5i’

Recommendation Q.714

SIGNALLING CONNECTION CONTROL PART PROCEDURES

1 Introduction

1.1 *General characteristics of signalling connection control procedures*

1.1.1 *Purpose*

This Recommendation describes the procedures performed by the Signalling Connection Control Part (SCCP) of Signalling System No. 7 to provide both connection-oriented and connectionless network services, and SCCP management services as defined in Recommendation Q.711. These procedures make use of the messages and information elements defined in Recommendation Q.712, whose formatting and coding aspects are specified in Recommendation Q.713.

1.1.2 *Protocol classes*

The protocol used by the SCCP to provide network services is subdivided into four protocol classes, defined as follows:

— Class 0: Basic connectionless class

— Class 1: Sequenced (MTP) connectionless class

— Class 2: Basic connection-oriented class

— Class 3: Flow control connection-oriented class

Due to the ongoing studies on the SCCP called and calling party address, the maximum of this value needs further study. It is also noted that the transfer of up to 255 octets of user data is allowed when the SCCP called and calling party address do not include global title.

The connectionless protocol classes provide those capabilities that are necessary to transfer one Network Service Data Unit (NSDU), (i.e., one user-to-user information block) in the user data field of a *Unitdata* message. The maximum length of a NSDU is restricted to X octets , since segmenting and reassembly are not provided by protocol classes 0 and 1.

The connection-oriented protocol classes (protocol classes 2 and 3) provide segmenting and reassembly capabilities. If a Network Service Data Unit is longer than 255 octets, it is split into multiple segments at the originating node, prior to transfer in the ‘‘data’’ field of *Data* messages. Each segment is less than or equal to 255 octets. At the destination node, the NSDU is reassembled.

1.1.2.1 *Protocol class 0*

Network Service Data Units passed by higher layers to the SCCP in the node of origin are delivered by the SCCP to higher layers in the destination node. They are transported independently of each other. Therefore, they may be delivered out-of-sequence. Thus, this protocol class corresponds to a pure connectionless network service.

1.1.2.2 *Protocol class 1*

In protocol class 1, the features of class 0 are complemented by an additional feature (i.e., sequence control parameter associated with the N-UNITDATA request primitive) which allows the higher layer to indicate to the SCCP that a given stream of NSDUs have to be delivered in-sequence. The

Signalling Link Selection (SLS) field is chosen by SCCP based on the value of the sequence control parameter. The SLS chosen for a stream of NSDUs with the same sequence control parameter will be identical. The SCCP will then encode the Signalling Link Selection (SLS) field in the routing label of messages relating to such NSDUs, so that their sequence is, under normal conditions, maintained by the signalling network as defined in Recommendation Q.704. Thus, this class corresponds to an enhanced connectionless service, where an additional sequencing feature is included.

1.1.2.3 *Protocol class 2*

In protocol class 2, bidirectional transfer of NSDUs between the user of the SCCP in the node of origin and the user of the SCCP in the destination node is performed by setting up a temporary or permanent signalling connection. A number of signalling connections may be multiplexed onto the same signalling relation, as defined in Recommendation Q.704; such a multiplexing is performed by using a pair of reference numbers, referred to as ‘‘local reference numbers’’. Messages belonging to a given signalling connection will contain the same value of the SLS field to ensure sequencing as described in § 1.1.2.2. Thus, this protocol class corresponds to a simple connection-oriented network service, where SCCP flow control and missequence detection are not provided.

1.1.2.4 *Protocol class 3*

In protocol class 3, the features of protocol class 2 are complemented by the inclusion of flow control, with its associated capability of expedited data transfer. Moreover, an additional capability of detection of message loss or mis-sequencing is included; in such a circumstance, the

signalling connection is reset and a corresponding notification is given by the SCCP to the higher layers.

1.1.3 *Signalling connections*

In all connection-oriented protocol classes, a signalling connection between the nodes of origin and destination may consist of:

— a single connection section, or

— a number of connection sections in tandem, which may belong to different interconnected signalling networks.

In the former case, the originating and destination nodes of the signalling connection coincide with the originating and destination nodes of a connection section. During the connection establishment phase, SCCP routing and relaying functions, as described in § 2 of this Recommendation, may be required at one or more intermediate nodes. Once the signalling connection has been established, though, SCCP functions are not required at intermediate nodes.

In the latter case, at any intermediate node where a message is received from a connection section and has to be sent on another connection section, the SCCP routing and relaying functions are involved during connection establishment. In addition, SCCP functions are required at intermediate nodes during Data Transfer and Connection Release to provide the association of connection sections.

1.1.4 *Compatibility and handling of unrecognized information*

1.1.4.1 *Rules for forward compatibility*

All implementations must recognize all messages in each protocol class offered, as indicated in Table 1/Q.713.

General rules for forward compatibility are specified in Recommendation Q.700.

1.1.4.2 *Handling of unrecognized messages or parameters*

Any message with unrecognized message type value should be discarded. Any unrecognized parameter within a message should be ignored. Notification to the originator of the message in these two cases is for further study.

1.2 *Overview of procedures for connection-oriented services*

1.2.1 *Connection establishment*

When the SCCP functions at the node of origin receive a request to establish a signalling connection, the ‘‘called party address’’ is analyzed to identify the node towards which a signalling connection should be established. The SCCP forwards a *Connection Request* (CR) message to that signalling point, using the MTP functions.

The SCCP in the node receiving the CR message via the MTP functions examines the ‘‘called party address’’ and one of the following actions takes place at the node:

a) If the ‘‘called party address’’ contained in the CR message corresponds to a user located in that signalling point and if the signalling connection may be established (i,e., establishment of a signalling connection is agreed to by the SCCP and local user), a *Connection Confirm* (CC) message is returned.

b) If the ‘‘called party address’’ does not correspond to a user at the signalling point, then information available in the message and at the node are examined to determine whether

an association of two connection sections is required at that node.

— If an association is required, then the SCCP establishes an (incoming) signalling connection section. Establishment of another (outgoing) connection section is initiated by sending a CR message towards the next node and this connection section is logically linked to the incoming connection section.

— If coupling of the connection section is not required in this node, no incoming or outgoing connection section is established. A CR message is forwarded towards the next destination using the MTP routing function.

If the SCCP receives a CR message and either the SCCP or the SCCP user does not wish to establish the connection, then a *Connection Refused* (CREF) message is transferred on the incoming connection section.

On receipt of a CC message, the SCCP completes the set-up of a connection section. Furthermore, if coupling of two adjacent connection sections is needed, a further CC message is forwarded to the preceding node.

If no coupling of adjacent connection sections was needed during set-up in the forward direction, the CC message can be sent directly to the node of origin even if a number of intermediate SCCP nodes was passed in the forward direction. The OPC of the node of origin is transmitted within the ‘‘calling party address’’ Field.

When the CR and CC messages have been exchanged between all the involved nodes as above described, and the corresponding indications have been given to the higher layer functions in the nodes of origin and destination, then the signalling connection is established and transmission of messages may commence.

1.2.2 *Data transfer*

Transfer of each NSDU is performed by one or more *Data* | DT) messages; a *more-data* | ndication is used if the NSDU is to be split among more than one DT message. If protocol class 3 is used, then SCCP flow control is utilized over each connection section of the signalling connection. If, in such a protocol class, abnormal conditions are detected, then the appropriate actions are taken on the signalling connection

(e.g., reset). Moreover in such a protocol class, expedited data may be sent using one *Expedited Data* message that bypasses the flow control procedures applying to *Data* messages.

A limited amount of data may also be transferred in the *Connection Request* , *Connection Refused* | nd *Connection Released* messages.

1.2.3 *Connection release*

When the signalling connection is terminated, a release sequence takes place by means of two messages called *Released* | nd *Release Complete* (RLC). The RLC message is normally sent in reaction to the receipt of a RLSD message.

1.3 *Overview of procedures for connectionless services*

1.3.1 *General*

When the SCCP functions at the node of origin receive from an SCCP user an NSDU to be transferred by the protocol class 0 or 1 connectionless service, the ‘‘called party address’’ and other relevant parameters, if required, are analyzed to identify the node towards which the message should be sent. The NSDU is then included as the ‘‘data’’ parameter in a *Unitdata* (UDT) message,

which is sent towards the node using the MTP functions. Upon receipt of the UDT message, the SCCP functions at that node perform the routing analysis as described in § 2 of this Recommendation and, if the destination of the UDT message is a local user, deliver the NSDU to the local higher layer functions. If the ‘‘called party address’’ is not at that node, then the UDT message is forwarded to the next node. This process continues until the NSDU has reached the ‘‘called party address’’.

1.4 *Structure of the SCCP and contents of specification*

The basic structure of the SCCP appears in Figure 1/Q.714. It consists of four functional blocks as follows:

a) *SCCP connection-oriented control:* Its purpose is to control the establishment and release of signalling connections and to provide for data transfer on signalling connections.

b) *SCCP connectionless control:* Its purpose is to provide for the connectionless transfer of data units.

c) *SCCP management:* Its purpose is to provide capabilities, in addition to the Signalling Route Management and flow control

functions of the MTP, to handle the congestion or failure of either the SCCP user or the signalling route to the SCCP user.

d) *SCCP routing:* Upon receipt of a message from the MTP or from functions a) or b) above, SCCP routing provides the necessary routing functions to either forward the message to the MTP for transfer, or pass the message to functions a) or b) above. A message whose ‘‘called party address’’ is a local user is passed to functions a or b, while one destined for a remote user is forwarded to the MTP for transfer to a distant SCCP user.

Section 2 of this specification describes the addressing and routing functions performed by the SCCP. Section 3 specifies the procedures for the connection-oriented services (protocol classes 2-3). Section 4 specifies the procedures for the connectionless services (protocol classes 0 and 1). Section 5 specifies the SCCP management procedures.

2 Addressing and routing

2.1 *SCCP addressing*

The ‘‘called and calling party addresses’’ contain the information necessary for the SCCP to determine an originating and destination node. In the case of the connection-oriented procedures, the addresses are the originating and destination points of the signalling connection, while in the case of the connectionless procedures, the addresses are the originating and destination points of the message.

When transferring connection-oriented or connectionless messages, two basic categories of addresses are distinguished by SCCP routing:

1) *Global Title* — A global title is an address, such as dialled-digits, which does not explicitly contain information that would allow routing in the signalling network, that is the translation function of the SCCP is required. This translation function could be performed on a distributed basis or on a centralized basis. The last case, where a request for translation is sent to a centralized data base, may be

accomplished, for example, with transaction capabilities (TC). This matter is for further study.

In case of an E.164-based global title with the nature of address indicator included, the sending sequence of address information will be the country code, followed by the national (significant) number. Within the destination signalling network, the address information may be the subscriber number or the national (significant) number, by choice of the value of nature of address indicator, as required by the administration concerned.

2) *DPC + SSN* — A Destination Point Code and Subsystem Number allows direct routing by the SCCP and MTP, that is, the translation function of the SCCP is not required.

2.2 *SCCP routing principles*

The SCCP routing control (SCRC) receives messages from the Message Transfer Part for routing and discrimination, after they have been received by the MTP from another node in the signalling network. SCRC also receives

internal messages from SCCP connection-oriented control (SCOC) and connectionless control (SCLC) and performs any necessary routing functions (e.g., address translation) before passing them to the MTP for transport in the signalling network or back to the SCCP connection-oriented or connectionless control.

Figure 1/Q.714, p.

2.2.1 *Receipt of SCCP message transferred by the MTP*

A message transferred by the MTP that requires routing will include the ‘‘called party address’’ parameter giving information for routing the message. These messages currently include the *Connection Request* message and all types of connectionless messages. Other messages are passed to connection-oriented control for processing.

If the ‘‘called party address’’ parameter is used for routing, it can take the following information:

1) *Subsystem Number (SSN) only* — This indicates that the receiving SCCP is the termination point of the message. The SSN is used to determine the local subsystem.

2) *Global Title (GT) only* — This indicates that translation is required. Translation of the Global Title results in a new destination point code (DPC) for routing the message, and possibly a new SSN or GT or both in the ‘‘called party address’’.

3) *SSN + GT* — In this case, the address indicator information is used to determine whether the SSN or the GT should be used for routing and processing via items 1) or 2) above, respectively.

2.2.2 *Messages from connection-oriented or connectionless control to SCCP routing control*

Addressing information, indicating the destination of the message, is included with every internal message received from connection-oriented or connectionless control. For connectionless messages, this addressing information is obtained from the ‘‘called address’’ parameter associated with the N-UNITDATA REQUEST primitive. For *Connection-Request* messages received by SCCP routing, the addressing information is obtained from the ‘‘Called address’’ parameter associated with the N-CONNECT REQUEST primitive. For connection-oriented messages other than a *Connection Request* message, the

addressing information (i.e., the DPC) is that associated with the connection section. The addressing information can take the following forms:

1) DPC

2) DPC + (SSN or GT or both)

3) GT

4) GT + SSN

The first form applies to connection-oriented messages except the *Connection Request* | message. The last three forms apply to connectionless messages and to the *Connection Request* message.

2.2.2.1 *DPC present*

If the DPC is present in the addressing information, then the DPC is passed to the MTP using the MTP-TRANSFER request primitive and:

1) if no other addressing information is available, no ‘‘called party address’’ is provided in the message;

2) if a SSN or GT or both are available, this information is used in the ‘‘called party address’’ with an indication of which is to be used for routing.

If the DPC is the node itself, then the information is passed to the specified internal subsystem.

2.2.2.2 *Translation required*

If the DPC is not present, then a global title translation is required before the message can be sent out. Translation results in a DPC and possibly a new SSN or new GT or both. If the GT and/or SSN resulting from a global title translation is different from the GT and/or SSN which was previously included in the called party address, the newly produced GT and/or SSN replaces the existing one. The routing procedures then continue as per § 2.2.2.1

2.3 *SCCP routing*

The SCCP routing functions are based on information contained in the ‘‘called party address’’.

2.3.1 *Receipt of SCCP message transferred by the MTP*

One of the following actions is taken by SCCP routing upon receipt of a message from the Message Transfer Part. The message is received by the SCCP when the MTP invokes a MTP-TRANSFER INDICATION.

1) If the message is a connection-oriented message other than a *Connection Request* | (CR) message, then SCCP routing passes the message to connection-oriented control.

2) If the routing indicator in the ‘‘called party address’’ does not indicate route on global title, then SCCP routing checks the status of the subsystem.

a) If the subsystem is available, the message is passed, based on the message type, to either connection-oriented control or connectionless control.

b) If the system is unavailable and:

— the message is a connectionless message, then the message return procedure is initiated;

— the message is a connection-oriented message (a CR message), then the connection refusal procedure is initiated.

In addition, if the subsystem is failed, SCCP management is notified that a message was received for a failed subsystem.

3) If the routing indicator in the ‘‘called party address’’ indicates route on global title, a translation of the global title must be performed.

a) If the translation of the global title exists, and both the DPC and SSN are determined, then:

i) if the DPC is the node itself, then the procedures in 2) above are followed;

ii) if the DPC is not the node itself, the DPC and SSN are available, and the message is a connectionless message, then the MTP-TRANSFER REQUEST primitive is invoked;

iii) if the DPC is not the node itself, the DPC and SSN are available, and the message is a connection-oriented message, then:

— if an association of connection sections is required, the message is passed to connection-oriented control;

— if no association of connection sections is required, the MTP-TRANSFER REQUEST primitive is invoked;

iv) if the DPC is not the node itself, and the DPC and/or SSN are not available and

— the message is a connectionless message, then the message return procedure is initiated;

— the message is a connection-oriented message (a CR message), then the connection refusal procedure is initiated.

b) If the translation of the global title exists and only a DPC or DPC + new GT is determined, then:

i) if the DPC is available, and the message is a connectionless message, then the MTP-TRANSFER REQUEST primitive is invoked;

ii) if the DPC is available, and the message is a connection-oriented message, then:

— if an association of the connection sections is required, then the message is passed to connection-oriented control;

— if no association of the connection section is required, then the MTP-TRANSFER REQUEST primitive is invoked.

iii) if the DPC is not available and

— the message is a connectionless message, then the message return procedure is initiated;

— the message is a connection-oriented message (a CR message), then the connection refusal procedure is initiated.

c) If the translation of the global title does not exist, and

— the message is a connectionless message, then the message return procedure is initiated;

— the message is a connection-oriented message (a CR message), then the connection refusal procedure is initiated.

2.3.2 *Messages from connectionless or connection-oriented control to SCCP routing control*

One of the following actions is taken by SCCP routing upon receipt of a message from connectionless control or connection-oriented control.

1) If the message is a *Connection Request* | message at an intermediate node (where connection sections are being associated), and:

a) the DPC is available, then the MTP-TRANSFER REQUEST primitive is invoked;

b) the DPC is not available, then the connection refusal procedure is initiated.

2) If the message is a connection-oriented message other than a *Connection Request* | message, and:

a) the DPC is available, then the MTP-TRANSFER REQUEST primitive is invoked;

b) the DPC is not available, then the connection release procedure is initiated.

3) If the ‘‘Called address’’ in the primitive associated with a *Connection Request* | or connectionless message includes a DPC, and:

a) the DPC and SSN are available, then the MTP-TRANSFER REQUEST primitive is invoked;

b) the DPC and/or SSN are not available, then:

— for connectionless messages, the message return procedure is initiated;

— for connection-oriented messages (CR messages), the connection refusal procedure is initiated.

The function of routing between local subsystems is implementation dependent.

c) the DPC is the node itself, then the procedures in § 2.3.1, 2) above are followed.

4) If the ‘‘Called address’’ in the primitive associated with a *Connection Request* | or connectionless message does not include a DPC, then a translation of the global title must be performed.

a) If the translation of the global title exists, and both the DPC and SSN are determined, then:

i) if the DPC is the node itself, then the procedures in § 2.3.1, 2), above are followed.

ii) if the DPC is not the node itself and DPC and SSN are available, then the MTP-TRANSFER REQUEST primitive is invoked;

iii) if the DPC is not the node itself, and the DPC and/or SSN are not available and:

— the message is a connectionless message, then the message return procedure is initiated;

— the message is a connection-oriented message (a CR message), then the connection refusal procedure is initiated.

b) If the translation of the global title exists and only a DPC or DPC + new GT is determined, then

i) if the DPC is available, then the MTP-TRANSFER REQUEST primitive is invoked;

ii) if the DPC is not available and:

— the message is a connectionless message, then the message return procedure is initiated;

— the message is a connection-oriented message (a CR message), then the connection refusal procedure is initiated.

c) If the translation of the global title does not exist, and:

— the message is a connectionless message, then the message return procedure is initiated;

— the message is a connection-oriented message (a CR message), then the connection refusal procedure is initiated.

2.4 *Routing failures*

The SCCP recognizes a number of reasons for failure in SCCP routing control. Examples of these reasons are:

1) translation does not exist for addresses of this nature;

2) translation does not exist for this specific address;

3) network/subsystem failure;

4) network/subsystem congestion, and

5) unequipped user.

The precise classification of the causes by which such failures are recognized is for further study.

When SCCP routing is unable to transfer a message due to the unavailability of a Point Code or Subsystem, one of above reasons is indicated in the *Connection Refused* message, the *Connection Released* message or the *Unitdata Service* message.

3 Connection-oriented procedures

3.1 *Connection establishment*

3.1.1 *General*

The connection establishment procedures consist of the functions required to establish a temporary signalling connection between two users of the Signalling Connection Control Part.

The connection establishment procedures are initiated by a SCCP user by invoking the N-CONNECT request primitive.

The ISDN-UP may initiate an SCCP connection in the same way as any other user, but may also request the SCCP to initiate a connection and return the information to the ISDN-UP for transmission in a call set-up message.

The signalling connections between two users of the Signalling Connection Control Part, which are referred to by the ‘‘Called/Calling address’’ parameters in the N-CONNECT REQUEST primitive, may be realized by the establishment of one or more connection sections. The SCCP user is not aware of how the SCCP provides the signalling connection (e.g. by one or more than one connection sections).

The realization of a signalling connection between two SCCP users then can be described by the following components:

1) one or more connection sections;

2) an originating node, where the ‘‘Calling address’’ is located;

3) zero or more intermediate nodes, where, for this signalling connection, there is no distribution to a SCCP user; and

4) a destination node, where the ‘‘Called address’’ is located.

The *Connection Request* | essage and the *Connection Confirm* | essage are used to set up connection sections.

3.1.2 *Local reference numbers*

During connection establishment both a source and destination local reference number are assigned independently to a connection section.

Source and destination local reference numbers are assigned at connection section set-up for a permanent connection section.

Once the destination reference number is known, it is a mandatory field for all messages transferred on that connection section.

Each node will select the local reference that will be used by the remote node as the destination local reference number field on a connection section for data transfer.

The local reference numbers remain unavailable for use on other connection sections until the connection section is released and the reference numbers are removed from their frozen state. See also § 3.3.2.

3.1.3 *Negotiation procedures*

3.1.3.1 *Protocol class negotiation*

During connection establishment it is possible to negotiate the protocol class of a signalling connection between two SCCP users.

The N-CONNECT REQUEST primitive contains a parameter, the ‘‘Quality of service parameter set’’, with the preferred quality of service proposed by the SCCP user for the signalling connection.

The SCCP at the originating, intermediate and destination nodes may alter the protocol class on a signalling connection so that the quality of service assigned to the signalling connection is less restrictive (e.g., a protocol class 2 connection may be provided if a protocol class 3

connection is proposed). Information concerning the present proposed protocol class within the SCCP is carried in the *Connection Request* message and the assigned protocol class appears in the *Connection Confirm* message.

At the destination node the SCCP user is notified of the proposed protocol class using the N-CONNECT INDICATION primitive.

The protocol class of a signalling connection may also be altered by the Called SCCP user in the same manner (i.e. less restrictive) when invoking the N-CONNECT RESPONSE primitive.

The Calling SCCP user is informed of the quality of service selected on the signalling connection using the N-CONNECT CONFIRMATION primitive.

3.1.3.2 *Flow control credit negotiation*

During connection establishment it is possible to negotiate the window size to be used on a signalling connection for the purpose of flow control. This window size remains fixed for the life of the signalling connection. The credit field in the CONNECTION REQUEST and CONNECTION CONFIRM messages is used to indicate the window size.

The N-CONNECT REQUEST primitive contains a parameter, the ‘‘Quality of service parameter set’’ with the preferred quality of service proposed by the SCCP user for the signalling connection.

The SCCP at the originating, intermediate and destination nodes may alter the window size on a signalling connection so that the quality of service assigned to the signalling connection is less restrictive (e.g., a smaller window size may be provided). Information concerning the present proposed window size within the SCCP is carried in the *Connection Request* message and the assigned window size appears in the *Connection Confirm* message.

At the destination node the SCCP user is notified of the proposed window size using the N-CONNECT indication primitive.

The window size of a signalling connection may also be altered by the Called SCCP user in the same manner (i.e. less restrictive) when invoking the N-CONNECT RESPONSE primitive.

The Calling SCCP user is informed of the quality of service selected on the signalling connection using the N-CONNECT confirm primitive.

3.1.4 *Actions at the origination node*

3.1.4.1 *Initial actions*

The N-CONNECT REQUEST primitive is invoked by the SCCP user at the originating node to request the establishment of a signalling connection to the ‘‘Called address’’ contained in the primitive. The node determines if resources are available.

If resources are not available, then the connection refusal procedure is initiated.

If resources are available, then the following actions take place at the originating node:

1) A source local reference number and an SLS code are assigned to the connection section.

2) The ‘‘Called address’’ is associated with the connection section.

3) The proposed protocol class is determined for the connection section.

4) If the protocol class provides for flow control, then an initial credit is indicated in the *Connection Request* message.

5) The *Connection Request* | essage is then forwarded to the SCCP routing function for transfer.

6) A timer T(conn est) is started.

The ISDN-UP may request the SCCP to set up a SCCP signalling connection and return the information normally carried in a *Connection request* | message to the ISDN-UP for transmission in a call set-up message.

When the ISDN-UP notifies the SCCP of the need for the connection, using the REQUEST Type 1 interface element, the SCCP determines if resources are available.

If resources are not available, then the connection refusal procedure is initiated. If resources are available, then the following actions take place at the origination node:

1) A source local reference number and an SLS code is assigned to the connection section.

2) An indication that the call request is from the ISDN-UP is associated with the connection section.

3) The proposed protocol class is determined for the connection section.

4) If the protocol class provides for flow control, then an initial credit is indicated.

5) The information that would normally be included in a Connection Request message is passed to the ISDN-UP for transfer using the REPLY interface element.

6) A timer T(conn est) is started.

3.1.4.2 *Subsequent actions*

When an originating node receives a *Connection Confirm* | essage, the following actions are performed:

1) The protocol class and initial credit for flow control of the connection section are updated if necessary.

2) The SCCP user is informed of the successful establishment of the signalling connection using the N-CONNECT CONFIRMATION primitive.

3) The received local reference number is associated with the connection section.

4) The timer T(conn est) is stopped.

5) The inactivity control timers, T(ias) and T(iar), are started.

When the SCCP user at an origination node invokes the N-DISCONNECT REQUEST primitive, no action is taken prior to receipt of a *Connection Confirm* or a *Connection Refused* message or expiration of the connection establishment timer.

When an originating node receives a *Connection Refused* | essage, the connection refusal procedure is completed at the origination node (see § 3.2.3).

When the connection establishment timer at the origination node expires, the N-DISCONNECT INDICATION primitive is invoked, the resources associated with the connection section are released, and the local reference number is frozen.

3.1.5 *Actions at an intermediate node*

3.1.5.1 *Initial actions*

When a *Connection Request* | essage is received at a node and the SCCP routing and discrimination function determines that the ‘‘called party

address’’ is not a local SCCP user and that a coupling is required at this node, the intermediate node determines if resources are available to establish the connection section.

If resources are not available at the node, then the connection refusal procedure is initiated.

If resources are available at the node, then the following actions are performed:

1) A local reference number and an SLS code are assigned to the incoming connection section. (*Note* — As an implementation option, a local reference number may be assigned later upon reception of a *Connection Confirm* message.)

2) A connection section is set up to the remote node determined by SCCP Routing:

— A local reference number and an SLS code are assigned to the outgoing connection section.

— The protocol class is proposed.

— An initial credit for flow control is assigned, if appropriate.

— The *Connection Request* | essage is forwarded to the SCCP Routing with the same addressing information as found in the incoming *Connection Request* message.

— The timer T(conn est) is started.

3) An association is made between the incoming and outgoing connection sections.

The ISDN-UP informs the SCCP that a connection request has been received using the REQUEST Type 2 interface element. The ISDN-UP passes the information contained in the ISDN-UP set-up message to the SCCP and indicates that a coupling is required at this node. The SCCP at the intermediate node then determines if resources are available to establish the connection section.

If resources are not available at the node, then the connection refusal procedure is initiated.

If resources are available at the node, then the following actions are performed:

1) A local reference number and an SLS code are assigned to the incoming connection section.

2) A local reference number and an SLS code is assigned to an outgoing connection section.

3) A protocol class is proposed.

4) An initial credit for flow control is assigned if appropriate.

5) An association is made between the incoming and outgoing connection sections.

6) The information that would normally be included in a conection request message is passed to the ISDN-UP for transfer in the REPLY interface element.

7) The timer T(conn est) is started.

3.1.5.2 *Subsequent actions*

When an intermediate node receives a *Connection Confirm* | essage, the following actions are performed:

1) The source local reference number in the *Connection Confirm* | essage is associated with the outgoing connection section.

2) The protocol class and credit for the connection section are assigned and identical to those found in the received *Connection Confirm* message.

3) A *Connection Confirm* | essage is transferred, using the SCCP routing function, to the originating node of the associated connection section. The protocol class and credit are identical to those indicated in the received *Connection Confirm* message.

4) The timer T(conn est) is stopped.

5) The inactivity control timers, T(ias) and T(iar), are started.

When an intermediate node receives a *Connection Refused*
| essage, the connection refusal procedure is completed at that node (see § 3.2.2).

When the connection establishment timer expires at an intermediate node, the following actions are performed:

1) The resources associated with the connection are released.

2) The local reference number is frozen (see § 3.3.2).

3) If the connection section was established using a REQUEST interface element, then the N-DISCONNECT INDICATION primitive is invoked.

4) The connection refusal procedure is initiated on the associated connection section (see § 3.2.1).

3.1.6 *Actions at destination node*

3.1.6.1 *Initial actions*

When a *Connection Request* | essage is received at a node, and the SCCP routing and discrimination function determines that the ‘‘called party address’’ is a local user, the destination node determines if resources are available to establish the connection section.

If resources are not available at the node, then the connection refusal procedure is initiated.

If the resources are available at the node, then the following actions are performed:

1) The protocol class is determined for the connection section. (*Note* — As an implementation option, a local reference number may also be assigned for the connection section.)

2) An initial credit for flow control is assigned if appropriate.

3) The node informs the SCCP user of a request to establish a connection using the N-CONNECT INDICATION primitive.

When the ISDN-UP informs the SCCP that a connection request has been received using the REQUEST Type 2 interface element, the ISDN-UP passes the information contained in the ISDN-UP set-up message to the SCCP, and informs the SCCP that the information is for a local user. The SCCP at the destination node determines if resources are available to establish the connection section.

If resources are not available at the node, then the connection refusal procedure is initiated.

If resources are available at the node, then the following actions are performed:

1) The protocol class is determined for the connection section.

2) An initial credit for flow control is assigned if appropriate.

3) The node informs the ISDN-UP of the request to establish a connection using the N-CONNECT INDICATION primitive.

3.1.6.2 *Subsequent actions*

When a N-CONNECT RESPONSE primitive is invoked by the SCCP user at a destination node, the following actions are performed:

1) A local reference number and an SLS code are assigned to the incoming connection section.

2) The protocol class and credit are updated for the connection section if necessary.

3) A *Connection Confirm* | message is transferred, using the SCCP routing function, to the originating node of the connection section.

4) The inactivity control timers, T(ias) and T(iar), are started.

3.2 *Connection refusal*

The purpose of the connection refusal procedure is to indicate to the Calling SCCP user function that the attempt to set up a signalling connection section was unsuccessful.

3.2.1 *Actions at node initiating connection refusal*

The connection refusal procedure is initiated by either the SCCP user or the SCCP itself:

1) The SCCP user at the destination node

a) uses the N-DISCONNECT REQUEST (originator indicates ‘‘user initiated’’) after the SCCP has invoked an N-CONNECT indication primitive. This is the case when the SCCP at the destination node has received the connection request directly from a preceding SCCP.

b) uses the refusal indicator in the REQUEST Type 2 interface element when the SCCP user has received the connection request embedded in a user part message.

If the reason for the refusal is ‘‘destination address unknown’’, then the maintenance function is alerted.

2) The SCCP initiates connection refusal (originator indicates ‘‘network initiated’’) due to:

a) limited resources at an originating, intermediate or destination node, or

b) expiration of the connection establishment timer at an originating or intermediate node.

Initiation of the connection refusal procedure by either the SCCP or the user results in the transfer of a *Connection Refused* | essage on the connection section. The refusal cause contains the value of the originator in the primitives; if the refusal procedure has been initiated by using the refusal indicator in the REQUEST Type 2 interface element, the refusal cause contains ‘‘SCCP user originated’’.

At the originating node, a connection refusal is initiated by invoking N-DISCONNECT INDICATION primitive.

If the connection refusal procedure is initiated at an intermediate node due to lack of resources, then a *Connection Refused* | message is transferred on the incoming connection section.

If the connection refusal procedure is initiated at an intermediate node due to expiration of the connection establishment timer, then the connection release procedure is initiated on that connection section (see § 3.3.4.1) and a *Connection Refused* message is transferred on the associated connection section.

In either of the two above cases at an intermediate node, if the connection set-up was initiated using a REQUEST interface element, then the SCCP user is informed by invoking the N-DISCONNECT INDICATION primitive.

3.2.2 *Actions at intermediate node not initiating connection refusal*

When a *Connection Refused* | essage is received on a connection section, the following actions are performed:

1) The resources associated with the connection section are released and the timer T(conn est) is stopped

2) If the connection was established using a REQUEST interface element, then the SCCP user is informed by invoking the N-DISCONNECT INDICATION primitive.

3) A *Connection Refused* | essage is transferred on the associated connection section.

4) The resources associated with the associated connection section are released.

3.2.3 *Actions at the origination node not initiating connection refusal*

When a *Connection Refused* | essage is received on a connection section, the following actions are performed:

1) The resources associated with the connection section are released and the timer T(conn est) is stopped

2) The SCCP user is informed by invoking the N-DISCONNECT INDICATION primitive.

3.3 *Connection release*

3.3.1 *General*

The connection release procedures consist of the functions required to release a temporary signalling connection between two users of the Signalling Connection Control Part. Two messages are required to initiate and complete connection release: *Released* and *Release Complete*

The release may be performed:

a) by either or both of the SCCP users to release an established connection.

b) by the SCCP to release an established connection.

All failures to maintain a connection are indicated in this way.

3.3.2 *Frozen reference*

The purpose of the frozen reference function is to prevent the initiation of incorrect procedures as a connection section due to receipt of a message, which is associated with a previously established connection section.

When a connection section is released, the local reference number associated with the connection section is not immediately available for reuse on another connection section. A mechanism should be chosen to sufficiently

reduce the probability of erroneously associating a message with a connection section. This particular mechanism is implementation dependent.

3.3.3 *Actions at an end node initiating connection release*

3.3.3.1 *Initial actions*

When a connection release is initiated at an end node of a signalling connection, by the SCCP user invoking a N-DISCONNECT REQUEST primitive or by the node itself, the following actions are performed at the initiating node:

1) A *Released* | essage is transferred on the connection section.

2) A release timer T(rel) is started.

3) If the release was initiated by the SCCP, then a N-DISCONNECT INDICATION primitive is invoked.

4) The inactivity control timers, T(ias) and T(iar), if still running, are stopped.

3.3.3.2 *Subsequent actions*

The following actions are performed at the originating node on a connection section for which a *Released* | essage has been previously transferred:

1) When a *Release Complete* | r *Released* | essage is received, the resources associated with the connection are released, the timer, T(rel), is stopped, and the local reference number is frozen.

2) When the release timer expires, a *Released* | essage is transferred on the connection section. The sending of the *Released* message is repeated every 4-15 seconds for an interval of up to one minute. At this point a maintenance function is alerted.

3.3.4 *Actions at intermediate node*

The connection release procedure is initiated at an intermediate node by the SCCP or by reception of a *Released* | essage on a connection section.

3.3.4.1 *Initial actions*

When a *Released* | essage is received on a connection section, the following actions then take place:

1) A *Release Complete* | essage is transferred on the connection section, the resources associated with the connection are released and the local reference number is frozen.

2) A *Released* | essage is transferred on the associated connection section; the reason is identical to the reason in the received message.

3) If the connection was established using a REQUEST interface element, then a N-DISCONNECT INDICATION primitive is invoked.

4) The release timer, T(rel), is started on the associated connection.

5) The inactivity control timers, T(ias) and T(iar), if still running, are stopped on both connection sections.

When the connection release procedure is initiated by the SCCP at the intermediate node during the data transfer phase, the following actions take place on both of the connection sections:

1) A *Released* | essage is transferred on the connection section.

2) If the connection section was established using an interface element, then a N-DISCONNECT INDICATION primitive is invoked.

3) The release timer, T(rel), is started.

4) The inactivity control timers, T(ias) and T(iar), if still running, are stopped on both connection sections.

3.3.4.2 *Subsequent actions*

The following actions are performed at an intermediate node during connection release:

1) When a *Release Complete* | r *Released* | essage is received on a connection section, the resources associated with the connection are released, the timer T(rel) is stopped, and the local reference number is frozen.

2) When the release timer expires, a *Released* | essage is transferred on the connection section. The sending of the Released message is repeated every 4-15 seconds for an interval

of up to one minute. At this point a maintenance function is alerted.

3.3.5 *Actions at an end node not initiating connection release*

When a *Released* | essage is received at an end node of a signalling connection, the following actions are performed on the connection section:

1) A *Release Complete* | essage is sent on the connection section.

2) The resources associated with the connection section are released, the SCCP user is informed that a release has occured by invoking the N-DISCONNECT INDICATION primitive, and the local reference number is frozen.

3) The inactivity control timers, T(ias) and T(iar), if still running, are stopped.

3.4 *Inactivity control*

The purpose of the inactivity control is to recover from:

1) loss of a *Connection Confirm* | essage during connections establishment, and

2) the unsignalled termination of a connection section during data transfer, and

3) a discrepancy in the connection data held at each end of a connection.

Two inactivity control timers, the receive inactivity control timer T(iar) and the send inactivity control timer T(ias), are required at each end of a connection section. The length of the receive inactivity timer must be longer than the length of the send inactivity timer.

When any message is sent on a connection section, the send inactivity control timer is reset.

When any message is sent on a connection section, the receive inactivity control timer is reset.

When the send inactivity timer, T(ias), expires, an *Inactivity Test*
| (IT) message is sent on the connection section.

The receiving SCCP checks the information contained in the IT message against the information held locally. If a discrepancy is detected, the following actions (Table 1/Q.714) are taken:

When the receive inactivity control timer, T(iar), expires, the connection release procedure is initiated on a temporary connection section and an OA&M function is alerted for a permanent connection section.

As an alternative to inactivity control timers in the SCCP, there is also the possibility of supervising a signalling connection by a SCCP user function.

H.T. [T1.714]
TABLE 1/Q.714

Discrepancy	Action
Source reference number	Release connection
Protocol class	Release connection
{ Sequencing/segmenting | ua) }	Reset connection
Credit | ua)	Reset connection

a) Does not apply to class 2 connection.

Table 1/Q.711 [T1.714], p.

3.5 *Data transfer*

3.5.1 *General*

The purpose of data transfer is to provide the functions necessary to transfer user information on a temporary or permanent signalling connection.

3.5.1.1 *Actions at the originating node*

The SCCP user at the originating node requests transfer of user data on a signalling connection by invoking the N-DATA REQUEST primitive.

The *Data* | essage is then generated, which must be transferred on the connection section. If flow control procedures apply to the connection section, these procedures must be enacted before the message can be forwarded on the connection section.

3.5.1.2 *Actions at the intermediate node*

If a signalling connection consists of more than one connection section, then one or more intermediate nodes are involved in the transfer of *Data* messages on the signalling connection.

When a valid *Data* | essage is received on an incoming connection section at an intermediate node, the associated outgoing connection section is determined. The intermediate node then forwards the *Data* message to the

associated outgoing connection section for transfer to the distant node. If flow control procedures apply to the connection sections, then the appropriate procedures must be enacted on both connection sections. On the incoming connection section, these procedures relate to the reception of a valid *Data* message and on the outgoing connection section, the procedures control the flow of *Data* messages on the connection section.

3.5.1.3 *Actions at the destination node*

When the destination node receives a valid *Data* | essage, the SCCP user (i.e., the Called Party Address) is notified by invoking the N-DATA INDICATION primitive. If flow control procedures apply to the signalling connection, the flow control procedures relating to the reception of a valid *Data* message are enacted.

3.5.2 *Flow control*

3.5.2.1 *General*

The flow control procedures apply during data transfer only, and are used to control the flow of *Data* | essages on each connection section.

The flow control procedures apply only to protocol class 3.

The reset procedure causes reinitialization of the flow control procedure.

The expedited data procedure is not affected by this flow control procedure.

3.5.2.2 *Sequence numbering*

For protocol class 3, for each direction of transmission on a connection section, the *Data* | essages are sequentially numbered.

The sequence numbering scheme of the *Data* | essages is performed modulo 128 on a connection section.

Upon initialization or reinitialization of a connection section, message send sequence numbers, P(S), are assigned to *Data* | essages on a connection section beginning with P(S) equal to 0. Each subsequent *Data* message sequence number is obtained by incrementing the last assigned value by 1. The sequence numbering scheme assigns sequence numbers up to 127.

3.5.2.3 *Flow control window*

A separate window is defined, for each direction of transmission, on a connection section in order to control the number of *Data* messages

authorized for transfer on a connection section. The window is an ordered set of W consecutive message send sequence numbers associated with the *Data* messages authorized for transfer on the connection section.

The lower window edge is the lowest sequence number in the window.

The sequence number of the first *Data* | essage not authorized for transfer on the connection is the value of the lower window edge plus W.

The maximum window size is set during connection establishment for temporary connection sections. For permanent connection sections, the window size is fixed at establishment. The maximum size cannot exceed 127.

Negotiation procedures during connection establishment allow for the negotiation of the window size.

3.5.2.4 *Flow control procedures*

3.5.2.4.1 *Transfer of Data messages*

If flow control procedures apply to a connection section, then all *Data* | essages on the connection section contain a send sequence number, P(S), and a receive sequence number, P(R). The procedure for determining the send sequence number to be used in a *Data* message is described in § 3.5.2.2.

The receive sequence number, P(R), is set equal to the value of the next send sequence number expected on the connection section and P(R) becomes the lower window edge of the receiving window.

An originating or intermediate node is authorized to transmit a *Data* message if the message send sequence number of the message is within the sending window. That is, if P(S) is greater than or equal to the lower window edge and less than the lower window edge plus W. When the send sequence number of a *Data* message is outside of the sending window, the node is not authorized to transmit the message.

3.5.2.4.2 *Transfer of Data Acknowledgement messages*

Data Acknowledgement | essages may be sent when there are no *Data* | essages to be transferred on a connection section

Further study is required to determine criterion to be used to decide when *Data Acknowledgement* | essages are sent for cases other than the congestion situation described in this section.

When a node transfers a *Data Acknowledgement* | essage on a connection section, it is indicating that the node is ready to receive W *Data*

messages within the window starting with the receive sequence number, P(R), found in the *Data Acknowledgement* message. That is, P(R) is the next send sequence number expected at the remote node on the connection section. Furthermore, P(R) also becomes the lower window edge of the receiving window.

A *Data Acknowledgement* | essage must be sent when a valid *Data* | message, as per § 3.5.2.4.3 on P(S) and P(R), is received and P(S) is equal to the upper edge of the receiving window and there are no *Data* messages to be transferred on the connection section. Sending of *Data Acknowledgement* messages before having reached the upper edge of the receiving window is also allowed during normal operation.

Data acknowledgement | messages may also be sent by a node encountering congestion on a connection section as described below:

Assuming nodes X and Y are the two ends of a connection section, the following procedures apply.

When a node (Y) experiences congestion on a connection section, it informs the remote node (X) using the *Data Acknowledgement* message with the credit set to zero.

Node (X) stops transferring *Data* | essages on the connection section.

Node X updates the window on the connection section using the value of the receive sequence number, P(R), in the *Data Acknowledgement* message.

Node X begins transfer of *Data* | essage when it receives a *Data Acknowledgement* | essage with a credit field greater than zero or when a *Reset* message is received on a connection section for which a *Data Acknowledgement* message with a credit field equal to zero had previously been received.

Node X updates the window on the connection using the credit value. The credit value in a *Data Acknowledgement* message must either equal zero or equal the initial credit agreed to at connection establishment.

3.5.2.4.3 *Reception of a Data or Data Acknowledgement message*

When an intermediate or destination node receives a *Data* message, it performs the following test on the send sequence number, P(S), contained in the *Data* message:

1) If P(S) is the next send sequence number expected and is within the window, then the node accepts the *Data* message and increments by one the value of the next sequence number expected on the connection section.

2) If P(S) is not the next send sequence number expected, then the reset procedure is initiated on the connection section.

3) If P(S) is not within the window, then this is considered a local procedure error and the connection reset procedure is initiated.

4) If P(S) is not equal to 0 for the first *Data* message received after initialization or reinitialization of the connection section, this is considered a local procedure error and the connection reset procedure is initiated.

The message receive sequence number, P(R), is included in *Data* , and *Data Acknowledgement* messages. When a node receives a *Data* or *Data Acknowledgement* message on a connection section, the value of the receive sequence number, P(R), implies that the remote node has accepted at least all *Data* messages numbered up to and including P(R) — 1. That is, the next expected send sequence number at the remote node is P(R). The receive sequence number, P(R), contains information from the node sending the message, which authorizes the transfer of a limited number of *Data* messages on the

connection section. When a node receives a *Data* or *Data Acknowledgement* message:

a) the receive sequence number, P(R), contained in the message becomes the lower window edge of the sending window:

1) if the value of P(R) is greater than or equal to the last P(R) received by the node on that connection section, and also,

2) if the value of the received P(R) is less than or equal to the P(S) of the next *Data* message to be transferred on that connection section;

b) the node initiates the reset procedure on the connection section if the receive sequence number, P(R), does not meet conditions 1) and 2).

3.5.3 *Segmenting and reassembly*

During the data transfer phase, the N-DATA REQUEST primitive is used to request transfer of octet-aligned data (NSDUs) on a signalling connection. NSDUs longer than 255 octets must be segmented before insertion into the ‘‘data’’ field of a *Data* message.

The more-data indicator (M-bit) is used to reassemble a NSDU that has been segmented for conveyance in multiple *Data* messages. The M-bit is set to 1 in all *Data* messages except the last message whose data field relates to a particular NSDU. In this way, the SCCP can reassemble the NSDU by combining the data fields of all *Data* messages with the M-bit set to 1 with the following *Data* message with the M-bit set to 0. The NSDU is then delivered to the SCCP user using the N-DATA INDICATION. *Data* messages in which the M-bit is set to 1 do not necessarily have the maximum length.

Segmentation and reassembly are not required if the length of the NSDU is less than or equal to 255 octets.

3.6 *Expedited data transfer*

3.6.1 *General*

The expedited data procedure applies only during the data transfer phase and is applicable to protocol class 3.

For the case of expedited data transfer, each message contains one NSDU, and no segmenting and reassembly is provided.

If an *Expedited Data* or *Expedited Data Acknowledgement* message is lost, then subsequent *Expedited Data* messages cannot be forwarded on the connection section.

3.6.2 *Actions at the originating node*

The expedited data transfer procedure is initiated by the user of the SCCP using the N-EXPEDITED-DATA REQUEST primitive, which contains up to 32 octets of user data.

When the SCCP user invokes the N-EXPEDITED-DATA REQUEST primitive, an *Expedited Data* message with up to 32 octets of user data is transferred on the connection section once *all* previous *Expedited Data* messages for the connection section have been acknowledged.

3.6.3 *Actions at intermediate node*

Upon receiving a valid *Expedited Data* message, an intermediate node confirms this message by transferring an *Expedited Data Acknowledgement* message on the incoming connection section. Withholding of the *Expedited Data Acknowledgement* message is a means of providing flow control of *Expedited Data* messages.

If a node receives another *Expedited Data* message on the incoming connection section before sending the *Expedited Data Acknowledgement* message, then the node will discard the subsequent message and reset the incoming connection section.

The intermediate node determines the associated outgoing connection section. An *Expedited Data* message is then transferred on the associated outgoing connection section, once *all* previous *Expedited Data* messages on that connection section have been acknowledged.

The *Expedited Data Acknowledgement* message must be sent before acknowledging subsequent *Data* or *Expedited Data* messages received on the incoming connection section.

3.6.4 *Actions at destination node*

The destination node of the connection section confirms a valid *Expedited Data* message by transferring an *Expedited Data Acknowledgement* message on the connection section. Withholding of the *Expedited Data Acknowledgement* message is a means of providing flow control of *Expedited Data* messages.

If a node receives another *Expedited Data* | essage on a connection section before sending the *Expedited Data Acknowledgement* message, then the node will discard the subsequent message and reset the connection section.

The destination node then invokes the N-EXPEDITED DATA INDICATION primitive.

The N-EXPEDITED-DATA INDICATION must be issued to the SCCP user at destination node before N-DATA or N-EXPEDITED-DATA indications resulting from any subsequently issued N-DATA or N-EXPEDITED-DATA requests at the originating node of that signalling connection. The initiation of the *Expedited Data Acknowledgement* message is implementation dependent.

3.7 *Reset*

3.7.1 *General*

The purpose of the reset procedure is to reinitialize a connection section. It is applicable only to protocol class 3. It is noted that the time sequence of the primitives in the reset procedure may be varied as long as it is consistent with Recommendation X.213.

For a connection reset initiated by the SCCP, Data or Expedited Data messages should not be transferred on the connection section prior to the completion of the reset procedure.

3.7.2 *Action at the initiating node*

3.7.2.1 *Initial actions*

When a connection reset is initiated, by the SCCP user invoking a N-RESET REQUEST primitive or by the node itself, the following actions are performed at the initiating node:

1) A *Reset Request* | essage is transferred on the connection section.

2) The send sequence number, P(S), for the next *Data* message is set to 0. The lower window edge is set to 0. The window size is reset to the initial credit value.

3) The SCCP user is informed that a reset has taken place by:

— invoking the N-RESET INDICATION primitive if the reset is network originated.

4) The reset timer T (reset) is started.

3.7.2.2 *Subsequent actions*

The following actions are performed at the initiating node on a connection section for which a *Reset Request* message has been previously transferred:

1) When a *Data* , *Data Acknowledgement* , *Expedited Data* , or *Expedited Data Acknowledgement* message is received, the message is discarded. When an N-DATA REQUEST or N-EXPEDITED DATA REQUEST primitive is received, the primitive is discarded or stored up to the completion of the reset procedure. The choice between these two is implementation dependent.

2) When the reset timer expires, the connection release procedure is initiated on a temporary connection section and maintenance functions are alerted for a permanent connection section.

3) When a *Reset Confirm* | r a *Reset Request* | essage is received on the connection section, the reset is completed provided the SCCP has previously received an N-RESET REQUEST or RESPONSE primitive from the SCCP user and, therefore, data

transfer is resumed and the timer T(reset) is stopped. The SCCP user is informed that the reset is completed by invoking the N-RESET CONFIRMATION primitive.

4) When a *Released* message is received on a temporary connection section, the release procedure is initiated and the timer, T (reset), is stopped.

3.7.3 *Actions at the intermediate node*

3.7.3.1 *Initial actions*

The connection reset procedure is initiated at the intermediate node either by the SCCP at the node itself or by the reception of a *Reset Request* message.

When a *Reset Request* | essage is received on a connection section, the following actions take place:

1) A *Reset Confirm* | essage is transferred on the connection section.

2) A *Reset Request* | essage is transferred on the associated connection section; the reason for reset is identical to the reason in the *Reset Request* message.

3) On both the connection section and the associated connection section, the send sequence number, P(S), for the next *Data* message to be transmitted is set to 0 and the lower window edge is set to 0. The window size is reset to the initial credit value on both connection sections.

4) The data transfer procedure is initiated on the connection section.

5) The reset timer, T (reset), is started on the associated connection section.

When the connection reset procedure is initiated by the SCCP at the intermediate node, the following actions take place on both of the connection sections:

1) A *Reset Request* | essage is transferred.

2) The send sequence number, P(S), for the next *Data* | essage is set to 0. The lower window edge is set to 0. The window size is reset to the initial credit value.

3) The reset timer T (reset) is started.

3.7.3.2 *Subsequent actions*

If the connection reset was initiated by reception of a *Reset Request* | message on a connection section, then the following actions are performed after initial actions are completed:

1) When a *Data* , *Data Acknowledgement* , *Expedited Data* , *Expedited Data Acknowledgement* message is received on the associated connection section, the message is discarded.

2) When the reset timer expires on the associated connection section, the connection release procedure is initiated on both temporary connection sections and the maintenance function is alerted on an associated permanent connection section.

3) When a *Released* | essage is received on a temporary connection section, the connection release procedure is initiated on both connection sections and the timer, T (reset), is stopped.

4) When a *Reset Confirm* | r *Reset Request* | essage is received on the associated connection section, the data transfer procedure is resumed and the timer, T (reset), is stopped.

If the connection reset was initiated by the SCCP at the intermediate node, then the following actions are performed once the initial actions are completed:

1) When a *Data* , *Data Acknowledgement* , *Expedited Data* , or *Expedited Data Acknowledgement* message is received on either connection section, the message is discarded.

2) When the reset timer expires on a temporary connection section, the connection release procedure is initiated on both connection sections, and on a permanent connection section a maintenance function is alerted.

3) When a *Released* | essage is received on a temporary connection section, the connection release procedure is initiated on both connection sections and the timer, T (reset), is stopped .

4) When a *Reset Confirm* | r *Reset Request* | essage is received on a connection section, data transfer is resumed on that connection and the timer, T (reset), is stopped.

3.7.4 *Actions at the destination node*

When a *Reset Request* | essage is received at a node, the following actions are performed on the connection section:

1) The send sequence number, P(S), for the next *Data* | essage is set to 0, the lower window edge is set to 0. The window size is reset to the initial credit value.

2) The SCCP user is informed that a reset has occurred by invoking the N-RESET INDICATION primitive.

3) A *Reset Confirm* | essage is transferred on the connection section after an N-RESET RESPONSE or REQUEST primitive is invoked by the user.

4) An N-RESET CONFIRMATION primitive is invoked to inform the SCCP user that the reset is completed and the data transfer can be resumed.

3.7.5 *Handling of messages during the reset procedures*

Once the reset procedure is initiated, the following actions are taken with respect to *Data* messages:

— those that have been transmitted, but for which an acknowledgement has not been received, are discarded, and

— those that have not been transmitted, but are contained in an M-bit sequence for which some *Data* messages have been transmitted, are discarded,

— those *Data* | essages that have been received, but which do not constitute an entire M-bit sequence, are discarded.

3.8 *Restart*

3.8.1 *General*

The purpose of the restart procedure is to provide a recovery mechanism for signalling connection sections in the event of a node failure.

3.8.2 *Actions at the recovered node*

3.8.2.1 *Initial actions*

When a node recovers from its failure, the following actions are performed:

1) A guard timer, T(guard) , is started.

2) If the recovered node has knowledge about the local reference numbers in use before failure, then the normal procedures for temporary signalling connections are resumed with the assumption that the local reference numbers which were in use before the node failure are not assigned at least during T(guard).

3) An OA&M function is informed for the re-establishment of permanent signalling connections.

The guard timer must be large enough, so that all the non-failed far end nodes can detect the failure and

can safely release the affected temporary signalling connection sections. This implies T(guard) | | (iar) | | (rel).

3.8.2.2 *Subsequent actions*

The following actions are performed at the recovered node, on every temporary signalling connection section if the node does not know the local reference numbers in use before failure, or only on the temporary signalling connection sections in operation before failure if the node has such knowledge:

a) Before the guard timer, T(guard), expires:

1) When a *Released* | message is received with both source and destination local reference numbers, a *Release Complete* message, with reversed local reference numbers, is returned to the originating point code.

2) Any other connection-oriented messages received are discarded.

b) When the guard timer, T(guard), expires, normal procedures are resumed.

3.8.3 *Actions at the non-failed far end node*

The inactivity control procedure, described in § 3.4, is used by the non-failed far end node to recover from the unsignalled termination of a connection section during data transfer.

3.9 *Permanent signalling connections*

Permanent signalling connections are set up administratively and connection establishment procedures and connection release procedures are not initiated by the SCCP user.

Permanent signalling connections are realized using one or more connection sections.

A permanent signalling connection is either in the data transfer phase or the reset phase. Therefore, all procedures relating to the data transfer phase for connection-oriented protocol classes and the reset procedures are applicable to permanent signalling connections.

3.10 *Abnormalities*

3.10.1 *General*

Errors can be classified into the three categories listed below. Examples of each category are included for clarification:

1) *Syntax errors* — This type of error occurs when a node receives a message that does not conform to the format specifications of the SCCP. Examples of syntax errors are:

— reception of message with an invalid message type code, and

— reception of a message with an unassigned local reference number.

2) *Logical errors* — This type of error occurs when a node receives a message that is not an acceptable input to the current state of the connection section, or whose value of P(S) or P(R) is invalid. Examples of logical errors are:

— reception of an acknowledgement message when the corresponding request message has not been sent,

— reception of a *Data* |essage whose data field length exceeds the maximum data field permitted on the connection section,

— reception of a second *Expedited Data* |essage before an *Expedited Data Acknowledgement* message has been sent, and

— reception of message whose value of P(R) is not greater than or equal to the last P(R) received and is not less than or equal to the next value of P(S) to be transmitted.

3) *Transmission errors* — This type of error occurs when a message is lost or delayed. Examples of transmission errors are:

— expiration of a timer before reception of the appropriate acknowledgement message.

3.10.2 *Action tables*

The action tables found in Recommendation Q.714, Annex B, include information, in addition to that found in the text of Recommendation Q.714, regarding the actions to be performed upon receipt of a message. In particular, these tables are helpful in determining the actions to be performed upon receipt of a message resulting in a logical error.

3.10.3 *Actions upon the reception of an ERR message*

Upon the reception of a *Protocol Data Unit Error* | (ERR) message at a node, the following actions are performed on the connection section for error causes other than ‘‘service class mismatch’’:

1) The resources associated with the connection are released.

2) The local reference number is frozen (see § 3.3.2).

Upon the reception of a *Protocol Data Unit Error* | ERR) message at a node with the error cause ‘‘service class mismatch’’, the connection release procedure is initiated by the SCCP at that node (see § 3.3).

4 Connectionless procedures

The connectionless procedures allow a user of the SCCP to request transfer of up to X octets of user data without first requesting establishment of a signalling connection.

The N-UNITDATA REQUEST and INDICATION primitives are used by the user of the SCCP to request transfer of user data by the SCCP and for the SCCP to indicate delivery of user data to the destination user. Parameters associated with the N-UNITDATA REQUEST primitive must contain all information necessary for the SCCP to deliver the user data to the destination.

Transfer of the user data is accomplished by including the user data in *Unitdata* messages.

The user of the SCCP should ensure that the total length of the user data and the SCCP address information does not exceed the total permissible length of the SCCP Unitdata message.

If user data of excessive length is presented by the user of the SCCP, the SCCP should not transmit a part of it to the remote user of the SCCP.

Whether or not the local SCCP user should be informed by the SCCP is implementation dependent.

When the user of the SCCP requests transfer of user data by issuing a N-UNITDATA REQUEST primitive, there are two classes of service that can be provided by the SCCP, protocol classes 0 and 1. These protocol classes are distinguished by their message sequencing characteristics.

When the user of the SCCP requests transfer of several messages by issuing multiple N-UNITDATA REQUEST primitives, the probability of these messages being received in sequence at the ‘‘Called address’’ depends on the protocol class designated in the request primitives. For protocol class 0 the sequence control parameter is not included in the N-UNITDATA REQUEST primitive and the SCCP may generate a different SLS for each of these messages. For protocol class 1 the sequence control parameter is included in the N-UNITDATA REQUEST primitive and, if the parameter is the same in each request primitive, then the SCCP will generate the same SLS for these messages.

The Message Transfer Part retains message sequencing for those messages with the same SLS field. The Signalling Connection Control Part relies on the services of the MTP for transfer of SCCP messages. Based on the characteristics of the MTP, the protocol class 1 service may be used in such a way that it provides a quality of service that has a lower probability of out-of-sequence messages than that provided by protocol class 0.

4.1 *Data transfer*

The N-UNITDATA REQUEST primitive is invoked by the SCCP user at an originating node to request connectionless data transfer service. The connectionless data transfer service is also used to transport SCCP management messages, which are transferred in the ‘‘data’’ field of *Unitdata* messages.

The *Unitdata* | message is then transferred, using SCCP and MTP routing functions, to the ‘‘Called address’’ indicated in the UNITDATA REQUEST primitive.

SCCP routing and relaying functions may be required at intermediate nodes, since complete translation and routing tables for all addresses are not required at every node.

When the *Unitdata* | essage cannot be transferred to its destination, the message return function is initiated.

The SCCP uses the services of the MTP and the MTP may, under severe network conditions, discard messages. Therefore, the user of the SCCP may not always be informed of non-delivery of user data. The MTP notifies the SCCP of unavailable signalling points using the MTP-STOP INDICATION and of congested signalling points using the MTP-PAUSE INDICATION. The SCCP then informs its users.

When a *Unitdata* | essage is received at the destination node, a N-UNITDATA INDICATION primitive is invoked except for the SCCP management messages. The SCCP management (SCMG) messages are passed to the SCMG entity instead.

4.2 *Message return*

The purpose of message return is to discard or return messages which encounter routing failure and cannot be delivered to their final destination.

The message return procedure is initiated if SCCP routing is unable to transfer a *Unitdata* or *Unitdata Service* message. The procedure may be

initiated, for example, as a result of insufficient translation information or the inaccessibility of a subsystem or point code. Specific reasons are enumerated in § 2.3.

a) If the message is a *Unitdata* | essage, and

— the option field is set to return message on error, then a *Unitdata Service* | essage is transferred to the ‘‘calling party address’’. (If the message is originated locally, then a N-NOTICE INDICATION primitive is invoked.)

— the option field is not set to return on error, then the message is discarded.

b) If the undeliverable message is a *Unitdata Service* | essage, it is discarded.

The user ‘‘data’’ field of the *Unitdata* | essage and the reason for return are included in the *Unitdata Service* message.

When a *Unitdata Service* message is received at the destination node, a N-NOTICE INDICATION primitive is invoked.

4.3 *Syntax error*

This type of error occurs when a node receives a message that does not conform to the format specifications of the SCCP. Examples of syntax errors are:

— unreasonable pointer value (e.g., points beyond the end of the message);

— mismatch between message type and protocol class parameters; and

— inconsistent address indicator and address contents.

When syntax error is detected for a connectionless message, the message is discarded. Checking for syntax errors beyond the processing required for the SCCP connectionless message routing is not mandatory.

5 SCCP management procedures

5.1 *General*

The purpose of SCCP management is to provide procedures to maintain network performance by rerouting or throttling traffic in the event of failure or congestion in the network.

Although SCCP management has its own subsystem number, the procedures in this section does not apply to it.

SCCP management is organized into two subfunctions: signalling point status management and subsystem status management. Signalling point status management and subsystem status management allow SCCP management to use information concerning the accessibility of signalling points and subsystems, respectively, to permit the network to adjust to failure, recovery and congestion.

SCCP management procedures rely on:

1) failure, recovery, and congestion information provided in the MTP-PAUSE INDICATION, MTP-RESUME INDICATION and MTP-STATUS INDICATION primitives; and

Subsystem congestion control is for further study.

2) subsystem failure, recovery and congestion information received in SCCP management messages

SCCP management information is currently defined to be transferred using SCCP connectionless service with no return on error requested. Formats of these messages appear in Recommendation Q.713.

The information pertaining to both single and replicated nodes or subsystems is used for SCCP management purposes. This allows ‘‘called party addresses’’ that are specified in the form of a global title to be translated to different point codes and/or subsystem numbers depending on network or subsystem status.

Replicated nodes or subsystems may relate to their replicates in one of several ways. (‘‘Replicate’’ is a term meaning one of a set of ‘‘multiple copies’’. A node of subsystem which is not replicated is termed ‘‘solitary’’.)

One mode uses a replicate in a dominant role. Traffic is split among several nodes/subsystems. Under normal conditions, each portion of the traffic is routed to a preferred, or ‘‘primary’’, node/subsystem. When the primary node/subsystem is inaccessible, this traffic is routed to a ‘‘backup’’ node/subsystem . When the primary node/subsystem recovers, it reassumes its normal traffic load.

A second mode uses a replicate in a replacement role. Consider two replicates, A and B, which are ‘‘alternatives’’. When A becomes inaccessible, its traffic is routed to B; but when A recovers, the traffic is not moved back to A. It is only when B becomes inaccessible that traffic is shifted back to A. In addition, other modes are possible.

The current SCCP management procedures are designed to manage solitary nodes/subsystems, and replicated nodes/subsystems which operate in a dominant mode and for which any given primary node/subsystem has only one backup (i.e., duplicated nodes/subsystems). Management procedures for nodes/subsystems which operate in a mode other than the dominant mode and which have more than one backup are for further study.

SCCP management procedures utilize the concept of a ‘‘concerned’’ subsystem or signalling point. In this context, a ‘‘concerned’’ entity means an entity with an immediate need to be informed of a particular signalling point/subsystem status change, independently of whether SCCP communication is in progress between the ‘‘concerned’’ entity and the affected entity with the status change

In some situations, the number of concerned subsystem or signalling points for a given subsystem may be zero. In this case, when the subsystem fails, or becomes unavailable, no broadcast of the subsystem prohibited message is performed. Instead, the response method is used to return the subsystem prohibited message. Similarly, no broadcast of the subsystem allowed message is performed for that given subsystem when it recovers. The response method is again used to return a subsystem allowed message in reply to a subsystem status test.

The signalling point prohibited, signalling point allowed and signalling point congested procedures, specified in §§ 5.2.2, 5.2.3 and 5.2.4 respectively, deal with the accessibility of a signalling point.

The subsystem prohibited and subsystem allowed procedures, detailed in §§ 5.3.2 and 5.3.3 respectively, deal with the accessibility of a subsystem.

An audit procedure to ensure that necessary subsystem management information is always available is specified in the subsystem status test procedure in § 5.3.4.

A subsystem may request to go out of service using the coordinated state change control procedure specified in § 5.3.5.

Local subsystems are informed of any related subsystem status by the local broadcast procedure specified in § 5.3.6.

Concerned signalling points are informed of any related subsystem status by the broadcast procedure specified in § 5.3.7.

5.2 *Signalling point status management*

5.2.1 *General*

Signalling point status management updates translation and status based on the information of network failure, recovery, or congestion provided by the MTP-PAUSE INDICATION, MTP-RESUME INDICATION, or MTP-STATUS INDICATION primitives. This allows alternative routing to backup signalling points and/or backup subsystems.

Further explicit definition of ‘‘concerned’’ subsystems or signalling points would be network/architecture/application dependent.

5.2.2 *Signalling point prohibited*

When SCCP management receives a MTP-PAUSE INDICATION relating to a destination that become inaccessible, SCCP management:

1) marks the translation as appropriate:

— ‘‘translate to backup node’’ if that signalling point has a backup;

— ‘‘translate to backup subsystem’’ for each subsystem at that signalling point for which a backup subsystem exists.

2) marks as ‘‘prohibited’’ the status of that signalling point and of each subsystem at that signalling point.

3) discontinues any subsystem status tests (§ 5.3.4) it may be conducting to any subsystems at that signalling point.

4) initiates a local broadcast (§ 5.3.6) of ‘‘User-out-of-service’’ information for each subsystem at that signalling point.

5) initiates a local broadcast (§ 5.3.6) of ‘‘signalling point inaccessible’’ information for that signalling point.

5.2.3 *Signalling point allowed*

When SCCP management receives a MTP-RESUME INDICATION relating to a destination that becomes accessible, SCCP management:

1) resets the congestion level of that signalling point.

2) marks the translation as appropriate:

— ‘‘translate to primary node’’ if that signalling point has a backup.

3) marks as ‘‘allowed’’ the status of that signalling point.

4) initiates the subsystem status test procedure (§ 5.3.4) with affected subsystems at that signalling point.

5) initiates a local broadcast (§ 5.3.6) of ‘‘signalling point inaccessible’’ information for that signalling point.

5.2.4 *Signalling point congested*

When SCCP management receives a MTP-status indication relating to signalling network congestion to a signalling point, SCCP management:

1) updates that signalling point status to reflect the congestion.

2) initiates a local broadcast (§ 5.3.6) of ‘‘signalling point congested’’ information for that signalling point.

5.3 *Subsystem status management*

5.3.1 *General*

Subsystem status management updates translation and status based on the information of failure, withdrawal, congestion , and recovery of

subsystems. This allows alternative routing to backup systems, if appropriate. Local users are informed of the status of their backup subsystems.

5.3.2 *Subsystem prohibited*

5.3.2.1 *Receipt of messages for a prohibited subsystem*

If SCCP routing control receives a message, whether originated locally or not, for a prohibited local system, SCCP routing control invokes subsystem prohibited control. A *Subsystem-Prohibited* message is sent to the originating signalling point if the originating subsystem is not local (the OPC is a parameter in the MTP-TRANSFER INDICATION primitive). The action, if any, to be taken, if the originating subsystem is local, is for further study.

5.3.2.2 *Receipt of Subsystem-Prohibited message or N-STATE REQUEST primitive or local user failed*

Under one of the following conditions:

a) SCCP management receives a *Subsystem-Prohibited* | message about a subsystem marked allowed, or

b) a N-STATE REQUEST primitive with ''User-out-of-service'' information is invoked by a subsystem marked allowed, or

c) SCCP management detects that a local subsystem has failed,

then SCCP management does the following:

1) marks the translation as appropriate:

— ''translate to backup subsystem'' if a backup subsystem exists for the prohibited subsystem.

2) marks as ''prohibited'' the status of that subsystem.

3) initiates a local broadcast (§ 5.3.6) of ''User-out-of-service'' information for the prohibited subsystem.

4) initiates the subsystem status test procedure (§ 5.3.4) if the prohibited subsystem is not local.

5) forwards the information throughout the network by initiating a broadcast (§ 5.3.7) of *Subsystem-Prohibited* messages to concerned signalling points.

6) cancels ''ignore subsystem status test'' and the associated timer if they are in progress and if the newly prohibited subsystem resides at the local node.

5.3.3 *Subsystem allowed*

Under one of the following conditions:

a) SCCP management receives a *Subsystem-Allowed* | message about a subsystem marked prohibited, or

b) a N-STATE REQUEST primitive with ''User-in-Service'' information is invoked by a subsystem marked prohibited,

then SCCP management does the following:

1) marks the translation as appropriate:

— ''translate to primary subsystem'' if that subsystem is duplicated and the primary subsystem is allowed;

— ''translate to backup subsystem'' if that subsystem is duplicated and the primary subsystem is prohibited.

2) marks as ''allowed'' the status of that subsystem.

3) initiates as a local broadcast (§ 5.3.6) of ''User-in-service'' information for the allowed subsystem.

4) discontinues the subsystem status test relating to that subsystem if such a test was in progress.

5) forwards the information throughout the network by initiating a broadcast (§ 5.3.7) of *Subsystem-Allowed* | messages to concerned signalling points.

5.3.4 *Subsystem status test*

5.3.4.1 *General*

The subsystem status test procedure is an audit procedure to verify the status of a subsystem marked prohibited.

5.3.4.2 *Actions at the initiating node*

A subsystem status test is initiated when:

a) a *Subsystem-Prohibited* | message is received (§ 5.3.2.2), or

b) a MTP-RESUME INDICATION primitive for a previously inaccessible signalling point is invoked (§ 5.2.3).

A subsystem status test associated with a failed subsystem is commenced by starting a timer (stat.info) and marking a test in progress. No further actions are taken until the timer expires.

Upon expiration of the timer, a *Subsystem-Status-Test* | message is sent to SCCP management at the node of the failed subsystem and the timer is reset.

The cycle continues until the test is terminated by another SCCP management function at that node. Termination of the test causes the timer and the test mark to be cancelled.

5.3.4.3 *Actions at the receiving node*

When SCCP management receives a *Subsystem-Status-Test* | message and there is no ‘‘ignore subsystem status test’’ in progress, it checks the status of the named subsystem. If the subsystem is allowed, a *Subsystem-Allowed* message is sent to the SCCP management at the node conducting the test. If the subsystem is prohibited, no reply is sent.

5.3.5 *Coordinated state change*

5.3.5.1 *General*

A duplicated subsystem may be withdrawn from service without degrading the performance of the network by using the coordinated state change procedure described below when its backup is not local. The procedure, if any, to be specified in case the primary and the backup subsystems are co-located, is for further study.

5.3.5.2 *Actions at the requesting node*

When a duplicated subsystem wishes to go out of service, it invokes a N-COORD REQUEST primitive. SCCP management at that node sends a *Subsystem-Out-of-Service-Request* message to the backup system, sets a timer (coord.chg) and marks the subsystem as ‘‘waiting for grant’’.

Arrival of a *Subsystem-Out-of-Service-Grant* | message at the requesting SCCP management causes the timer (coord.chg) to be cancelled, the ‘‘waiting for grant’’ state to be cancelled, and a N-COORD CONFIRMATION primitive to be invoked to the requesting subsystem. *Subsystem-Prohibited* messages are broadcast (§ 5.3.7) to concerned signalling points.

In addition, an ‘‘ignore subsystem status test’’ timer is started and the requesting subsystem is marked as ‘‘ignore subsystem status test’’. Subsystem

status tests are ignored until the ‘‘ignore subsystem status test’’ timer expires or the marked subsystem invokes a N-STATE REQUEST primitive with ‘‘User-out-of-service’’ information.

If no ‘‘waiting for grant’’ is associated with the subsystem named in the *Subsystem-Out-of-Service-Grant* | message, the *Subsystem-Out-of-Service-Grant* | message is discarded and no further action is taken.

If the timer associated with the subsystem waiting for the grant expires before a *Subsystem-Out-of-Service-Grant* message is received, the ‘‘waiting for grant’’ is cancelled and the request is implicitly denied.

5.3.5.3 *Actions at the requested node*

The possibility of introducing an explicit Subsystem-Out-of-Service-Denial message containing additional information and associated primitive is for further study.

When the SCCP management at the node at which the backup subsystem is located receives the *Subsystem-Out-of-Service-Request* message, it checks the status of local resources the SCCP has sufficient resources to assume the

---

Local resources are whatever is critical to this particular node, and are implementation dependent.

increased load, it invokes a N-COORD INDICATION primitive to the backup subsystem. If the SCCP does not have sufficient resources, no further action is taken

If the backup system has sufficient resources to allow its mate to go out of service, it informs SCCP management by invoking a N-COORD RESPONSE primitive. A *Subsystem-Out-of-Service Grant* message is sent to SCCP management at the requesting node. If the backup subsystem does not have sufficient resources, no reply is returned

5.3.6 *Local broadcast*

5.3.6.1 *General*

The local broadcast procedure provides a mechanism to inform local allowed concerned subsystems of any related subsystem/signalling point status information received.

5.3.6.2 *User-out-of-service*

These cases are applicable when the SCCP is used for routing between local subsystems. This function is implementation dependent.
A local broadcast of ‘‘User-out-of-service’’ information is initiated when:

a) a *Subsystem-Prohibited* | message is received about a subsystem marked allowed (§ 5.3.2.2), or

b) an N-STATE REQUEST primitive with ‘‘User-out-of-service’’ information is invoked by a subsystem marked allowed (§ 5.3.2.2) , or

c) a local subsystem failure is detected by SCCP management (§ 5.3.2.2) ,

d) an MTP-PAUSE indication primitive is received (§ 5.2.2).

SCCP management then informs local allowed concerned SCCP subsystems about the subsystem status by invoking N-STATE indication primitive with ‘‘User-out-of-service’’ information.

5.3.6.3 *User-in-service*

A local broadcast of ‘‘subsystem-in-service’’ information is initiated when:

a) a *Subsystem-Allowed* | message is received about a subsystem marked prohibited (§ 5.3.3), or

b) an N-STATE REQUEST primitive with ‘‘User-in-service’’ information is invoked by a subsystem marked prohibited (§ 5.3.3).

SCCP management then informs local allowed concerned SCCP subsystems, except the newly allowed one, about the subsystem status by invoking an N-STATE indication primitive with ‘‘User-in-service’’ information.

5.3.6.4 *Signalling point inaccessible*

A local broadcast of ‘‘signalling point inaccessible’’ information is initiated when an MTP-RESUME primitive is received. SCCP management then

informs local allowed concerned SCCP subsystems about the signalling point status by invoking an N-PCSTATE INDICATION primitive with ‘‘signalling point accessible’’ information.

5.3.6.5 *Signalling point accessible*

A local broadcast of ‘‘signalling point accessible’’ information is initiated when an MTP-RESUME primitive is received. SCCP management then informs local allowed concerned SCCP subsystems about the signalling point status by invoking an N-PCSTATE INDICATION primitive with ‘‘signalling point accessible’’ information.

5.3.6.6 *Signalling point congested*

A local broadcast of ‘‘signalling point congested’’ information is initiated when an MTP-STATUS primitive is received. SCCP management then informs local allowed concerned SCCP subsystems about the signalling point status by invoking an N-PCSTATE indication primitive with ‘‘signalling point congested (level)’’ information.

5.3.7 *Broadcast*

5.3.7.1 *General*

The broadcast procedure provides a mechanism that may be used to inform concerned signalling points of any related subsystem status change at local or adjacent signalling points. It is an optional procedure supplementary to that defined in § 5.3.2.1. This procedure is suggested not to be used on a signalling point restart. This matter is for further study.

In some circumstances, the number of concerned signalling points is zero and no broadcast is performed. The action taken in this case is described in § 5.1.

5.3.7.2 *Subsystem prohibited*

A broadcast of *Subsystem-Prohibited* | messages is initiated when:

a) a *Subsystem Prohibited* | message is received about a subsystem presently marked allowed (§ 5.3.2.2), and the affected point code identified in the SSP message is the same as that of the informer signalling point, or

b) an N-STATE REQUEST primitive with ''User-out-of-service'' information is invoked by a subsystem marked allowed (§ 5.3.2.2), or

c) a local subsystem failure is detected by SCCP management § 5.3.2.2), or

d) a *Subsystem-Out-of-Service-Grant* | message arrives for a subsystem marked ''waiting for grant'' (§ 5.3.5.2).

This broadcast permits SCCP management to inform all concerned signalling points, except the informer signalling point, about the subsystem status by *Subsystem-Prohibited* messages. SCCP management does not broadcast if the point code of the prohibited subsystem is different from that of the informer signalling point which originates the *Subsystem-Prohibited* message.

5.3.7.3 *Subsystem allowed*

A broadcast of *Subsystem-Allowed* | messages is initiated when:

a) a *Subsystem-Allowed* | message is received about a subsystem presently marked prohibited (§ 5.3.3), and the affected point code identified in the SSA message is the same as that of the informer signalling point, or

b) an N-STATE REQUEST primitive with ''User-in-service'' information is invoked by a subsystem marked prohibited (§ 5.3.3).

This broadcast permits SCCP management to inform all concerned signalling points, except the informer signalling point, about the subsystem status by *Subsystem-Allowed* messages. SCCP management does not broadcast if the point code of the allowed subsystem is different from that of the informer signalling point which originates the *Subsystem-Allowed* message.

5.4 *SCMG restart*

Note — This section is for further study.)

On a signalling point restart, an indication is given to the SCCP by the MTP about the signalling points which are accessible after the restart actions. For each accessible, concerned signalling point, all subsystems there are marked allowed. The response method is used to determine the status of SCCP subsystems in those signalling points in the absence of the receipt of subsystem allowed, and subsystem prohibited messages, which may have been broadcast from them.

At the restarted signalling point, the status of its own subsystems are not broadcast to concerned signalling points. In this case, the response method is used to inform other nodes attempting to access prohibited subsystems at the restarted signalling points.

The SCCP management procedures specified in Recommendation Q.714, § 5.2, describe the normal operation of management procedures, and do not describe signalling point restart actions.

ANNEX A

(to Recommendation Q.714)

State diagrams for the signalling connection

control part of Signalling System No. 7

A.1 *Introduction*

This Annex contains the definitions for the symbols used and defines the states of the signalling point X/Y interface and the transitions between states in the normal case.

Annex B contains the full definition of actions, if any, to be taken on the receipt of messages by a signalling point.

A.2 *Symbol definition of the state diagrams at the message interface between two nodes (signalling points: X and Y)* (see Figures A-1/Q.714 and A-2/Q.714)

Figure A-1/Q.714, p.

Figure A-2/Q.714, p.

A.3 *Order definition of the state diagrams*

For the sake of clarity, the normal procedure at the interface is described in a number of small state diagrams. In order to describe the normal procedure fully, it is necessary to allocate a priority to the different figures and to relate a higher order diagram with a lower one. This has been done by the following means:

— Figures A-3/Q.714, A-4/Q.714, A-5/Q.714 and A-6/Q.714 are arranged in order of priority, with Figure A-3/Q.714 having the highest priority and subsequent figures having lower priority. Priority means that when a message belonging to a higher order diagram is transferred, that diagram is applicable and the lower order one is not.

— The relation with a state in a lower order diagram is given by including that state inside an ellipse in the higher order diagram.

— The message abbreviations are those defined in Recommendation Q.712.

Figure A-3/Q.714, p.5

Figure A-4/Q.714, p.6

Figure A-5/Q.714, p.7

Figure A-6/Q.714, p.8

MONTAGE: ANNEXE B SUR LE RESTE DE CETTE PAGE