INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.633

Addendum 1

(09/98)

SERIES X: DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS

OSI networking and system aspects – Efficiency

Information technology – Open Systems Interconnection – Network Fast Byte Protocol

Addendum 1: SDL specifications

ITU-T Recommendation X.633 – Addendum 1

(Previously CCITT Recommendation)

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS

PUBLIC DATA NETWORKS	
Services and facilities	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalling and switching	X.50–X.89
Network aspects	X.90–X.149
Maintenance	X.150–X.179
Administrative arrangements	X.180–X.199
OPEN SYSTEMS INTERCONNECTION	
Model and notation	X.200–X.209
Service definitions	X.210–X.219
Connection-mode protocol specifications	X.220–X.229
Connectionless-mode protocol specifications	X.230–X.239
PICS proformas	X.240–X.259
Protocol Identification	X.260–X.269
Security Protocols	X.270–X.279
Layer Managed Objects	X.280–X.289
Conformance testing	X.290–X.299
INTERWORKING BETWEEN NETWORKS	
General	X.300–X.349
Satellite data transmission systems	X.350–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	
Networking	X.600–X.629
Efficiency	**X.630–X.639**
Quality of service	X.640–X.649
Naming, Addressing and Registration	X.650–X.679
Abstract Syntax Notation One (ASN.1)	X.680–X.699
OSI MANAGEMENT	
Systems Management framework and architecture	X.700–X.709
Management Communication Service and Protocol	X.710–X.719
Structure of Management Information	X.720–X.729
Management functions and ODMA functions	X.730–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	
Commitment, Concurrency and Recovery	X.850–X.859
Transaction processing	X.860–X.879
Remote operations	X.880–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999

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

ITU-T RECOMMENDATION X.633

INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – NETWORK FAST BYTE PROTOCOL

ADDENDUM 1

SDL specifications

Summary

This Addendum to Recommendation X.633 contains an SDL specification of the Network Fast Byte Protocol description. The Network Fast Byte Protocol applies to the provision of the OSI Connection-mode Network Service in end systems, and eliminates the roundtrip delay associated with the establishment and release of a network connection, and requires very low PCI overhead. The Network Fast Byte Protocol is intended for use in situations in which enhancements to the data link QoS are not required, and efficiency of operation (e.g. reduction of roundtrip delays on establishment and release) is of primary concern.

Source

Addendum 1 to ITU-T Recommendation X.633, was prepared by ITU-T Study Group 7 (1997-2000) and was approved under the WTSC Resolution No. 1 procedure on the 25th of september 1998.

FOREWORD

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

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

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1.

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 term *recognized operating agency (ROA)* includes any individual, company, corporation or governmental organization that operates a public correspondence service. The terms *Administration*, *ROA* and *public correspondence* are defined in the *Constitution of the ITU (Geneva, 1992)*.

INTELLECTUAL PROPERTY RIGHTS

The ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. The 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, the 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.

TABLE OF CONTENTS

Addendum 1 to Recommendation X.633

INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – NETWORK FAST BYTE PROTOCOL

ADDENDUM 1

SDL specifications

(Geneva, 1998)

Annex C

SDL specification of the Network Fast Byte Protocol entity

C.1 The system and block structure of the Network Fast Byte Protocol entity

The SDL system diagram of the Network Fast Byte Protocol entity is shown in Figure C.1 and the SDL block structure in Figure C.2.

C.2 Interaction with Management and Control Planes

C.2.1 Management Plane

The data transfer with the Q-bit set is modeled as a data transfer between the Layer Management entities of the Network Fast Byte Protocol entities. The service primitives are defined as:

- MN-DATA.request (from Layer Management entity to Network Fast Byte Protocol entity); and
- MN-DATA.indication (from Network Fast Byte Protocol entity to Layer Management entity).

The parameters of these service primitives are identical to the N-DATA.request and N-DATA.indication primitives.

No other interactions with the Management Plane are specified.

C.2.2 Control Plane

Interaction between the Network Fast Byte Protocol entity and the Control Plane (C-plane) is outside the scope of this annex.

C.3 Procedure of the Network Fast Byte Protocol entity

The SDL diagrams of the procedure of the Network Fast Byte Protocol entity are given in this subclause. If there exists any difference between the prose description also given in this subclause and the SDL diagrams, the SDL diagrams take precedence. On the other hand, if there exists differences between the specification in this annex and the one in clause 6/X.633, the specification in clause 6/X.633 takes precedence.

NOTE – In the SDL diagrams of this subclause, the octets in all PDUs and SDUs, i.e. the TSDU, are numbered from "1" to at most "65535".

The operation of the Network Fast Byte Protocol entity is modeled as a state machine consisting of the following states:

– **Idle:** Each Network Fast Byte Protocol entity is conceptually initiated in the Idle state and returns to this state upon the release of a connection.

– **Outgoing Connection Pending**: A Network Fast Byte Protocol entity requesting a connection with its peer is in the Outgoing Connection Pending state until it receives acknowledgment from its peer.

– **Incoming Connection Pending**: A Network Fast Byte Protocol 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.

– **Outgoing Resynchronization Pending**: A Network Fast Byte Protocol entity requesting resynchronization (RESET) of the connection with its peer is in the Outgoing Resynchronization Pending state.

– **Incoming Resynchronization Pending**: A Network Fast Byte Protocol entity that has received a resynchronization request (RESET) from its peer and is waiting for its user's response is in the Incoming Resynchronization Pending state.

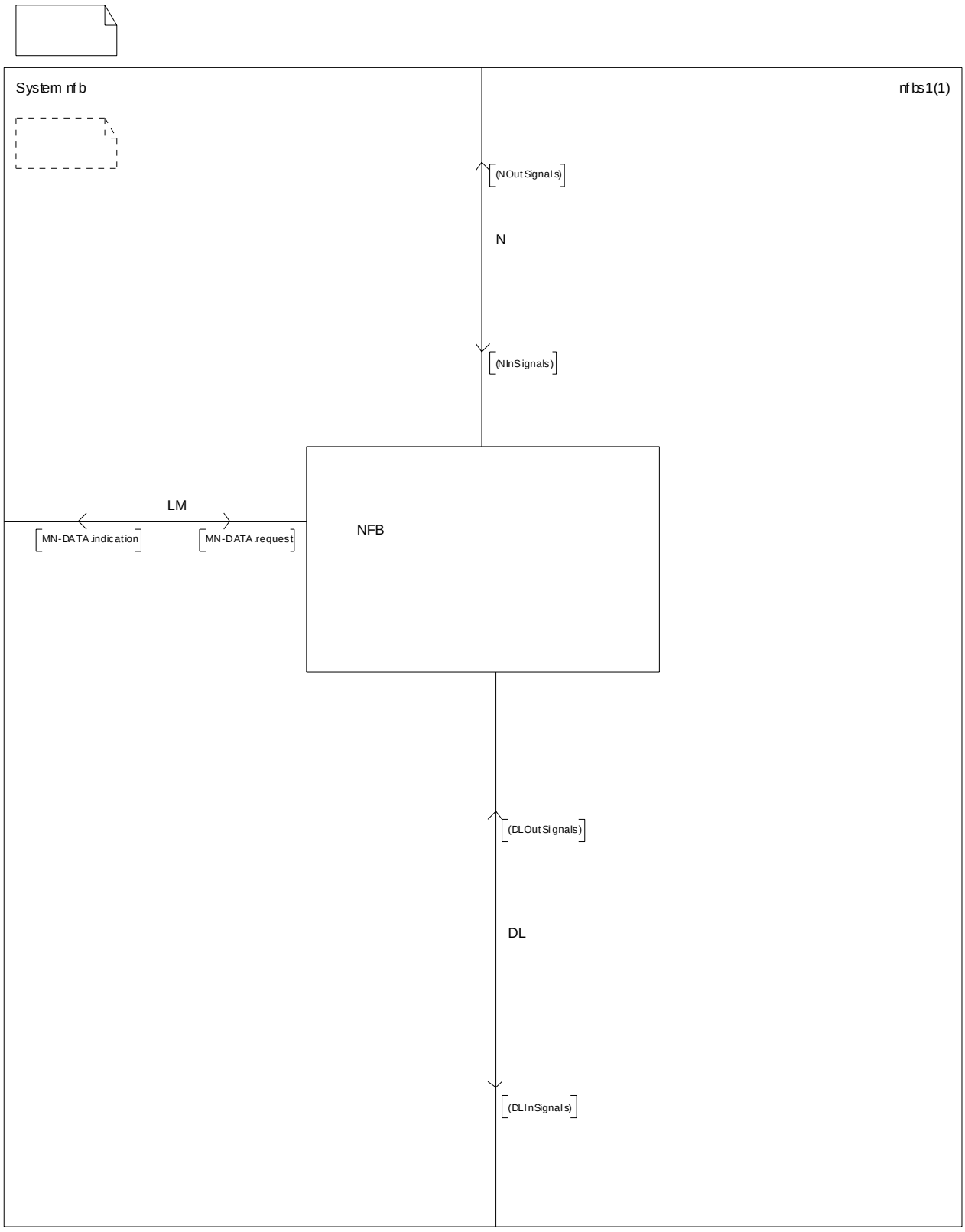


Figure C.1/X.633 – Network Fast Byte system

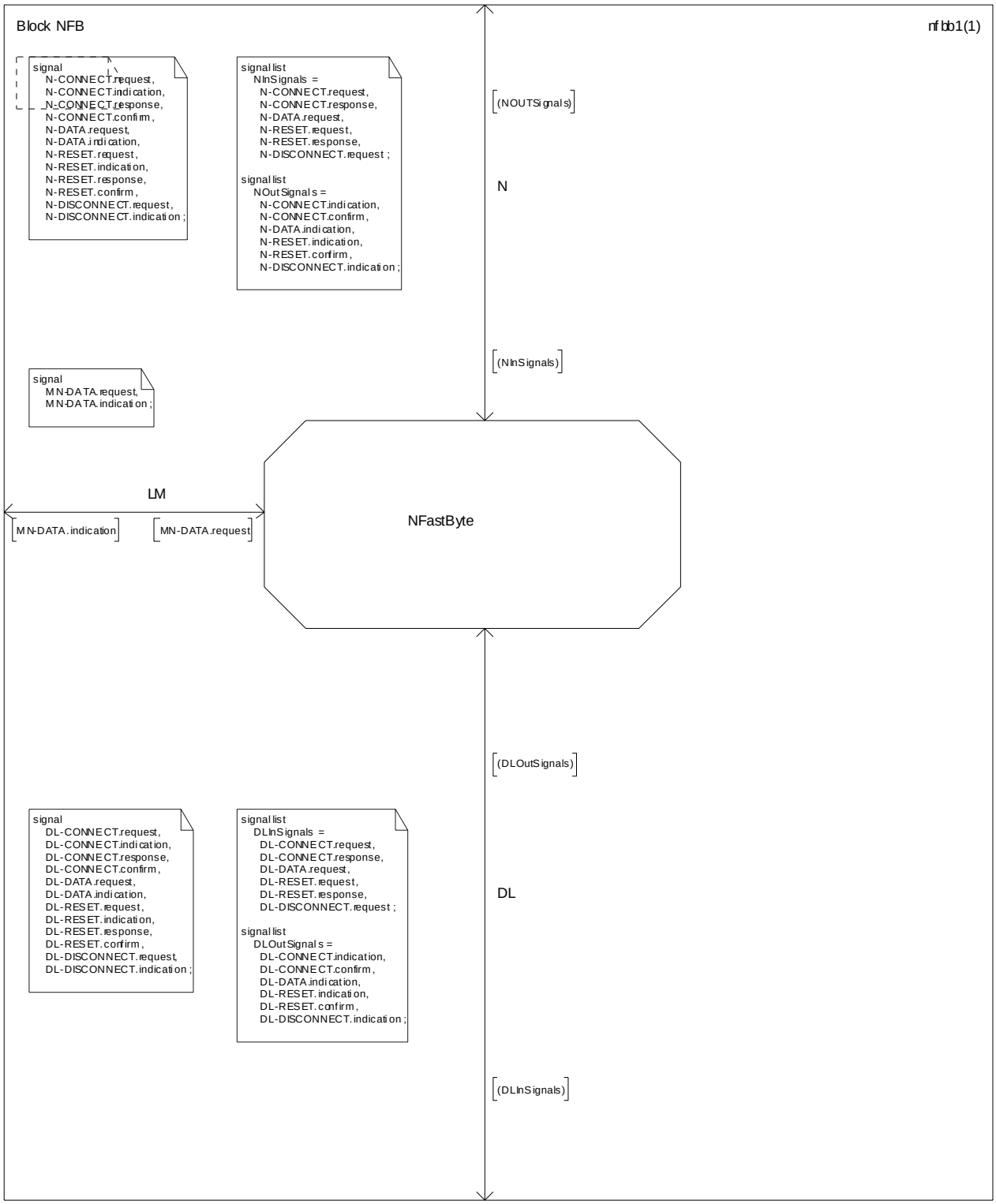


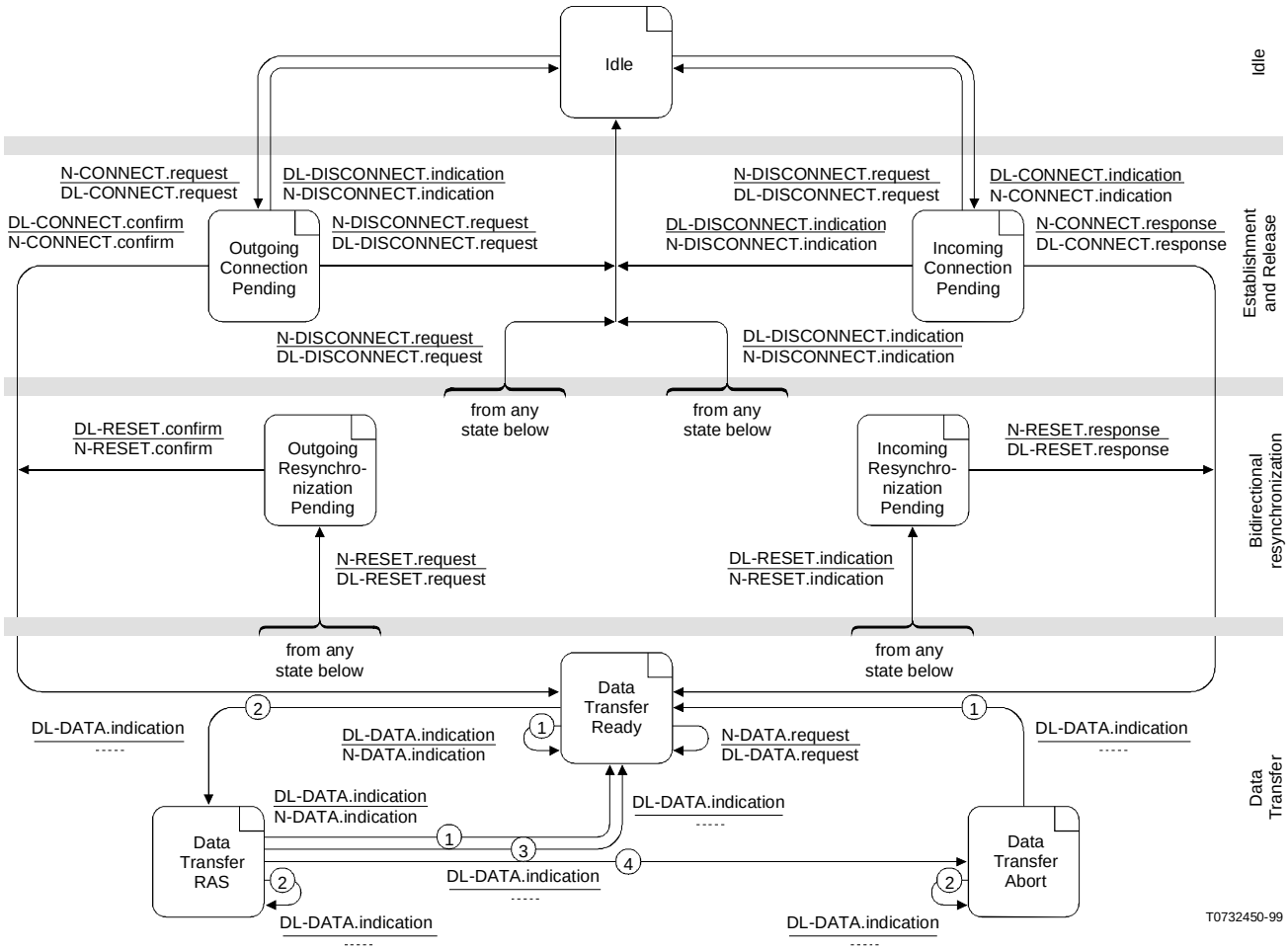
Figure C.2/X.633 – Network Fast Byte Block structure

– **Data Transfer Ready**: Upon successful completion of the connection establishment, or resynchronization procedures, both peer Network Fast Byte Protocol entities will be in Data Transfer Ready state and data transfer can take place. No NSDU is currently reassembled.

– **Data Transfer RAS**: Some but not all information for an NSDU currently being reassembled has arrived and is being buffered.

– **Data Transfer Abort**: The Q-bit of the different segments of the NSDU are not equal or the maximum permissible length of an NSDU has been exceeded during reassembly. The Network Fast Byte Protocol entity remains in this state until an end of an NSDU is received (EON-bit).

The state transition diagram for the Network Fast Byte Protocol entity is shown in Figure C.3.



1) DL-DATA.indication received and end of the NSDU detected.
2) DL-DATA.indication received and no end of the NSDU detected.
3) ConnRcvNPDULength exceeded or Q-bit change and end of the NSDU detected.
4) ConnRcvNPDULength exceeded or Q-bit change and no end of the NSDU detected.

Figure C.3/X.633 – State transition diagram for the Network Fast Byte Protocol entity

The description of the operations of the Network Fast Byte Protocol entity make use of the following state variables:

CdAddr	A character string holding the "Called Address" as extracted out of an NPDU.
CgAddr	A character string holding the "Calling Address" as extracted out of an NPDU.
RsAddr	A character string holding the "Responding Address" as extracted out of an NPDU.
NPDU	An octet string holding the NPDU being constructed in the outgoing direction or the NPDU just having been received.

NSDU	An octet string holding the NSDU being reassembled in the incoming direction.
ptrRAS	An index into the variable NSDU indicating where to place the next information during reassembly.
ptrSEG	An index into the parameter NSUserData indicating where to extract data during segmentation.
ptrPDU	An index into the variable NPDU having been received indicating where to retrieve the next information during interpretation of the received NPDU.
lenPDU	The length of an NPDU having been received.
lenSDU	The (remaining) length of an NSDU being segmented.
len	Temporary variable holding the length of a character string.
EONbit	A boolean variable being set to TRUE if the EON-bit in the received NPDU has been set.
SndNPDULength	A temporary integer variable during connection establishment used for negotiation of the maximum size of an NPDU in the outgoing direction.
ConnSndNPDULength	An integer variable indicating the maximum size of an NPDU in the outgoing direction.
RcvNPDULength	A temporary integer variable during connection establishment used for negotiation of the maximum size of an NPDU in the incoming direction.
ConnRcvNPDULength	An integer variable indicating the maximum size of an NPDU in the incoming direction.
tmpNullPCI	A temporary boolean variable during connection establishment used for negotiation of the NullPCI capability.
ConnSndNullPCI	A boolean variable indicating the NullPCI capability in the outgoing direction.
ConnRcvNullPCI	A boolean variable indicating the NullPCI capability in the incoming direction.
currQbit	A boolean variable being set to TRUE if the Q-bit in the received NPDU has been set.
ConnQbit	A boolean variable being set to TRUE if Layer Management data transfer (the use of the Q-bit) is allowed on this connection.
Terminate	A boolean variable being set to TRUE after leaving state "Idle" if the process should stop after return to "Idle".

The Network Fast Byte Protocol entity maintains the following parameters:

DLmaxLength	The maximum size of the DLSUserData parameter.
NmaxRcvLength	The maximum size of the NSUserData parameter in the incoming direction.
NmaxSndLength	The maximum size of the NSUserData parameter in the outgoing direction.
QbitAllowed	A boolean value being set to TRUE if Layer Management data transfer (the use of the Q-bit) is enabled.
CPlaneNegotiation	A boolean value being set to TRUE if the parameter negotiation takes place outside the Network Fast Byte Protocol entity, e.g. in the C-plane.
NullPCI	A boolean value being set to TRUE if NullPCI procedures are enabled.
Termination	A boolean value being set to TRUE if if the Network Fast Byte Protocol entity should stop after return to the state "Idle".

The SDL definition of the Network Fast Byte Protocol entity process is shown in Figure C.4.

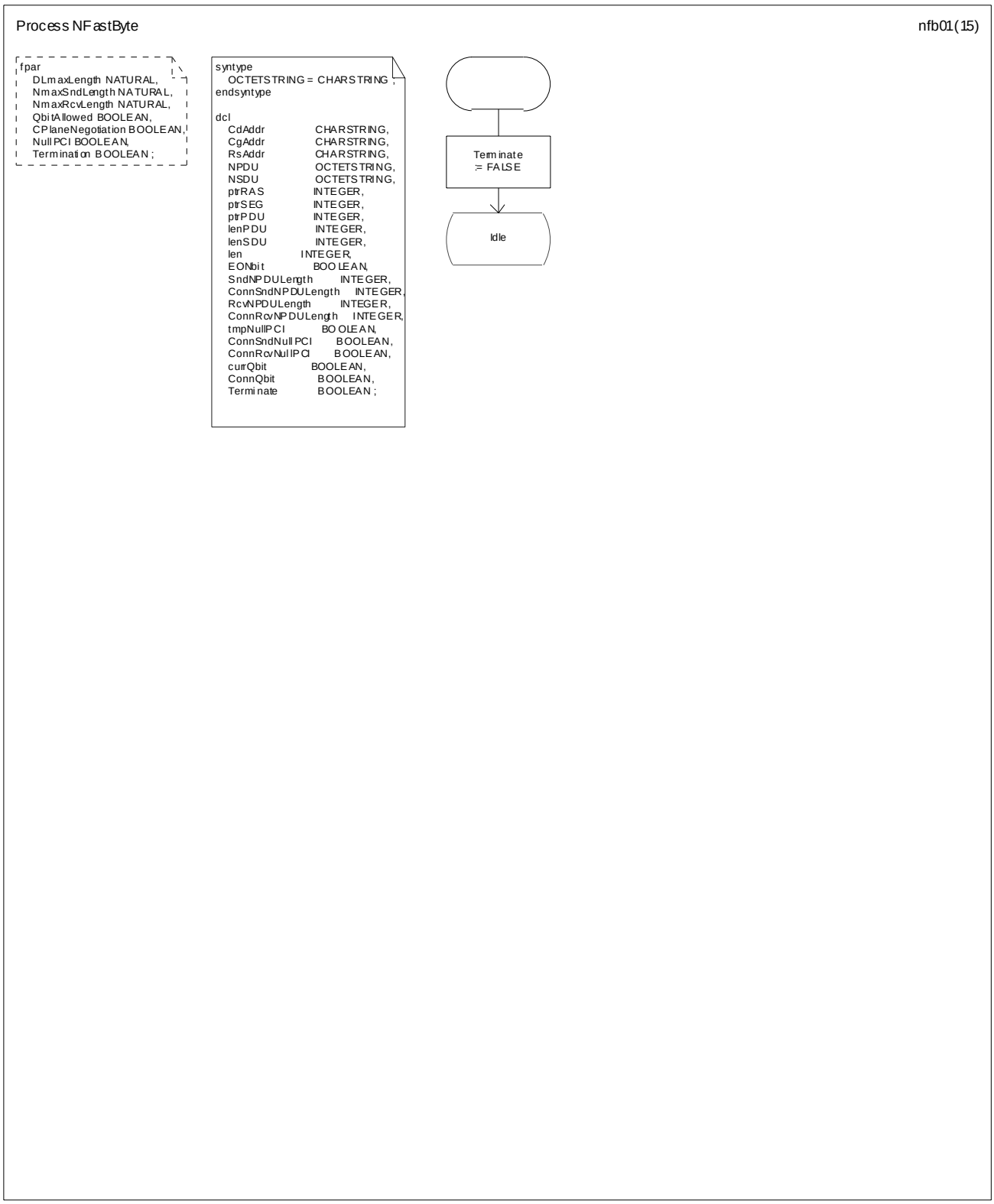


Figure C.4/X.633 (Sheet 1/15) – Network Fast Byte process

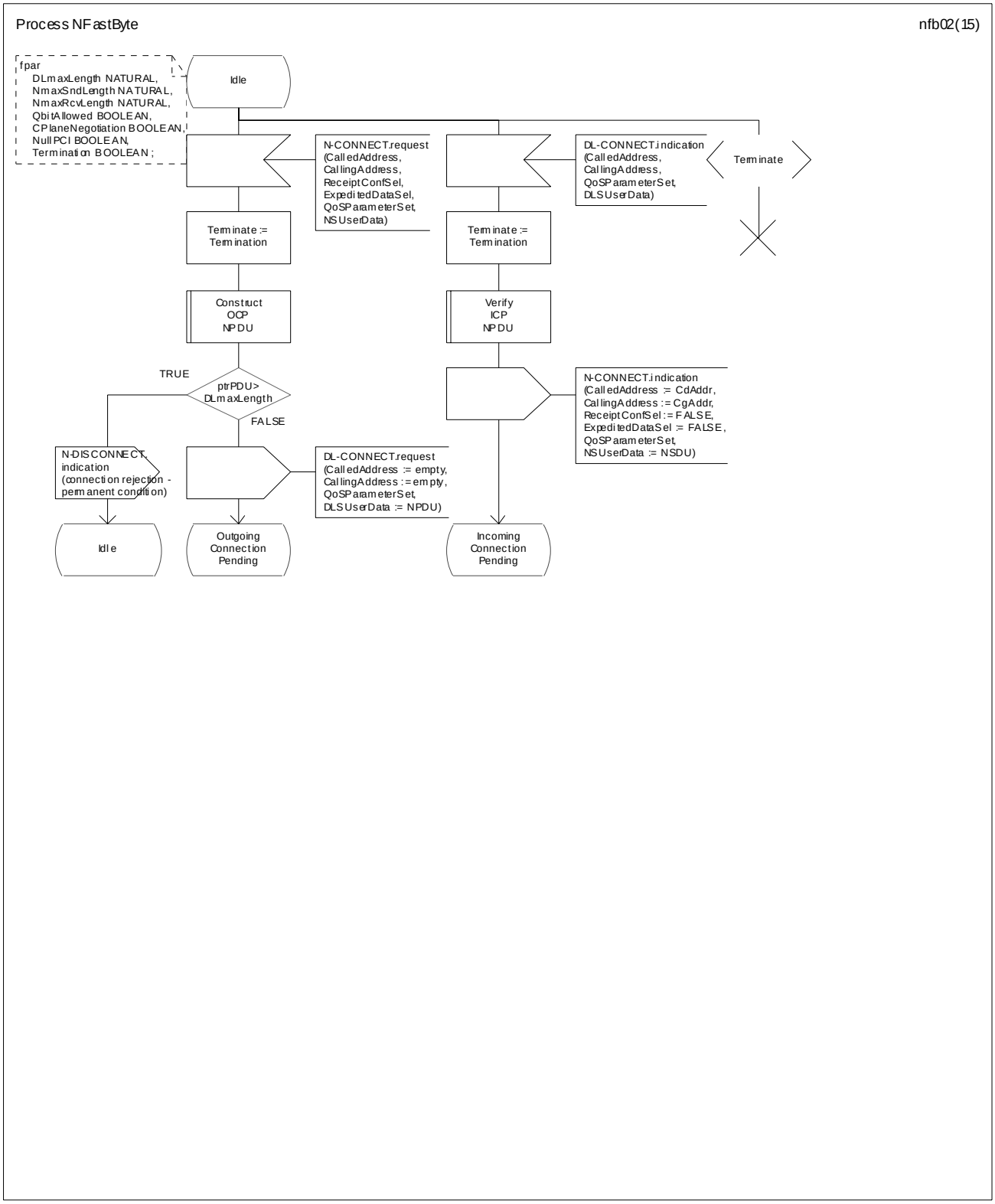


Figure C.4/X.633 (Sheet 2/15) – Network Fast Byte process

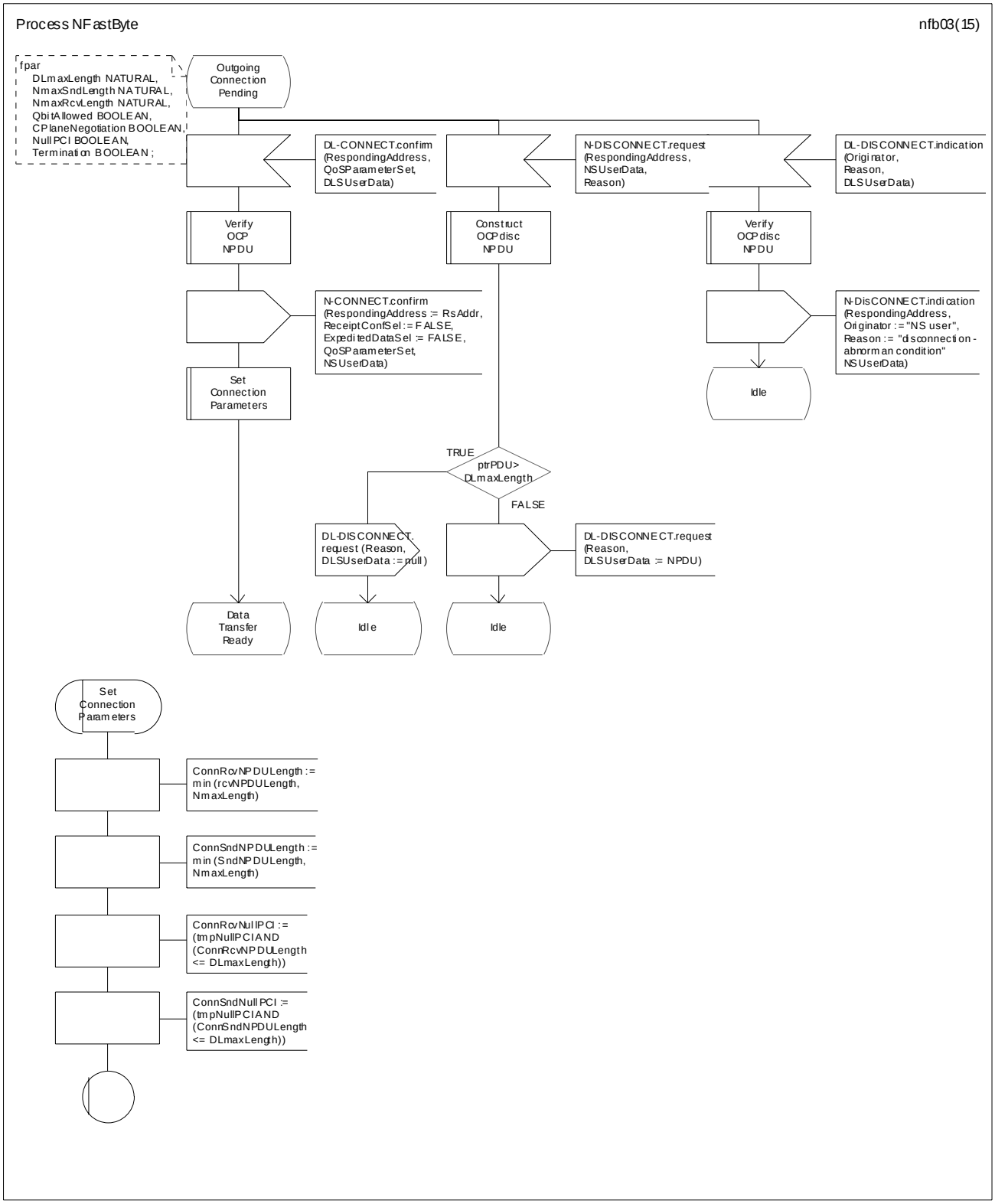


Figure C.4/X.633 (Sheet 3/15) – Network Fast Byte process

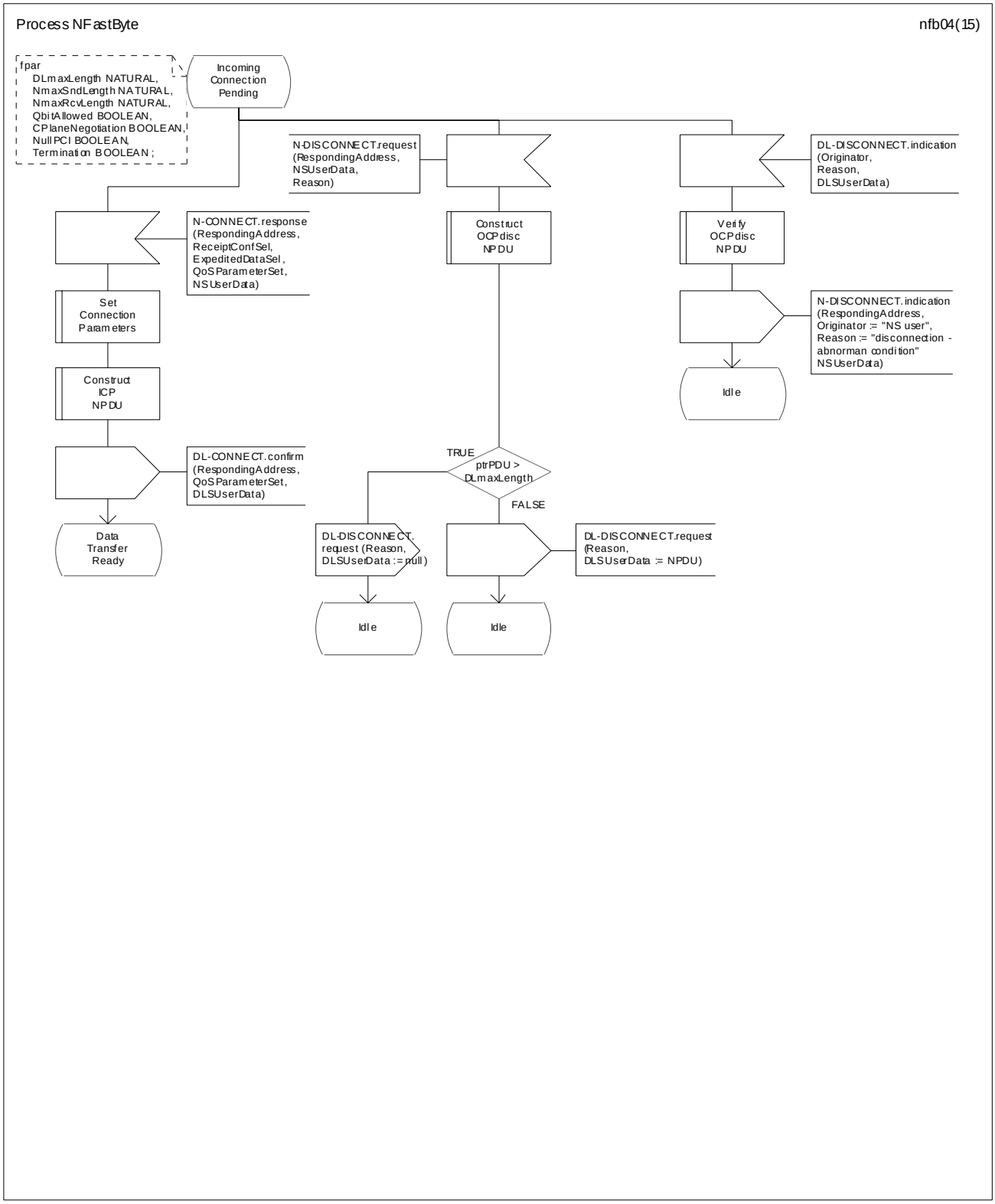


Figure C.4/X.633 (Sheet 4/15) – Network Fast Byte process

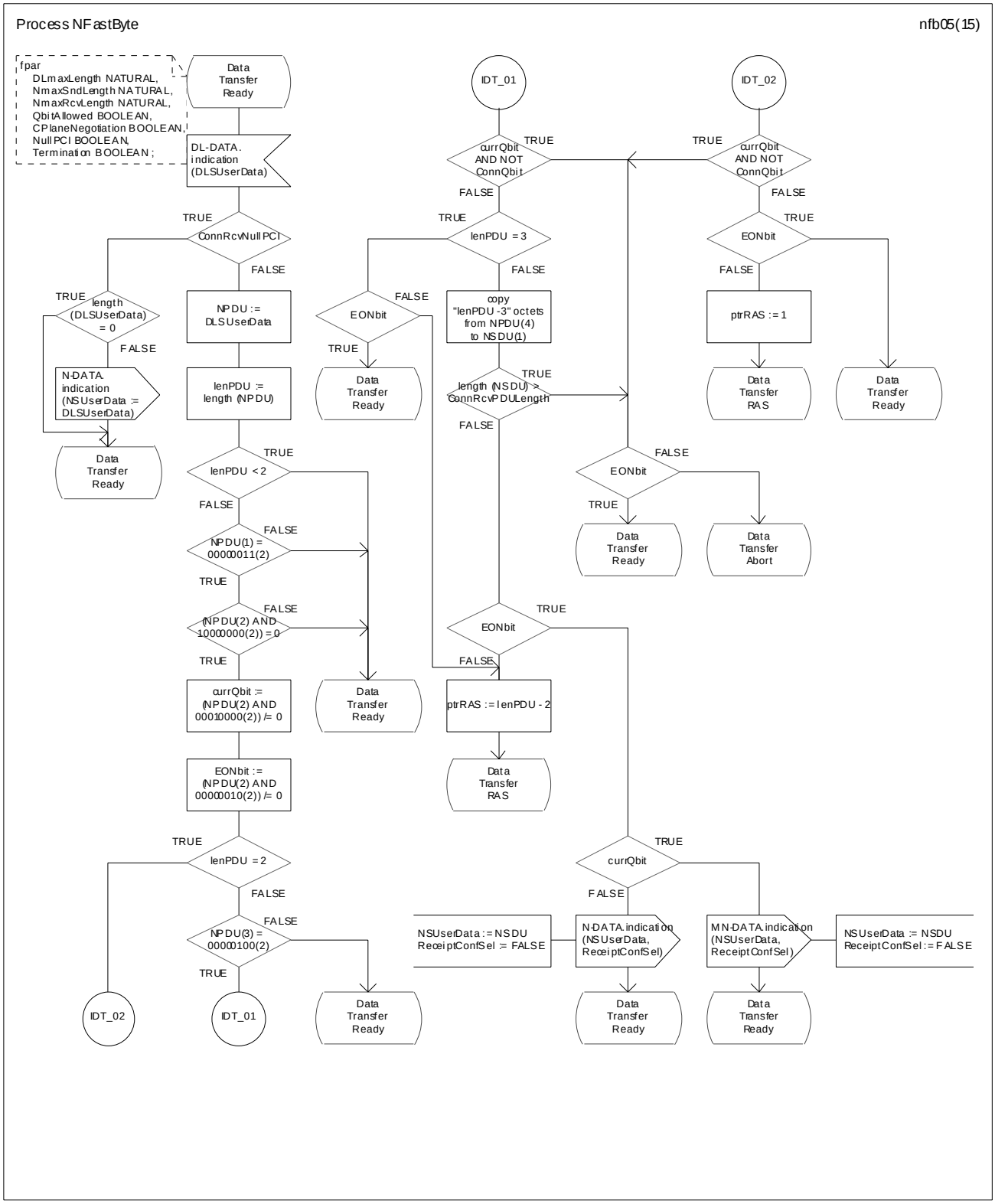


Figure C.4/X.633 (Sheet 5/15) – Network Fast Byte process

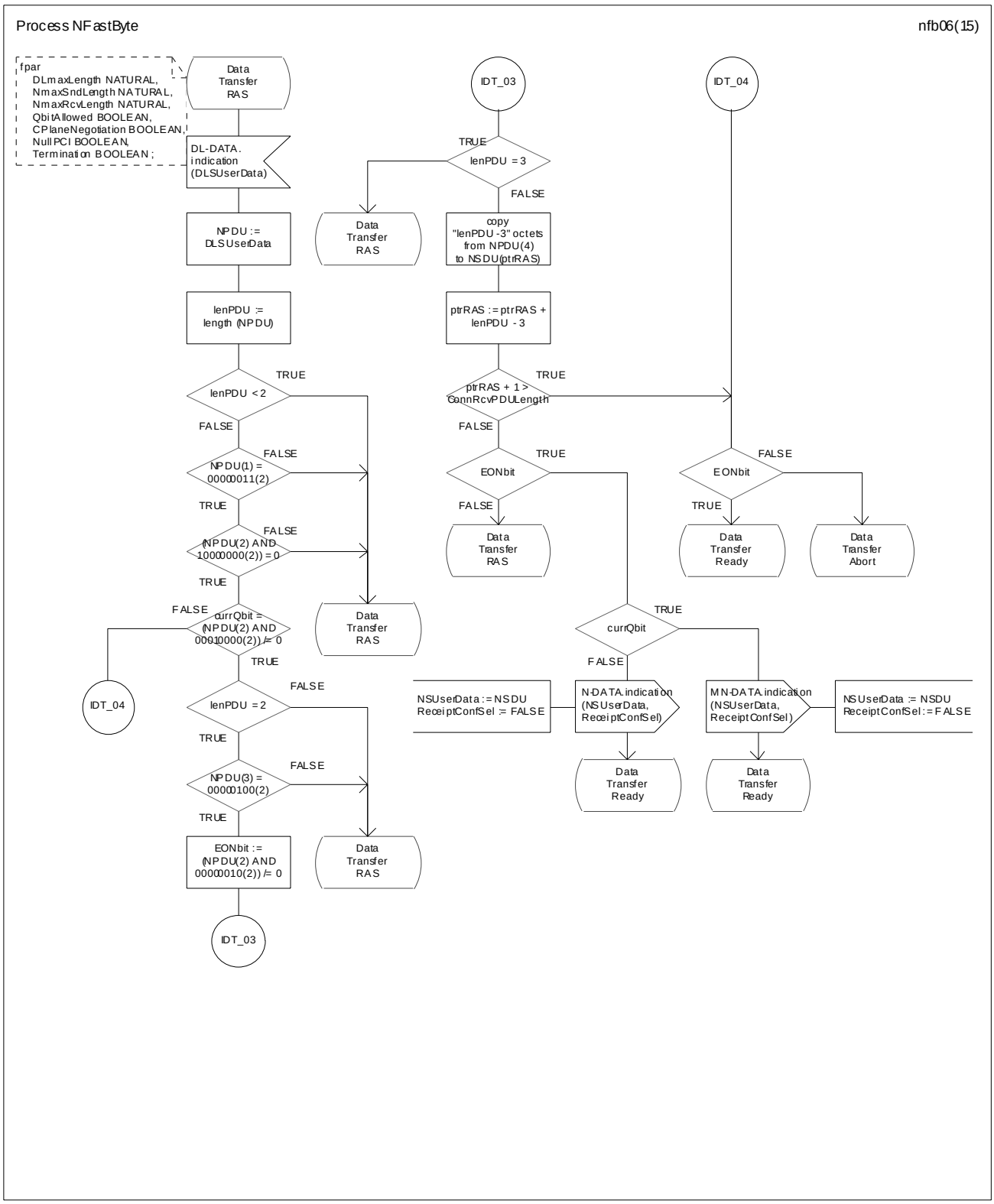


Figure C.4/X.633 (Sheet 6/15) – Network Fast Byte process

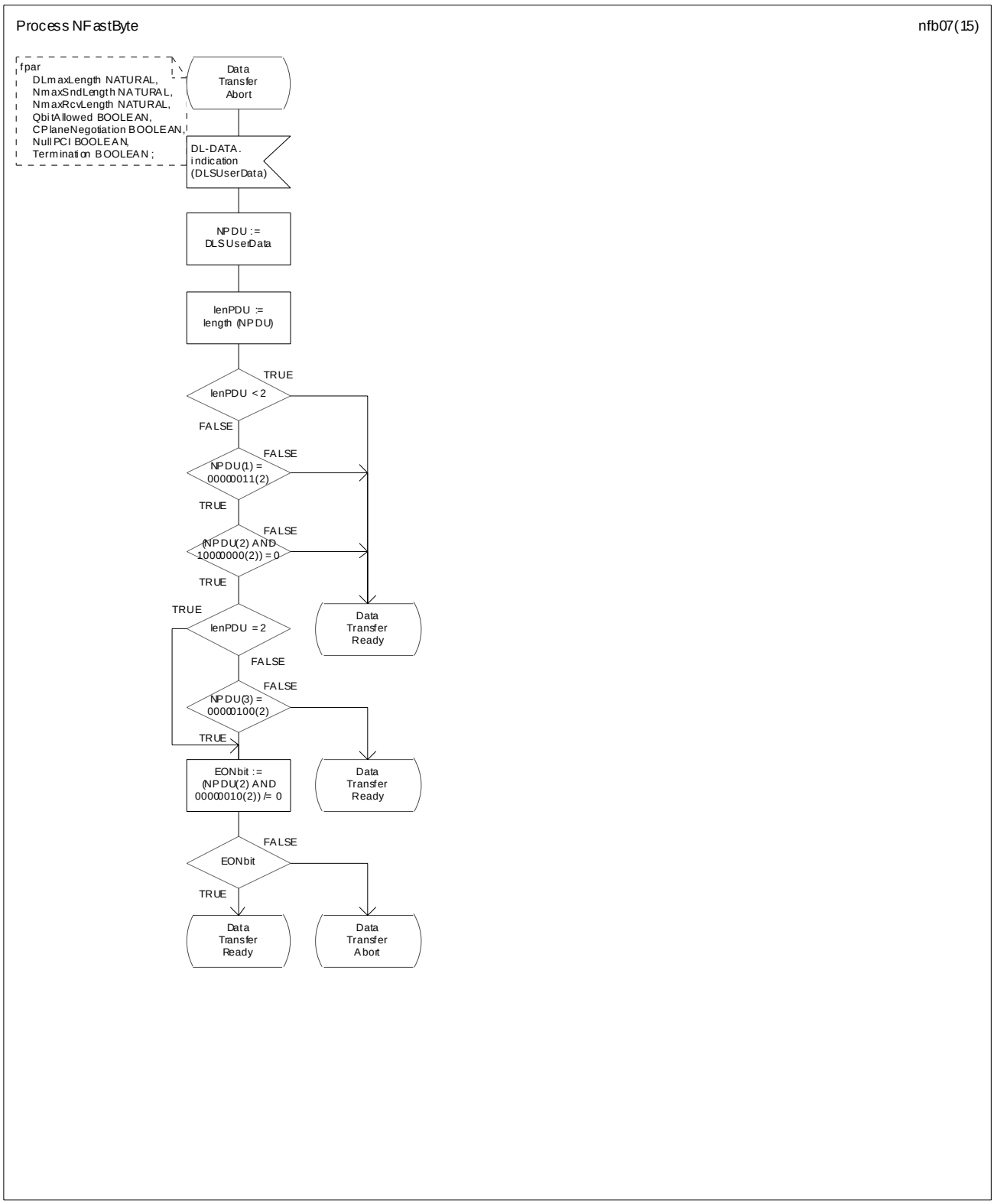


Figure C.4/X.633 (Sheet 7/15) – Network Fast Byte process

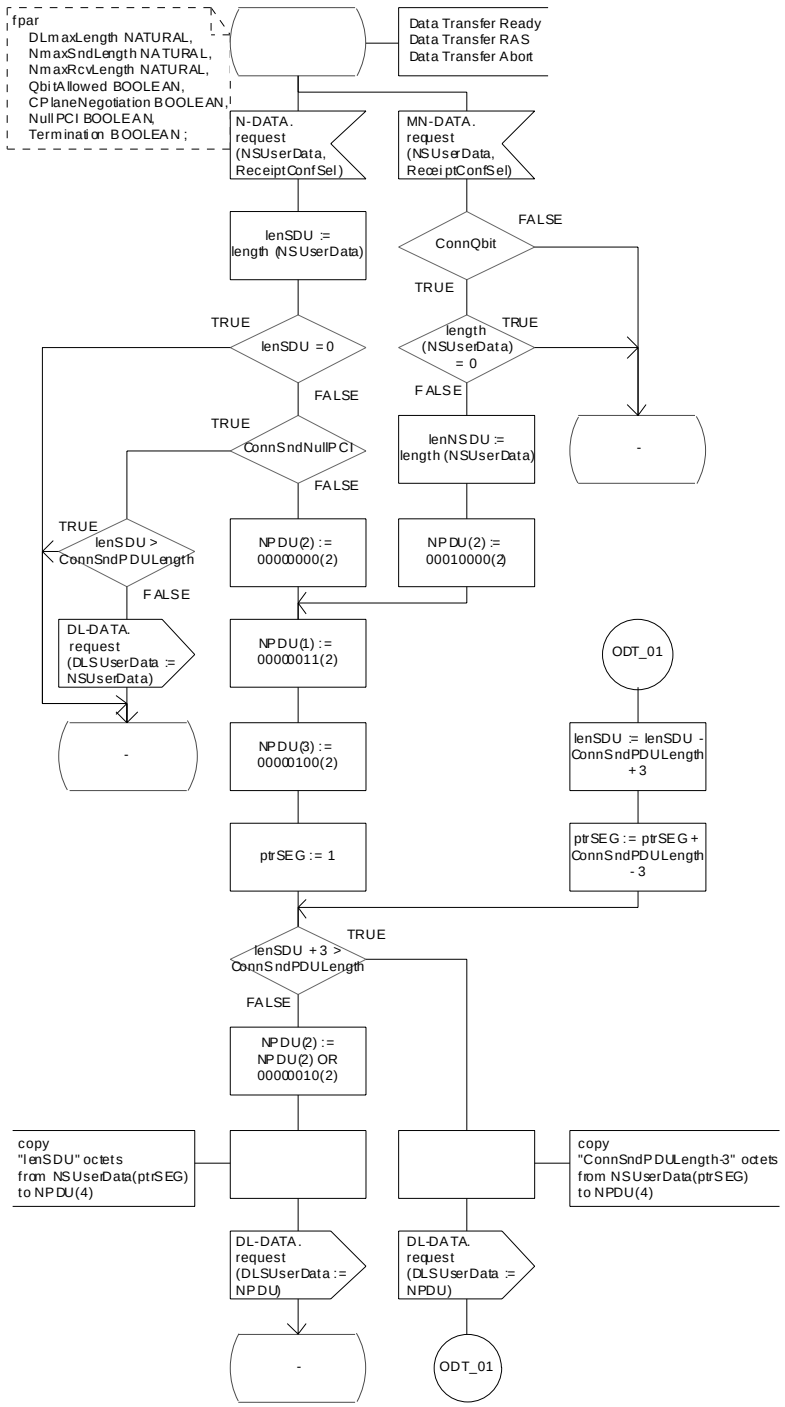


Figure C.4/X.633 (Sheet 8/15) – Network Fast Byte process

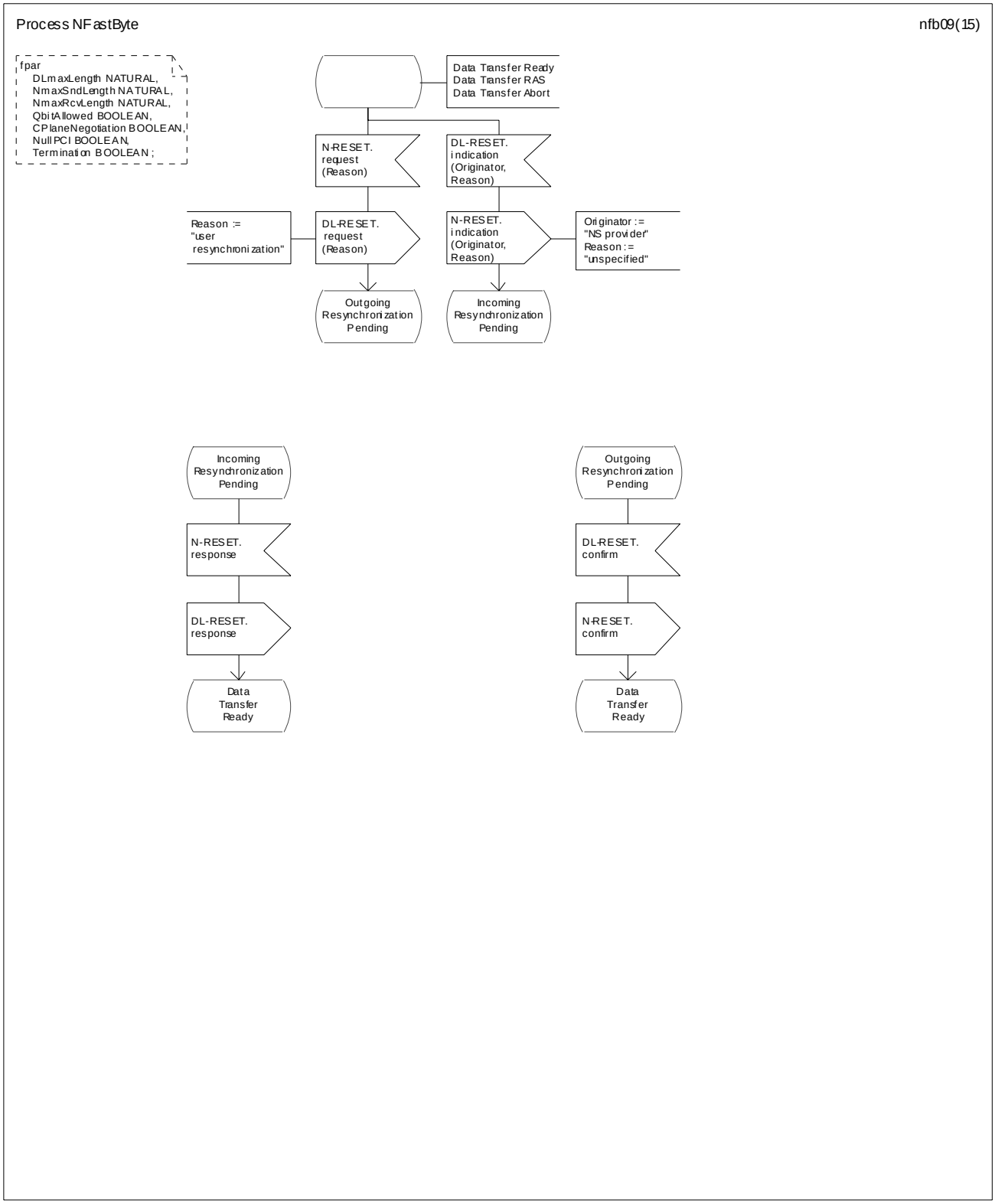


Figure C.4/X.633 (Sheet 9/15) – Network Fast Byte process

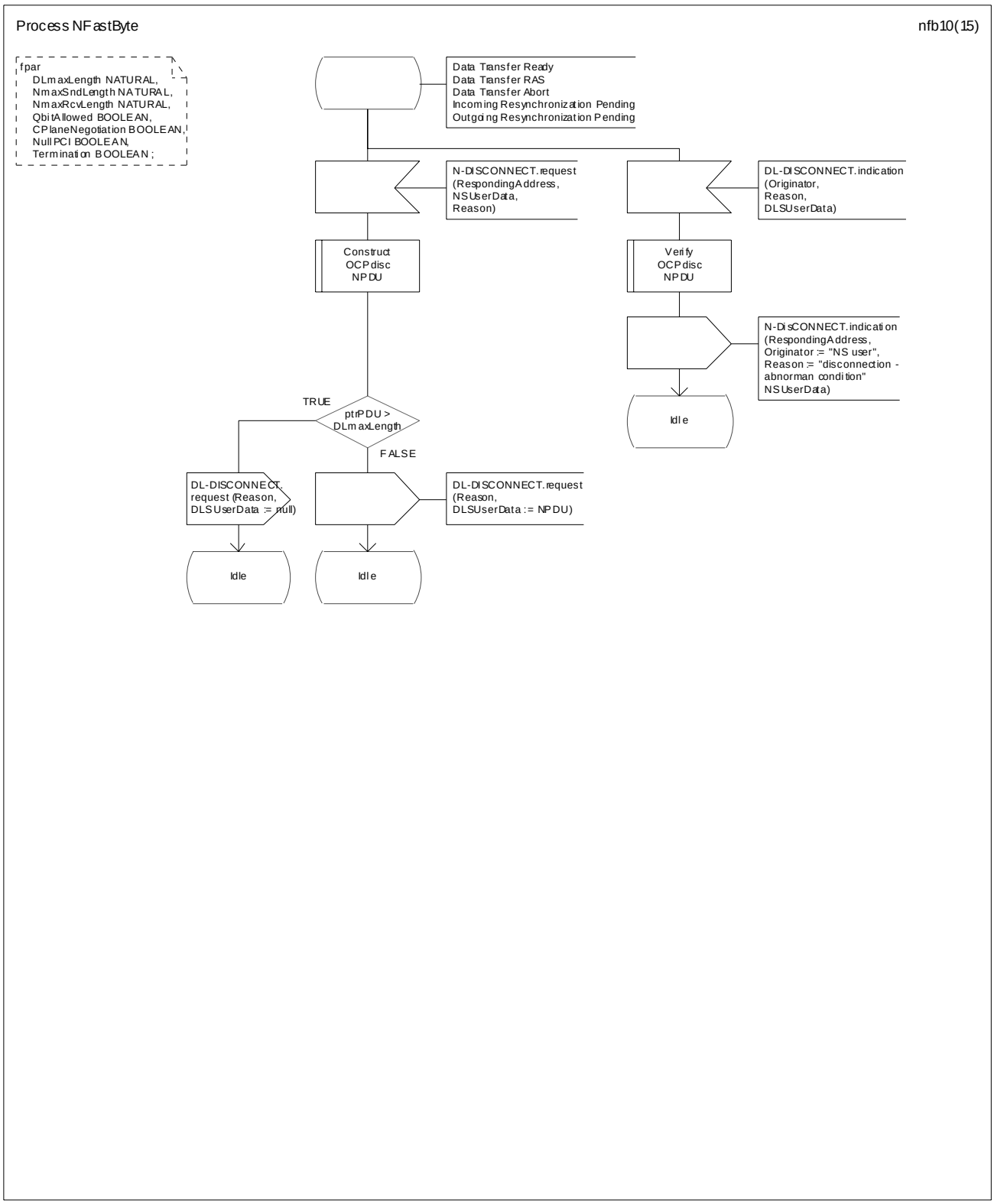


Figure C.4/X.633 (Sheet 10/15) – Network Fast Byte process

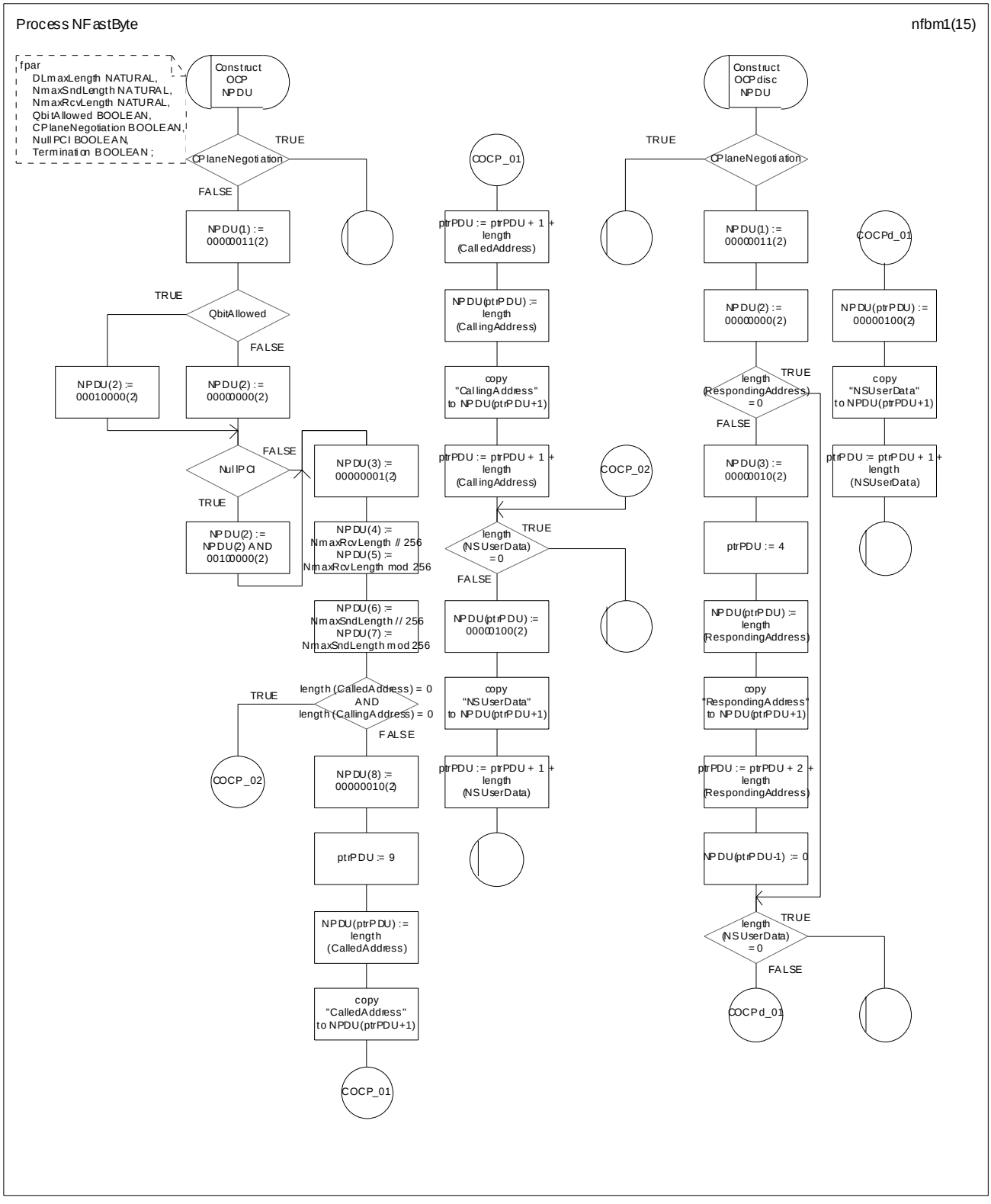


Figure C.4/X.633 (Sheet 11/15) – Network Fast Byte process

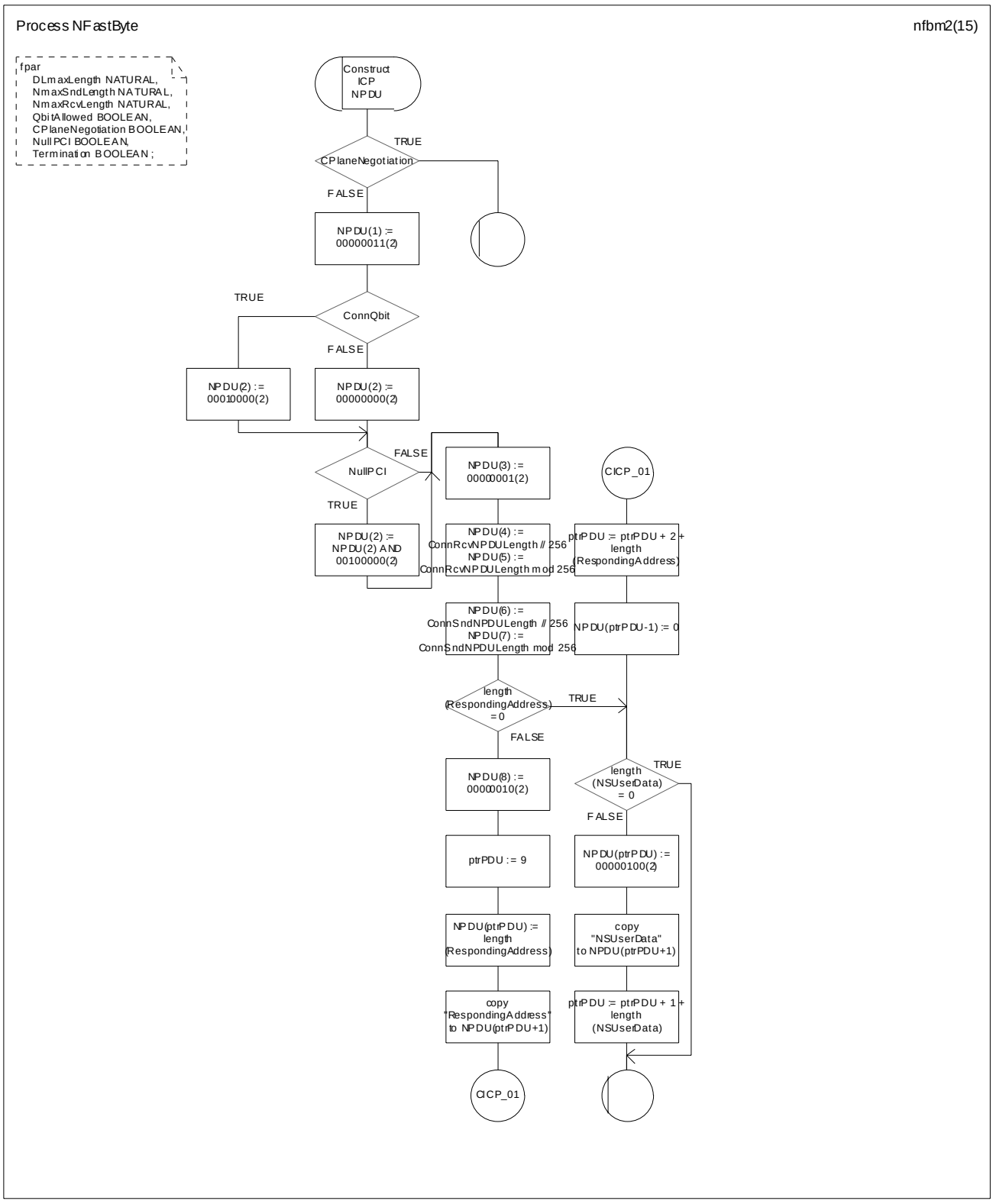


Figure C.4/X.633 (Sheet 12/15) – Network Fast Byte process

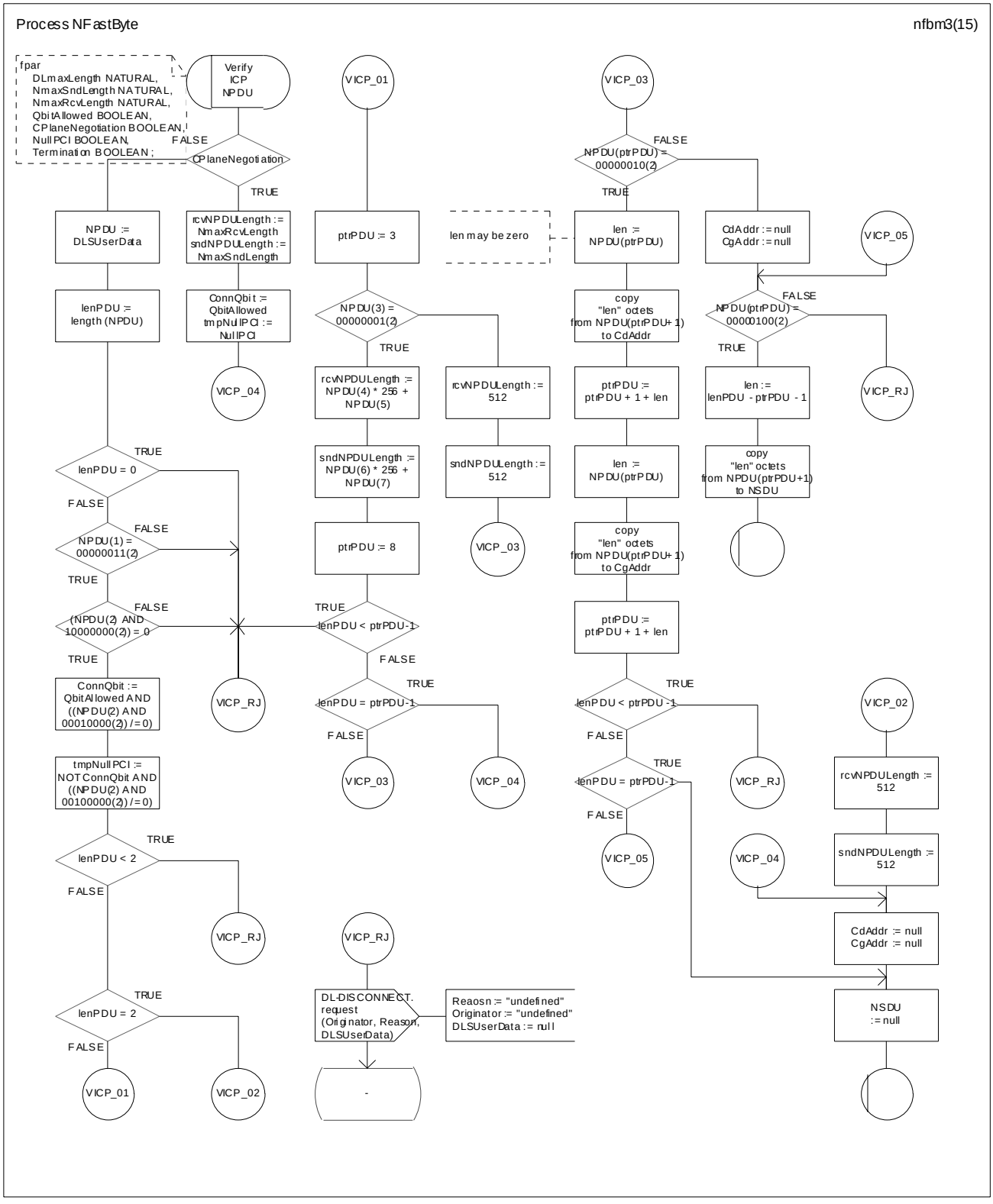


Figure C.4/X.633 (Sheet 13/15) – Network Fast Byte process

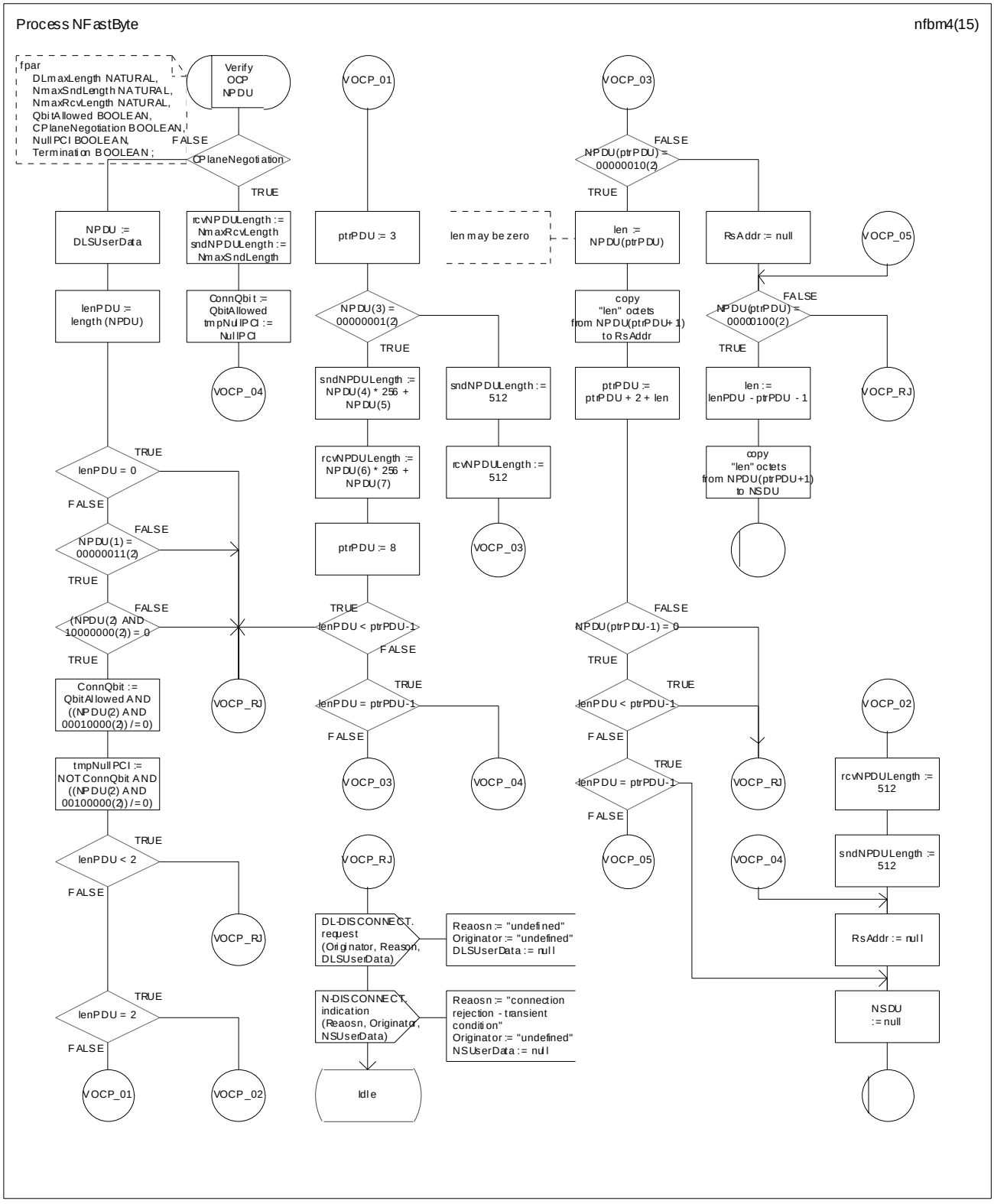


Figure C.4/X.633 (Sheet 14/15) – Network Fast Byte process

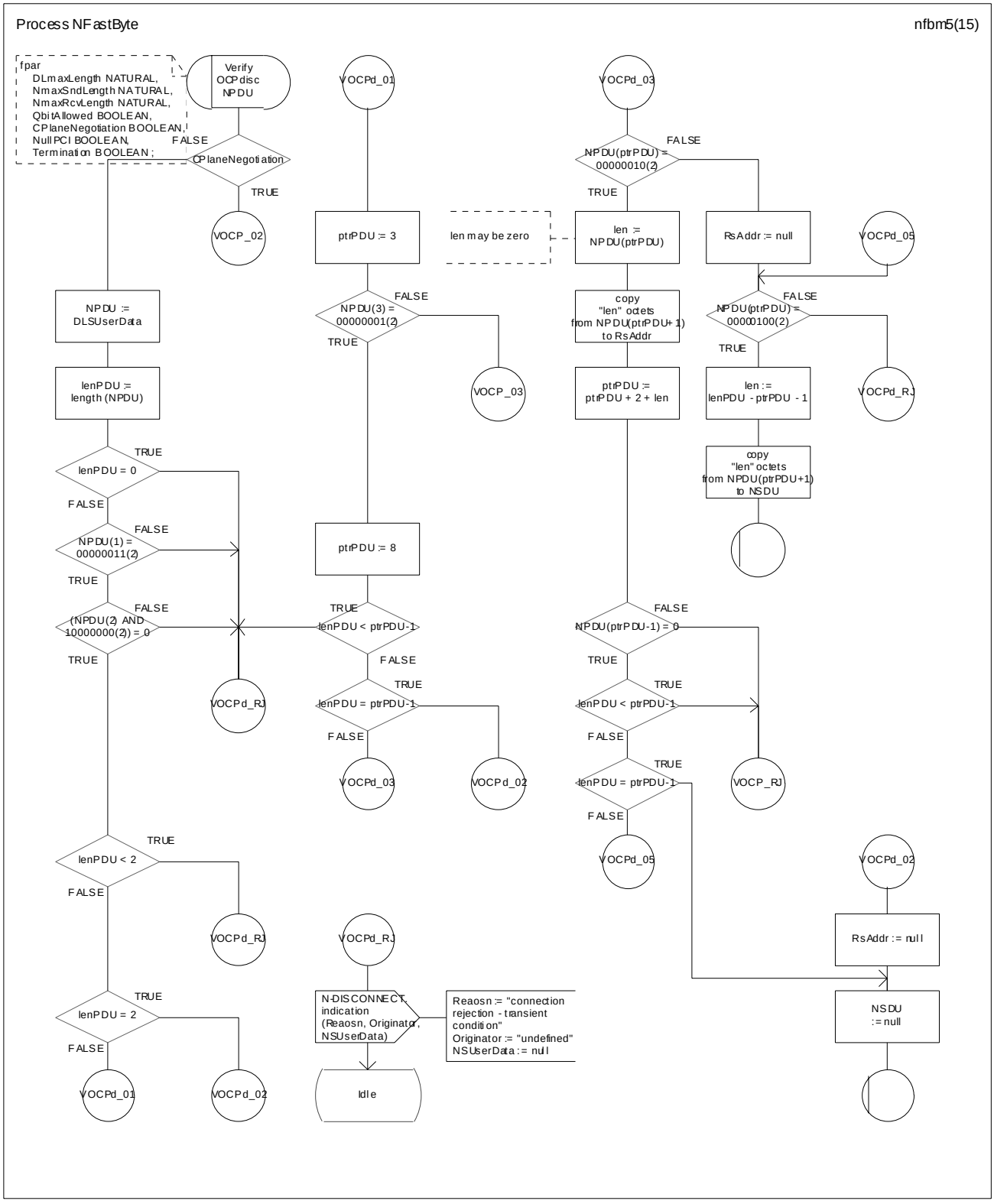


Figure C.4/X.633 (Sheet 15/15) – Network Fast Byte process

C.3.1 Procedures in the state "Idle"

1) Upon receipt of an N-CONNECT.request, an NPDU is constructed that is then transferred in the DLS-User-Data of the DL-CONNECT.request; the process enters state "Outgoing Connection Pending".

   If negotiation takes place outside the Network Fast Byte Protocol entity, no NPDU is transmitted. Otherwise, the NPDU constructed follows the specification in clause 7/X.633. The NullPCI capability is enabled as indicated in the parameter "NullPCI", the Q-bit is enabled if the parameter "QbitAllowed" is TRUE, the NPDU lengths are taken from the parameter "NmaxSndLength" and "NmaxRcvLength", and the called and calling addresses were received as parameters of the N-CONNECT.request. If NS-User-Data was also received with the primitive, this is also copied into the NPDU.

   NOTE 1 – The variable "Terminate" is set to TRUE after leaving state "Idle" if the parameter "Termination" has been set to TRUE also.

2) If the NPDU constructed above exceeds the maximum permissible length of the DLS-User-Data, an N-DISCONNECT.indication is returned to the Network Fast Byte Protocol entity user; the process remains in state "Idle".

3) Upon receipt of a DL-CONNECT.indication, the NPDU received in the DLS-User-Data is verified and the data extracted (if negotiation takes place outside the Network Fast Byte Protocol entity, no NPDU is verified and no data is extracted). The data items are then sent in the parameters of the N-CONNECT.indication to the Network Fast Byte Protocol entity user; the process enters state "Incoming Connection Pending".

   The verification and extraction is specified in macro "Verify ICP NPDU" and is as follows:

   a) If no NPDU has been received or the first octet does not contain the value "00000011_2" or the extension bit in octet 2 is set, the connection establishment is rejected with a DL-DISCONNECT.request.

   b) The variable "ConnQbit" is set if Q-bit processing is supported and the Q-bit in octet 2 is set.

   c) If octet 3 contains the value "00000001_2", the next 4 octets contain negotiation values for the maximum NPDU length; otherwise, the negotiation starts with the default maximum NPDU length values (512 octets).

   d) If the next octet contains the value "00000010_2", the next octets contain the called and calling address (these are temporarily stored in variables "CdAddr" and "CgAddr"); otherwise, both variables are set to the null character string.

   e) If the next octet contains the value "00000100_2", the information in the remaining octets is copied into the variable "NSDU" in preparation of sending it with the NS-User-Data in the N-CONNECT.indication primitive.

   If during the data extraction an improper NPDU is detected (e.g. a truncated NPDU), the connection establishment is rejected with a DL-DISCONNECT.request and an N-DISCONNECT.request.

   NOTE 2 – The variable "Terminate" is set to TRUE after leaving state "Idle" if the parameter "Termination" has been set to TRUE also.

4) If upon return to state "Idle" the variable "Terminate" is TRUE, the process stops.

C.3.2 Procedures in the state "Outgoing Connection Pending"

1) Upon receipt of a DL-CONNECT.confirm, the NPDU received in the DLS-User-Data is verified and the data extracted (if negotiation takes place outside the Network Fast Byte Protocol entity, no NPDU is verified and no data is extracted). These data items are then sent in the parameters of the N-CONNECT.confirm to the Network Fast Byte Protocol entity user. The connection parameters are set before the process enters state "Data Transfer Ready".

   The verification and extraction is specified in macro "OCP NPDU" and is as follows:

   a) If no NPDU has been received or the first octet does not contain the value "00000011_2" or the extension bit in octet 2 is set, the connection establishment is rejected with a DL-DISCONNECT.request and an N-DISCONNECT.request.

   b) The variable "ConnQbit" is set if Q-bit processing is supported and the Q-bit in octet 2 is set.

   c) If octet 3 contains the value "00000001_2", the next 4 octets contain negotiation values for the maximum NPDU length; otherwise, the negotiation starts with the default maximum NPDU length values (512 octets).

   d) If the next octet contains the value "00000010_2", the next octets contain the responding address (this is temporarily stored in variable "RsAddr"); otherwise, "RsAddr" is set to the null character string.

   e) If the next octet contains the value "00000100_2", the information in the remaining octets is copied into the variable "NSDU" in preparation of sending it with the NS-User-Data in the N-CONNECT.indication primitive.

   If during the data extraction an improper NPDU is detected (e.g. a truncated NPDU), the connection establishment is rejected with a DL-DISCONNECT.request and an N-DISCONNECT.request.

The following connection parameters are set:

- ConnSndNPDULength
- ConnRcvNPDULength
- ConnSndNullPCI
- ConnRcvNullPCI

NOTE – The permissibility of the Q-bit for the connection was established upon verification of the NPDU received.

2) If an N-DISCONNECT.request primitive is received, the Network Fast Byte Protocol entity user abandons the connection establishment. An NPDU is constructed that is then transferred in the DLS-User-Data of the DL-DISCONNECT.request; the process returns to state "Idle".

   If negotiation takes place outside the Network Fast Byte Protocol entity, no NPDU is transmitted. Otherwise, the NPDU constructed follows the specification in clause 7/X.633. The second octet of the header is set to zero. If a responding address has been communicated in a parameter, the corresponding control part is constructed (no calling address is included. If NS-User-Data was also received with the primitive, this is also copied into the NPDU.

3) If the NPDU constructed above exceeds the maximum permissible length of the DLS-User-Data, a DL-DISCONNECT.request without DLS-User-Date is sent to the Data Link layer; the process returns to state "Idle".

4) Upon receipt of a DL-DISCONNECT.indication, connection establishment is rejected. The NPDU received in the DLS-User-Data is verified and the data extracted (if negotiation takes place outside the Network Fast Byte Protocol entity, no NPDU is verified and no data is extracted). These data items are then sent in the parameters of the N-DISCONNECT.indication to the Network Fast Byte Protocol entity user; the process returns to state "Idle".

   The verification and extraction is specified in macro "Verify OCPdisc NPDU" and is as follows:

   a) If no NPDU has been received or the first octet does not contain the value "00000011_2" or the extension bit in octet 2 is set, the connection establishment rejection is notified with a standard N-DISCONNECT.request.

   b) If octet 3 contains the value "00000001_2", the next 4 octets are skipped.

   c) If the next octet contains the value "00000010_2", the next octets contain the responding address (this is temporarily stored in variable "RsAddr"); otherwise, "RsAddr" is set to the null character string.

   d) If the next octet contains the value "00000100_2", the information in the remaining octets is copied into the variable "NSDU" in preparation of sending it with the NS-User-Data in the N-DISCONNECT.indication primitive.

   If during the data extraction an improper NPDU is detected (e.g. a truncated NPDU), the connection establishment rejection is notified with a standard N-DISCONNECT.request.

C.3.3 Procedures in the state "Incoming Connection Pending"

1) Upon receipt of an N-CONNECT.response, the connection parameters are set and an NPDU is constructed that is then transferred in the DLS-User-Data of the DL-CONNECT.response; the process enters state "Data Transfer Ready".

   The following connection parameters are set:

   - ConnSndNPDULength
   - ConnRcvNPDULength
   - ConnSndNullPCI
   - ConnRcvNullPCI

   NOTE – The permissibility of the Q-bit for the connection was established upon verification of the NPDU received.

   If negotiation takes place outside the Network Fast Byte Protocol entity, no NPDU is transmitted. Otherwise, the NPDU constructed follows the specification in clause 7/X.633. The NullPCI capability is enabled as indicated in the parameter "NullPCI", the Q-bit is enabled if it was enabled in the NPDU received, the NPDU lengths are taken from the connection parameters above, and the responding address was received as parameters of the N-CONNECT.response. If NS-User-Data was also received with the primitive, this is also copied into the NPDU.

2) The procedures upon receipt of an N-DISCONNECT.request are described in C.3.2 items 2) and 3).

3) The procedures upon receipt of a DL-DISCONNECT.indication are described in C.3.2 item 4).

C.3.4 Procedures in the state "Data Transfer Ready"

1) The Network Fast Byte Protocol entity user submits data for transmission with the NS-User-Data parameter of an N-DATA.request. If no NS-User-Data is present, the primitive is ignored. If the NullPCI capability has been negotiated in the outgoing direction, the NS-User-Data is copied to the DLS-User-Data of the DL-DATA.request primitive; the process remains in the same state.

2) If the NullPCI capability in the outgoing direction is not enabled, an NPDU header is constructed (octets 1 and 2) and octet 3 introduces the data part. If this header and the complete NS-User-Data fits into a single NPDU, this is transmitted in a single DL-DATA.request; the process remains in the same state.

3) If the header and the complete NS-User-Data does not fit into a single NPDU, a segmentation process is executed whereby as many maximum size NPDUs (with the EON-bit set to zero) as necessary are sent before the last NPDU with the remainder of the NSDU and the EON-bit set is submitted with the DL-DATA.request primitive for transmission. The process remains in the same state.

4) The Layer Management entity submits data for transmission in the NS-User-Data parameter of an MN-DATA.request. The NullPCI capability is incompatible with the Q-bit capability; hence, an NPDU header with the Q-bit set is constructed (octets 1 and 2) and octet 3 introduces the data part. If this header and the complete NS-User-Data fit into a single NPDU, this is transmitted in a single DL-DATA.request; the process remains in the same state. If segmentation is necessary, the same procedure as in item 3) above is executed.

5) If the NullPCI capability has been negotiated in the incoming direction and a DL-DATA.indication is received, the DLS-User-Data is communicated in its entirety in the NS-User-Data parameter of an N-DATA.indication primitive to the Network Fast Byte Protocol entity user. The process remains in state "Data Transfer Ready".

6) If the NullPCI capability in the incoming direction is not enabled, the header is verified (improper NPDUs are discarded and the process remains in state "Data Transfer Ready") and the Q-bit and EON-bit are extracted. The remaining data is copied to the beginning of the variable "NSDU". If the EON-bit is set and:

   a) the Q-bit is not set, the NSDU is delivered to the Network Fast Byte Protocol entity user in the NS-User-Data parameter of an N-DATA.indication primitive; or if

   b) the Q-bit is set, the NSDU is delivered to Layer Management in the NS-User-Data parameter of an MN-DATA.indication primitive.

   The process remains in state "Data Transfer Ready". If on the other hand the EON-bit is not set, the variable "ptrRAS" is set to indicate the place where reassembly continues in the NSDU and the process enters state "Data Transfer RAS".

7) If the Q-bit is set and the Q-bit capability is not enabled for this connection or the data received exceeds the maximum permissible size, the data is discarded. If in addition:

   a) the EON-bit is not set, the process remains in state "Data Transfer Ready"; or if

   b) the EON-bit is set, the process enters state "Data Transfer Abort".

8) The procedures upon receipt of an N-DISCONNECT.request are described in C.3.2 items 2) and 3).

9) The procedures upon receipt of a DL-DISCONNECT.indication are described in C.3.2 item 4).

10) Upon receipt of an N-RESET.request, the process issues a DL-RESET.request primitive and enters state "Outgoing Resynchronization Pending".

11) Upon receipt of a DL-RESET.indication, the process issues an N-RESET.indication primitive and enters state "Incoming Resynchronization Pending".

C.3.5 Procedures in the state "Data Transfer RAS"

1) If a DL-DATA.indication is received, the header is verified (improper NPDUs are discarded and the process returns to state "Data Transfer RAS") and the Q-bit and EON-bit are extracted. The remaining data is copied to the variable "NSDU" (indicated by variable "ptrRAS"). If the EON-bit is set and

   a) the Q-bit is not set, the reassembled NSDU is delivered to the Network Fast Byte Protocol entity user in the NS-User-Data parameter of an N-DATA.indication primitive; or if

   b) the Q-bit is set, the reassembled NSDU is delivered to Layer Management in the NS-User-Data parameter of an MN-DATA.indication primitive.

   The process returns to state "Data Transfer Ready". If on the other hand the EON-bit is not set, the variable "ptrRAS" is set to indicate the place where reassembly continues in the NSDU and the process remains in state "Data Transfer RAS".

2) If the Q-bit in this NPDU does not match the Q-bit of the first NPDU of the currently reassembling NSDU or if the data received exceeds the maximum permissible size, the complete NSDU is discarded. If in addition:

   a) the EON-bit is not set, the process returns to state "Data Transfer Ready"; or if

   b) the EON-bit is set, the process enters state "Data Transfer Abort".

3) The procedures upon receipt of an N-DATA.request are described in C.3.4 items 1), 2), and 3).

4) The procedures upon receipt of an MN-DATA.request are described in C.3.4 item 4).

5) The procedures upon receipt of an N-DISCONNECT.request are described in C.3.2 items 2) and 3).

6) The procedures upon receipt of a DL-DISCONNECT.indication are described in C.3.2 item 4).

7) The procedures upon receipt of an N-RESET.request are described in C.3.4 item 10).

8) The procedures upon receipt of a DL-RESET.indication are described in C.3.4 item 11).

C.3.6 Procedures in the state "Data Transfer Abort"

1) If a DL-DATA.indication is received, the header is verified (improper NPDUs are discarded and the process returns to state "Data Transfer Ready") and the EON-bit is extracted. If the EON-bit is set, the process returns to state "Data Transfer Ready". If on the other hand the EON-bit is not set, the process remains in state "Data Transfer Abort".

2) The procedures upon receipt of an N-DATA.request are described in C.3.4 items 1), 2), and 3).

3) The procedures upon receipt of an MN-DATA.request are described in C.3.4 item 4).

4) The procedures upon receipt of an N-DISCONNECT.request are described in C.3.2 items 2) and 3).

5) The procedures upon receipt of a DL-DISCONNECT.indication are described in C.3.2 item 4).

6) The procedures upon receipt of an N-RESET.request are described in C.3.4 item 10).

7) The procedures upon receipt of a DL-RESET.indication are described in C.3.4 item 11).

C.3.7 Procedures in the state "Outgoing Resynchronization Pending"

1) Upon receipt of a DL-RESET.confirm, the process issues a N-RESET.confirm primitive and returns to state "Data Transfer Ready".

2) The procedures upon receipt of an N-DISCONNECT.request are described in C.3.2 items 2) and 3).

3) The procedures upon receipt of a DL-DISCONNECT.indication are described in C.3.2 item 4).

C.3.8 Procedures in the state "Incoming Resynchronization Pending"

1) Upon receipt of an N-RESET.response, the process issues a DL-RESET.response primitive and enters state "Data Transfer Ready".

2) The procedures upon receipt of an N-DISCONNECT.request are described in C.3.2 items 2) and 3).

3) The procedures upon receipt of a DL-DISCONNECT.indication are described in C.3.2 item 4).

Annex D

SDL specification of Annex B of the Network Fast Byte Protocol

D.1 The system and block structure

The SDL system diagram of the Synchronization and Coordination entity for the Fast Byte Network Protocol is shown in Figure D.1 and the SDL block structure in Figure D.2.

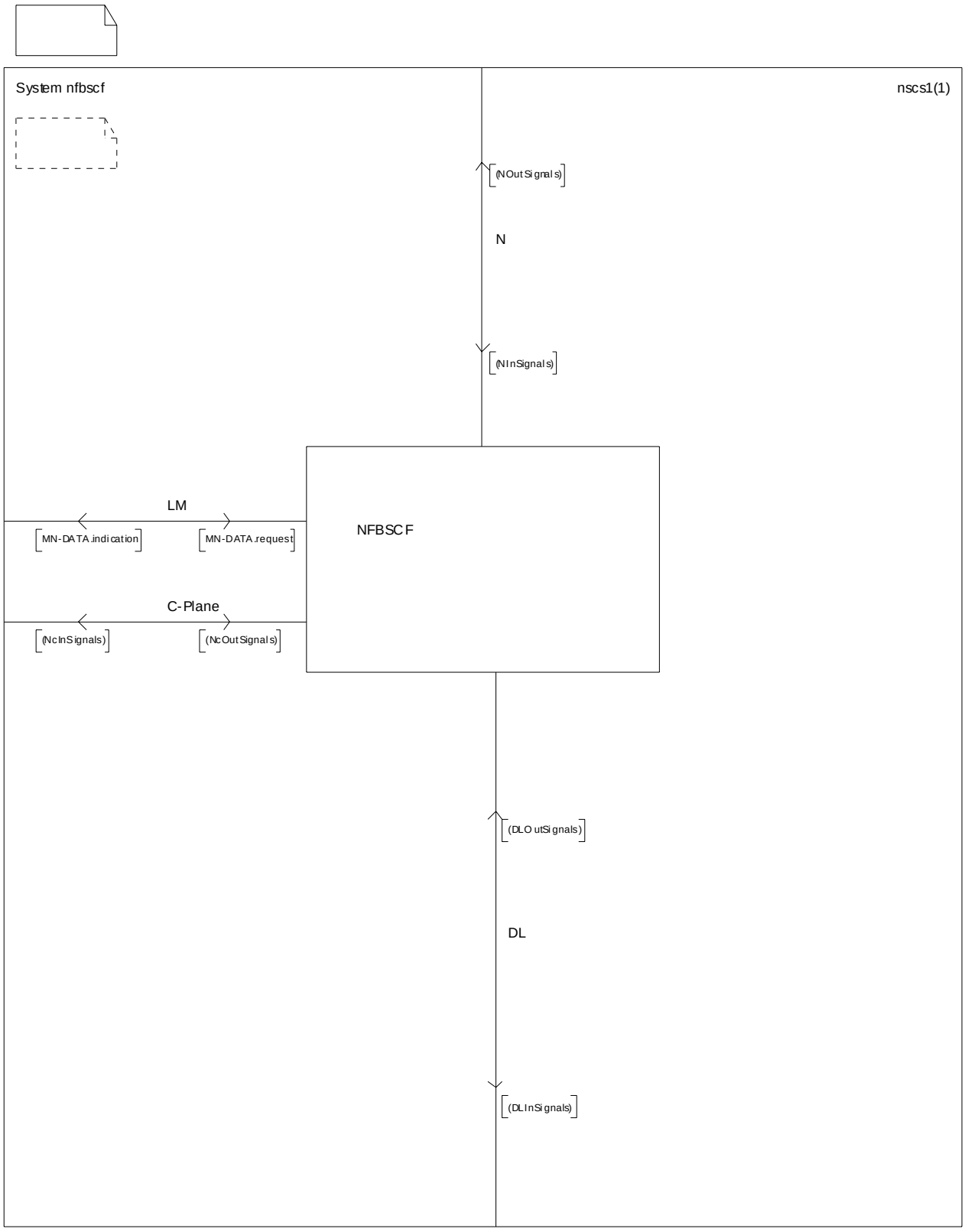


Figure D.1/X.633 – System of the Synchronization and Coordination Function of the Network Fast Byte Protocol

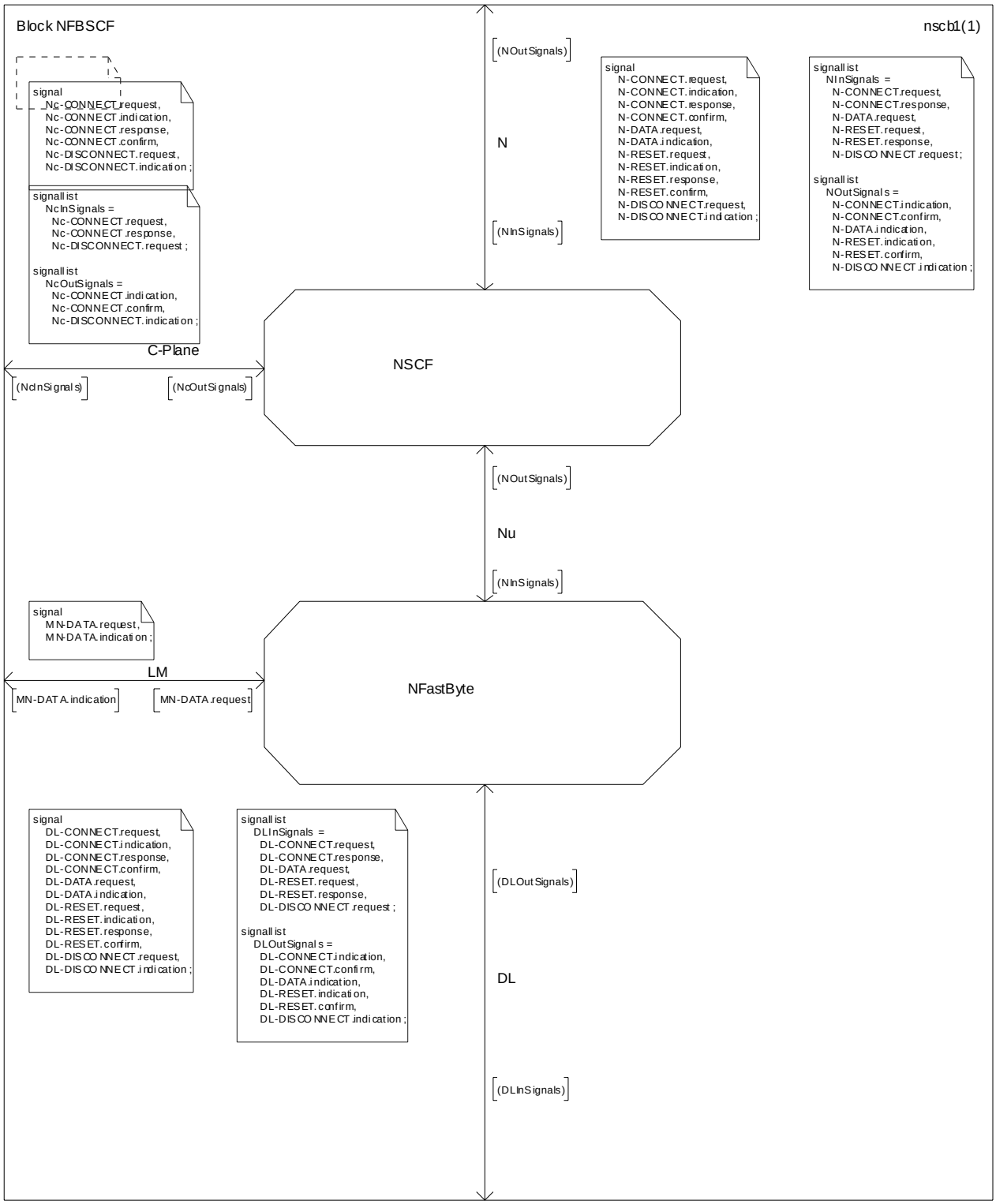


Figure D.2/X.633 – Block structure of the Synchronization and Coordination Function of the Network Fast Byte Protocol

D.2 Interaction with Management and Control Planes

D.2.1 Management Plane

No interactions with the Management Plane are specified.

D.2.2 Control Plane

This annex specifies a Synchronization and Coordination Function (SCF) below a Service Access Point (SAP) that offers the Connection-Oriented Network Service (CONS). The C-plane signalling is synchronized and coordinated with the connection establishment and release requests from the user and the Network Fast Byte Protocol entity.

D.3 Procedure of the Synchronization and Coordination entity

The SDL diagrams of the procedure of the Synchronization and Coordination entity are given in this subclause. If there exists any difference between the prose description also given in this subclause and the SDL diagrams, the SDL diagrams take precedence. On the other hand, if there exists differences between the specification in this Annex and the one in Annex B/X.633, the specification in Annex B/X.633 takes precedence.

NOTE – In the SDL diagrams of this subclause, the octets in all PDUs and SDUs, i.e. the NSDU, are numbered from "1" to at most "65535".

Two modes of operation for the Synchronization and Coordination entity are defined:

1) according to B.2.1 ("Fast Byte Protocol specification – Exclusive use of the subnetwork User-plane") that is referred to as "negotiation takes place in the U-plane"; and

2) according to subclause B.2.2 ("Fast Byte Protocol specification – Use of the subnetwork User-plane and Control-plane") that is referred to as "negotiation takes place in the C-plane".

The operation of the Synchronization and Coordination entity is modeled as a state machine consisting of the following states:

– **Idle:** Each Synchronization and Coordination entity is conceptually initiated in the Idle state and returns to this state upon the release of a connection.

– **Outgoing C-plane Connection Pending:** A Synchronization and Coordination entity requesting a connection with its peer is in the Outgoing C-plane Connection Pending state until it receives acknowledgment from its peer via the C-plane.

– **Outgoing U-plane Connection Pending:** A Synchronization and Coordination entity requesting a connection with its peer is in the Outgoing U-plane Connection Pending state until it receives acknowledgment from its peer via the U-plane.

– **Incoming C-plane Connection Pending:** A Synchronization and Coordination entity that has received a connection request from its peer via the C-plane and is:

  - waiting for its user's response (if negotiation takes place in the C-plane); or
  - waiting for the U-plane connection indication (if negotiation takes place in the U-plane),

  is in the Incoming C-plane Connection Pending state.

– **Incoming U-plane Connection Pending:** A Synchronization and Coordination entity that has received a connection request from its peer and is:

  - waiting for its user's response (if negotiation takes place in the U-plane); or
  - waiting for the U-plane connection indication (if negotiation takes place in the C-plane),

  is in the Incoming U-plane Connection Pending state.

– **Outgoing Disconnection Pending:** A Synchronization and Coordination entity requesting the release of the connection where negotiation takes place in the U-plane and is awaiting the connection release indication from the C-plane is in the Outgoing Disconnection Pending state; this state is protected by a timer.

– **Data Transfer Ready:** Upon successful completion of the connection establishment, both peer Synchronization and Coordination entities will be in Data Transfer Ready state and data transfer or resynchronization can take place.

The description of the operations of the Synchronization and Coordination entity make use of the following state variables:

CdAddr	A character string holding the "Called Address" as extracted out of an NPDU.
CgAddr	A character string holding the "Calling Address" as extracted out of an NPDU.
RsAddr	A character string holding the "Responding Address" as extracted out of an NPDU.
NPDU	An octet string holding the NPDU being constructed in the outgoing direction or the NPDU just having been received.
NSDU	An octet string holding the NSDU being reassembled in the incoming direction.
ptrPDU	An index into the variable NPDU having been received indicating where to retrieve the next information during interpretation of the received NPDU.
lenPDU	The length of an NPDU having been received.
SndNPDULength	A temporary integer variable during connection establishment used for negotiation of the maximum size of an NPDU in the outgoing direction.
ConnSndNPDULength	An integer variable indicating the maximum size of an NPDU in the outgoing direction.
RcvNPDULength	A temporary integer variable during connection establishment used for negotiation of the maximum size of an NPDU in the incoming direction.
ConnRcvNPDULength	An integer variable indicating the maximum size of an NPDU in the incoming direction.
tmpNullPCI	A temporary boolean variable during connection establishment used for negotiation of the NullPCI capability.
currQbit	A boolean variable being set to TRUE if the Q-bit in the received NPDU has been set.
ConnQbit	A boolean variable being set to TRUE if Layer Management data transfer (the use of the Q-bit) is allowed on this connection.

The Synchronization and Coordination entity maintains the following parameters:

DLmaxLength	The maximum size of the DLSUserData parameter.
NmaxRcvLength	The maximum size of the NSUserData parameter in the incoming direction.
NmaxSndLength	The maximum size of the NSUserData parameter in the outgoing direction.
QbitAllowed	A boolean value being set to TRUE if Layer Management data transfer (the use of the Q-bit) is enabled.
CPlaneNegotiation	A boolean value being set to TRUE if the parameter negotiation takes place outside the Synchronization and Coordination entity, e.g. in the C-plane.

The SDL definition of the Synchronization and Coordination entity process is shown in Figure D.3.

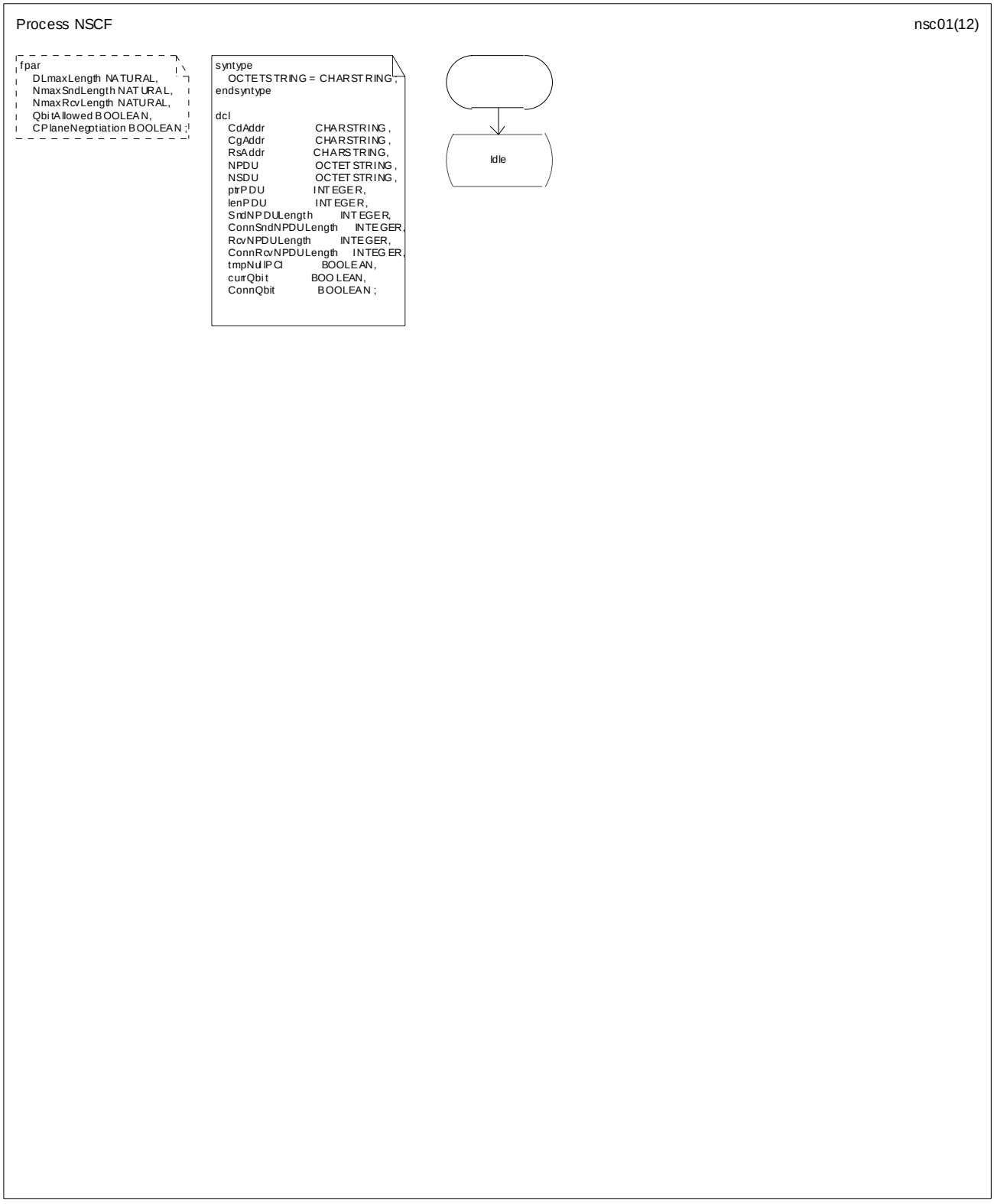


Figure D.3/X.633 (Sheet 1 of 12) – Synchronization and Coordination process

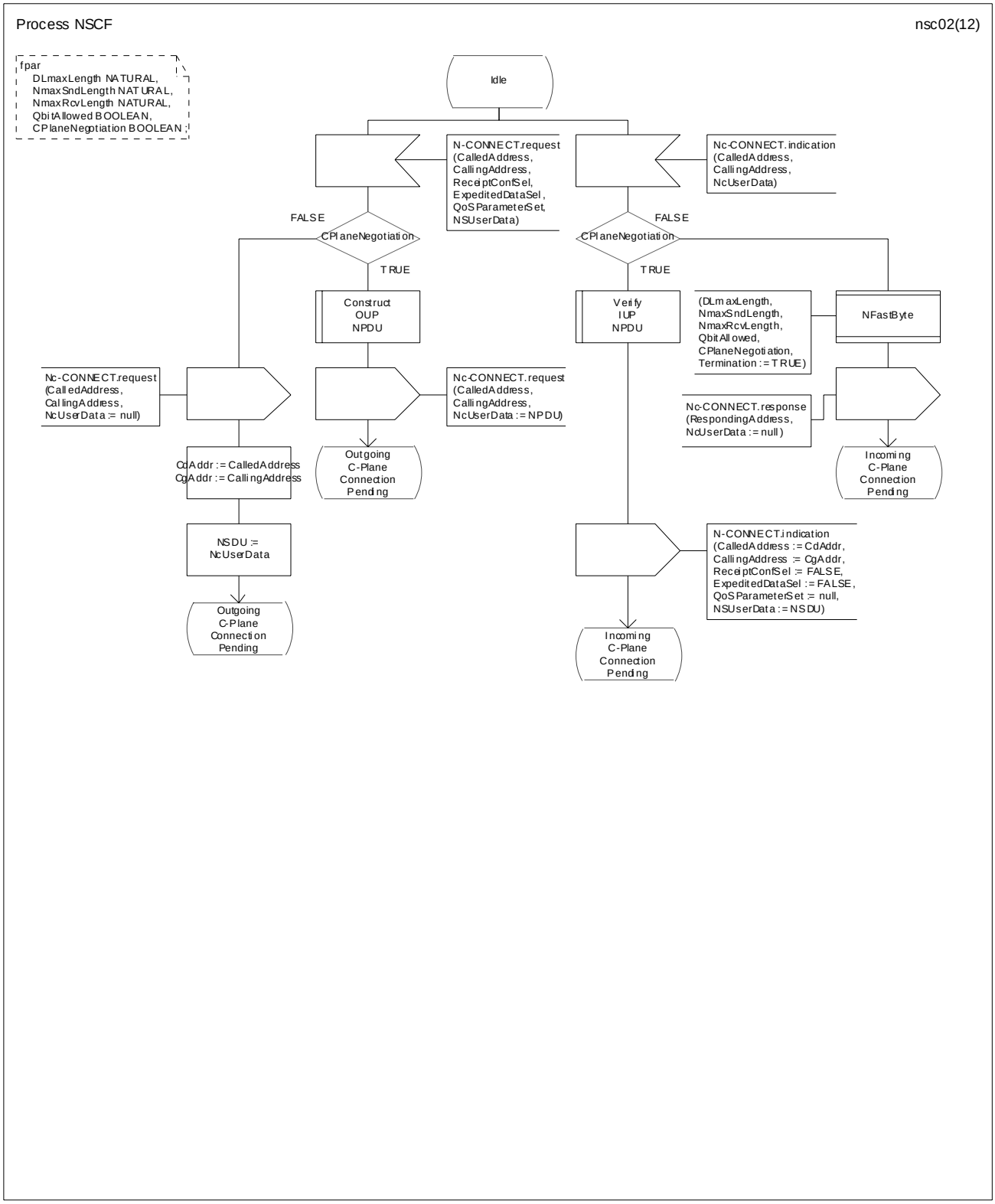


Figure D.3/X.633 (Sheet 2 of 12) – Synchronization and Coordination process

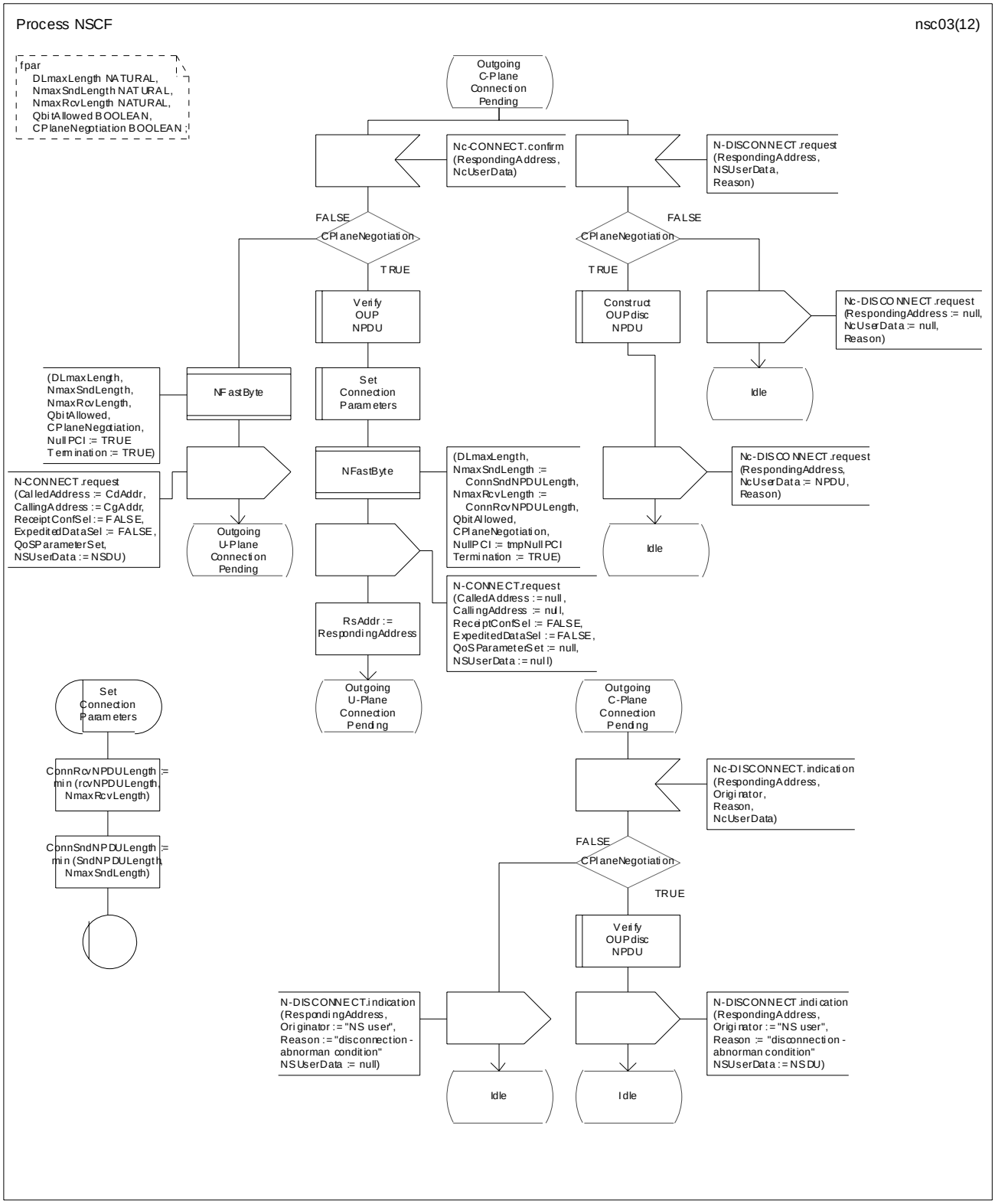


Figure D.3/X.633 (Sheet 3 of 12) – Synchronization and Coordination process

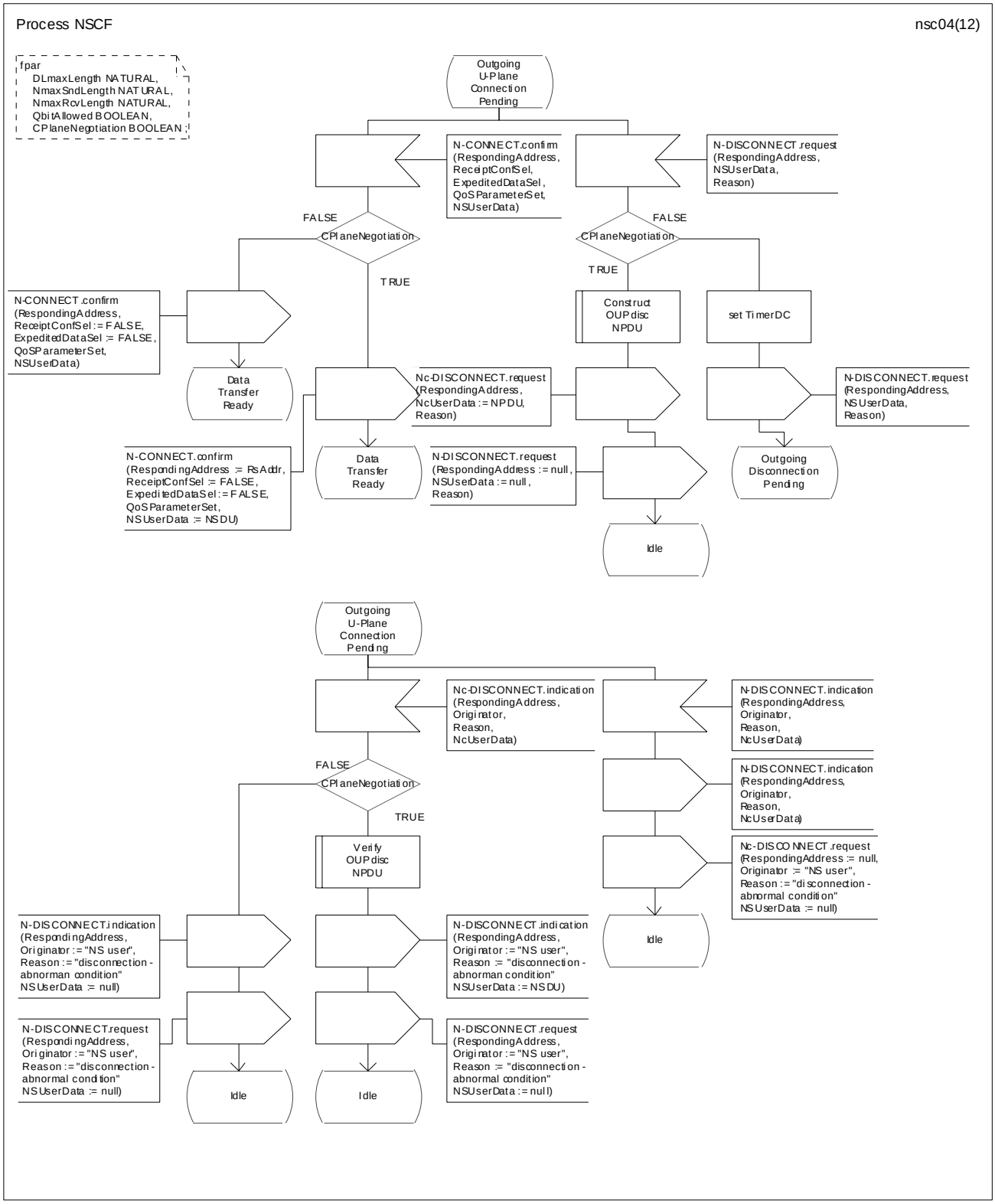


Figure D.3/X.633 (Sheet 4 of 12) – Synchronization and Coordination process

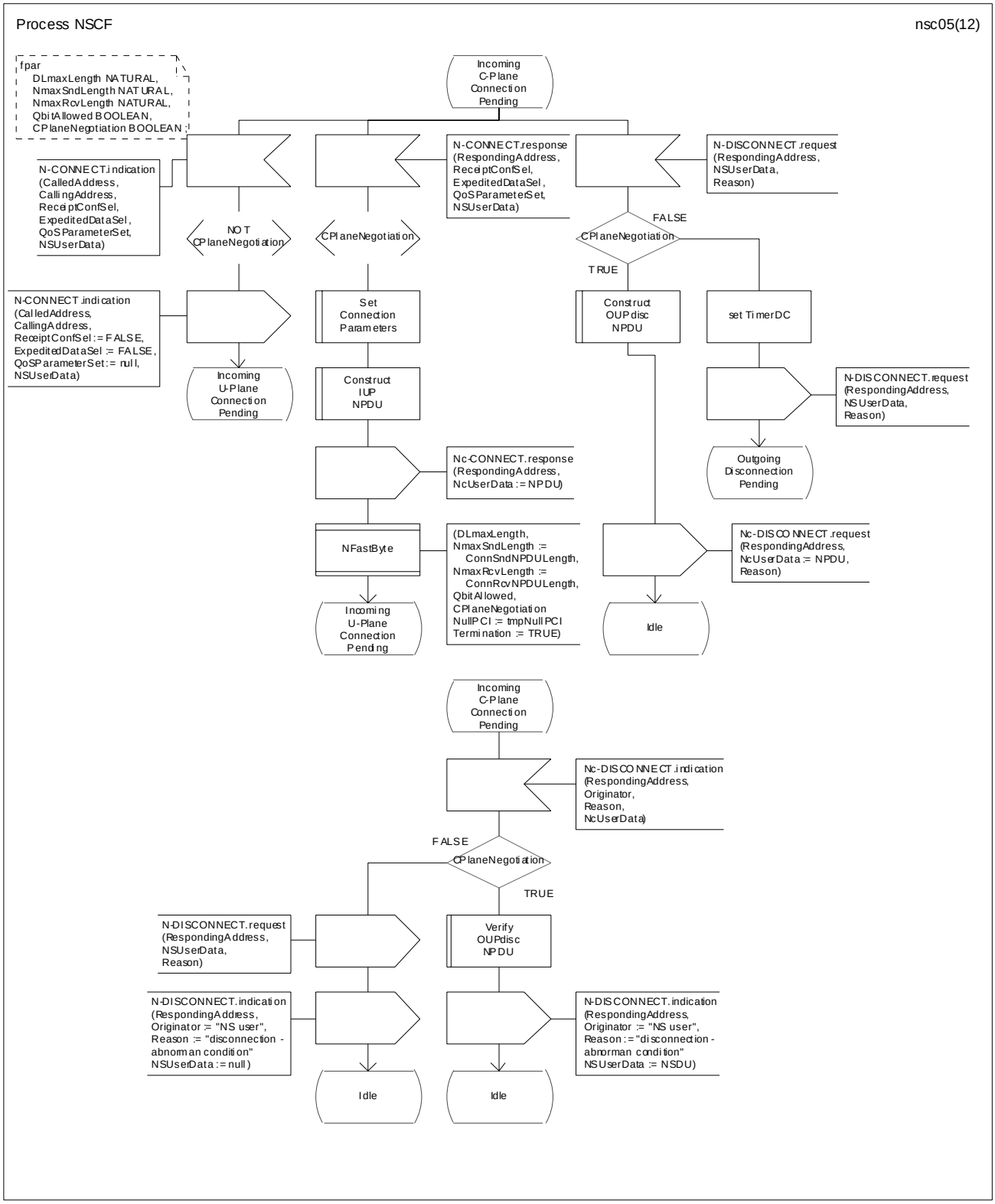


Figure D.3/X.633 (Sheet 5 of 12) – Synchronization and Coordination process

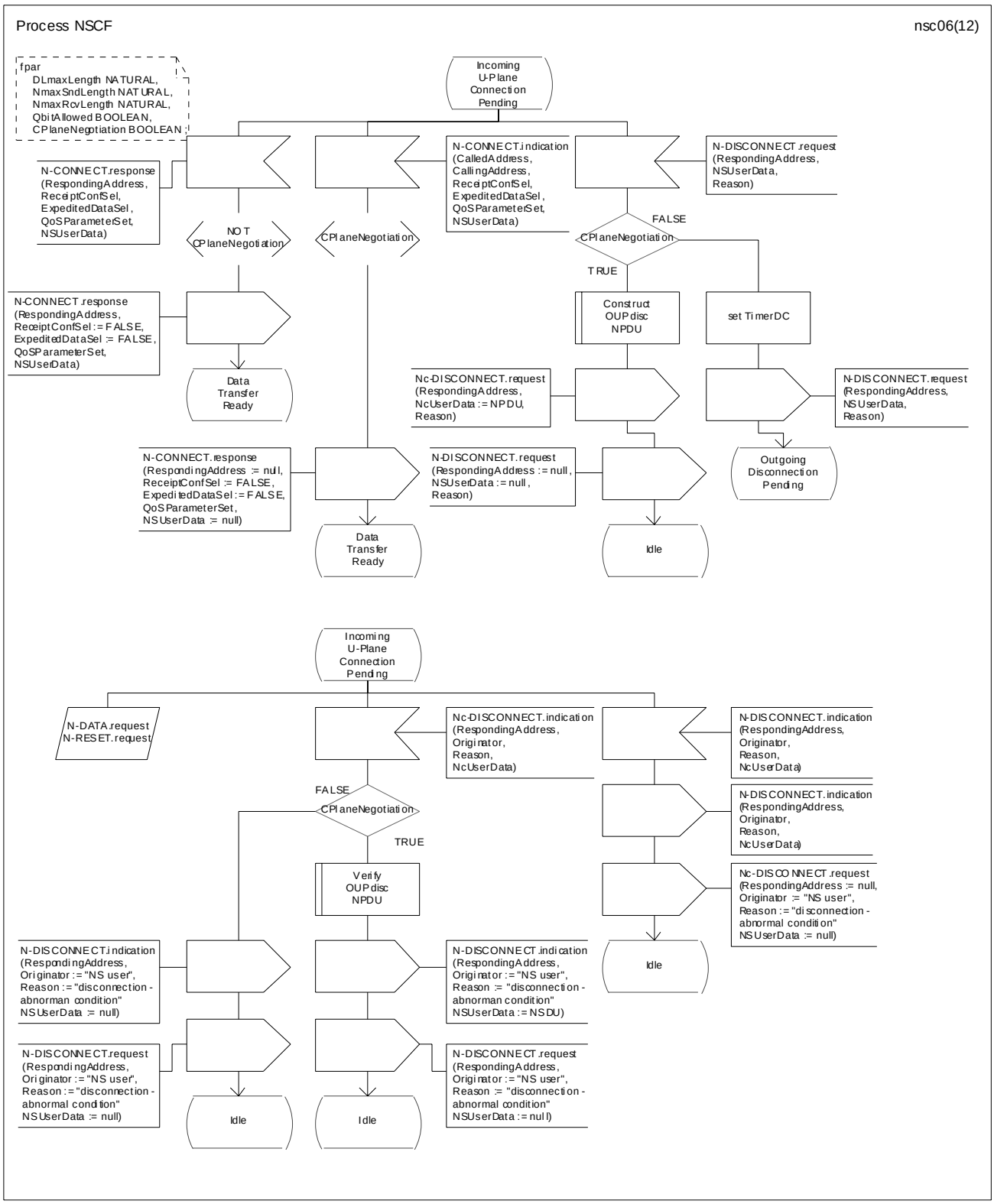


Figure D.3/X.633 (Sheet 6 of 12) – Synchronization and Coordination process

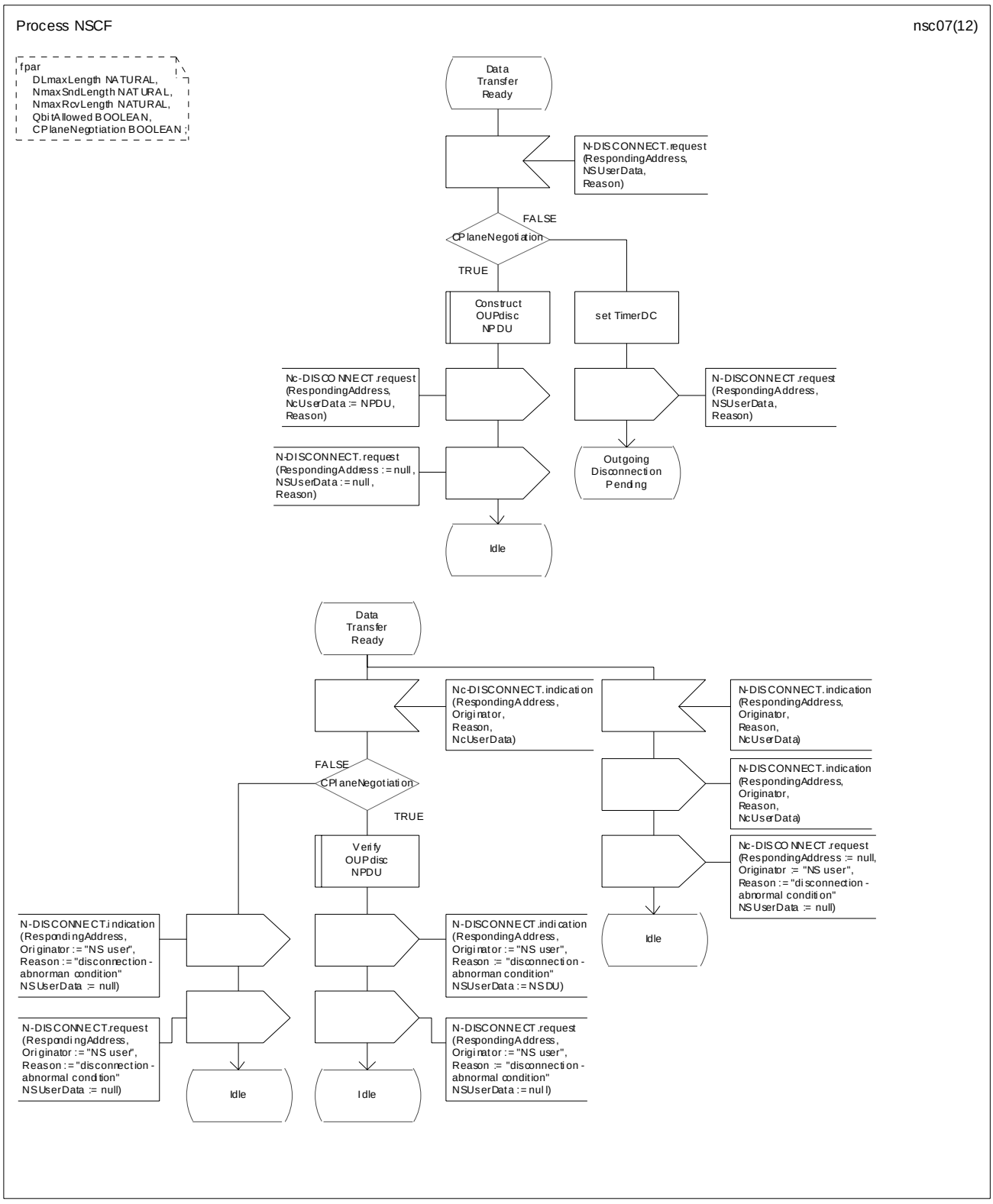


Figure D.3/X.633 (Sheet 7 of 12) – Synchronization and Coordination process

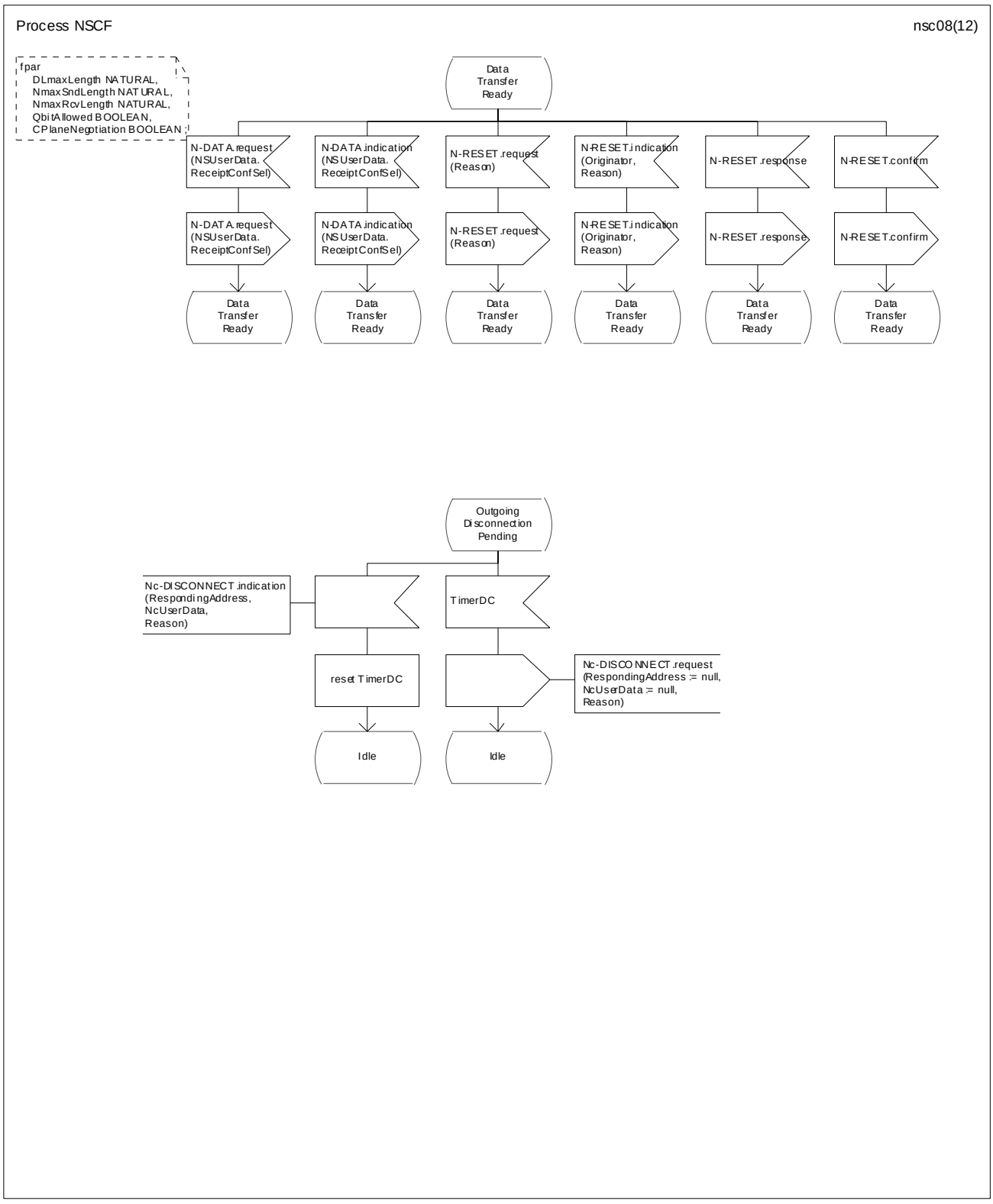


Figure D.3/X.633 (Sheet 8 of 12) – Synchronization and Coordination process

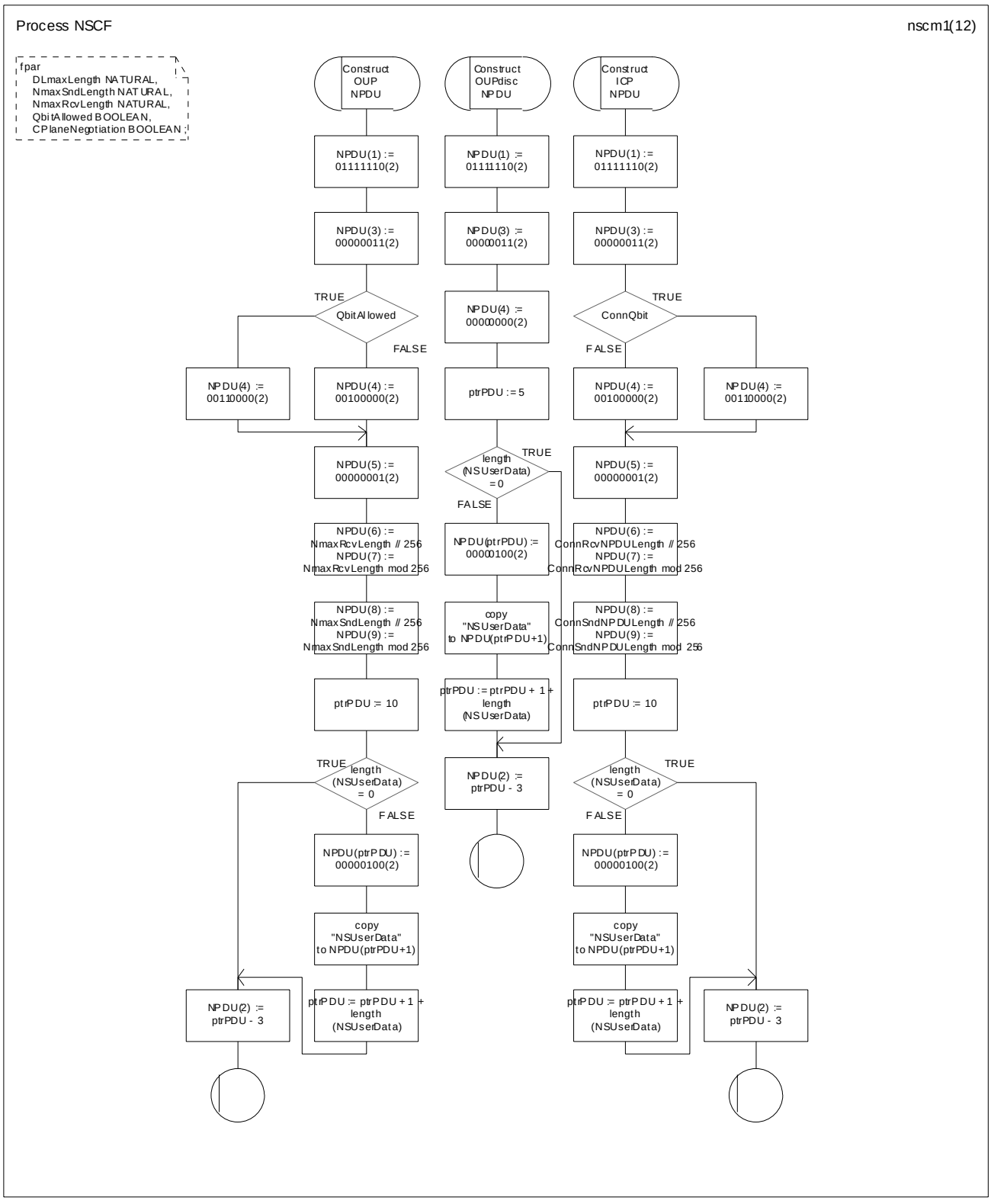


Figure D.3/X.633 (Sheet 9 of 12) – Synchronization and Coordination process

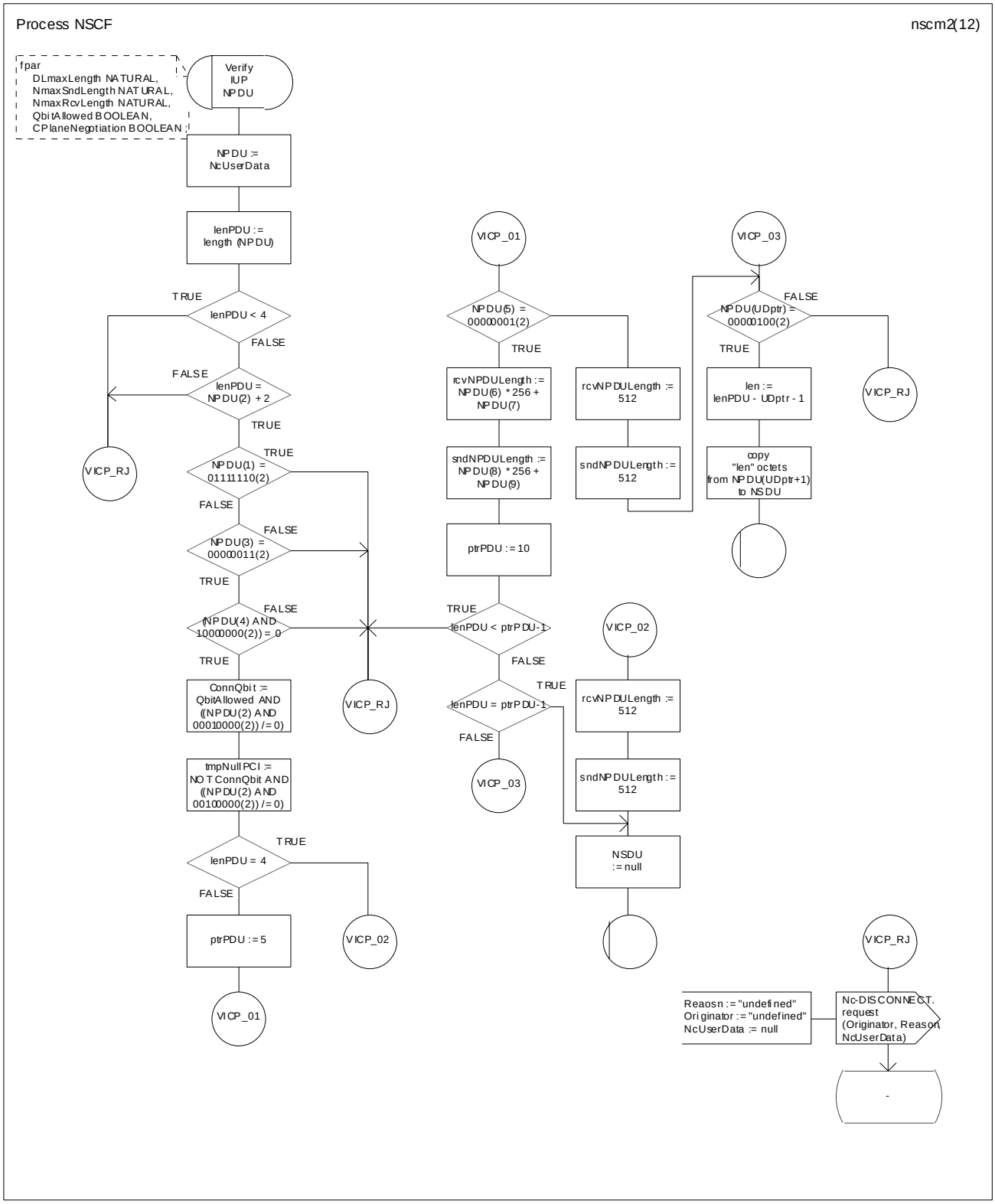


Figure D.3/X.633 (Sheet 10 of 12) – Synchronization and Coordination process

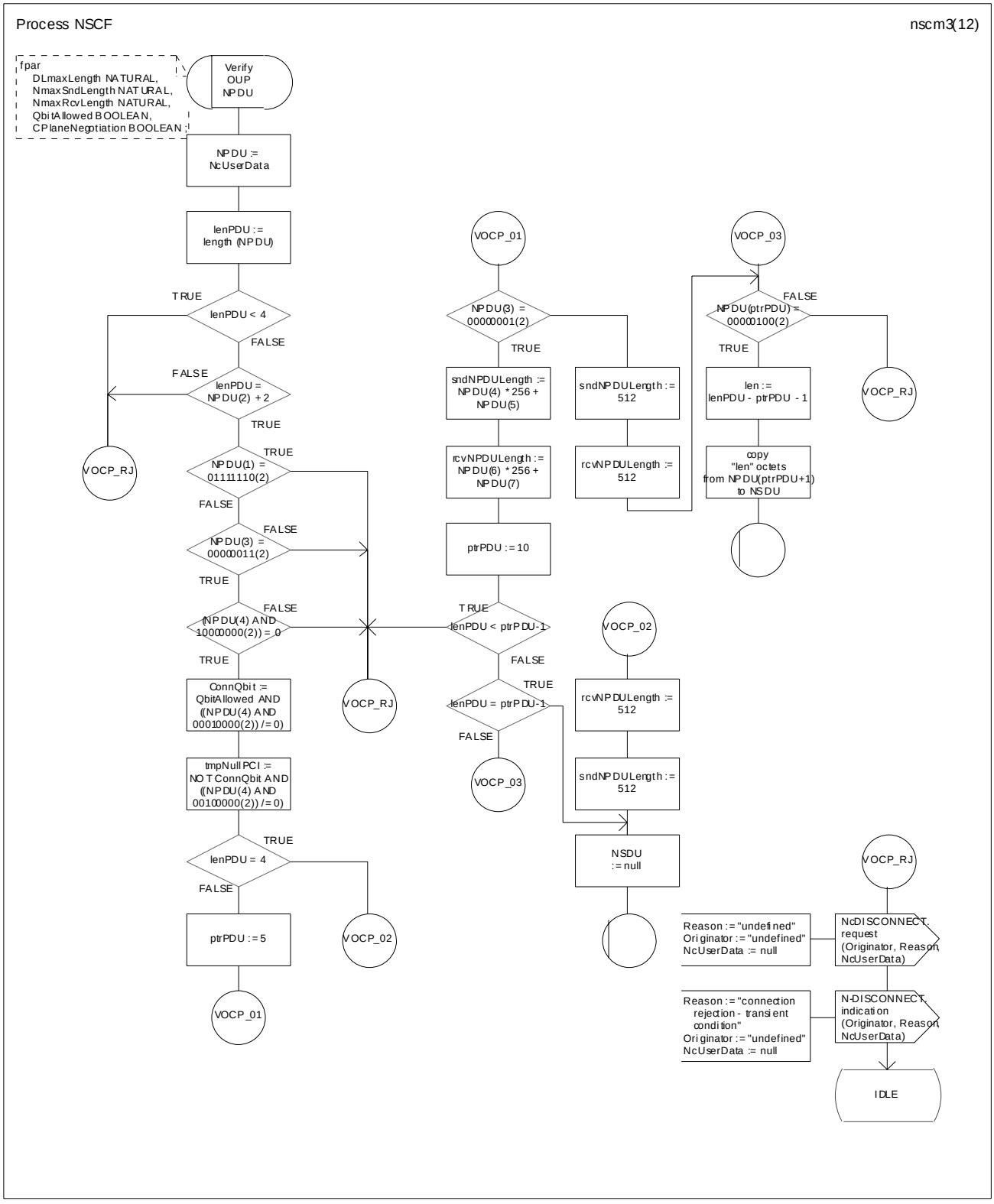


Figure D.3/X.633 (Sheet 11 of 12) – Synchronization and Coordination process

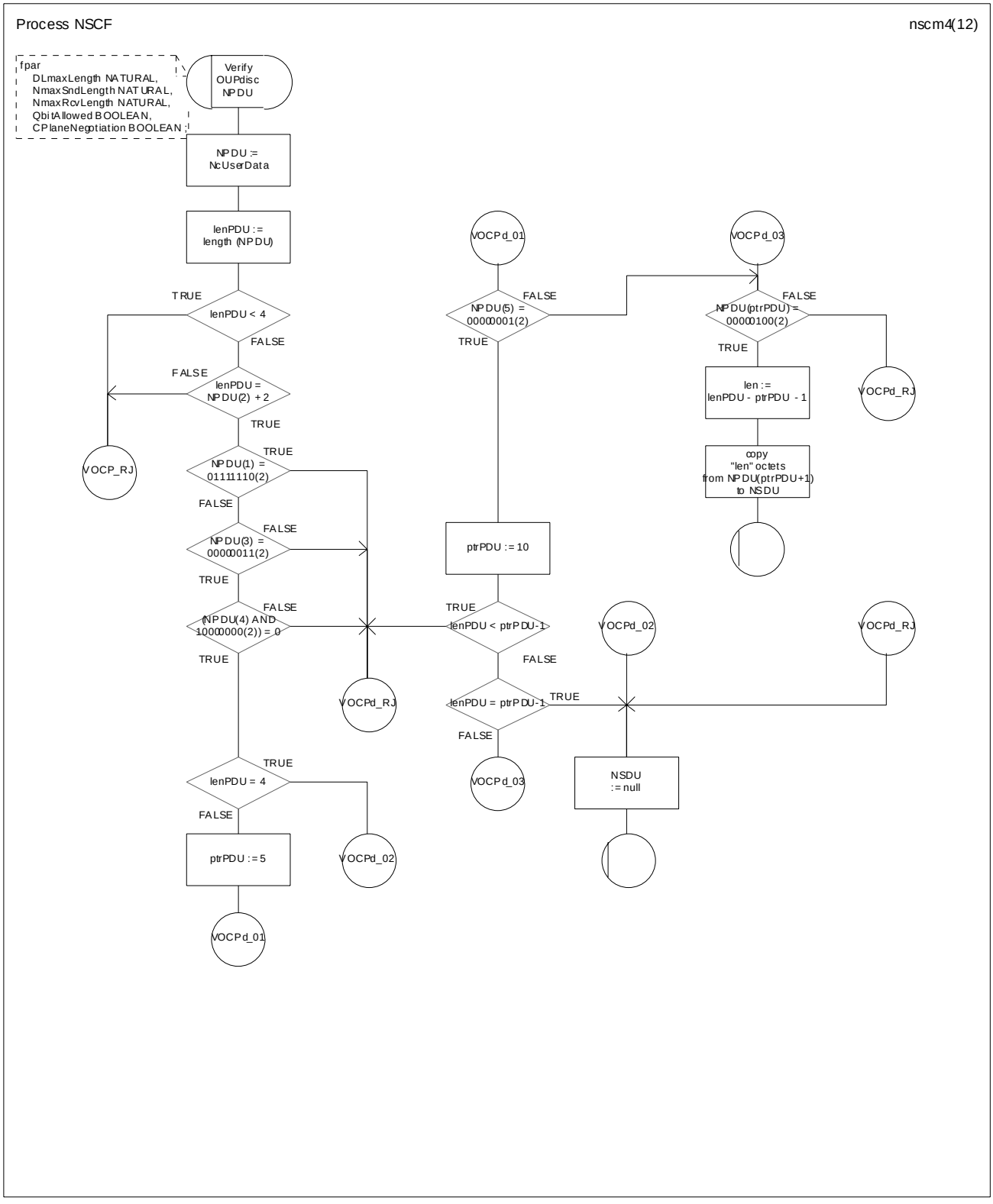


Figure D.3/X.633 (Sheet 12 of 12) – Synchronization and Coordination process

D.3.1 Procedures in the state "Idle"

1) Upon receipt of an N-CONNECT.request and if negotiation takes place in the C-plane, an NPDU is constructed that is then transferred in the User-to-User Information Element of the C-plane connection establishment request; the process enters state "Outgoing C-plane Connection Pending".

   The NPDU constructed follows the specification in Annex B. The NullPCI capability is set, the Q-bit is enabled if the parameter "QbitAllowed" is TRUE, and the NPDU lengths are taken from the parameter "NmaxSndLength" and "NmaxRcvLength". If NS-User-Data was also received with the primitive, this is also copied into the NPDU.

2) Upon receipt of an N-CONNECT.request and if negotiation takes place in the U-plane, a connection establishment is requested in the C-plane; the process enters state "Outgoing C-plane Connection Pending".

3) Upon receipt of an indication of a connection establishment in the C-plane and if negotiation takes place in the C-plane, the NPDU received in the User-to-User Information Element is verified and the data extracted. The data items are then sent in the parameters of the N-CONNECT.indication to the Synchronization and Coordination entity user; the process enters state "Incoming C-plane Connection Pending".

   The verification and extraction are specified in macro "Verify IUP NPDU" and are as follows:

   a) The User-to-User Information Element must be at least 4 octets long and the value in octet 2 must match the length of the User-to-User Information Element; otherwise, the connection establishment is rejected with a connection release request in the C-plane.

   b) If the first octet does not contain the value "01111110_2" or the third octet does not contain the value "00000011_2" or the extension bit in octet 4 is set, the connection establishment is rejected with a connection release request in the C-plane.

   c) The variable "ConnQbit" is set if Q-bit processing is supported and the Q-bit in octet 4 is set.

   d) The variable "tmpNullPCI" is set if variable "ConnQbit" is not set and the NullPCI-bit in octet 4 is set.

   e) If octet 5 contains the value "00000001_2", the next 4 octets contain negotiation values for the maximum NPDU length; otherwise the negotiation starts with the default maximum NPDU length values (512 octets).

   f) If the next octet contains the value "00000100_2", the information in the remaining octets is copied into the variable "NSDU" in preparation of sending it with the NS-User-Data in the N-CONNECT.indication primitive.

   If during the data extraction an improper NPDU is detected (e.g. a truncated NPDU), the connection establishment is rejected with a connection release request in the C-plane.

4) Upon receipt of an indication of a connection establishment in the C-plane and if negotiation takes place in the U-plane, a Network Fast Byte Protocol entity is created and the connection establishment in the C-plane is accepted; the process enters state "Incoming C-plane Connection Pending".

D.3.2 Procedures in the state "Outgoing C-plane Connection Pending"

1) Upon receipt of an indication of a confirmation of the connection establishment in the C-plane (and if negotiation takes place in the C-plane), the NPDU received in the User-to-User Information Element is verified and the data extracted. The data items are then used to set the connection parameters and are saved for later communication to the user. A Network Fast Byte Protocol entity is created and an N-CONNECT.request without parameters is sent to it; the process enters state "Incoming U-plane Connection Pending".

   The verification and extraction are specified in macro "Verify OUP NPDU" and are as follows:

   a) The User-to-User Information Element must be at least 4 octets long and the value in octet 2 must match the length of the User-to-User Information Element; otherwise, the connection establishment is rejected with a connection release request in the C-plane and the Synchronization and Coordination entity user is informed with an N-DISCONNECT.indication.

   b) If the first octet does not contain the value "01111110_2" or the third octet does not contain the value "00000011_2" or the extension bit in octet 4 is set, the connection establishment is rejected with a connection release request in the C-plane and the Synchronization and Coordination entity user is informed with an N-DISCONNECT.indication.

   c) The variable "ConnQbit" is set if Q-bit processing is supported and the Q-bit in octet 4 is set.

   d) The variable "tmpNullPCI" is set if variable "ConnQbit" is not set and the NullPCI-bit in octet 4 is set.

   e) If octet 5 contains the value "00000001_2", the next 4 octets contain negotiation values for the maximum NPDU length; otherwise, the negotiation starts with the default maximum NPDU length values (512 octets).

f) If the next octet contains the value "00000100_2", the information in the remaining octets is copied into the variable "NSDU" in preparation of sending it with the NS-User-Data in the N-CONNECT.indication primitive.

If during the data extraction an improper NPDU is detected (e.g. a truncated NPDU), the connection establishment is rejected with a connection release request in the C-plane and the Synchronization and Coordination entity user is informed with an N-DISCONNECT.indication.

2) Upon receipt of an indication of a confirmation of the connection establishment in the C-plane (and if negotiation takes place in the U-plane), a Network Fast Byte Protocol entity is created and the original N-CONNECT.request is sent to it; the process enters state "Incoming U-plane Connection Pending".

3) If an N-DISCONNECT.request primitive is received, the Synchronization and Coordination entity user abandons the connection establishment. If negotiation takes place in the C-plane, an NPDU is constructed that is then transferred in the User-to-User Information Element of a connection release request in the C-plane. Otherwise, a connection release in the C-plane without a User-to-User Information Element is requested. The process returns in both cases to state "Idle".

The NPDU constructed follows the specification in Annex B. The fourth octet of the header is set to zero. If NS-User-Data was also received with the primitive, this is also copied into the NPDU.

4) Upon receipt of an indication of a connection release in the C-plane and if negotiation takes place in the C-plane, the NPDU received in the User-to-User Information Element is verified and the data extracted. The data items are then sent in the parameters of the N-DISCONNECT.indication to the Synchronization and Coordination entity user; the process returns to state "Idle".

a) The User-to-User Information Element must be at least 4 octets long and the value in octet 2 must match the length of the User-to-User Information Element; otherwise, no data is extracted.

b) If the first octet does not contain the value "01111110_2" or the third octet does not contain the value "00000011_2" or the extension bit in octet 4 is set, no data is extracted.

c) If octet 5 contains the value "00000001_2", the next 4 octets contain negotiation values for the maximum NPDU length; these octets are skipped.

d) If the next octet contains the value "00000100_2", the information in the remaining octets is copied into the variable "NSDU" in preparation of sending it with the NS-User-Data in the N-CONNECT.indication primitive.

If during the data extraction an improper NPDU is detected (e.g. a truncated NPDU), no data is extracted.

5) Upon receipt of an indication of a connection release in the C-plane and if negotiation takes place in the U-plane, no NPDU is received in the User-to-User Information Element. An N-DISCONNECT.indication is sent to the Synchronization and Coordination entity user; the process returns to state "Idle".

D.3.3 Procedures in the state "Outgoing U-plane Connection Pending"

1) Upon receipt of an N-CONNECT.confirm, an N-CONNECT.confirm is sent to the Synchronization and Coordination entity user. If negotiation takes place in the C-plane, the parameters were saved in state "Outgoing C-plane Connection Pending"; otherwise, the parameters are taken from the primitive received. The process enters in both cases state "Data Transfer Ready".

2) If an N-DISCONNECT.request primitive is received, the Synchronization and Coordination entity user abandons the connection establishment. If negotiation takes place in the C-plane, an NPDU is constructed that is then transferred in the User-to-User Information Element of a connection release request in the C-plane; in addition, an N-DISCONNECT.request primitive is sent to the Network Fast Byte Protocol entity. The process returns to state "Idle".

The NPDU constructed follows the specification in Annex B. The fourth octet of the header is set to zero. If NS-User-Data was also received with the primitive, this is also copied into the NPDU.

3) If an N-DISCONNECT.request primitive is received and if negotiation takes place in the C-plane, no NPDU is constructed. The parameters are copied into an N-DISCONNECT.request primitive that is sent to the Network Fast Byte Protocol entity. Timer "TimerDC" is started and the process enters state "Outgoing Disconnection Pending".

4) Upon receipt of an indication of a connection release in the C-plane and if negotiation takes place in the C-plane, the NPDU received in the User-to-User Information Element is verified and the data extracted. The data items are then sent in the parameters of the N-DISCONNECT.indication to the Synchronization and Coordination entity user, an N-DISCONNECT.request is also sent to the Network Fast Byte Protocol entity. The process returns to state "Idle".

The verification and extraction is specified in macro "Verify OCPdisc NPDU" and is as follows:

a) The User-to-User Information Element must at least be 4 octets long and the value in octet 2 must match the length of the User-to-User Information Element; otherwise, no data is extracted.

b) If the first octet does not contain the value "01111110_2" or the third octet does not contain the value "00000011_2" or the extension bit in octet 4 is set, no data is extracted.

c) If octet 5 contains the value "00000001_2", the next 4 octets contain negotiation values for the maximum NPDU length; these octets are skipped.

d) If the next octet contains the value "00000100_2", the information in the remaining octets is copied into the variable "NSDU" in preparation of sending it with the NS-User-Data in the N-CONNECT.indication primitive.

If during the data extraction an improper NPDU is detected (e.g. a truncated NPDU), no data is extracted.

5) Upon receipt of an indication of a connection release in the C-plane and if negotiation takes place in the U-plane, no NPDU is received in the User-to-User Information Element. An N-DISCONNECT.indication without parameters is sent to the Synchronization and Coordination entity user, an N-DISCONNECT.request is also sent to the Network Fast Byte Protocol entity. The process returns to state "Idle".

6) If an N-DISCONNECT.indication primitive is received from the Network Fast Byte Protocol entity, an N-DISCONNECT.indication with the received parameters is sent to the Synchronization and Coordination entity user; a connection release in the C-plane is also requested. The process returns to state "Idle".

D.3.4 Procedures in the state "Incoming C-plane Connection Pending"

1) Upon receipt of an N-CONNECT.response from the Synchronization and Coordination entity user (this can only happen if negotiation takes place in the C-plane), the parameter values are used to set the connection parameters. An NPDU is constructed that is then transferred in the User-to-User Information Element of a connection acceptance request in the C-plane. A Network Fast Byte Protocol entity is created. The process enters state "Incoming U-plane Connection Pending".

   The NPDU constructed follows the specification in Annex B. The NullPCI capability is set, the Q-bit is enabled if the parameter "ConnQbit" is TRUE, and the NPDU lengths are taken from the parameter "ConnSndPDULength" and "ConnRcvPDULength". If NS-User-Data was also received with the primitive, this is also copied into the NPDU.

2) Upon receipt of an N-CONNECT.indication from the Network Fast Byte Protocol entity (this can only happen if negotiation takes place in the U-plane), the parameters are copied into the N-CONNECT.indication to the Synchronization and Coordination entity user; the process enters state "Incoming U-plane Connection Pending".

3) The procedures upon receipt of an N-DISCONNECT.request are described in D.3.3 items 2) and 3).

4) The procedures upon receipt of an indication of a connection release in the C-plane are described in D.3.3 items 4) and 5).

5) The procedures upon receipt of an N-DISCONNECT.indication from the Network Fast Byte Protocol entity are described in D.3.3 item 6).

D.3.5 Procedures in the state "Incoming U-plane Connection Pending"

1) Upon receipt of an N-CONNECT.indication from the Network Fast Byte Protocol entity (this can only happen if negotiation takes place in the C-plane), an N-CONNECT.response without parameters is sent to the Network Fast Byte Protocol entity. The process enters state "Data Transfer Ready".

2) Upon receipt of an N-CONNECT.response from the Synchronization and Coordination entity user (this can only happen if negotiation takes place in the U-plane), the parameters are copied into the N-CONNECT.response to the Network Fast Byte Protocol entity. The process enters state "Data Transfer Ready".

3) The primitives N-DATA.request and N-RESET.request are saved until the process enters state "Data Transfer Ready".

4) The procedures upon receipt of an N-DISCONNECT.request are described in D.3.3 items 2) and 3).

5) The procedures upon receipt of an indication of a connection release in the C-plane are described in D.3.3 items 4) and 5).

6) The procedures upon receipt of an N-DISCONNECT.indication from the Network Fast Byte Protocol entity are described in D.3.3 item 6).

D.3.6 Procedures in the state "Data Transfer Ready"

1) The procedures upon receipt of an N-DISCONNECT.request are described in D.3.3 items 2) and 3).

2) The procedures upon receipt of an indication of a connection release in the C-plane are described in D.3.3 items 4) and 5).

3) The procedures upon receipt of an N-DISCONNECT.indication from the Network Fast Byte Protocol entity are described in D.3.3 item 6).

4) Upon receipt of an N-DATA.request, an N-RESET.request, or an N-RESET.response primitive, the respective primitive with the same parameter values is sent to the Network Fast Byte Protocol entity; the process remains in state "Data Transfer Ready".

5) Upon receipt of an N-DATA.indication, an N-RESET.indication, or an N-RESET.confirm primitive, the respective primitive with the same parameter values is sent to the Synchronization and Coordination entity user; the process remains in state "Data Transfer Ready".

D.3.7 Procedures in the state "Outgoing Disconnection Pending"

1) Upon receipt of an indication of a connection release in the C-plane, timer "TimerDC" is reset and the process returns to state "Idle".

2) Upon expiry of timer "TimerDC", a connection release in the C-plane without parameters is requested and the process returns to state "Idle".

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the 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
Series Z	Languages and general software aspects for telecommunication systems