со ссылкой на зависимость от несуществующего канала, то терминал должен отвечать сообщением OpenLogicalChannelReject с кодом причины invalidDependentChannel. Пример приведен ниже:

- 1) Двунаправленный логический канал для передачи видеоданных открыт между терминалом и терминалом В, как это показано на рисунке VI.1. Сообщение OpenLogicalChannel для двунаправленного логического канала включает RefPictureSelectionCapability в возможности H263VideoCapability.

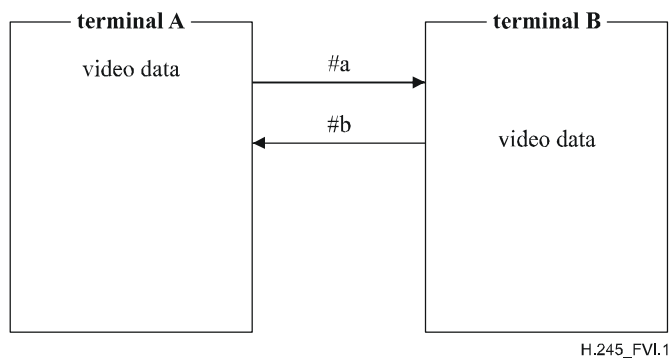


Рисунок VI.1/H.245 – Логические каналы видеоданных

- 2) Затем открывается двунаправленный логический канал для передачи сообщений обратного видеоканала, как показано на рисунке VI.2. В этом примере предполагается, что терминал А посылает запрос на открытие двунаправленного логического канала. (Если терминал В посылает запрос на открытие канала, то параметр forwardLogicalChannelDependency заменяется на параметр reverseLogicalChannelDependency и наоборот.) Сообщение OpenLogicalChannel этого логического канала включает параметр forwardLogicalChannelDependency в forwardLogicalChannelParameters, показывающий номер логического канала #a на рисунке VI.2 и reverseLogicalChannelDependency в reverseLogicalChannelParameters, показывающий номер логического канала LC #b, а также separateVideoBackChannel.

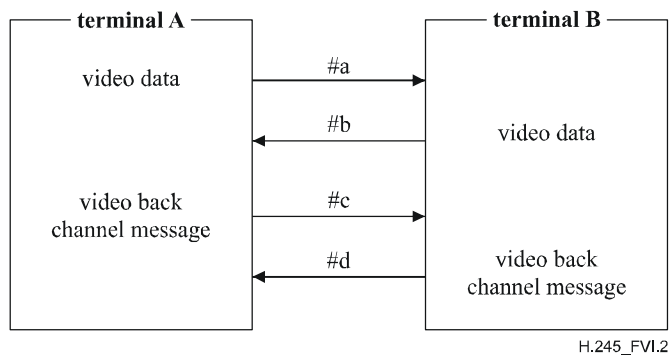


Рисунок VI.2/H.245 – Логические каналы в режиме отдельных логических каналов

- 3) После того как установлен логический канал для передачи сообщений обратного видеоканала, терминал А передает видеоданные в логический канал #а и принимает из логического канала #d сообщения обратного видеоканала, которые соответствуют видеоданным, переданным в логический канал #а. Тем же способом терминал А получает видеоданные из логического канала #b и посылает в логический канал #с сообщения обратного видеоканала, которые соответствуют видеоданным от терминала В.

Примеры установки параметров возможности H263Capability в каждом сообщении OpenLogicalChannel приведены в таблице VI.2. Для простоты показана только часть возможностей H263Capability.

Таблица VI.2/Н.245 – Примеры установки возможности H263Capability для сообщений OpenLogicalChannel

Параметр H263Capability	Возможности H263Capability в сообщениях OpenLogicalChannel		
	#а, #b	#с, #d	#е, #f
sqcifMPI	NP	NP	NP
qcifMPI	3	NP	3
cifMPI	NP	NP	NP
cif4MPI	NP	NP	NP
cif16MPI	NP	NP	NP
MaxBitRate	240	10	240
refPictureSelection		NP	
additionalPictureMemory	Не установлен	–	Не установлен
videoMuxCapability	False (Ложь)	–	(должен быть) True (Истина)
videoBackChannelSendCapability	ackAndNackMessage	–	AckAndNackMessage
separateVideoBackChannel	False (Ложь)	True (Истина)	False (Ложь)
NP	Не присутствует		

Пример 2: режим VideoMux

Если терминал показывает videoMuxCapability в RefPictureSelectionCapability во время обмена возможностями, другой терминал может использовать этот режим для передачи сообщений обратного видеоканала. Поскольку сообщения обратного видеоканала мультиплексированы в кодированный поток видеоданных, терминалам не нужно организовывать дополнительный логический канал для сообщений обратного видеоканала. Пример приведен ниже:

- 1) Двухнаправленный логический канал для передачи видеоданных открыт сообщением OpenLogicalChannel, включающим refPictureSelectionCapability со значениями true (истина) в режиме "VideoMux" в своей возможности H263VideoCapability (см. рисунок VI.3).
- 2) После установки логического канала для видеоданных терминал А передает видеоданные в логический канал #е и принимает из логического канала #f сообщения обратного видеоканала, которые соответствуют видеоданным, переданным в логический канал #е, мультиплексируются в видеоданные из терминала В.

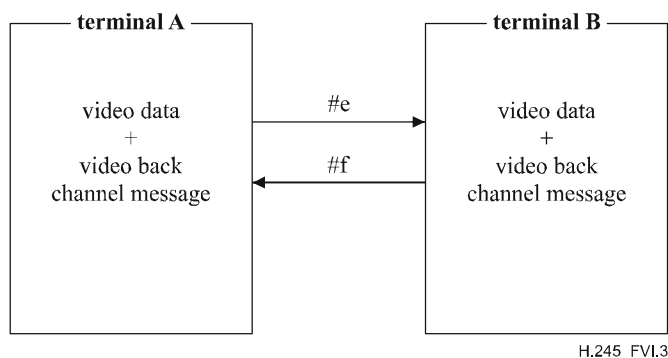


Рисунок VI.3/Н.245 – Логические каналы в режиме VideoMux

Примеры установки параметров возможности H263Capability во всех сообщениях OpenLogicalChannel приведены в таблице VI.2.

Пример 3: Режим отдельных логических каналов в однонаправленной видеосвязи

Этот пример соответствует случаю, когда только терминал А посылает видеоданные, а терминал В передает только сообщения обратного видеоканала (см. рисунок VI.4). Установки параметров настройки для логических каналов #g и #h приведены в таблице VI.3.

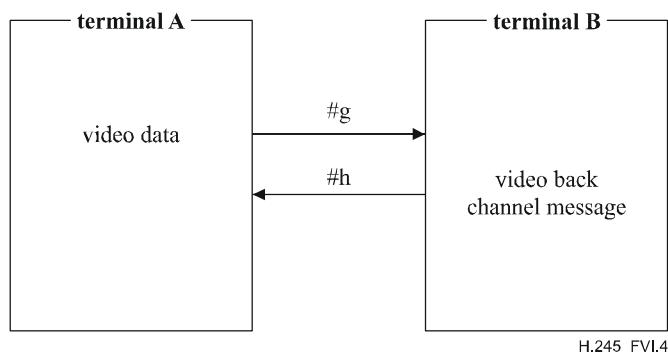


Рисунок VI.4/Н.245 – Режим отдельных логических каналов при однонаправленной видеосвязи

Пример 4: Существование режима отдельных логических каналов с режимом VideoMux

Пример, приведенный на рисунке VI.5, описывает случай, когда только терминал А использует режим отдельных логических каналов для получения сообщений обратного видеоканала от терминала В через логический канал #l, а терминал В использует режим VideoMux для получения сообщений обратного видеоканала через логический канал #i. Этот пример может выглядеть нереалистичным, но в нем показана возможная конфигурация. Установки параметров для всех логических каналов приведены в таблице VI.3.

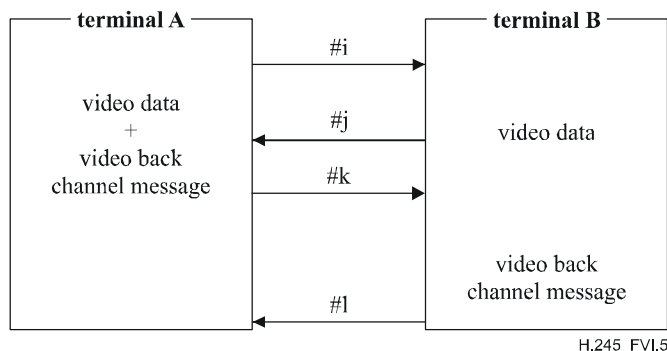


Рисунок VI.5/Н.245 – Сосуществование режима отдельных логических каналов с режимом VideoMux

Таблица VI.3/Н.245 – Примеры установки возможности H263Capability для сообщений OpenLogicalChannel

Возможность H263Capability в сообщении OpenLogicalChannel						
Параметр H263Capability	#g	#h	#i	#j	#k	#l
sqcifMPI	NP	NP	NP	NP	NP	NP
qcifMPI	3	NP	3	3	NP	NP
cifMPI	NP	NP	NP	NP	NP	NP
cif4MPI	NP	NP	NP	NP	NP	NP
cif16MPI	NP	NP	NP	NP	NP	NP
maxBitRate	240	10	240	240	10	10
refPictureSelection		NP			NP	NP
additionalPictureMemory	не определен	–	не определен	не определен	–	–
videoMuxCapability	F	–	F	(должно быть) T	–	–
videoBackChannelSendCapability	Сообщение ackAndNack	–	Сообщение AckAndNack	Сообщение ackAndNack	–	–
separateVideoBackChannel	F	T	F	F	F	T
NP	Не присутствует					
T	True (Истина)					
F	False (Ложь)					

Добавление VII

Процедура и шаблон для определения новых возможностей в качестве универсальных возможностей Н.245

В настоящем Добавлении определяется процедура и шаблон определения новых возможностей в виде универсальных возможностей Н.245. В нем также приводится пример того, как этот шаблон может быть использован для описания кодека Н.261 вместо синтаксиса АСН.1, который использован в настоящей Рекомендации. Этот новый механизм для определения возможностей в настоящей Рекомендации предназначен для применения ко всем новым возможностям, которые добавляются к настоящей Рекомендации; этот механизм не предназначен для использования для переопределения существующих возможностей.

Описания возможностей, относящихся к Рекомендациям МСЭ-Т, должны быть определены в приложениях к настоящей Рекомендации либо в самой Рекомендации (например, в Рекомендации МСЭ-Т Н.283).

Другие описания возможности могут быть определены в Приложениях к настоящей Рекомендации или в других документах.

Универсальные возможности *GenericCapabilities*, которые включают и свертываемые, и несвертываемые последовательности, не должны содержать структуры *GenericParameter* различных типов (свертываемые, несвертываемые), которые используют один и тот же идентификатор параметра *parameterIdentifier*.

ПРИМЕЧАНИЕ 1. – Такое повторное использование одного идентификатора параметра *parameterIdentifier* может вызвать конфликт значений *parameterIdentifier*, если параметр автоматически преобразуется в системе, например в системе Н.320, которая не различает свертываемые и несвертываемые параметры.

Стандартному полю *parameterIdentifier* универсального параметра *GenericParameter* не следует присваивать значение 0.

ПРИМЕЧАНИЕ 2. – Такое назначение значения 0 порождает конфликт при автоматическом преобразовании в сигнализацию Н.320, например, см. Приложение А/Н.239 и Рекомендацию МСЭ-Т Н.241.

VII.1 Процедура

VII.1.1 Определение универсальных возможностей в настоящей Рекомендации

В случае, если определение должно быть включено в Приложения к настоящей Рекомендации, следует выполнять приведенную ниже процедуру:

- 1) Определить идентификатор объекта OBJECT IDENTIFIER для этой возможности и включить его в перечень в Приложении D.
- 2) Определить возможность как универсальную возможность в новом Приложении к настоящей Рекомендации.

Идентификатор объекта OBJECT IDENTIFIER имеет следующую форму: `{itu-t (0) recommendation (0) h (8) 245 generic-capabilities (1) capability-class capability-name}`.

capability-class (класс возможности) является одним из следующих классов: video (видео) (0), audio (аудио) (1), data (данные) (2), control (управление) (3) или multiplex (мультиплексирование) (4). Значение *capability-name* (название возможности) назначается по порядку чисел для каждого класса возможностей.

VII.1.2 Определение универсальных возможностей в других Рекомендациях МСЭ

В том случае, если определение должно быть включено в другие Рекомендации МСЭ, следует выполнять приведенную ниже процедуру:

- 1) Определить идентификатор объекта для этой возможности в самой Рекомендации и включить его в перечень в Приложении VIII.
- 2) Определить эту возможность как универсальную возможность в новом приложении к соответствующей Рекомендации.

VII.1.3 Определение универсальных возможностей в стандартах, разработанных не в МСЭ

В случае, если определение должно быть включено в стандарт, разработанный не в МСЭ, следует выполнять приведенную ниже процедуру:

- 1) Определить идентификатор объекта для этой возможности в соответствующем стандарте и включить его в перечень в Добавлении VIII.
- 2) Определить эту возможность как универсальную возможностью в соответствующем стандарте.

VII.2 Шаблон

VII.2.1 Идентификатор возможности

При описании каждой *GenericCapability* должна быть определена таблица VII.1 в одном экземпляре.

Таблица VII.1/Н.245 – Шаблон идентификатора возможности

Название возможности	Название кодека, например H.261
Класс возможности	Класс возможности, например видео, аудио и т. д.
Тип идентификатора возможности	Тип идентификатора, определяющего кодек: standard (стандартный), h221NonStandard (нестандартный h221) или uuid.
Значение идентификатора возможности	Значение тега кодека, например {itu-t (0) recommendation (0) h (8) 261 generic-capabilities (1) 0}. Значение универсальных возможностей определяет тип или набор параметров, относящихся к данной возможности. Обратите внимание на то, что реальный формат этого идентификатора объекта задается при определении возможности, но его следует задавать с учетом возможных расширений.
maxBitRate	Поле maxBitRate или должно быть включено, или не должно быть включено, или оно является необязательным.
nonCollapsingRaw	Спецификация формата OCTET STRING ("строка октетов"), а также показывает, должна ли она включаться.
transport	Показывает, что поле transport ("транспорт") или должно быть включено, или не должно быть включено, или оно является необязательным.

VII.2.2 Параметры возможности

Этот пункт применим к свертываемым и несвертываемым универсальным параметрам. Для каждого универсального параметра должна быть определена таблица VII.2 в одном экземпляре. Шаблон должен быть разделен на части, чтобы отличить параметры, предназначенные для согласования возможностей, от параметров, предназначенных для сигнализации логического канала.

Таблица VII.2/Н.245 – Шаблон параметра возможности

Название параметра	Название параметра, например cifMPI
Описание параметра	Описательное название параметра, например "Определяет минимальный интервал между изображениями при разрешении CIF"
Значение идентификатора параметра	Целое число, которое определяет данный "стандартный" параметр
Статус параметра	Параметр является обязательным или условно обязательным (например зависящим от другого параметра), или необязательным.
Тип параметра	Тип параметра: logical (логический), booleanArray, unsignedMin, unsignedMax, unsigned32Min, unsigned32Max, octetString [или genericParameter – универсальный параметр].
Заменяет	Параметры, которые этот параметр заменяет. Этот элемент таблицы должен иметь значения 0, 1 или большее число, соответствующее числу параметров, которые заменяет данный параметр. Формат параметра должен иметь следующий вид: название параметра ("значение идентификатора-параметра"), например qcifMPI (0)
<p>ПРИМЕЧАНИЕ. – Эта таблица не разрешает определять тип тега параметра ParameterTag (standard (стандартный), h221NonStandard (нестандартный h221) или uuid), поскольку он предназначен только в описаниях стандартных возможностей.</p>	

VII.3 Пример шаблона – H.261

VII.3.1 Идентификатор возможности H.261

Таблица VII.3/H.245 – Пример идентификатора возможности H.261

Название возможности	ITU-T Rec. H.261 (МСЭ-Т H.261)
Класс возможности	Видеокодек
Тип идентификатора возможности	Стандартный
Значение идентификатора возможности	{itu-t (0) recommendation (0) h (8) 261 generic-capabilities (1) 0}. Это первый (и единственный) набор параметров, определенных для Рекомендации МСЭ-Т H.261.
maxBitRate	Поле maxBitRate всегда должно быть включено.
nonCollapsingRaw	Это поле не должно быть включено.
Transport	Это поле не должно быть включено.

VII.3.2 Параметры возможности H.261

Обратите внимание на отсутствие таблицы для поля "максимальная скорость передачи данных", которое имеется в синтаксисе АСН.1 в Рекомендации МСЭ-Т H.245 для Рекомендации МСЭ-Т H.261. Это происходит потому, что максимальная скорость передачи данных задается на верхнем уровне структуры GeneriCapability. Также обратите внимание на то, что параметры temporalSpatialTradeOffCapability и stillImageTransmission могут быть объединены в один универсальный параметр GenericParameter типа booleanArray.

Таблица VII.4/H.245 – Пример параметра qcifMPI возможности H.261

Название параметра	qcifMPI
Описание параметра	Если данный параметр присутствует, то он показывает минимальный интервал между изображениями в единицах измерения 1/29,97 для кодирования и/или декодирования кадров QCIF, а если он отсутствует, то никаких характеристик изображения для кадров QCIF не указано.
Значение идентификатора параметра	0
Статус параметра	Необязательный
Тип параметра	unsignedMax
Заменяет	–

Таблица VII.5/H.245 – Пример параметра cifMPI возможности H.261

Название параметра	cifMPI
Описание параметра	Если данный параметр присутствует, то он показывает минимальный интервал между изображениями в единицах измерения 1/29,97 для кодирования и/или декодирования кадров CIF, а если он отсутствует, то никаких характеристик изображения для кадров CIF не указано.
Значение идентификатора параметра	1
Статус параметра	Необязательный
Тип параметра	unsignedMax
Заменяет	qcifMPI (0)

Таблица VII.6/Н.245 – Пример параметра temporalSpatialTradeOffCapability возможности Н.261

Название параметра	temporalSpatialTradeOffCapability
Описание параметра	Наличие данного параметра показывает, что кодер способен изменять свое соотношение между временным и пространственным разрешением в соответствии с командами удаленного терминала. Данный параметр не имеет никакого смысла, если он является частью возможности приема.
Значение идентификатора параметра	2
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Таблица VII.7/Н.245 – Пример параметра stillImageTransmission возможности Н.261

Название параметра	stillImageTransmission
Описание параметра	Наличие данного параметра обозначает характеристику для неподвижных изображений в соответствии с Приложением D/Н.261.
Значение идентификатора параметра	3
Статус параметра	Необязательный
Тип параметра	Логический
Заменяет	–

Добавление VIII

Список универсальных возможностей и универсальных сообщений, определенных в других Рекомендациях/Стандартах, помимо настоящей Рекомендации

В таблице VIII.1 перечислены универсальные возможности, определенные в других Рекомендациях или стандартах, помимо настоящей Рекомендации.

Таблица VIII.1/Н.245 – Список универсальных возможностей, определенных в других Рекомендациях/стандартах, помимо настоящей Рекомендации

Название возможности	Класс возможности	Идентификатор возможности	Название рекомендации или стандарта, устанавливающего эту возможность
Н.283	Протокол передачи данных	{itu-t (0) recommendation (0) h (8) 283 generic-capabilities (1) 0}	Рекомендация МСЭ-Т Н.283
G.722.1	Протокол передачи звука	{itu-t (0) recommendation (0) g (7) 7221 generic-capabilities (1) 0}	Рекомендация МСЭ-Т G.722.1
Н.324	Протокол передачи данных	{itu-t (0) recommendation (0) h (8) 324 generic-capabilities (1) http (0)}	Рекомендация МСЭ-Т Н.324

Таблица VIII.1/Н.245 – Список универсальных возможностей, определенных в других Рекомендациях/стандартах, помимо настоящей Рекомендации

Название возможности	Класс возможности	Идентификатор возможности	Название рекомендации или стандарта, устанавливающего эту возможность
H.263	Протокол передачи видео	{itu-t (0) recommendation (0) h (8) 263 generic-capabilities (1) 0}	Рекомендация МСЭ-Т Н.263 ПРИМЕЧАНИЕ. – Использование этой возможности для сообщения о "профилях и уровнях" Н.263 в Приложении X/Н.263 всегда должно сопровождаться параллельно передачей сигналов тех же режимов в H263VideoCapability. Это необходимо для обеспечения того, чтобы системы, которые не распознают универсальные возможности Н.263, могли взаимодействовать с более новыми системами.
H.224	Протокол передачи данных	{itu-t (0) recommendation (0) h (8) 224 generic-capabilities (1) 0}	Рекомендация МСЭ-Т Н.224
G.722.2	Протокол передачи звука	{itu-t (0) recommendation (0) g (7) 7222 generic-capabilities (1) 0}	Рекомендация МСЭ-Т G.722.2
G.726	Протокол передачи звука	{itu-t (0) recommendation (0) g (7) 726 generic-capabilities (1) version2003 (0)}	Рекомендация МСЭ-Т G.726
H.241/Н.264	Протокол передачи видео	{itu-t (0) recommendation (0) h (8) 241 specificVideoCodecCapabilities (0) h264 (0) generic-capabilities (1)}	Рекомендация МСЭ-Т Н.241
h239ControlCapability	Протокол управления	{itu-t (0) recommendation (0) h (8) 239 generic-capabilities (1) h239ControlCapability (1)}	Рекомендация МСЭ-Т Н.239
h239ExtendedVideoCapability	Протокол передачи видео	{itu-t (0) recommendation (0) h (8) 239 generic-capabilities (1) h239ExtendedVideoCapability (2)}	Рекомендация МСЭ-Т Н.239

В таблице VIII.2 перечислены универсальные сообщения, определенные в других Рекомендациях или Стандартах, помимо настоящей Рекомендации.

Таблица VIII.2/Н.245 – Список универсальных сообщений, определенных в других Рекомендациях/стандартах, кроме настоящей Рекомендации

Название сообщения	Класс сообщения	Идентификатор сообщения	Название рекомендации или стандарта, устанавливающего это сообщение
H.239	Универсальное сообщение	{itu-t (0) recommendation (0) h (8) 239 generic-message (2)}	Рекомендация МСЭ-Т Н.239

Добавление IX

Использование языка АСН.1 в настоящей Рекомендации

В настоящем Добавлении перечислены понятия языка АСН.1, которые использовались в настоящей Рекомендации. 16-я Исследовательская комиссия намерена ограничить расширения этой Рекомендации использованием только этих понятий. Дополнительные понятия языка АСН.1 будут рассматриваться только в исключительных обстоятельствах.

IX.1 Сопровождение данных тегами

Все теги в этой Рекомендации относятся к автоматическим тегам (AUTOMATIC TAGS).

IX.2 Типы

В определениях языка АСН.1 в этой Рекомендации встречаются следующие типы:

BIT STRING	IA5String	OCTET STRING
BMPString	INTEGER	SEQUENCE
BOOLEAN	NULL	SEQUENCE OF
CHOICE	NumericString	SET
GeneralString	OBJECT IDENTIFIER	SET OF

IX.3 Ограничения и диапазоны

В настоящей Рекомендации используются ограничения размеров ("SIZE": string (строки), set-of и sequence-of), ограничения диапазонов (целые) и разрешенные алфавиты ("FROM").

IX.4 Расширяемость

В настоящей Рекомендации используется маркер расширения (троеточие "...").

СЕРИИ РЕКОМЕНДАЦИЙ МСЭ-Т

Серия А	Организация работы МСЭ-Т
Серия В	Средства выражения: определения, символы, классификация
Серия С	Общая статистика электросвязи
Серия D	Общие принципы тарификации
Серия E	Общая эксплуатация сети, телефонная служба, функционирование служб и человеческие факторы
Серия F	Нетелефонные службы электросвязи
Серия G	Системы и среда передачи, цифровые системы и сети
Серия H	Аудиовизуальные и мультимедийные системы
Серия I	Цифровая сеть с интеграцией служб
Серия J	Кабельные сети и передача сигналов телевизионных и звуковых программ и других мультимедийных сигналов
Серия K	Защита от помех
Серия L	Конструкция, прокладка и защита кабелей и других элементов линейно-кабельных сооружений
Серия M	TMN и техническое обслуживание сетей: международные системы передачи, телефонные, телеграфные, факсимильные и арендованные каналы
Серия N	Техническое обслуживание: международные каналы передачи звуковых и телевизионных программ
Серия O	Требования к измерительной аппаратуре
Серия P	Качество телефонной передачи, телефонные установки, сети местных линий
Серия Q	Коммутация и сигнализация
Серия R	Телеграфная передача
Серия S	Оконечное оборудование для телеграфных служб
Серия T	Оконечное оборудование для телематических служб
Серия U	Телеграфная коммутация
Серия V	Передача данных по телефонной сети
Серия X	Сети передачи данных и взаимосвязь открытых систем
Серия Y	Глобальная информационная инфраструктура, аспекты межсетевого протокола и сети последующих поколений
Серия Z	Языки и общие аспекты программного обеспечения для систем электросвязи