INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Q.2111

(12/1999)

SERIES Q: SWITCHING AND SIGNALLING

Broadband ISDN – Signalling ATM adaptation layer (SAAL)

B-ISDN ATM adaptation layer – Service specific connection oriented protocol in a multilink and connectionless environment (SSCOPMCE)

ITU-T Recommendation Q.2111

(Formerly CCITT Recommendation)

ITU-T Q-SERIES RECOMMENDATIONS

SWITCHING AND SIGNALLING

SIGNALLING IN THE INTERNATIONAL MANUAL SERVICE	Q.1–Q.3
INTERNATIONAL AUTOMATIC AND SEMI-AUTOMATIC WORKING	Q.4–Q.59
FUNCTIONS AND INFORMATION FLOWS FOR SERVICES IN THE ISDN	Q.60–Q.99
CLAUSES APPLICABLE TO ITU-T STANDARD SYSTEMS	Q.100–Q.119
SPECIFICATIONS OF SIGNALLING SYSTEMS No. 4 AND No. 5	Q.120–Q.249
SPECIFICATIONS OF SIGNALLING SYSTEM No. 6	Q.250–Q.309
SPECIFICATIONS OF SIGNALLING SYSTEM R1	Q.310–Q.399
SPECIFICATIONS OF SIGNALLING SYSTEM R2	Q.400–Q.499
DIGITAL EXCHANGES	Q.500–Q.599
INTERWORKING OF SIGNALLING SYSTEMS	Q.600–Q.699
SPECIFICATIONS OF SIGNALLING SYSTEM No. 7	Q.700–Q.849
DIGITAL SUBSCRIBER SIGNALLING SYSTEM No. 1	Q.850–Q.999
PUBLIC LAND MOBILE NETWORK	Q.1000–Q.1099
INTERWORKING WITH SATELLITE MOBILE SYSTEMS	Q.1100–Q.1199
INTELLIGENT NETWORK	Q.1200–Q.1699
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR IMT-2000	Q.1700–Q.1799
BROADBAND ISDN	Q.2000–Q.2999
General aspects	Q.2000–Q.2099
Signalling ATM adaptation layer (SAAL)	**Q.2100–Q.2199**
Signalling network protocols	Q.2200–Q.2299
Common aspects of B-ISDN application protocols for access signalling and network signalling and interworking	Q.2600–Q.2699
B-ISDN application protocols for the network signalling	Q.2700–Q.2899
B-ISDN application protocols for access signalling	Q.2900–Q.2999

For further details, please refer to the list of ITU-T Recommendations.

ITU-T Recommendation Q.2111

B-ISDN ATM adaptation layer – Service specific connection oriented protocol in a multilink and connectionless environment (SSCOPMCE)

Summary

The protocol defined in this Recommendation is called the Service Specific Connection Oriented Protocol in a Multilink and Connectionless Environment (SSCOPMCE) and provides assured data delivery between AAL connection endpoints.

SSCOPMCE is a peer-to-peer protocol which provides the following functions:

- transfer of user data with sequence integrity;
- error correction by selective retransmission;
- flow control;
- connection control;
- error reporting to layer management;
- connection maintenance in the prolonged absence of data transfer;
- local data retrieval by the user;
- error detection of protocol control information;
- status reporting; and
- out-of-sequence delivery.

This Recommendation enhances the specification in ITU-T Q.2110 to allow the SSCOPMCE mechanism to be deployed not only on a single ATM connection but also:

- on multiple ATM connections between the same endpoints; or
- on a connectionless network.

When deployed on a single ATM connection, the protocol defined in this Recommendation is interoperable with the protocol specified in ITU-T Recommendation Q.2110.

This Recommendation describes the necessary elements for layer to layer communication, the elements for peer-to-peer communication, a detailed protocol specification, and examples of assured data operation.

Source

ITU-T Recommendation Q.2111 was prepared by ITU-T Study Group 11 (1997-2000) and approved under the WTSC Resolution 1 procedure on 3 December 1999.

Keywords

ATM Adaptation Layer (AAL), Asynchronous Transfer Mode (ATM), Broadband Integrated Services Digital Network (B-ISDN), Connectionless (CL), Multilink (ML), Network Node Interface (NNI), Signalling AAL (SAAL), Service Specific Coordination Function (SSCF), Service Specific Connection Oriented Protocol (SSCOP), Service Specific Connection Oriented Protocol in a Multilink and Connectionless Environment (SSCOPMCE), Service Specific Convergence Sublayer (SSCS), User-to-Network Interface (UNI).

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. 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 turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSC Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

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

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

CONTENTS

Page

ITU-T Recommendation Q.2111

B-ISDN ATM adaptation layer – Service specific connection oriented protocol in a multilink and connectionless environment (SSCOPMCE)

1 Scope

This ITU-T Recommendation describes the Service Specific Connection Oriented Protocol (SSCOP [6]) enhanced to operate in a multilink or connectionless environment. It specifies the peer-to-peer protocol for the transfer of information and control between any pair of SSCOPMCE entities, the interactions between the SSCOPMCE and its user, the interactions between the SSCOPMCE and the layer below, and the interactions between the SSCOPMCE and layer management.

2 References

2.1 Normative 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 currently valid ITU-T Recommendations is regularly published.

[1] ITU-T Recommendation X.200 (1994) | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*.

[2] ITU-T Recommendation X.210 (1993) | ISO/IEC 10731:1994, *Information technology – Open Systems Interconnection – Basic Reference Model – Conventions for the definition of OSI services*.

[3] ITU-T Recommendation I.150 (1999), *B-ISDN asynchronous transfer mode functional characteristics*.

[4] ITU-T Recommendation I.361 (1999), *B-ISDN ATM layer specification*.

[5] ITU-T Recommendation I.363.5 (1996), *B-ISDN ATM adaptation layer (AAL) specification: Type 5 AAL*.

[6] ITU-T Recommendation Q.2110 (1994), *B-ISDN ATM adaptation layer – Service specific connection oriented protocol (SSCOP)*.

[7] IETF RFC 768, *User Datagram Protocol*.

[8] IETF RFC 791, *Internet Protocol*.

[9] IETF RFC 792, *Internet Control Message Protocol*.

[10] IETF RFC 1122, *Requirements for Internet Hosts – Communication Layers*.

2.2 Additional references (informative)

The references listed in this subclause provide informative background information for the reader.

[11] ITU-T Recommendation Q.2100 (1994), *B-ISDN Signalling ATM Adaptation Layer (SAAL) – Overview description*.

[12] ITU-T Recommendation Q.2130 (1994), *B-ISDN signalling ATM adaptation layer – Service specific coordination function for support of signalling at the user-to-network interface (SSCF at UNI)*.

[13] ITU-T Recommendation Q.2140 (1995), *B-ISDN ATM adaptation layer – Service specific coordination function for signalling at the network node interface (SSCF at NNI)*.

[14] ITU-T Recommendation I.365.2 (1995), *B-ISDN ATM adaptation layer sublayers – Service specific coordination function to provide the connection oriented network service* (SSCF-CONS).

[15] ITU-T Recommendation I.365.3 (1995), *B-ISDN ATM adaptation layer sublayers – Service specific coordination function to provide the connection oriented transport service*.

[16] ITU-T Recommendation I.363.2 (1997), *B-ISDN ATM adaptation layer (AAL) Specification: Type 2 AAL*.

[17] ITU-T Recommendation I.366.1 (1998), *Segmentation and reassembly service specific convergence sublayer for the AAL Type 2*.

[18] ITU-T Recommendation Q.2119 (1996), *B-ISDN ATM adaptation layer − Convergence function for SSCOP above the frame relay core service*.

3 Definitions

The definition of the SSCOPMCE takes into consideration the principles and terminology of ITU-T Recommendations X.200 [1] and X.210 [2] – The reference model and layer service conventions for Open Systems Interconnection (OSI).

NOTE 1 − The physical layer is currently defined in ITU-T Recommendations I.150 [3] and I.361 [4].

NOTE 2 − SSCOP is defined in ITU-T Recommendation Q.2110 [6]; it is deployed on top of an ATM adaptation layer common part on a single ATM connection. The protocol defined in this ITU-T Recommendation is interoperable with the protocol specified in ITU-T Recommendation Q.2110 when deployed in the same environment.

4 Abbreviations

AA	ATM Adaptation
AAL	ATM Adaptation Layer
ATM	Asynchronous Transfer Mode
BGAK	Begin Acknowledge (PDU)
BGN	Begin (PDU)
BGREJ	Begin Reject (PDU)
B-ISDN	Broadband Integrated Services Digital Network
BR	Buffer Release
CP-AAL	Common Part of the AAL
CPCS	Common Part Convergence Sublayer
END	End (PDU)
ENDAK	End Acknowledge (PDU)
ER	Error Recovery (PDU)

ERAK	Error Recovery Acknowledge (PDU)
ID	Interface Data
IP	Internet Protocol
LM	Layer Management
MAA	Management ATM Adaptation
MaxCC	Maximum Connection Control (Count)
MaxPD	Maximum Poll Data (Count)
MaxSTAT	Maximum STAT (Count)
MD	Management Data (PDU)
MSB	Most Significant Bit
MTP	Message Transfer Part
MU	Message Unit
NNI	Network Node Interface
OSI	Open Systems Interconnection
PAD	Padding
PCI	Protocol Control Information
PD	POLL Data
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
PL	Pad Length
POLL	Poll (PDU)
QOS	Quality of Service
R	Reserved (field)
RN	Retrieval Number
RS	Resynchronization (PDU)
RSAK	Resynchronization Acknowledge (PDU)
Rsvd	Reserved (field)
S	Source (field)
SAAL	Signalling ATM Adaptation Layer
SAP	Service Access Point
SAR	Segmentation And Reassembly
SD	Sequenced Data (PDU)
SDL	Specification and Description Language
SDU	Service Data Unit
SN	Sequence Number
SSCF	Service Specific Coordination Function
SSCOP	Service Specific Connection Oriented Protocol

SSCOPMCE	Service Specific Connection Oriented Protocol in a multilink and connectionless environment
SSCS	Service Specific Convergence Sublayer
STAT	Solicited Status (PDU)
UD	Unnumbered Data (PDU)
UDP	User Datagram Protocol
UNI	User Network Interface
USTAT	Unsolicited Status (PDU)
UU	User-to-User
VR	Receiver state Variable
VT	Transmitter state Variable

5 General

5.1 History

The Service Specific Connection Oriented Protocol in a Multilink and Connectionless Environment (SSCOPMCE) is an extension of the Service Specific Connection Oriented Protocol (SSCOP), Q.2110 [6], which was defined for use in the Service Specific Convergence Sublayer (SSCS) of the ATM Adaptation Layer (AAL). The extensions provided by this ITU-T Recommendation enables SSCOPMCE to operate in other environments than an AAL as well. SSCOPMCE is used to transfer variable length Service Data Units (SDUs) between users of SSCOPMCE.

In general, SSCOPMCE is defined as a generic protocol engine applicable to various environments. Nevertheless, when implemented in conjunction with a specific Service Specific Convergence Sublayer or SSCOPMCE user, a restricted implementation may be chosen.

NOTE – Although permitted, restricted implementations of SSCOPMCE are discouraged.

5.2 Services provided

SSCOPMCE provides the following services:

a) GA1: Assured data transfer between two SSCOPMCE users:

The SSCOPMCE provides the capabilities to transfer SSCOPMCE-SDUs from one SSCOPMCE user to one other SSCOPMCE user through the Common Part Sublayer of the AAL (CP-AAL). The service offers peer-to-peer operation:

- Data transfer of SSCOPMCE-SDUs of up to 65 528 octets.
- SSCOPMCE-SDU contents and sequence integrity is guaranteed by SSCOPMCE.

b) Unassured data transfer between two SSCOPMCE users:

The SSCOPMCE provides the capabilities to transfer SSCOPMCE-SDUs from one SSCOPMCE user to one other SSCOPMCE user through the Common Part Sublayer of the AAL (CP-AAL). The service offers unidirectional operation:

- Data transfer of SSCOPMCE-SDUs of up to 65 528 octets.
- Neither SSCOPMCE-SDU contents nor sequence integrity is guaranteed by SSCOPMCE.

c) Unassured data transfer between two SSCOPMCE layer management entities:

The SSCOPMCE provides the capabilities to transfer layer management data from one SSCOPMCE layer management entity to one other SSCOPMCE layer management entity through the Common Part Sublayer of the AAL (CP-AAL). The service offers unidirectional operation:

- Data transfer of layer management data of up to 65 528 octets.
- Neither layer management data contents nor sequence integrity is guaranteed by SSCOPMCE.

d) Connection establishment, release, and resynchronization:

In order to provide the assured data transfer service, SSCOPMCE makes use of SSCOPMCE connections. The service offers peer-to-peer operation:

- Data transfer of user-to-user data of up to 65 524 octets.
- Only contents integrity is assured.

e) Out of sequence delivery:

The sender may request that the sequence integrity shall not be maintained on an SSCOPMCE-SDU by SDU basis. In this case, SSCOPMCE provides only for contents integrity.

f) Local data retrieval by the user:

The local SSCOPMCE user may retrieve in-sequence SDUs which have not yet been released by the SSCOPMCE entity.

g) Error reporting to layer management:

SSCOPMCE reports operational errors, e.g. number of retransmissions, etc. to layer management.

5.3 Modes of operation

This ITU-T Recommendation defines three modes of operation:

"A" Multilink environment:

In this mode, SSCOPMCE performs its functions by utilizing one or more CPCS connections, i.e. ATM connections with the AAL type 5 Common Part; these CPCS connections are called "links". Links may be added or removed during the operation of the SSCOPMCE protocol entity. When in the data transfer phase, SSCOPMCE detects links with insufficient Quality of Service and removes such links from the procedures.

"B" Connectionless environment:

In this mode, SSCOPMCE performs its functions by utilizing a connectionless environment. One or more links to the connectionless environment may be deployed. Links may be added or removed during the operation of the SSCOPMCE protocol entity.

NOTE – The convergence function between the functionality of the AAL type 5 Common Part and the connectionless environment is not specified in this ITU-T Recommendation.

"C" Compatibility:

In this mode, interoperability with SSCOP defined in ITU-T Recommendation Q.2110 [6] is guaranteed. SSCOPMCE performs its functions by utilizing one CPCS connection, i.e. one ATM connection with the AAL type 5 Common Part.

At least one of these modes must be implemented. During the lifetime of an SSCOPMCE protocol entity, the mode of operation cannot be changed.

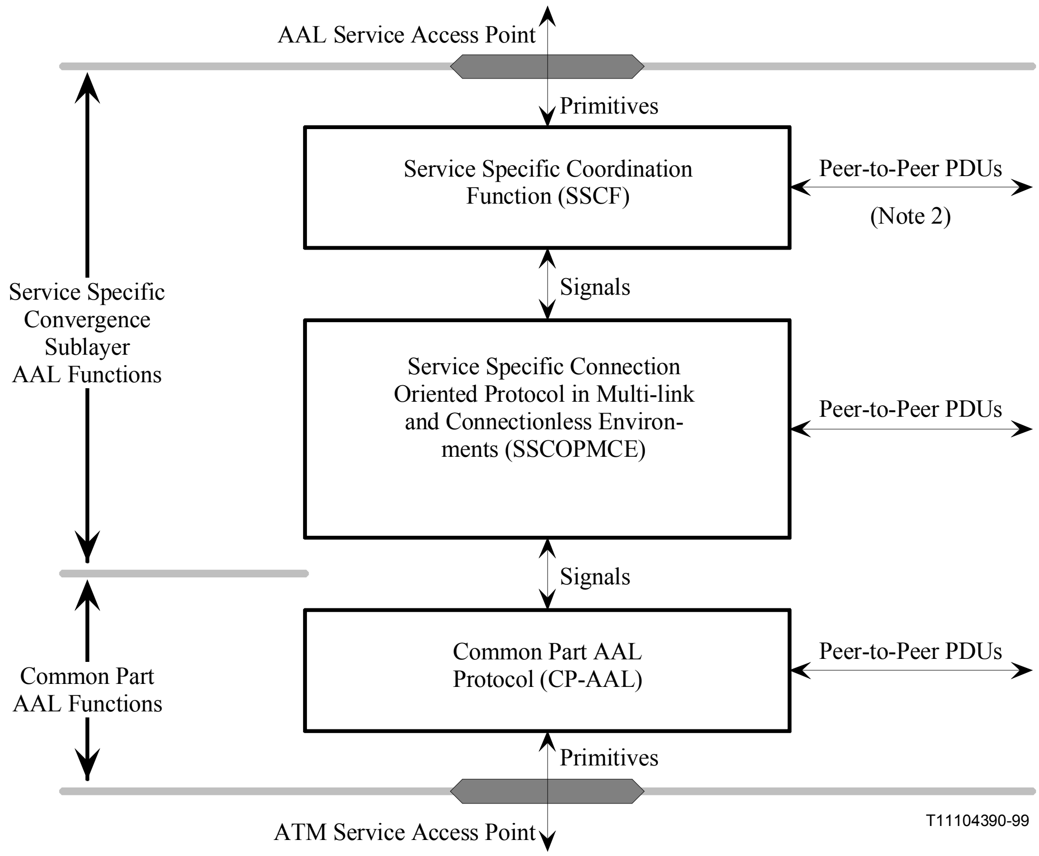
In "Compatibility Mode", SSCOPMCE provides its service to a Service Specific Coordination Function (SSCF), as shown in Figure 1. The SSCF maps the service of SSCOPMCE to the needs of the AAL user. The SSCFs are specified in other Recommendations.

Example Service Specific Coordination Functions (SSCF) are defined in ITU-T Q.2130 [12], Q.2140 [13], I.365.2 [14], and I.365.3 [15].

SSCOPMCE, in the Compatibility Mode, utilizes the service of CP-AAL (Common Part AAL) protocols that provide unassured information transfer and a mechanism for detecting corruption of SSCOPMCE Protocol Data Units (PDUs).

The CP-AAL protocols are specified in ITU-T I.363.5 [5]. The CP-AAL service is also provided by ITU-T I.366.1 [17] (with transmission error detection enabled) deployed on an AAL type 2 service [16] or by ITU-T Q.2119 [18] deployed on a Frame Relay service.

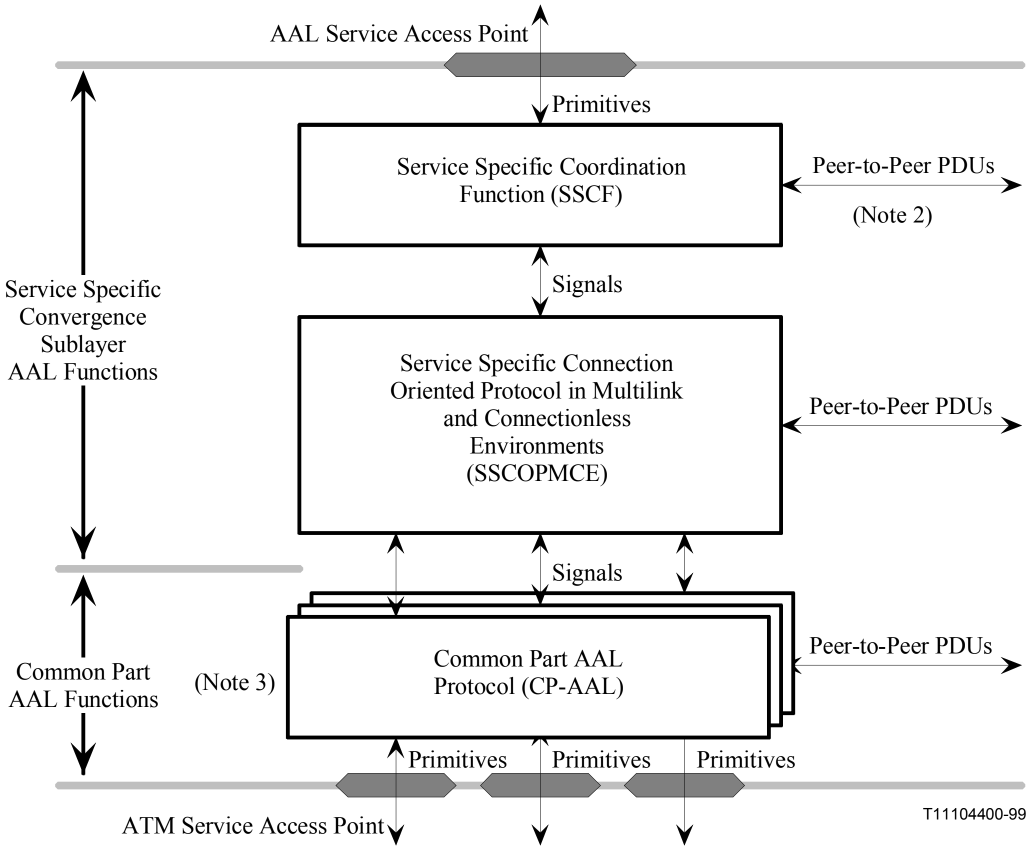
As shown in Figure 1, the AAL is functionally divided into the common part and the Service Specific Convergence Sublayer (SSCS). The SSCS is specific to the needs of the particular service application and in some cases may be functionally null. The SSCOPMCE can operate over different AAL Common Part Protocols and may be used by different SSCFs; the SSCF is specific to the needs of the service application. The Common Part protocols are specified in ITU-T I.363.5 [5].



NOTE 1 – The figure represents the allocation of functions and is not extended to illustrate sublayers as definided by OSI modelling principles.
NOTE 2 – A particular SSCF may include a protocol for the exchange of PDUs.

Figure 1/Q.2111 – SSCOPMCE within the AAL structure

A representation of SSCOPMCE operating in a multilink environment is shown in Figure 2. This figure reflects that the SSCOPMCE entity can still operate within an AAL in this mode. The only significant architectural difference from the compatibility mode is the use of multiple ATM connections. Figure 2 shows the use of multiple instances of AAL common parts to connect the SSCOPMCE to the ATM connections.



NOTE 1 – The figure represents the allocation of functions and is not extended to illustrate sublayers as defined by OSI modelling principles.
NOTE 2 – A particular SSCF may include a protocol for the exchange of PDUs.
NOTE 3 – A single SSCOPMCE entity may interact with CP-AAL entities of different types.

Figure 2/Q.2111 – SSCOPMCE in multilink environment

A representation of SSCOPMCE operating in a connectionless environment is shown in Figure 3. In this mode the SSCOPMCE entity operates more like a transport layer protocol and is clearly outside the scope of an AAL. Through the lower Service Access Point (SAP), a connectionless service, such as IP (IETF RFC 791 [8]) or UDP (IETF RFC 768 [7]), delivers SSCOPMCE PDU payload fragments for delivery to the user of the SSCOPMCE entity and accepts SSCOPMCE PDUs for transfer to the peer user. Annex C describes the mapping of primitives across this SAP with IP and UDP.

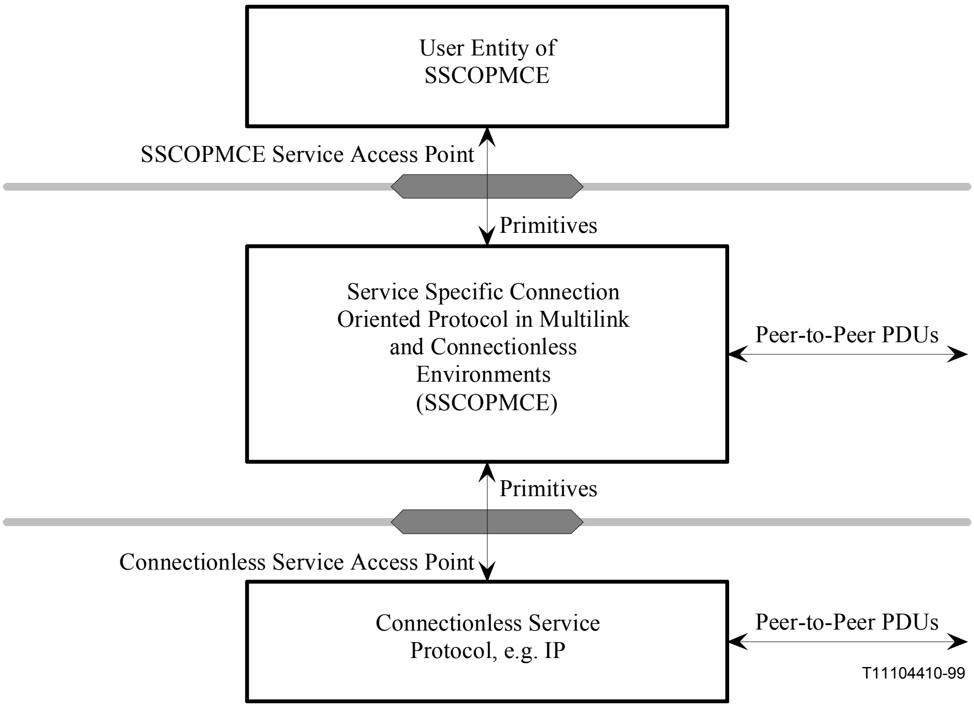


Figure 3/Q.2111 – SSCOPMCE in a connectionless environment

5.4 Addition and removal of "links"

SSCOPMCE PDUs are transmitted over links. Links may be added and/or removed during the lifetime of an SSCOPMCE entity. Depending on the mode of operation, the following applies:

a) In the multilink mode (Mode "A"), a link is a CPCS connection provided by the service of the CP-AAL.

b) In the connectionless mode (Mode "B"), a link may be expressed by different pairs of endpoint addresses; it is also possible that different links use different media (e.g. terrestrial and satellite, etc.).

c) In the compatibility mode (Mode "C"), a link is a CPCS connection provided by the service of the CP-AAL; exactly one link is used.

NOTE – The definition in this ITU-T Recommendation assumes that at start-up no link exists.

6 Functions of the SSCOPMCE

SSCOPMCE performs the following functions:

a) *Sequence Integrity*

This function preserves the order of SSCOPMCE SDUs that were submitted for transfer by SSCOPMCE.

b) *Error Correction by Selective Retransmission*

Through a sequencing mechanism, the receiving SSCOPMCE entity can detect missing SSCOPMCE SDUs. This function corrects sequence errors through retransmission.

c) *Flow Control*

This function allows an SSCOPMCE receiver to control the rate at which the peer SSCOPMCE transmitter entity may send information.

d) *Error Reporting to Layer Management*

This function indicates to layer management errors which have occurred.

e) *Keep Alive*

This function verifies that the two peer SSCOPMCE entities participating in a connection are remaining in a link connection established state even in the case of a prolonged absence of data transfer.

f) *Local Data Retrieval*

This function allows the local SSCOPMCE user to retrieve in-sequence SDUs which have not yet been released by the SSCOPMCE entity.

g) *Connection Control*

This function performs the establishment, release, and resynchronization of an SSCOPMCE connection. It also allows the transmission of variable length user-to-user information without a guarantee of delivery.

h) *Transfer of User-Data*

This function is used for the conveyance of user data between users of the SSCOPMCE. SSCOPMCE supports both assured and unassured data transfer.

i) *Protocol Error Detection and Recovery*

This function detects and recovers from errors in the operation of the protocol.

j) *Status Reporting*

This function allows the transmitter and receiver peer entities to exchange status information.

k) *Mode of Operation*

This function allows two peer SSCOPMCE protocol entities to operate in a multilink environment, in a connectionless environment, or in an environment where the peer SSCOP entity operates according to the definitions in ITU-T Recommendation Q.2110 [6].

l) *Out-of-sequence delivery*

This function for the receiving SSCOPMCE entity allows the delivery of SDUs to the receiving SSCOPMCE user without regard to the order in which the SDUs were submitted for transfer by the transmitting SSCOPMCE user; this function is selectable on an SD PDU by SD PDU basis. The data is transmitted reliably but need not be delivered in-sequence.

NOTE – A possible application of this function is the provision of a multi-stream environment on top of SSCOPMCE with SSCOPMCE providing reliable transmission and the multi-streaming SSCOPMCE user providing the resequencing for individual streams. Head-of-line blocking of unrelated streams would be avoided.

m) *Transfer of Management-Data*

This function is used for the conveyance of layer management data between management entities of the SSCOPMCE. SSCOPMCE supports unassured data transfer.

7 Elements for layer to layer communication

This clause defines the signals and state transition diagram for sequences of signals between SSCOPMCE and its user. The term "signal" is used instead of "primitive" in order to reflect the fact that between SSCOPMCE and its user there is no service access point defined.

7.1 Signals between SSCOPMCE and its user

The following repertoire of AA-signals between SSCOPMCE and its user is defined (see Table 1):

Table 1/Q.2111 – SSCOPMCE signals and parameters

Primitive Generic Name	Type			
	Request	**Indication**	**Response**	**Confirm**
AA-ESTABLISH	SSCOP-UU, BR	SSCOP-UU	SSCOP-UU, BR	SSCOP-UU
AA-RELEASE	SSCOP-UU	SSCOP-UU, Source	–	(Note)
AA-DATA	MU OOS	MU, OOS, SN	–	–
AA-RESYNC	SSCOP-UU	SSCOP-UU	(Note)	(Note)
AA-RECOVER	–	(Note)	(Note)	–
AA-UNITDATA	MU	MU	–	–
AA-RETRIEVE	RN	MU	–	–
AA-RETRIEVE COMPLETE	–	(Note)	–	–
– This primitive is not defined. NOTE – This primitive has no parameters.				

7.1.1 Signal definition

The definition of these signals is as follows:

a) The **AA-ESTABLISH signals** are used to establish a point-to-point connection for assured information transfer between peer user entities.

b) The **AA-RELEASE signals** are used to terminate a point-to-point connection for assured information transfer between peer user entities.

c) The **AA-DATA signals** are used for the assured point-to-point transfer of SDUs between peer user entities.

d) The **AA-RESYNC signals** are used to resynchronize the SSCOPMCE connection.

e) The **AA-RECOVER signals** are used during recovery from protocol errors.

f) The **AA-UNITDATA signals** are used for the non-assured, broadcast and point-to-point, transfer of SDUs between peer user entities.

g) **AA-RETRIEVE signals** are used to retrieve SDUs submitted by the user for transmission but not yet released by the transmitter.

h) **AA-RETRIEVE COMPLETE signal** is used to indicate that there are no additional SDUs to be returned to the SSCOPMCE user.

7.1.2 Parameter definition

Table 1 lists the parameters associated with each SSCOPMCE signal. The definition of the parameters is as follows:

a) The **Message Unit (MU)** parameter is used during information transfer to convey a variable-length message. In AA-DATA.request and AA-UNITDATA.request signals this parameter is mapped transparently into the Information field of an SSCOPMCE PDU. For AA-DATA.indication and AA-UNITDATA.indication signals, this parameter contains the contents of the information field of the received SSCOPMCE PDU. In AA-RETRIEVE.indication signals, this parameter contains a message unit returned to the SSCOPMCE user from either the transmitter queue (data not yet sent) or the transmitter buffer (data sent but not yet acknowledged). The MU is an integral multiple of one octet.

b) The **SSCOP User-to-User information (SSCOP-UU) parameter** is used during connection control to convey a variable-length user-to-user message. The transfer of SSCOP-UU in BGN, BGAK, BGREJ, RS, and END PDUs cannot be guaranteed. In request and response signals, this parameter is mapped transparently into the SSCOP-UU (SSCOP User-to-User) field of an SSCOP PDU. For indication and confirm signals, this parameter contains the contents of the SSCOP-UU field of the received SSCOPMCE PDU. The SSCOP-UU is an integral multiple of one octet, if it is present. The SSCOP-UU may be null (no data present).

c) The **Sequence Number (SN) parameter** indicates the value of N(S) in the received SD PDU, and may be used to support the data retrieval operation.

d) The **Retrieval Number (RN) parameter** is used to support data retrieval. The value RN + 1 indicates the value of the N(S) for the first SD PDU to be retrieved. A value of "Unknown" indicates that only the not yet transmitted SD PDUs are to be retrieved. A value of "Total" indicates that all the SD PDUs in both the transmission buffer and transmission queue are to be retrieved.

e) The **Buffer Release (BR) parameter** indicates whether the transmitter may release its buffers upon subsequent release of the connection. This parameter also allows for the release of selectively acknowledged messages from the transmitter buffer. A value of "TRUE" indicates that the transmission buffer and transmission queue may be released, and a value of "FALSE" indicates that the transmission buffer and transmission queue may not be released.

f) The **source parameter** indicates to the SSCOP user whether the SSCOP layer or the peer SSCOP user originated the connection release. This parameter assumes one of two values: "SSCOP" or "User". If "SSCOP" is indicated, the user should disregard the SSCOP-UU parameter, if present.

g) The **Out-Of-Sequence delivery (OOS) parameter** is used in the AAL-Data primitives to request/indicate whether or not out of service delivery of the MU is/has been allowed. At the transmitter, if the OOS parameter is zero (or if it is absent) in-sequence delivery of the MU at the receiver is required. If the OOS parameter is one, out-of-sequence delivery of the MU is allowed. The value of the OOS parameter in the AAL-Data.request primitive is mapped into the delivery (D) bit of the resulting SD PDU in order to indicate the chosen delivery option to the receiver.

7.1.3 State transition diagram for sequences of signals

This subclause defines the constraints on the sequences in which the signals may occur. The sequences are related to the states at one point-to-point SSCOPMCE endpoint, between SSCOPMCE and its user.

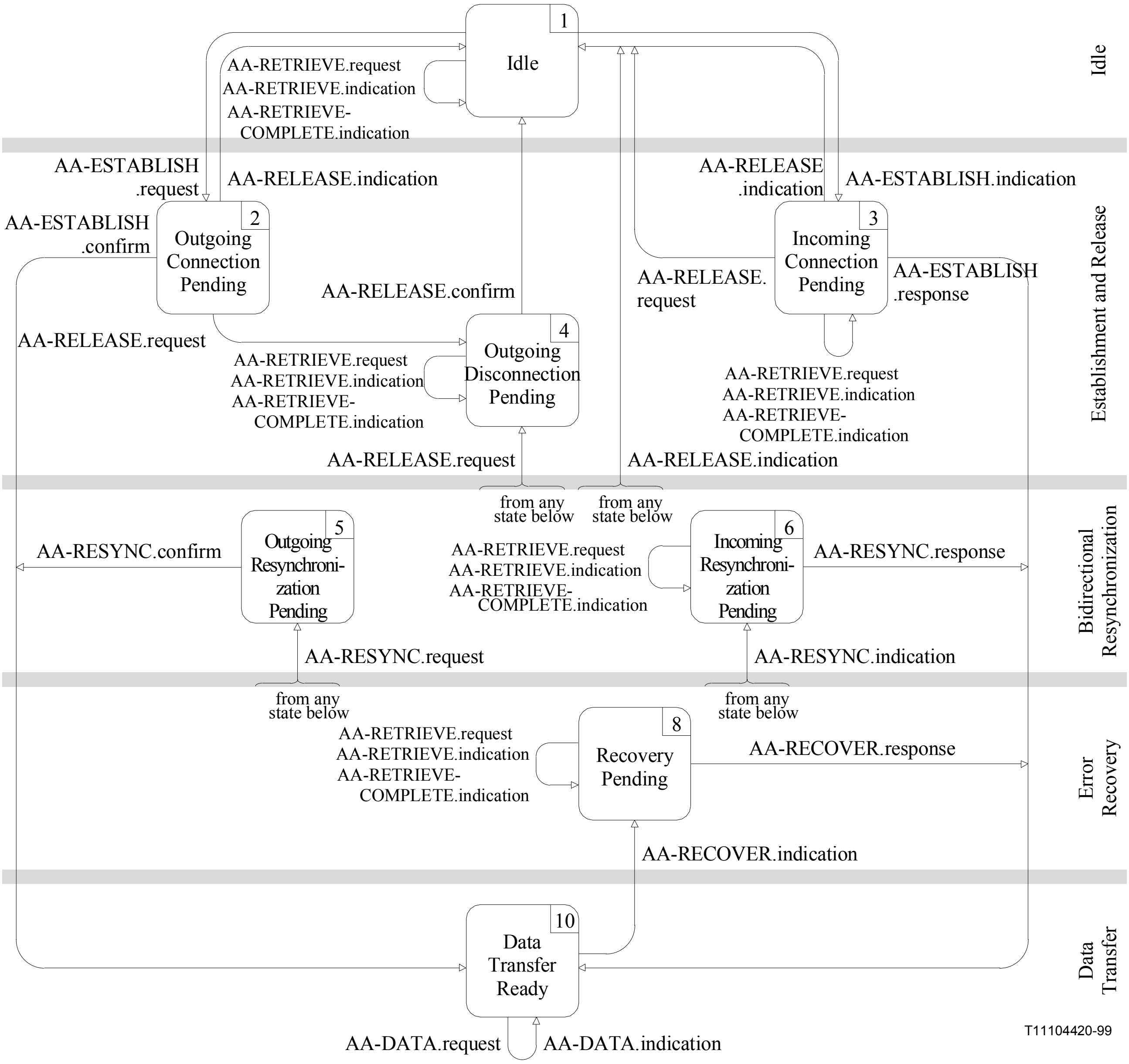
The possible overall sequences of signals at a point-to-point SSCOPMCE endpoint are defined in the state transition diagram, Figure 4. The model illustrates the behaviour of SSCOPMCE as seen by the user. This model assumes that a request or response signal is never issued at the same time as an indication or confirm signal. The model also assumes that the signals are serviced immediately and in zero time. In the diagram:

a) the signals AA-UNITDATA.request and AA-UNITDATA.indication are associated with Unacknowledged Data Transfer and thus permitted in any state – they are not shown;

b) any other signal which is not shown as resulting in a transition (from one state to the same state, or from one state to a different state) is not permitted in that state;

c) it is assumed that the signals passed between SSCOPMCE and its user are coordinated so that collisions do not occur;

d) the Idle state (State 1) reflects the absence of a connection. It is the initial and final state of any sequence, and once it has been re-entered, the connection is released.



NOTE – The SSCOPMCE connection state Recovery Pending (State 8) covers the SSCOPMCE states Recovery Response Pending (State 8) and Incoming Recovery Pending (State 9). Which of these states applies is not visible at the boundary between SSCOPMCE and its user. The State Outgoing Recovery Pending (State 7) is never visible at the boundary between SSCOPMCE and its user.

Figure 4/Q.2111 – State transition diagram for sequences of signals between SSCOPMCE and its user

7.2 Signals between SSCOPMCE and SSCS layer management

NOTE – This subclause contains the existing MAA signals and their parameters unmodified. New signals to add and remove transmission links as well as adjust the value of Timer_RESEQ have been added to those defined in ITU-T Recommendation Q.2110 [6].

The following repertoire of MAA-signals between SSCOPMCE and layer management is defined (see Table 2):

Table 2/Q.2111 – SSCOPMCE signals and parameters between SSCOPMCE and layer management

Primitive Generic Name	Type			
	Request	Indication	Response	Confirm
MAA-ERROR	–	Code, Count	–	–
MAA-UNITDATA	MU	MU	–	–
MAA-SET-TIMER	Tval	–	–	–
MAA-ADD-LINK	LinkID	–	–	–
MAA-REMOVE-LINK	LinkID	LinkID	–	–
– This primitive is not defined. NOTE – This primitive has no parameters.				

7.2.1 Signal definition

The definition of these signals is as follows:

a) **MAA-ERROR signals** are used to report SSCOPMCE protocol errors and certain events to layer management.

b) **MAA-UNITDATA signals** are used for the non-assured, broadcast and point-to-point, transfer of SDUs between SSCOPMCE and peer layer management entities.

c) **MAA-SET-TIMER signals** are used to set the value of the resequence Timer_RESEQ to adjust the delay for requests for retransmission to the environment.

d) **MAA-ADD-LINK signals** are used to add a new transmission link in a multilink environment or to add a new link to the connectionless transport medium.

e) **MAA-REMOVE-LINK signals** are used to remove a transmission link.

7.2.2 Parameter definition

Table 2 lists the parameters associated with each SSCOPMCE signal. The definition of the parameters is as follows:

a) The **Message Unit (MU)** parameter is used during information transfer to convey a variable-length message. In MAA-UNITDATA.request signals, this parameter is mapped transparently into the Information field of an MD PDU. For MAA-UNITDATA.indication signals, this parameter contains the contents of the information field of the received MD PDU. The MU is an integral multiple of one octet.

b) The **code parameter** indicates the type of protocol error that occurred. The Code parameters are defined in Annex A.

c) The **count parameter** indicates the number of SD PDU retransmissions that occurred.

d) The **Tval parameter** indicates the value to be used when Timer_RESEQ is set.

e) The **LinkID parameter** identifies a transmission link.

7.3 Signals between SSCOPMCE and CPCS

NOTE – The structure of this subclause has been adapted according to 7.1 and 7.2.

This Recommendation assumes the CPCS message mode of operation without the option "corrupted data delivery". Use of Streaming Mode is for further study.

The following repertoire of CPCS-signals between the CPCS and SSCOPMCE is defined (see Table 3):

Table 3/Q.2111 – SSCOPMCE signals and parameters between SSCOPMCE and CPCS

Primitive Generic Name	Type	
	Invoke	Signal
CPCS-UNITDATA	ID, LP, CI, CPCS-UU	ID, LP, CI, CPCS-UU

7.3.1 Signal definition

The definition of these signals is as follows:

a) **CPCS-UNITDATA signals** are used for the transport of SSCOP PDUs between the two peer SSCOPMCE protocol entities.

7.3.2 Parameter definition

Table 3 lists the parameters associated with each CPCS signal. The definition of the parameters is as follows:

a) The **Interface Data (ID)** parameter is used during information transfer to convey a variable-length SSCOP PDU from one peer SSCOP protocol entity to the other.

b) The **Loss Parameter (LP)** parameter is not used. For CPCS-UNITDATA.invoke, CPCS-LP is set to "0". For CPCS-UNITDATA.signal, this parameter is ignored.

c) The **Congestion Indicator (CI)** parameter is not used. For CPCS-UNITDATA.invoke, CPCS-CI is set to "0". For CPCS-UNITDATA.signal, the use of this parameter is for further study.

d) The **CPCS User-to-User (CPCS-UU)** parameter is not used. For CPCS-UNITDATA.invoke, CPCS-UU is set to "0" For CPCS-UNITDATA.signal, this parameter is ignored.

7.3.3 Connectionless environment

The signals defined in this subclause were originally defined for the ATM adaptation layer environment; they bear little resemblance to any known connectionless environment. For such environments a convergence function is needed (this is to a large extent a modelling artifact). Such a function for IP or UDP based communications is defined in Annex C.

8 Protocol elements for peer-to-peer communications

8.1 SSCOPMCE PDUs

The Protocol Data Units (PDUs) are listed and described in Table 4.

Table 4/Q.2111 – SSCOPMCE PDU names and definitions

Functionality	PDU name	PDU type field	Description
Establishment	BGN	0001	Request Initialization
	BGAK	0010	Request Acknowledgement
	BGREJ	0111	Connection Reject
Release	END	0011	Disconnect Command
	ENDAK	0100	Disconnect Acknowledgement
Resynchronization	RS	0101	Resynchronization Command
	RSAK	0110	Resynchronization Acknowledgement
Recovery	ER	1001	Recovery Command
	ERAK	1111	Recovery Acknowledgement
Assured Data Transfer	SD	1000	Sequenced Connection-mode Data
	POLL	1010	Transmitter State Information with request for Receive State Information
	STAT	1011	Solicited Receiver State Information
	USTAT	1100	Unsolicited Receiver State Information
Unacknowledged Data Transfer	UD	1101	Unnumbered User Data
Management Data Transfer	MD	1110	Unnumbered Management Data

The SSCOPMCE PDU definitions are:

a) *BGN PDU (Begin)*

The **BGN PDU** is used to establish an SSCOPMCE connection between two peer entities. The BGN PDU requests the clearing of the peer's transmitter and receiver buffers, and the initialization of the peer's transmitter and receiver state variables.

b) *BGAK PDU (Begin Acknowledge)*

The **BGAK PDU** is used to acknowledge the acceptance of a connection request from the peer.

c) *BGREJ PDU (Begin Reject)*

The **BGREJ PDU** is used to reject the connection request of the peer SSCOPMCE entity.

d) *END PDU (End)*

The **END PDU** is used to release an SSCOPMCE connection between two peer entities.

e) *ENDAK PDU (End Acknowledge)*

The **ENDAK PDU** is used to confirm the release of an SSCOPMCE connection.

f) *RS PDU (Resynchronization)*

The **RS PDU** is used to resynchronize the buffers and data transfer state variables.

g) *RSAK PDU (Resynchronization Acknowledge)*

The **RSAK PDU** is used to acknowledge the acceptance of a resynchronization requested by the peer SSCOPMCE entity.

h) *ER PDU (Error Recovery)*

The **ER PDU** is used to recover from protocol errors.

i) *ERAK PDU (Error Recovery Acknowledge)*

The **ERAK PDU** is used to acknowledge the recovery from protocol error.

j) *SD PDU (Sequenced Data)*

The **SD PDU** is used to transfer, across an SSCOPMCE connection, sequentially numbered PDUs containing information fields provided by the SSCOPMCE user.

k) *POLL PDU (Status Request)*

The **POLL PDU** is used to request, across an SSCOPMCE connection, status information about the peer SSCOPMCE entity.

l) *STAT PDU (Solicited Status Response)*

The **STAT PDU** is used to respond to a status request (POLL PDU) received from a peer SSCOPMCE entity. It contains information regarding the reception status of SD PDUs, and credit information for the peer transmitter.

NOTE – The STAT PDU is also sent unprompted in cases the receiver wants to update the credit and no opportunity to send a USTAT or prompted (via a POLL PDU) STAT PDU is available.

m) *USTAT PDU (Unsolicited Status Response)*

The **USTAT PDU** is used to respond to a detection of one or more new missing SD PDUs, based on the examination of the sequence number of the SD PDU. It contains information regarding the reception status of SD PDUs and credit information for the peer transmitter.

n) *UD PDU (Unnumbered Data)*

The **UD PDU** is used for unassured data transfer between two SSCOPMCE users. When an SSCOPMCE user requests unacknowledged information transfer, the UD PDU is used to send information to the peer without affecting SSCOPMCE states or variables. UD PDUs do not carry a sequence number and therefore, the UD PDU may be lost without notification.

o) *MD PDU (Management Data)*

The **MD PDU** is used for unassured management data transfer between two management entities. When a management entity requests unacknowledged information transfer, the MD PDU is used to send information to the peer management entity without affecting SSCOPMCE states or variables. MD PDUs do not carry a sequence number and therefore, the MD PDU may be lost without notification.

An **invalid PDU** is a PDU which:

1) has an unknown PDU type code; or

2) is not 32-bit aligned; or

3) is not the proper length for a PDU of the stated type.

Invalid PDUs shall be discarded without notification to the sender. No additional action is taken as a result of that PDU (length violations from items 2) and 3) above are reported to layer management).

8.2 SSCOPMCE PDU formats

Figures 5 through 18 illustrate the formats of the SSCOP PDUs. There are 16 defined PDU types listed in 8.1. SSCOP PDU fields are defined in 8.5.

Octets

	1			2	3	4
1	SSCOP-UU (maximum MaxUU octets)					
...				PAD (0-3 octets)		
N-1	N(S)					N(SQ)
N	PL	Rsvd	PDU Type	N(W)		
	8 7	6 5	4 3 2 1			

Figure 5/Q.2111 – Begin PDU (BGN PDU)

Octets

	1			2	3	4
1	SSCOP-UU (maximum MaxUU octets)					
...				PAD (0-3 octets)		
N-1	N(S)					N(SQ)
N	PL	Rsvd	PDU Type	N(W)		
	8 7	6 5	4 3 2 1			

Figure 6/Q.2111 – Begin acknowledge PDU (BGAK PDU)

Octets

	1			2	3	4
1	SSCOP-UU (maximum MaxUU octets)					
...				PAD (0-3 octets)		
N-1	Reserved					
N	PL	Rsvd	PDU Type	Reserved		
	8 7	6 5	4 3 2 1			

Figure 7/Q.2111 – Begin reject PDU (BGREJ PDU)

Octets

	1				2	3	4
1	SSCOP-UU (maximum MaxUU octets)						
...					PAD (0-3 octets)		
N-1	Reserved						N(SQ)
N	PL	R	S	PDU Type	Reserved		
	8 7	6	5	4 3 2 1			

Figure 8/Q.2111 – End PDU (END PDU)

Octets

	1		2	3	4
1	Reserved				
2	Reserved	PDU Type	Reserved		
	8 7 6 5	4 3 2 1			

Figure 9/Q.2111 – End acknowledge PDU (ENDAK PDU)

Octets

	1	2	3	4
1	SSCOP-UU (maximum MaxUU octets)			
...		PAD (0-3 octets)		
N-1	N(S)			N(SQ)
N	PL / Rsvd / PDU Type	N(W)		
	8 7 6 5 4 3 2 1			

Figure 10/Q.2111 – Resynchronization PDU (RS PDU)

Octets

	1	2	3	4
1	N(S)			N(SQ)
2	Reserved / PDU Type	N(W)		
	8 7 6 5 4 3 2 1			

Figure 11/Q.2111 – Resynchronization acknowledge PDU (RSAK PDU)

Octets

	1	2	3	4
1	N(S)			N(SQ)
2	Reserved / PDU Type	N(W)		
	8 7 6 5 4 3 2 1			

Figure 12/Q.2111 – Error recovery PDU (ER PDU)

Octets

	1	2	3	4
1	N(S)			N(SQ)
2	Reserved / PDU Type	N(W)		
	8 7 6 5 4 3 2 1			

Figure 13/Q.2111 – Error recovery acknowledge PDU (ERAK PDU)

Octets

	1	2	3	4
1	SSCOP-UU (maximum MaxUU octets)			
...		PAD (0-3 octets)		
N	PL / D / R / PDU Type	N(S)		
	8 7 6 5 4 3 2 1			

Figure 14/Q.2111 – Sequenced data PDU (SD PDU)

Octets

	1	2	3	4
1	N(SQ)	N(PS)		
2	Reserved / PDU Type	N(S)		
	8 7 6 5 4 3 2 1			

Figure 15/Q.2111 – Poll PDU (POLL PDU)

Octets

	1		2	3	4
1	PAD		List element 1		
2	PAD		List element 2		
...			⋮		
L	PAD		List element L		
L+1	N(SS)		N(PS)		
L+2	N(SQ)		N(MR)		
L+3	Reserved	PDU Type	N(R)		
	8 7 6 5	4 3 2 1			

NOTE – List elements are SD PDU N(S).

Figure 16/Q.2111 – Solicited status PDU (STAT PDU)

Octets

	1		2	3	4
1	PAD		List element 1		
2	PAD		List element 2		
3	N(SQ)		N(MR)		
4	Reserved	PDU Type	N(R)		
	8 7 6 5	4 3 2 1			

NOTE – List elements are SD PDU N(S).

Figure 17/Q.2111 – Unsolicited status PDU (USTAT PDU)

Octets

	1			2	3	4
1	Information (maximum MaxSD octets)					
...				PAD (0-3 octets)		
N	PL	Rsvd	PDU Type	Reserved		
	8 7	6 5	4 3 2 1			

Figure 18/Q.2111 – Unnumbered data PDU (UD PDU) or Management data PDU (MD PDU)

The following features of these formats are noted:

8.2.1 Coding conventions

NOTE 1 – The following three paragraphs are an excerpt of the relevant information in 2.1/I.361. Repeating it here makes the Recommendation more transparent.

When a field is contained within a single octet, the lowest bit number of the field represents the lowest order value.

When a field spans more than one octet, the order of bit values within each octet progressively decreases as the octet number increases; the lowest bit number associated with the field represents the lowest order value.

This leads to the following conventions:

- bits within an octet are sent in decreasing order, starting with bit 8;
- octets are sent in increasing order, starting with octet 1;
- for all fields, the first bit sent is the Most Significant Bit (MSB).

This coding conforms to the coding conventions specified in 2.1/I.361 [4].

NOTE 2 – SSCOPMCE is trailer oriented; i.e. the protocol control information is transmitted last.

8.2.2 Padding (PAD) field

a) *SD, MD, and UD PDUs*

Between the end of the SD, MD, or UD PDU information field and the trailer, there will be from 0 to 3 unused octets. These unused octets are called the Padding (PAD) field; they are strictly used as filler octets and do not convey any information. Any coding is acceptable. This padding field complements the PDU to an integral number of 4 octets.

The Pad Length (PL) field in each PDU indicates the number of PAD octets present in the PDU. It can take on any integer value from 0 to 3. If the length of the information field is zero, the PL field is coded as zero.

b) *BGN, BGAK, BGREJ, END, and RS PDUs*

The SSCOPMCE may convey a variable length User-to-User information field (SSCOP-UU) in a BGN, BGAK, BGREJ, END, or RS PDU. If this field is present in a PDU, it is padded with 0 to 3 filler octets in order to complement the PDU length to an integral multiple of 4 octets. These unused octets are called the Padding (PAD) field; they are strictly used as filler octets and do not convey any information. Any coding is acceptable.

The Pad Length (PL) field in each PDU indicates the number of PAD octets present in the PDU. It can take on any integer value from 0 to 3. If no SSCOP-UU field is present in the PDU, the PL field is coded as zero.

c) *STAT and USTAT PDUs*

In any STAT or USTAT PDU containing list elements, a one-octet PAD field precedes the 3-octet sequence number. These unused PAD octets are used to complement each list element field to four octets in length and do not convey any information. Any coding is acceptable.

8.2.3 Reserved field

There is a field of reserved bits (R, Rsvd, Reserved) in each PDU. One function of the reserved field is to achieve 32-bit alignment. Other functions are for further study. Where no functions other than 32-bit alignment are defined, this field shall be coded as zero. This field shall be ignored by the receiver.

8.2.4 PDU length

The maximum length of the information field in SD, UD, and MD PDUs is k octets. The maximum value of k is 65 528 octets. The value of k is established as part of size negotiation procedures outside SSCOPMCE, upon bilateral agreement, may be specified by an SSCF Recommendation utilizing SSCOPMCE, or may be derived from the maximum length PDU size for protocols using SSCOPMCE. The minimum value of k is 0 octets.

The maximum length of a variable length SSCOP-UU field is j octets. The maximum value of j is 65 524 octets. The value of j is established upon bilateral agreement, may be specified by an SSCF Recommendation utilizing SSCOPMCE, or may be derived from requirements of protocols utilizing SSCOPMCE. The minimum value of j is 0 octets.

8.2.5 STAT and USTAT PDU codings

USTAT PDUs contain two list elements. STAT PDUs contain zero or more list elements. Transmitted STAT messages may be segmented into more than one STAT PDU.

The processing of a STAT PDU does not rely on information in other STAT PDUs. This is true even for the case when multiple STAT PDUs are generated in response to a single POLL PDU, and one or more of these STAT PDUs is lost.

The list elements are used for selective retransmission requests. Every odd element represents the first PDU of a gap of missing PDUs, and every even element represents the first PDU of a received sequence, except possibly the last one. Appendix I provides examples on how to code the list elements.

8.3 States of SSCOPMCE protocol entity

This subclause describes the states of an SSCOPMCE entity. These states are used in the specification of the peer-to-peer protocol. The states are conceptual and reflect general conditions of the SSCOPMCE entity in the sequences of signals and PDU exchanges with its user and peer, respectively. In addition, other conditions are used in the description, in order to avoid identification of additional states, as detailed in the SDLs. The basic states are:

State 0 Guard

After creation of a SSCOPMCE entity, this state is entered until the Timer_GUARD expires. During this time, PDUs belonging to a previous creation of an SSCOPMCE entity do no longer exist. In addition, a possible peer entity detected the absence of its peer.

State 1 Idle

Each SSCOPMCE entity enters the Idle state (State 1) upon initialization and returns to this state upon the release of a connection.

State 2 Outgoing Connection Pending

An SSCOPMCE entity requesting a connection with its peer is in the Outgoing Connection Pending state (State 2) until it receives acknowledgement from its peer.

State 3 Incoming Connection Pending

An SSCOPMCE entity that has received a connection request from its peer and is waiting for its user's response is in the Incoming Connection Pending state (State 3).

State 4 Outgoing Disconnection Pending

An SSCOPMCE entity requesting release of the peer-to-peer connection goes to the Outgoing Disconnection Pending state (State 4) until it receives confirmation that the peer entity has released and transitioned to the Idle state (State 1), after which it does the same.

State 5 Outgoing Resynchronization Pending

An SSCOPMCE entity requesting resynchronization of the connection with its peer is in the Outgoing Resynchronization Pending state (State 5).

State 6 Incoming Resynchronization Pending

An SSCOPMCE entity that has received a resynchronization request from its peer and is waiting for its user's response is in the Incoming Resynchronization Pending state (State 6).

State 7 Outgoing Recovery Pending

An SSCOPMCE entity requesting recovery with its peer of an existing connection is in the Outgoing Recovery Pending state (State 7).

State 8 Recovery Response Pending

An SSCOPMCE entity which has completed recovery, notified its user, and is awaiting response is in the Recovery Response Pending state (State 8).

State 9 Incoming Recovery Pending

An SSCOPMCE entity that has received a recovery request from its peer and is waiting for its user's response is in the Incoming Recovery Pending state (State 9).

State 10 Data Transfer Ready

Upon successful completion of the connection establishment, resynchronization, or error recovery procedures, both peer SSCOPMCE entities will be in Data Transfer Ready state (State 10) and assured data transfer can take place.

8.4 SSCOPMCE state variables

This subclause describes the state variables used in the specification of the peer-to-peer protocol.

8.4.1 Value range and arithmetic

SD and POLL PDUs are sequentially and independently numbered and may have the value 0 through n minus 1 (where n is the modulus of the sequence numbers). The modulus equals 2^{24} and the sequence numbers cycle through the entire range, 0 through $2^{24} - 1$. All arithmetic operations on the following state variables and sequence numbers contained in this Recommendation are affected by the modulus: VT(S), VT(PS), VT(A), VT(H), VT(PA), VT(MS), VR(R), VR(H), VR(S), VR(PS), VR(SPS), and VR(MR). In addition, the state variables VT(SQ) and VR(SQ) use modulo 256 arithmetic.

When performing arithmetic comparisons of transmitter variables involving sequence numbers of SD PDUs, VT(A) minus 2^{23} is assumed to be the base; when involving sequence numbers of POLL PDUs, VT(PA) minus 2^{23} is assumed to be the base. When performing arithmetic comparisons of receiver variables involving sequence numbers of SD PDUs, VR(R) minus 2^{23} is assumed to be the base; when involving sequence numbers of POLL PDUs, VR(PS) minus 2^{23} is assumed to be the base.

NOTE – For comparisons involving more than two terms no base arithmetic is required.

When performing arithmetic comparisons of receiver variables involving the sequence number N(SQ) of SD PDUs, VR(SQ) minus 2^7 is assumed to be the base.

The state variables VT(SS) and VR(SS) may assume values from "0" to "255".

The size of the boolean array of state variables PT(x) and PR(x) must be at least equal to the maximum supported links at a time.

The state variable nlinks may assume values from "0" at least to the maximum supported links at a time.

8.4.2 State variables at the transmitter

SSCOPMCE maintains the following state variables at the transmitter.

a) *VT(S) – Send state variable*

The sequence number of the next SD PDU to be transmitted for the first time (i.e. excluding retransmissions). Incremented after transmission of a SD PDU for the first time (i.e. excluding retransmissions).

b) *VT(PS) – Poll Send state variable*

The current value of the poll sequence number. Incremented before transmission of the next POLL PDU.

c) *VT(A) – Acknowledge state variable*

The sequence number of the next in-sequence SD PDU expected to be acknowledged, which forms the lower edge of the window of acceptable acknowledgements. VT(A) is updated upon acknowledgement of in-sequence SD PDUs.

d) *VT(PA) – Poll acknowledge state variable*

The poll sequence number of the next STAT PDU expected to be received, which forms the lower edge of the acceptable N(PS) window for STAT PDUs. If a STAT PDU is accepted, VT(PA) is set to STAT.N(PS).

e) *VT(MS) – Maximum Send state variable*

The sequence number of the first SD PDU not allowed by the peer receiver [i.e. the receiver will allow up to VT(MS) −1]. This value represents the upper edge of the transmit window. The transmitter shall not transmit a new SD PDU if VT(S) ≥ VT(MS). VT(MS) is updated based on receipt of a USTAT PDU, STAT PDU, BGN PDU, BGAK PDU, RS PDU, RSAK PDU, ER PDU, or ERAK PDU.

f) *VT(PD) – Poll Data state variable*

When acknowledgements are outstanding, this state variable represents the number of SD PDUs transmitted between transmission of a POLL PDU, or the number of SD PDUs transmitted before the transmission of the first POLL PDU after Timer_POLL became active. VT(PD) is incremented upon transmission of an SD PDU, and reset to zero upon transmission of a POLL PDU.

g) *VT(CC) – Connection Control state variable*

The number of unacknowledged BGN, END, ER, or RS PDUs. VT(CC) is incremented upon transmission of a BGN, END, ER, or RS PDU. If an END PDU is transmitted in response to a protocol error, SSCOP does not wait for an ENDAK PDU [i.e. SSCOP moves directly to state 1 (Idle)] and VT(CC) is not incremented.

h) *VT(SQ) – Transmitter Connection Sequence state variable*

This state variable is used to allow the receiver to identify retransmitted BGN, ER, and RS PDUs. This state variable is initialized to 0 upon creation of the SSCOPMCE process and incremented and then mapped into the N(SQ) field before the initial transmission of either a BGN, RS, or ER PDU. In addition VT(SQ) is mapped into the N(SQ) field of all other control PDUs with the exception of BGREJ and ENDAK.

j) *VT(P) – POLL requested state variable*

The indication (if TRUE) that a POLL PDU should have been transmitted but the POLL window was closed, i.e. there must be $2^{23} - 2$ STAT PDUs outstanding.

k) *VT(H) – Transmitter Highest Reported SD PDU state variable*

This state variable stores the highest sequence number of SD PDUs reported in any STAT or USTAT PDU. In a STAT PDU the highest reported sequence number is either the one in the last list element or, if no list elements are present, N(R). In the USTAT PDU the highest reported sequence number is the one in list element 2. VT(H) is used to determine whether or not STAT and USTAT PDUs are received in the correct sequence in order to not take into account old credit information. The credit information contained in a USTAT is only taken into account if VT(H) is smaller than list element 2 of the USTAT PDU. The credit information contained in a STAT PDU is not taken into account if VT(H) is higher than the highest reported sequence number in the STAT PDU.

l) *VT(SS) – Transmitter STAT Sequence state variable*

This state variable keeps track of the N(SS), the STAT sequence number contained in received STAT PDUs and is used to determine the sequence of STAT PDUs containing the same Poll Sequence number N(PS). VT(SS) is set to N(SS) when N(PS) in a received STAT PDU is greater than VT(PA) or when N(PS) is equal to VT(PA) and N(SS) is bigger than VT(SS). When N(PS) is equal to VT(PA) and N(SS) is smaller than VT(SS) then the credit information contained in the STAT PDU is ignored.

NOTE 1 – The following transmitter variables are already used in ITU-T Recommendation Q.2110 but have not been recorded in the equivalent subclause of ITU-T Recommendation Q.2110.

m) *VT(TB) – Transmission buffer*

The Transmission buffer is (for the purpose of the definition) an array that is indexed by sequence numbers. Each entry in the array contains the following components:

- MU (Message Unit):

  This component holds the payload of the SD PDU not yet delivered.

- Ind (Indicator):

  This component can assume two different values, namely "E" (empty), "F" (full, i.e. the Message Unit component contains information), and "R" (retransmitted, i.e. the Message Unit component contains information that has at least once been retransmitted).

- Out-of-sequence delivery (D) bit:

  This component holds the value of the Out-of-sequence parameter from the AA-DATA.request primitive.

- Poll sequence number (PS):

  This component holds the value of the poll sequence VT(PS) when the SD PDU has last been transmitted.

NOTE 2 – The sequence number need not be stored in a component, since conceptionally the receive buffer is indexed by the sequence number.

NOTE 3 – In reality, the size of the buffer must be related to the credit given to the transmitter; it might have to store all SD PDUs allowed by the credit (e.g. for retransmission purposes).

n) *Transmission queue*

This state variable is used to hold messages submitted via AA-DATA.requests awaiting transmission. This state variable is a first-in-first-out queue and is not modelled in more detail.

o) *Retransmission queue*

This state variable is used to hold messages awaiting retransmission. This state variable is a first-in-first-out queue and is not modelled in more detail.

p) *Unassured queue*

This state variable is used to hold messages submitted via AA-UNITDATA.requests awaiting transmission. This state variable is a first-in-first-out queue and is not modelled in more detail.

q) *Management queue*

This state variable is used to hold messages submitted via MAA-UNITDATA.requests awaiting transmission. This state variable is a first-in-first-out queue and is not modelled in more detail.

NOTE 4 – Clear-buffers and Credit are transmitter state variables and not SSCOPMCE parameters; their definitions are identical to those found in 8.7/Q.2110 except that "YES" was replaced by "TRUE" and "NO" by "FALSE".

r) *Clear-buffers*

This variable is set upon connection establishment. It holds one of two values: "TRUE" or "FALSE". If this parameter is set to "TRUE", SSCOPMCE can release its transmission buffer and transmission queue upon connection release. If this parameter is set to "FALSE", SSCOPMCE cannot release its transmission buffer and transmission queue upon connection release. Additionally, if this parameter is set to "FALSE", SSCOPMCE cannot release selectively acknowledged messages from its transmission buffer if older ones are still outstanding.

s) *Credit*

This variable is used to coordinate credit notifications to layer management. When SSCOPMCE is blocked from transmitting a new SD PDU due to insufficient credit, "Credit" is assigned the value "FALSE". When SSCOP is permitted to transmit a new SD PDU, "Credit" is assigned the value of "TRUE". Credit is initially assigned "TRUE".

t) *PT(x) — STAT receive state variable*

This state variable is an array of boolean components. The size of the array is equal to the number of ("in-service") links in the set. Each component is set to FALSE when Timer_NO-RESPONSE is started. The receipt of a STAT PDU sets the component associated with the link on which the STAT PDU was received to TRUE. When Timer_NO-RESPONSE expires, any link is removed for which the appropriate PT(x) is still set to FALSE.

8.4.3 State variables at the receiver

SSCOPMCE maintains the following state variables at the receiver:

a) *VR(R) – Receive state variable*

The sequence number of the next in-sequence SD PDU expected to be received. Incremented upon receipt of the next in-sequence SD PDU.

b) *VR(H) – Highest expected state variable*

The sequence number of the next highest expected SD PDU. Updated in two ways:

1) reception of a new SD PDU; and
2) reception of a POLL PDU.

c) *VR(W) – Window size*

This state variable contains the maximum window size and the state variable VR(MR) is derived from it.

d) *VR(MR) – Maximum acceptable Receive state variable*

The sequence number of the first SD PDU not allowed by the receiver [(i.e. the receiver will allow up to VR(MR) −1)]. The receiver shall discard SD PDUs with N(S) ≥ VR(MR), (in one case, such an SD PDU may cause the transmission of a USTAT). Updating VR(MR) is implementation dependent, but VR(MR) should not be set to a value < VR(H). An example of how VR(MR) may be determined is included in Appendix III.

e) *VR(SQ) – Receiver Connection Sequence state variable*

This state variable is used to identify old, current, or new BGN, ER, and RS PDUs. Upon reception of a BGN, ER, or RS PDU, this state variable is compared to the value of N(SQ). If the value of N(SQ) is between VR(SQ) −128 and VR(SQ) −1, the PDU is considered "old" and is discarded; if the value of N(SQ) is equal to VR(SQ), the PDU is considered "current", i.e. retransmitted. In the other cases, the PDU is considered "new", the PDU is processed and VR(SQ) is set to N(SQ).

f) *VR(PS) – Receive poll sequence variable*

The sequence number of the last interpreted POLL PDU received. Updated upon the receipt of a POLL PDU with a "higher" sequence number. This variable assists in determining whether a received POLL PDU from a previous, the current, or a new poll cycle has been received.

g) *VR(SPS) – Receive poll sequence variable for STAT PDUs*

This state variable identifies the poll cycle to which a STAT PDU belongs.

NOTE 1 – VR(SPS) is set to VR(PS) when it is decided to send the first STAT PDU in response to a POLL PDU.

NOTE 2 – The following four state variables in the receiver assist the SSCOP receiver in the resequencing of SD PDUs in a multilink or connectionless environment.

h) *VR(S) – Receive state variable in POLL PDU*

The sequence number of the poll cycle that is contained in the POLL PDU. This variable assists in constructing the STAT PDU when – at expiry of Timer_RESEQ – a POLL PDU is no longer available for this information.

i) *VR(P) – Poll Active state variable*

This state variable is set to TRUE if a POLL PDU has been received but the requested STAT PDU has not yet been transmitted.

j) *VR(SS) – Receiver STAT Sequence state variable*

This state variable keeps track of the number of STAT PDUs transmitted within one POLL cycle and is mapped into N(SS) of the STAT PDU. VR(SS) is set to zero before sending a STAT in response to a POLL PDU and is incremented after transmission of any STAT PDU. The highest valueVR(SS) may take is 255.

k) *RB(R) – Receiver buffer*

The Receiver buffer is (for the purpose of the definition) an array that is indexed by sequence numbers. Each entry in the array contains the following components:

- MU (Message Unit):

  This component holds the payload of the SD PDU not yet delivered.

- Ind (Indicator):

  This component can assume four three different values, namely "E" (empty), "F" (full, i.e. the Message Unit component contains information), "D" (out-of-sequence delivery), and "U" (USTAT PDU sent to request retransmission).

- Tm (Time):

  This component holds the time at which it was detected that an SD PDU is missing.

- Out-of-sequence delivery (D) bit:

  This component holds the value of the Out-of-sequence parameter from the AA-DATA.request primitive.

NOTE 3 – The sequence number need not be stored in a component as conceptionally the receive buffer is indexed by the sequence number.

NOTE 4 – In reality, the size of the buffer must be related to the credit given to the transmitter; it might have to store all but the first SD PDU allowed by the credit (e.g. when awaiting the retransmission of the first SD PDU).

l) *PR(x) – Poll receive state variable*

This state variable is an array of boolean components. The size of the array is equal to the number of ("in-service") links in the set. Each component is set to FALSE at the beginning of the poll cycle. The receipt of a POLL PDU sets the component associated with the link on which the POLL PDU was received to TRUE. When all components are set to TRUE, Timer_RESEQ can be stopped and the STAT PDU transmitted.

8.4.4 Common state variables

SSCOPMCE maintains the following state variables for both the transmitter and the receiver:

a) *LinkSet – Set of link identifiers*

This variable keeps a record of the set of links over which SSCOP currently operates.

b) *nlinks – Number of active links*

This variable keeps track of the number of links over which SSCOPMCE currently operates.

8.5 SSCOPMCE PDU parameters

a) *N(S)*

VT(S) is mapped to N(S) whenever a new SD or POLL PDU is generated and is used to determine lost SD PDUs. VT(S) is also mapped into N(S) whenever a new BGN, BGAK, RS, RSAK, ER, or ERAK PDU is generated and is used to declare the starting point of the sequence numbers in SD and POLL PDUs.

NOTE – When BGAK, RSAK, or ERAK PDUs are retransmitted N(S) is set to VT(A) in some cases.

b) *Information field*

The information field of an SD, MD, or UD PDU is mapped from the "Message Unit" (MU) parameter of an AA-DATA, MAA-UNITDATA, or AA-UNITDATA request, respectively. It is mapped to a "Message Unit" (MU) parameter of an AA-DATA, MAA-UNITDATA, or AA-UNITDATA indication, respectively.

c) *N(PS)*

VT(PS) (after VT(PS) has been incremented) is mapped to N(PS) whenever a POLL PDU is generated. The receiver of a POLL PDU maps the received POLL.N(PS) into the field STAT.N(PS). In addition, to facilitate error recovery procedures, the current value of VT(PS) is mapped into N(PS) and stored in the transmitter buffer with the corresponding SD PDU whenever an SD PDU is sent.

d) *N(R)*

VR(R) is mapped to N(R) whenever a STAT or USTAT PDU is generated.

e) *N(MR)*

VR(MR) is mapped to N(MR) whenever a STAT, or USTAT PDU is generated. This is the basis for credit granting by the receiver when in state 10 (Data Transfer Ready).

f) *N(W)*

VR(W) is mapped to N(W) whenever a BGN, BGAK, RS, RSAK, ER, or ERAK PDU is generated. This is the basis for credit granting by the receiver in the connection control services.

g) *SSCOP-UU*

The SSCOP-UU in a BGN, BGAK, BGREJ, END or RS PDU is mapped to and from the "SSCOP-UU" parameter of the corresponding SSCOP signal.

h) *Source (S) bit*

In an END PDU this bit conveys whether the originator of the release was the SSCOP or the SSCOP user. When the transmission of an END PDU is stimulated by the user, this bit is set to 0. When the transmission of an END PDU is stimulated by the SSCOP, this bit is set to 1. This bit is mapped into the "Source" field of an AA-RELEASE.indication.

i) *N(SQ)*

This field carries the connection sequence value. VT(SQ) is mapped to N(SQ) whenever a new BGN, BGAK, END, RS, RSAK, ER, or ERAK PDU is transmitted. This field is used by the receiver together with VR(SQ) to determine whether a BGN, RS, and ER PDU are old, current, or new.

This field is also used as a transaction identifier in the POLL, STAT, and USTAT PDUs and allows a receiver together with VR(SQ) to distinguish current PDUs.

j) *N(SS)*

VR(SS) is mapped to N(SS) whenever a STAT PDU is generated.

k) *Out-of-sequence delivery (D) bit*

In an SD PDU this bit conveys the information, whether or not out-of-sequence delivery of the contents of the information field of the SD PDU at the receiver is allowed. If the D bit is set to one, out-of-service delivery is allowed. If it is set to zero, in-sequence delivery is required. In addition, the value of the received D bit is mapped into the OOS parameter of the AAL-Data.indication primitive. At the transmitter, the value of the OOS parameter of the AAL-Data.request primitive is mapped into the D bit of the resulting SD PDU.

l) *PDU Type field*

The type field codings are listed in Table 3.

m) *Padding (PAD) field*

See 8.2.2.

n) *Pad Length (PL) field*

See 8.2.2.

o) *Reserved (R, Rsvd, Reserved) field*

See 8.2.3.

8.6 SSCOPMCE timers

With the timers of the transmitter, an SSCOP connection is partitioned into phases.

a) *The active phase*

In this phase, a Timer_POLL is running to assure that the peer receiver is polled often enough (via POLL PDUs) to return its status (STAT PDU); this in turn is needed for advancing the credit window and efficient recovery from transmission errors.

The SSCOP transmitter is always in the active phase if there are SD PDUs to be transmitted or if there are any acknowledgements outstanding.

The POLL and STAT PDUs are subject to transmission errors. SSCOPMCE provides for an uninterrupted flow of information by not insisting on a reply to every POLL PDU. In order to detect a broken connection, a Timer_NO-RESPONSE is running parallel to Timer_POLL. At least one STAT PDU needs to be received on a link before Timer_NO-RESPONSE expires. Failing this, the SSCOPMCE removes the link. If no link remains the SSCOPMCE connection is released.

NOTE 1 – It is possible that the time interval between two consecutively received STAT PDUs reaches twice the value of Timer_NO-RESPONSE without the link being removed.

Timer_POLL is optimized to maintain the flow of information and may be shorter or longer than the round-trip delay. Timer_NO-RESPONSE must be at least the sum of Timer_KEEP-ALIVE and one round-trip delay.

b) *The transient phase*

When Timer_POLL expires (and the POLL PDU has been sent) and there are no outstanding acknowledgements or new data pending receipt of credit, the transient phase is entered. Instead of Timer_POLL the Timer_KEEP-ALIVE is started. Also in this phase, POLL or STAT PDUs may be lost. Such loss is protected by Timer_NO-RESPONSE determining the maximum time interval during which at least one STAT PDU needs to be received on each link.

The transient phase reverts back to the active phase whenever a new SD PDU is transmitted or if at expiry of Timer_KEEP-ALIVE, new data is pending receipt of credit.

The Timer_KEEP-ALIVE is generally greater than Timer_POLL and greater than a round-trip delay; POLL PDUs are transmitted less often.

c) *The idle phase*

When upon receipt of a STAT PDUs on every link the Timer_KEEP-ALIVE is still running, both Timer_KEEP-ALIVE and Timer_NO-RESPONSE are stopped and Timer_IDLE is started. In this phase, no POLL PDUs are sent. At the expiry of Timer_IDLE, the transient phase is entered again.

NOTE 2 – Timer_KEEP-ALIVE only expires when a STAT PDU on at least one link is not received.

NOTE 3 – The absolute maximum tolerated interval between reception of STAT PDUs is the sum of Timer_IDLE and two times Timer_NO-RESPONSE.

The idle phase reverts back to the active phase whenever a new SD PDU is transmitted or whenever new data is pending receipt of credit.

Timer_IDLE may be considerably greater than Timer_KEEP-ALIVE.

d) *Connection control phase*

During establishment and release of a connection and during resynchronization or recovery, transmission of PDUs is protected by the Timer_CC. It determines the time between transmission of BGN, END, RS, and ER PDUs as long as an acknowledgement to these PDUs has not been received.

At entry into the Data Transfer state, the active phase is entered.

Timer_CC should be somewhat greater than a round-trip delay.

The receiver needs one timer, Timer_RESEQ. The value of this timer is based on the transmission time of the "longest" SD PDU on the "slowest" link. The environment for mode "A" (multilink) or "B" (connectionless) does not provide sequence integrity of SSCOPMCE PDUs; hence, if a sequence gap is detected or a POLL PDU received, Timer_RESEQ is started and only at expiry of this timer is a USTAT or a STAT PDU transmitted.

NOTE 4 – In mode "C" the Timer_RESEQ can be set to zero.

The Timer_GUARD is used to delay the start-up of a new SSCOPMCE entity to assure that PDUs belonging to a previous creation of an SSCOPMCE entity do no longer exist. During this delay, a possible peer entity should have detected the absence of its peer and reverted to state 1 (Idle).

Timer_GUARD should be set depending on timer values in the peer SSCOPMCE entity. It should be somewhat greater than the maximum of [(Timer_IDLE + Timer_NO-RESPONSE + Timer_RESEQ, 2 * Timer_NO-RESPONSE) minus the time it takes to re-incarnate the instance of an SSCOPMCE entity].

The values of the SSCOPMCE protocol timers are application specific and may be defined in the appropriate SSCF Recommendation which references this ITU-T Recommendation. The tolerance of protocol timers is not addressed in this ITU-T Recommendation. These timers should be configurable for different operational environments (e.g. signalling vs. data transfer environments, or environments including satellite links).

8.7 SSCOP parameters

NOTE 1 – Clear-buffers and Credit are transmitter state variables and have been moved to 8.4.

The value of each SSCOPMCE protocol parameter is application specific and may be defined in the appropriate SSCF Recommendation which references this ITU-T Recommendation.

a) *MODE*

The mode of operation which can be:

"A" multilink environment;

"B" connectionless environment; or

"C" compatibility environment with Q.2110.

b) *MaxCC*

Maximum value for the state variable VT(CC), corresponding to the maximum number of transmissions of a BGN, END, ER, or RS PDU.

c) *MaxPD*

Maximum acceptable value for the state variable VT(PD) before sending a POLL PDU and resetting VT(PD) to zero. This parameter is an upper limit for counter VT(PD) that sends a POLL PDU at least after every (MaxPD) SD PDUs.

d) *MaxSTAT*

Maximum number of list elements placed in a STAT PDU. When the number of list elements exceeds MaxSTAT, the STAT message shall be segmented. All of the PDUs carrying the segmented STAT message, except possibly the last one, contain MaxSTAT list items. This parameter is not used by the receiver of a STAT PDU for length checking, but is only used by the sender of the STAT message for segmentation purposes. This parameter should be an odd integer greater than or equal to 3.

The default value of MaxSTAT is 67. This parameter can be changed on an implementation basis.

NOTE 2 – The default value causes the STAT PDU to fill 6 ATM cells using AAL type 5 common part. In addition, the total length of a STAT PDU should not exceed the maximum length of an SD PDU.

e) *k*

The maximum number of octets in the Information field of an SD, UD, or MD PDU. The maximum value for k is 65 528 octets, the minimum is 0 octets.

f) *j*

The maximum number of octets in the SSCOP-UU field of a BGN, BGAK, BGREJ, END, or RS PDU. The maximum value for j is 65 524 octets, the minimum is 0 octets.

g) *Timer values*

The value of Timer_CC, Timer_POLL, Timer_KEEP-ALIVE, Timer_IDLE, Timer_NO-RESPONSE, and Timer_GUARD must be set at the creation of an SSCOP protocol entity. On the other hand, Timer_RESEQ is also set to an initial value at creation of the entity; it is expected, though, that layer management may evaluate the error reports of unnecessary retransmission (error code "Y") and adjust the value of the timer.

NOTE 3 – In the SDL diagrams, the following designations of timer values are used: "tCC_val" (Timer_CC), "tpoll_val" (Timer_POLL), "tKA_val" (Timer_KEEP-ALIVE), "tidle_val" (Timer_IDLE), "tNR_val" (Timer_NO-RESPONSE), "treseq_val" (Timer_RESEQ), and "tguard_val" (Timer_GUARD).

8.8 SSCOPMCE credit and flow control

8.8.1 Credit and peer-to-peer flow control

Credit is granted by the SSCOPMCE receiver to allow the peer SSCOPMCE transmitter to transmit new SD PDUs. The process by which a receiver entity determines credit is not subject to standardization, but is related to the buffer availability and the bandwidth/delay of the connection. The credit value is conveyed to the transmitter in the N(W) field of each BGN, BGAK, RS, RSAK, ER, and ERAK PDU as a relative window information, and in the N(MR) field of each STAT and USTAT PDU as a absolute sequence number. In addition, the receiver may transmit an spontaneous STAT PDU to modify the credit value. N(MR) is mapped to the variable VT(MS) at the transmitter. The credit value sent to the transmitter in the STAT and USTAT PDUs is the sequence number of the first SD PDU that the receiver will not accept. The transmitter does not transmit any SD PDUs that exceed the credit allowed. The receiver discards any SD PDUs that exceed the credit allowed, (however, such an SD PDU may cause the transmission of a USTAT PDU).

Previously granted credit can be reduced in order for the receiver to perform flow control, but the receiver credit variable VR(MR) cannot be reduced below the value VR(H). In other words, if a receiver has accepted and acknowledged the receipt of the SD PDU numbered VR(H) – 1, the credit value VR(MR) must be greater than or equal to VR(H).

Two possibilities exist to improve performance and peer-to-peer flow control:

i) After emptying the retransmission queue the transmitter may optionally send a POLL PDU independently of Timer_POLL or the value of MaxPD. In such a situation it is likely that one or more gaps will be closed at the receiver and immediately soliciting an acknowledgement and credit update can be of advantage.

ii) As credit is used to control resource usage at the receiver (e.g. the reception buffer) the times of granting and withdrawing of credit should be under the control of the receiver. Therefore the receiver may transmit a STAT PDU anytime (i.e. not only when prompted by a POLL PDU) when an immediate credit update seems opportune. This could, e.g., be the case after recovery from local receiver congestion or after the closure of a gap in the reception buffer.

The operating window of the protocols at the transmitter is lower bounded by VT(A) and upper bounded by the credit available [VT(MS) – 1]. The modulus of the protocol limits the operating window to $2^{23} - 1$. Therefore, at the receiver, the credit granted, using modulo arithmetic, must be a value between VR(H) and VR(R) + $2^{23} - 1$. If VR(MR) = VR(R) = VR(H), the operating window is zero. If VR(MR) = VR(R) + $2^{23} - 1$, the operating window is maximum.

The SSCOPMCE receiver allocates a buffer to support each connection. In principle, the receiver buffer available should match or exceed the credit granted to the transmitter, to avoid the discard of successfully transmitted data. However, if limited buffers are available for a connection, it is possible to grant credit in excess of the available buffer capacity. This method may obtain a higher throughput than can be achieved by limiting the credit to the available buffer, with the possibility that data may need to be discarded if errors occur. The receiver cannot discard previously received and acknowledged, but not yet delivered, SD PDUs. The receiver must also allocate sufficient buffer capacity to receive and deliver the SD PDU numbered VR(R), at all times unless VR(R) = VR(H) = VR(MR). The granting of credit in excess of buffer capacity should only be performed if limited buffers to support the connection are available and if the SSCOPMCE receiver can still maintain the Quality of Service (QoS) required for the connection through this method.

8.8.2 Local flow control

SSCOPMCE events, such as reception of PDUs and external and internal signals, are normally processed in the order in which they occurred. However, events pertaining to the exchange of SSCOPMCE connection status information have priority over data transfer.

An implementation may detect congestion (for example, a long queuing delay) in its lower protocol layers. If so, data transfer should be temporarily suspended in order to give priority to connection control messages. The means by which an SSCOPMCE entity decides whether or not it is congested depends on the protocol environment, including protocol timer values, and is not subject to standardization.

If an SSCOPMCE entity detects local congestion ("lower layer busy" in the SDL specification), it can elect to suspend the servicing of AA-DATA.request signals, AA-UNITDATA.request signals, and MAA-UNITDATA.request signals. It can also suspend the retransmission of requested SD PDUs. The data transfer procedures allow this to occur without causing protocol errors.

In a multilink environment, scheduling of the transmission of the PDUs on a specific link is not specified (except for the POLL and STAT PDUs that – in mode "A" – must be transmitted on every link). The detection of local congestion ("lower layer busy") pertains to the situation where SD PDUs, UD PDUs, and MD PDUs cannot be transmitted on any link.

Therefore, in terms of transmitting PDUs to the peer receiver, all types of PDUs except SD PDU, UD PDU, and MD PDU are given highest priority. A priority among the SD PDUs, UD PDUs, and MD PDUs is not defined. Among the SD PDUs, retransmissions have priority over new transmission if both types are pending. These priorities are only internal to SSCOPMCE.

The SSCOPMCE's local flow control at its user's interface is implementation dependent.

8.8.3 Network congestion

SSCOPMCE may use a "networks" as "links". Such a network may experience internal network congestion resulting in the loss of PDUs. Via adjusting the credit given to the transmitter, layer management of the receiver may control the amount of data the transmitter may send and can thus react to network congestion. The exact method of these reactions is beyond the scope of this ITU-T Recommendation.

9 Specification of SSCOPMCE

This clause provides a set of SDL diagrams defining the procedures of the Service Specific Connection Oriented Protocol in a multilink and connectionless environment. These SDL diagrams are the definitive description of the procedures and in case of conflict with the text, the SDL diagrams take precedence.

A Protocol Implementation Conformance Statement (PICS) proforma can be found in Appendix I. The PICS proforma is normative in the sense that if a Protocol Implementation Conformance Statement is made, this proforma shall be used.

9.1 Overview

Figure 19 gives an overview over the states of SSCOPMCE and the major transitions between them. The states allow SSCOPMCE a number of connection control services and their relation. Examples of SSCOPMCE operations are shown in Appendix II.

9.1.0 Guard

After creation of a SSCOPMCE entity, this state is entered until the Timer_GUARD expires. During this time, PDUs belonging to a previous creation of an SSCOPMCE entity do no longer exist. In addition, a possible peer entity detected the absence of its peer.

9.1.1 Idle

In this state (State 1: Idle), no connection is established. Only unassured and management data may be communicated.

9.1.2 Establishment and release

The states in this connection control service assist the SSCOPMCE user in establishing and releasing connections for the assured data transfer service. Establishment and release takes precedence over all of the other connection control services (resynchronization and recovery). The following states are defined:

- *State 2 – Outgoing Connection Pending*

  In this state, the local user instructed SSCOPMCE to establish a new connection with its peer and awaits the peer's response.

- *State 3 – Incoming Connection Pending*

  In this state, SSCOPMCE received the indication that its peer wants to establish a new connection and has notified its user. It is awaiting the response of that user.

- *State 4 – Outgoing Disconnection Pending*

  In this state, the user has instructed the SSCOPMCE to, or the SSCOPMCE itself has initiated, release the current connection. It awaits confirmation from its peer.

9.1.3 Bidirectional resynchronization

The states in this connection control service assist the SSCOPMCE in a resynchronization of both data transfer directions. The bidirectional resynchronization takes precedence over the recovery service. The following states are defined:

- *State 5 – Outgoing Resynchronization Pending*

  In this state, the local user initiated a resynchronization. SSCOPMCE's peer has been informed and its response is awaited.

- *State 6 – Incoming Resynchronization Pending*

  In this state, the peer SSCOPMCE has requested a resynchronization. The SSCOPMCE user has been notified and its response is awaited.

9.1.4 Recovery

The states in this connection control service assist SSCOPMCE in recovering from protocol errors relating to the assured data transfer (sequence number problems). The following states are defined:

- *State 7 – Outgoing Recovery Pending*

  In this state, the SSCOPMCE has detected a sequence number problem and has instructed its peer to recover. The confirmation from its peer is awaited.

- *State 8 – Recovery Response Pending*

  In this state, the SSCOPMCE that detected the sequence number problem received the confirmation from its peer, informed its user and is awaiting the response from that user.

- *State 9 – Incoming Recovery Pending*

  In this state, the peer SSCOPMCE detected a sequence number problem and informed this SSCOPMCE which in turn informed its user. The response of that user is awaited.

9.1.5 Data transfer

This state permits assured data transfer. Connection establishment, release, resynchronization, and recovery procedures will take the state machine out of this state.

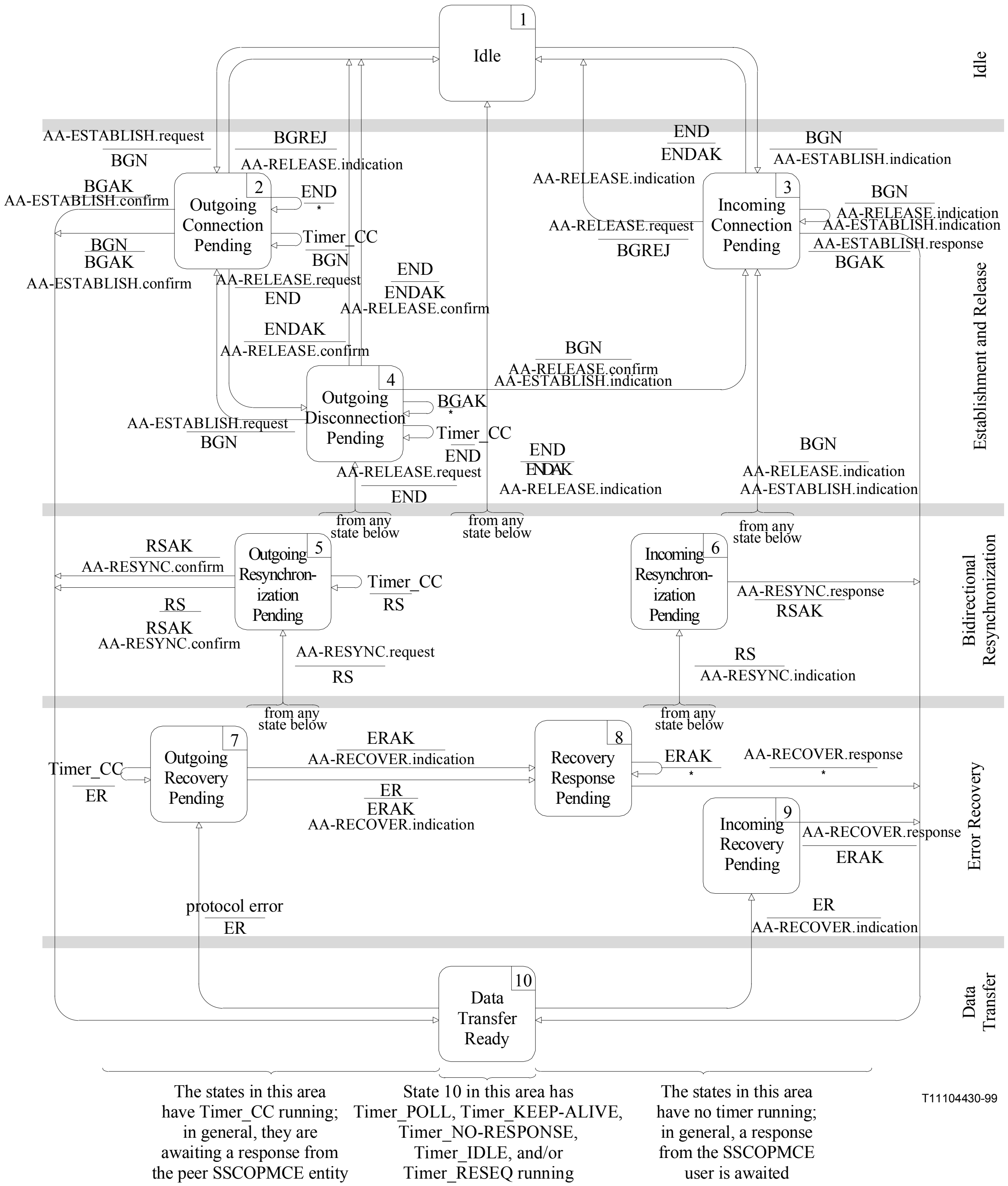
- *State 10 – Data Transfer Ready*

  In this state, assured data transfer takes place.

9.2 SDL diagrams

The SDL diagrams are represented in Figures 20 to 22.

NOTE – The operation of SSCOPMCE remains essentially the same as defined in ITU-T Recommendation Q.2110 [6]. The main modification lies with the fact that the receiver can no longer rely on the sequence integrity of a CPCS connection.



NOTE – The state 0 (Guard) is not shown.

Figure 19/Q.2111 – Overview of SSCOPMCE states and major transitions between them

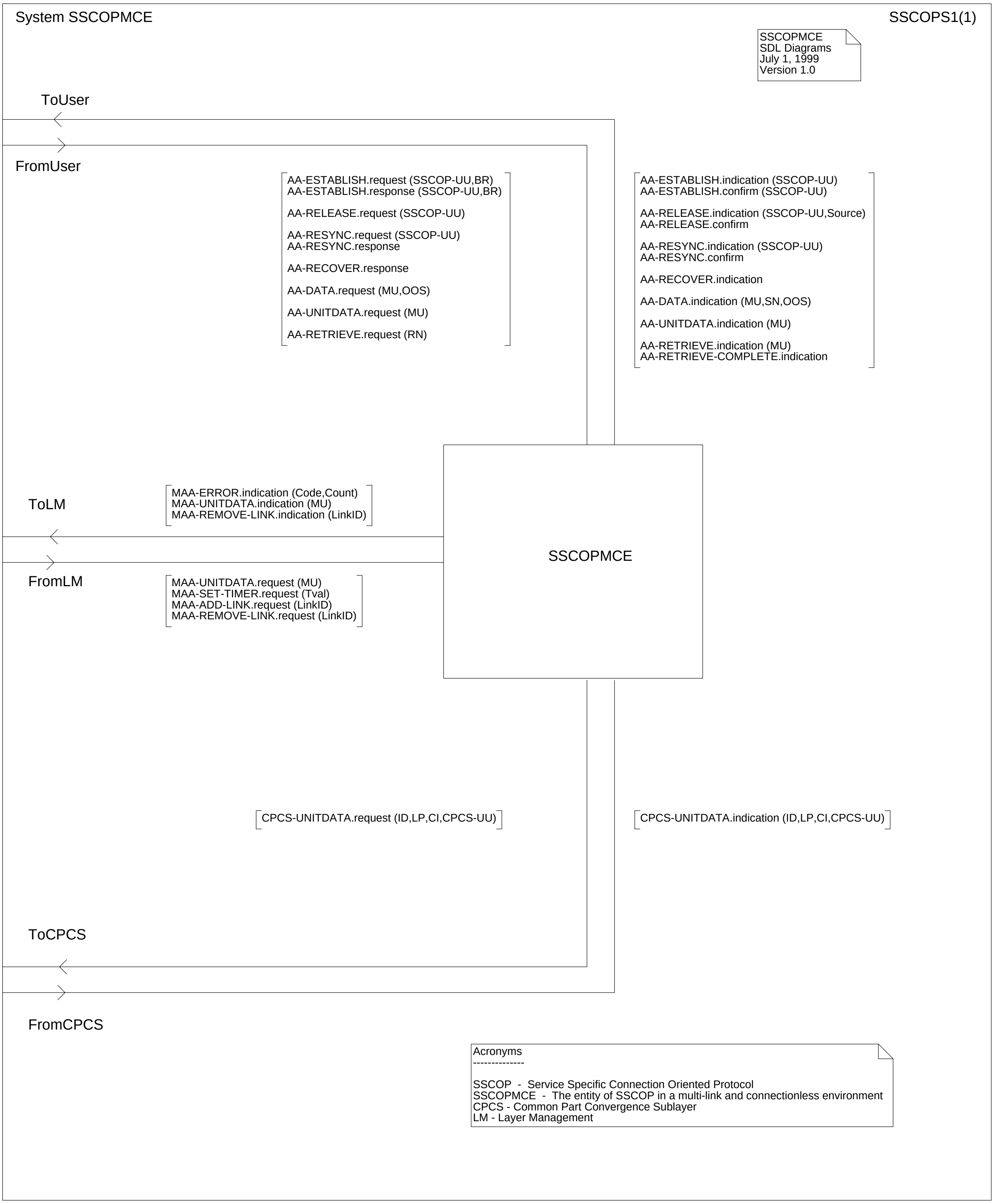


Figure 20/Q.2111 – SDL system of the multilink SSCOPMCE protocol entity

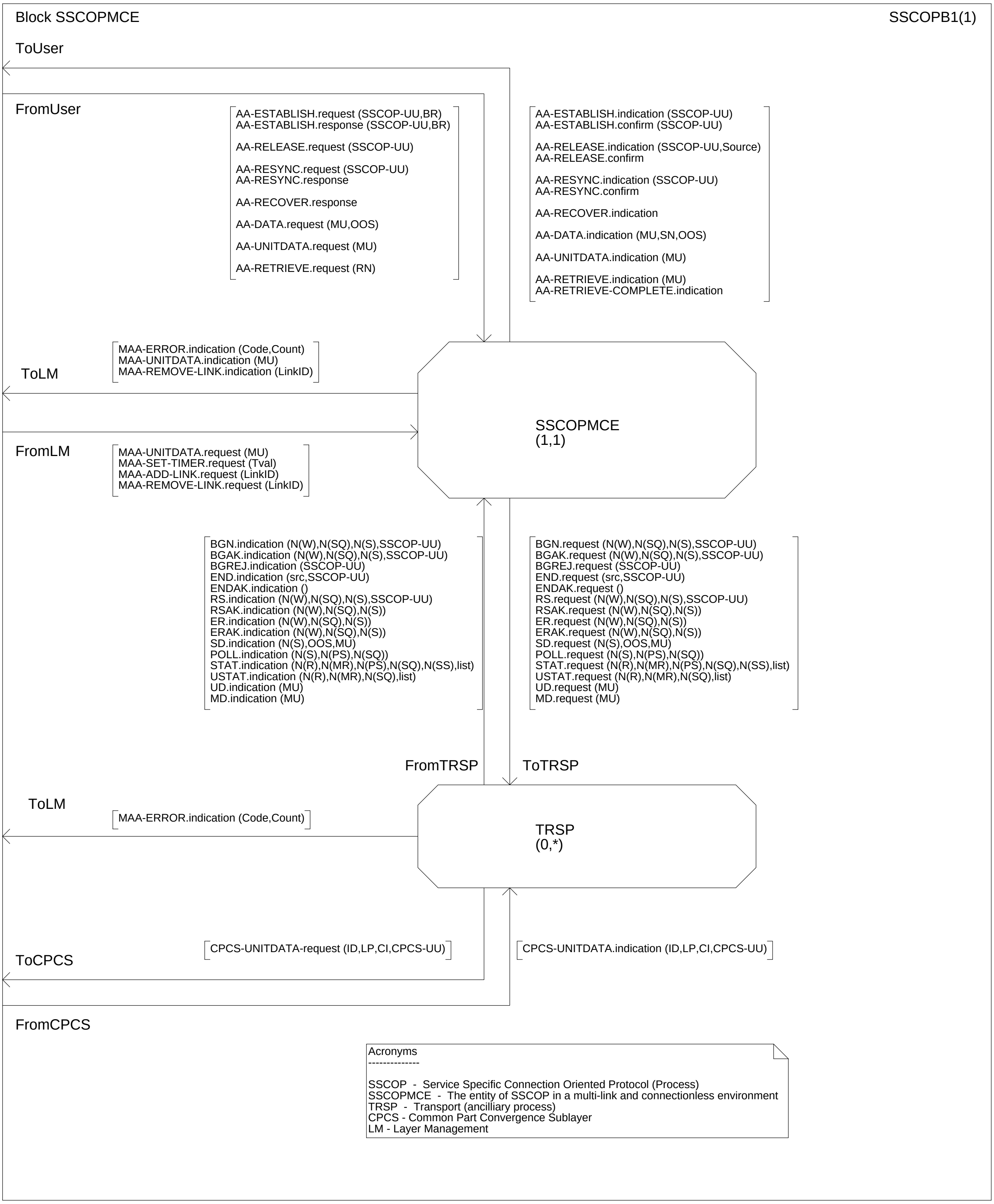


Figure 21/Q.2111 – SDL block structure of the multilink SSCOPMCE protocol entity

Process SSCOPMCE SSCOP01(38)

Signals to/from User (defined in subclause 7.1; parameters are listed between parentheses):

AA-ESTABLISH.request (SSCOP-UU,BR)
AA-ESTABLISH.indication (SSCOP-UU)
AA-ESTABLISH.response (SSCOP-UU,BR)
AA-ESTABLISH.confirm (SSCOP-UU)

AA-RELEASE.request (SSCOP-UU)
AA-RELEASE.indication (SSCOP-UU,Source)
AA-RELEASE.confirm

AA-RESYNC.request (SSCOP-UU)
AA-RESYNC.indication (SSCOP-UU)
AA-RESYNC.response
AA-RESYNC.confirm

AA-RECOVER.indication
AA-RECOVER.response

AA-RETRIEVE.request (RN)
AA-RETRIEVE.indication (MU)

AA-RETRIEVE-COMPLETE.indication

AA-DATA.request (MU,OOS)
AA-DATA.indication (MU,SN,OOS)

AA-UNITDATA.request (MU)
AA-UNITDATA.indication (MU)

Default Parameter Values of SSCOP signals In order to simplify the SDL representation of SSCOP, the SDL diagrams assume default values of parameters in SSCOP indication and confirm primitives. Unless otherwise specified in the SDL diagrams, the parameters of the indication and confirm primitives shall contain the default values specified here (described by the format "PDU.field")

AA-ESTABLISH.indication	SSCOP-UU	BGN.SSCOP-UU
AA-ESTABLISH.confirm	SSCOP-UU	BGAK.SSCOP-UU
AA-RELEASE.indication	SSCOP-UU	END.SSCOP-UU
	Source	(no default)
AA-DATA.indication	MU	SD.information
	SN	SD.N(S)
	OOS	SD.D
AA-RESYNCH.indication	SSCOP-UU	RS.SSCOP-UU
AA-UNITDATA.indication	MU	UD.information
AA-RETRIEVE.indication	MU	Retrieved information
MAA-UNITDATA.indication	MU	MD.information

Signals to/from Layer Management (defined in subclause 7.2; parameters are listed between parantheses)

MAA-ERROR.indication (Code,Count)

MAA-UNITDATA.request (MU)
MAA-UNITDATA.indication (MU)

MAA-ADD-LINK.request (LinkID)
MAA-REMOVE-LINK.request (LinkID)
MAA-REMOVE-LINK.indication (LinkID)

MAA-SET-TIMER.request (Tval)

Signals to/from CPCS (defined in subclause 7.3; parameters are listed between paranthese)

CPCS-UNITDATA.invoke (ID,LP,CI,CPCS-UU)
CPCS-UNITDATA.signal (ID,LP,CI,CPCS-UU)

Messages to/from SSCOP (defined in subclause 8.1; messages are placed in the ID parameter of the CPCS-UNITDATA.invoke and CPCS-UNITDATA.signal primitive)

BGAK, BGN, BGREJ, END, ENDAK, ER, ERAK, MD, POLL, SD, STAT, RS, RSAK, UD, USTAT

NOTE -- The construction of the PDUs and placement in the ID parameter is performed by the ancilliary process TRSP; this process similarly decomposes received ID parameters into its constituent parts.

In order to simplify the SDL representation of SSCOP, the SDL diagrams assume default values for the fields in the SSCOP PDUs. Unless otherwise specified in the SDL diagrams, the fields (i.e., SSCOP PDU parameters) of the transmitted SSCOP PDUs shall be assigned the default values specified here (default values are either state variables, signal parameter values, or received PDU parameters).

BGN	N(W)	VR(W)
	N(S)	VT(S) (note 1)
	N(SQ)	VT(SQ)
	SSCOP-UU	AA-ESTABLISH.request (SSCOP-UU)
BGAK	N(W)	VR(W)
	N(S)	VT(S) (note 1, 2)
	N(SQ)	VT(SQ)
	SSCOP-UU	AA-ESTABLISH.response (SSCOP-UU)
BGREJ	SSCOP-UU	AA-RELEASE.request (SSCOP-UU)
END	SSCOP-UU	AA-RELEASE.request (SSCOP-UU)
	S	"0"
RS	N(W)	VR(W)
	N(S)	VT(S) (note 1)
	N(SQ)	VT(SQ)
	SSCOP-UU	AA-RESYNCH.request (SSCOP-UU)
RSAK	N(W)	VR(W)
	N(S)	VT(S) (note 1, 2)
	N(SQ)	VT(SQ)
ER	N(W)	VR(W)
	N(S)	VT(S) (note 1)
	N(SQ)	VT(SQ)
ERAK	N(W)	VR(W)
	N(S)	VT(S) (note 1, 2)
	N(SQ)	VT(SQ)
SD	N(S)	VT(S)
	D	AA-DATA.request (OOS)
	Information	AA-DATA.request (MU)
POLL	N(PS)	VT(PS)
	N(S)	VT(S)
	N(SQ)	VT(SQ)
STAT	N(R)	VR(R)
	N(MR)	VR(MR)
	N(PS)	VR(SPS)
	N(SQ)	VT(SQ)
	N(SS)	VR(SS)
USTAT	N(R)	VR(R)
	N(MR)	VR(MR)
	N(SQ)	VT(SQ)
UD	Information	AA-UNITDATA.request (MU)
MD	Information	MAA-UNITDATA.request (MU)

Notes
1. If MODE (of operation) is set to "C", i.e., compatibility mode with Q.2110, the field N(S) is set to zero (compatible with a reserved field in Q.2110).

2. In some retransmitted PDUs, N(S) is set to VT(A).

Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity ***(sheet 1 of 38)***

Process SSCOPMCE SSCOP02(38)

Note 1 (on the use of queues):
To enable a satisfactory representation of the SSCOP entity, conceptual queues for the SD, MD, and UD PDUs have been explicitely brought out. These conceptual queues are finite but unbounded and should in no way restrict the implementation of the point-to-point procedures. Three state variables have been provided in order to cause the servicing of these queues to be initiated: "nSDPDUs", "nUDPDUs", and "nMDPDUs". In the SDL diagrams, these variables (when non-zero) serve as permanent enabling conditions and are treated with higher priority than other signals entering this process. The data itself is kept in the queues and buffers.

The SDL diagrams assume the following queues and buffers at the transmitter: Transmission queue (assured data not yet sent), Retransmission queue (SD PDUs that are waiting to be resent), Unassured queue (unassured data that has not yet been sent), Management queue (management data that has not yet been sent), and Transmission buffer (SD PDUs awaiting acknowledgement). At the receiver, a Receiver buffer is assumed for resequencing of SD PDUs.

Note 2:
Signals which are ignored for a given state (inopportune signals) are not included in the SDL diagrams.

Note 3:
The responses by the SSCOP entity to the reception of an inappropriate PDU are described by these SDL diagrams.

Note 4:
Modulo arithmetic is performed on the following state variables: VT(S), VT(A), VT(MS), VR(R), VR(H), VR(MR), VT(PS), VT(PA), VR(PS), VT(SQ), and VR(SQ). VT denotes a transmitter variable, and VR denotes a receiver variable. The modulus equals 2E24 for the first eight variables listed and the modulus equals 256 for VT(SQ) and VR(SQ). For modulo comparisons involving the state variables VT(S), VT(A), and VT(MS), the base for comparisons is VT(A)-2E23. For modulo comparisons involving the state variables VR(R), VR(H), and VR(MR), the base for comparisons is VR(R)-2E23. For modulo comparisons involving the state variables VT(PS), VT(PA), and VR(PS) the base for comparisons is VT(PA)-2E23.

Note 5:
The variables "i", "seq1", "seq2", "Count", and "List_Length" are used in the SDL diagrams. These are loop variables used to illustrate the loop only and do not constitute SSCOP state variables or parameters.

Note 6:
Where "NULL" codings of the SSCOP-UU PDU field are specified, this implies that the field is absent and the PL field is coded as zero.

Note 7:
Whenever a PDU containing an N(W) field is resent identical to the last PDU sent, the N(W) field may reflect an updated VR(W) value.

Note 8:
If no definition is stated on which link a particular PDU is transmitted, the transmitter selects any one of the available links for the transfer of the PDU. PDUs may be assigned to the individual links on a round robin basis. On the other hand, the scheduling mechanism could be based on expected arrival times taking into consideration not only the length of the PDU to be transmitted but also the transfer characteristics and the quality of service of the individual links. Such scheduling mechanisms are implementation dependent, and are not defined in this Recommendation.

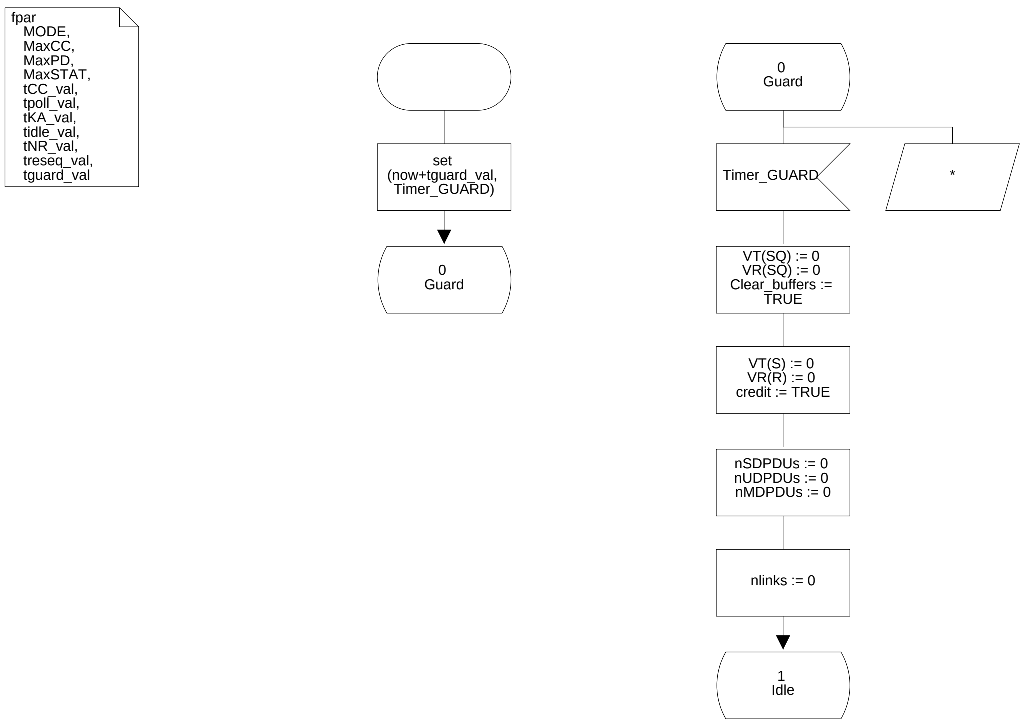


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 2 of 38)*

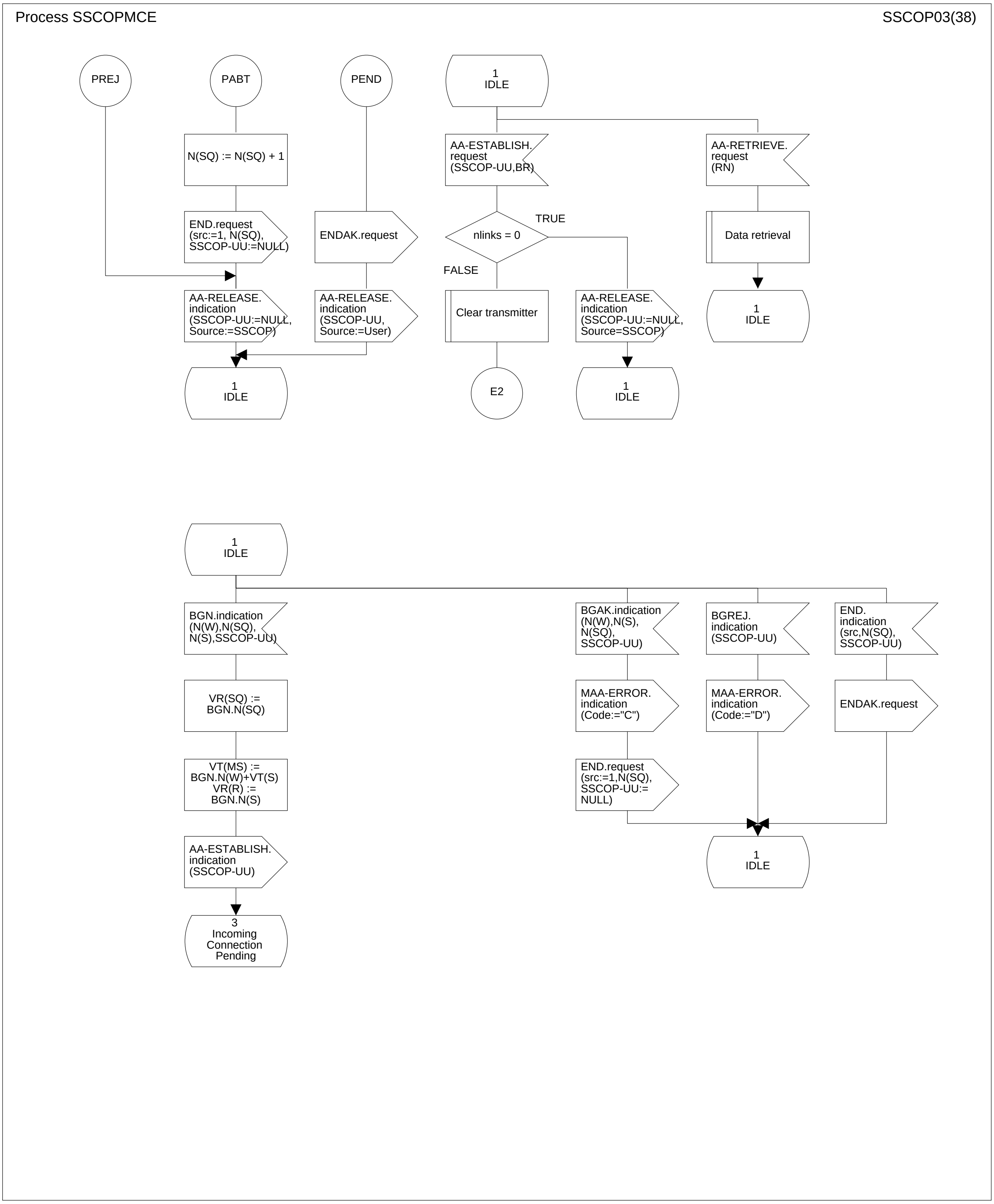


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity ***(sheet 3 of 38)***

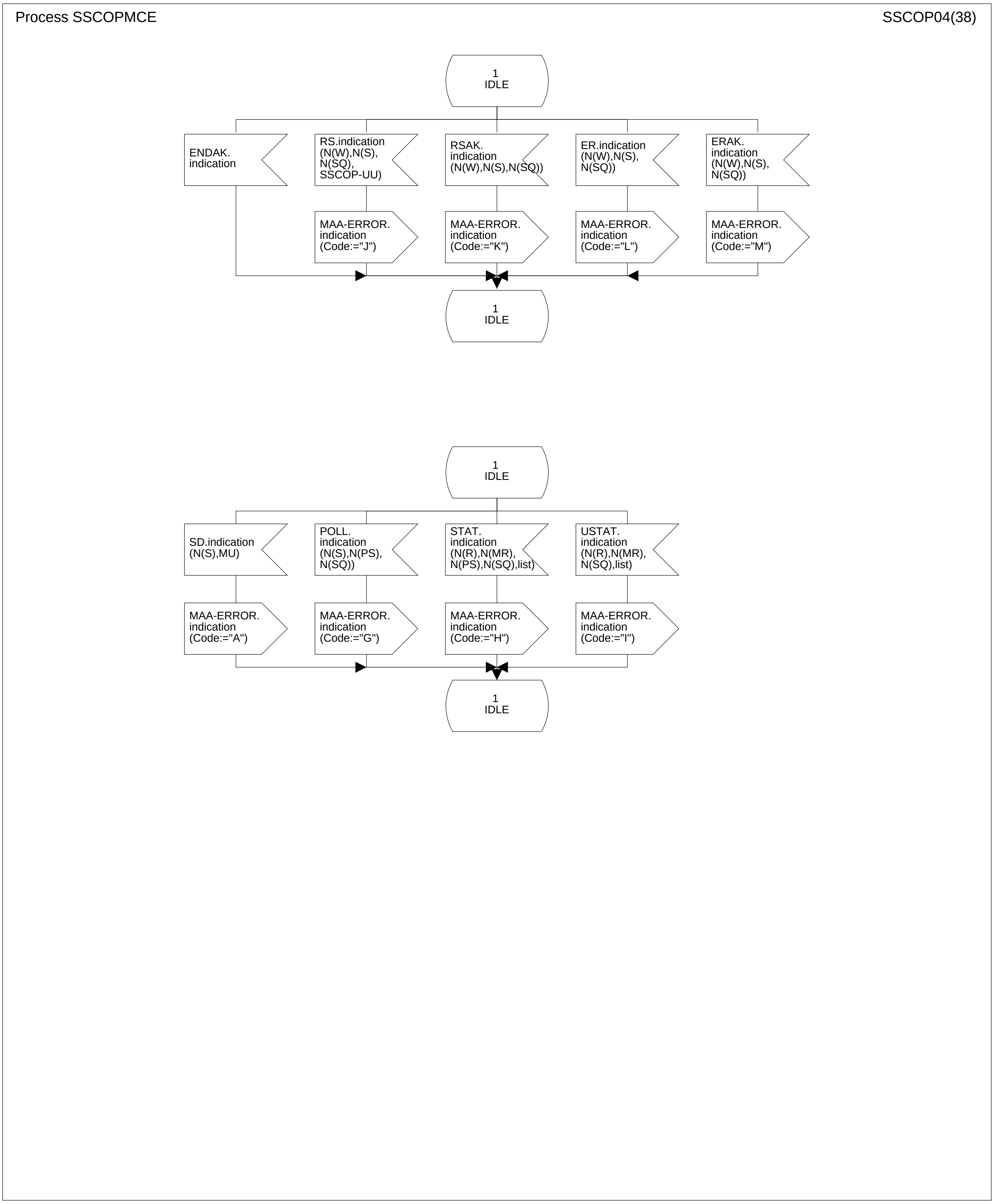


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity ***(sheet 4 of 38)***

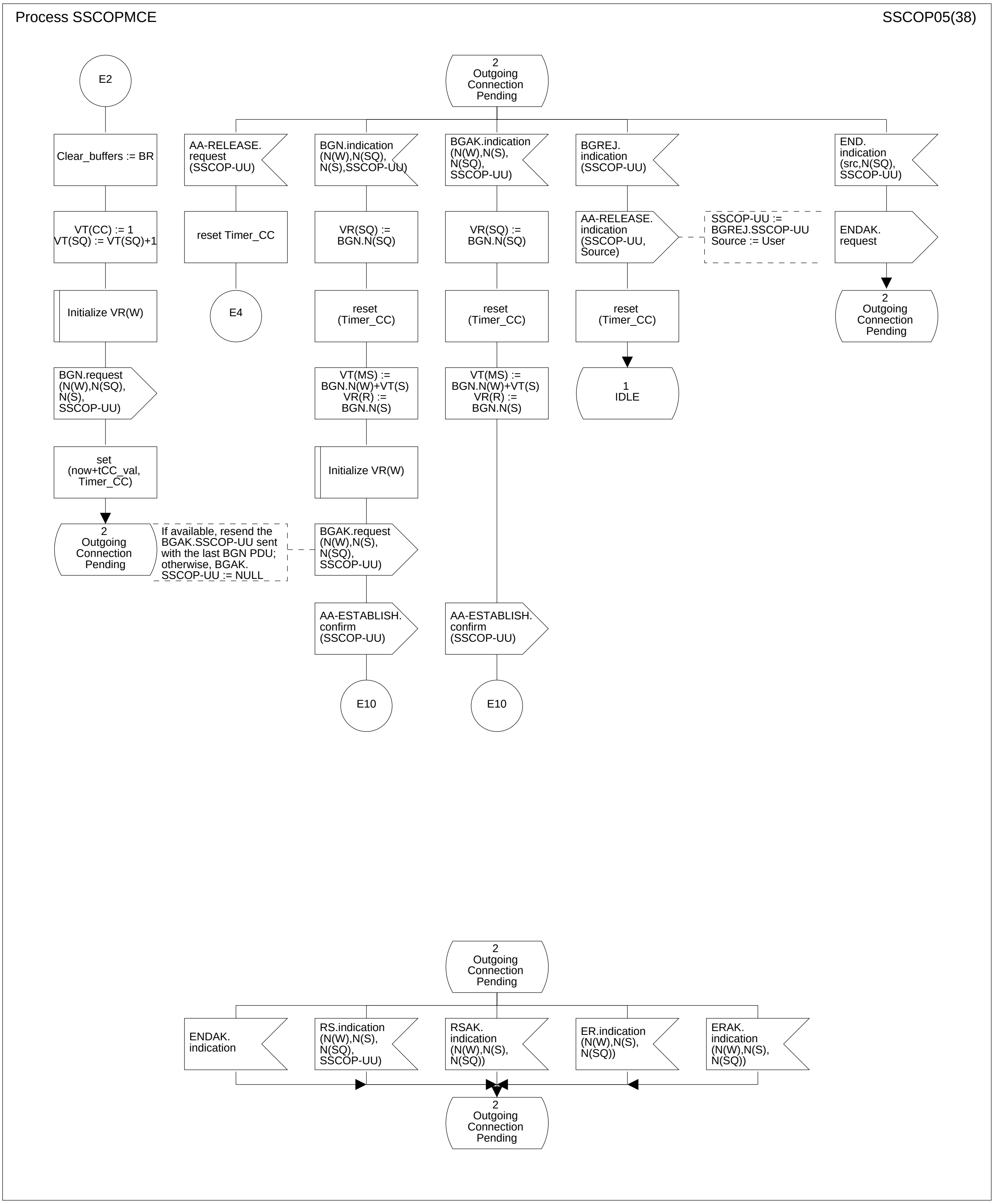


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity ***(sheet 5 of 38)***

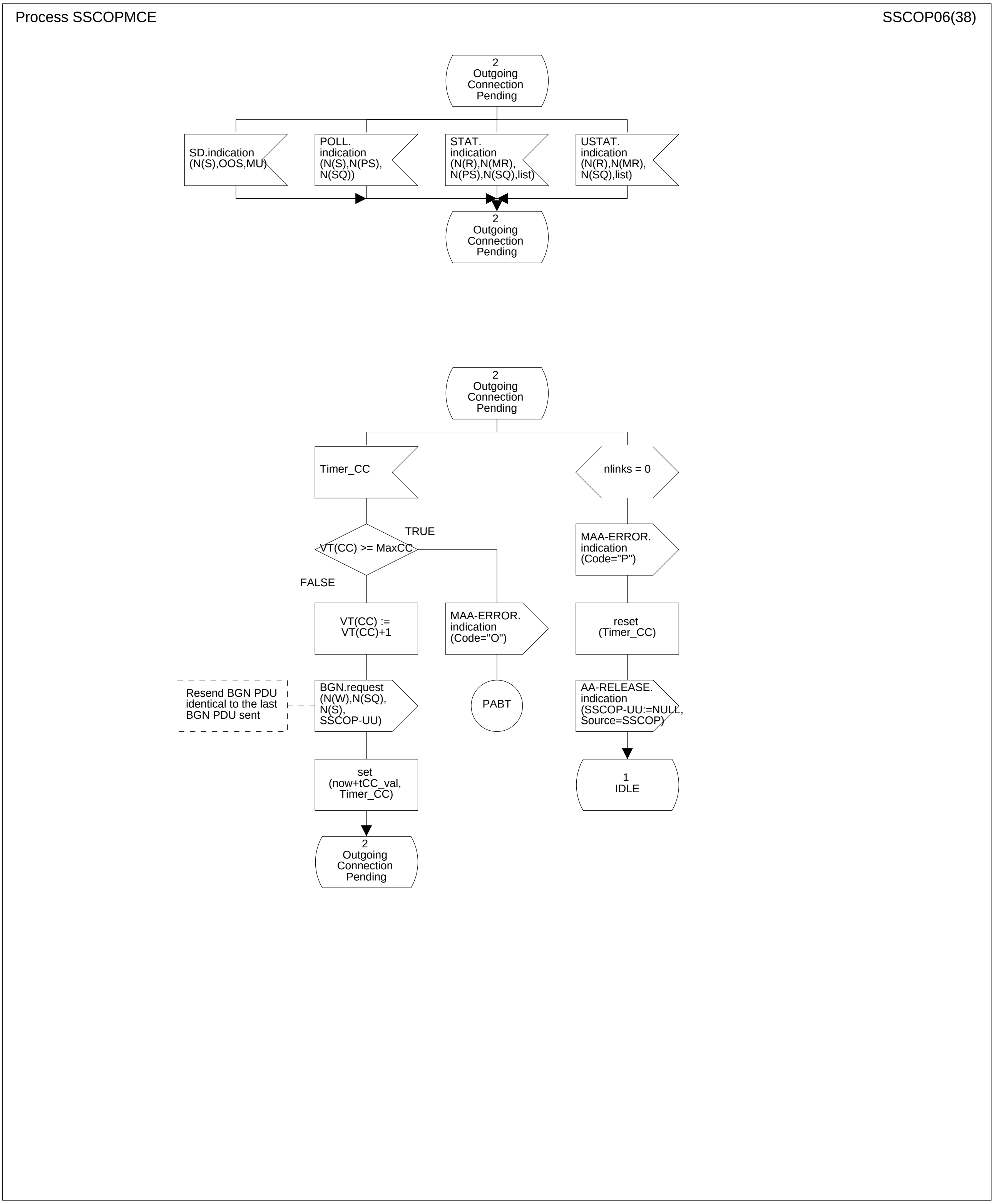


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 6 of 38)*

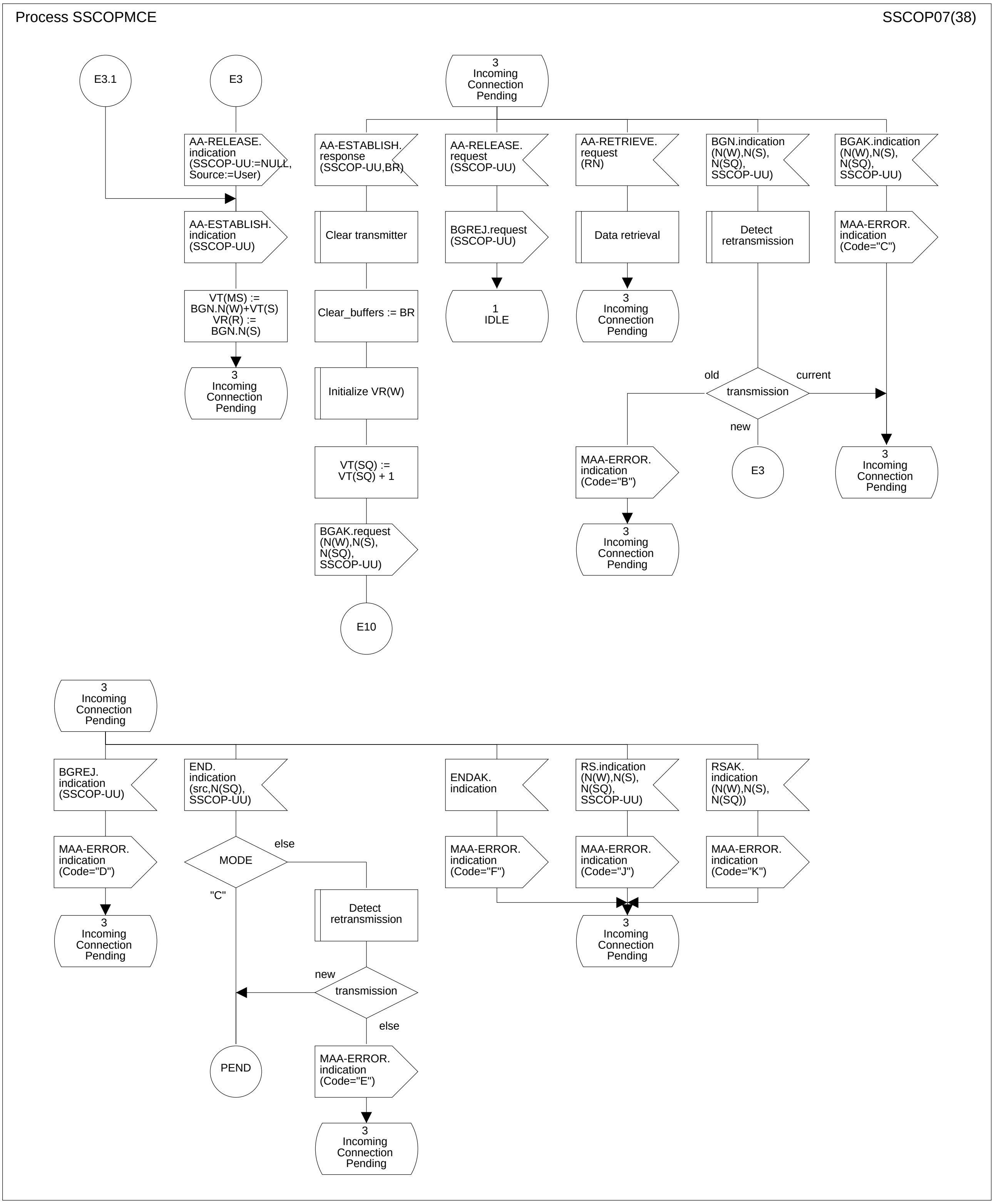


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 7 of 38)*

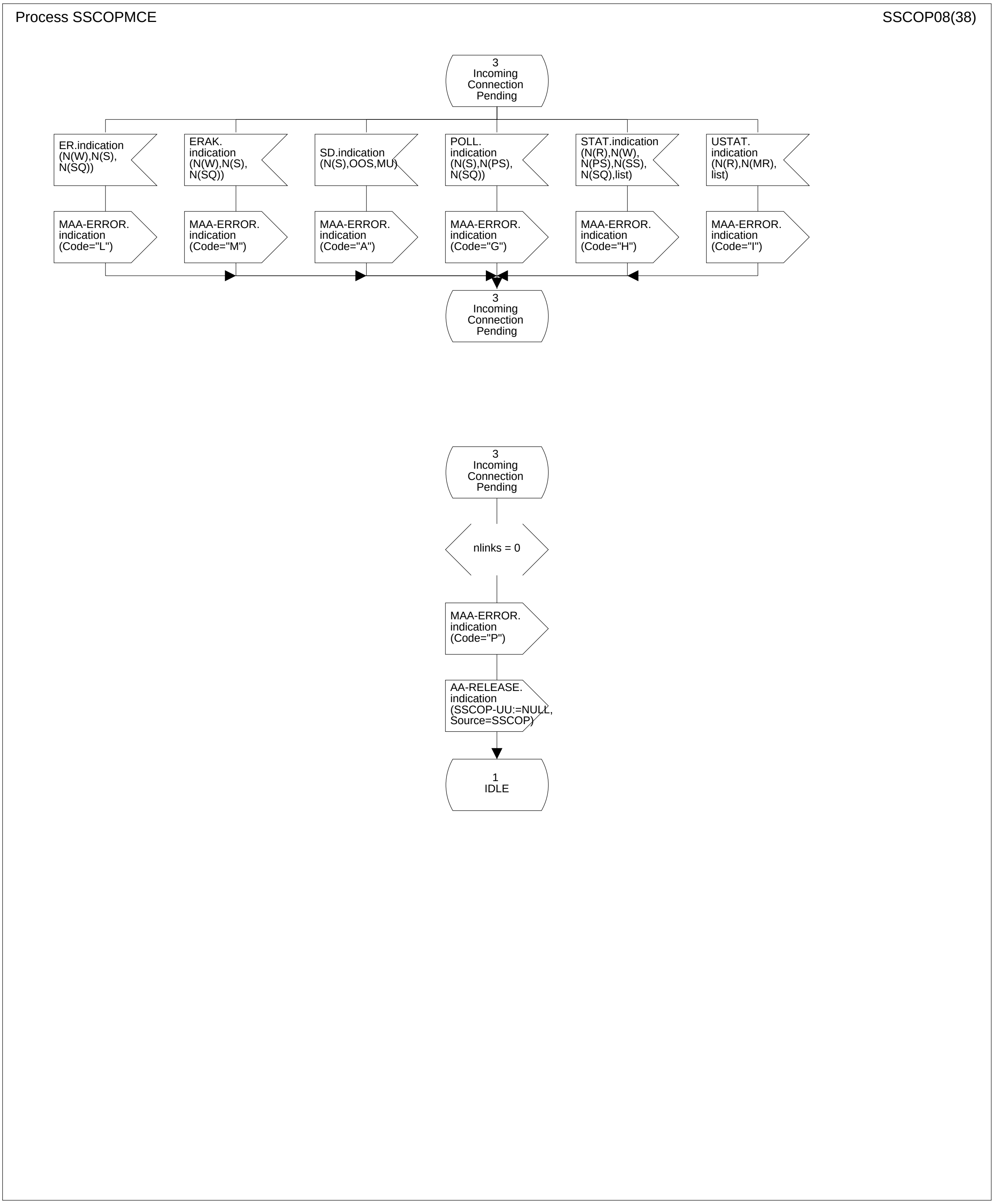


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 8 of 38)*

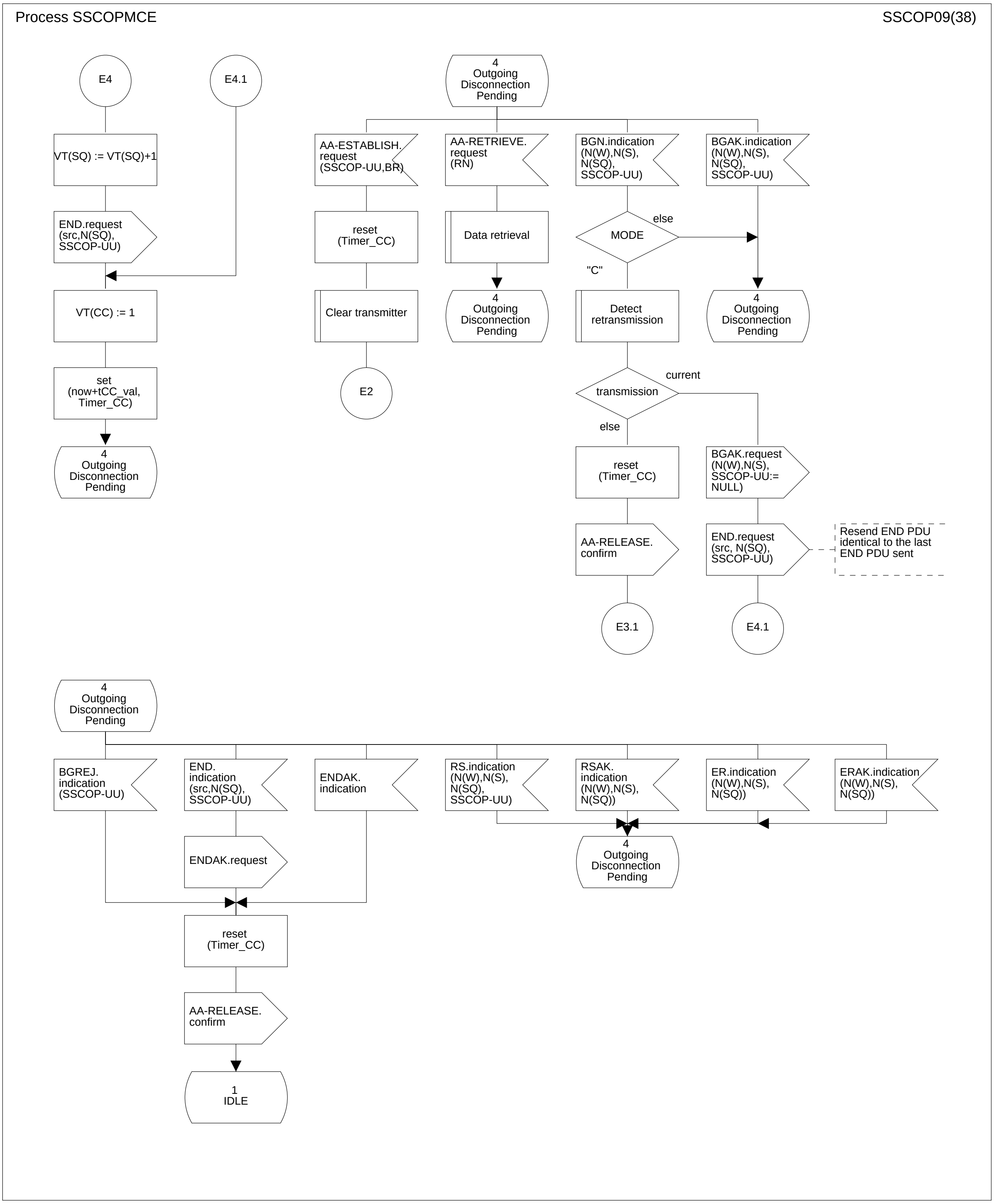


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 9 of 38)*

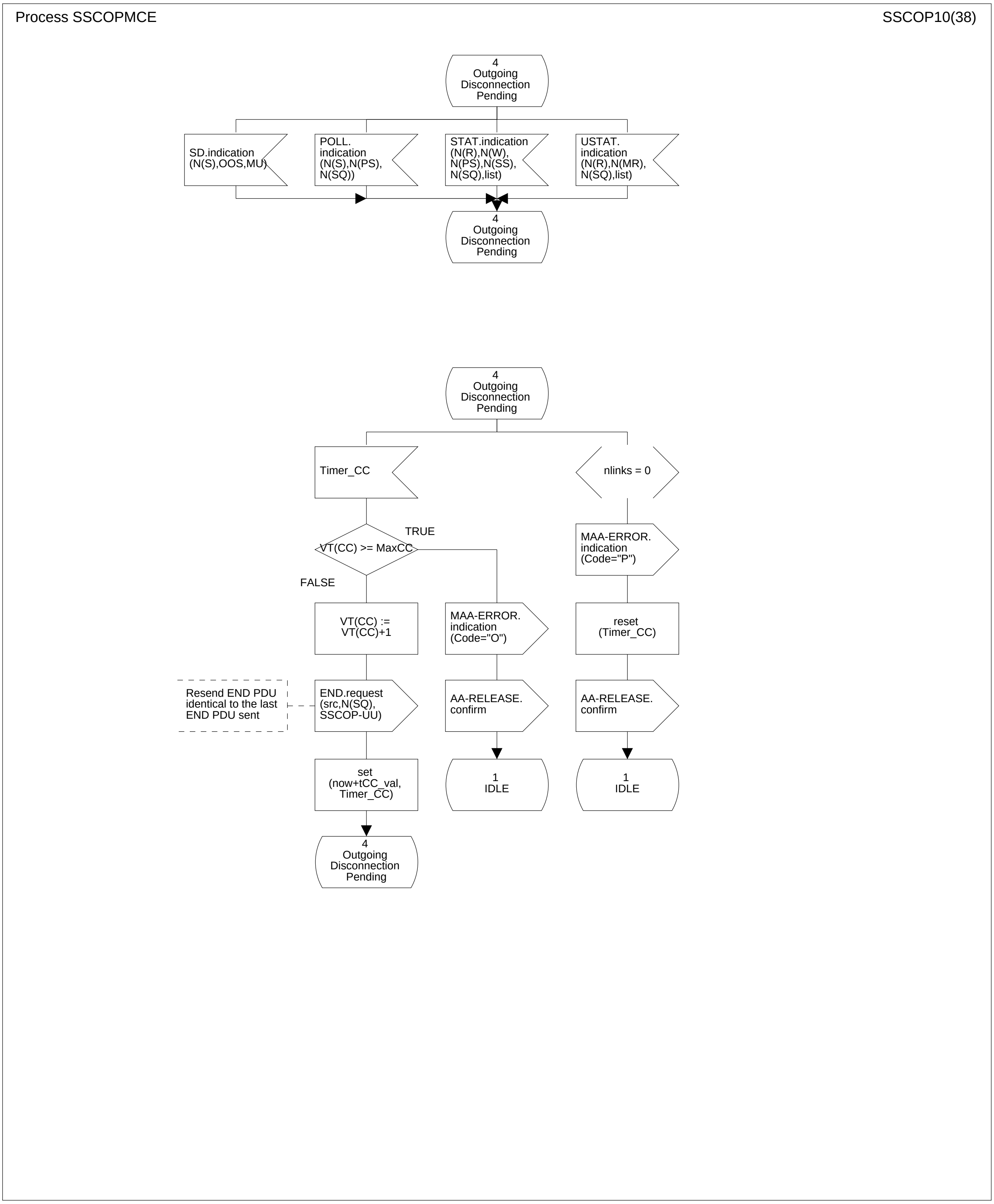


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 10 of 38)*

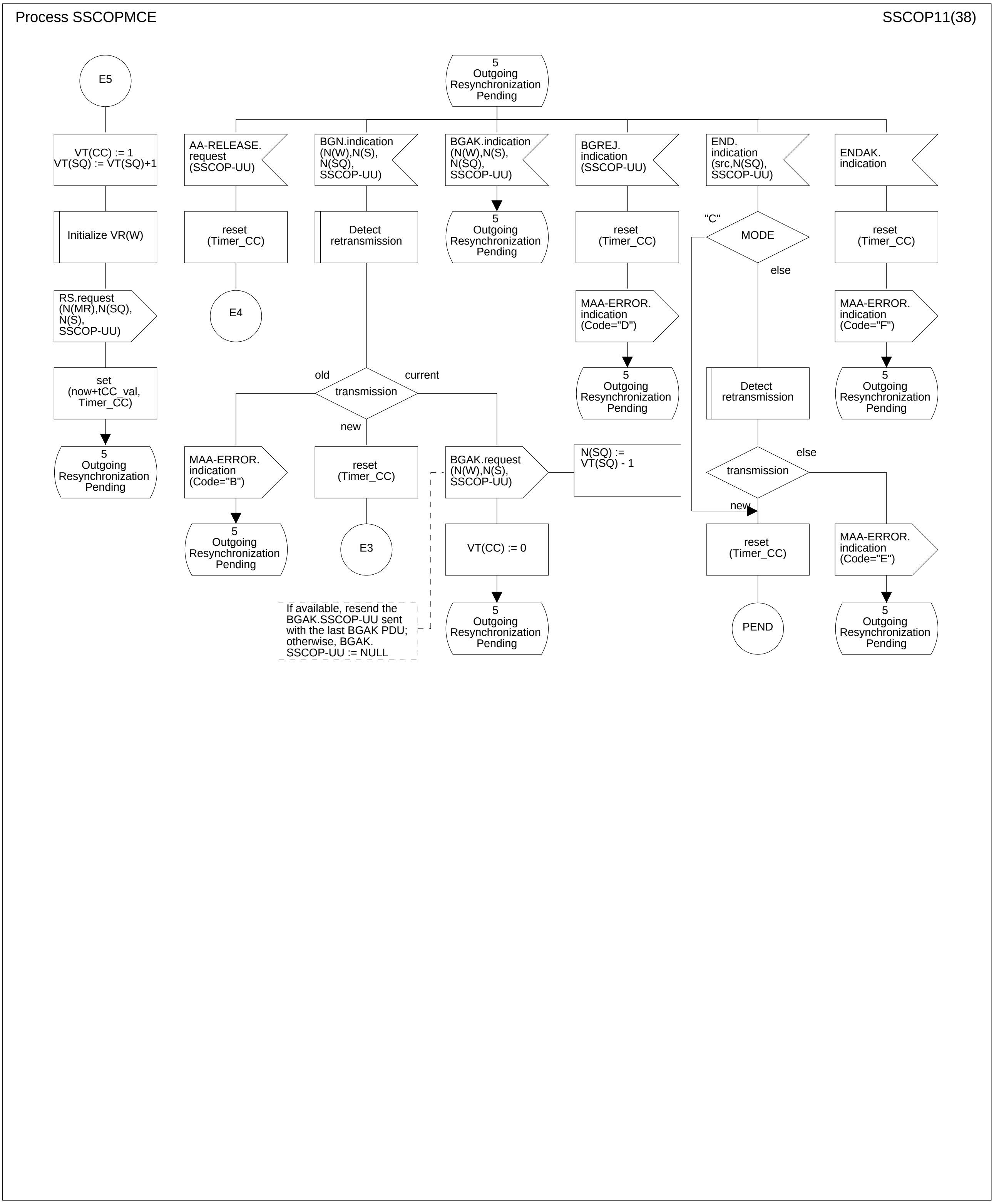


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 11 of 38)*

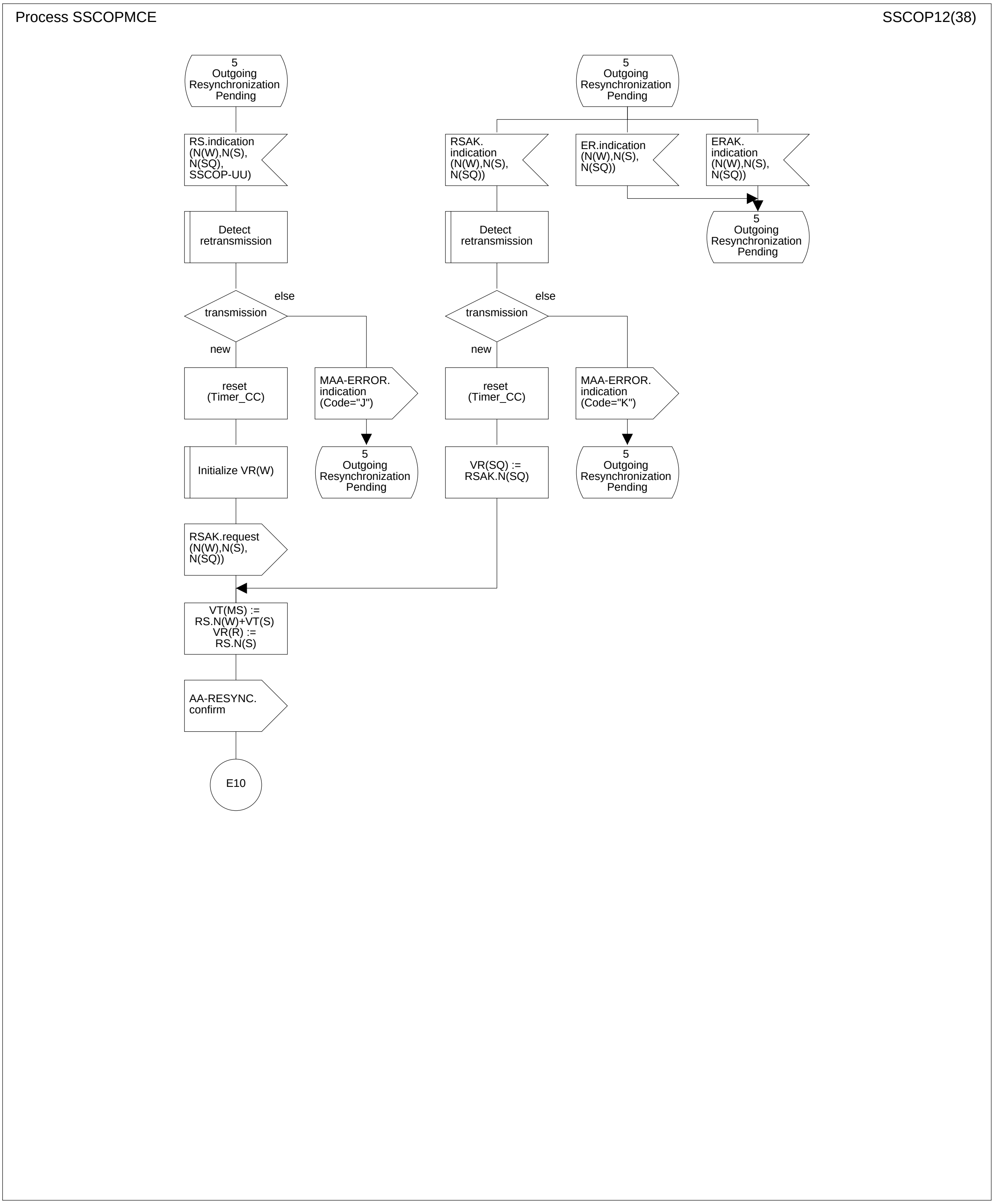


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 12 of 38)*

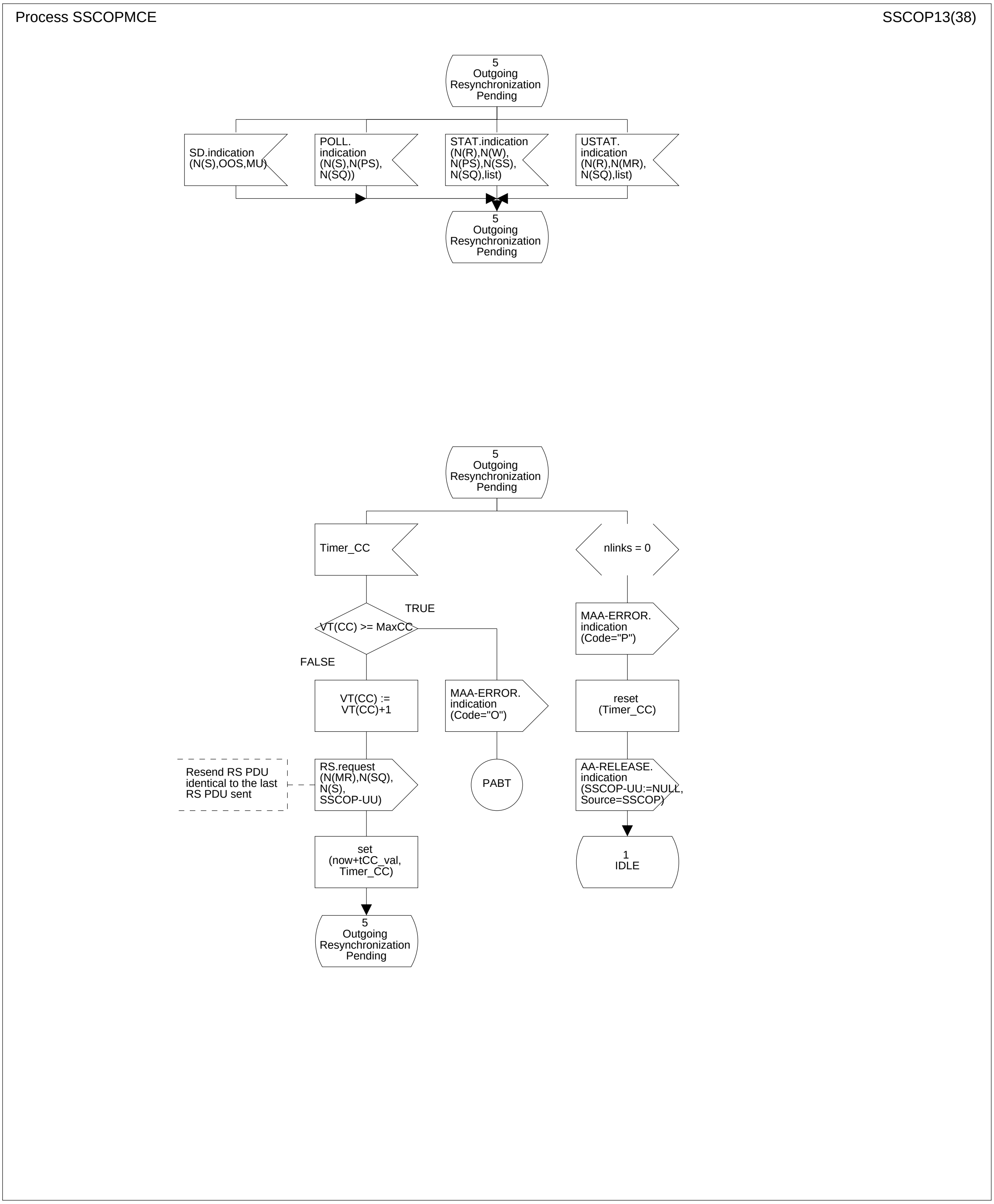


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 13 of 38)*

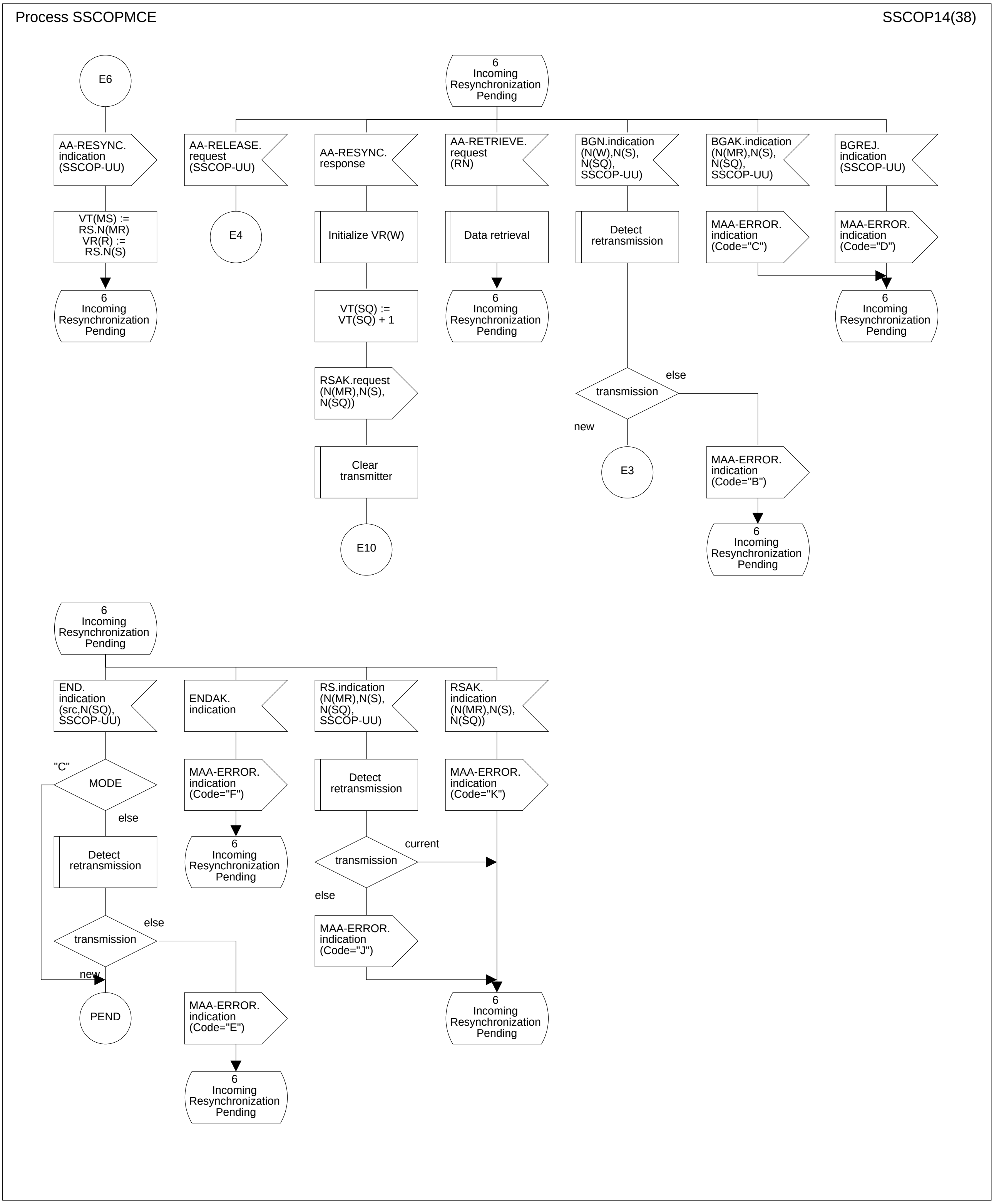


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 14 of 38)*

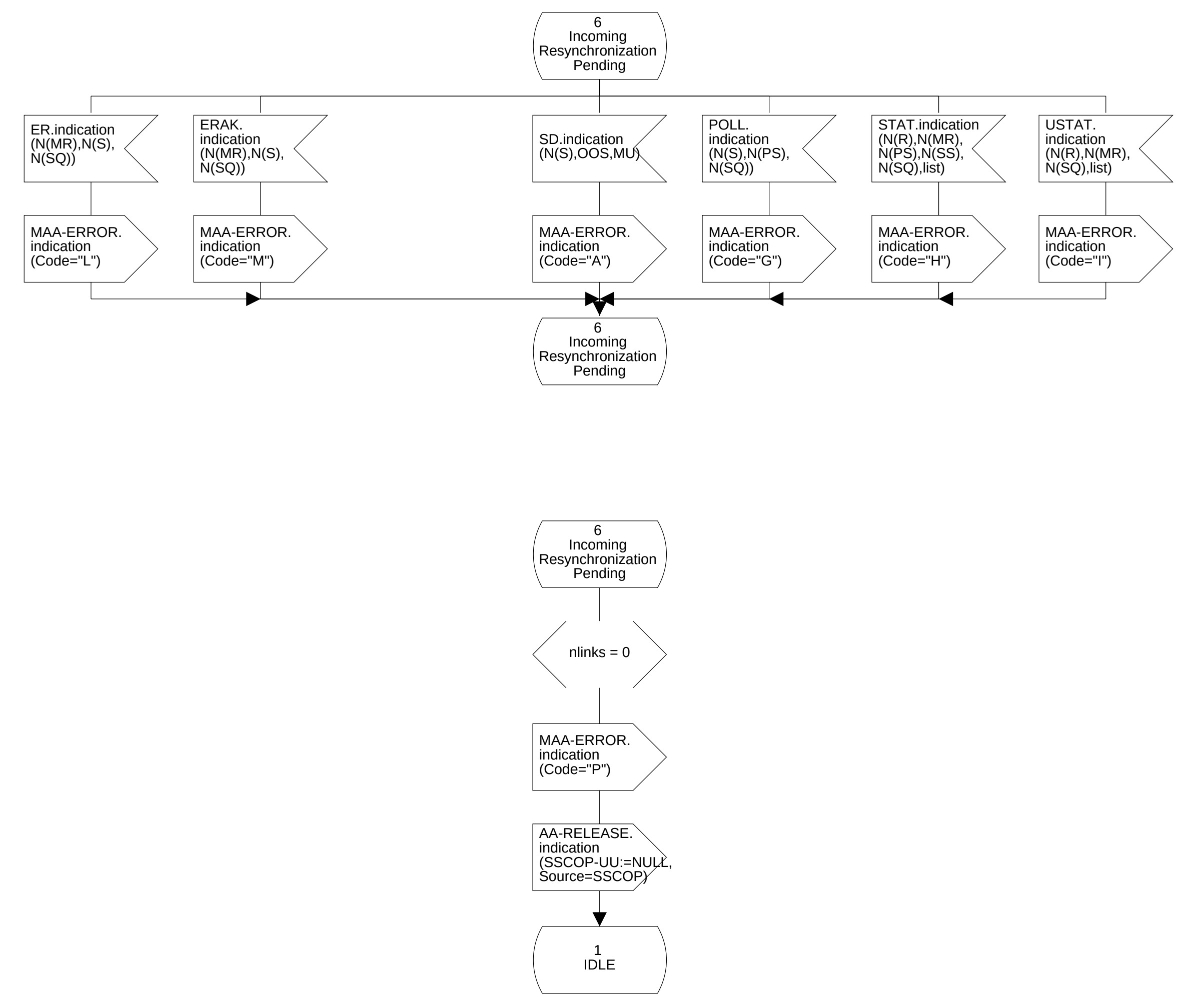


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity ***(sheet 15 of 38)***

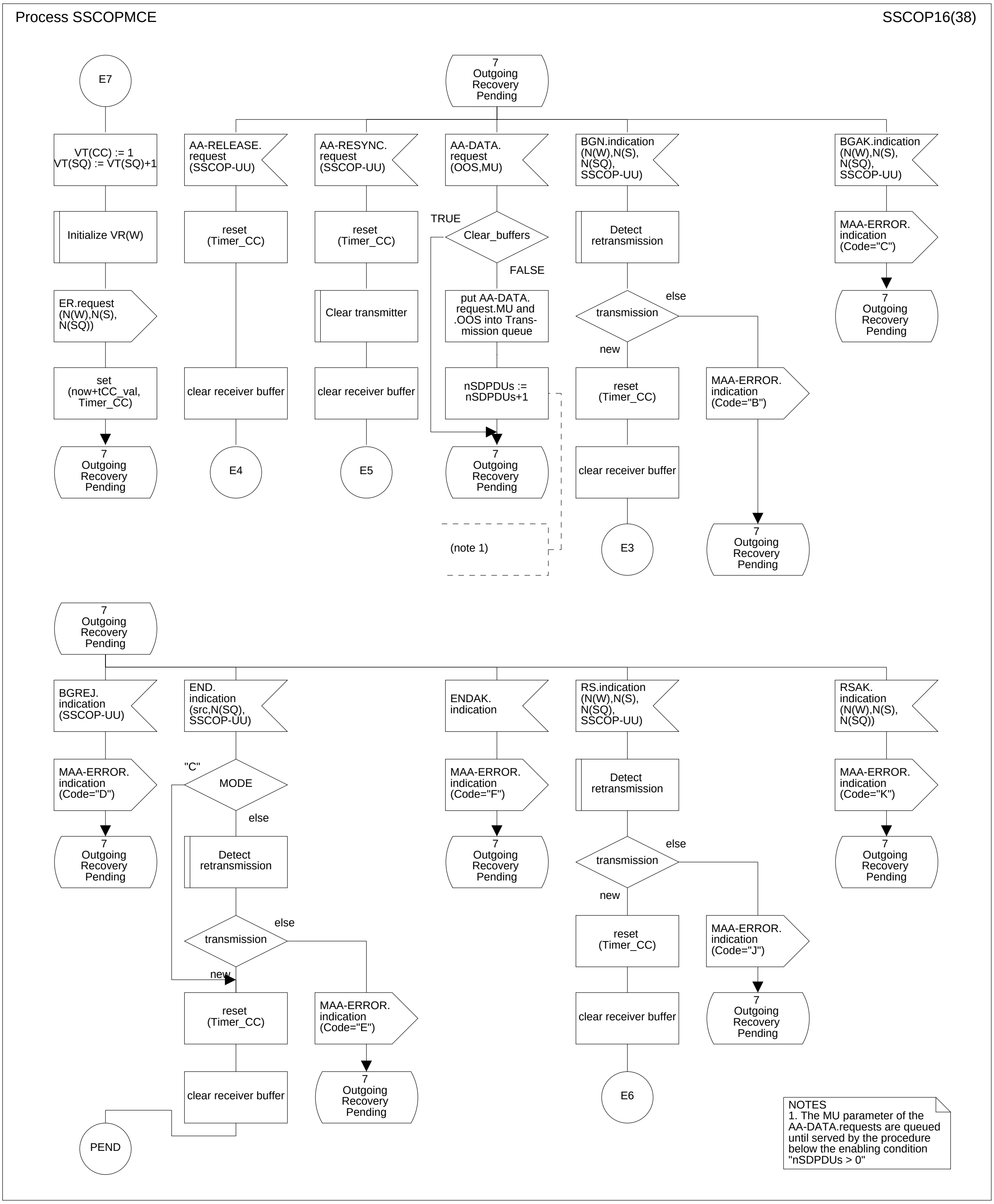


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 16 of 38)*

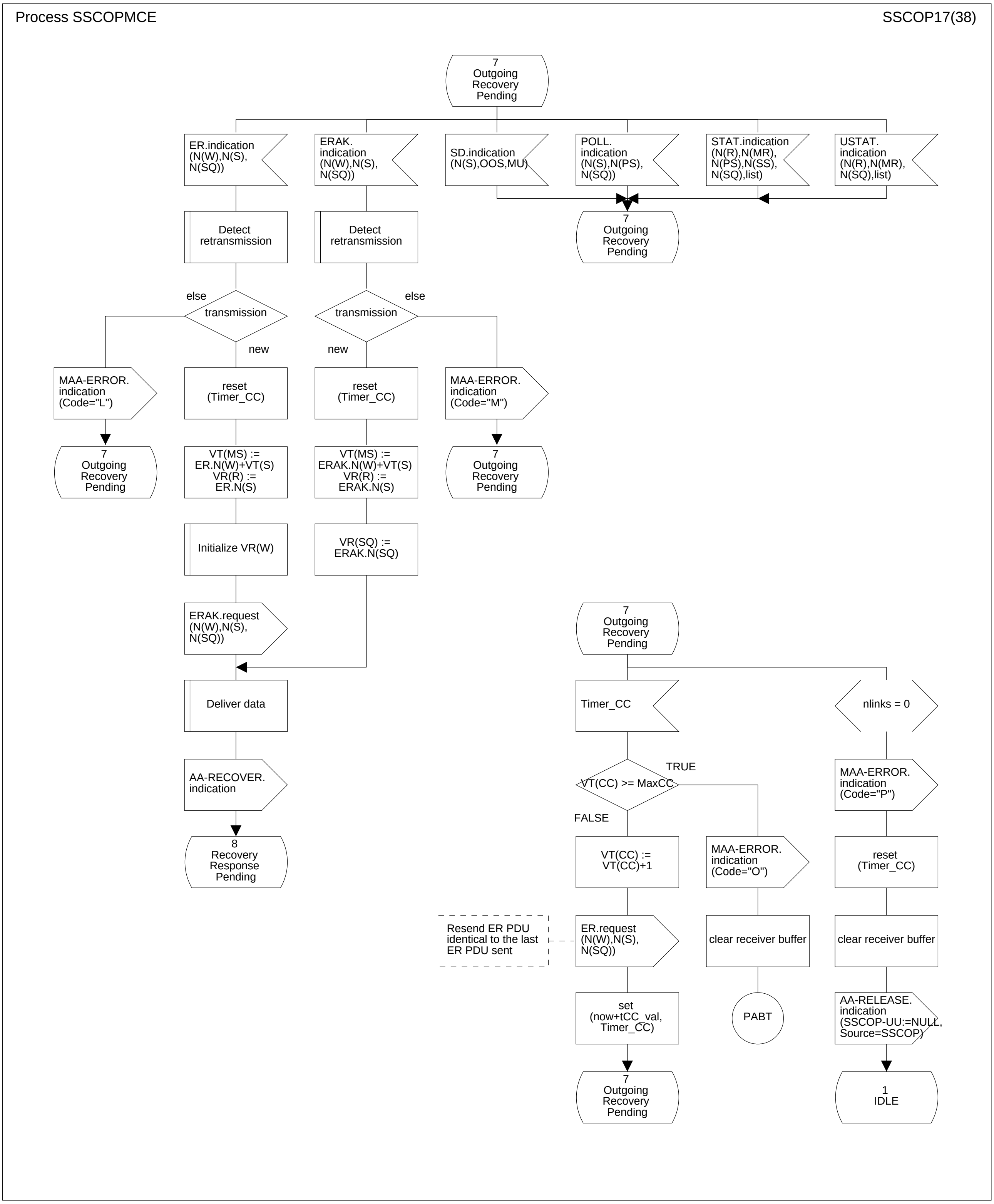


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 17 of 38)*

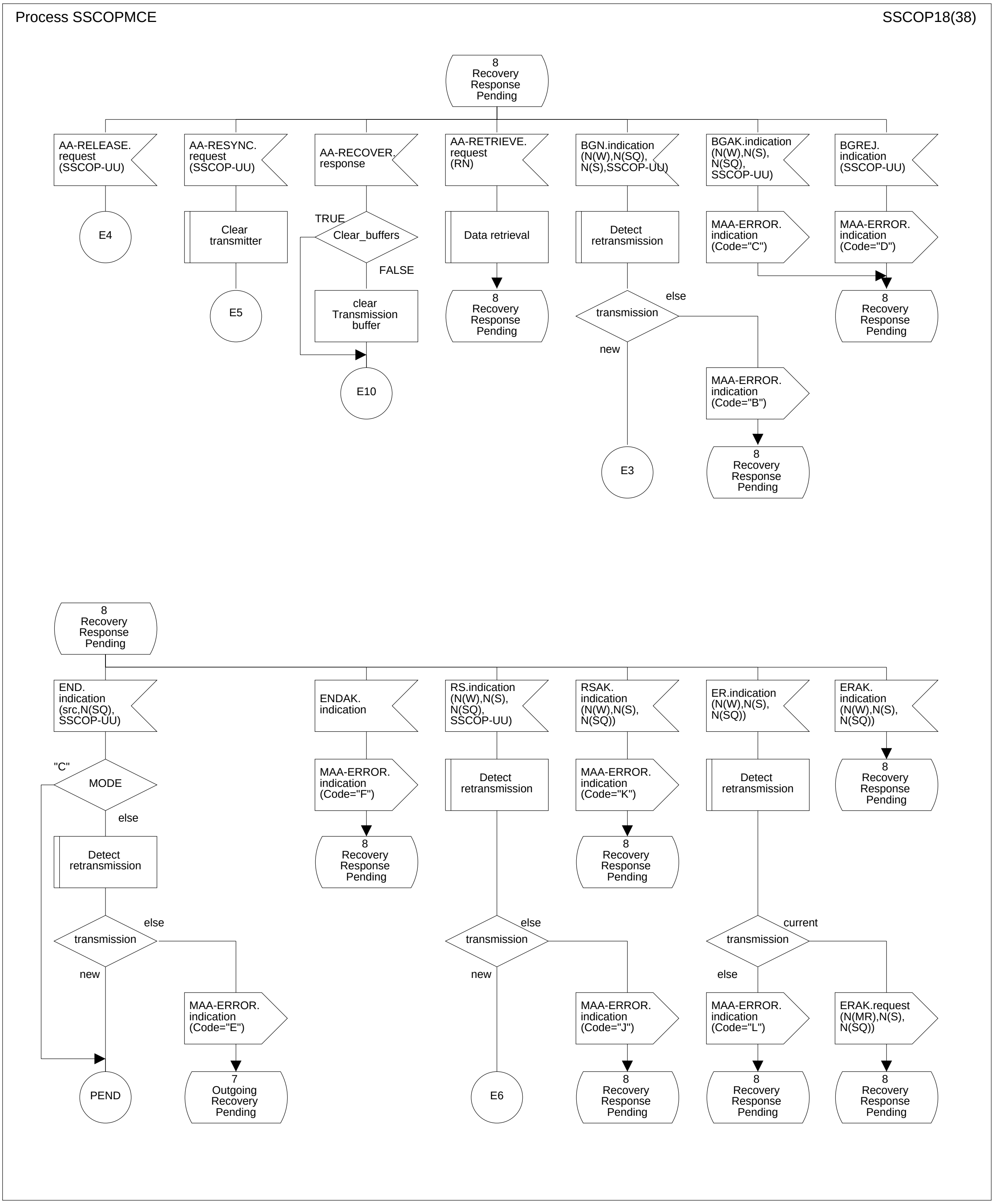


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 18 of 38)*

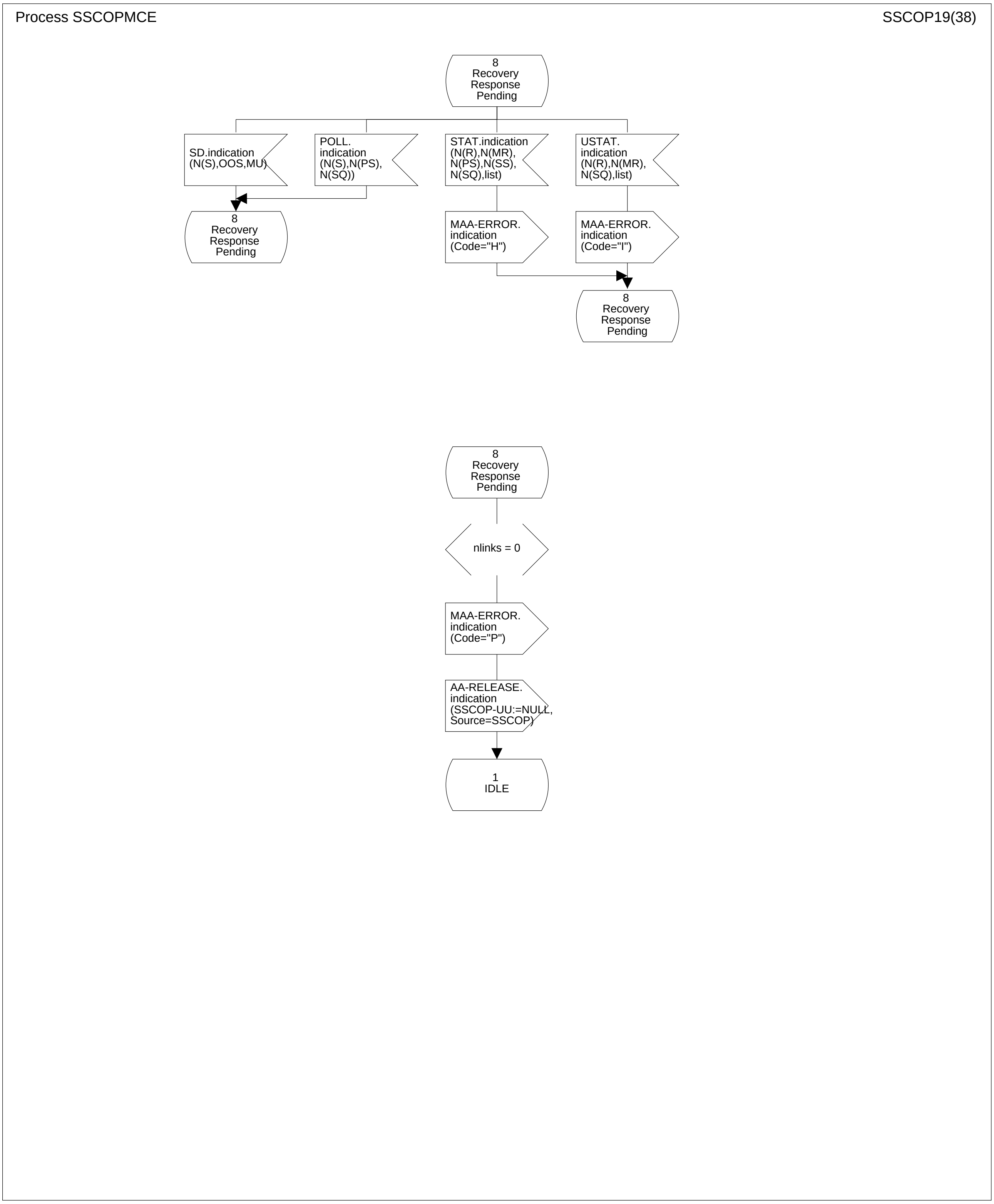


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 19 of 38)*

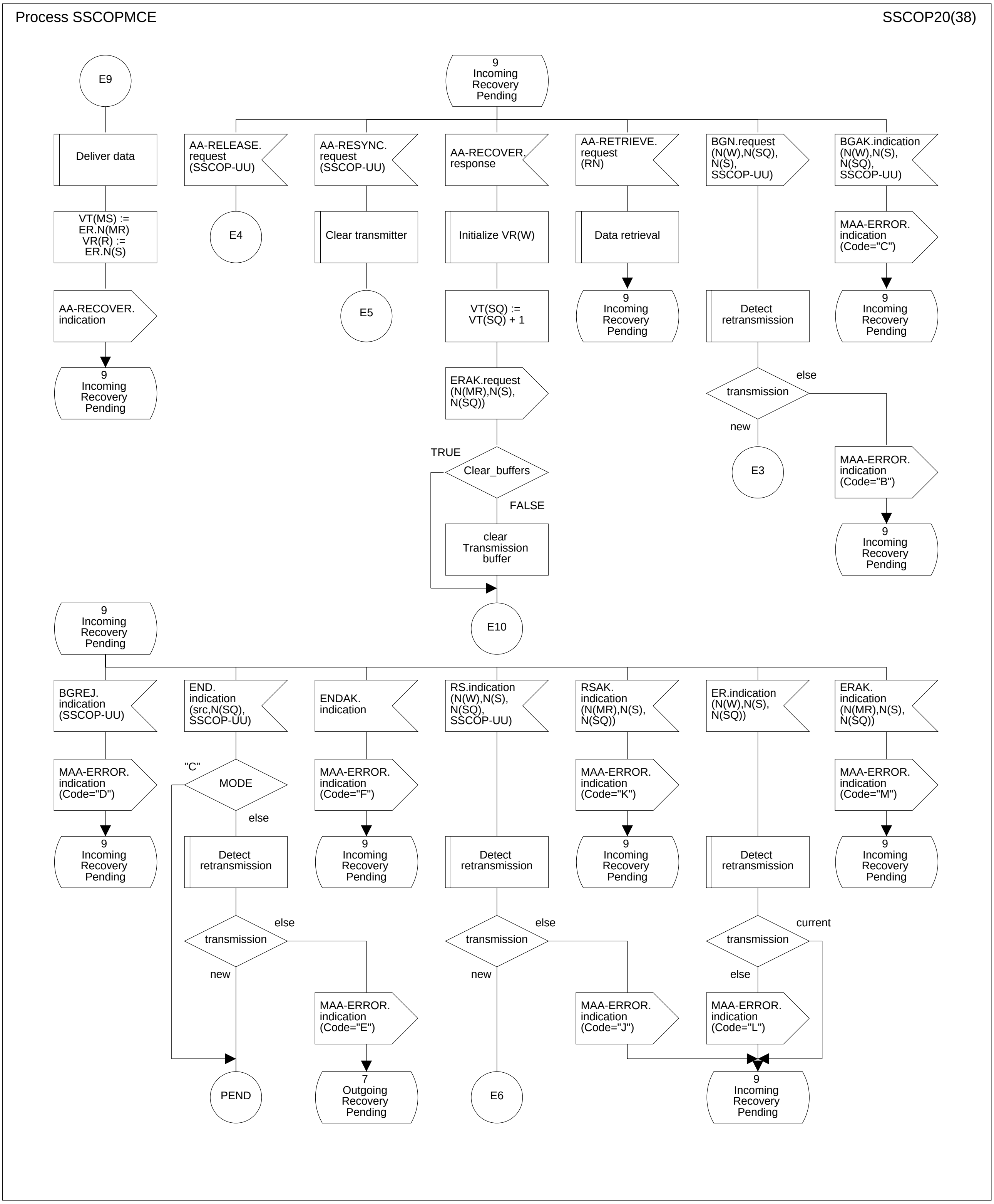


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 20 of 38)*

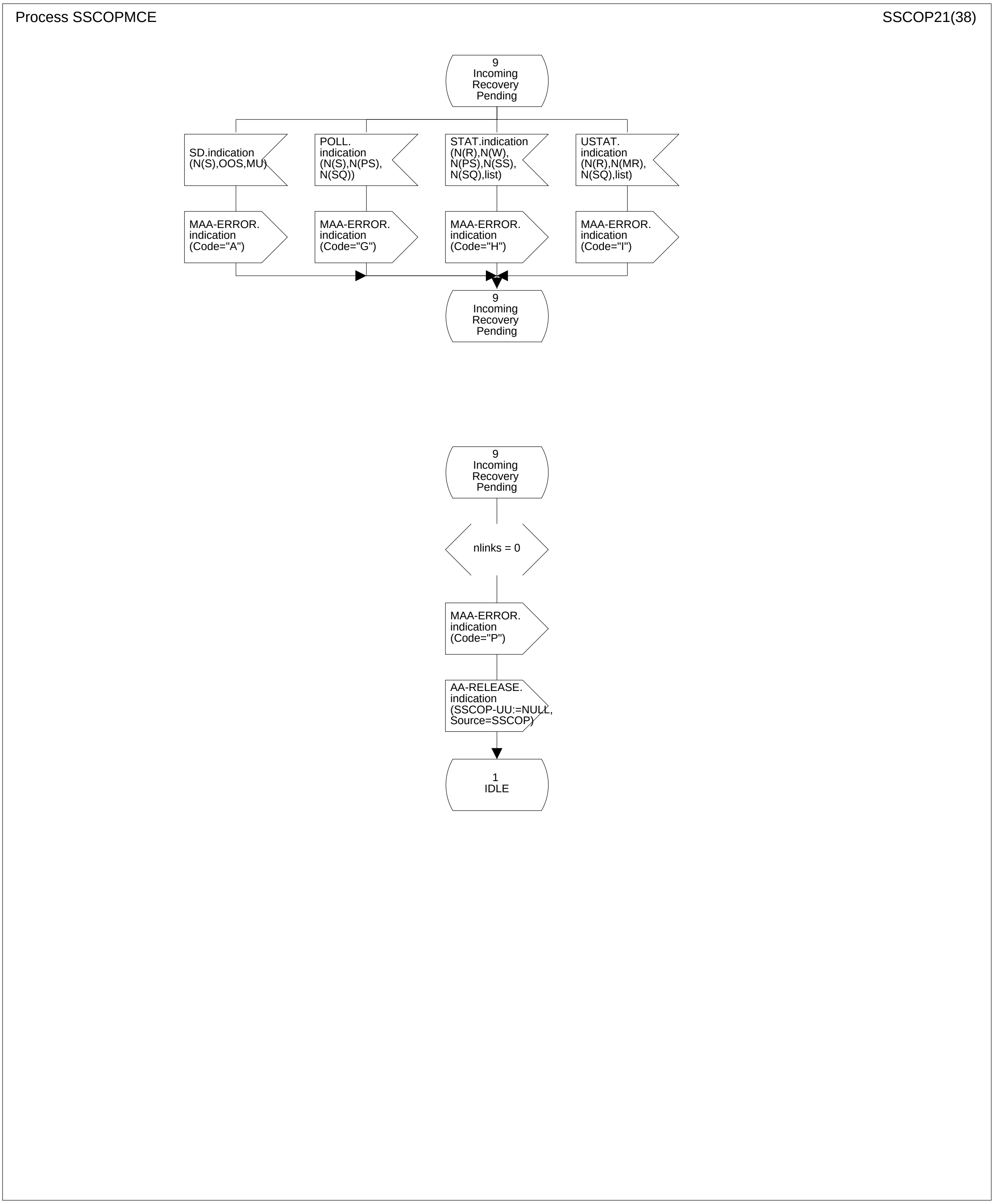


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 21 of 38)*

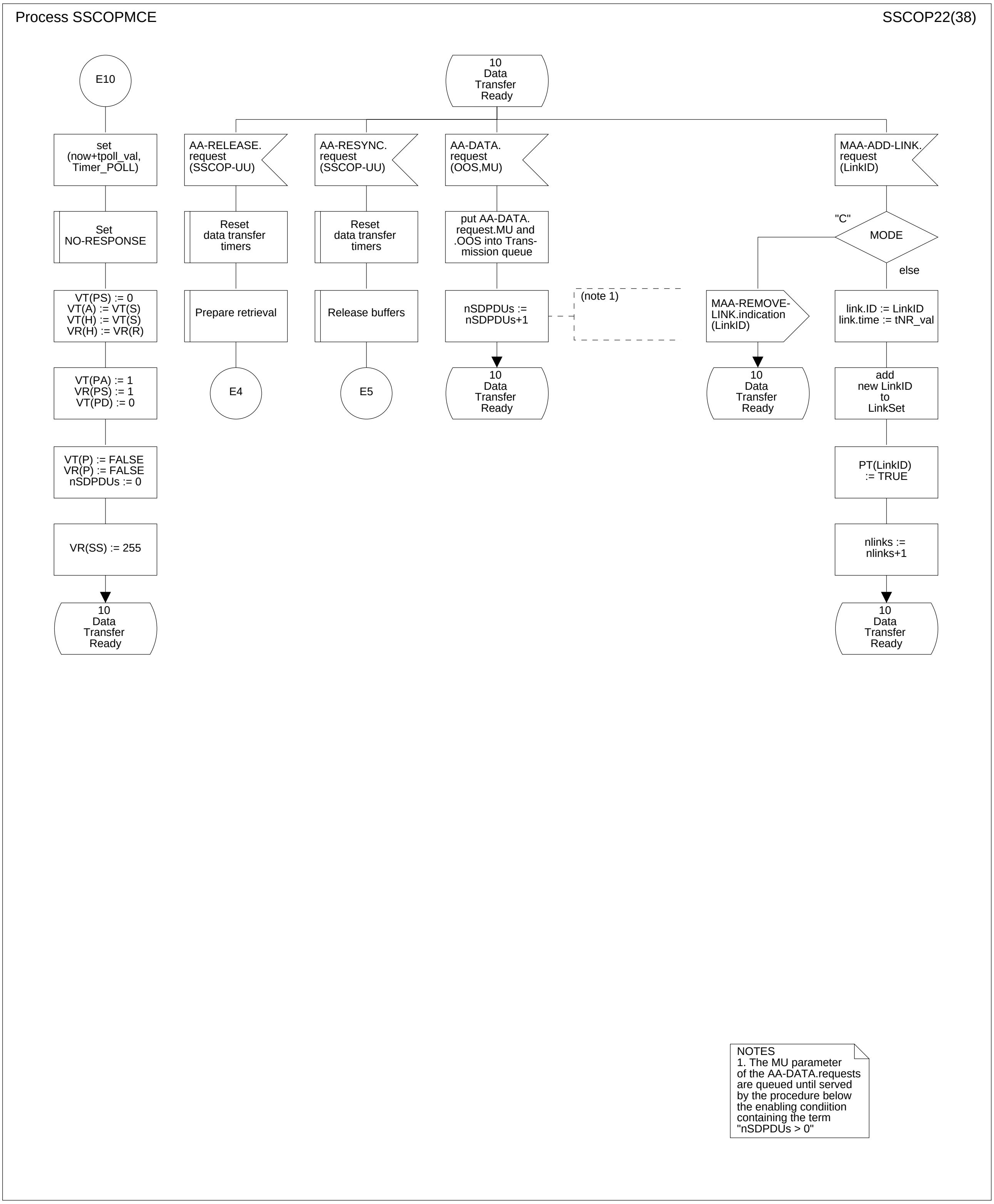


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 22 of 38)*

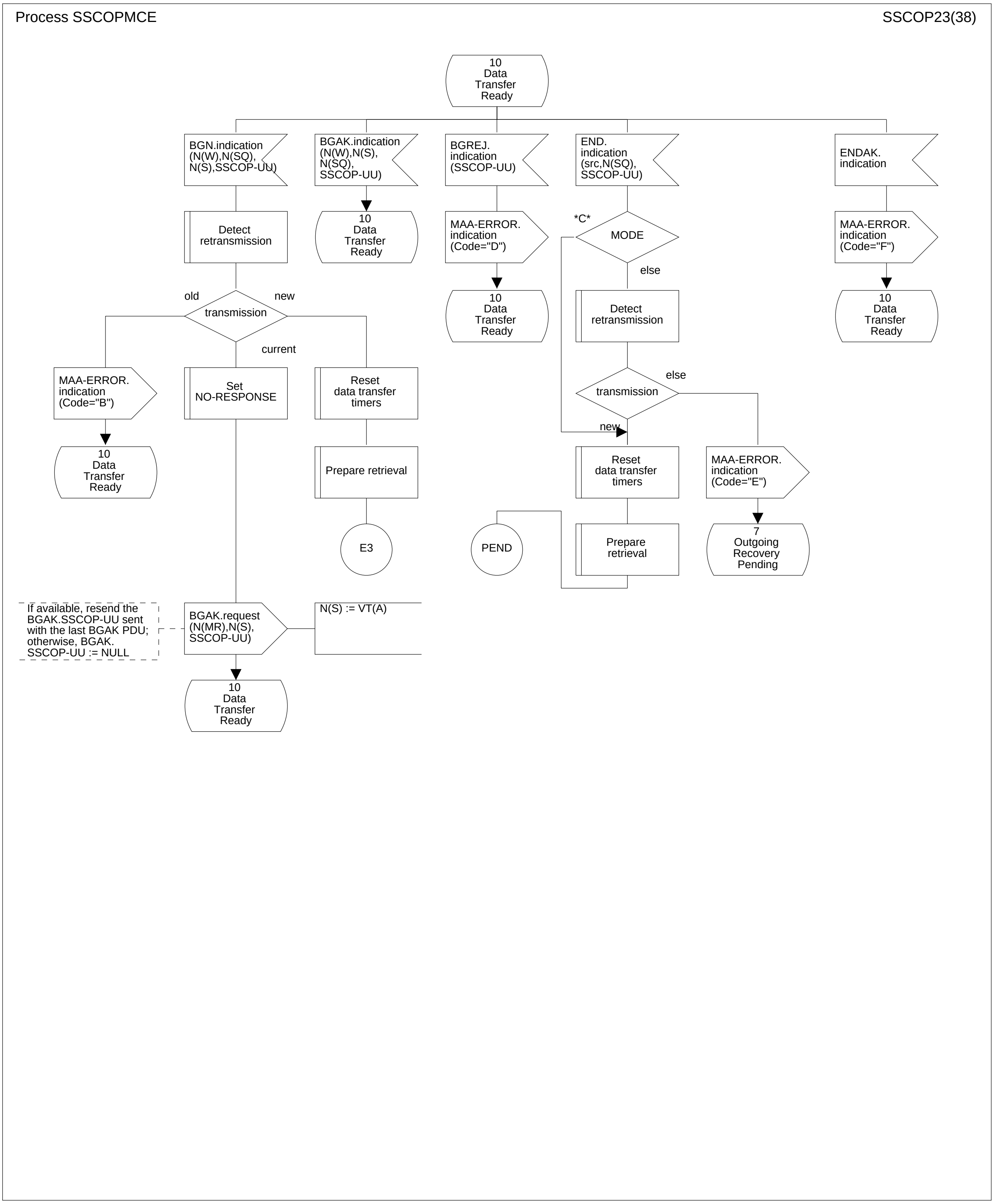


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 23 of 38)*

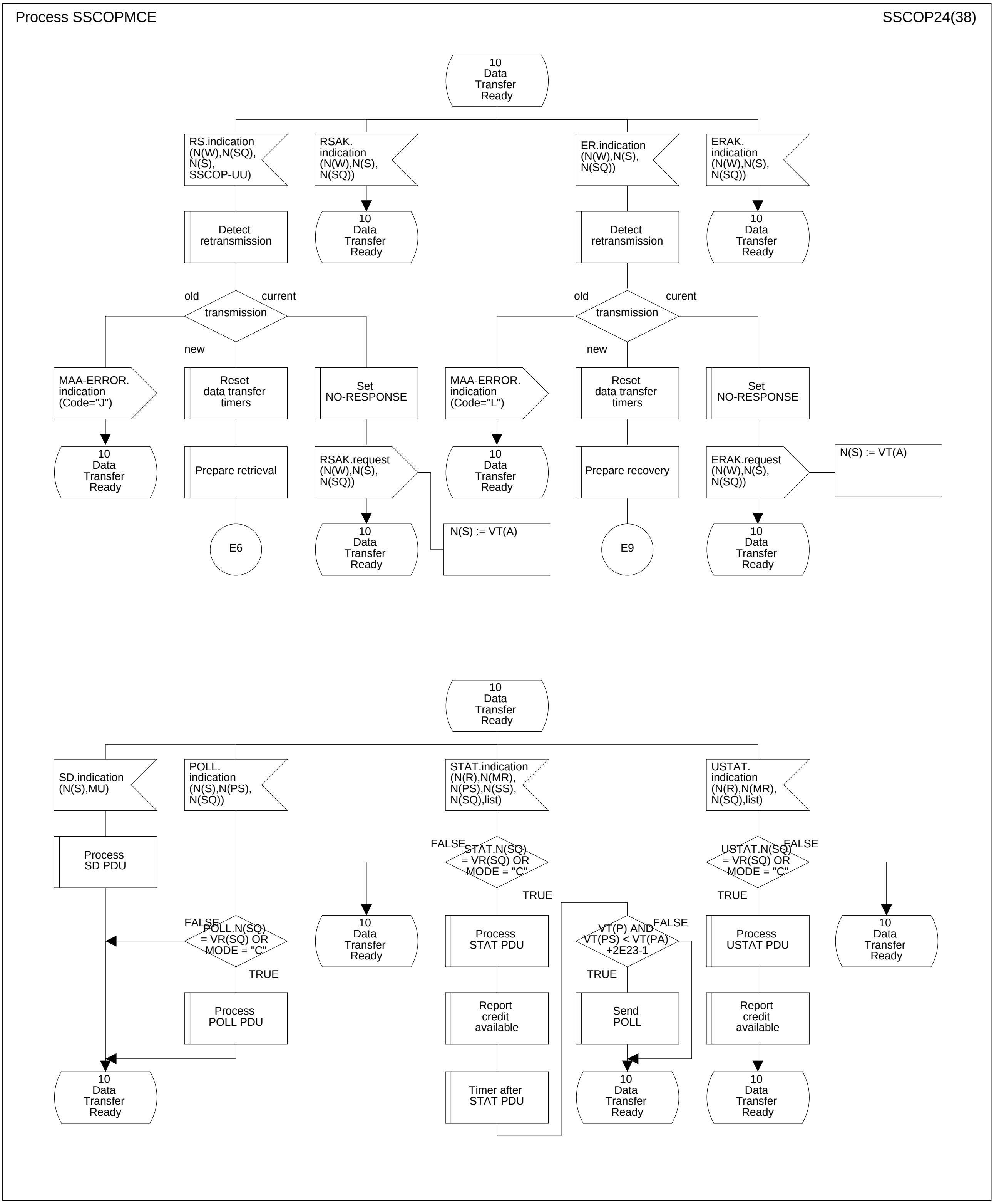


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 24 of 38)*

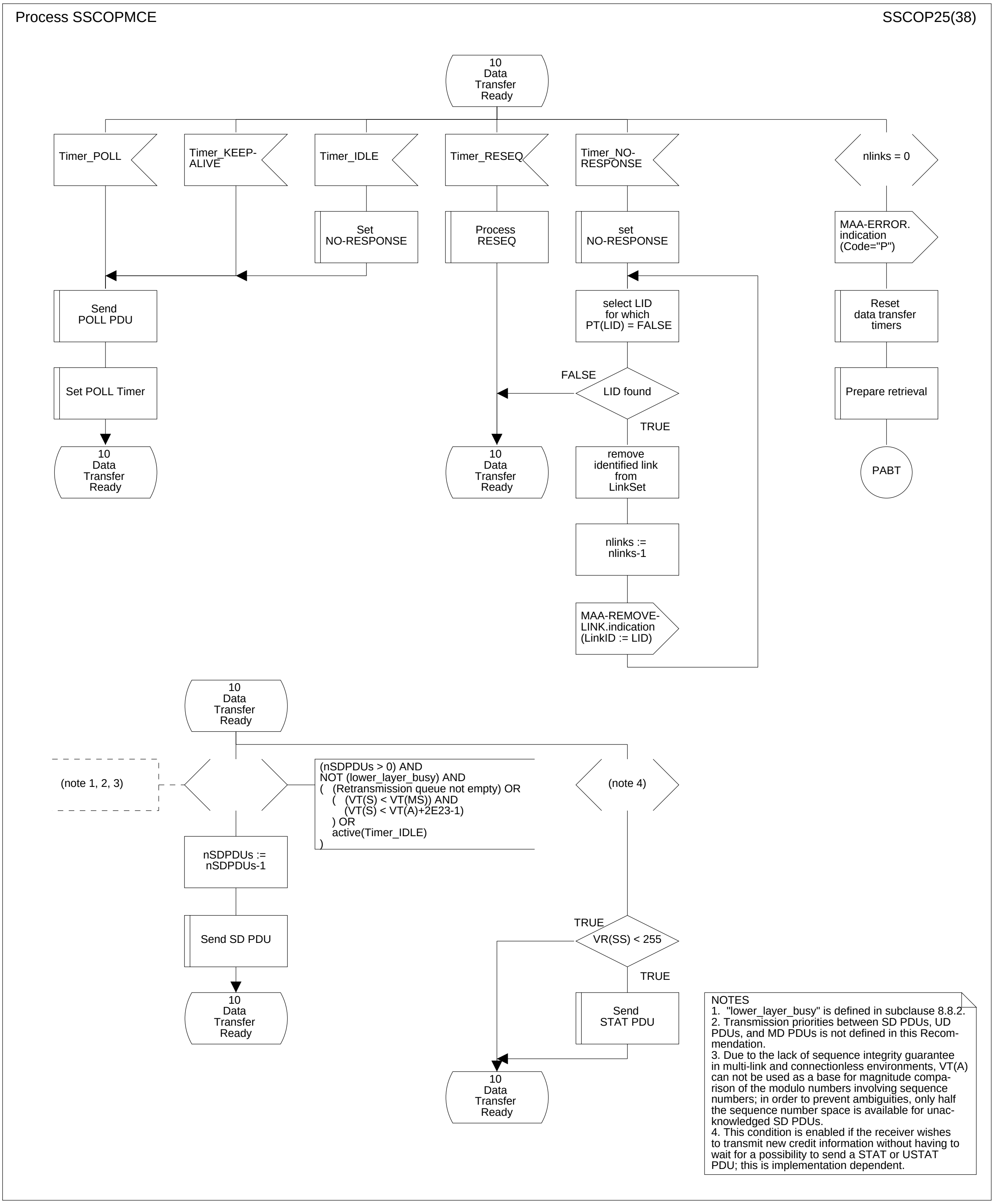


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 25 of 38)*

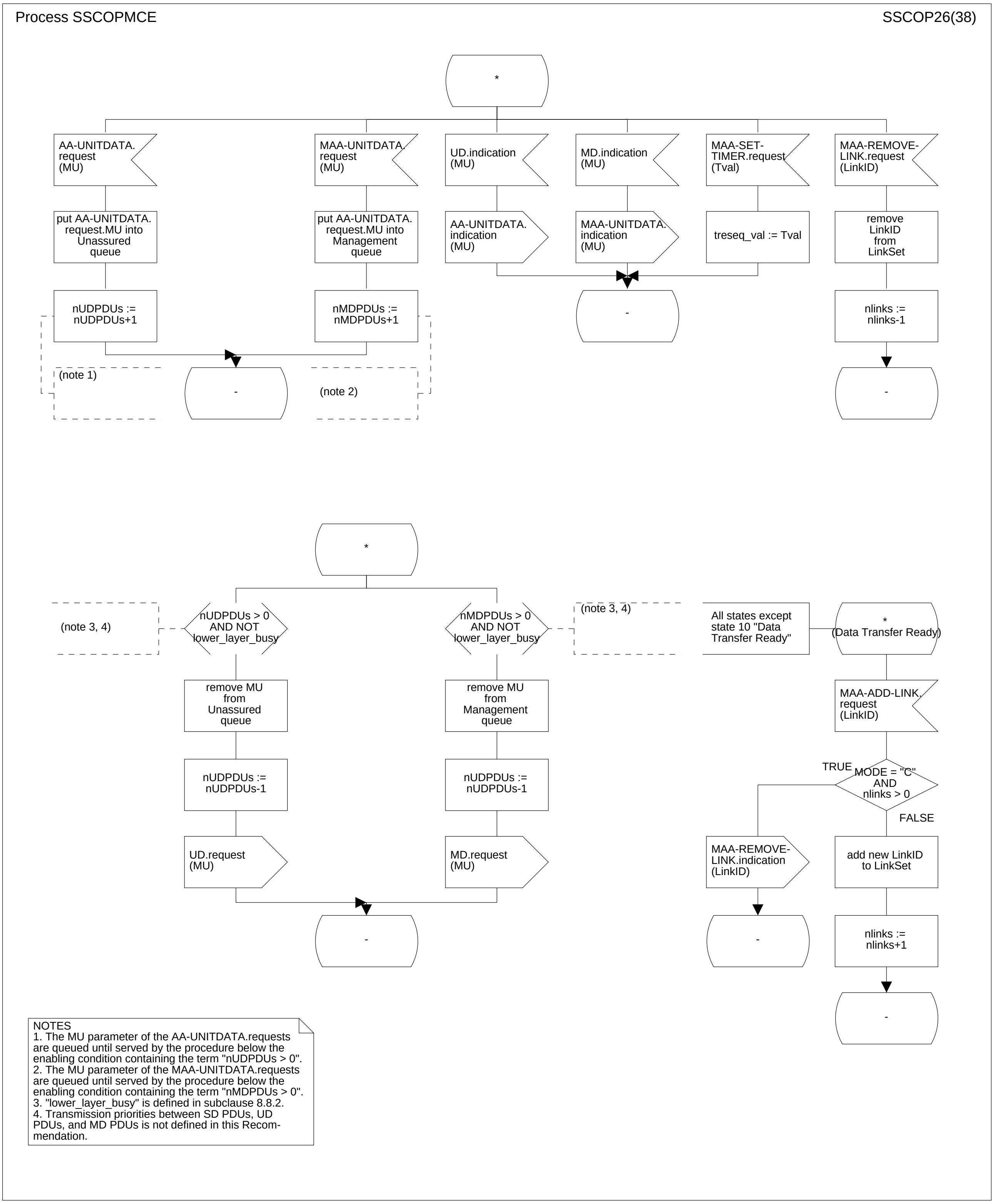


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 26 of 38)*

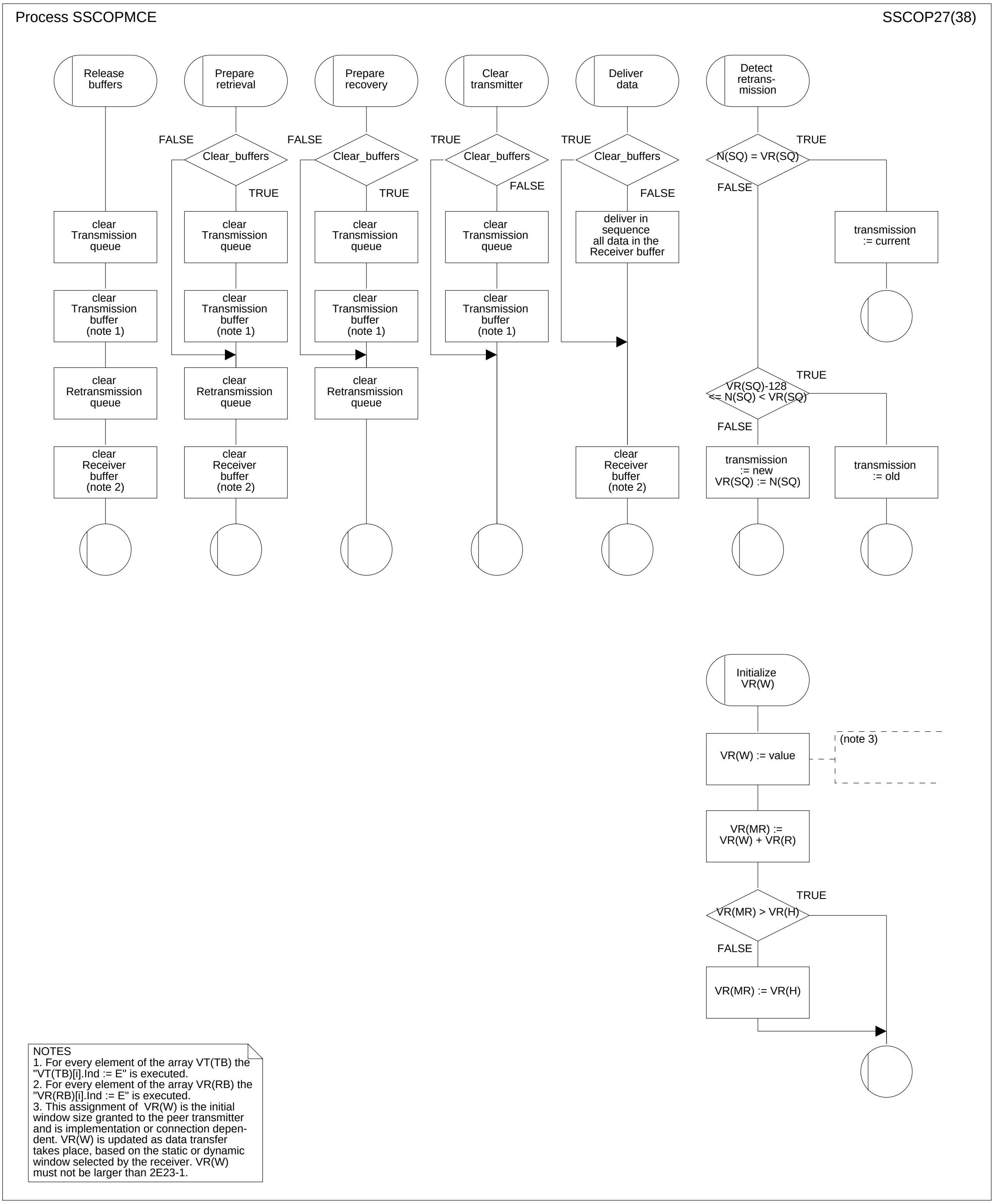


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 27 of 38)*

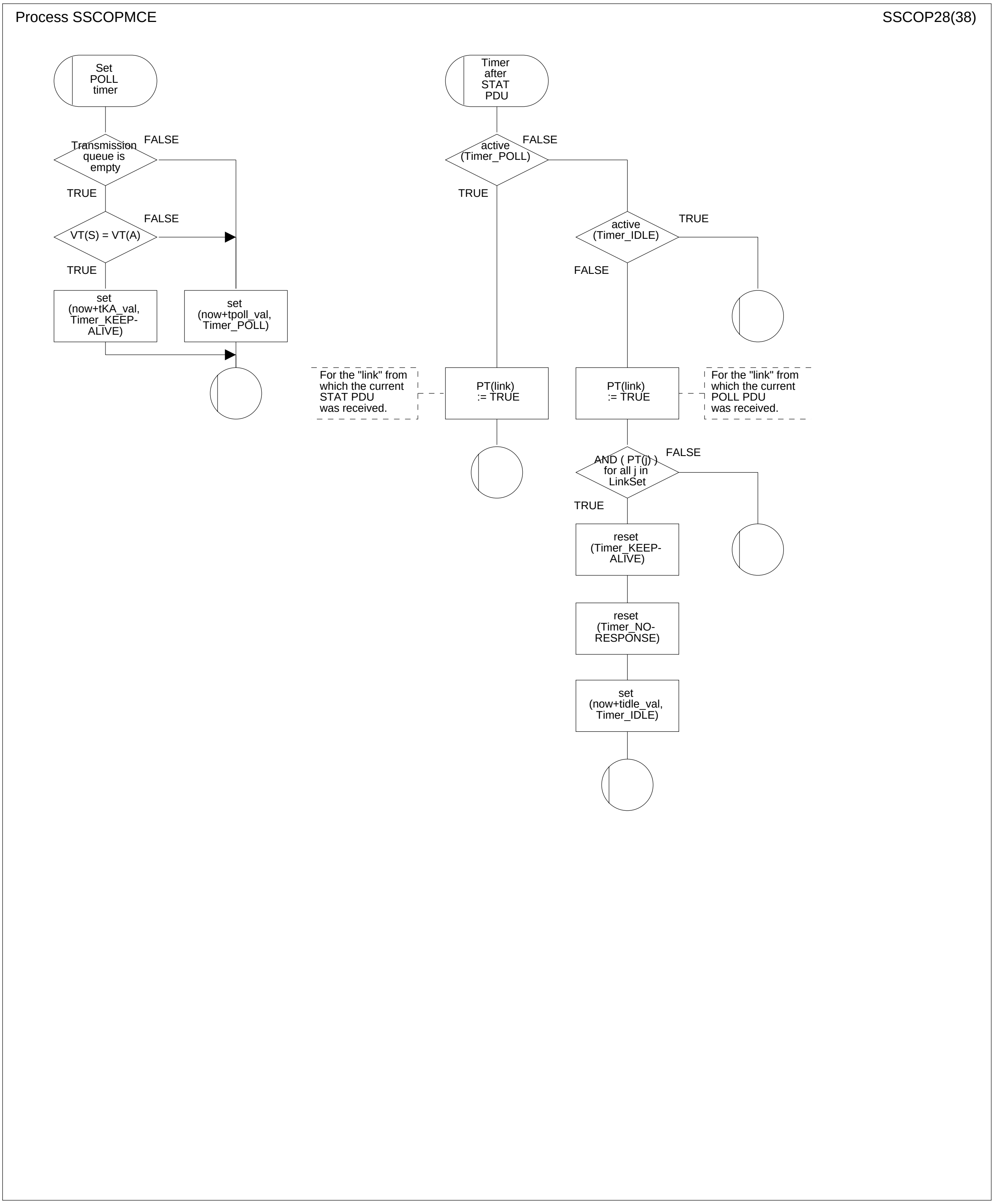


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 28 of 38)*

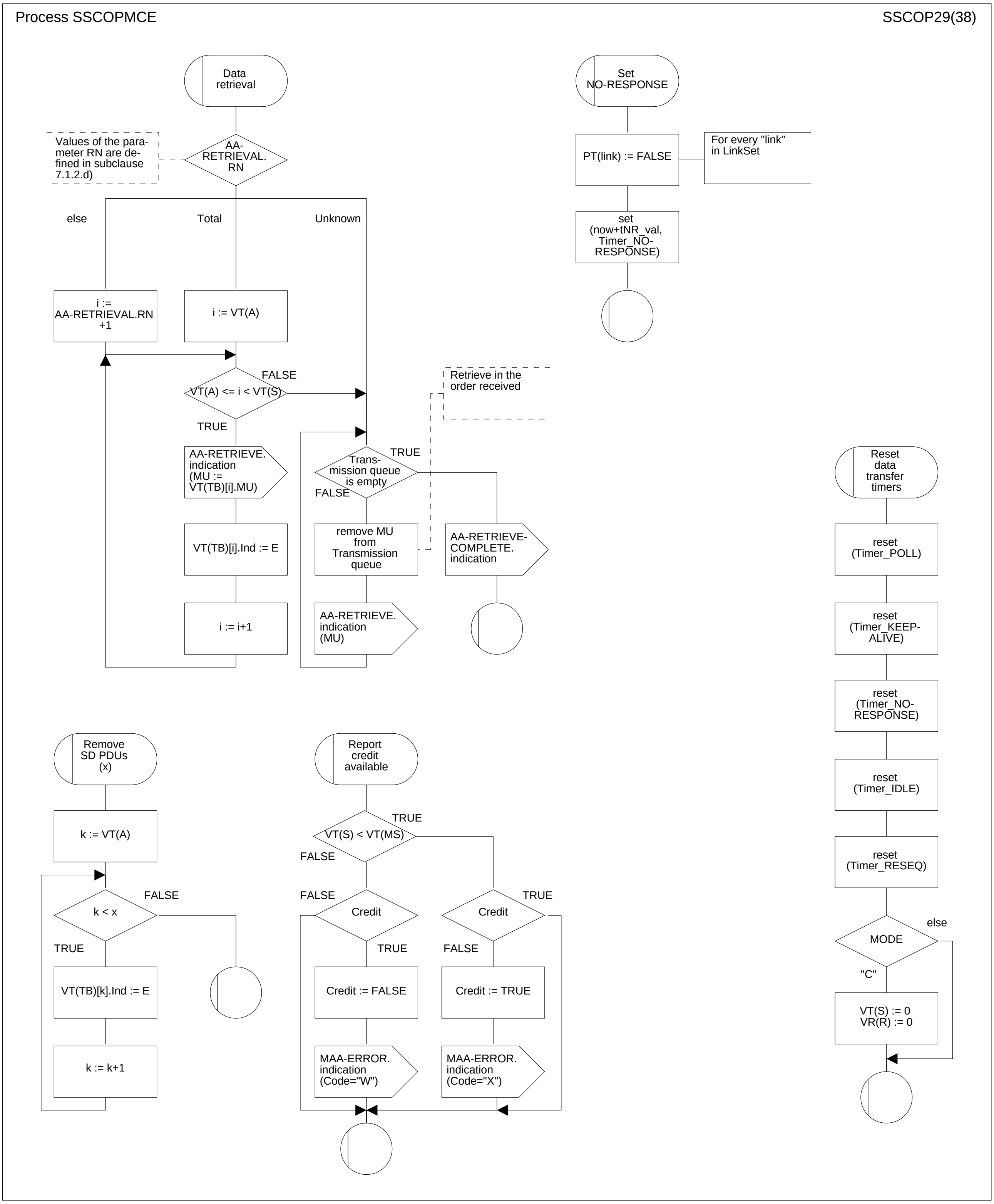


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity ***(sheet 29 of 38)***

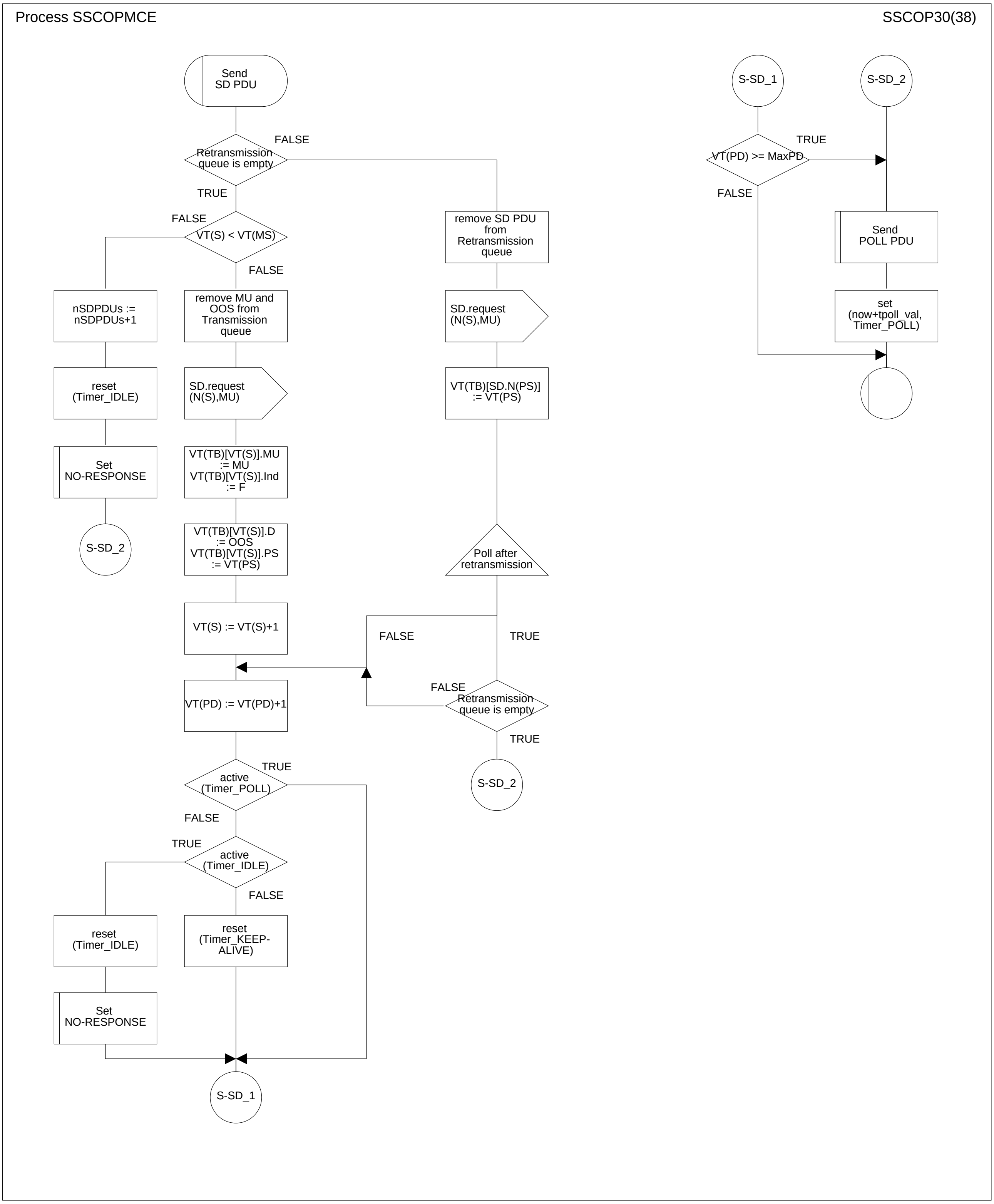


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity ***(sheet 30 of 38)***

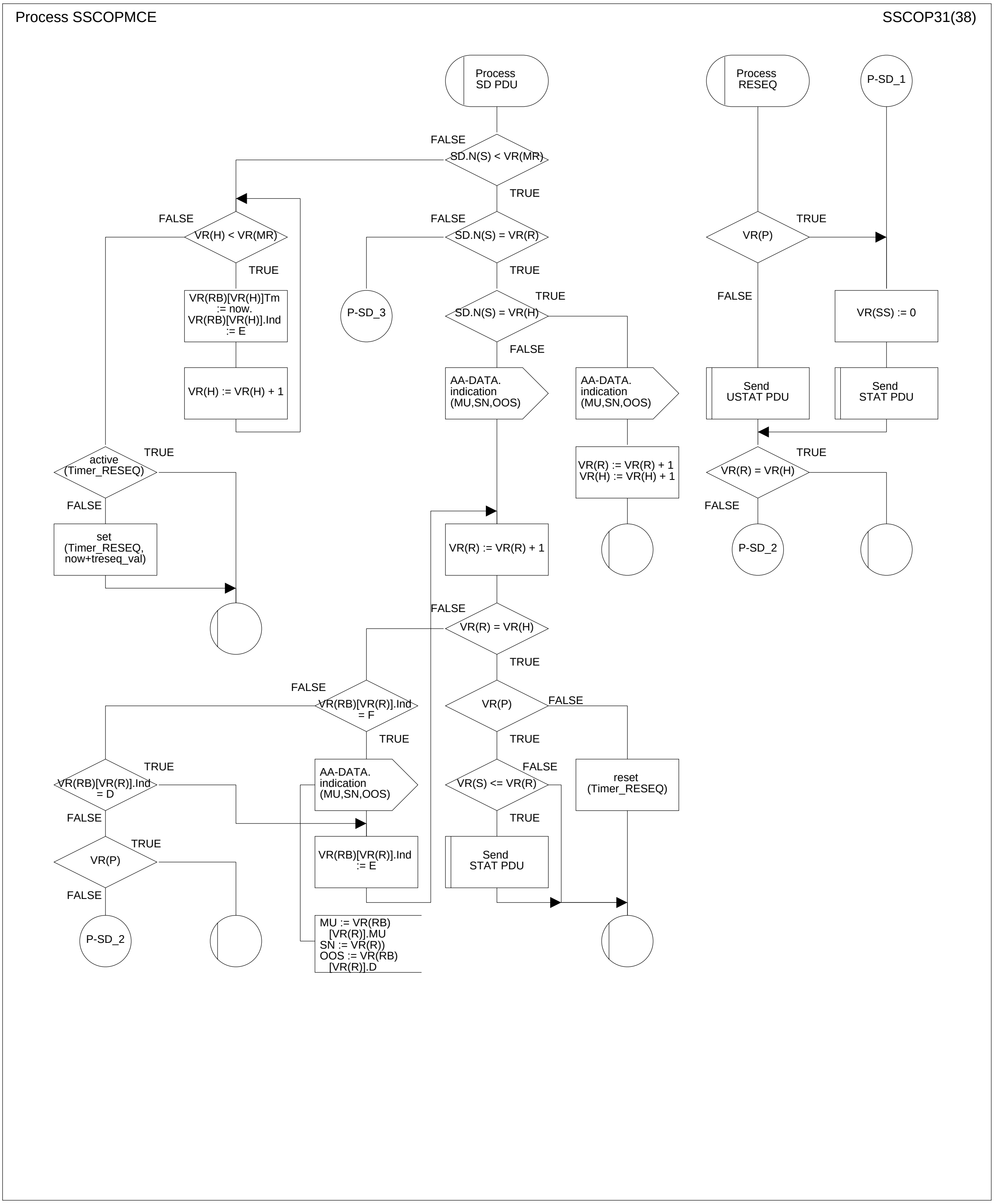


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 31 of 38)*

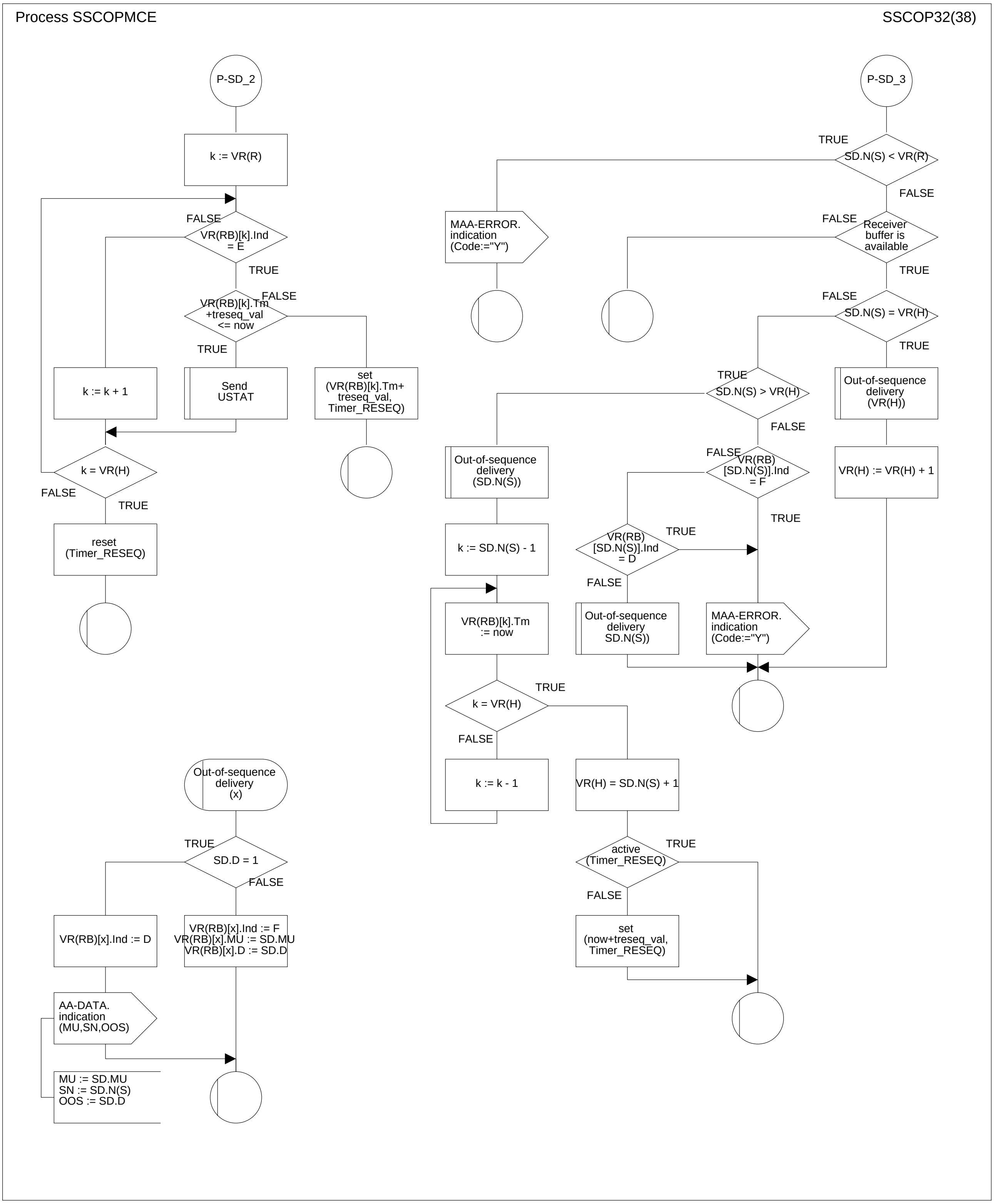


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 32 of 38)*

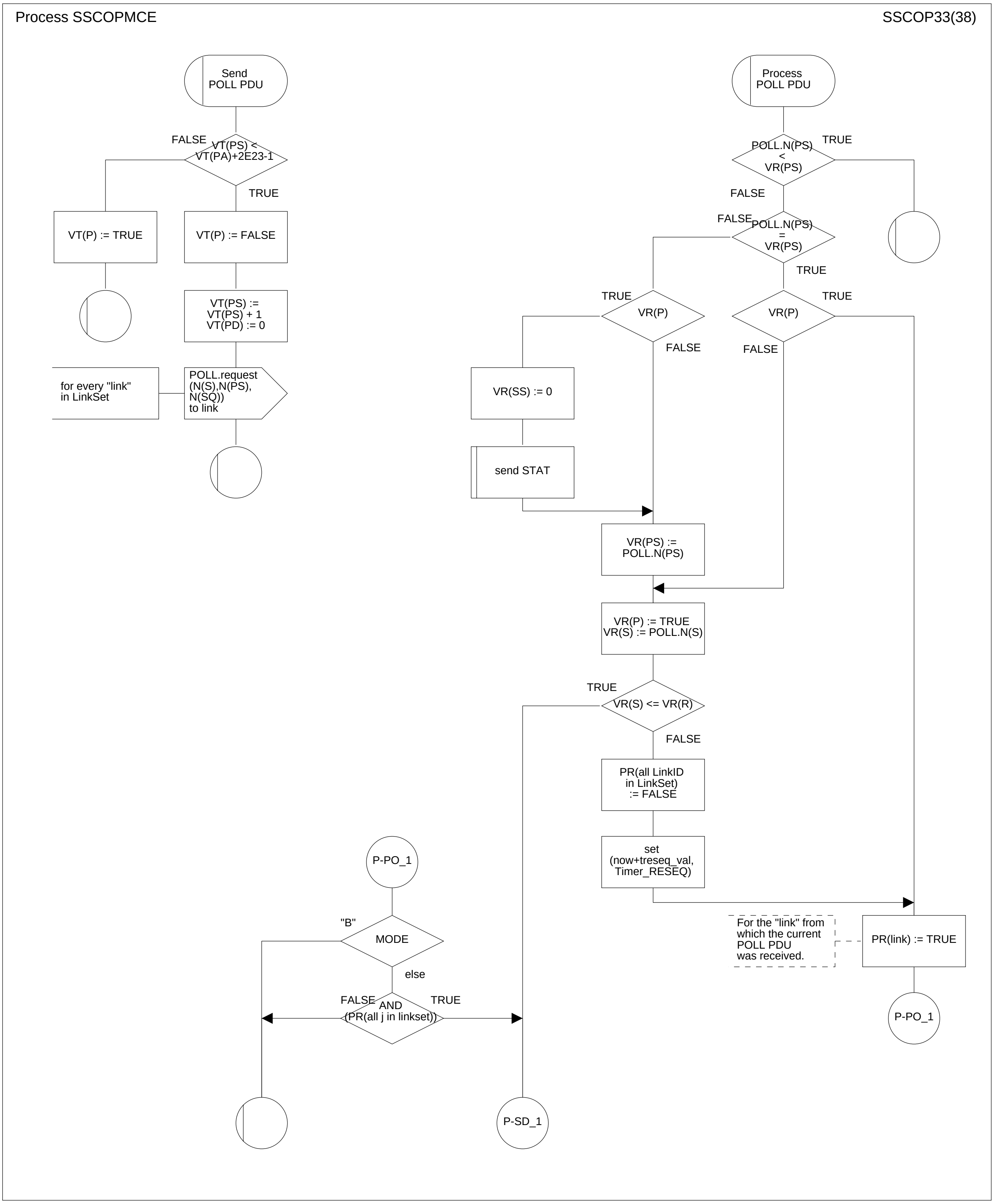


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 33 of 38)*

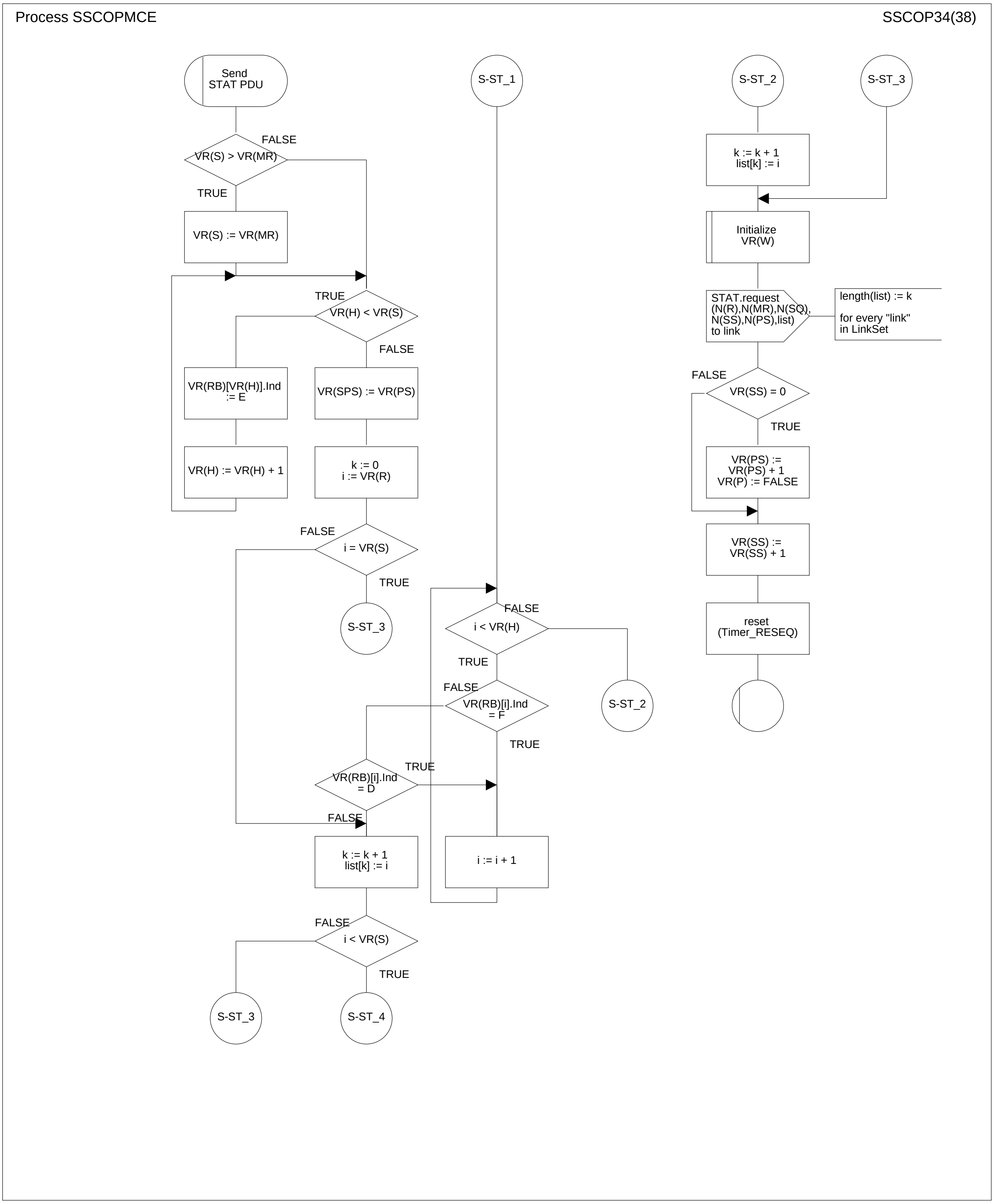


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 34 of 38)*

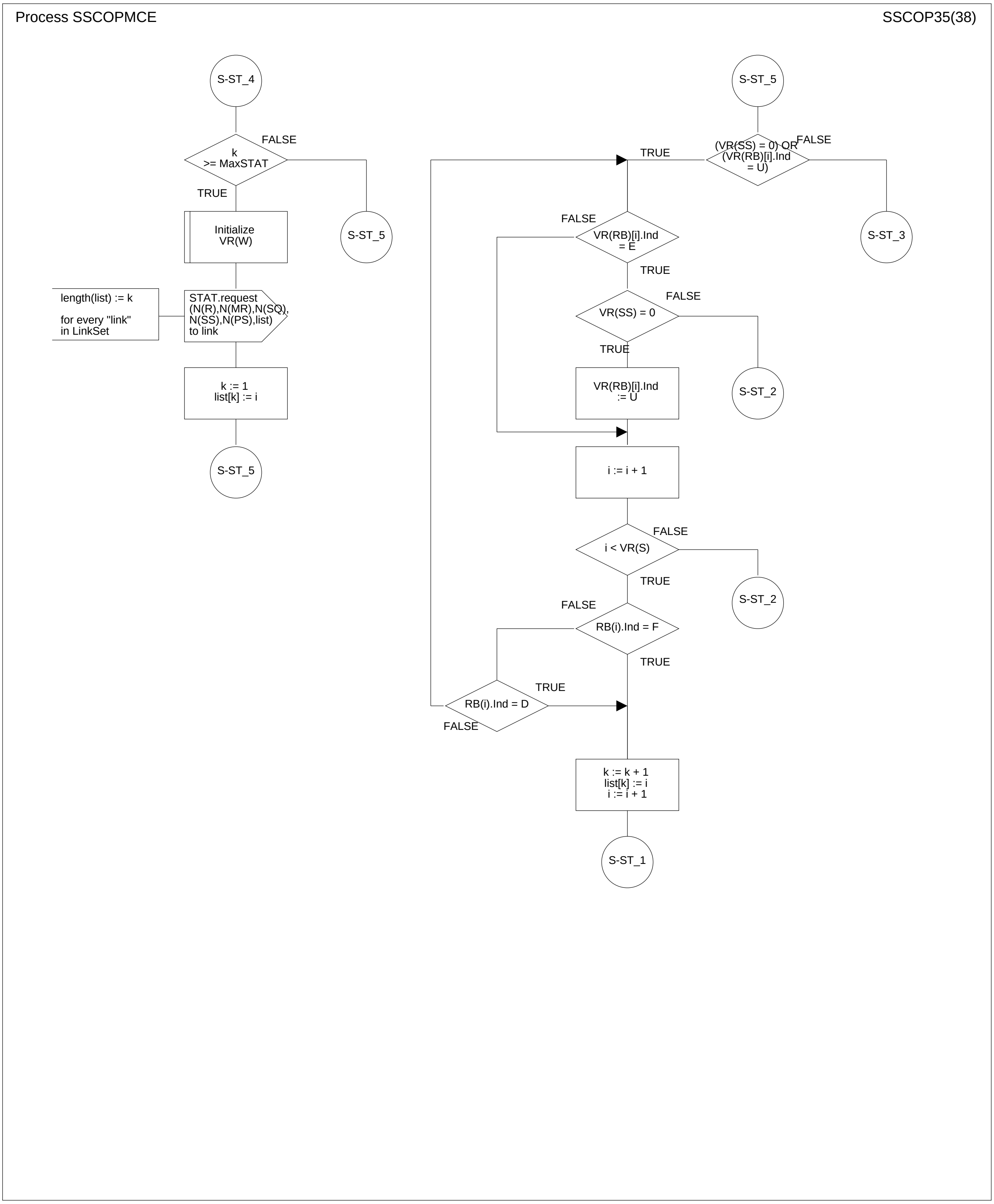


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 35 of 38)*

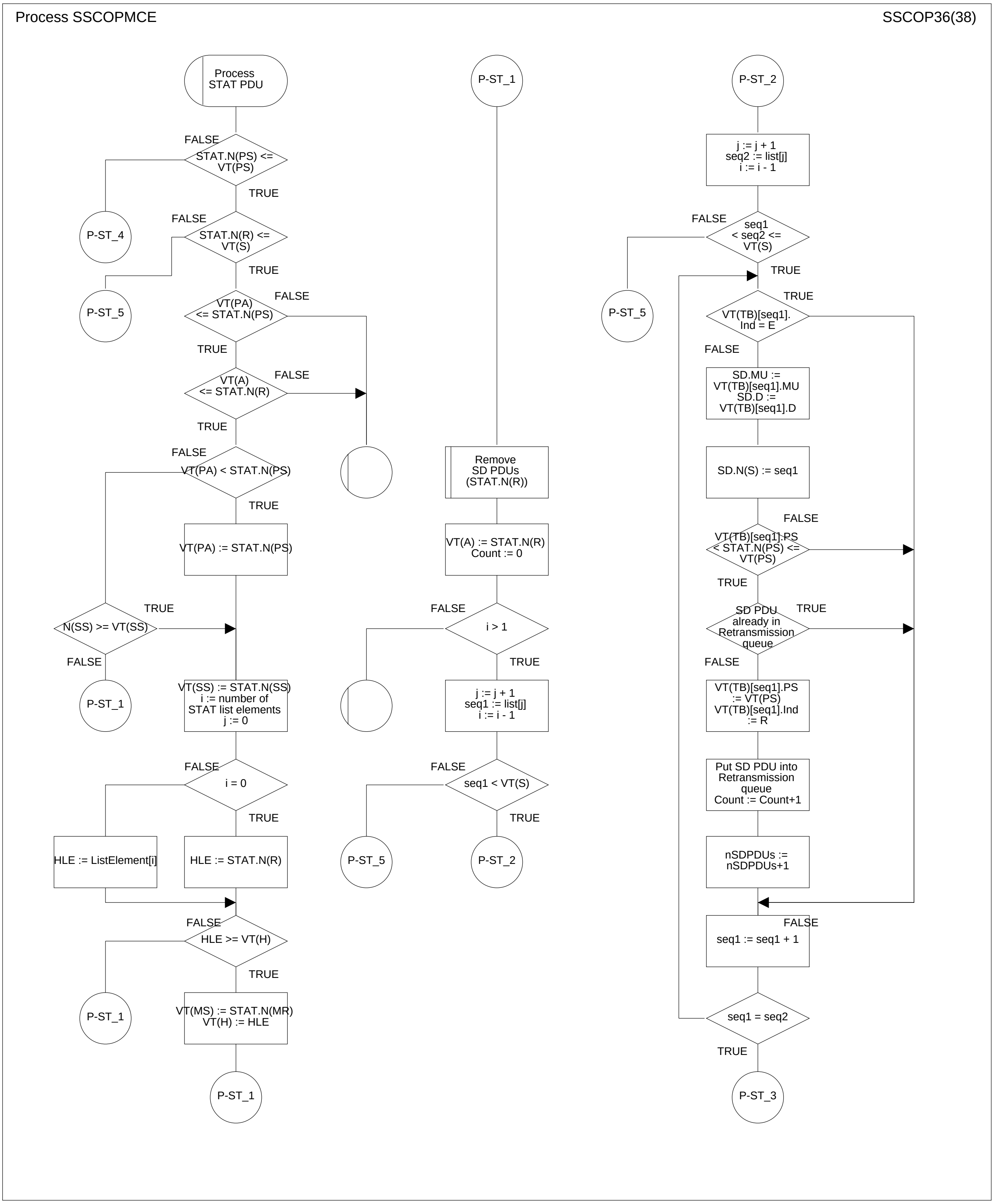


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 36 of 38)*

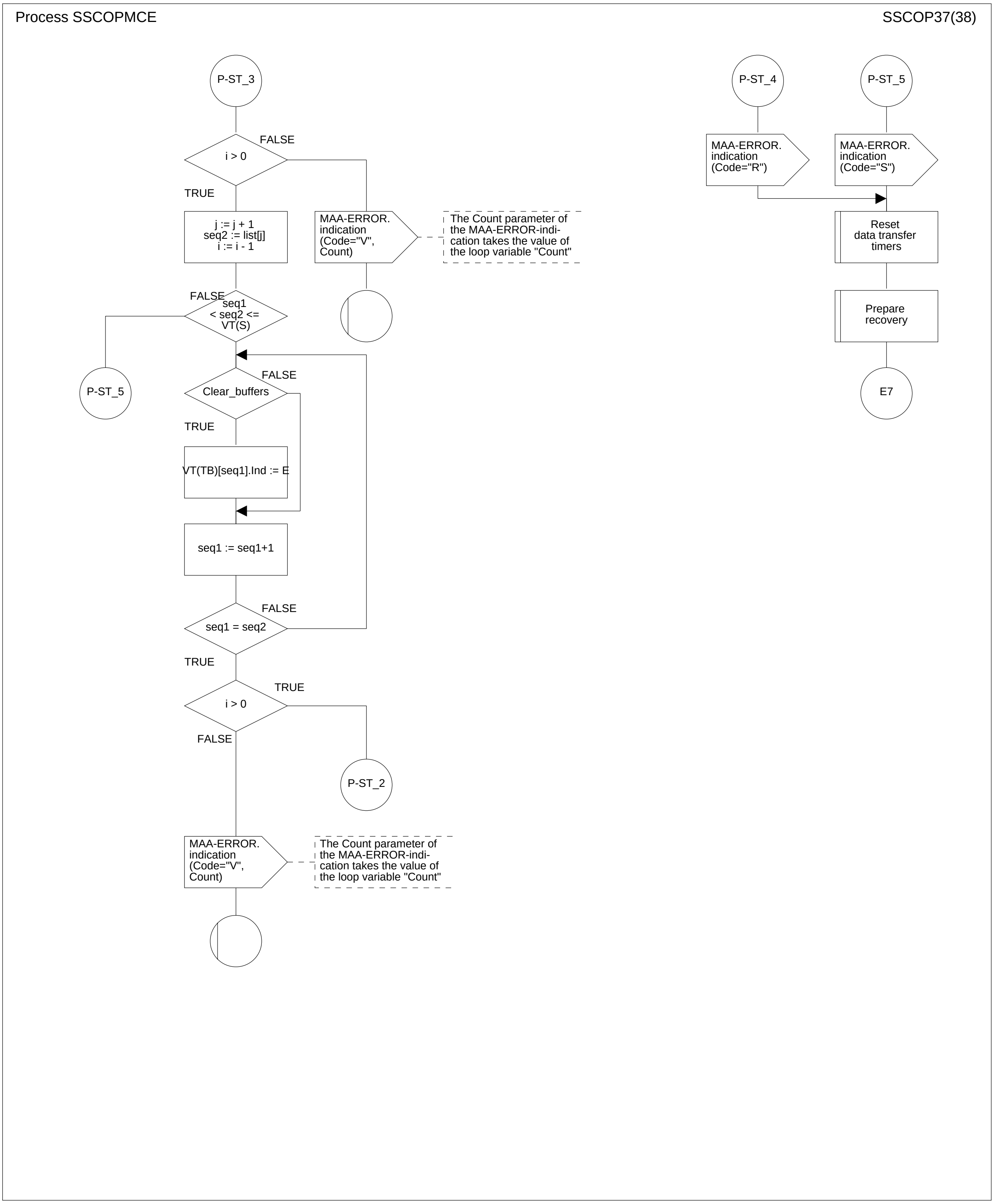


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity ***(sheet 37 of 38)***

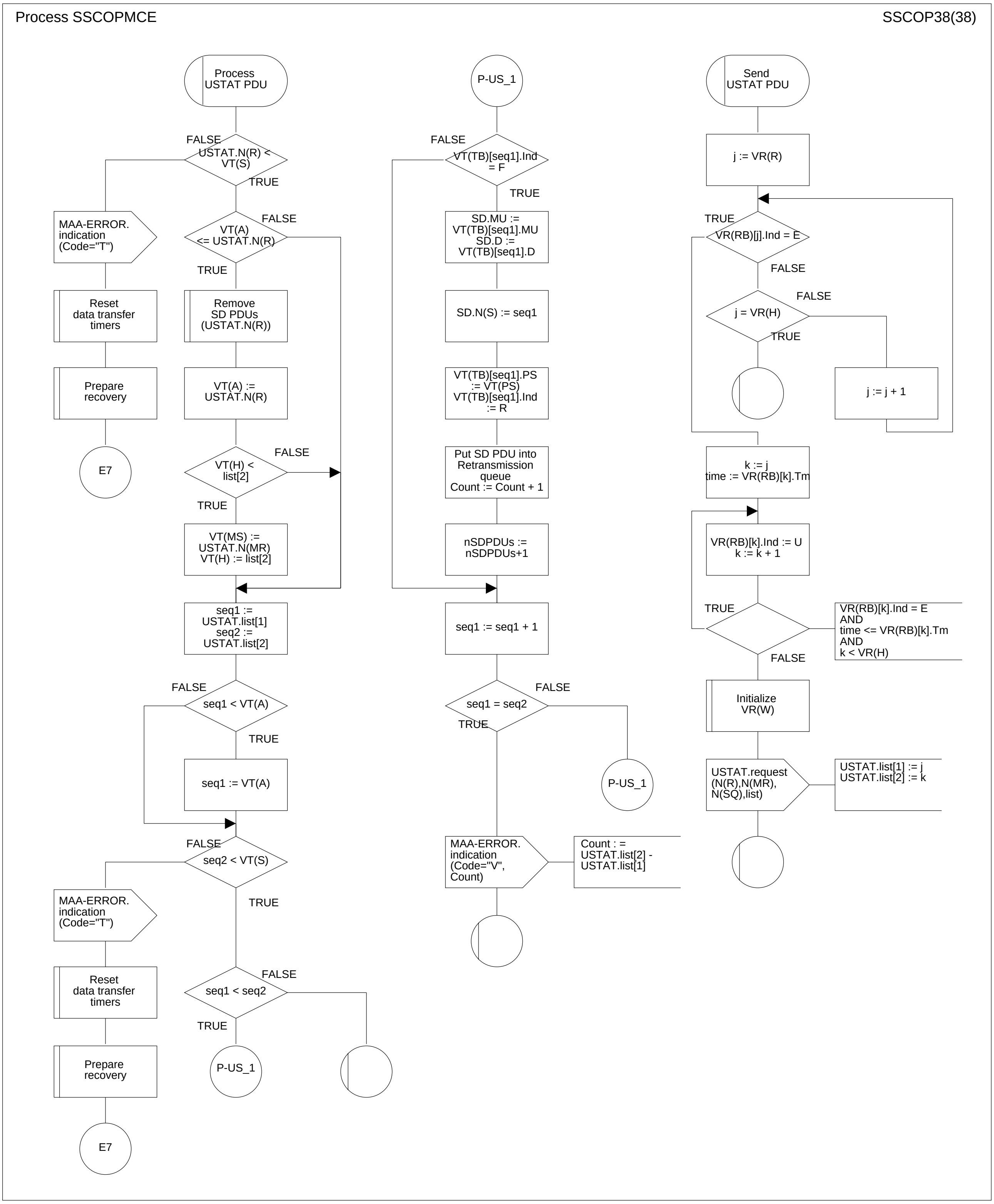


Figure 22/Q.2111 – SDL diagram of the multilink SSCOPMCE protocol entity *(sheet 38 of 38)*

ANNEX A

Management error indications

A number of events will cause errors to be indicated to the layer management entity. The associated error parameter contains the error code that describes the specific error conditions.

The column entitled "Error condition" together with the "Affected states" describes specific protocol error events and the basic state of the SSCOPMCE entity at the point that the MAA-ERROR indication primitive is generated (see Table A.1).

Table A.1/Q.2111 – Error types and codes

Error Type	Error Code	Error Condition	Affected States
Receipt of unsolicited or inappropriate PDU	A	SD PDU	1, 3, 6, 9
	B	BGN PDU	3, 5, 6, 7, 8, 9, 10
	C	BGAK PDU	1, 3, 6, 7, 8, 9
	D	BGREJ PDU	1, 3, 5, 6, 7, 8, 9, 10
	E	END PDU	3, 5, 6, 7, 8, 9, 10
	F	ENDAK PDU	3, 5, 6, 7, 8, 9, 10
	G	POLL PDU	1, 3, 6, 9
	H	STAT PDU	1, 3, 6, 8, 9
	I	USTAT PDU	1, 3, 6, 8, 9
	J	RS	1, 3, 5, 6, 7, 8, 9
	K	RSAK PDU	1, 3, 5, 6, 7, 8, 9
	L	ER	1, 3, 6, 7, 8, 9, 10
	M	ERAK	1, 3, 6, 7, 9
Unsuccessful retransmission	O	VT(CC) ≥ MaxCC	2, 4, 5, 7
	P	No link available	2, 3, 4, 5, 6, 7, 8, 9, 10
Other list elements error type	Q	SD or POLL, N(S) error	None
	R	STAT N(PS) error	10
	S	STAT N(R) or list elements error	10
	T	USTAT N(R) or list elements error	10
	U	PDU length violation	ALL (Note)
SD loss	V	SD PDUs must be retransmitted	10
Credit Condition	W	Lack of credit	10
	X	Credit obtained	10
Unnecessary retransmission	Y	SD PDU already received (i.e. delivered or in Receive buffer)	10
NOTE – This error is reported by the ancillary TRSP process (see Annex B).			

ANNEX B

SDL diagrams for the ancillary process TRSP

See Figure B.1.

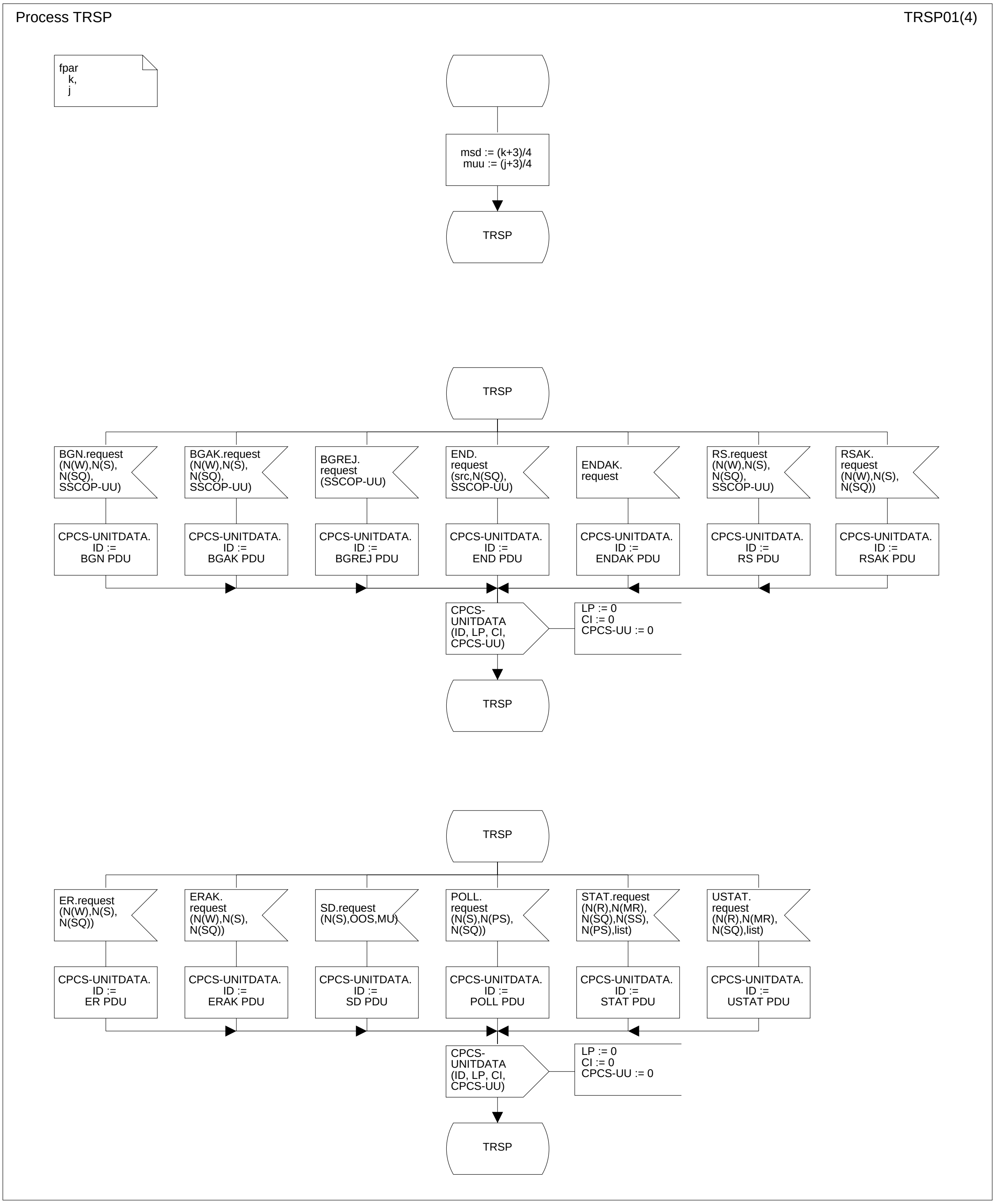


Figure B.1/Q.2111 – SDL diagram of the ancillary process TRSP *(sheet 1 of 4)*

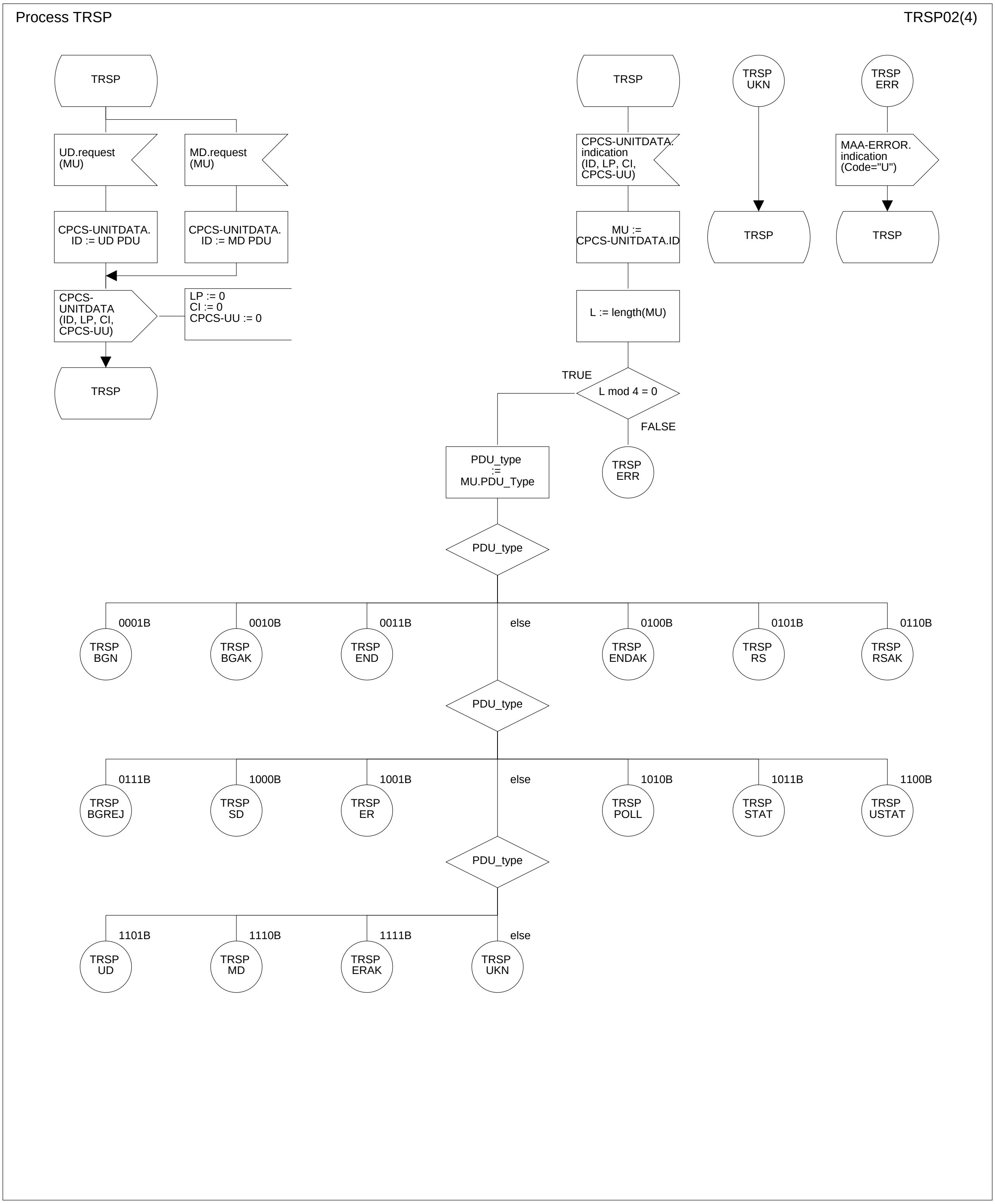


Figure B.1/Q.2111 – SDL diagram of the ancillary process TRSP *(sheet 2 of 4)*

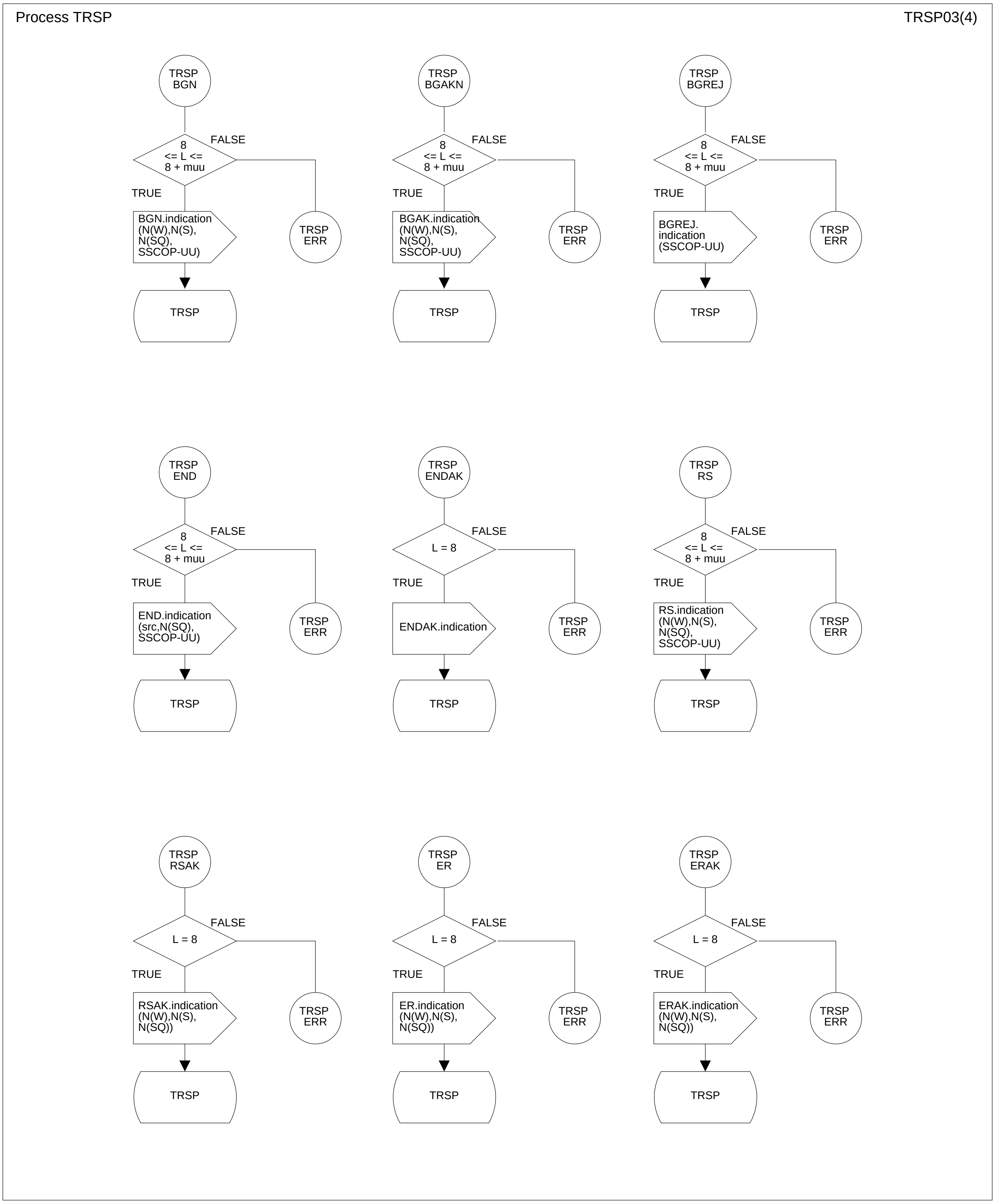


Figure B.1/Q.2111 – SDL diagram of the ancillary process TRSP *(sheet 3 of 4)*

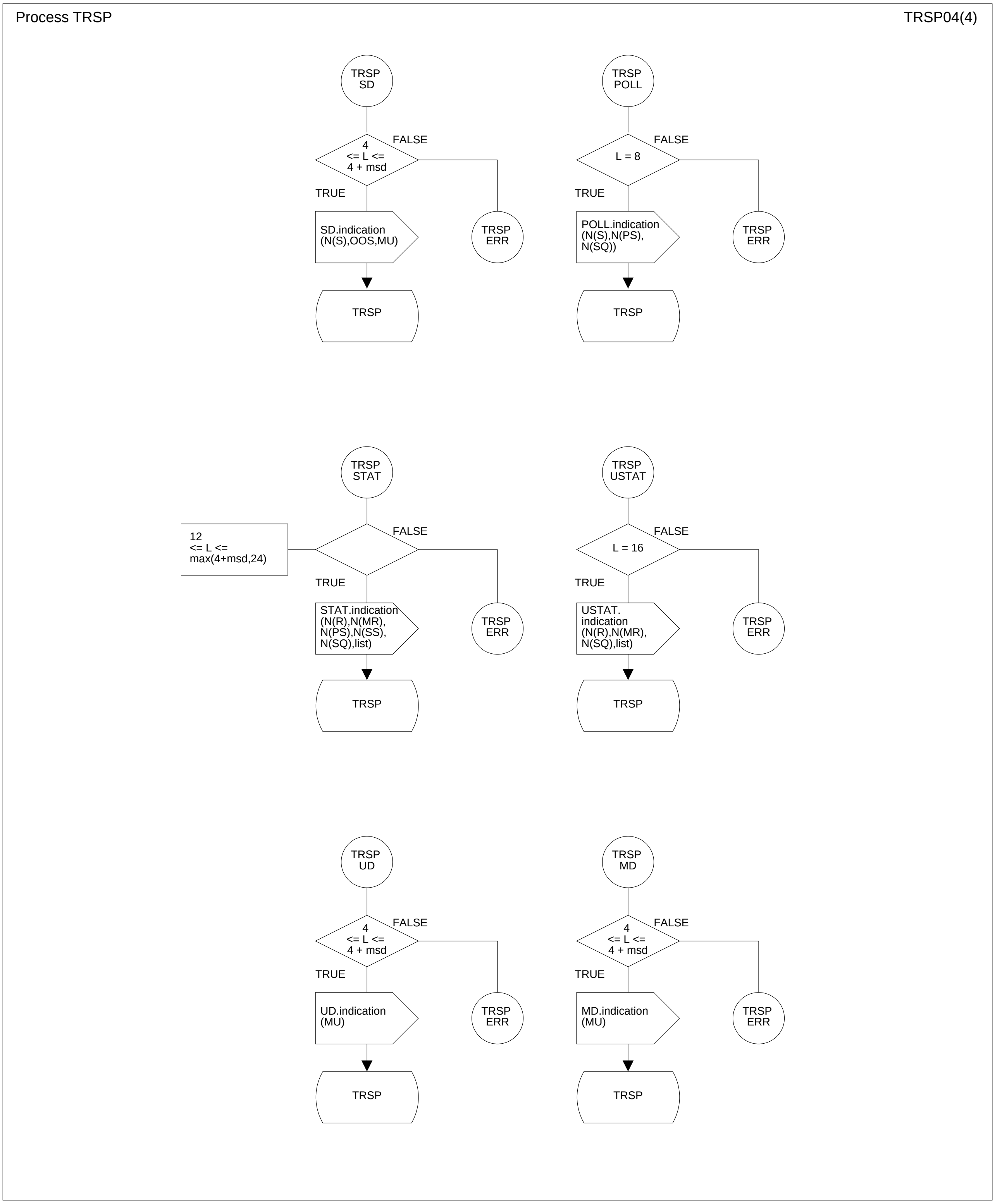


Figure B.1/Q.2111 – SDL diagram of the ancillary process TRSP *(sheet 4 of 4)*

ANNEX C

Convergence function for SSCOPMCE above IP or UDP

C.1 General description

The convergence function for SSCOPMCE above IP provides for the possibility to deploy SSCOPMCE on top of the connectionless service provided by IP. The IP service utilizes protocol defined in IETF RFCs 791 [8] and 1122 [10]. Alternatively, UDP service, as defined in IETF RFC 768 [7], may be used. Both alternatives are discussed in this annex.

EA1: All protocol stacks that include SSCOPMCE can, therefore, also be used in IP-based networks. A particular application of this arrangement is a protocol stack for SS No. 7 signalling.

NOTE − The convergence function of this annex, being based upon IETF RFC 791 [8], is designed specifically for operation with IPv4. If use is desired of an application, e.g. the IETF defined "DIFFSERV", that is not compatible with IETF RFC 791 [8], then this annex is not applicable.

C.2 Functions of the convergence function

The purpose of the convergence function is to map information between SSCOPMCE and IP (or UDP) PDUs. Appropriate headers must be created as is customarily done in the IP (or UDP) environment.

C.3 Specification of the convergence function

Clause 7.3 defines the primitives and parameters used at the lower boundary of the SSCOPMCE protocol entity. It shows that the parameters of the CPCS-UNITDATA.invoke primitive are used to model the transfer of information from SSCOPMCE protocol entity to the entity serving it. It also shows that the parameters of the CPCS-UNITDATA.signal primitive are used to model the transfer of information from the entity serving the SSCOPMCE protocol entity to that SSCOPMCE protocol entity.

C.3.1 The IP interface to its users

C.3.1.1 Description of the IP upper interface

The user interface to the IP is described by example in IETF RFC 791 [8] in a quasi-formal way through the exchange of "SEND" and "RECEIVE" primitives (although the language is modelled on descriptions of function calls in an operating system). All IP implementations must provide a certain minimum set of services to guarantee that all IP implementations can support the same protocol hierarchy.

Since Internet protocol is a datagram protocol, there is minimal memory or state maintained between datagram transmissions, and each call on the Internet protocol module by the user supplies all information necessary for the IP to perform the service requested.

When the user sends a datagram, it transmits the SEND primitive, supplying all the arguments. The Internet protocol module, on receiving this primitive, checks the arguments and prepares and sends the message. If either the arguments are bad, or the network does not accept the datagram, a reasonable report must be made to the user as to the cause of the problem, but the details of such reports are up to individual implementations.

When a datagram arrives at the Internet protocol module from the network, the information contained in the datagram is passed from the datagram to the user. If the user addressed does not exist, an ICMP error message is returned to the sender and the data is discarded, as described in IETF RFCs 792 [9] and 1122 [10].

IETF RFC 791 [8] defines the contents of the IP packet header as shown in Figure C.1.

The fields of the header shown in Figure C.1 are defined in IETF RFC 791 [8] as follows:

Version (4 bits)

The Version field indicates the format of the Internet header.

IHL (4 bits)

Internet Header Length is the length of the Internet header in 32 bit words, and thus points to the beginning of the data. Note that the minimum value for a correct header is 5.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

NOTE – Each tick mark represents one bit position.

Figure C.1/Q.2111 – Example Internet datagram header

Type of Service (8 bits)

The Type of Service provides an indication of the abstract parameters of the quality of service desired. These parameters are to be used to guide the selection of the actual service parameters when transmitting a datagram through a particular network. Several networks offer service precedence, which somehow treats high precedence traffic as more important than other traffic (generally by accepting only traffic above a certain precedence at time of high load). The major choice is a three way trade-off between low-delay, high-reliability, and high-throughput.

NOTE – According to IETF RFC 1122 [10], "minimal monetary cost" may also be a choice.

Total Length (16 bits)

Total Length is the length of the datagram, measured in octets, including Internet header and data. This field allows the length of a datagram to be up to 65 535 octets.

NOTE – The maximal Internet header is 60 octets, and a typical Internet header is 20 octets.

Identification (16 bits)

An identifying value assigned by the sender to aid in assembling the fragments of a datagram.

Flags (3 bits)

Various Control Flags.

Bit 0 reserved, must be zero
Bit 1 (DF) 0 = May Fragment, 1 = Don't Fragment.
Bit 2 (MF) 0 = Last Fragment, 1 = More Fragments.

Fragment Offset (13 bits)

This field indicates where in the datagram this fragment belongs. The fragment offset is measured in units of 8 octets (64 bits). The first fragment has offset zero.

Time to Live (8 bits)

This field indicates the maximum time the datagram is allowed to remain in the Internet system. If this field contains the value zero, then the datagram must be destroyed by an intermediate host (but not by the destination host). This field is modified in Internet header processing. The time is measured in units of seconds, but since every module that processes a datagram must decrease the TTL by at least one even if it processes the datagram in less than a second, the TTL must be thought of only as an upper bound on the time a datagram may exist. The intention is to cause undeliverable datagrams to be discarded, and to bound the maximum datagram lifetime.

Protocol (8 bits)

This field indicates the next level protocol used in the data portion of the Internet datagram. The values for various protocols are specified by the IETF. The numeric value for SSCOPMCE is "128".

Header Checksum (16 bits)

A checksum on the header only. Since some header fields change (e.g. time to live), this is recomputed and verified at each point that the Internet header is processed.

The checksum field is the 16 bit one's complement of the one's complement sum of all 16 bit words in the header. For purposes of computing the checksum, the value of the checksum field is zero.

Source Address (32 bits)

The source address. See section 3.2/IETF RFC 791 [8].

Destination Address (32 bits)

The destination address. See section 3.2/IETF RFC 791 [8].

Options (variable)

The options may appear or not in datagrams. They must be implemented by all IP modules (host and gateways). What is optional is their transmission in any particular datagram, not their implementation.

In some environments the security option may be required in all datagrams.

The option field is variable in length. There may be zero or more options. The specific coding of the options field may be found in IETF RFC 791 [8].

C.3.1.2 Transmitter Side Mapping

Figure C.2 shows the Service Data Unit and Parameters passed between the SSCOPMCE/Convergence Function and the IP layer at the transmitting side. In this figure, it can be seen that the relevant fields of the IP packet header should be coded as shown in Table C.1.

C.3.1.3 Receiver Side Mapping

Figure C.3 shows the Service Data Unit and Parameters passed between SSCOPMCE/Convergence Function and the IP layer at the receiving side.

Table C.1/Q.2111 – Transmitter side mapping

Version	(Note 1)	
Internet Header Length (IHL)	(Note 1)	
Type of Service	00000000	If "Cell Loss Priority" = 1
	00010000	If "Cell Loss Priority" = 0
Total Length (TL)	(Note 1)	
Identification	(Note 2)	
Flags	000	May fragment; last fragment
	001	May fragment; more fragments
Fragment Offset	(Note 1)	
Time to Live (TTL)	(Note 2)	
Protocol (PROT)	(Note 2)	"128"
IP Header Checksum	(Note 1)	
Source Address	(Note 2)	the IP address of the source node
Destination Address	(Note 2)	the IP address of the destination node
Options	(Note 1)	(Note 4)
Data	(Note 3)	1 to (65 535 – IHL)

NOTE 1 – Coding of this parameter is handled by the IP module using guidance provided in IETF RFC 791 [8].

NOTE 2 – Coding of this parameter is handled by the convergence function using the rules specified in IETF RFC 791 [8].

NOTE 3 – The SSCOP-PDU is appended with the CPCS-PDU trailer, coded as specified in ITU-T Recommendation I.363.5 [5].

NOTE 4 – For the purpose of this Recommendation, the user options "Loose Source and Record Route," "Strict Source and Record Route," "Record Route," and "Timestamp" apply. Other user options shall not be used and shall be silently ignored when received (see IETF RFC 1122 [10] section 3.2.1.8). It should be noted that the options "No Operation" (Type 1) and "End of List" (Type 0) are to be handled within the IP module; therefore, they are not passed to the transport layer.

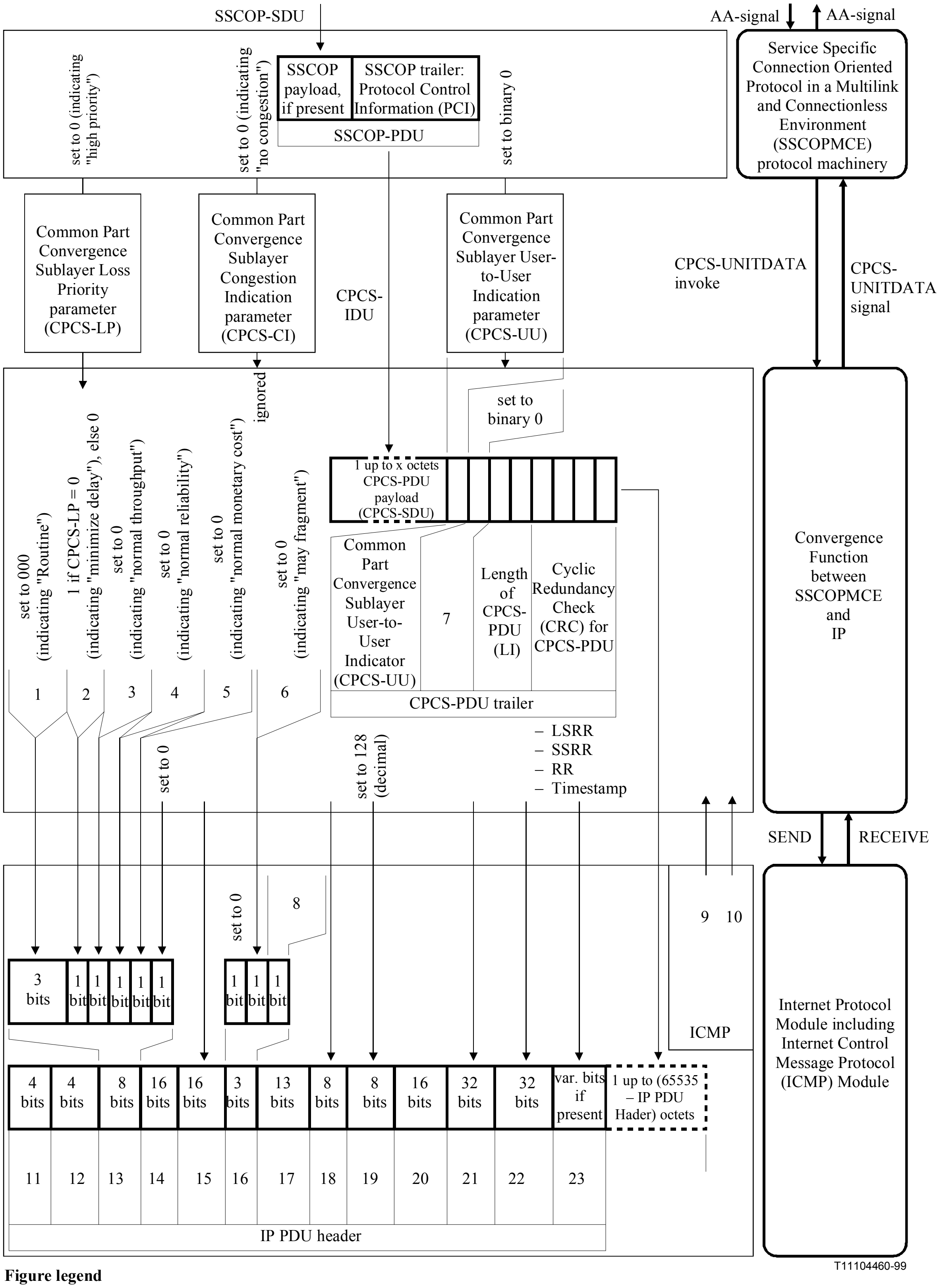


Figure legend

1 Precedence	9 "Time Exceeded" notification	17 Fragment Offset (FO)
2 Delay (D)	10 "Parameter Problem" notification	18 Time of Live (TTL)
3 Throughput (T)	11 Version	19 Protocol (PROT)
4 Reliability (R)	12 Internet Header Length (IHL)	20 IP Header Checksum
5 Monetary Cost (MC)	13 Type of Service (TOS)	21 Source Address
6 Don't Fragment (DF)	14 Total Length (TL)	22 Destination Address
7 Common Part Indicator (CPI)	15 Identification (ID)	23 Options
8 More Fragments (MF)	16 Flags	

Figure C.2/Q.2111 – Service Data Unit and Parameters passed between SSCOPMCE/ Convergence Function and IP Layer – Transmitting side

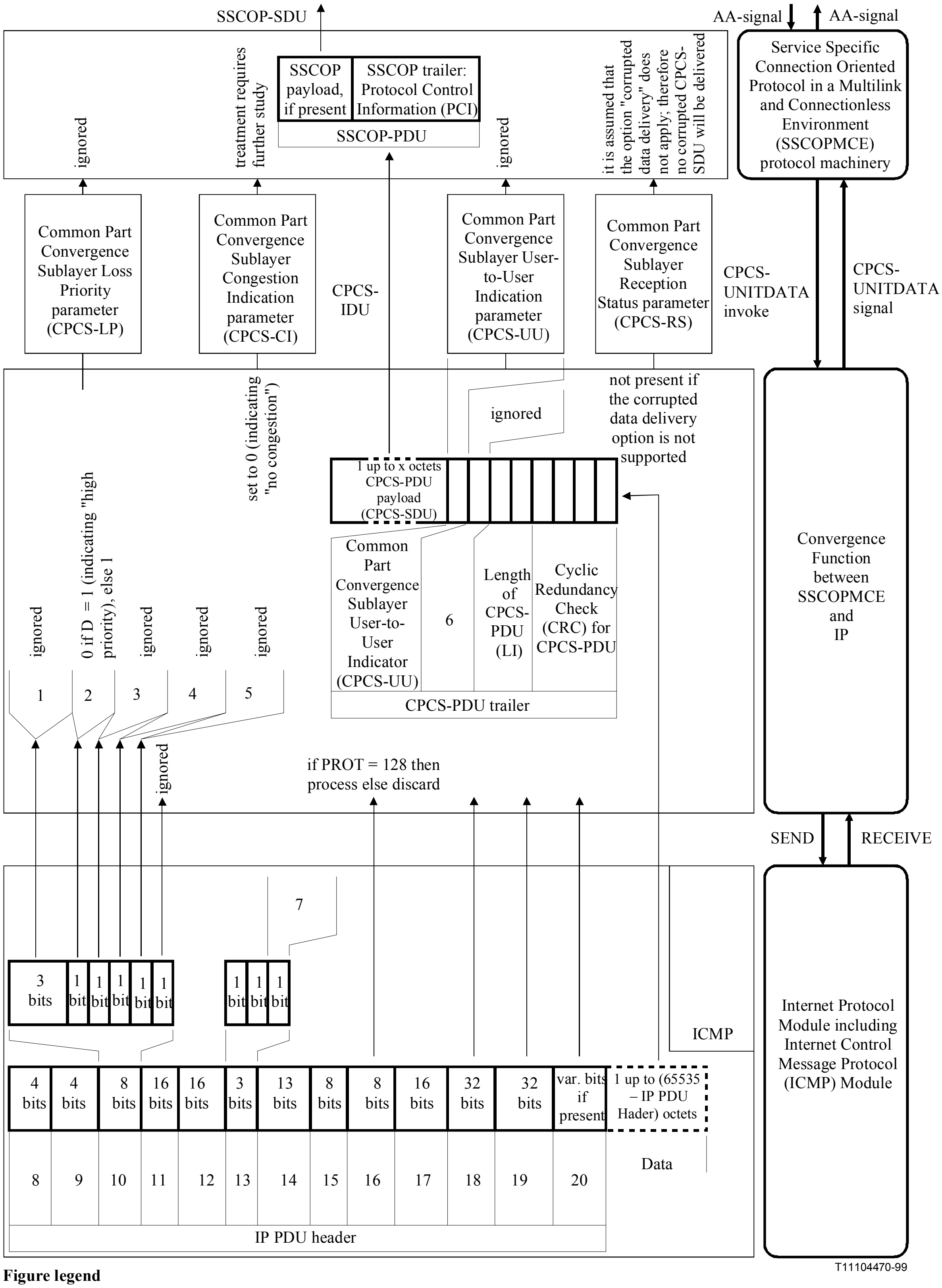


Figure legend

1 Precedence
2 Delay (D)
3 Throughput (T)
4 Reliability (R)
5 Monetary Cost (MC)
6 Common Part Indicator (CPI)
7 More Fragments (MF)
8 Version
9 Internet Header Length (IHL)
10 Type of Service (TOS)
11 Total Length (TL)
12 Identification (ID)
13 Flags
14 Fragment Offset (FO)
15 Time of Live (TTL)
16 Protocol (PROT)
17 IP Header Checksum
18 Source Address
19 Destination Address
20 Options

Figure C.3/Q.2111 – Service Data Unit and Parameters passed between SSCOPMCE/ Convergence Function and IP Layer – Receiving side

C.3.2 The UDP interface to its users

C.3.2.1 Description of the UDP upper interface

IETF RFC 768 [7] defines the parameters of the UDP packet header as shown in Figure C.4.

```
 0      7 8     15 16    23 24    31
+--------+--------+--------+--------+
|     Source      |   Destination   |
|      Port       |      Port       |
+--------+--------+--------+--------+
|                 |                 |
|     Length      |    Checksum     |
+--------+--------+--------+--------+
|                                   |
|          data octets ...          |
+---------------- ...               |
```

Figure C.4/Q.2111 – UDP header format

The fields of the header shown in Figure C.4 are defined in IETF RFC 768 [7] as follows:

Source Port (16 bits)

Source Port is an optional field, when meaningful, it indicates the port of the sending process, and may be assumed to be the port to which a reply should be addressed in the absence of any other information. If not used, a value of zero is inserted.

Destination Port (16 bits)

Destination Port has a meaning within the context of a particular Internet destination address.

Length (16 bits)

Length is the length in octets of this user datagram including this header and the data. (This means the minimum value of the length is eight.)

Checksum (16 bits)

Checksum is the 16-bit one's complement of the one's complement sum of a pseudo-header of information from the IP header, the UDP header, and the data, padded with zero octets at the end (if necessary) to make a multiple of two octets.

The pseudo-header conceptually prefixed to the UDP header contains the source address, the destination address, the protocol, and the UDP length. This information gives protection against misrouted datagrams. This checksum procedure is the same as is used in TCP.

NOTE – For the purpose of this ITU-T Recommendation the source address, the destination address, and the protocol are modelled as parameters.

C.3.2.2 Transmitter Side Mapping

Figure C.5 shows the Service Data Unit and Parameters passed between the SSCOPMCE/Convergence Function and the UDP and IP modules at the transmitting side.

C.3.2.3 Receiver Side Mapping

Figure C.6 shows the Service Data Unit and Parameters passed between the SSCOPMCE/Convergence Function and the UDP and IP modules at the receiving side.

C.4 Layer Management

There are no interactions with layer management defined.

It is for further study whether there exists a need for the Convergence Function to invoke the services of the Internet Control Message Protocol (ICMP) to notify the peer of error situations, such as Protocol Unreachable and Port Unreachable, in the absence of a peer-to-peer mechanism (see IETF RFC 1122 [10] section 3.2.2.1).

It is for further study whether SSCOPMCE should provide positive and/or negative advice to modify the routing of messages upon "Dead Gateway Detection" (see IETF RFC 1122 [10] section 3.3.1.4).

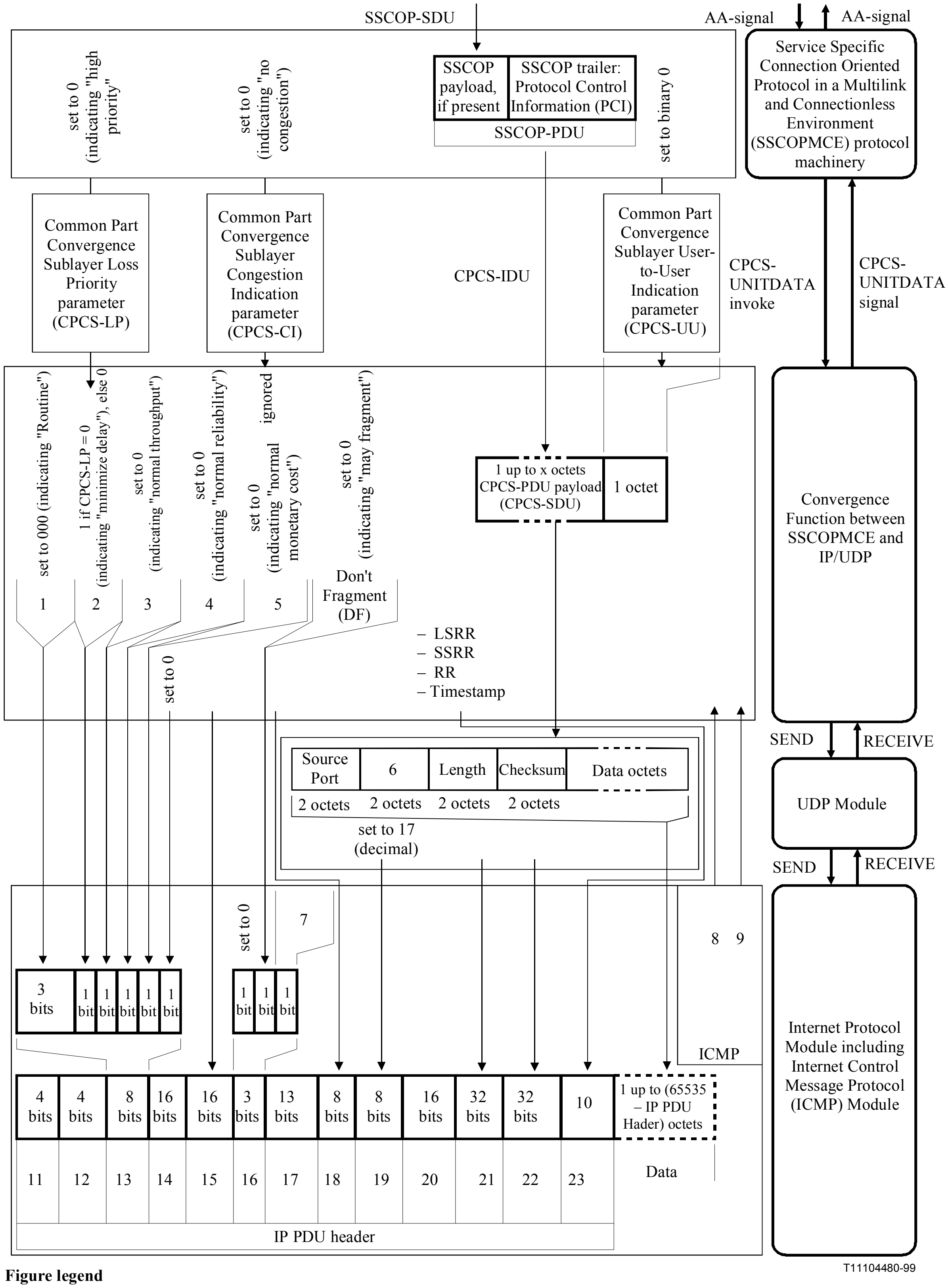


Figure legend

1 Precedence	9 "Parameter Problem" notification	17 Fragment Offset (FO)
2 Delay (D)	10 Var. bits if present	18 Time of Live (TTL)
3 Throughput (T)	11 Version	19 Protocol (PROT)
4 Reliability (R)	12 Internet Header Length (IHL)	20 IP Header Checksum
5 Monetary Cost (MC)	13 Type of Service (TOS)	21 Source Address
6 Destination Port	14 Total Length (TL)	22 Destination Address
7 More Fragments (MF)	15 Identification (ID)	23 Options
8 "Time Exceeded" notification	16 Flags	

Figure C.5/Q.2111 – Service Data Unit and Parameters passed between SSCOPMCE/ Convergence Function and UDP/IP Layer – Transmitting side

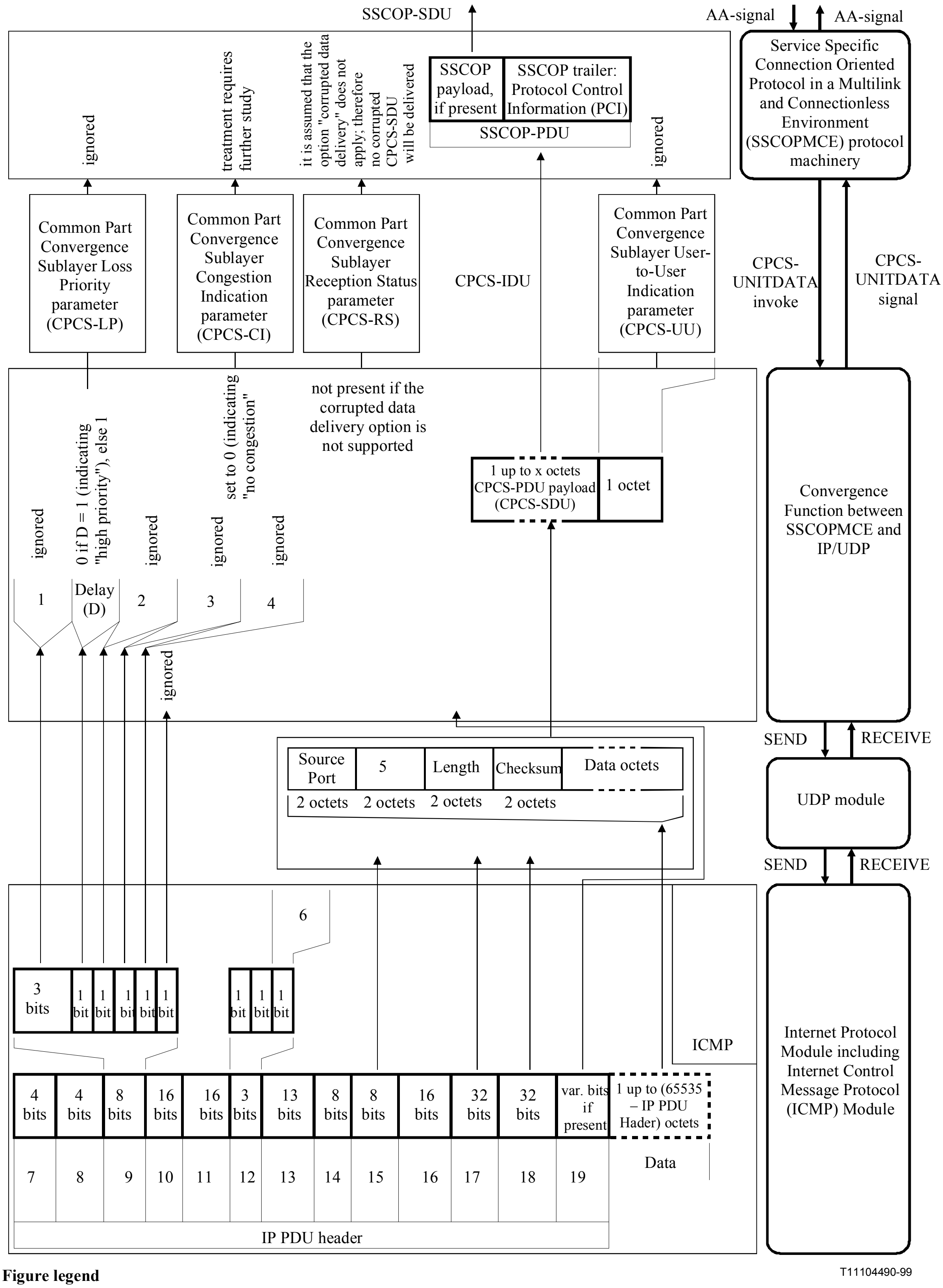


Figure legend

1 Precedence
2 Throughput (T)
3 Reliability (R)
4 Monetary Cost (MC)
5 Destination Port
6 More Fragments (MF)
7 Version
8 Internet Header Length (IHL)
9 Type of Service (TOS)
10 Total Length (TL)
11 Identification (ID)
12 Flags
13 Fragment Offset (FO)
14 Time of Live (TTL)
15 Protocol (PROT)
16 IP Header Checksum
17 Source Address
18 Destination Address
19 Options

Figure C.6/Q.2111 – Service Data Unit and Parameters passed between SSCOPMCE/Convergence Function and UDP/IP Layer – Receiving side

APPENDIX I[1]

Protocol Implementation Conformance Statement (PICS) Proforma

I.1 Introduction

Prior to the conformance testing and the interoperability testing of Implementations Under Test (IUTs), it is necessary to have the PICS (Protocol Implementation Conformance Statement) document for an implementation.

This particular PICS deals with the implementation of the Service Specific Connection Oriented Protocol in a Multilink and Connectionless Environment (SSCOPMCE).

I.1.1 Scope

This appendix provides the PICS proforma for the Service Specific Connection Oriented Protocol in a Multilink and Connectionless Environment (SSCOPMCE) [1], in compliance with the relevant requirements, and in accordance with the relevant guidelines, given in ITU-T Recommendation X.296 [3].

I.1.2 Normative references

[1] ITU-T Recommendation Q.2111 (1999), *B-ISDN ATM adaptation layer – Service Specific Connection Oriented Protocol in a Multilink and Connectionless Environment (SSCOPMCE)*.

[2] ITU-T Recommendation X.290 (1995), *OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications – General concepts*.

ISO/IEC 9646-1:1994, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 1: General concepts.*

[3] ITU-T Recommendation X.296 (1995), *OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications – Implementation conformance statements*.

ISO/IEC 9646-7:1995, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 7: Implementation Conformance Statements.*

I.1.3 Definitions

IUT Implementation Under Test

M Mandatory

N/A Not applicable

NOT item not supported; absence of item

O Optional

O.<n> Optional, but, if chosen, support is required for either at least one or only one of the options in the group labelled by the same numeral <n>

PDU Protocol Data Unit

[1] **Copyright release for PICS proforma**

Users of this Recommendation may freely reproduce the PICS proforma in this appendix so that it can be used for its intended purpose, and may further publish the completed PICS.

PICS	Protocol Implementation Conformance Statement
S.<i>	Supplementary information number i
SDU	Service Data Unit
SUT	System Under Test
X.<i>	Exceptional information number i

I.1.4 Conformance Statement

The supplier of a protocol implementation which is claimed to conform to the Service Specific Connection Oriented Protocol in a Multilink and Connectionless Environment Specification (SSCOPMCE) is required to complete a copy of the PICS proforma provided in I.2 and is required to provide the information necessary to identify both the supplier and the implementation.

I.2 PICS Proforma

I.2.1 Identification of the PICS Proforma Corrigenda

Identification of corrigenda applied to this PICS proforma	Rec. Q.2111 (1999) Corr.: Corr.:

I.2.2 Instructions for Completing the PICS Proforma

The PICS Proforma is a fixed-format questionnaire. Answers to the questionnaire should be provided in the rightmost columns, either by simply indicating a restricted choice (such as Yes or No), or by entering a value or a set of range of values.

A supplier may also provide additional information, categorized as exceptional or supplementary information. An exception item should contain the appropriate rationale.

The supplementary information is not mandatory and the PICS is complete without such information. The presence of optional supplementary or exception information should not affect test execution, and will in no way affect interoperability verification.

NOTE – Where an implementation is capable of being configured in more than one way, a single PICS may be able to describe all such configurations. However, the supplier has the choice of providing more than one PICS, each covering some subset of the implementation's configuration capabilities, in case this makes for easier or clearer presentation of the information.

I.2.3 Identification of the Implementation

Implementation Under Test (IUT)

Identification

IUT Name: ______________________________

IUT Version: ______________________________

____

System Under Test

SUT Name: ______________________________

____

Hardware Configuration: ______________________________

____

____

Operating System: ______________________________

Product Supplier

Name: ______________________________

Address: ______________________________

____

Telephone Number: ______________________________

Facsimile Number: ______________________________

Email Address (optional): ______________________________

Additional Information: ______________________________

Client

Name: ______________________________

Address: ______________________________

____

Telephone Number: ______________________________

Facsimile Number: ______________________________

Email Address (optional): ______________________________

Additional Information: ______________________________

PICS Contact Person

Name: ______________________________

Address: ______________________________

____

Telephone Number: ______________________________

Facsimile Number: ______________________________

Email Address (optional): ______________________________

Additional Information: ______________________________

Identification of the protocol

This PICS proforma applies to the following document:
ITU-T Recommendation Q.2111, "Service Specific Connection Oriented Protocol in a Multilink and Connectionless Environment (SSCOPMCE)"

I.2.4 Global Statement of Conformance

The implementation described in this PICS meets all of the mandatory requirements of the reference protocol.

___Yes

___No

NOTE – Answering "No" indicates non-conformance to the specified protocol. Non-supported mandatory capabilities are to be identified in the PICS, with an explanation of why the implementation is non-conforming.

I.2.4.1 Roles

Item number	Item description	Ref.	Status	Support
R1	Transmitter and Receiver as a general protocol engine	5.1	O.1	
R2	Transmitter and Receiver in a restricted protocol engine	5.1	O.1	
O.1	Support of one and only one of these items is required.			

I.2.4.2 Major capabilities

Item number	Item description	Ref.	Status	Support
MC1	Multilink mode (Mode "A")	5.3; 6 k); 8.7 a)	O.1	
MC2	Connectionless mode (Mode "B")	5.3; 6 k); 8.7 a)	O.1	
MC3	Compatibility mode (to Q.2110 procedures – Mode "C")	5.3; 6 k); 8.7 a)	O.1	
MC4	Assured data transfer between two SSCOPMCE users	5.2; 6 h)	R1 M R2 O	
MC5	Unassured data transfer between two SSCOPMCE users	5.2; 6 h)	R1 M R2 O	
MC6	Unassured data transfer between two SSCOPMCE layer management entities	5.2; 6 h)	R1 M R2 O	
MC7	Connection establishment, release, and resynchronization	5.2; 6 g)	R1 OR MC4 M R2 O	
MC8	Out of sequence delivery	5.2; 6 l)	R1 M R2 AND MC4 O R2 AND NOT MC4 N/A	
MC9	Local data retrieval by the user	5.2; 6 f)	R1 M R2 AND MC4 O R2 AND NOT MC4 N/A	

Item number	Item description	Ref.	Status	Support
MC10	Error reporting to layer management	5.2; 6 d)	R1 M R2 AND MC4 O R2 AND NOT MC4 N/A	
MC11	Adding and removing links	5.4; 7.2.1 d), e)	M	
O.1	Support of at least one of these items is required.			

I.2.4.3 SSCOPMCE protocol functions

Item number	Protocol function	Ref.	Status	Support
PF1	Assured data transfer with sequence integrity	6 a), h); 7.1.1 c); 8.1 j)	MC4 M NOT MC4 N/A	
PF2	Assured data transfer with error correction by selective retransmission	6 b), h); 7.1.1 c); 8.1 j)	MC4 M NOT MC4 N/A	
PF3	Assured data transfer with flow control	6 c), h); 7.1.1 c); 8.1 j)	MC4 M NOT MC4 N/A	
PF4	Keep alive function	6 e)	MC4 M NOT MC4 N/A	
PF5	Connection establishment for the management of assured data transfer	6 g); 7.1.1 a); 8.1 a), b), c)	MC4 M NOT MC4 N/A	
PF6	Connection release for the management of assured data transfer	6 g); 7.1.1 b); 8.1 d), e)	MC4 M NOT MC4 N/A	
PF7	Connection resynchronization for the management of assured data transfer	6 g); 7.1.1 d); 8.1.f), g)	MC4 M NOT MC4 N/A	
PF8	Protocol error detection and recovery	6 i); 7.1.1 e); 8.1 h), i)	MC4 M NOT MC4 N/A	
PF9	Status reporting	6 j); 8.1 k), l), m)	MC4 M NOT MC4 N/A	
PF10	Error reporting to layer management	6 d); 7.2.1 a)	MC4 AND MC10 M ELSE N/A	
PF11	Local data retrieval	6 f); 7.1.1 g), h)	MC4 AND MC9 M ELSE N/A	
PF12	Out of sequence delivery	6 l), h); 7.1.1 c); 8.1 j)	MC4 AND MC8 M ELSE N/A	

Item number	Protocol function	Ref.	Status	Support
PF13	Unassured data transfer between users	6 h); 7.1.1 f); 8.1 n)	MC5 M NOT MC5 N/A	
PF14	Transfer of Management-Data	6 m); 7.2.1 b); 8.1 o)	MC6 M NOT MC6 N/A	

I.2.4.4 PDUs

Item number	Item description	Ref.	Status	Support
PDU type				
PDU1 (Note)	BGN PDU	8.1 a); Fig. 5	MC4 M NOT MC4 N/A	
PDU2 (Note)	BGAK PDU	8.1 b); Fig. 6	MC4 M NOT MC4 N/A	
PDU3 (Note)	BGREJ PDU	8.1 c); Fig. 7	MC4 M NOT MC4 N/A	
PDU4 (Note)	END PDU	8.1 d); Fig. 8	MC4 M NOT MC4 N/A	
PDU5 (Note)	ENDAK PDU	8.1 e); Fig. 9	MC4 M NOT MC4 N/A	
PDU6 (Note)	RS PDU	8.1 f); Fig. 10	MC4 M NOT MC4 N/A	
PDU7 (Note)	RSAK PDU	8.1 g); Fig. 11	MC4 M NOT MC4 N/A	
PDU8 (Note)	ER PDU	8.1 h); Fig. 12	MC4 M NOT MC4 N/A	
PDU9 (Note)	ERAK PDU	8.1 i); Fig. 13	MC4 M NOT MC4 N/A	
PDU10 (Note)	SD PDU	8.1 j); Fig. 14	MC4 M NOT MC4 N/A	
PDU11 (Note)	POLL PDU	8.1 k); Fig. 15	MC4 M NOT MC4 N/A	
PDU12 (Note)	STAT PDU	8.1 l); Fig. 16	MC4 M NOT MC4 N/A	
PDU13 (Note)	USTAT PDU	8.1 m); Fig. 17	MC4 M NOT MC4 N/A	
PDU14 (Note)	UD PDU	8.1 n); Fig. 18	MC5 M NOT MC5 N/A	
PDU15 (Note)	MD PDU	8.1 o); Fig. 18	MC6 M NOT MC6 N/A	
PDU16	Invalid PDU recognition and discard	8.1.	M	
Formats				
PDU17	Coding conventions	8.2.1.	M	
PDU18	Padding in SD PDUs and use of PL field	8.2.2 a)	MC4 M NOT MC4 N/A	
PDU19	Padding in UD PDUs and use of PL field	8.2.2 a)	MC5 M NOT MC5 N/A	

Item number	Item description	Ref.	Status	Support
PDU20	Padding in MD PDUs and use of PL field	8.2.2 a)	MC6 M NOT MC6 N/A	
PDU21	Padding in BGN, BGAK, BGREJ, END, and RS PDUs and use of PL field	8.2.2 b)	MC4 M NOT MC4 N/A	
PDU22	Padding in STAT and USTAT PDUs	8.2.2 c)	MC4 M NOT MC4 N/A	
PDU23	Reserved fields	8.2.3	M	
PDU24	PDU Length	8.2.4	M	
PDU25	Coding of the list elements in STAT and USTAT PDUs	8.2.5	MC4 M NOT MC4 N/A	
PDU26	Segmentation of STAT PDUs	8.2.5	MC4 M NOT MC4 N/A	
NOTE 1 – The coding of the fields of the PDUs is specified in 8.5.				

I.2.4.5 Arithmetic operations on state variables

This subclause is applicable only if the major capability MC4 is implemented.

Item number	Item description	Ref.	Status	Support
Modulo arithmetic				
AO1	Modulo 2^{24} arithmetic of state variables VT(S), VT(A), VT(MS), VT(H), VR(R), VR(H), VR(MR), and VR(S)	8.4.1	M	
AO2	Modulo 2^{24} arithmetic of state variables VT(PS), VT(PA), VR(PS), and VR(SPS)	8.4.1	M	
AO3	Modulo 2^{8} arithmetic of state variables VT(SQ) and VR(SQ)	8.4.1	M	
Base for comparison				
AO4	VT(A) – 2^{23} when involving SD PDU sequemce numbers at the transmitter	8.4.1	M	
AO5	VR(R) – 2^{23} when involving SD PDU sequemce numbers at the receiver	8.4.1	M	
AO6	VT(PA) – 2^{23} when involving POLL PDU sequemce numbers at the transmitter	8.4.1	M	
AO7	VR(PS) – 2^{23} when involving POLL PDU sequemce numbers at the receiver	8.4.1	M	
AO8	VR(SQ) – 2^{7} when involving N(SQ) of SD PDUs	8.4.1	M	

I.2.4.6 Value range of state variables

This subclause is applicable only if the major capability MC4 is implemented.

Item number	Item description	Ref.	Status	Support
VR1	Value range for VT(PD) of "0" to the maximum permissible value of "MaxPD"	I.2.5.1	M	
VR2	Value range for VT(CC) of "0" to the maximum permissible value of "MaxCC"	I.2.5.1	M	
VR3	Value range for VT(SS) of "0" to "255"	8.4.1	M	

Item number	Item description	Ref.	Status	Support
VR4	Value range for VR(SS) of "0" to "255"	8.4.1	M	
VR5	Size of the boolean array of VT(x) at least the maximum permissible value of "MaxLinks"	8.4.1	M	
VR6	Size of the boolean array of VR(x) at least the maximum permissible value of "MaxLinks"	8.4.1	M	
VR7	Value range for nlinks of "0" to the maximum permissible value of "MaxLinks"	8.4.1	M	

I.2.4.7 Protocol features

This subclause is applicable only if the major capability MC4 is implemented.

NOTE – The protocol features refer to the SDL diagrams; any implementation showing to the environment the same behaviour as the SDL diagrams is conforming.

I.2.4.7.1 Start-up

Item number	Protocol Feature	Ref.	Status	Support
PSU1	State "Guard" and Timer_GUARD	Figure 22 (2 of 38)	M	
PSU2	Initialization of state variables	Figure 22 (2 of 38)	M	

I.2.4.7.2 Connection control procedures

Item number	Protocol Feature	Ref.	Status	Support
PCC1	Connection establishment and release	Figure 22 (3 to 10 of 38)	M	
PCC2	Connection resynchronization	Figure 22 (11 to 15 of 38)	MC4 AND PF7 M ELSE N/A (Note 1)	
PCC3	Connection recovery	Figure 22 (16 to 21 of 38)	MC4 AND PF8 M ELSE N/A (Note 2)	
PCC4	Active Timer_CC in states 2, and 4	Figure 22 (5 and 9 of 38)	M	
PCC5	Active Timer_CC in state 5	Figure 22 (11 of 38)	MC4 AND PF7 M ELSE N/A	
PCC6	Active Timer_CC in state 7	Figure 22 (16 of 38)	MC4 AND PF8 M ELSE N/A	
PCC7	Exiting state 10 "Data Transfer Ready"	Figure 22 (22 to 24 of 38)	M	
NOTE 1 – If States 5 and 6 are not implemented neither the AA_RESYNC primitives nor recognition of RS and RSAK PDUs is possible. NOTE 2 – If States 7, 8, and 9 are not implemented neither the AA_RECOVER primitives nor recognition of ES and ESAK PDUs is possible. NOTE 3 – Some of the connection control procedures make use of macros defined in Figure 22 (27 to 29 of 38).				

I.2.4.7.3 Assured data transfer procedure

Item number	Protocol Feature	Ref.	Status	Support
PAD1	Pre- and postprocessing procedures on receipt of POLL, STAT, and USTAT PDUs	Figure 22 (24 of 38)	M	
PAD2	Procedures after timer expiries	Figure 22 (25 of 38)	M	
PAD3	Procedures after "enabling conditions"	Figure 22 (25 of 38)	M	
PAD4	Procedures for sending an SD PDU	Figure 22 (30 of 38)	M	
PAD5	Procedures for processing a received SD PDU	Figure 22 (31 and 32 of 38)	M	
PAD6	Procedures for sending a POLL PDU	Figure 22 (33 of 38)	M	
PAD7	Procedures for processing a received POLL PDU	Figure 22 (33 of 38)	M	
PAD8	Procedures for sending a STAT PDU	Figure 22 (34 and 35 of 38)	M	
PAD9	Procedures for processing a received STAT PDU	Figure 22 (36 and 37 of 38)	M	
PAD10	Procedures for sending a USTAT PDU	Figure 22 (38 of 38)	M	
PAD11	Procedures for processing a received USTAT PDU	Figure 22 (38 of 38)	M	
NOTE – Some of the assured data transfer procedures make use of macros defined in Figure 22 (27 to 29 of 38).				

I.2.5 Supported values

I.2.5.1 Timers

This subclause is applicable only if the major capability MC4 is implemented.

Item number	Item description	Ref.	Status	Support	Values	
					Allowed	Supported
T1	Timer_CC	8.6; Figure 22 (6, 10, 13, and 17 of 38)	M		(Note)	
T2	Timer_POLL	8.6; Figure 22 (25 of 38)	M		(Note)	
T3	Timer_KEEP-ALIVE	8.6; Figure 22 (25 of 38)	M		(Note)	
T4	Timer_NO-RESPONSE	8.6; Figure 22 (25 of 38)	M		(Note)	
T5	Timer_IDLE	8.6; Figure 22 (25 of 38)	M		(Note)	
T6	Timer_RESEQ	8.6; Figure 22 (25 of 38)	M		(Note)	
T7	Timer_GUARD	8.6; Figure 22 (2 of 38)	M		(Note)	
NOTE – This ITU-T Recommendation does not specify any allowed values.						

I.2.5.2 Parameters for data transfer

Item number	Item description	Ref.	Status	Support	Values	
					Allowed	Supported
Assured data transfer						
P1	Maximum number of transmissions of a BGN, END, ER, or RS PDU ("MaxCC")	8.7; Figure 22 (6, 10, 13, and 17 of 38)	MC4 M ELSE N/A		(Note 1)	
P2	Upper limit of transmitted SD PDUs before sending a POLL PDU ("MaxPD")	8.7; Figure 22 (30 of 38)	MC4 M ELSE N/A		(Note 1)	
P3	Maximum number of list elements placed in a STAT PDU ("MaxSTAT")	8.7; Figure 22 (35 of 38)	MC4 M ELSE N/A		(Note 1)	
P4	The maximum number of octets in the Information field of an SD PDU ("k")	8.2.4; 8.7	MC4 M ELSE N/A		0 … 65 528	
P5	The maximum number of octets in the SSCOP-UU field of a BGN, BGAK, BGREJ, END, or RS PDU ("j")	8.2.4; 8.7	MC4 M ELSE N/A		0 … 65 524	
Unassured data transfer						
P6	The maximum number of octets in the Information field of an UD PDU ("k")	8.2.4; 8.7.	MC5 M ELSE N/A		0 … 65 528	
P7	The maximum number of octets in the Information field of an MD PDU ("k")	8.2.4; 8.7.	MC6 M ELSE N/A		0 … 65 528	
Assured and unassured data transfer						
P8	The maximum number of simultaneously supported links ("MaxLinks")	I.2.4; (Note 2)	MC4 OR MC5 OR MC6 M ELSE N/A		(Note 1)	
NOTE 1 – This Recommendation does not specify either minimal required nor maximum allowed values. NOTE 2 – This Recommendation does not specify actions on attempted exceeding of this value.						

APPENDIX II

Examples of SSCOPMCE operation

II.1 Semantics of the list elements

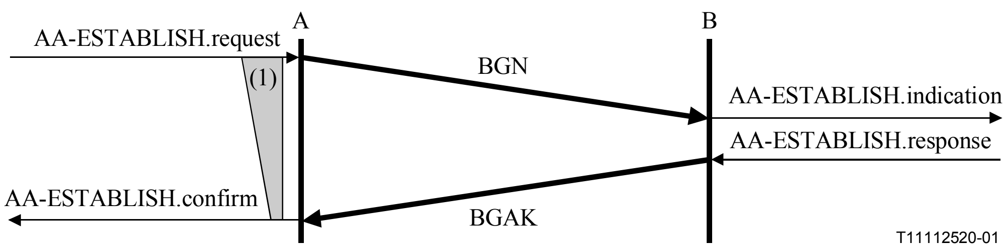
Table II.1 demonstrates the semantics of the list elements within a STAT and USTAT PDU.

Table II.1/Q.2111 – Examples of STAT and USTAT PDU semantics

RECEIVED SD PDUs	Received POLL PDU	Responding PDU
1,x,x,4	Missing sequence of SD PDUs detected	USTAT(N(R)=2,{2,4})
1,x,x,4	POLL(N(S)=5)	STAT(N(R)=2,{2,4,5})
1,x,x,x	POLL(N(S)=5)	STAT(N(R)=2,{2,5})
1,x,x,4,5	POLL(N(S)=6)	STAT(N(R)=2,{2,4,6})
1,x,x,4,5,x,x	POLL(N(S)=8)	STAT(N(R)=2,{2,4,6,8})
1,x,x,4,5,x,x,8,9	POLL(N(S)=10)	STAT(N(R)=2,{2,4,6,8,10})
NOTE 1 – Elements in "{ }" are STAT list elements. NOTE 2 – Only concerned fields are shown. NOTE 3 – x represents PDUs lost during transmission.		

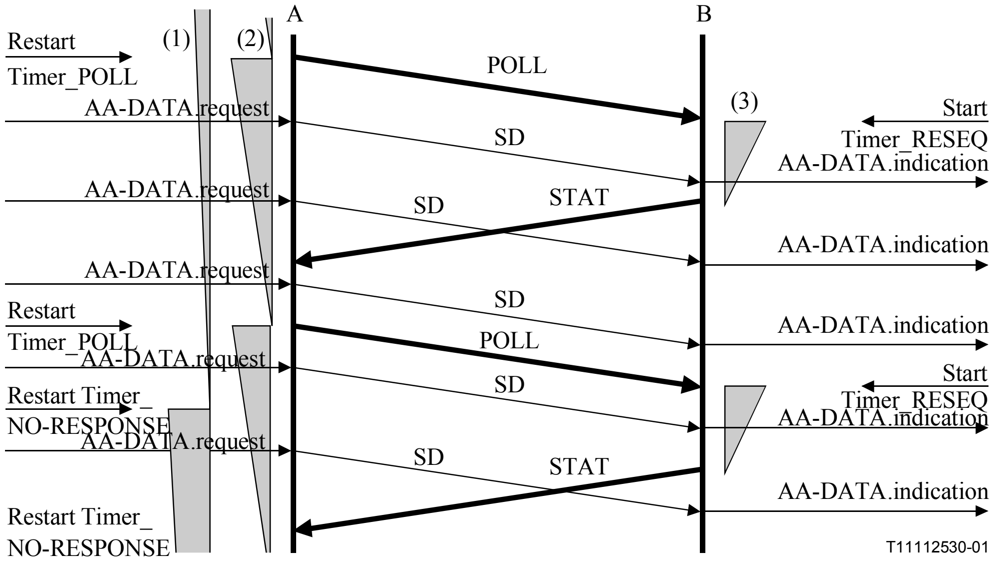
II.2 Error-free operation

Figures II.1 through II.4 show time flow diagrams of connection-establishment, data transfer, connection-release and resynchronization. All four time flow diagrams depict error-free operation, and are intended to provide a high-level appreciation of the protocol operation.



(1) Timer_CC

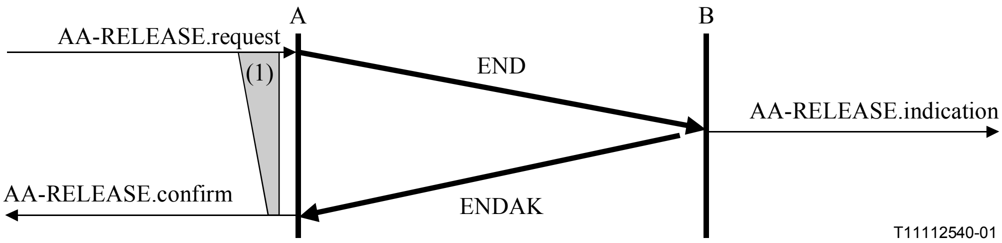
Figure II.1/Q.2111 – Time flow diagram for SSCOPMCE connection establishment



(1) Timer_NO-RESPONSE
(2) Timer_POLL
(3) Timer_RESEQ

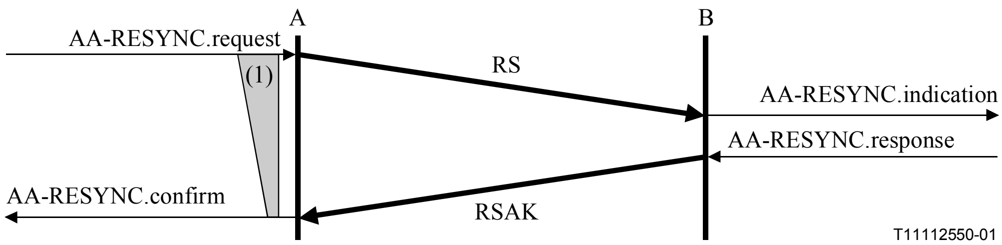
NOTE – Timer_NO-RESPONSE is only started if not all SD PDUs have been received and delivered; missing SD PDUs are not shown in this figure.

Figure II.2/Q.2111 – Time flow diagram for SSCOPMCE assured data transfer



(1) Timer_CC

Figure II.3/Q.2111 – Time flow diagram for SSCOPMCE connection release



(1) Timer_CC

Figure II.4/Q.2111 – Time flow diagram for SSCOPMCE connection resynchronization

II.3 Error recovery by STAT and USTAT PDUs

The following examples demonstrate the operation of the protocol under different scenarios. Each example illustrates a specific aspect of the protocol operation. The following conventions are used in the figures:

- The numbers shown at the transmitter represent SD sequence numbers, N(S), and the associated poll sequence numbers, N(PS), enclosed in parentheses.
- The numbers shown at the receiver represent the received SD PDU sequence numbers. An "X" in the delivered column indicates that an SD PDU has been delivered to the upper layer. An "X" in the Rx column represents a missing SD PDU.
- POLL PDU is represented as: POLL(N(S), N(PS)).
- STAT PDU is represented as: STAT(N(R), N(PS), N(MR), list elements).
- USTAT PDU is represented as: USTAT(N(R), N(MR), list elements).

Figure II.5 shows the operation of the protocol in the error-free case. SD PDUs are received in-sequence and delivered to the upper layer. Each SD PDU is not acknowledged separately, but a group of SD PDUs is acknowledged via a STAT PDU in response to a POLL PDU.

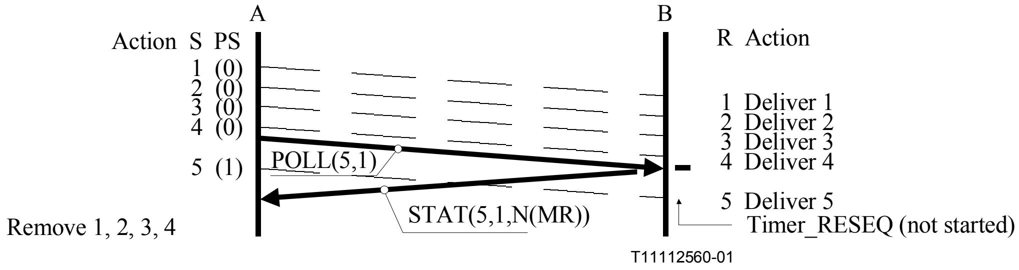
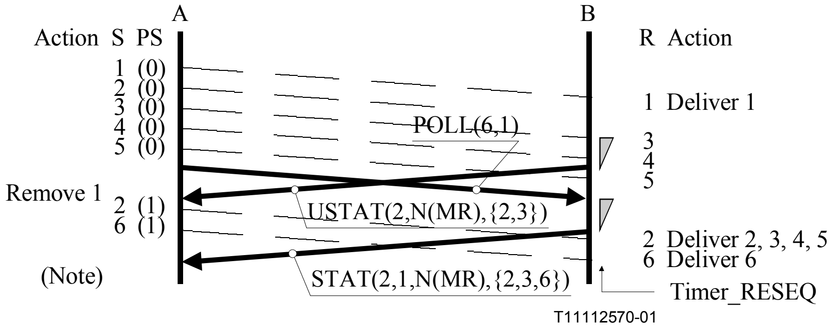


Figure II.5/Q.2111 – Operation in the error-free case

Figure II.6 shows error recovery via the USTAT PDU. Upon receiving SD PDU 3 and concluding that SD PDU 2 is missing, the receiver sends a USTAT PDU requesting retransmission of 2. Upon receiving this USTAT PDU, the transmitter retransmits SD PDU 2. Note that a subsequent STAT PDU is received, unnecessary retransmission is avoided due to the N(PS) comparison. The N(PS) associated with SD PDU 2 is 1, and it is not less than the N(PS) in the received STAT PDU(1); hence, 2 is not retransmitted. Upon receiving 2, the receiver delivers it and all subsequent in-sequence saved SD PDUs.



NOTE – The SD PDU with N(S) = 2 is not retransmitted another time as it has already been sent during this poll cycle.

Figure II.6/Q.2111 – Error recovery via USTAT PDU

Figure II.7 shows error recovery via STAT PDU. It demonstrates the case where an USTAT PDU is lost; however, recovery is still achieved via the STAT PDU.

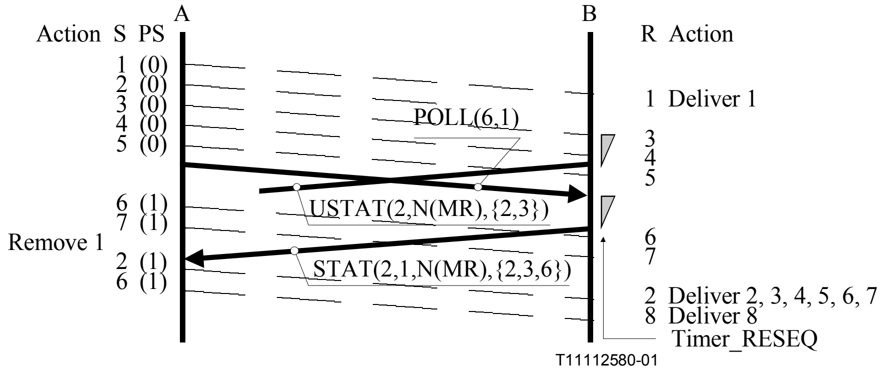


Figure II.7/Q.2111 – Error recovery via STAT PDU

Figure II.8 shows error recovery via STAT PDU of the last transmitted SD PDUs. It demonstrates the case where the last transmitted SD PDUs are all lost. In this case, an USTAT PDU cannot be generated, since the receiver has no knowledge that those SD PDUs have been transmitted and lost. However, error recovery is still achieved via the STAT PDU in response to the POLL PDU. Note that when SD PDU 5 is later on received, it does not cause a generation of a USTAT PDU.

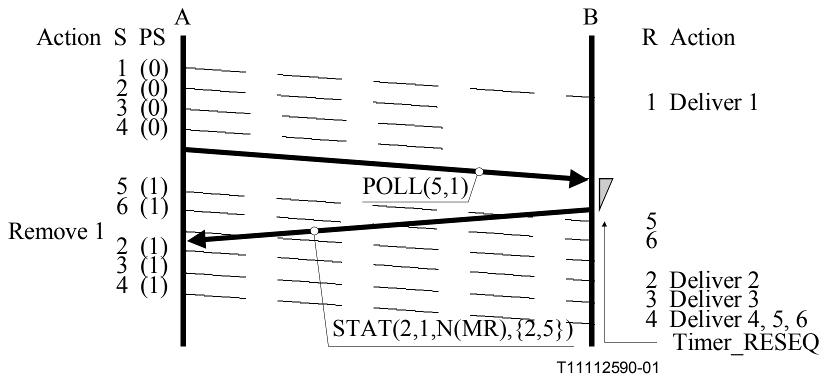


Figure II.8/Q.2111 – Error recovery via STAT PDU of the last transmitted SD PDUs

Figure II.9 shows error recovery via STAT and USTAT PDU of the last transmitted SD PDUs. The STAT is generated similar to the example in Figure II.8. However, subsequent SD PDUs not reported in this STAT and detected as missing when SD PDU 7 is received later on are recovered via the USTAT.

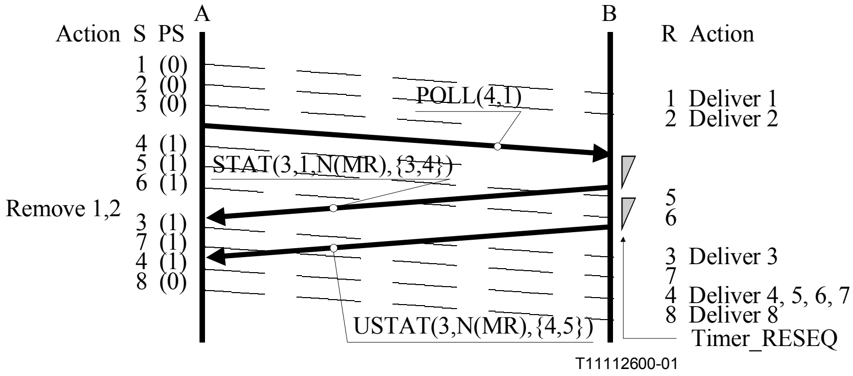


Figure II.9/Q.2111 – Error recovery via STAT and USTAT PDU of the last transmitted SD PDUs

Figure II.10 is similar to Figure II.9, but in this case the STAT PDU is lost. The example shows that the subsequent STAT PDU completes the error recovery.

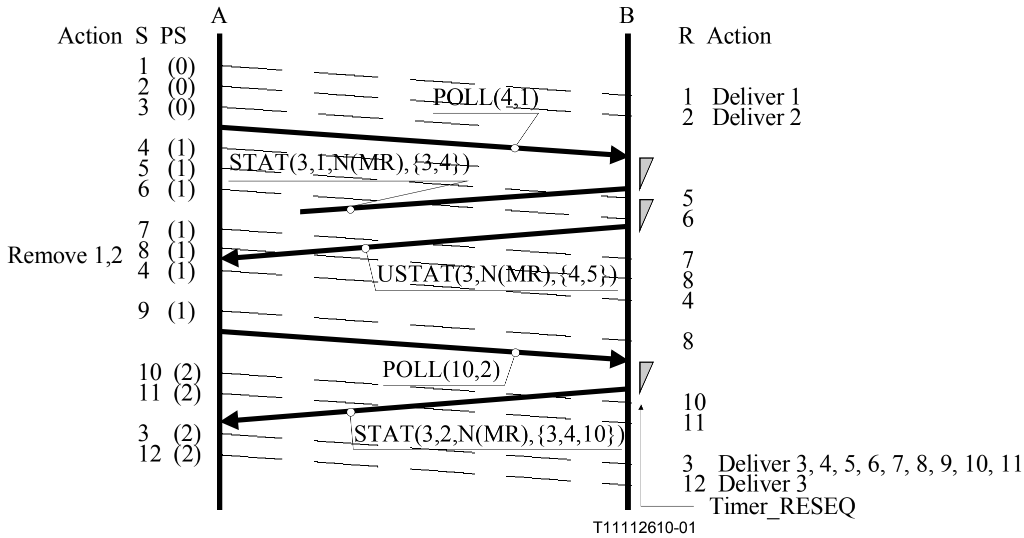
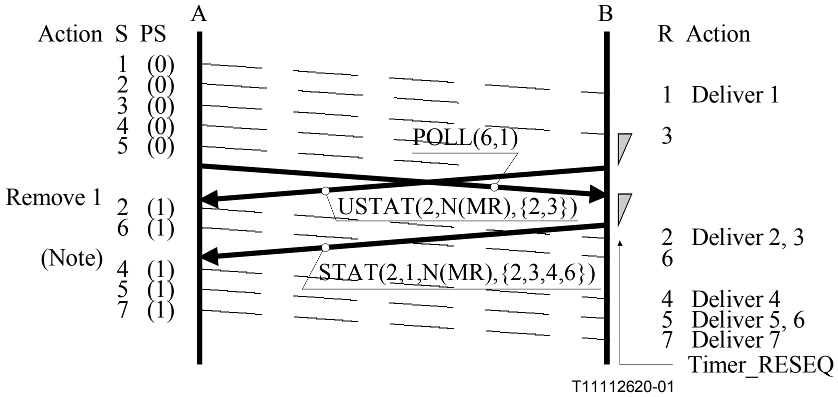


Figure II.10/Q.2111 – Error recovery via STAT and USTAT PDUs with loss of a STAT PDU

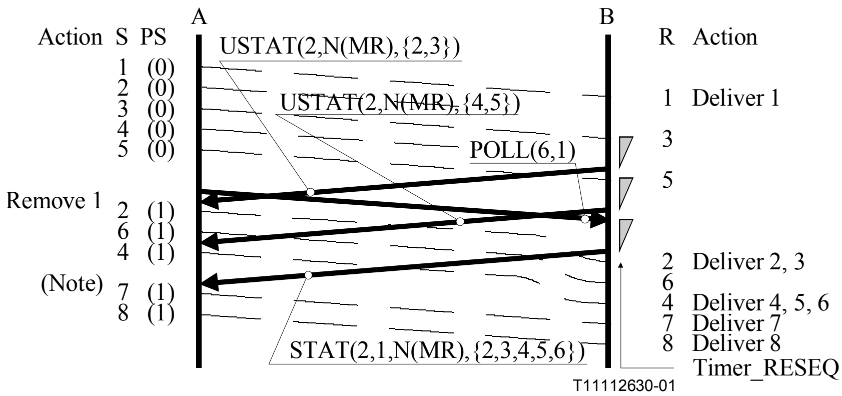
Figure II.11 is similar to Figure II.8; however, in this case four list elements are included in the STAT PDU indicating two missing SD sequences.



NOTE – The SD PDU with N(S) = 2 is not retransmitted another time as it has already been sent during this poll cycle.

Figure II.11/Q.2111 – Error recovery of two missing SD sequences via STAT and USTAT PDUs (last SD PDU missing)

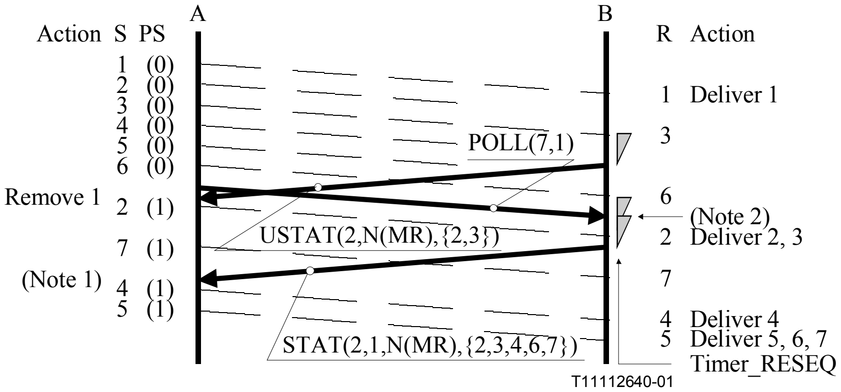
Figure II.12 is similar to Figure II.6; however, in this case five list elements are included in the STAT PDU indicating two missing SD sequences.



NOTE – The SD PDU with N(S) = 4 is not retransmitted another time as it has already been sent during this poll cycle.

Figure II.12/Q.2111 – Error recovery of two missing SD sequences via STAT and USTAT PDUs (last SD PDU not missing)

Figure II.13 is similar to Figure II.12. Also in this case four list elements are included in the STAT PDU indicating two missing SD sequences; however, after detecting the 2nd missing SD sequence Timer_RESEQ does not expire before receipt of a POLL PDU. The USTAT PDU is not sent and Timer_RESEQ is restarted.



NOTE 1 – The SD PDU with N(S) = 2 is not retransmitted another time as it has already been sent during this poll cycle.

NOTE 2 – Timer_RESEQ is restarted upon receipt of the POLL PDU; the USTAT PDU is not sent.

Figure II.13/Q.2111 – Error recovery of two missing SD sequences via STAT and USTAT PDUs (last SD PDU not missing)

Figure II.14 shows a successful recovery with a USTAT PDU alone; at the time the STAT PDU is generated, the missing SD PDU has already been retransmitted and need not be reported further.

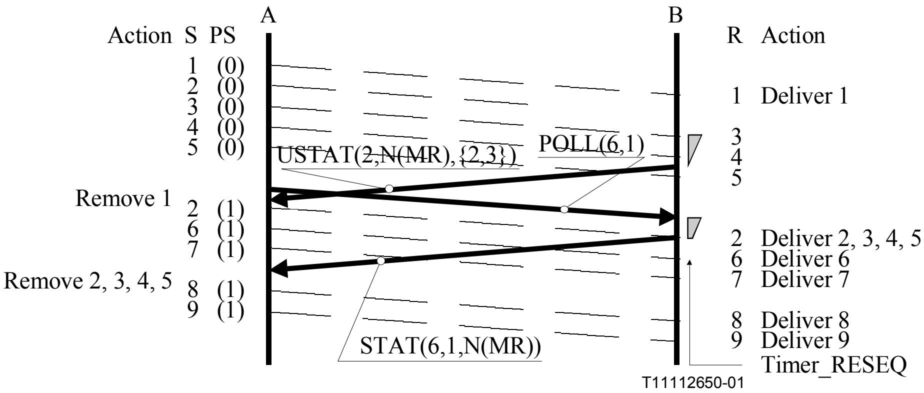


Figure II.14/Q.2111 – Error recovery of two missing SD sequences via STAT and USTAT PDUs (last SD PDU not missing)

II.4 Error recovery of PDUs received in a different order than transmitted

SSCOPMCE does not rely on the sequence integrity of the transport mechanism for its PDUs. This subclause illustrates examples, how SSCOPMCE operates with PDUs received out-of-sequence, i.e. in a different order than transmitted.

Figure II.15 shows the case where SD PDU with N(S) = 2 is received after the SD PDUs with N(S) = 3 and N(S) = 4. After receipt of SD PDU with N(S) = 3 the receiver detects that an SD PDU is missing and starts Timer_RESEQ. The SD PDUs witn N(S) = 3 and N(S) = 4 are stored in the receiver buffer. After receipt of SD PDU with N(S) = 2 no more SD PDU is detected to be missing. All SD PDUs can be delivered and Timer_RESEQ is stopped; no USTAT PDU is transmitted!

NOTE – Such reordering of SD PDUs can happen in various ways. Already a system with two equal links (in a multi-link situation) can lead to this situation in the following scenario. Assume that SD PDUs with

N(S) = 2, 3, and 4 are in the transmission queue. The SD PDU with N(S) = 2 is considerably longer than the other two. After submission of SD PDU with N(S) = 2 to the first link, SD PDU with N(S) = 3 can be submitted immediately to the 2nd link. After this short SD PDU is transmitted, the 2nd link becomes available to transmit the SD PDU with N(S) = 4 which is also short and is transmitted before all of SD PDU with N(S) = 2 has been fully transmitted.

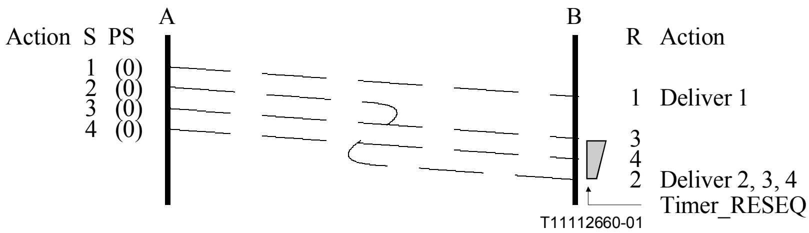


Figure II.15/Q.2111 – Recovery of out-of-sequence receipt of SD PDUs with Timer_RESEQ

Figure II.16 shows another case where SD PDUs are received in a different order than transmitted. SD PDUs with N(S) = 1 … 7 are transmitted in order. SD PDU with N(S) = 2 is received after the SD PDUs with N(S) = 6 and SD PDU with N(S) = 5 is received after the SD PDUs with N(S) = 7. The following actions take place:

a) After receipt of SD PDU with N(S) = 3 Timer_RESEQ is started.

b) After receipt of SD PDU with N(S) = 6 the current time is stored in the receiver buffer entry associated with SD PDU with N(S) = 5 (conceptually starting another Timer_RESEQ).

c) After receipt of SD PDU with N(S) = 2 Timer_RESEQ is restarted with the value equivalent to a starting time after receipt of SD PDU with N(S) = 6, SD PDUs with N(S) = 2, 3, and 4 are delivered.

d) After receipt of SD PDU with N(S) = 5 Timer_RESEQ is stopped; SD PDUs with N(S) = 5, 6, and 7 are delivered.

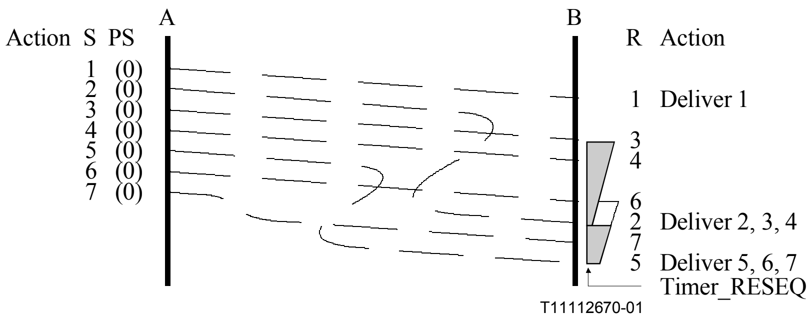


Figure II.16/Q.2111 – Recovery of out-of-sequence receipt of several SD PDUs with Timer_RESEQ

II.5 Early acknowledgment of SD PDUs

Figure II.17 shows a situation where further SD PDUs are received correctly while Timer_RESEQ is running after receipt of a POLL PDU. At expiry of this timer, the STAT PDU is constructed. At this time, SD PDU with N(S) = 5 has already been received and delivered. This is reflected in the N(R) field of the STAT PDU. Upon receipt of this STAT PDU the transmitter can free the transmission buffer of SD PDUs with N(S) = 1, 2, 3, 4 and as well 5.

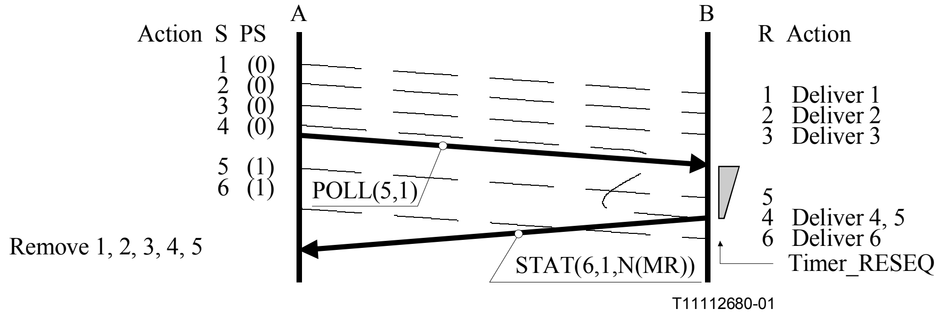


Figure II.17/Q.2111 – Acknowledgment of SD PDUs received after a POLL PDU but before Timer_RESEQ expires

Figure II.18 shows a similar situation as Figure II.16. In this case, however, an SD PDU sequence is detected missing while Timer_RESEQ is running. At expiry of this timer, the correctly received, re-sequenced, and delivered SD PDUs with N(S) = 5 and 6 are acknowledged as well. The sequence of missing SD PDUs, however, is not reported in the STAT PDU. The conceptual Timer_RESEQ that was started at receipt of SD PDUs with N(S) = 9 (reflected by a time stamp is stored in the receiver buffer entry associated with SD PDU with N(S) = 7 and 8) is still running; expiry of this timer needs to be awaited before reporting the missing sequence. After the STAT PDU is transmitted Timer_RESEQ is restarted as if it would have been started after receipt of SD PDUs with N(S) = 9.

NOTE – Figure II.17 also shows that if the missing sequence of SD PDUs with N(S) = 7 and 8 would have been reported in the STAT PDU, unnecessary retransmissions of these PDUs would have occurred.

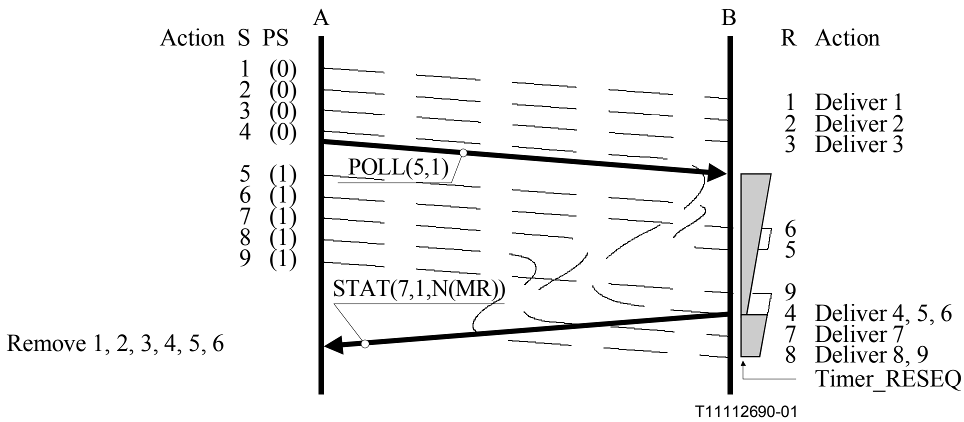


Figure II.18/Q.2111 – Acknowledgment of SD PDUs received after a POLL PDU with missing sequence of SD PDUs

APPENDIX III

Summary of buffer and state variable management

Table III.1 shows the status of the various buffers and the state variables at the time when a particular state is entered.

Table III.1/Q.2111 – Buffer and state variable management ***(Part 1 of 2)***

	Idle **1**	**Outgoing Connection Pending** **2**	**Incoming Connection Pending** **3**	**Outgoing Disconnection Pending** **4**	**Outgoing Resynchronization Pending** **5**
Clear Transmission Queue	C	U	C	C	U
Clear Transmission Buffer	C	U	C	C	U
Clear Retransmission Queue	U	U	U	U	U
Clear Receiver Buffer	U	U	U	U	U
Reset Receiver State Variables					
Reset Transmitter State Variables					
Data Retrieval Allowed	Y		Y	Y	

U Buffer/Queue is empty unconditionally upon entry into the state.

C Buffer/Queue is empty conditionally, i.e. if "Clear-Buffer = NO", upon entry into the state.

A The buffer is cleared unconditionally upon entry into the state, unless when entering from State 8 or 9 and if "Clear-Buffer = NO" (the data is subsequently transmitted).

D If "Clear-Buffer = NO", the content of the buffer is delivered with possible sequence gaps; if "Clear-Buffer = YES", the buffer is cleared upon entry into this state.

R Data transfer state variables are reset upon entry into the state.

Y Data retrieval allowed.

NOTE 1 – Data can be inserted into the Retransmission queue only in State 10, "Data Transfer Ready". As this buffer is cleared unconditionally upon entering any other state, by default, it is empty upon entry into State 10.

NOTE 2 – Data can be inserted into the Receiver buffer only in State 10, "Data Transfer Ready". As this buffer is cleared unconditionally before State 10 is entered from any other possible state, the Receiver buffer is empty unconditionally when entering State 10, by default, also.

Table III.1/Q.2111 – Buffer and state variable management ***(Part 2 of 2)***

	Incoming Resynchronization Pending **6**	**Outgoing Recovery Pending** **7**	**Recovery Response Pending** **8**	**Incoming Recovery Pending** **9**	**Data Transfer Ready** **10**
Clear Transmission Queue	C	C	C	C	A
Clear Transmission Buffer	C	C	C	C	U
Clear Retransmission Queue	U	U	U	U	(Note 1)
Clear Receiver Buffer	U		D	D	(Note 2)
Reset Receiver State Variables					R
Reset Transmitter State Variables					R
Data Retrieval Allowed	Y		Y	Y	

U Buffer/Queue is empty unconditionally upon entry into the state.

C Buffer/Queue is empty conditionally, i.e. if "Clear-Buffer = NO", upon entry into the state.

A The buffer is cleared unconditionally upon entry into the state, unless when entering from State 8 or 9 and if "Clear-Buffer = NO" (the data is subsequently transmitted).

D If "Clear-Buffer = NO", the content of the buffer is delivered with possible sequence gaps; if "Clear-Buffer = YES", the buffer is cleared upon entry into this state.

R Data transfer state variables are reset upon entry into the state.

Y Data retrieval allowed.

NOTE 1 – Data can be inserted into the Retransmission queue only in State 10, "Data Transfer Ready". As this buffer is cleared unconditionally upon entering any other state, by default, it is empty upon entry into State 10.

NOTE 2 – Data can be inserted into the Receiver buffer only in State 10, "Data Transfer Ready". As this buffer is cleared unconditionally before State 10 is entered from any other possible state, the Receiver buffer is empty unconditionally when entering State 10, by default, also.

APPENDIX IV

Default window size for SSCOP

This appendix may be used to set the window size parameter (conveyed in the SSCOP N(MR) field) of SSCOP. The following formula can be used to calculate a window size that is sufficient to keep the transmitter active. Alternatively, the window can be optimized to the particular connection or implementation. For example, the window size may be reduced to perform flow control or buffer management. During a connection, the window size can change dynamically based on local requirements.

$$k = 2 + (2 * Timer\_POLL + 6 * Ttd) * Ru/(8 * Ld)$$

where:

k	is the window size
Ttd	is the end-to-end transit delay (seconds)
Timer_POLL	is the POLL timer value for the peer entity (seconds)
Ru	is the SSCOP throughput (bits/s)
Ld	is the data frame size in octets

Information regarding the end-to-end transit delay, throughput, and frame size should be available at the SSCOP endpoints, or may be derived from signalling messages. The Timer_POLL used by the peer can be identified based on the received POLL PDU frequency; alternatively, the selected Timer_POLL value used at the local transmitter can be used.

The values of the Timer_POLL and the round trip delay have implications on the size of buffers needed to support the connection. If the window size implies that excessive buffers are needed for the connection, an implementation may consider shortening the Timer_POLL value at the transmitter, or decoupling the receiver buffer from the offered window at the receiver.

The window passed to the transmitter is conveyed by a sequence number in the N(MR) field of certain SSCOP PDUs. The difference between this sequence number (VR(MR)) and the next in-sequence to be received (VR(R)) is the window at the receiver.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	**Switching and signalling**
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communications
Series Y	Global information infrastructure and Internet protocol aspects
Series Z	Languages and general software aspects for telecommunication systems

Geneva, 2001