INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.223

(03/96)

SERIES H: TRANSMISSION OF NON-TELEPHONE SIGNALS

Infrastructure of audiovisual services - Transmission multiplexing and synchronization

Multiplexing protocol for low bit rate multimedia communication

ITU-T Recommendation H.223

(Previously «CCITT Recommendation»)

FOREWORD

The ITU-T (Telecommunication Standardization Sector) is a permanent organ of the International Telecommunication Union (ITU). The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, March 1-12, 1993).

ITU-T Recommendation H.223 was prepared by ITU-T Study Group 15 (1993-1996) and was approved under the WTSC Resolution No. 1 procedure on the 19th of March 1996.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

CONTENTS

SUMMARY

This Recommendation specifies a packet-oriented multiplexing protocol for low bit rate multimedia communication. This protocol can be used between two low bit rate multimedia terminals, or between a low bit rate multimedia terminal and a multipoint control unit or an interworking adapter. The protocol allows the transfer of any combination of digital voice/audio, digital video/image and data information over a single communication link. This protocol provides low delay and low overhead by using segmentation and reassembly and by combining information from different logical channels in a single packet. The control procedures necessary to implement this multiplexing protocol are specified in Recommendation H.245.

MULTIPLEXING PROTOCOL FOR LOW BIT RATE MULTIMEDIA COMMUNICATION

(Geneva, 1996)

1 General

This Recommendation specifies the frame structure, format of fields and procedures of the packet multiplexing protocol for low bit rate multimedia communication. This protocol can be used between two low bit rate multimedia terminals, or between a low bit rate multimedia terminal and a Multipoint Control Unit (MCU) or an InterWorking adapter (IWA). The control procedures necessary to implement this multiplexing protocol are specified in Recommendation H.245.

In this Recommendation communication between different protocol layers is modelled as a set of abstract primitives, which represent a logical exchange of information. The decomposition of functionality into (sub)layers, as well as the description of the primitives, do not imply a particular method of implementation. In particular, layers may exchange the contents of a logical unit (an SDU) in a "streaming" mode where information exchange may start before the transferring layer has the complete unit in its possession.

2 Normative References

The following Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

[1] ITU-T Recommendation H.245 (1996), *Control protocol for multimedia communication*.

[2] ITU-T Recommendation V.42 (1993), *Error-correcting procedures for DCEs using asynchronous-to-synchronous conversion*.

[3] ITU-T Recommendation H.324 (1996), *Terminal for low bit rate multimedia communication*.

[4] CCITT Recommendation Q.922 (1992), *ISDN data link layer specification for frame mode bearer services*.

3 Definitions and format conventions

3.1 Definition of terms

For the purposes of this Recommendation, the following definitions apply.

3.1.1 adaptation layer (AL): The upper of the two layers of the multiplexer of this Recommendation.

3.1.2 AL-PDU: An information unit exchanged between peer Adaptation Layer entities. An AL-PDU is conveyed as one MUX-SDU.

3.1.3 AL-SDU: A logical information unit whose integrity is preserved in transfer from one AL user to the peer AL user.

3.1.4 AL user: A higher-layer entity which makes use of the services of the Adaptation Layer.

3.1.5 control channel: A logical channel which carries H.245 control messages.

3.1.6 header error control (HEC) field: A 3-bit CRC field in the MUX-PDU header which is used to detect errors that affect the MC field.

3.1.7 logical channel number (LCN): A unique integer between 0 and 65535 assigned to a logical channel.

3.1.8 multiplex code (MC) field: A 4-bit field in the MUX-PDU header which specifies, by reference to a multiplex table entry, the logical channel that each octet in the information field belongs.

3.1.9 multiplex (MUX) layer: The lower of the two layers of the multiplexer of this Recommendation.

3.1.10 multiplex table: A table with up to 16 entries which specifies the multiplexing pattern for the information field of a MUX-PDU.

3.1.11 MUX-PDU: An information unit exchanged between peer MUX layer entities.

3.1.12 MUX-SDU: A logical information unit whose integrity is preserved in transfer from one Adaptation Layer to the peer Adaptation Layer.

3.1.13 non-segmentable logical channel: A logical channel whose MUX-SDUs may not be segmented. MUX-SDUs from a non-segmentable logical channel are transmitted in consecutive octets of a single MUX-PDU.

3.1.14 packet marker (PM) field: A one-bit field used to mark the end of a MUX-SDU from a segmentable logical channel.

3.1.15 protocol data unit (PDU): A unit of information exchanged between peer protocol layer entities.

3.1.16 quality of service (QOS): The quality of the service that individual information streams receive from the multiplexer, as measured by parameters such as bit rate, delay jitter, loss, etc.

3.1.17 segmentable logical channel: A logical channel whose MUX-SDUs may be segmented. Segmentation allows the temporary suspension of the transmission of a MUX-SDU in order to transmit octets from another MUX-SDU.

3.1.18 service data unit (SDU): A logical unit of information whose integrity is preserved in transfer from one protocol layer entity to the peer protocol layer entity.

3.1.19 slot: A consecutive sequence of octets within a single MUX-PDU, described by a single H.245 MultiplexElement structure oftype logicalChannelNumber. Each slot holds an integral number of octets from a single MUX-SDU.

3.2 Format conventions

The numbering, field mapping and bit transmission conventions used in this Recommendation are consistent with those used in Recommendation V.42.

3.2.1 Numbering convention

The basic numbering convention used in this Recommendation is illustrated in Figure 1. The bits in each information unit are grouped into octets. The bits of an octet are shown horizontally and are numbered from 1 to 8. Multiple octets are shown vertically and are numbered from 1 to n.

3.2.2 Order of bit transmission

The octets are transmitted in ascending numerical order; inside an octet, bit 1 is the first bit to be transmitted.

3.2.3 Field mapping convention

When a field is contained within a single octet, the lowest-numbered bit of the field represents the lowest-order value (or the least significant bit).

When a field spans more than one octet, the highest-numbered bit of the first octet represents the highest-order value, and the lowest-numbered bit of the last octet represents the lowest-order value.

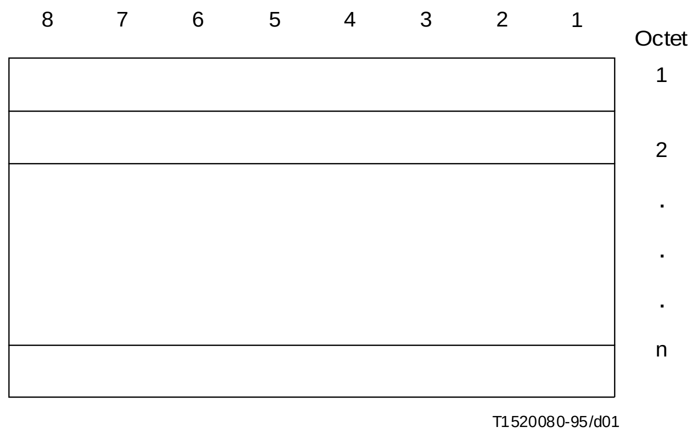


FIGURE 1/H.223

Format convention

An exception to the preceding field-mapping convention is the Cyclic Redundancy Code (CRC) field. In this case, the lowest-numbered bit of the first octet is the highest-order term of the polynomial representing the CRC field; the highest-numbered bit of the last octet is the lowest-order term of the polynomial representing the CRC field.

4 Abbreviations

For the purposes of this Recommendation, the following abbreviations are used.

AL	Adaptation Layer
AL1-AL3	Adaptation Layer 1-3
CRC	Cyclic Redundancy Code
DRTX	Decline Retransmission
EI	Error Indication
HDLC	High-level Data Link Control
HEC	Header Error Control
IWA	InterWorking Adapter
LAPM	Link Access Procedure for Modems
LCN	Logical Channel Number
MC	Multiplex Code
MUX	Multiplex
PDU	Protocol Data Unit
PM	Packet Marker
PT	Packet Type
QOS	Quality of Service
SREJ	Selective Reject
SDU	Service Data Unit
SN	Sequence Number

5 Overview

This Recommendation specifies a packet-oriented multiplexing protocol designed for the exchange of one or more information streams between higher-layer entities such as data and control protocols and audio and video codecs.

In this Recommendation, each information stream is represented by a uni-directional logical channel which is identified by a unique Logical Channel Number (LCN), an integer between 0 and 65535. LCN0 is a permanent logical channel assigned to the H.245 control channel. All other logical channels are dynamically opened and closed by the transmitter using the H.245 OpenLogicalChannel and CloseLogicalChannel messages. All necessary attributes of the logical channel are specified in the OpenLogicalChannel message. For applications that require a reverse channel, a procedure for opening bi-directional logical channels is also defined in Recommendation H.245.

The general structure of the multiplexer is shown in Figure 2. The multiplexer consists of two distinct layers: a Multiplex (MUX) layer and an Adaptation Layer (AL).

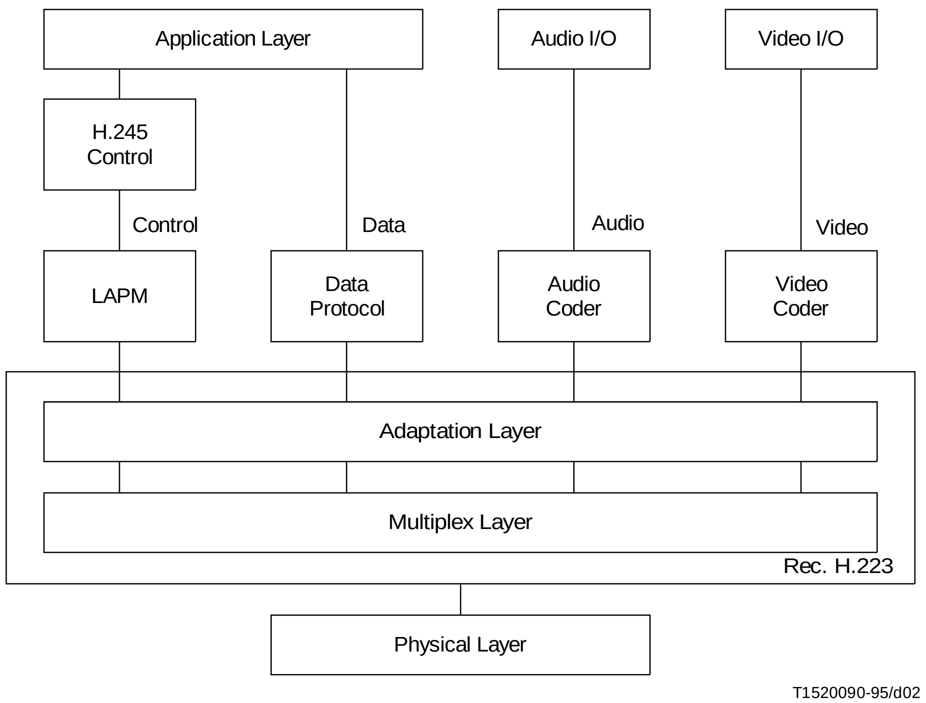


FIGURE 2/H.223

Protocol stack for Recommendation H.223

5.1 Multiplex (MUX) Layer Overview

The MUX layer is responsible for transferring information received from the AL to the far end using the services of an underlying physical layer. The MUX layer exchanges information with the AL in logical units called MUX-SDUs. MUX-SDUs always contain an integral number of octets that belong to a single logical channel. MUX-SDUs typically represent information blocks whose start and end marks the location of fields which need to be interpreted in the receiver.

MUX-SDUs are transferred by the MUX layer to the far end in one or more variable-length packets called MUX-PDUs. MUX-PDUs consist of a one-octet header, followed by a variable number of octets in the information field. MUX-PDU's are delimited by HDLC flags. The HDLC zero-bit insertion method is used to ensure that a flag is not simulated within the MUX-PDU.

Octets from multiple logical channels may be present in a single MUX-PDU information field. The header octet contains a 4-bit Multiplex Code (MC) field which specifies, by reference to a multiplex table entry, the logical channel to which each octet in the information field belongs. Multiplex table entry 0 is permanently assigned to the control channel. Other multiplex table entries are formed by the transmitter and are signalled to the far end via the control channel prior to their use.

Multiplex table entries specify a pattern of slots each assigned to a single logical channel. Any one of 16 multiplex table entries may be used in any given MUX-PDU. This allows rapid, low-overhead switching of the number of bits allocated to each logical channel from one MUX-PDU to the next. The construction of multiplex table entries and their use in MUX-PDUs is entirely under the control of the transmitter, subject to certain receiver capabilities.

When a logical channel is opened, it is designated to be either non-segmentable or segmentable. MUX-SDUs from segmentable logical channels may be broken into segments which are then transferred to the far end in one or more MUX-PDUs. Such segmentation is useful in providing improved Quality of Service (QOS), for example, by allowing the temporary suspension of the transmission of a long MUX-SDU from a segmentable data logical channel, in order to transmit a MUX-SDU from a non-segmentable audio logical channel.

5.2 Adaptation layer overview

The unit of information exchanged between the AL and the higher-layer AL users is an AL-SDU. The method of mapping information streams from higher layers into AL-SDUs is outside the scope of this Recommendation, and is specified in the System Recommendation that uses H.223.

AL-SDUs contain an integer number of octets. The AL adapts AL-SDUs to the MUX layer by adding, where appropriate, additional octets for purposes such as error detection, sequence numbering, and retransmission. The information unit exchanged between peer AL entities is called an AL-PDU. An AL-PDU is conveyed as one MUX-SDU.

Three different types of ALs, named AL1 through AL3, are specified in this Recommendation:

- AL1 is designed primarily for the transfer of data or control information. AL1 does not provide any error control; all necessary error protection should be provided by the AL1 user.

  In the framed transfer mode, AL1 receives information from its higher layer (e.g. a data link layer protocol such as LAPM/V.42 or LAPF/Q.922, which provides error control) in variable-length AL-SDUs, and simply passes these to the MUX layer in MUX-SDUs without any modifications.

  In the unframed mode, AL1 is used to transfer an unframed sequence of octets from an AL1 user. In this mode, one AL-SDU represents the entire sequence and is assumed to continue indefinitely.

- AL2 is designed primarily for the transfer of digital audio.

  AL2 receives information from its higher layer (e.g. an audio encoder) in AL-SDUs, possibly of variable-length and passes these to the MUX layer in MUX-SDUs, after adding 1 octet for an 8-bit CRC, and optionally adding 1 octet for sequence numbering.

- AL3 is designed primarily for the transfer of digital video.

  AL3 receives information from its higher layer (e.g. a video encoder) in variable-length AL-SDUs and passes these to the MUX layer in MUX-SDUs, after adding 2 octets for a 16-bit CRC, and optionally adding 1 or 2 control octets. AL3 includes a retransmission protocol designed for video.

6 Multiplex (MUX) layer specification

6.1 Framework of MUX layer

The MUX layer provides the capabilities to transfer MUX-SDUs from the sending AL to the receiving AL using the services of a physical layer below. MUX-SDUs shall always contain an integral number of octets. MUX-SDUs that belong to a given logical channel shall be transferred by the MUX layer in the same order they are received from the AL above.

6.2 Primitives exchanged between the MUX layer and the AL

The MUX layer may interface with one or more ALs. The information exchanged between the MUX layer and each individual AL includes the following primitives:

- MUX-DATA.request (MUX-SDU).
- MUX-DATA.indication (MUX-SDU).
- MUX-Abort.request.
- MUX-Abort.indication

6.2.1 Description of Primitives

- MUX-DATA.request: This primitive is issued to the MUX layer by an AL sending entity to request the transfer of a MUX-SDU to the corresponding receiving entity.
- MUX-DATA.indication: This primitive is issued by the MUX layer to an AL receiving entity to indicate the arrival of a MUX-SDU from the corresponding sending entity.
- MUX-Abort.request: This primitive is issued to the MUX layer by an AL sending entity to signal that a partially delivered MUX-SDU is to be discarded. This primitive may be used by all AL types.
- MUX-Abort.indication: This primitive is issued by the MUX layer to an AL receiving entity to signal that a partially delivered MUX-SDU is to be discarded.

6.2.2 Description of parameter

- MUX-SDU: This parameter contains the information of an integral number of octets from or to an AL. A MUX-SDU shall contain exactly one complete AL-PDU.

6.3 MUX-PDU framing

All MUX-PDUs shall be delimited using HDLC flags.

6.3.1 Flag

All MUX-PDUs shall be preceded and followed by the flag consisting of the unique bit pattern "01111110". The flag preceding the MUX-PDU is defined as the opening flag. The flag following the MUX-PDU is defined as the closing flag. The closing flag may also serve as the opening flag of the next MUX-PDU. However, all receivers conforming to this Recommendation shall accommodate receipt of more than one consecutive flag, as the flag may be transmitted repetitively between MUX-PDUs.

6.3.1.1 Transparency

The transmitter shall examine the MUX-PDU content between the opening and closing flags, and shall insert a "0" bit after all sequences of five contiguous "1" bits to ensure that a flag is not simulated within the MUX-PDU. The receiver shall examine the received bit stream between the opening and closing flags and shall discard any "0" bit which directly follows five contiguous "1" bits.

6.4 MUX-PDU format and coding

All MUX-PDUs shall conform to the format shown in Figure 3.

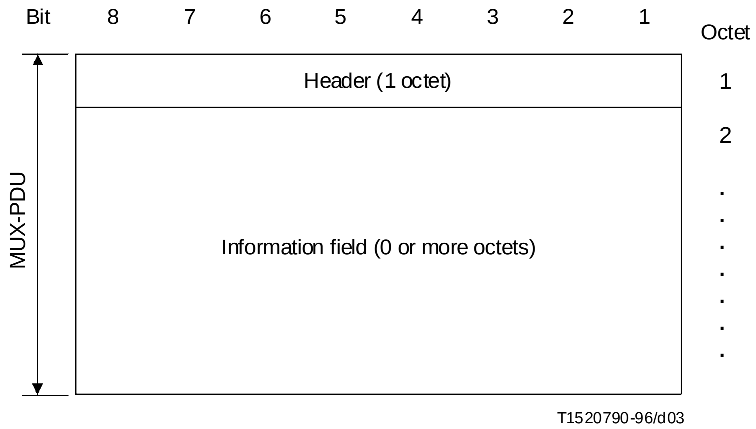
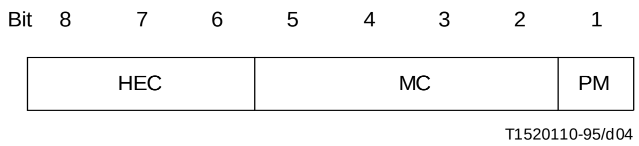


FIGURE 3/H.223

MUX-PDU format

6.4.1 Header

The format of the header shall conform to the format shown in Figure 4.



MC Multiplex Code
HEC Header Error Control
PM Packet Marker

FIGURE 4/H.223

Header format of the MUX-PDU

6.4.1.1 Multiplex Code (MC) field

The 4-bit MC field specifies to which logical channel each octet of the MUX-PDU information field belongs, by referencing an entry in the multiplex table. The field represents the number of the multiplex table entry, from 0 to 15. Multiplex table entry 0 is permanently assigned to the control channel, and shall always represent a pattern of octets assigned to the control channel (LCN0) which continue until the closing flag. Prior to their use, multiplex table entries 1 to 15 are sent to the far end in H.245 MultiplexEntrySend messages, according to the procedure and syntax described in Recommendation H.245.

Unless specified otherwise in the System Recommendation, at the start of communication, only table entry 0 is available, and table entries 1 to 15 are deactivated. The multiplex table entries used in each direction of transmission are independent of each other, and may be different.

Receivers should discard any MUX-PDU whose MC field references a deactivated multiplex table entry. Receivers should also discard any MUX-PDU which contains octets for a logical channel which is not open.

All receivers conforming to this Recommendation shall signal their capability to receive and correctly interpret either basic or enhanced MultiplexEntryDescriptors (for definitions, see Recommendation H.245) using the receive capability indication h223MultiplexTableCapability, specified in Recommendation H.245.

Receivers signalling basic h223MuxTableCapability shall be capable of receiving and correctly interpreting MultiplexEntryDescriptors which satisfy the following constraints:

- Maximum elementList size: 2
- Maximum nesting depth: 1
- Maximum subElementList size: 2

and whose first MultiplexElement in the elementList does not use a non-segmentable logical channel more than once and whose second MulitplexElement in the elementList uses only segmentable logical channels.

Receivers signalling enhanced capability shall signal their capability to receive and correctly interpret MultiplexEntry Descriptors, according to the H.245 h223MultiplexTableCapability indication. A receiver signalling enhanced capability is also capable of receiving and correctly interpreting all MultiplexEntryDescriptors that are included in the basic capability.

NOTE – The MC field of each MUX-PDU should be selected in a manner that provides each information stream the Quality of Service (QOS) it needs. This is the responsibility of a local multiplexing implementation which is outside the scope of this Recommendation.

6.4.1.2 Header Error Control (HEC) field

The 3-bit HEC field provides error detection capabilities over the MC field using a 3-bit CRC.

The HEC field shall contain the remainder of the division (modulo 2) by the generator polynomial $P(x) = x^3 + x + 1$ of the product x^3 multiplied by the content of the MC field. The polynomial representing the content of the MC field is generated using bit number 2 (i.e. the least significant bit) in the MC field as the coefficient of the highest-order term. The polynomial representing the content of the CRC field is generated using bit number 6 (i.e. the least significant bit) as the coefficient of the highest-order term. Table 1 shows the values of the 3-bit HEC field as a function of the 4-bit MC field.

Receivers should discard any MUX-PDU whose HEC field fails the error check.

6.4.1.3 Packet Marker (PM) field

The 1-bit PM field shall be used to mark the end of MUX-SDUs of segmentable logical channels, as described in 6.5.

TABLE 1/H.223

Values of the HEC field as a function of the values of the MC field

MC Field	HEC Field
Bit number 5 4 3 2	Bit number 8 7 6
0 0 0 0	0 0 0
0 0 0 1	1 0 1
0 0 1 0	1 1 1
0 0 1 1	0 1 0
0 1 0 0	0 1 1
0 1 0 1	1 1 0
0 1 1 0	1 0 0
0 1 1 1	0 0 1
1 0 0 0	1 1 0
1 0 0 1	0 1 1
1 0 1 0	0 0 1
1 0 1 1	1 0 0
1 1 0 0	1 0 1
1 1 0 1	0 0 0
1 1 1 0	0 1 0
1 1 1 1	1 1 1

6.4.2 Information field

The multiplex table entry selected by the MC field specifies the multiplexing pattern for the information field, according to the multiplex table entry syntax described in Recommendation H.245. Octets from multiple logical channels may be present in the information field. The information field may be terminated on any octet boundary by closing the MUX-PDU with a closing flag, except a MUX-SDU from a non-segmentable logical channel shall not be interrupted.

The procedure given in this paragraph is optional and is used only when it is required by the System Recommendation that uses Recommendation H.223: when this option is used, the transmitter shall exclusive-OR each octet in the information field with the octet 000uxyz0 prior to applying the transparency procedure where “uxyz” represent the bits of the MC field where z corresponds to the least significant bit (bit number 2) of the field. The receiver shall perform the same operation to restore the original information field content. This procedure is followed to ensure that errors affecting the MC field will alter the received information field octets, with high probability ensuring the failure of any CRC checks applied to the information field content.

NOTE 1 – When the above procedure is not used, the multiplexer should be designed such that in case of undetected errors affecting the MC field, the failure of any CRC checks applied to the information field is ensured with high probability.

The length of the information field is not limited, however transmitters should consider the error characteristics of the underlying physical medium when choosing the length of the information field. In the event of bit errors that affect the MC field, the entire MUX-PDU may be lost.

NOTE 2 – In the receiver, the MUX layer may pass the octets of the information field to the AL in a “streaming” mode before having the complete MUX-PDU in its possession.

6.4.3 Abort

A MUX-PDU which has no information field shall be interpreted in the receiver as an abort, if its PM field is set to "0" and its MC field is the same as that of the previously received MUX-PDU. The MUX-SDU to be aborted is the one which occupied the last octet in the previously received MUX-PDU.

6.5 Marking of MUX-SDU boundaries

It is necessary to detect the boundaries of MUX-SDUs in the receiver in order to identify the location of all fields which the receiver must interpret in the AL and/or in a frame-oriented higher-layer. This shall be accomplished as follows:

For non-segmentable logical channels, each MUX-SDU shall begin coincident with a slot specified in a single MultiplexElement structure whose type is logicalChannelNumber (see Recommendation H.245), and shall end after the specified repeatCount, or at the closing flag of the current MUX-PDU, whichever occurs first. The actual length of the MUX-SDU may be smaller than the length of the slot, provided that the current MUX-PDU is terminated by a closing flag immediately after the MUX-SDU. Since the size of each MUX-SDU may vary, multiple multiplex table entries may be defined to match the possible lengths of MUX-SDUs, in order to mix these MUX-SDUs with octets from other logical channels. It should be noted that the definitions given here together with the conditions given in Recommendation H.245 imply that it is allowed to place more than one MUX-SDU from a non-segmentable logical channel in one MUX-PDU, but only when the remote receiver has indicated the enhanced multiplex capability.

For segmentable logical channels, each MUX-SDU may be broken into segments and these segments may be transferred in one or more MUX-PDUs. The PM field in the MUX-PDU header shall be used to mark the end of each MUX-SDU. Specifically, the PM field shall be set to "1" to indicate that the last octet of the previous MUX-PDU was the final octet of the terminating MUX-SDU. As a result of this procedure, only one segmentable MUX-SDU is permitted to terminate within a MUX-PDU; as soon as the end of any MUX-SDU from a segmentable logical channel is reached, the MUX-PDU shall be terminated with a closing flag and the PM field in the next MUX-PDU shall be set to "1". In all other circumstances, the PM field shall be set to "0". Another result of this procedure is that a MUX-PDU will never contain octets from two different MUX-SDUs of the same segmentable logical channel.

An empty MUX-PDU with no information field shall be transmitted for the purpose of terminating a MUX-SDU from a segmentable logical channel, if the transmitter has no information to send immediately after closing the MUX-PDU. The PM field of this MUX-PDU shall be set to "1", and the MC field shall be the same as that of the previous MUX-PDU.

6.6 Examples

Table 2 includes examples of MultiplexEntryDescriptors which include 1, 2 or 3 MultiplexElements in the elementList. Each row in the table corresponds to a MultiplexEntryDescriptor. For each MultiplexEntryDescriptor, the number of MultiplexElements in the elementList, the nesting depth and the subelementList size are given in separate columns.

Five logical channels are assumed as follows: LCN0: control, LCN1: audio I, LCN2: data, LCN3: video, LCN4: audio II. Audio logical channels are designated as non-segmentable, and all others are designated as segmentable.

The first five rows show examples of basic MultiplexEntryDescriptors:

The first two rows show how the entire MUX-PDU information field can be assigned to a single logical channel. It should be noted that the entry shown in row 1 can be used to send audio MUX-SDUs of any length, but not to send more than one audio MUX-SDU.

The third row illustrates how video may be transmitted after an audio MUX-SDU in a single MUX-PDU.

The fourth row shows how data and video may be mixed using a repeating pattern of 1 octet of data and 3 octets of video.

The fifth row shows how a short audio MUX-SDU, possibly representing background noise information sent during a silence period, may be mixed using a repeating pattern of data and video. This entry is used later in this subclause to illustrate the construction of the information field.

The last three rows show examples of enhanced MultiplexEntryDescriptors:

The sixth row illustrates how audio may be mixed with octets from video, data and control channels.

The seventh row shows a MultiplexEntryDescriptor with 3 MultiplexElements used to send two audio MUX-SDUs from two different audio logical channels, mixed with octets from data and video channels.

Finally, the eighth row shows an example of 2-level nesting, where an audio MUX-SDU is followed by an alternating pattern of data and video octets which repeats five times, and where the entire pattern including the audio MUX-SDU repeats untilClosingFlag.

TABLE 2/H.223

Examples of MultiplexEntryDescriptors
(LCN: logicalChannelNumber, RC: repeat Count, UCF: untilClosingFlag)

Row	MultiplexEntryDescriptor	Element ListSize	Nesting Depth	Subelement ListSize	Example
1	{LCN1,RC UCF}	1	0	0	All audio
2	{LCN3,RC UCF}	1	0	0	All video
3	{LCN1,RC21},{LCN3,RC UCF}	2	0	0	Audio, All video
4	{{LCN2,RC1},{LCN3,RC3},RC UCF}	1	1	2	1:3 data video
5	{LCN1,RC4},{{LCN2,RC1}, {LCN3,RC2},RC UCF}	2	1	2	Audio, 1:2 data video
6	{LCN1,RC21},{{LCN2,RC2}, {LCN3,RC6},{LCN0,RC1}RC UCF}	2	1	3	Audio, 2:6:1 data video control
7	{LCN1,RC21},{LCN4,RC25}, {{LCN2,RC1},{LCN3,RC1}RC UCF}	3	1	2	Audio I, Audio II, 1:1 data video
8	{{LCN1,RC25},{{LCN2,RC1}, {LCN3,RC1},RC5},RC UCF}	1	2	2	2-level nesting

Figure 5 shows an example of the construction of the information field from the MultiplexEntryDescriptor and illustrates the use of the PM field. The MultiplexEntryDescriptor shown in row 5 of Table 2 is used in this example.

Suppose that in this example at one instant of time, the multiplexer has three MUX-SDUs ready for transmission: a 4-octet MUX-SDU from LCN1, a 3-octet MUX-SDU from LCN2 and a 3-octet MUX-SDU from LCN3.

The MUX-PDU is formed starting with the 4-octet MUX-SDU from LCN1, and continuing with a 1-octet segment from LCN2, a 2-octet segment from LCN3, a 1-octet segment from LCN2 and another 1-octet segment from LCN3. Since the end of the MUX-SDU from LCN3 is then reached, the MUX-PDU is closed with a flag, and the PM field in the next MUX-PDU is set. The last octet of the MUX-SDU from LCN2 may be transmitted in any of the following MUX-PDUs.

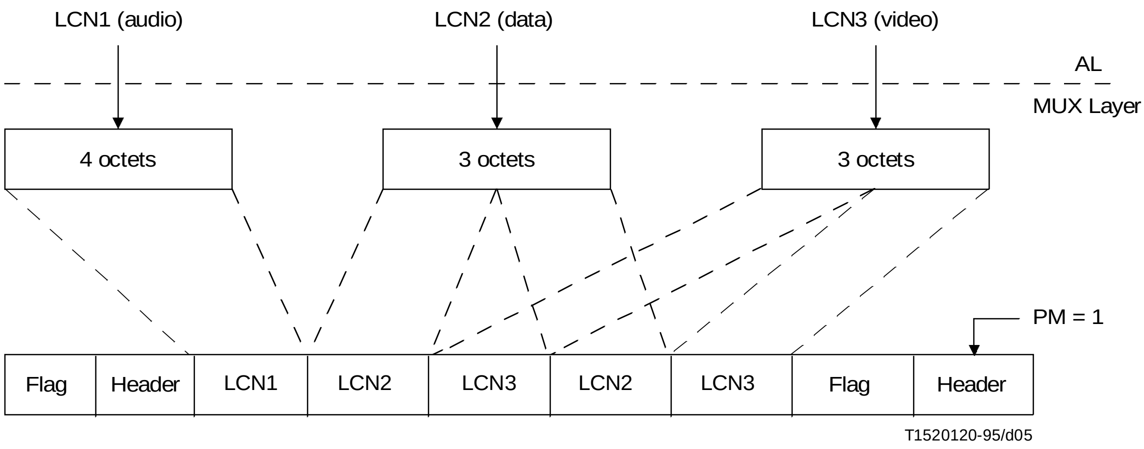


FIGURE 5/H.223

Information field example

7 Adaptation Layer (AL) specification

7.1 Introduction

This clause describes the interactions between the AL and the higher layer above, and the AL and the MUX layer, as well as end-to-end operations between peer ALs. The Adaptation Layer (AL) enhances the services provided by the underlying MUX layer to support functions required by AL users and to support the mapping between the MUX layer and the layer above. Three different types of ALs, named AL1, AL2 and AL3, are specified.

The AL is selected by the transmitter using the OpenLogicalChannel message of Recommendation H.245 at the time a logical channel is opened. Any one of the three types of ALs may be used to carry a given logical channel subject to restrictions that may be imposed by the System Recommendation that uses Recommendation H.223. The AL includes a few optional fields which are selected by the transmitter in the OpenLogicalChannel message at the time a logical channel is opened.

The unit of information exchanged between the AL and a higher-layer entity is called an AL-SDU. AL-SDUs may be of variable length. The maximum length of AL-SDUs is determined by the AL user. The method of mapping the information stream from the higher layer to AL-SDUs is outside the scope of this Recommendation, and is defined in the System Recommendation (e.g. Recommendation Recommendation H.324) that uses Recommendation H.223. AL-SDUs that belong to a given logical channel shall be transferred by the AL in the same order they are received from the higher-layer entity. AL transfers a complete AL-SDU received from the AL user in a single AL-PDU. An AL-PDU is mapped directly to a single MUX-SDU, the parameter of the MUX layer primitive, and vice versa.

7.2 Adaptation Layer Type 1 (AL1) specification

7.2.1 Framework of AL1

AL1 is designed primarily for the transfer of data or control information.

AL1 does not provide any error detection or correction capability. Therefore, any necessary error control, possibly including a retransmission procedure, should be provided by the higher layer.

AL1 provides two transfer modes:

a) framed transfer mode; and

b) unframed transfer mode.

In the framed transfer mode, AL1 may be used to transfer frames generated by a higher-layer protocol such as the data link layer protocol LAPM/V.42 or LAPF/Q.922. In this case, frames are first mapped to AL-SDUs and these are then passed by AL1 in MUX-SDUs to the MUX layer.

AL1 may also be used to carry an unframed octet sequence. In this mode, any internal framing present in the octet sequence is not visible to AL1 which passes the octets received from the higher layer to the MUX layer without paying any attention to framing.

The transfer mode of AL1 is selected by the transmitter in the H.245 OpenLogicalChannel message.

Logical channels which are transferred by AL1 using the unframed transfer mode shall be designated as segmentable, so that the octet transmission can be interrupted to send octets from other information streams. However, since the AL-SDU continues indefinitely, for such logical channels, the PM field shall never be set to "1".

7.2.2 Primitives exchanged between AL1 and AL1 user

The information exchanged between AL1 and the AL1 user includes the following primitives:

- AL-DATA.request (AL-SDU).
- AL-DATA.indication (AL-SDU).
- AL-Abort.request.
- AL-Abort.indication.

7.2.2.1 Description of primitives

- AL-DATA.request: This primitive is issued by an AL1 user to AL1 to request the transfer of an AL-SDU to its corresponding receiving entity.
- AL-DATA.indication: This primitive is issued to an AL1 user by AL1 to indicate the arrival of an AL-SDU.
- AL-Abort.request: This primitive is issued to AL1 by an AL1 user to signal that a partially delivered AL-SDU is to be aborted. This primitive is not used in the unframed transfer mode.
- AL-Abort.indication: This primitive is issued by AL1 to an AL1 user to signal that a partially delivered AL-SDU is to be aborted. This primitive is not used in the unframed transfer mode.

7.2.2.2 Description of parameter

- AL-SDU: This parameter specifies the unit of information exchanged between AL1 and the AL1 user. Each AL-SDU shall contain an integral number of octets. The length of AL-SDUs may be variable. The maximum size of AL-SDUs shall be determined by the AL1 user. The octets in an AL-SDU are numbered from 1 to n, and in each octet, bits are numbered from 1 to 8. Bit 1 of octet 1 is transmitted first.

7.2.3 Procedures for Abort

Abort procedures may be used when information is exchanged between layers in a streaming mode.

When an AL-Abort.request primitive is sent from the AL1 user to AL1 in order to abort a partially delivered AL-SDU, AL1 shall immediately send a MUX-Abort.request primitive to the MUX layer, if a MUX-SDU containing that AL-SDU has already been partially delivered to the MUX layer.

In the AL1 receiver, when a MUX-Abort.indication primitive is received from the MUX layer, AL1 shall immediately send an AL-Abort.indication primitive to the AL1 user, if that AL-SDU has already been partially delivered to the AL1 user.

Abort procedures shall not be used in the unframed transfer mode.

7.3 Adaptation Layer Type 2 (AL2) specification

7.3.1 Framework of AL2

AL2 is designed primarily for the transfer of digital audio.

AL2 provides an 8-bit CRC for error-detection. AL2 also supports optional sequence numbering which may be used to detect missing and misdelivered AL-PDUs. AL2 transfers variable-length AL-SDUs of integral number of octets.

7.3.2 Primitives exchanged between AL2 and AL2 user

The information exchanged between AL2 and an AL2 user includes the following primitives:

- AL-DATA.request (AL-SDU).
- AL-DATA.indication (AL-SDU, EI).
- AL-Abort.request.

7.3.2.1 Description of primitives

- AL-DATA.request: This primitive is issued by an AL2 user to AL2 to request the transfer of an AL-SDU to a corresponding AL2 user.
- AL-DATA.indication: This primitive is issued to an AL2 user by AL2 to indicate the arrival of an AL-SDU.
- AL-Abort.request: This primitive is issued to AL2 by an AL2 user to signal that a partially delivered AL-SDU is to be aborted.

7.3.2.2 Description of parameters

- AL-SDU: This parameter specifies the unit of information exchanged between AL2 and the AL2 user. Each AL-SDU shall contain an integral number of octets. The length of AL-SDUs may be variable. The maximum length of AL-SDUs that an AL2 receiver can accept shall be signalled via the H.245 control channel. The octets in an AL-SDU are numbered from 1 to n, and in each octet, bits are numbered from 1 to 8. Bit 1 of the octet 1 is transmitted first. An AL2 receiving entity may deliver an empty AL-SDU to the AL2 user to indicate that an AL-SDU is missing.
- Error Indication (EI): This parameter may be used in the AL2 receiver to pass error indications to the AL2 user. The precise procedures for using this parameter, and its numerical coding, are outside the scope of this Recommendation.

7.3.3 Functions, format and coding of AL2

7.3.3.1 Functions of AL2

AL2 provides the following functions:

- Error detection and indication;
- Optional sequence numbering.

7.3.3.2 Format and coding of AL2

The format of the AL-PDU is illustrated in Figure 6.

7.3.3.2.1 Sequence Number (SN) field

The optional 8-bit SN provides a capability for sequencing AL-PDUs. The sequence number may be used by the AL2 receiving entity to detect missing and misdelivered AL-PDUs.

All receivers conforming to this Recommendation shall be capable of receiving and correctly interpreting AL-PDUs that include the SN field. The use of the SN field shall be determined by the transmitter and shall be signalled to the far end in the OpenLogicalChannel message of Recommendation H.245.

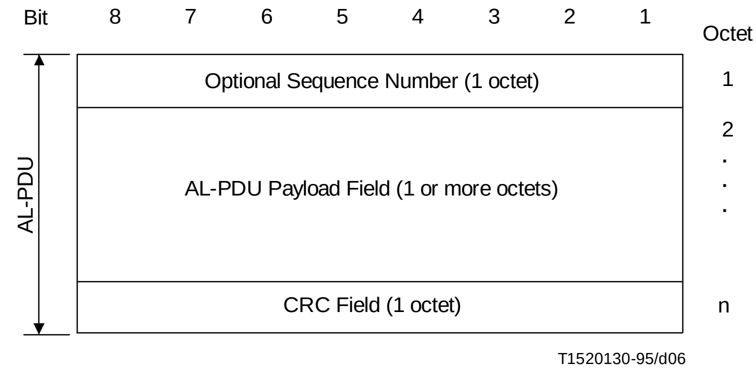


FIGURE 6/H.223

AL-PDU format for AL2

When the SN field is in use, the AL2 receiver may detect that an AL-PDU is missing or has been misdelivered by the MUX layer. The AL2 receiver should discard any misdelivered AL-PDUs that it detects.

7.3.3.2.2 AL-PDU payload field

The AL-PDU payload field contains a complete AL-SDU, where the first octet corresponds to the first octet of the AL-SDU.

7.3.3.2.3 CRC field

The 8-bit CRC provides an error detection capability across the entire AL-PDU.

The 8-bit CRC field shall contain the remainder of the division (modulo 2) by the generator polynomial $p(x) = x^8 + x^2 + x + 1$ of the product x^8 multiplied by the content of the AL-PDU, excluding the CRC field, and including the SN field, if it is used. The polynomial representing the content of the AL-PDU is generated using bit number 1 of the first octet as the coefficient of the highest-order term.

As a typical implementation in the transmitter, the initial content of the register of the device computing the remainder of the division is preset to all 0's and is then modified by division by the generator polynomial (as described above) of the content of the AL-PDU, not including the bits in the CRC field; the resulting remainder is transmitted as the 8-bit CRC. The coefficient of the highest-order term of the remainder polynomial corresponds to bit number 1 of the CRC field.

NOTE – In contrast to the CRC procedure used for the 16-bit CRC in AL3, the CRC procedure used here does not include any pre- or post-conditioning.

7.3.4 Procedures for Abort

Abort procedure may be used when information is exchanged between layers in a streaming mode.

When an AL-Abort.request primitive is sent from the AL2 user to AL2 in order to abort a partially delivered AL-SDU, AL2 shall immediately send a MUX-Abort.request primitive to the MUX layer, if that AL-SDU has already been partially delivered to the MUX layer.

In the AL2 receiver, when a MUX-Abort.indication primitive is received from the MUX layer, any partially received AL-PDU should be discarded.

7.3.5 Procedures for sequence numbering

The following procedures apply when the SN field is in use.

Once a logical channel using AL2 is successfully opened according to the procedure defined in Recommendation H.245, the first AL-PDU transmitted by the AL2 sending entity shall have the SN field set to 0. For each subsequent transmitted AL-PDU which belongs to that logical channel, the value of the SN field shall be incremented by 1 modulo 256.

7.3.6 Procedures for error control

When the CRC check fails at the AL2 receiver, the associated AL-SDU may be delivered to the AL2 user, together with an appropriate error indication, via the AL-DATA.indication primitive.

When the SN field is in use, the AL2 receiver may detect that an AL-PDU is missing or has been misdelivered by the MUX layer. The AL2 receiver should discard any misdelivered AL-PDUs that it detects. For each missing AL-PDU that it detects, the AL2 receiver may deliver to the AL2 user an empty AL-SDU, together with an appropriate error indication, via the AL-DATA.indication primitive.

7.4 Adaptation Layer Type 3 (AL3) specification

7.4.1 Framework of AL3

AL3 is designed primarily for the transfer of digital video.

AL3 includes a 16-bit CRC for error-detection. AL3 also supports optional sequence numbering which may be used to detect missing and misdelivered AL-PDUs. AL3 transfers variable-length AL-SDUs and provides an optional retransmission procedure, designed primarily for video.

7.4.2 Primitives exchanged between AL3 and AL3 user

The information exchanged between AL3 and the AL3 user includes the following primitives:

- AL-DATA.request (AL-SDU).
- AL-DATA.indication (AL-SDU, EI).
- AL-Abort.request.
- AL-DRTX.indication.

7.4.2.1 Description of primitives

- AL-DATA.request: This primitive is issued by an AL3 user to AL3 to request the transfer of an AL-SDU to a corresponding AL3 user.
- AL-DATA.indication: This primitive is issued to an AL3 user by AL3 to indicate the arrival of an AL-SDU.
- AL-Abort.request: This primitive is issued to AL3 by an AL3 user to signal that a partially delivered AL-SDU is to be aborted.
- AL-DRTX.indication: This primitive is issued to an AL3 user by AL3 to indicate that a declined retransmission condition has occurred in the local transmitter.

7.4.2.2 Description of parameters

- AL-SDU: This parameter specifies the information exchanged between AL3 and the AL3 user. The length of the AL-SDU may be variable. Each transmitted AL-SDU shall contain an integral number of octets. The maximum size of AL-SDUs that an AL3 receiver can accept shall be signalled via the H.245 control channel.

  An AL3 receiving entity may deliver an empty AL-SDU to the AL3 user to indicate that an AL-SDU has been lost.
- Error Indication (EI): This parameter may be used by the AL3 receiver to pass error indications to the AL3 user. The precise procedures for using this parameter, and its numerical coding, are outside the scope of this Recommendation.

7.4.3 Functions, format and coding of AL3

7.4.3.1 Functions of AL3

AL3 provides the following functions:

- Error detection and indication.
- Optional sequence numbering.
- Optional support of retransmission.

7.4.3.2 Format and coding of AL3

The format of the AL-PDU is illustrated in Figure 7.

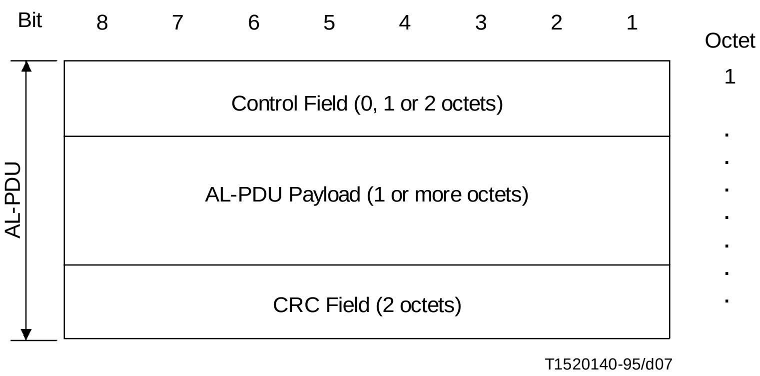
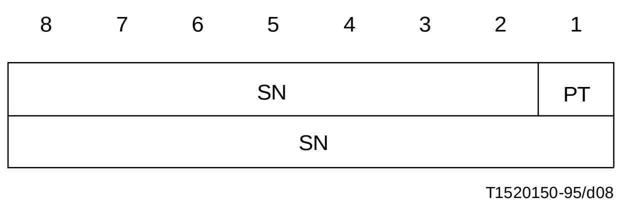


FIGURE 7/H.223

AL-PDU format for AL3

7.4.3.2.1 Control field

The optional control field consists of a Payload Type (PT) field, which indicates the function of the AL-PDU payload, and a Sequence Number (SN) field, as shown in Figure 8.



SN Sequence Number
PT Payload Type

FIGURE 8/H.223

Control field format of the AL-PDU for AL3

All receivers conforming to this Recommendation shall be capable of receiving and correctly interpreting AL-PDUs with 0-, 1- or 2-octet control fields. The actual number of octets in the control field is determined by the transmitter and shall be signalled to the far end in the OpenLogicalChannel message of Recommendation H.245.

When the control field is absent, the retransmission procedure is not used. However, the System Recommendation that uses Recommendation H.223 may require the control field to be present.

7.4.3.2.1.1 Payload Type (PT) field

The 1-bit PT field indicates the payload type of the AL-PDU. When the PT field is set to "1", the AL-PDU payload field shall contain an AL-SDU. Such an AL-PDU is referred to as an I-PDU. When the PT field is set to "0", the AL-PDU payload field shall contain a supervisory message used in the retransmission procedure. Such an AL-PDU is referred to as an S-PDU.

7.4.3.2.1.2 Sequence Number (SN) field

The sequence number field shall be 7 or 15 bits, depending on the length of the control field. In I-PDUs, the SN field shall contain a send sequence number N(S). In S-PDUs, the SN field shall contain a receive sequence number N(R) of an I-PDU as defined in 7.4.6.1.6.

Using the SN field, the AL3 receiver may detect that an AL-PDU is missing or has been misdelivered by the MUX layer.

The AL3 receiver should discard any misdelivered AL-PDUs that it detects.

7.4.3.2.2 AL-PDU payload field

The payload field of an I-PDU shall contain a complete AL-SDU received from the AL3 user where the first octet of the AL-PDU payload field shall be the first octet of the AL-SDU.

The 1-octet payload field of an S-PDU carries a supervisory message as defined in 7.4.6.2.

7.4.3.2.3 CRC field

The 16-bit CRC provides an error detection capability across the entire AL-PDU, including the control field, if used. The CRC and the CRC procedures are the same as those used in LAPM/V.42 and LAPF/Q.922.

The CRC has generator polynomial $g(x) = x^{16} + x^{12} + x^5 + 1$.

The CRC field shall be the one's complement of the sum (modulo 2) of

a) the remainder of $x^k (x^{15} + x^{14} + x^{13} + x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1)$ divided (modulo 2) by the generator polynomial $g(x) = x^{16} + x^{12} + x^5 + 1$, where k is the number of bits in the AL-PDU, not including the bits in the CRC field; and

b) the remainder of the division (modulo 2) by the generator polynomial $g(x) = x^{16} + x^{12} + x^5 + 1$, of the product of x^{16} multiplied by the content of the AL-PDU, excluding the bits in the CRC field. The polynomial representing the content of the AL-PDU is generated using bit number 1 of the first octet as the coefficient of the highest-order term.

As a typical implementation in the transmitter, the initial content of the register of the device computing the remainder of the division is preset to all 1's and is then modified by division by the generator polynomial (as described above) of the content of the AL-PDU, not including the bits in the CRC field; the one's complement of the resulting remainder is transmitted as the 16-bit CRC. The one's complement of the coefficient of the highest-order term of the remainder polynomial corresponds to bit number 1 of the first octet of the 16-bit CRC field. The one's complement of the coefficient of the lowest-order term of the remainder polynomial corresponds to bit number 8 of the second octet of the 16-bit CRC field.

NOTE – In contrast to the CRC procedure used for the 8-bit CRC in AL2, the CRC procedure used here includes pre- and post-conditioning.

7.4.4 Procedures for Abort

Abort procedures may be used when information is exchanged between layers in a streaming mode.

When an AL-Abort.request primitive is sent from the AL3 user to AL3 in order to abort a partially delivered AL-SDU, AL3 shall immediately send a MUX-Abort.request primitive to the MUX layer, if that AL-SDU has already been partially delivered to the MUX layer.

In the AL3 receiver, when a MUX-Abort.indication primitive is received from the MUX layer, any partially received AL-PDU should be discarded.

7.4.5 Procedures for Error Control

7.4.5.1 Invalid AL-PDUs

An invalid AL-PDU is one which:

a) has fewer than the minimum number of octets specified in 7.4.3.2, depending on the length of the control field; or

b) does not contain an integral number of octets; or

c) is longer than the maximum AL-PDU size; or

d) contains a CRC error.

An AL-PDU which is not invalid is referred to as a valid AL-PDU.

7.4.5.2 Error control control field absent

When the control field is absent, in case of a CRC failure at the AL3 receiver, the associated AL-SDU may be delivered to the AL3 user, together with an appropriate EI parameter, via the AL-DATA.indication primitive.

7.4.5.3 Error control control field present

When the control field is present, the AL3 receiver has the option of invoking the retransmission procedure. The sending AL3 entity shall respond to a retransmission request according to the procedures defined in 7.4.6.3.4. The error control procedures for retransmission are described in 7.4.6.

7.4.5.3.1 No retransmission

When the control field is in use and the AL3 receiver does not invoke the retransmission procedure, the following error control procedures may be used.

When the CRC check fails at the AL3 receiver, the associated AL-SDU may be delivered to the AL3 user, together with an appropriate EI parameter, via the AL-DATA.indication primitive.

Using the SN field, the AL3 receiver may detect that an AL-PDU is missing or has been misdelivered by the MUX layer.

The AL3 receiver should discard any misdelivered AL-PDUs that it detects.

For each missing AL-PDU that it detects, the AL3 receiver may deliver to the AL3 user an empty AL-SDU, together with an appropriate EI parameter, via the AL-DATA.indication primitive.

7.4.6 Retransmission procedure

The transmitter procedures defined in this subclause shall be used when the control field is present. The receiver procedures defined in this subclause shall be used when retransmission is used.

7.4.6.1 Definitions

7.4.6.1.1 Modulus

Each I-PDU, defined in 7.4.3.2.1.1, is sequentially numbered modulo 128 (2^{15}) and may have the value 0 through 127 (32767).

NOTE – All arithmetic operations on state variables and sequence numbers contained in this section are modulo 128 (2^{15}).

7.4.6.1.2 Send state variable V(S)

V(S) is an internal variable of the transmitting AL3 entity. It denotes the sequence number of the next I-PDU to be transmitted. V(S) can take on the values 0 through 127 (32767). The value of V(S) shall be incremented by 1 after each in-sequence I-PDU is passed to the MUX layer in a MUX-SDU.

7.4.6.1.3 Send sequence number N(S)

Only I-PDUs contain N(S), the send sequence number of transmitted I-PDUs. At the time that an in-sequence I-PDU is designated for transmission, the value of N(S) is set equal to V(S).

7.4.6.1.4 Send buffer B_S

Each AL3 entity shall maintain a send buffer, B_S, used for storing the most recently transmitted I-PDUs. The minimum size of B_S that all AL3 transmitters must support is specified in the System Recommendation (e.g. Recommendation H.324) that uses H.223. The actual size of B_S shall be indicated to the far end in the H.245 OpenLogicalChannel message.

7.4.6.1.5 Receive state variable V(R)

V(R) is an internal variable of the AL3 receiving entity. It denotes the sequence number of the next in-sequence I-PDU expected to be received. V(R) can take on the values 0 through 127 (32767). The value of V(R) shall be incremented by 1 with the receipt of a valid, in-sequence I-PDU whose N(S) equals V(R).

7.4.6.1.6 Receive sequence number N(R)

Only S-PDUs contain N(R), the send sequence number of an I-PDU that is referred to by the S-PDU.

7.4.6.2 Supervisory messages

S-PDUs carry supervisory messages. Each S-PDU contains one single-octet message. Table 3 shows the code assignment for the single-octet supervisory messages defined in AL3.

TABLE 3/H.223

Code Assignment for supervisory messages

Supervisory Message	Code Number	Binary Code
SREJ	0	00000000
DRTX	255	11111111
Reserved	1-254	

7.4.6.2.1 Selective reject (SREJ) message

SREJ is used by an AL3 receiver to request the retransmission of the single I-PDU numbered N(R).

An SREJ PDU shall not be transmitted more than once for the same I-PDU.

7.4.6.2.2 Declined retransmission (DRTX) message

Since the error recovery procedures defined here only support negative acknowledgment, in certain conditions, the I-PDU(s) transmitted previously may have been discarded before the request for retransmission is received. The DRTX message is used by an AL3 transmitter to decline the requested retransmission of an I-PDU, when that I-PDU is not available in the send buffer at the time the SREJ PDU is received.

7.4.6.3 Detailed procedures

7.4.6.3.1 Initialization procedures

The retransmission procedures require that a reverse logical channel exist for sending supervisory messages.

Once the reverse logical channel has been established according to the procedure defined in Recommendation H.245, the AL3 entity shall:

– set V(S), V(R) to 0;

– clear any existing exception conditions.

7.4.6.3.2 Transmitting in-sequence I-PDUs

Information received from the AL3 user in an AL-SDU by means of an AL-DATA.request primitive shall be passed to the MUX layer in an I-PDU using the frame structure defined in 7.4.3.2. The SN field of the I-PDU shall be assigned the value V(S). V(S) shall be incremented by 1 after the I-PDU has been passed to the MUX layer.

7.4.6.3.3 Receiving in-sequence I-PDUs

When an AL3 entity receives a valid I-PDU, whose N(S) is equal to the current V(R), the AL3 entity shall increment its V(R) by 1.

7.4.6.3.4 Receiving SREJ PDUs

On receipt of a valid SREJ PDU, the AL3 entity shall act as follows:

a) if the I-PDU whose N(S) is equal to the N(R) of the SREJ PDU is still in the send buffer, the AL3 entity shall pass the corresponding I-PDU to the MUX layer as soon as possible. No other previously transmitted I-PDUs shall be retransmitted as a result of receiving the SREJ PDU.

b) if the I-PDU whose N(S) is equal to the N(R) of the SREJ PDU has been previously discarded, the AL3 entity shall enter a declined-retransmission exception condition. The procedures for this exception condition are defined in 7.4.6.4.5.

7.4.6.4 Exception condition reporting and recovery

Exception conditions may occur as a result of errors on the physical connection or procedural errors by an AL3 entity.

The error-recovery procedures that are available following the detection of an exception condition by an AL3 entity are defined in this subclause.

7.4.6.4.1 Receiving invalid AL-PDUs

When a received AL-PDU is invalid, it is either discarded or saved for possible delivery later to the AL3 user.

7.4.6.4.2 N(S) sequence error

When there are no other outstanding exception conditions, an N(S) sequence error exception condition occurs in the receiving AL3 entity when a valid I-PDU is received containing an N(S) value that is not equal to the V(R) at the receiver. In this case, V(R) shall not be incremented, and one or more SREJ PDUs, each containing a different N(R), may be transmitted by the AL3 receiving entity to initiate an exception condition recovery for each SREJ PDU. After passing each SREJ PDU to the MUX layer, the AL3 entity shall start a local timer. Several factors that affect the length of the timer are given in Appendix IV/V.42. A different timer is maintained for each outstanding SREJ PDU. Successive SREJ PDUs are transmitted in the order indicated by their N(R) field.

For each SREJ PDU that it transmits, the AL3 receiver may pass an empty AL-SDU or an invalid received AL-SDU (previously saved), with an appropriate EI parameter, to the AL3 user via the AL-DATA.indication primitive.

When the retransmitted I-PDU with N(S) = V(R) is received, the exception condition for that I-PDU shall be cleared. The AL3 receiver should pass the associated AL-SDU, together with an appropriate EI parameter, to the AL3 user via the AL-DATA.indication primitive. When the exception condition is cleared, the associated timer shall be stopped and V(R) shall be incremented as many times as necessary so that V(R) represents the send sequence number of the next expected in-sequence I-PDU.

When a retransmitted I-PDU with N(S) ≠ V(R) is received, the AL3 receiver shall clear all exception conditions for any SREJ PDUs that may have been sent before the SREJ PDU for which the retransmission is received, by stopping the associated timers. For each exception condition cleared, the AL3 receiver shall increment V(R) by 1, and may deliver an empty AL-SDU, together with an appropriate EI parameter, to the AL3 user via the AL-DATA.indication primitive, prior to delivering the AL-SDU associated with the received I-PDU.

The information in all other received valid I-PDUs should be delivered to the AL3 user in AL-SDUs, together with an appropriate EI parameter.

7.4.6.4.3 N(R) sequence error

An N(R) sequence error exception condition occurs, when a valid S-PDU is received that contains an invalid N(R) value. An invalid N(R) value will result, when a first SREJ PDU is received with sequence number N(R) = N1, and then another SREJ PDU is received with N(R) = N2 and [V(S) – N2] is greater than or equal to [V(S) – N1].

An invalid N(R) may also result, when the N(R) value in a DRTX PDU is not equal to the N(R) value in an outstanding SREJ PDU.

The AL3 entity should ignore the message in such S-PDUs.

7.4.6.4.4 Procedure on expiration of timer

If the timer expires, the associated exception condition shall be cleared by stopping the timer and incrementing V(R). The AL3 receiver may pass an empty AL-SDU or an invalid received AL-SDU (previously saved), with an appropriate error indication, to the AL3 user via the AL-DATA.indication primitive.

7.4.6.4.5 Declined-retransmission condition

7.4.6.4.5.1 Error recovery procedures at the AL3 transmitter

Upon receiving an SREJ retransmission request, when the AL3 transmitter does not have the requested I-PDU stored in the send buffer, it shall:

- send a declined-retransmission (DRTX) PDU, whose N(R) value is equal to the N(R) value of the received SREJ PDU as soon as possible;
- send an AL-DRTX.indication to the AL3 user;
- resume transmission of AL-PDUs not yet transmitted.

7.4.6.4.5.2 Error recovery procedures at the AL3 receiver

When a DRTX message is received, the associated exception condition should be cleared by stopping the timer and incrementing V(R). The AL3 receiver may pass an empty AL-SDU or an invalid received AL-SDU (previously saved), with an appropriate error indication, to the AL3 user via the AL-DATA.indication primitive.

7.4.6.4.6 Unsolicited supervisory PDUs

An unsolicited DRTX PDU received by the AL3 entity should be ignored.

Any supervisory PDUs whose message code is reserved should be ignored.