ITU Telecommunication Standardization Sector
Study Group 15
Experts Group for Video Coding and Systems
in ATM and Other Network Environments

Document AVC-744
31 January 1995

SOURCE: Stuart Dunstan, Siemens Ltd
TITLE: Draft of H.222.1 following the Kamifukuoka meeting
PURPOSE:

CONTENTS

Draft Recommendation H.222.1

MULTIMEDIA MULTIPLEX AND SYNCHRONIZATION FOR AUDIOVISUAL COMMUNICATION IN ATM ENVIRONMENTS

1. Scope

This Recommendation describes the multiplexing and synchronization of multimedia information, for audiovisual communication in ATM environments. This Recommendation specifies the peer-to-peer protocol, and the interactions with the AAL.

2. References

The following ITU-T 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.222.0 - Coding of Moving Pictures and Associated Audio - ISO/IEC 13818-1.

[2] ITU-T Recommendation I.362 - BISDN Adaptation Layer (AAL) Functional Description

[3] ITU-T Recommendation I.363 - BISDN ATM Adaptation Layer (AAL) Specification

[4] CCITT Recommendation X.200 - Reference model of open systems interconnection for CCITT applications.

[5] CCITT Recommendation X.210 - OSI layer service conventions.

3. Terms and definitions

For the purpose of this Recommendation, the following definitions apply:

multiplex identifier field: the packet header field in Recommendation H.222.1 which is the basis upon which multiplexing is performed. In the Program Stream this corresponds to the stream_id field, plus extension fields. In the Transport Stream this corresponds to the PID field.

subchannel: a logical channel in Recommendation H.222.1 formed from packets having a unique multiplex identifier field value. A subchannel carries one H.222.1 elementary stream. A subchannel is unidirectional. There may be many subchannels within one ATM Virtual Channel.

4. Abbreviations

For the purpose of this Recommendation the following abbreviations are used:

ATM Asynchronous Transfer Mode

AAL	ATM Adaptation Layer
PDU	Protocol Data Unit
SCSE	Subchannel Signalling Entity

5. General

This Recommendation deals with the multiplexing and synchronisation of multiple multimedia signals, for use in audiovisual communications in ATM environments. The multimedia signals may be coded audio or video, or other data signals.

This Recommendation is suitable for various applications such as conversational services, distributive services, retrieval services, and messaging services.

This Recommendation is applicable to both unidirectional and bidirectional physical connections. A bidirectional connection may be symmetrical or asymmetrical.

This Recommendation may also be suitable for use in environments other than ATM.

This Recommendation specifies two separate and independent protocols. They are:

- H.222.1 Program Stream
- H.222.1 Transport Stream

These two protocols are based upon the Program Stream and Transport Stream respectively, defined in Recommendation H.222.0 [1]. An overview of H.222.1 and its relation to Recommendation H.222.0, is shown in Figure 1.

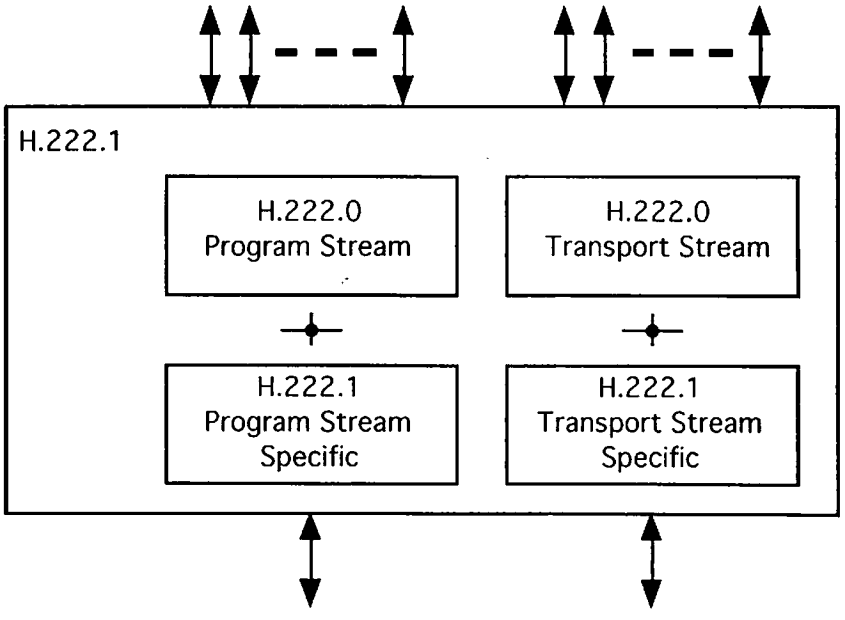


Figure 1/H.222.1.
H.222.1 overview.

This Recommendation uses the services provided by the AAL. The AAL is specified in ITU-T Recommendation I.363 [2]. The use of AAL type 1, type 2, and type 5 is specified here.

An in band subchannel signalling function is provided. Both unacknowledged and acknowledged signalling procedures are supported.

Signals relating to error conditions at the receive side are also specified.

6. Functions provided by H.222.1

H.222.1 provides the following functions.

a) multiplexing

Multiplexing is based on a sequence of PDUs, each of which carry consecutive data from one and only one media source type i.e audio, video, or other data signal. In the H.222.1 Program Stream these PDUs may be of variable length and of relatively large in size. In the H.222.1 Transport Stream these PDUs are of fixed length and of relatively small size. The H.222.1 Transport Stream has a large multiplex capacity.

b) timebase recovery

A program is a collection of associated media, all of which refer to a common time base. The H.222.1 Program Stream supports one and only one program. The H.222.1 Transport Stream supports multiple programs.

The send side and receive side each have their own time bases. Time stamps attached to specific PDUs identify the intended time of arrival of the PDU at the receive side. Synchronisation of the receive side time base with the send side time base is achieved using these time stamps.

c) media presentation synchronization

Additional time stamps identify times at which entities in each media are to be presented to the end user.

d) jitter removal

H.222.1 specifies procedures for removal of timing jitter on received PDUs.

e) buffer management

Rules are specified so as to avoid underflow and overflow of receive side buffers. This is achieved by a hypothetical receive side timing model, which specifies timing relationships between outgoing PDUs at the send side.

f) security and access control

Security and access control functions are provided by media encryption. Support for entitlement control and management messages is also provided.

g) inband signalling

The multiplexing function provides multiple connection end points at the user/H.222.1 service boundary. Protocol is provided that signals to the receive side the association between a PDU and a connection end point. The nature of the information carried by the connection is also described.

h) error reporting

Protocol at the receive side reports error conditions to the H.222.1 user.

i) trick mode

Mechanisms to support video recorder like control functionality e.g. fast forward rewind etc, are included.

j) network maintenance

A network maintenance function, which monitors channel errors, is available.

k) remultiplex support

In the H.222.1 Transport Stream mechanisms to assist in the addition and removal of individual elementary streams are provided. This function only has meaning at a network element. This function is not explicitly supported in the H.222.1 Program Stream.

l) priority

In the H.222.1 Transport Stream, one of two priorities may be indicated for each PDU. This function only has meaning at a network element. This function is not explicitly supported in the H.222.1 Program Stream.

{Ed: It should be determined as to whether a terminal must support all of these functions or a subset of them. It may be that H.24X lists mandatory and optional H.222.1 functions at a coder and a decoder. Optional H.222.1 functions are determined during the H.24X terminal capability exchange}.

7. H.222.1 Program Stream

7.1. H.222.1 Program Stream syntax and semantics

The H.222.0 Program Stream syntax and semantics apply, as defined in clause 2.5.3 of Rec. H.222.0.

The H.222.0 Program Stream Map applies as defined in clause 2.5.4 of Rec. H.222.0. The use of the Program Stream Map in the H.222.1 Program Stream is further defined in the following section on terminal signalling.

7.2. H.222.1 Program Stream timing model

The Program Stream System Target Decoder as defined in clause 2.5.2 of Rec. H.222.0, applies.

7.3. ITU-T Rec. H.222.1 stream types in the H.222.1 Program Stream

ITU-T Rec. H.222.1 elementary stream types are coded using the ITU-T Rec. H.222.1 type xx stream_id. The first byte of the PES packet payload with this stream_id value is defined as a *multiplex_identification* field. ITU-T H.222.1 defined descriptors identify the elementary stream type in the PSM.

The specific ITU-T Rec. H.222.1 type, and the exact coding of the *multiplex_identification* field is for further study.

8. H.2221 Transport Stream protocol

8.1. H.222.1 Transport Stream syntax and semantics

The H.222.0 Transport Stream syntax and semantics apply, as defined in clause 2.4.3 of Rec. H.222.0.

The H.222.0 Program Specific Information applies as defined in clause 2.4.4 of Rec. H.222.0. The use of the PSI in the H.222.1 Transport Stream is further defined in the following section on terminal signalling.

8.2. H.222.1 Transport Stream timing model

The Transport Stream System Target Decoder as defined in clause 2.4.2 of Rec. H.222.0, applies.

8.3. ITU-T Rec. H.222.1 stream types in the H.222.1 Transport Stream

ITU-T Rec. H.222.1 elementary stream types are coded using the ITU-T Rec. H.222.1 type xx stream_id. The PES packet payload is coded as per H.222.0 i.e. there is no *multiplex_identification* field. ITU-T H.222.1 defined descriptors identify the elementary stream type in the PSI tables.

The specific ITU-T Rec. H.222.1 type stream_id is for further study.

9. H.222.1 Program Stream and H.222.1 Transport Stream specific functions

For further study

10. Interaction with the AAL

{Ed: The intent of this section is to specify how the primitives at the AAL-SAP will be used.}

10.1. AAL type 1

expected services from AAL

For further study

10.2. AAL type 2

For further study

10.3. AAL type 5

For further study

{Ed: With respect to AAL type 5, it is expected that there will be co ordination with the ATM Forum proposals for the Video On Demand application. Other applications may use different methods}.

11. Subchannel signalling procedures

Subchannel signalling is the process which establishes and releases a subchannel between peer send and receive H.222.1 entities. Subchannel signalling involves management of a unique multiplex identifier value at the send side, and transmission of this value, and information about the

audio visual data to be carried within the subchannel, to the receiver. Two subchannel signalling procedures are described here, being acknowledged and unacknowledged.

A unidirectional connection may only use unacknowledged procedures. In this case in band signalling information is transmitted periodically.

A bidirectional connection may use unacknowledged or acknowledged procedures. In some applications acknowledged signalling procedures may offer the following advantages,

- error conditions can be defined e.g undefined subchannel multiplexing identifier.
- synchronisation of call phases especially in the case where the remote user is not a human i.e a computer. Transmission cannot begin on a sub channel until it has been correctly established.

11.1. Subchannel signalling unacknowledged procedures

Subchannel signalling using unacknowledged procedures achieve reliable operation through the repeated transmission of signalling information. Subchannel signalling associates and describes all of the elementary streams that compose a program. In the H.222.1 Transport Stream the signalling also identifies one program from many. The frequency of repetition is not specified here.

{Ed: Should H.222.1 say something about the frequency of repetition of PSM/PSI tables in the unacknowledged mode?}.

Signalling in the H.222.1 Program Stream, for unacknowledged operation, is coded using the H.222.0 Program Stream Map, as defined in H.222.0 clause 2.5.4.

Signalling in the H.222.1 Transport Stream, for unacknowledged operation, is coded using the Program Specific Information (PSI), as defined in H.222.0 clause 2.4.4.

11.2. Subchannel signalling acknowledged procedures

Subchannel signalling using acknowledged procedures are specified in Annex A. The procedures are expressed in terms of the Subchannel Signalling Entity (SCSE), which deals with the establishment and release of one subchannel. There is an outgoing SCSE and an incoming SCSE. Peer SCSE pairs deal with the one subchannel, which is by definition uni directional. Two way audiovisual/data communication requires two SCSE pairs, one in the forward direction and one in the reverse direction.

The SCSE is concerned only with H.222.1 peer terminal to terminal signalling and not with the transfer of user data. The SCSE simply states that audio visual and other data may only be transferred on a particular subchannel in the ESTABLISHED state.

12. Hierarchical coded video and multiple ATM Virtual Channels

Some applications may require that the hierarchal layers of the scalability profiles in H.262 be mapped to separate virtual channels. This allows

- the video layer to be routed only where it is required, and
- to select a network QOS appropriate for each layer.

The exact way in which hierarchical video is handled is for the further study.

PCR in all or in some

{Ed: Separate VCs imply multiple instances of H.222.1}.

13. Descriptors

13.1. ITU-T Rec. H.222.0 descriptor priority

This clause defines priority rules for a H.222.1 decoder in the case of semantic conflict of ITU-T Rec. H.222.0 descriptors.

Appendix A recommends H.222.1 encoder rules for H.222.0 descriptors. When applied these rules result in bitstreams that avoid semantic conflict.

H.222.0 program level descriptors and elementary stream level descriptors may sometimes be in conflict. To deal with this conflict four priority rules are defined. These rules are as follows;

A. The descriptor at the elementary stream level overrides a descriptor with similar meaning at the program level. However, the descriptor at the program level applies to all of the elementary streams of the program (descriptors that are not relevant are ignored).

B. The descriptor used at the program level is applied only to that program and has no meaning for an individual elementary stream. Similarly, an elementary stream descriptor has meaning only for that elementary stream, and has no meaning at the program level.

C. The descriptor is ignored when it is present at the program level. It is applied to the associated elementary stream only when it is present at the elementary stream level.

D. The descriptor is ignored when it is present at the program level. It is applied to the associated program only when it is present in ~~program~~ elementary level.

In general programs and elementary streams descriptors are ignored where the meaning of that descriptor conflicts with some other semantic e.g. an audio descriptor when contained in an elementary stream with a video stream_type value.

The priority rules for the H.222.0 defined descriptors are shown in Table 1.

Table 1/H.222.1.

H.222.1 decoder priority rules for H.222.0 defined descriptors.

tag no.	descriptor	priority rule
2	video_stream_descriptor	A
3	audio_stream_descriptor	A
4	hierarchy_descriptor	C
5	registration_descriptor	NS
6	data_stream_alignment_descriptor	A
7	target_background_grid_descriptor	A
8	video_window_descriptor	A
9	CA_descriptor	NS
10	ISO_639_language_descriptor	A
11	system_clock_descriptor	D
12	multiplex_buffer_utilization_descriptor	A
13	copyright_descriptor	B
14	maximum_bitrate_descriptor	~~B~~ NS
15	private_data_indicator_descriptor	NS
16	smoothing_buffer_descriptor	B[1]
17	STD_descriptor	A
18	IBP_descriptor	A

NS - not specified in this recommendation

Note:

1). When the descriptor at the program level conflicts with a descriptor of similar meaning at the elementary stream level, the descriptor at the program level takes precedence.

13.2. Descriptors

Table 2 defines the descriptor_tag values for descriptors defined in this recommendation.

Table 2/H.222.1

H.222.1 descriptor_tag values

descriptor_tag	Identification
0-63	ITU-T Rec. H.222.0 defined
64	Reserved
65	ITU-T_video_stream_descriptor
66	ITU-T_audio_stream_descriptor
67	ITU-T_data_stream_descriptor
68	multiplex_identification_descriptor
69	scse_pdu_type_descriptor
70-255	reserved

{Ed: Descriptor semantics and their usage are to be included}.

13.2.1. ITU-T video stream descriptor

Table 3/H.222.1

Coding of ITU-T video stream descriptor

Syntax	No. of bits	Identifier
ITU-T_video_stream_descriptor() {		
descriptor_tag	**8**	**uimsbf**
descriptor_length	**8**	**uimsbf**
coding_algorithm	**8**	**uimsbf**
if(coding_algorithm == H.261){		
picture_format	**3**	**uimsbf**
picture_rate	**3**	**uimsbf**
reserved	**2**	**uimsbf**
}		
}		

Table 4/H.222.1

coding_algorithm

value	description
0x00	forbidden
0x01	H.261
0x02 - 0xff	reserved

Table 5/H.222.1

picture_format coding

value	description
000	CIF
001	QCIF
010 - 111	reserved

Table 6/H.222.1

picture_rate coding

value	description
000	1/29.97
001	2/29.97
010	3/29.97
011	4/29.97
100 - 111	reserved

13.2.2. ITU-T audio stream descriptor

Table 7/H.222.1

Coding of ITU-T audio stream descriptor

Syntax	**No. of bits**	**Identifier**
ITU-T_audio_stream_descriptor() {		
descriptor_tag	**8**	**uimsbf**
descriptor_length	**8**	**uimsbf**
coding_algorithm	**8**	**uimsbf**
reserved	**8**	**uimsbf**
}		

Table 8/H.222.1

coding_algorithm

value	description
0x00	forbidden
0x01	A-law
0x02	u-law
0x03	G.721
0x04	G.722 (mode 1)
0x05	G.722 (mode 2)
0x06	G.722 (mode 3)
0x07	G.728
0x08 - 0xff	reserved

13.2.3. ITU-T data stream descriptor

Table 9/H.222.1

Coding of ITU-T data stream descriptor

Syntax	No. of bits	Identifier
ITU-T_data_stream_descriptor() {		
descriptor_tag	**8**	**uimsbf**
descriptor_length	**8**	**uimsbf**
protocol	**8**	**uimsbf**
reserved	**8**	**uimsbf**
}		

Table 10/H.222.1

protocol

value	description
0x00	forbidden
0x01	H.24X[1]
0x02	video frame synchronous subchannel
0x03 - 0xff	reserved

Note:

1. "H.24X" implies H.24X supported by an appropriate protocol stack.

13.2.4. Multiplex identification descriptor

The multiplex_identification_descriptor maps the associated ITU-T stream type descriptor to an MID field value. The multiplex_identification_descriptor is only coded in elementary streams with stream_id value equal to ITU-T Rec. H.222.1 type xx. The multiplex_identification_descriptor is only coded at the elementary stream level of the Program Stream PSM table. It is not coded in the Transport Stream. The multiplex_identification_descriptor is shown in Table 11.

Table 11/H.222.1

Coding of the multiplex identification descriptor

Syntax	No. of bits	Identifier
multiplex_identification_descriptor() {		
descriptor_tag	**8**	**uimsbf**
descriptor_length	**8**	**uimsbf**
MID_value	**8**	**uimsbf**
}		

The coding of the MID_value field is for further study.

13.2.5. SCSE PDU type descriptor

The SCSE PDU type descriptor is defined in Annex A of this Recommendation.

14. Video frame synchronous signalling

Synchronisation between a video frame synchronous signal and the video frame is achieved using the PTS field in the PES packet header. The PTS refers to a specific event coded in the PES packet payload. The coded video elementary stream and the video frame synchronous subchannel share the same STC. A video frame synchronous subchannel is identified by the ITU-T_data_stream_descriptor coded in the PSM/PSI tables, with the protocol parameter equal to video frame asynchronous subchannel.

Other details are for further study.

15. Mode changing

Mode changing may be implemented using the multiplexing capabilities of H.222.0. No special support beyond this is provided.

Details are for further study.

- end -

tools are available H.111 H.245

may channels are opened. Some of the may ~~not~~ not be used. according to Hibi's contribution

ANNEX A

(This annex forms an integral part of this Recommendation)

SUBCHANNEL SIGNALLING ACKNOWLEDGED PROCEDURES

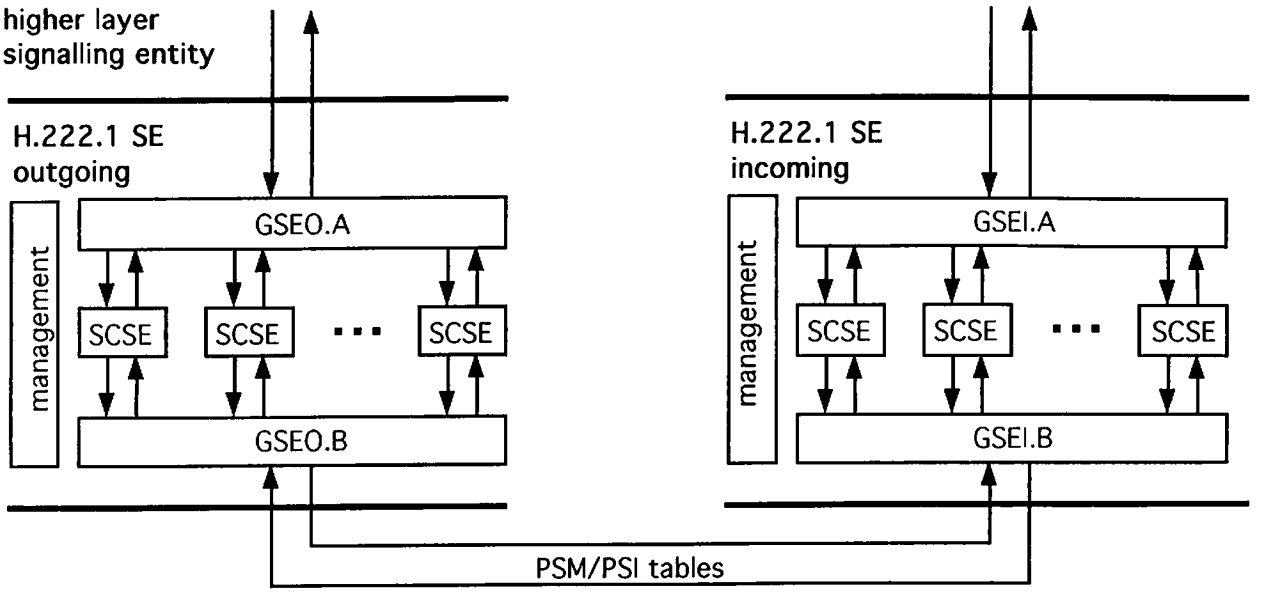
A.1. Introduction

Subchannel signalling using acknowledgment procedures are specified here. The procedures have the following characteristics:

- reliable set up and release of subchannels using acknowledgment procedures.
- the subchannel may only be used for audiovisual and data communication while in the ESTABLISHED state.
- procedures are specified separately for outgoing and incoming subchannel connections. Since a subchannel is defined as being unidirectional, two way audiovisual and data communication requires the procedures to be performed twice; once in the forward calling direction, and once in the backward calling direction.

  There is no connection between the incoming subchannel procedures and the outgoing subchannel procedures at one side, other than via signals to and from the higher layer signalling entity.
- BGN and END PDUs are re transmitted if no acknowledgment is received in a specified time. An error signal is generated if after a specified number of retries no acknowledgment PDU has been received.
- PDUs are coded using PSM and PSI tables in the Program Stream and Transport Stream respectively. A descriptor identifies the meaning of the tables.
- error conditions are reported.

A representation of peer H.222.1 signalling entities is shown in Figure A.1.



SE: Signalling Entity:
GSEO: Global Signalling Entity Outgoing
GSEI: Global Signalling Entity Incoming

SCSE: Subchannel Signalling Entity:
PSM: Program Stream Map (Program Stream)
PSI: Program Specific Information (Transport Stream)

Figure A.1/H.222.1

Overview of H.222.1 Signalling Entities

In Figure A.1 each block has the following functionality:

Global Signalling Entity Outgoing A (GSEO.A)

Upon request from the higher layer signalling entity to set up a subchannel, GSEO.A initiates an SCSE and allocates a multiplex identifier value to it. GSEO.A maps higher layer signalling entity signals for established subchannels to/from the appropriate SCSE. Upon request from the higher layer signalling entity to release a subchannel GSEO.A releases the multiplex identifier value for later reuse.

Subchannel Signalling Entity (SCSE)

The SCSE is responsible for the reliable establishment and release of one subchannel. One SCSE has associated with it a unique multiplex identifier value. Peer SCSEs have the same multiplex identifier value. There is an outgoing SCSE and an incoming SCSE.

Global Signalling Entity Outgoing B (GSEO.B)

GSEO.B maps the output from many SCSEs into PSM/PSI tables. The PSM/PSI tables may be used to commonly establish/acknowledge/release many subchannels at one time.

Management

At the outgoing side some coordination amongst many SCSEs may be required. At the incoming side a management function to deal with the reporting of errors may be required.

Global Signalling Entity Incoming B (GSEI.B)

GSEI.B performs a de multiplexing function, directing incoming PDUs to the appropriate SCSE. It also performs an error reporting function, detecting incoming PDUs with a multiplex field value for which no SCSE exists.

Global Signalling Entity Incoming A (GSEI.A)

GSEI.A maps signals from SCSEs to the higher layer signalling entity at the receiver.

The following specifies the SCSE, for each of an outgoing and an incoming subchannel. In each case procedures are specified in terms of signals and states at the interface between the SCSE and the Global Signalling Entity A. Signalling information is transferred to the peer signalling entity via appropriate PDUs.

A.2. Communication between SCSE and its controlling entity

A.2.1. Signals between SCSE and its controlling entity

Communication between the SCSE, and the higher layer entity which controls it, is performed using the signals shown in Table A.1. These signals are for the purpose of defining sub channel signalling procedures and are not meant to specify or constrain implementation.

Table A.1/H.222.1

Signals and parameters

generic name	type			
	request	indication	response	confirm
ESTABLISH	DESC-PARAM	DESC-PARAM	-	-
RELEASE	-	-	not defined	-
STATUS	DESC-PARAM	DESC-PARAM	not defined	not defined
ERROR	not defined	ERRCODE	not defined	not defined

A.2.2. Signal definition

The definition of these signals is as follows:

a) The ESTABLISH signals are used to establish a subchannel for audiovisual and data communication

b) The RELEASE signals are used to terminate a subchannel.

c) The ERROR signal reports H.222.1 signalling errors and other conditions.

A.2.3. Parameter definition

The definition of the signal parameters should in Table A.1 are as follows:

a) The DESC-PARAM parameter is a descriptor parameter. It conveys information about the type of data carried in the subchannel. DESC-PARAM is mapped to an appropriate descriptor and carried in a BGN PDU. There may be many DESC-PARAM parameters associated with each ESTABLISH signal.

b) The ERRCODE parameter indicates the type of SCSE error.

{Ed: A table of error codes that agrees with the SDL error indications should be included}.

A.2.4. SCSE states

The following states are used to specify the allowed sequence of signals between the SCSE and the SCSE controlling entity, and the exchange of peer-to-peer PDUs. The states are conceptual and are not meant to constrain implementation. The states are specified separately for each of an outgoing SCSE and an incoming SCSE. The states for an outgoing SCSE are:

State 0: RELEASED

The subchannel is released. The subchannel can not be used to send outgoing elementary stream data.

State 1: AWAITING ESTABLISHMENT

The SCSE is waiting to establish a subchannel with a peer incoming SCSE. An ESTABLISH request signal has been received from the SCSE controlling entity. A BGN PDU has been sent and a BGAK PDU is awaited. The subchannel can not be used to send outgoing elementary stream data.

State 2: ESTABLISHED

An SCSE peer-to-peer subchannel connection has been established. A BGAK PDU was received from the peer incoming SCSE and an ESTABLISH confirm signal was sent to the SCSE controlling entity. The subchannel may be used to send outgoing elementary stream data.

State 3: AWAITING RELEASE

The SCSE is waiting to release a subchannel with the peer incoming SCSE. A RELEASE request was received from the SCSE controlling entity and an END PDU was sent to the peer SCSE. The subchannel can not be used to send outgoing elementary stream data.

The states for an incoming SCSE are:

State 0: RELEASED

The subchannel is released. The subchannel can not be used to receive incoming elementary stream data.

State 1: AWAITING ESTABLISHMENT

The SCSE is waiting to establish a subchannel with a peer outgoing SCSE. An BGN PDU was received and an ESTABLISH indication signal has sent to the SCSE controlling entity. The subchannel can not be used to receive incoming elementary stream data.

State 2: ESTABLISHED

An SCSE peer-to-peer subchannel connection has been established. An ESTABLISH response signal was received from the SCSE controlling entity and a BGAK PDU was sent to the peer outgoing SCSE. The subchannel may be used to receive incoming elementary stream data.

A.2.5. State transition diagram

The allowed sequence of signals between the SCSE and its controlling entity is defined here. The allowed sequence of signals relates to states of the SCSE as viewed from the SCSE controlling

entity. The allowed sequences are specified separately for each of an outgoing subchannel connection and an incoming subchannel connection, as shown in Figure A.2 and Figure A.3 respectively.

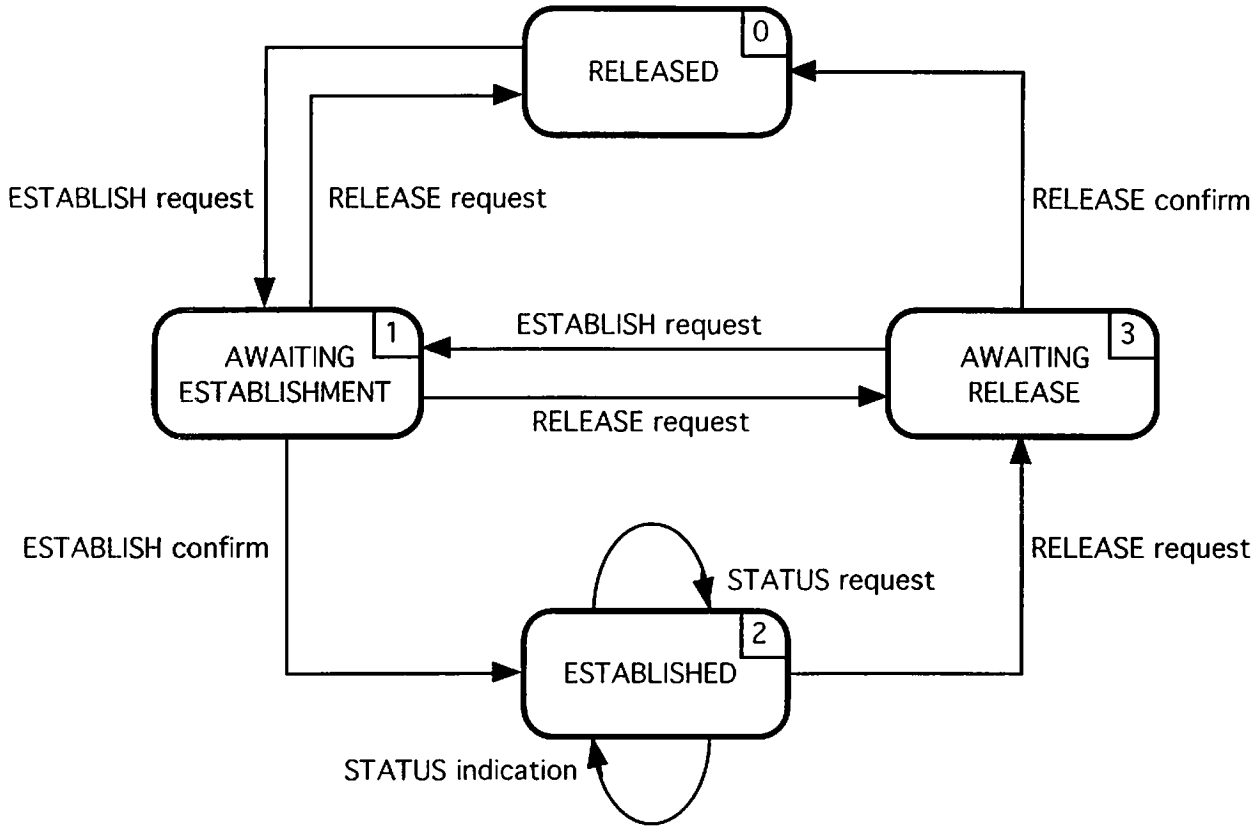


Figure A.2/H.222.1

State transition diagram for sequence of signals at SCSE outgoing

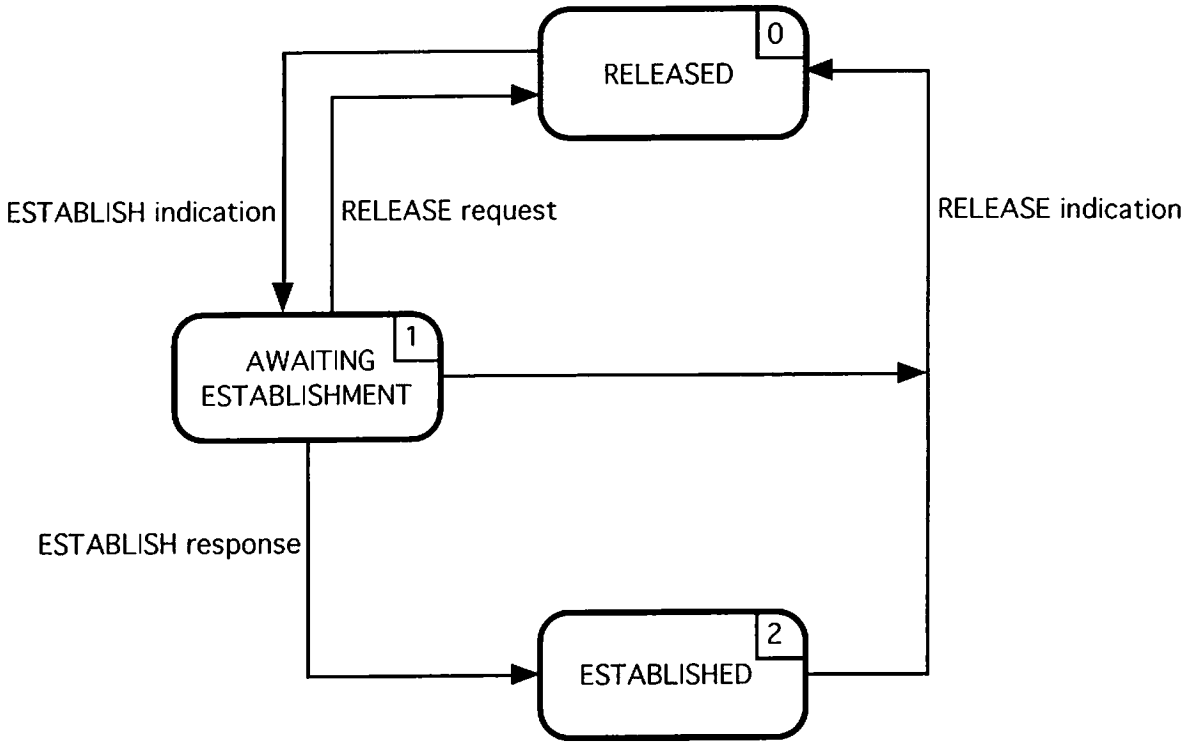


Figure A.3/H.222.1

State transition diagram for sequence of signals at SCSE incoming

A.3. Peer to peer SCSE communication

A.3.1. Protocol Data Units

Table A.2 shows the SCSE PDUs and their function.

Table A.2/H.222.1

SCSE PDU names

function	PDU name	direction	description
establishment	BGN BGAK BGREJ	O -> I O <- I O <- I	connection request connection acknowledgment connection rejection
release	END ENDAK	O -> I O <- I	disconnection request disconnection acknowledgment
status	STATREQ STAT	O -> I O <- I	status request status

Key: O - outgoing, I - incoming.

The SCSE PDUs have the following definition:

a) BGN (begin)

The BGN PDU is used to establish a subchannel connection between an outgoing SCSE and a peer incoming SCSE.

b) BGAK (begin acknowledgment)

The BGAK PDU acknowledges the acceptance of the subchannel connection request from the peer SCSE.

c) BGREJ (begin reject)

The BGREJ PDU is used to reject the subchannel connection request from the peer SCSE.

d) END (end)

The END PDU is used by the outgoing SCSE to release a subchannel connection between two peer SCSEs.

e) ENDAK (end acknowledge)

The ENDAK PDU is used to confirm the release of a subchannel connection.

f) STATREQ (status request)

The STATREQ PDU is used by the outgoing SCSE to request status information from the peer incoming SCSE.

g) STAT (status)

The STAT PDU conveys status information about the incoming SCSE to the peer outgoing SCSE.

A.3.2. Coding of Protocol Data Units

The SCSE PDUs are implemented using H.222.0 syntax elements. In the case of the Program Stream SCSE PDUs are formed using the Program Stream Map. In the case of the Transport Stream, SCSE PDUs are formed using PSI tables. The pdu_type_descriptor in the PSM/PSI tables indicates the type of SCSE PDU. The type of PDU indicates how the PSM/PSI tables are to be interpreted. Absence of this descriptor indicates that unacknowledged signalling procedures are being used, and the meaning of the PSM or PSI tables is as given in H.222.0.

The SCSE relates to the establishment and release of one subchannel. However the coding of SCSE PDUs using H.222.0 syntax elements allows one or more subchannels to be established, or released, at one time. Typically at the start of a call many subchannels are established. In the Program Stream/Transport Stream this corresponds to the PSM/PSI tables respectively describing the complete program to be established. The presence of the pdu_type_descriptor at the program level indicates a BGN PDU. At the end of the call all subchannels may be released using the PSM/PSI tables, with the pdu_type_descriptor, included at the program level, indicating the END PDU. Apart from the pdu_type_descriptor, no program or elementary stream descriptors are coded in the END PDU. Between this time individual subchannels may be established or released using the PSM/PSI tables. In this case the PSM/PSI tables state explicitly to which subchannel the PDU refers to.

Table A.3 indicates the PSM/PSI coding requirements for each SCSE PDU type.

Table A.3/H.222.1

SCSE PDU type and PSM/PSI coding requirements

PDU name	pdu_type descriptor	program descriptors[1]	at least one elementary stream[2]	elementary stream descriptors
BGN	M	O	M	O
BGAK	M	-	M	-
BGREJ	M	-	M	-
END	M	-	O[3]	-
ENDAK	M	-	O[3]	-
STATREQ	M	-	-	-
STAT[4]	M	M	M	M

Key: M - mandatory, O - optional, "-" - not coded.

Notes:

1) Some PDUs include no program descriptors. An exception is the pdu_type_descriptor which may be included at the program level.

2) This column indicates the requirement for explicit reference to at least one elementary stream in the PDU.

3) Reference to one or more elementary streams is optional here. If no elementary stream is specified then the PDU refers to all established subchannels. If one or more elementary streams are listed then the PDU refers only to those listed.

4) The STAT PDU returns a complete listing of all established subchannels i.e. the PSM/PSI tables list information on all elementary streams.

The syntax and semantics of the PSM/PSI tables apply as defined in H.222.0, with the exception of the syntax elements listed in Table A.4.

Table A.4/H.222.1

H.222.0 coding exceptions

H.222.0 table	syntax element	coding
program_stream_map()	current_next_indicator	don't care
	program_stream_map_version	don't care
TS_program_map_section()	version_number	don't care
	current_next_indicator	don't care

A.3.3. PDU type descriptor

The SCSE pdu_type_descriptor indicates the SCSE PDU type for the specified elementary stream with which it is associated. If the scse_pdu_type_descriptor is included at the elementary stream level then it applies just to that elementary stream. If the scse_pdu_type_descriptor is included at the program level then it applies to all the elementary streams included in that program. The coding of the scse_pdu_type_descriptor is shown in Table A.5.

Table A.5/H.222.1

Coding of the SCSE PDU type descriptor

Syntax	No. of bits	Identifier
scse_pdu_type_descriptor() {		
descriptor_tag	**8**	**uimsbf**
descriptor_length	**8**	**uimsbf**
pdu_type	**8**	**uimsbf**
for (i = 0; i < N ; i++){		
parameter	**8**	**bslbf**
}		
}		

The pdu_type field and the parameter field are coded as shown in Table A.6.

Table A.6/H.222.1

Coding of pdu_type and parameter fields

PDU	pdu_type coding	number of parameters	parameter
BGN	0000 0000	1	N(SQ)
BGAK	0000 0001	none	-
BGREJ	0000 0010	1 ???	N(CAUSE) ???
END	0000 0011	none	-
ENDAK	0000 0100	none	-
STATREQ	0000 0101	none	-
STAT	0000 0110	none	-

A.3.4. SCSE state variables

The following state variables are defined at the outgoing SCSE:

VT(RPT)

This state variable counts the number of times a BGN or an END PDU is transmitted in the AWAITING ESTABLISHMENT or AWAITING RELEASE states respectively. The variable is set to zero upon entry to these states. It is incremented by 1 for each BGN or END PDU that is transmitted.

VT(SQ)

This state variable is used to identify retransmitted BGN PDUs. It is set to 0 when the SCSE process begins and is incremented and mapped to the N(SQ) field before initial transmission of a BGN PDU[1].

Note 1: Since the PSM/PSI tables may transmit BGN PDUs from many SCSEs, at the same time, coordination of VT(SQ) state variables among many SCSEs may be required, if the PDU type descriptor is included at the program level in the PSM/PSI tables.

The following state variable is defined at the incoming SCSE:

VR(SQ)

This state variable is used to detect retransmitted BGN PDUs. Upon reception of a BGN PDU it is compared to the value of N(SQ). If the values are different, the PDU is processed and VR(SQ) is set to the value of N(SQ). If they are the same, the PDU is identified as a retransmission.

A.3.5. SCSE timers

The following timer is specified for the outgoing SCSE:

timer_RPT

This timer is used during the AWAITING ESTABLISHMENT and AWAITING RELEASE states. It specifies the time before repeated transmission of a BGN or an END PDU respectively, during which no BGAK or ENDAK PDU has been received.

A.3.6. SCSE parameters

The following parameters are defined for the outgoing SCSE:

max_RPT

The maximum value of VT(RPT), which is the maximum number of times a BGN or END PDU may be transmitted in the AWAITING ESTABLISHMENT and AWAITING RELEASE states.

A.4. SCSE procedures

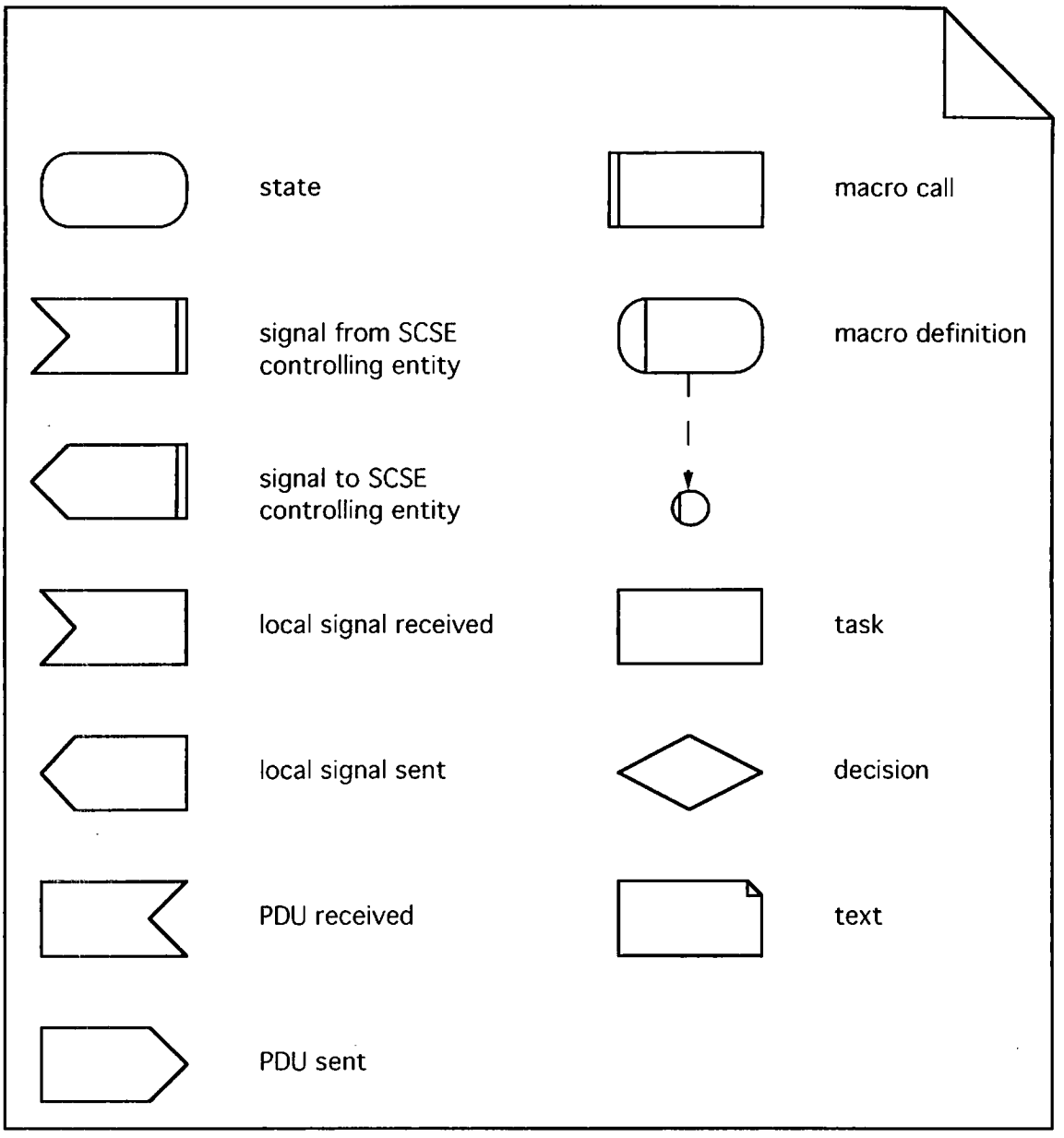


Figure A.4/H.222.1
SDL key

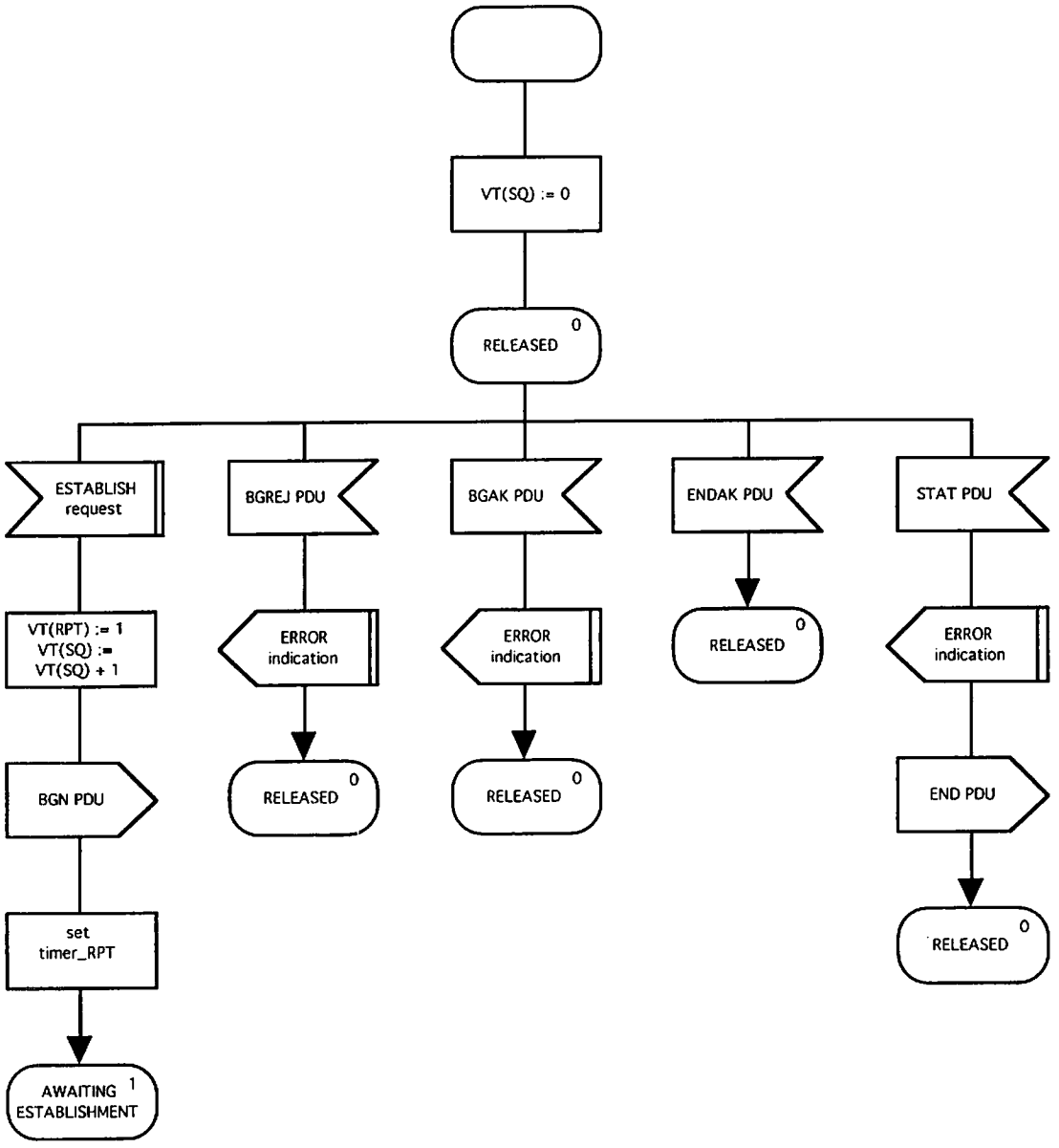


Figure A.5/H.222.1 (sheet 1 of 5)

Outgoing

{Ed: mistake in following: state following ENDAK should be AWAITING ESTABLISHMENT}

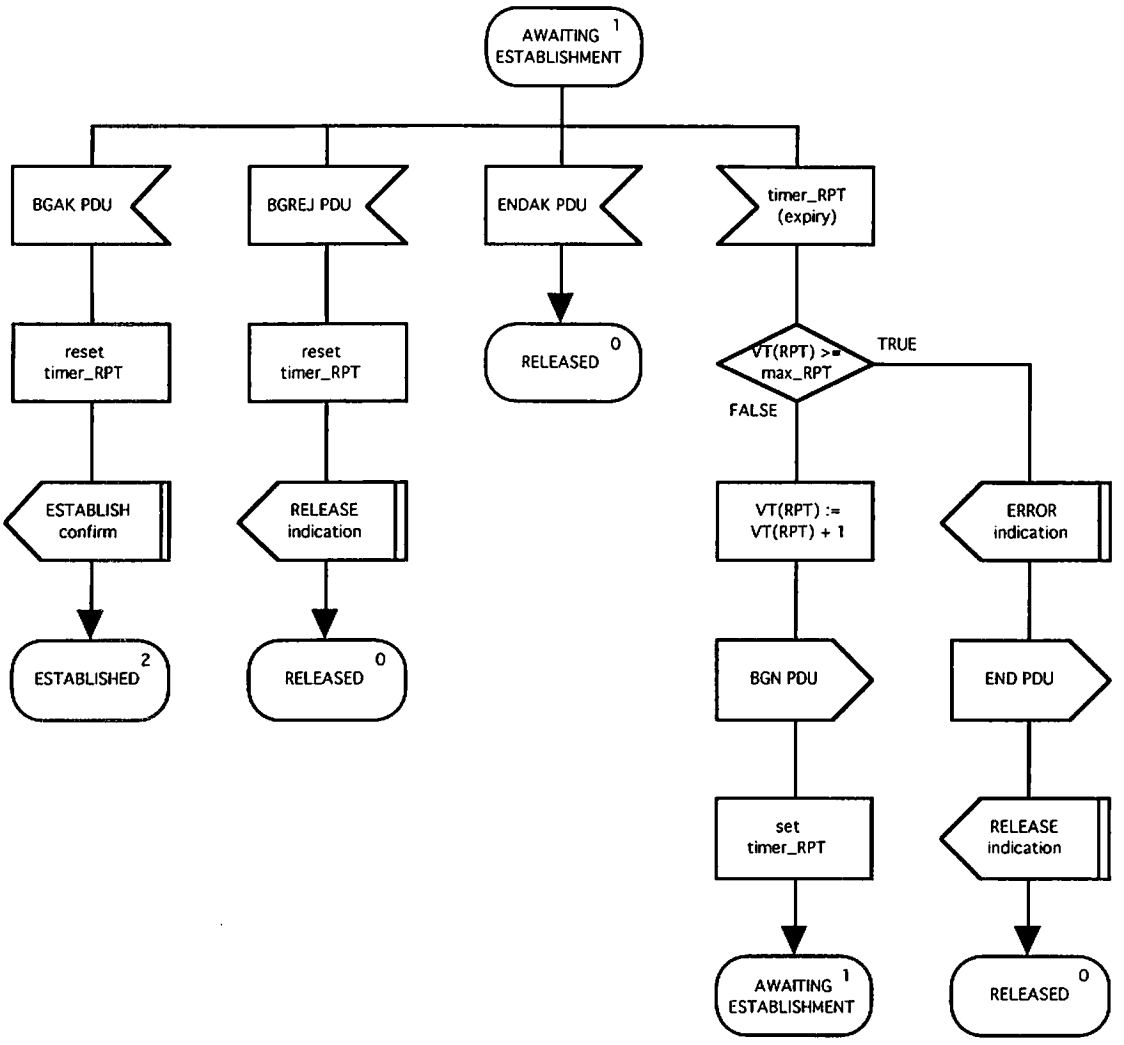


Figure A.5/H.222.1 (sheet 2 of 5)

Outgoing

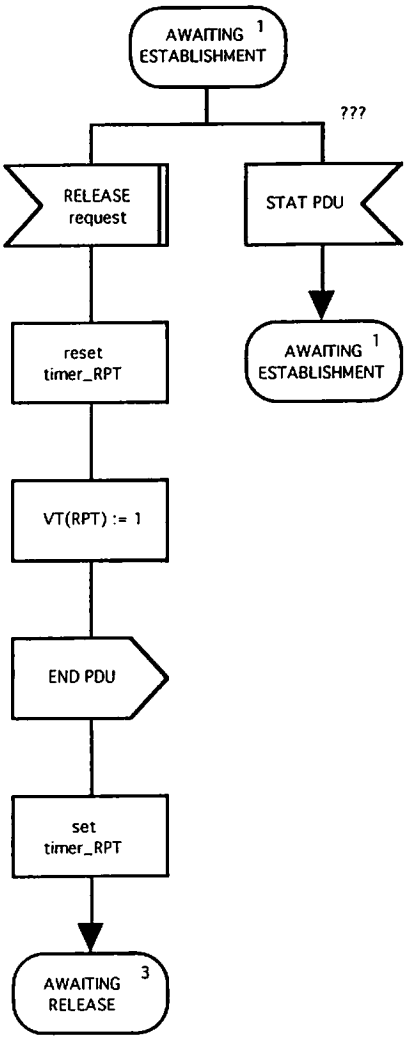


Figure A.5/H.222.1 (sheet 3 of 5)

Outgoing

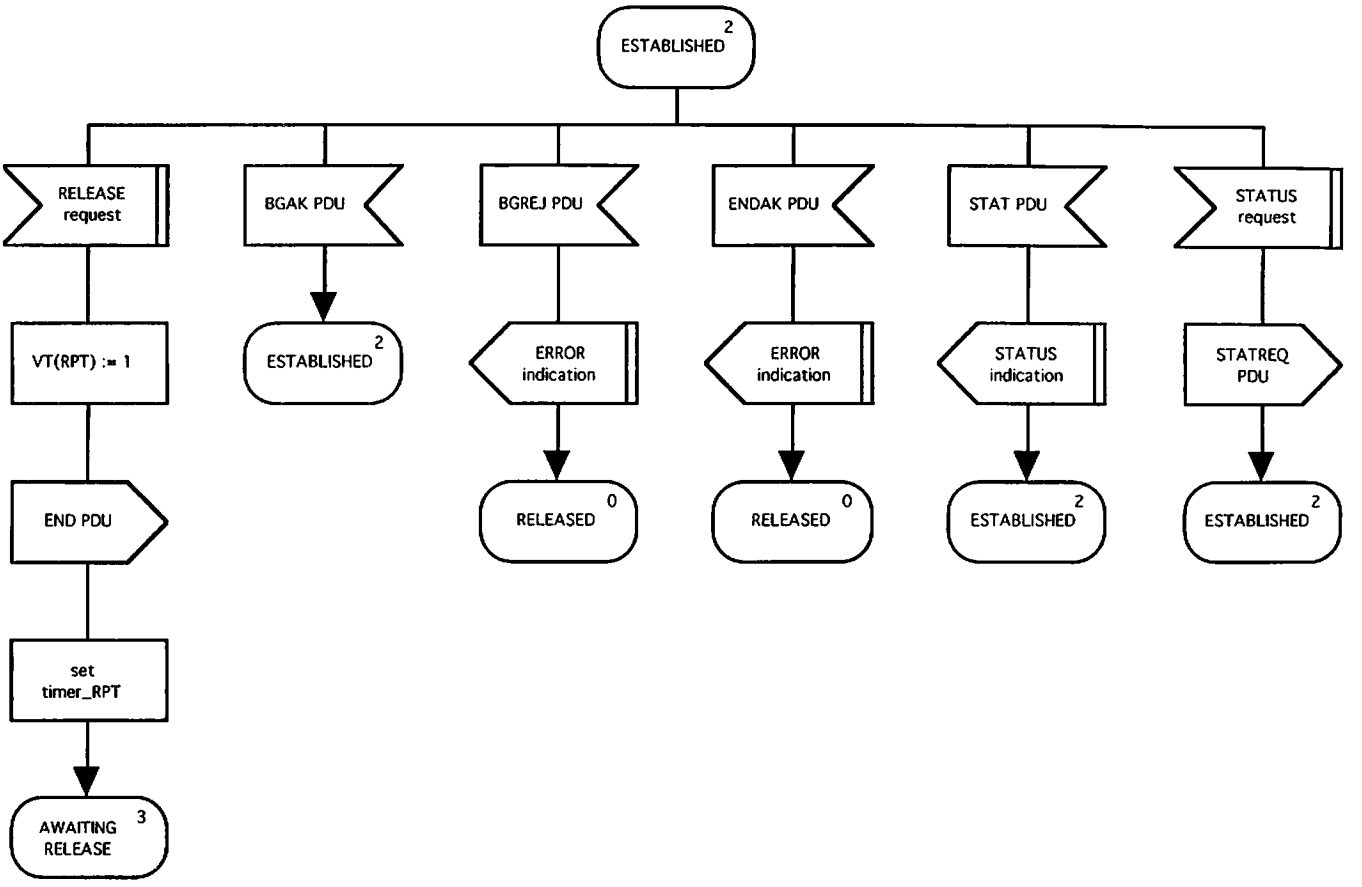


Figure A.5/H.222.1 (sheet 4 of 5)
Outgoing

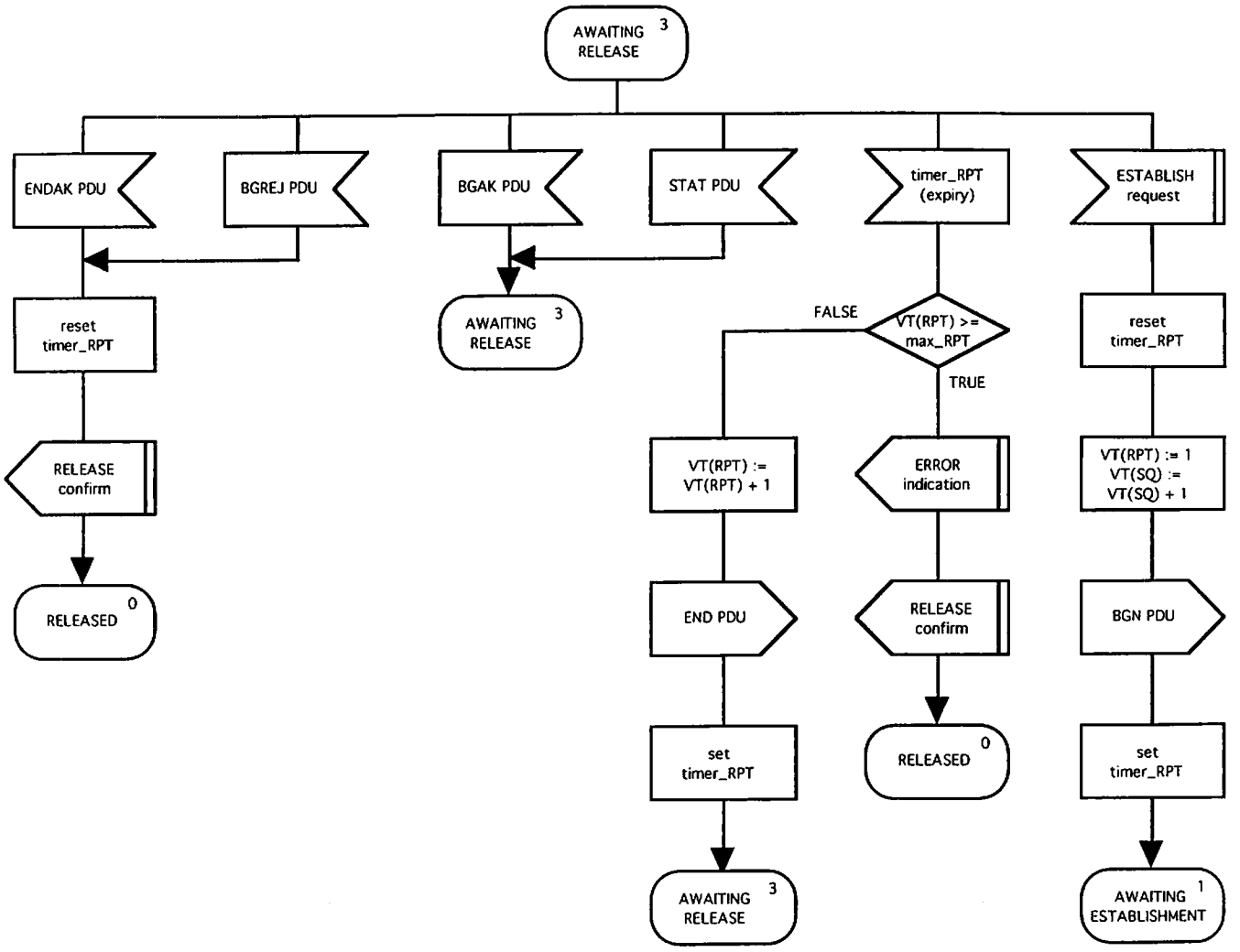


Figure A.5/H.222.1 (sheet 5 of 5)

Outgoing

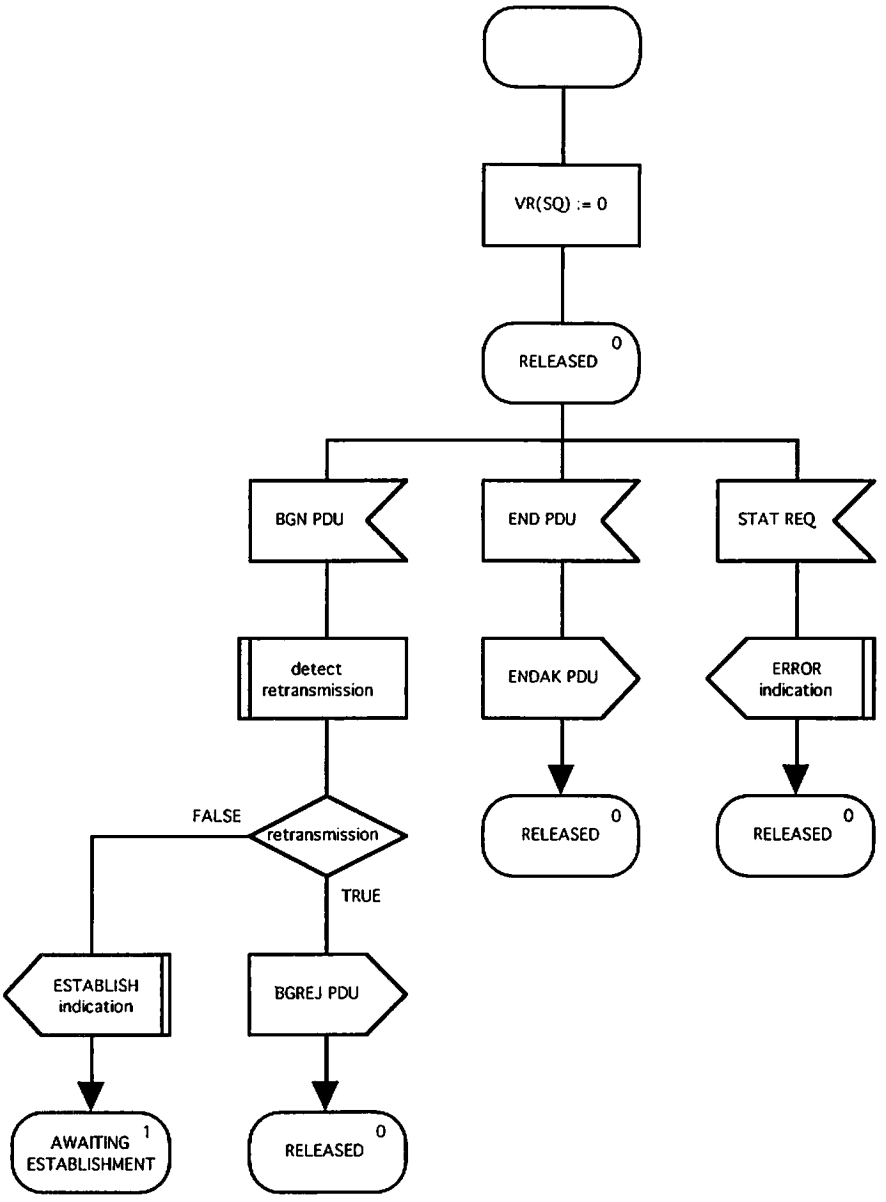


Figure A.6/H.222.1 (sheet 1 of 4)

Incoming

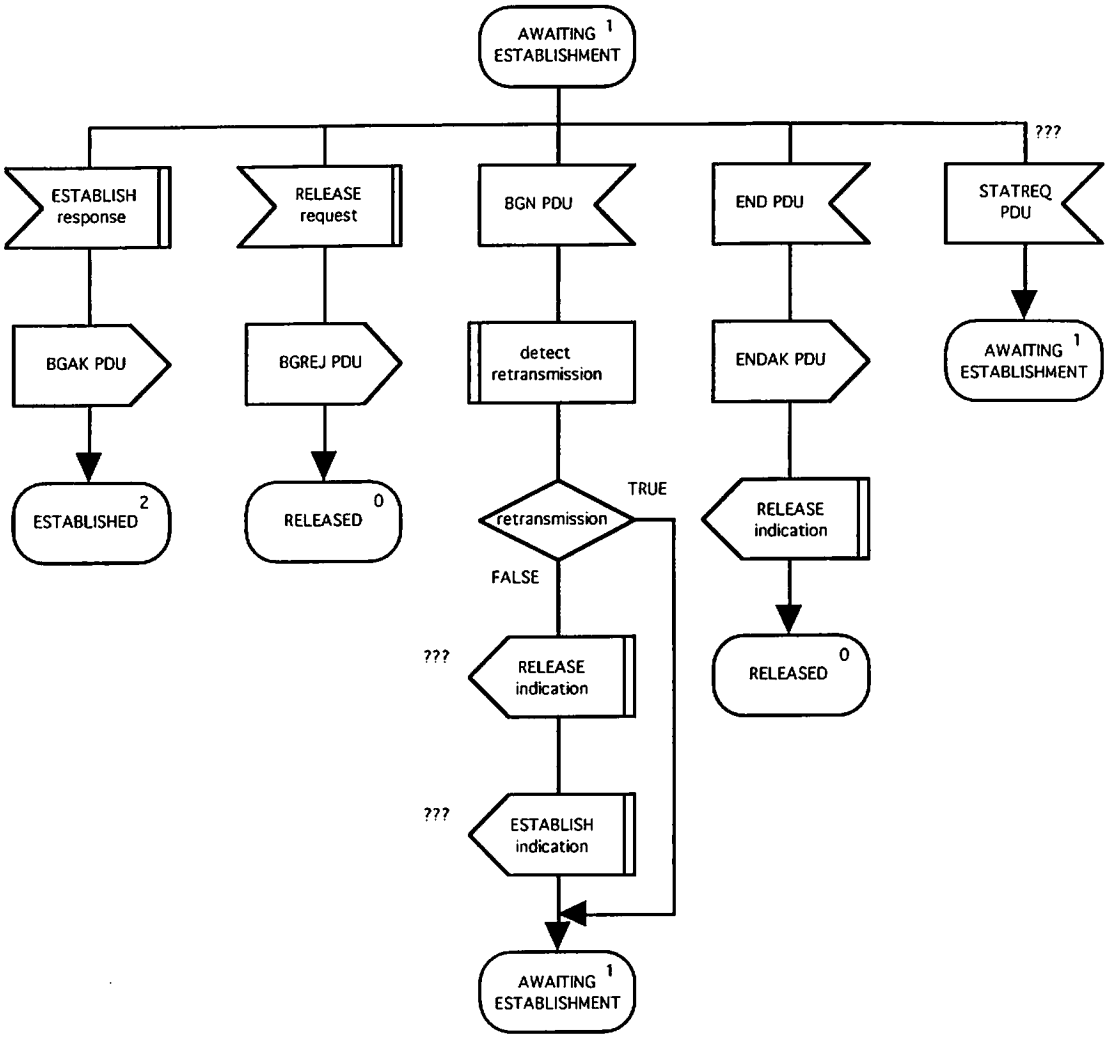


Figure A.6/H.222.1 (sheet 2 of 4)
Incoming

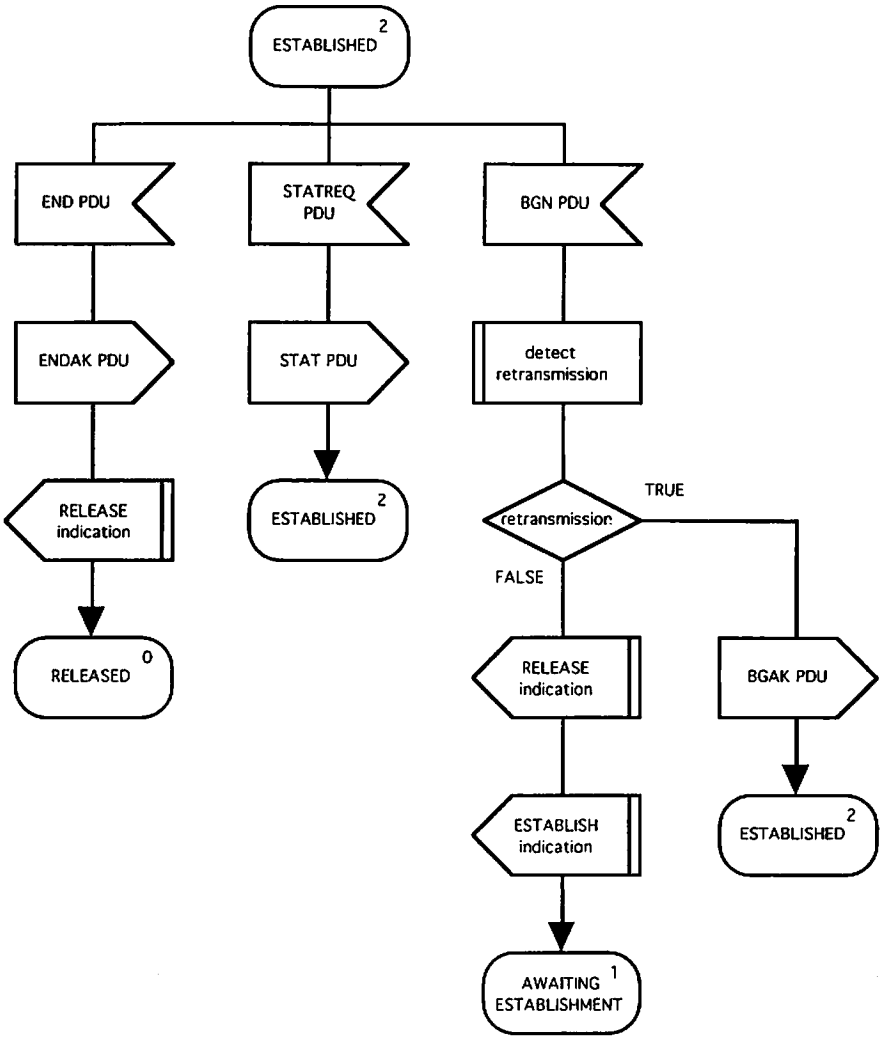


Figure A.6/H.222.1 (sheet 3 of 4)
Incoming

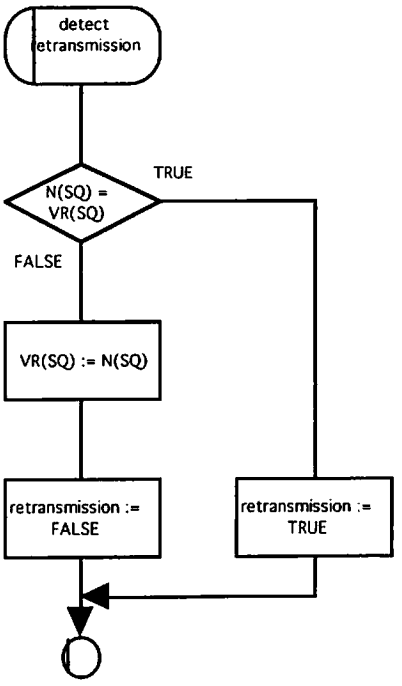


Figure A.6/H.222.1 (sheet 4 of 4)
Incoming

APPENDIX I

RECOMMENDED USAGE OF H.222.0 DESCRIPTORS

(This appendix does not form an integral part of this Recommendation)

Table I.1

Recommended usage of ITU-T defined H.222.0 descriptors

tag value	descriptor	program level	elementary stream level
2	video_stream_descriptor	.	X
3	audio_stream_descriptor	.	X
4	hierarchy_descriptor	.	X
5	registration_descriptor	NS	NS
6	data_stream_alignment_descriptor	.	X
7	target_background_grid_descriptor	.	X
8	video_window_descriptor	.	X
9	CA_descriptor	NS	NS
10	ISO_639_language_descriptor	.	X
11	system_clock_descriptor	X	.
12	multiplex_buffer_utilization_descriptor	.	X
13	copyright_descriptor	X	X
14	maximum_bitrate_descriptor	X NS	X NS
15	private_data_indicator_descriptor	NS	NS
16	smoothing_buffer_descriptor	X	X
17	STD_descriptor	.	X
18	IBP_descriptor	.	X

Notes:

NS - not specified in this Appendix

X - presence of the descriptor is permitted

. - presence of the descriptor is disallowed

- end -