

BSI**DD 99 : Part 1 : 1985**

UDC 681.324.02

© British Standards Institution. No part of this publication may be photocopied or otherwise reproduced without the prior permission in writing of BSI

Draft for Development

Logical link control for local area networks

Part 1. Technical specification (ISO/DP 8802/2;
IEEE 802.2 Draft E March 1984)

To avoid delay in preparing the content of this publication for printing, it was reproduced from the only available copy of the finally agreed document. The resultant poor print quality is very much regretted. The price has been adjusted accordingly.

**Documents**

Document Sharing Hub

British Standards Institution

Contents

	Page		Page
National foreword	3	7.4 Procedures for Data Link Set-Up and Disconnection	63
Committees responsible	Back cover	7.5 Procedures for Information Transfer	64
Draft for Development		7.6 Procedures for Resetting	67
Abbreviations	5	7.7 FRMR Exception Conditions	68
Definitions	7	7.8 List of Data Link Connection Parameters	69
1. Introduction	10	7.9 Formal Description of the Type 2 Procedures	70
1.1 Scope and Purpose	10	National appendix	
1.2 Standards Compatibility	11	A. British Standards and Drafts for Development for OSI and data communications	100
2. LLC Sublayer Interface Service Specifications	12	Tables	
2.1 Network Layer/LLC Sublayer Interface Service Specification	16	6.9.2-1 Station Component State Transition Table	55
2.2 LLC Sublayer/MAC Sublayer Interface Service Specification	28	6.9.2-2 Station Component Options Table	56
2.3 LLC Sublayer/LLC Sublayer Management Function Interface Service Specification	30	6.9.3 Service Access Point Component State Transition Table	60
3. LLC Protocol Data Unit (PDU) Structure	30	Figures	
3.1 General	30	1.1 Relationship to LAN Reference Model	10
3.2 LLC PDU Format	30	2.a Service Primitives	13
3.3 Elements of the LLC PDU	31	2.b Time Sequence Diagram	15
4. LLC Types and Classes of Procedure	33	3.2 LLC PDU Format	31
4.1 General	33	3.3.a DSAP and SSAP Address Field Formats	32
4.2 Classes of LLCs	34	3.3.b Global DSAP Address Field Format	32
5. LLC Elements of Procedure	36	4.1 Balanced Data Link Connection Configuration	34
5.1 General	36	4.2 Classes of LLC	35
5.2 Control Field Formats	36	5.2 LLC PDU Control Field Formats	37
5.3 Control Field Parameters	38	5.4.1.1 Type 1 Operation Command Control Field Bit Assignments	40
5.4 Commands and Responses	39	5.4.1.1.2 XID Information Field Basic Format	41
6. LLC Description of the Type 1 Procedures	48	5.4.1.2 Type 1 Operation Response Control Field Bit Assignments	42
6.1 Modes of Operation	48	5.4.2.1 Information Transfer Format Control Field Bits	43
6.2 Procedure for Addressing	48	5.4.2.2 Supervisory Format Control Field Bits	44
6.3 Procedure for the Use of the P/F Bit	48	5.4.2.3.a Unnumbered Format Control Field Bits	45
6.4 Procedures for Logical Data Link Set-Up and Disconnection	49	5.4.2.3.b Unnumbered Command and Response Control Field Bit Assignment	46
6.5 Procedures for Information Transfer	49	5.4.2.3.5 FRMR Information Field Format	47
6.6 Uses of the XID Command PDU and Response PDU	49	6.9.1 Component Relationships	52
6.7 Uses of the TEST Command PDU and Response PDU	49	6.9.2 Station Component State Diagram	54
6.8 List of Logical Data Link Parameters	50	6.9.3 Link Service Access Point State Diagram	59
6.9 Formal Description of the Type 1 Procedures	50	7.9.1 Connection Component State Diagram	72
7. LLC Description of the Type 2 Procedures	62		
7.1 Modes	62		
7.2 Procedures for Addressing	63		
7.3 Procedures for the Use of the P/F Bit	63		

This publication is not to be regarded as a British Standard.

It is being issued in the Draft for Development series of publications and is of a provisional nature because the application of standardization to LAN technology is rapidly advancing and not yet sufficiently stable, pending the outcome of deliberations in the International Standards Committee (ISO/TC 97, SC 6 Telecommunications and Information Exchange between Systems).

It should be applied on this provisional basis, so that information experience of its practical application may be obtained.

A review of this Draft for Development will be carried out not later than two years after its publication.

Notification of the start of the review period, with a request for the submission of comments from users of this Draft for Development will be made in an announcement in the appropriate issue of *BSI News*. Observations which it is felt should receive attention before the official call for comments will be welcomed.

According to the replies received, the responsible BSI Committee will judge whether the Draft for Development can be converted into a British Standard or what other action should be taken.

National foreword

This Draft for Development has been prepared under the direction of the Office and Information Standards Committee, at the request of the Department of Trade and Industry.

The primary objective of this Draft for Development is to give advance guidance to product and systems designers concerned with local area networks.

The text of the Draft for Development is identical with Draft E (Revised March 1984) of the standard finalized and approved by the Institute of Electrical and Electronics Engineers, Inc., IEEE 802.2 'Logical link control'. Earlier drafts of the IEEE standard were submitted to the International Organization for Standardization (ISO) TC 97/SC 6, Telecommunications and Information Exchange between Systems, for study, and Draft E (September 1983) was circulated as Draft Proposal ISO/DP 8802/2 with a closing date for voting of 29 February 1984. The text of this Draft for Development has also been submitted for circulation as Draft International Standard ISO/DIS 8802/2.

For ease of reproduction the IEEE text has been reproduced unchanged.

The UK has actively participated through BSI Technical Committee OIS/6, in the development of this Draft for Development. The Technical Committee will review any agreed amendments or addenda to the ISO text and will, where appropriate, update the text of this Draft for Development via an amendment or revision. When the ISO Draft reaches an appropriate level of agreement and stability, this Draft for Development may then be converted to a British Standard.

Other British Standard publications on local area networks include:

- DD 98 CSMA/CD local area networks
Part 1 Technical specification
Part 2 Guidance for implementors
- DD 100 Token bus local area networks
Part 1 Technical specification
- BS 6531 10 Mbps slotted ring local area network
Part 1 Specification for the coding of bits and structure of slots and mini-packets
Part 2 Specification for configuration
Part 3 Specification for free-standing repeaters
Part 4 Specification for basic and enhanced class nodes with Type 1 node/DTE interface
Part 5 Specification for monitor
Part 6 Specification for logging station
Part 7 Specification for slave power supplies
- BS 6532 Data terminal equipment for attachment to 10 Mbps slotted ring local area network
Part 1 Specification for media access control procedures for data terminal equipment
Part 2 Specification for implementation requirements for media access control in general purpose data terminal equipment

BS 6531 and BS 6532 have been submitted to ISO and will be issued in due course as a single Draft Proposal, ISO/DP 8802/6. IEEE 802.6 'Metropolitan area networks' has not been submitted to ISO.

This Draft for Development has been designated Part 1 in anticipation that an eventual Part 2 will be published as a technical guide to Part 1.

In accord with Article 4 (iii) of the Memorandum of Understanding between HM Government and BSI (1982), the Department of Trade and Industry recommend the use of this Part of DD 100 in the UK as part of the FOCUS Intercept Strategy.

For information, a list of published British Standards and Drafts for Development (together with those scheduled for publication at the time of publication of this Draft for Development for OSI data communications is attached as national appendix A. Press Releases for new OSI and data communications British Standard publications will be issued to the computing press, and details of all new publications will be recorded in full in *BSI News*. A Standing Order Scheme (SOS) is available to subscribing members of BSI, by which interested parties will receive all OSI and data communications British Standard publications, i.e. including British Standards, Drafts for Development and amendments, as soon as they are published thus ensuring prompt receipt of all publications in this area. Full details of the SOS may be obtained from Marketing Department, BSI, Linford Wood, Milton Keynes, MK14 6LE (Telephone 0908 320033) quoting reference SOS Subsection 444.

Terminology and conventions. The text of the IEEE 802.2 has been approved as suitable for publication as a Draft for Development. Some terminology and certain conventions are not identical with those used in Drafts for Development; attention is drawn especially to the following.

Wherever the words 'IEEE standard 802.2' appear, they should be read as 'DD 99 : Part 1'.

Cross-references

The publications to which reference is made in the text bear the following relationships to British Standard publications.

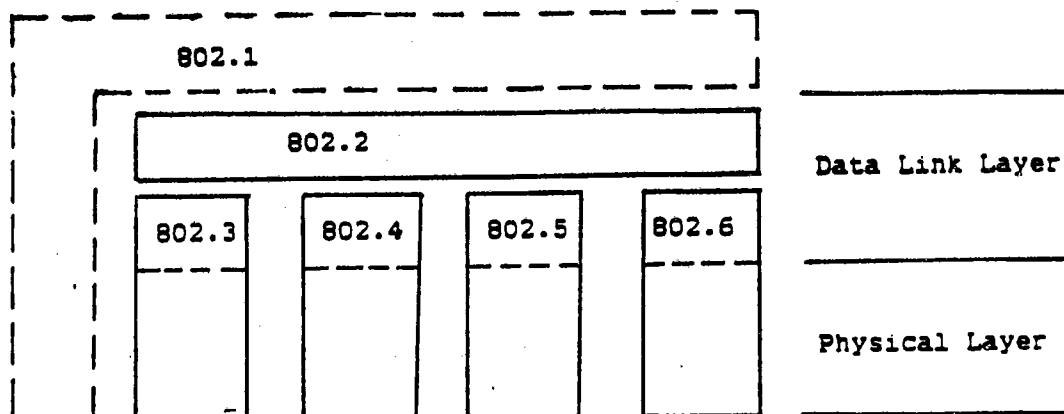
ISO/DRS 7809 will be published as an identical British Standard, BS 5397 : Part 5 which will supersede BS 5397 : Parts 3 and 4. ISO/DRS 7809 is a draft revision of ISO 7809-1984.

ISO/DRS 4335 was published as ISO 4335-1984 on 15 December 1984 and will be published as an identical British Standard, BS 5397 : Part 2 : 1985 which will supersede BS 5397 : Part 2 : 1982.

ISO 7498-1984 is identical to BS 6568 : 1984. It is technically equivalent to CCITT Draft Recommendation X.200.

FOREWORD

IEEE Standard 802 is a family of standards for local area networks that deals with the physical and data link layers as defined by the ISO Open Systems Interconnection Basic Reference Model. The relationship between the various members of the IEEE 802 family is shown below:



The Standard defines four media access technologies each appropriate for particular applications or system costs objectives. These media access standards are defined for a variety of alternative physical media. The four standards defining these media access technologies are:

- IEEE Standard 802.3, a bus utilizing CSMA/CD as the access method.
- IEEE Standard 802.4, a bus utilizing token passing as the access method.
- IEEE Standard 802.5, a ring utilizing token passing as the access method.
- IEEE Standard 802.6, a metropolitan area network access method (under study).

The Logical Link Control protocol standard, IEEE 802.2 (the subject of this standard), is intended to be used in conjunction with the four media access standards.

A companion document (in the form of a Technical Report or Guideline), IEEE 802.1, describes the relationship among these standards and their relationship to the ISO Open Systems Interconnection Basic Reference Model in more detail. This companion document also explains the relationship of the IEEE 802 Standards to higher layer protocols, and discusses internetworking and network management issues.

The reader of this Standard (IEEE 802.2) is urged to become familiar with the other IEEE 802.X standards.

ABBREVIATIONS

ABM	Asynchronous Balanced Mode
ACK	ACKnowledge
ADCCP	Advanced Data Communication Control Procedures
ADM	Asynchronous Disconnected Mode
ANSI	American National Standards Institute
C	Command
CCITT	International Telegraph and Telephone Consultative Committee
COMVII	Commission VII
C/R	Command/Response
DA	Destination Address
DCE	Data Circuit-terminating Equipment
DIS	Draft International Standard
DISC	DISConnect
DM	Disconnected Mode
DSAP	Destination Service Access Point
DTE	Data Terminal Equipment
F	Final
FCS	Frame Check Sequence
FRMR	FRaMe Reject
HDLC	High level Data Link Control
I	Information
I	Information transfer format
IEEE	Institute of Electrical and Electronic Engineers
ISO	International Organization for Standardization
LAN	Local Area Network
LAPB	Link Access Procedure, Balanced
LLC	Logical Link Control

LSAP	Link layer Service Access Point
LSB	Least Significant Bit
LSDU	Link layer Service Data Unit
M	Modifier function bit
MAC	Medium Access Control
N(R)	Receive sequence Number
N(S)	Send sequence Number
OSI	Open Systems Interconnection
P	Poll
PDU	Protocol Data Unit
P/F	Poll/Final
PHY	PHYSical
R	Response
REJ	REJect
RNR	Receive Not Ready
RR	Receive Ready
S	Supervisory format
S	Supervisory function bit
SA	Source Address
SABME	Set Asynchronous Balanced Mode Extended
SAP	Service Access Point
SSAP	Source Service Access Point
TEST	TEST
U	Unnumbered format
UA	Unnumbered Acknowledgment
UI	Unnumbered Information
V(R)	Receive state Variable
V(S)	Send state Variable
XID	eXchange IDentification

DEFINITIONS

For the purpose of this standard the following definition apply:

Accept: The condition assumed by a LLC upon accepting a correctly received PDU for processing.

Address Fields (DSAP and SSAP): The ordered pair of service access point addresses at the beginning of an LLC PDU which identifies the LLC(s) designated to receive the PDU and the LLC sending the PDU. Each address field is one octet in length.

Basic Status: A LLC's capability to send or receive a PDU containing an information field.

Command: In data communications, an instruction represented in the control field of a PDU and transmitted by a LLC. It causes the addressed LLC(s) to execute a specific data link control function.

Command PDU: All PDUs transmitted by a LLC in which the C/R bit is equal to "0."

Control Field (C): The field immediately following the DSAP and SSAP address fields of a PDU. The content of the control field is interpreted by the receiving destination LLC(s) designated by the DSAP address field:

- a. As a command, from the source LLC designated by the SSAP address field, instructing the performance of some specific function;
- b. As a response, from the source LLC designated by the SSAP address field.

Data Link: As assembly of two or more terminal installations and the interconnecting communications channel operating according to a particular method that permits information to be exchanged; in this context the term "terminal installation" does not include the data source and the data sink.

Data Link Layer: The conceptual layer of control or processing logic existing in the hierarchical structure of a station that is responsible for maintaining control of the data link. The data link layer functions provide an interface between the station higher layer logic and the data link. These functions include address/control field interpretation, channel access and command PDU/response PDU generation, transmission and interpretation.

Exception Condition: The condition assumed by a LLC upon receipt of a command PDU which it cannot execute due to either a transmission error or an internal processing malfunction.

Global (Broadcast) DSAP Address: The predefined LLC DSAP address (all ones) used as a broadcast (all parties) address. It can never be the address of a single LLC on the data link.

Group (Multicast) DSAP Address: A destination address assigned to a collection of LLCs to facilitate their being addressed collectively. The least significant bit shall be set equal to "1".

Higher Layer: The conceptual layer of control or processing logic existing in the hierarchical structure of a station that is above the data link layer and upon which the performance of data link layer functions are dependent; for example, device control, buffer allocation, LLC station management, etc.

Information Field (I): The sequence of octets occurring between the control field and the end of the LLC PDU. The information field contents of I, TEST and UI PDUs are not interpreted at the LLC sublayer.

Invalid Frame: A PDU that either: a. Does not contain an integral number of octets, b. Does not contain at least two address octets and a control octet, c. Is identified by the Physical Layer or MAC Sublayer as containing data bit errors.

LLC: That part of a data station that supports the Logical Link Control functions of one or more logical links. The LLC generates command PDUs and response PDUs for transmission, and interprets received command PDUs and response PDUs. Specific responsibilities assigned to a LLC include:

- a. Initiation of control signal interchange;
- b. Organization of data flow;
- c. Interpretation of received command PDUs and generation of appropriate response PDUs, and;
- d. Actions regarding error control and error recovery functions in the LLC Sublayer.

MAC: That part of a data station that supports the Medium Access Control functions that reside just below the Logical Link Control sublayer. The MAC procedures include framing/deframing data units, performing error checking, and acquiring the right to use the underlying physical medium.

N-layer: A subdivision of the architecture, constituted by subsystems of the same rank (N).

N-user: An N+1 entity that uses the services of the N-layer, and below, to communicate with another N+1 entity.

Octet: A bit-oriented element that consists of eight contiguous binary bits.

Peer Protocol: The sequence of message exchanges between two entities in the same layer that utilize the services of the underlying layers to effect the successful transfer of data and/or control information from one location to another location.

Protocol Data Unit (PDU): The sequence of contiguous octets delivered as a unit from or to the MAC Sublayer. A valid LLC PDU is at least 3 octets in length, and contains two address fields and a control field. A PDU may or may not include an information field in addition.

Response: In data communications, a reply represented in the control field of a response PDU. It advises the addressed destination LLC of the action taken by the source LLC to one or more command PDUs.

Response PDU: All PDUs sent by a LLC in which the C/R bit is equal to "1".

Services: The capabilities and features provided by an N-layer to an N-user.

Service_class (use in primitives): A parameter used to convey the type of service required or desired (e.g., priority).

1. INTRODUCTION

1.1. Scope and Purpose This standard is one of a set of standards produced to facilitate the interconnection of computers and terminals on a Local Area Network (LAN). It is related to the other standards by the Reference Model for Open Systems Interconnection.

NOTE: The exact relationship of the layers described in this International Standard to the layers defined by the OSI Reference Model is under study.

This standard describes the functions, and features and protocol of the Logical Link Control (LLC) Sublayer in the IEEE 802 Local Area Network Protocol. The LLC sublayer constitutes the top sublayer in the Data Link Layer (see Figure 1.1) and is common to the various medium access methods that are defined and supported by the IEEE 802 activity. Separate standards describe each medium access method individually and indicate the additional features and functions that are provided by the Medium Access Control (MAC) Sublayer in each case to complete the functionality of the Data Link Layer as defined in the 802 architectural reference model.

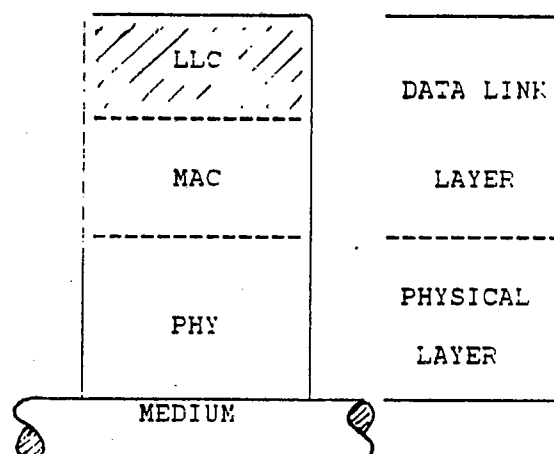


Figure 1.1. Relationship to LAN Reference Model

This standard describes the LLC Sublayer Interface Service Specifications to the Network Layer (Layer 3), to the MAC Sublayer, and to the LLC Sublayer Management function. The interface service specification to the Network Layer provides a description of the various services that the Logical Link Control Sublayer, plus underlying layers and sublayers, offer to the Network Layer, as viewed from the Network Layer. The interface service specification to the MAC Sublayer provides a description of the services that the LLC Sublayer requires of the MAC Sublayer. These services are defined so as to be independent of the form of the medium access methodology, and of the nature of the medium itself. The interface service specification to the LLC Sublayer Management function provides a description of the management services that are provided to the LLC Sublayer. All of the above Interface Service Specifications are given in the form of primitives that represent in an abstract way the logical exchange of information and control between the LLC Sublayer and the identified service function (Network Layer, MAC Sublayer or LLC Sublayer Management function). They do not specify or constrain the implementation of entities or

interfaces.

This standard provides a description of the peer-to-peer protocol procedures that are defined for the transfer of information and control between any pair of Data Link Layer service access points on a local area network. The LLC Procedures are independent of the type of medium access method used in the particular local area network.

To satisfy a broad range of potential applications, two types of data link control operation are included (see Section 4). The first type of operation (see Section 6) provides a data-link-connectionless service across a data link with minimum protocol complexity. This type of operation may be useful when higher layers provide any essential recovery and sequencing services so that these do not need replicating in the Data Link Layer. In addition, this type of operation may prove useful in applications where it is not essential to guarantee the delivery of every Data Link Layer data unit. This type of service is described in this standard in terms of "logical data links". The second type of operation (see Section 7) provides a data-link-connection-oriented service across a data link comparable to existing data link control procedures provided in national and international standards such as ADCCP (see 1.3 (3)) or HDLC (see 1.3 (1)). This service includes support of sequenced delivery of Data Link Layer data units, and a comprehensive set of Data Link Layer error recovery techniques. This second type of service is described in this standard in terms of "data link connections".

This standard identifies two distinct "classes" of LLC operation. Class I provides data-link-connectionless service only. Class II provides data-link-connection-oriented service plus data-link-connectionless service. Either class of operation may be supported.

The basic protocols described herein are peer protocols for use in multi-station multi-access environments. Because of the multi-station, multi-access environment, it shall be possible for a station to be involved in a multiplicity of peer protocol data exchanges with a multiplicity of different stations over a multiplicity of different logical data links and/or data link connections that are carried by a single Physical Layer over a single physical medium. Each unique to-from pairing at the Data Link Layer shall define a separate logical data link or data link connection with separate logical parameters and variables. Except where noted, the procedures described in this chapter shall relate to each Data Link Layer logical data link or data link connection separately and independently from any other logical data link or data link connection that might exist at the stations involved.

1.2. Standards Compatibility. The peer protocol procedures defined in Section 5 utilize some of the concepts and principles, as well as commands and responses, of the balanced link control procedures known as Asynchronous Balanced Mode (ABM), as defined in ISO/DIS 7809-1983 (The ABM procedures provided the basis upon which the CCITT Recommendation X.25 Level 2 LAPB procedures were defined.) The frame structure defined for the Data Link Layer procedures as a whole is defined in part in Section 3 of this section and in part in those standards that define the various medium access control (MAC) procedures. The combination of a MAC sublayer address and an LLC sublayer address is unique to each Data Link Layer service access point in the local area network. (NOTE - This division of Data Link Layer addressing space into separate MAC and LLC address fields is not presently a part of any present ISO or ANSI Data Link Layer standard.)

1.3. References.

(1) ISO

ISO/DRS 4335 Revised, Information processing systems, Data communications, High-level data link control procedures, Consolidation of elements of procedures

ISO 7809-1984, Information processing systems, Data communications, High-level data link control procedures, Consolidation of classes of procedures

ISO 7498-1984, Information processing systems, Open systems interconnection, Basic reference model

(2) CCITT

Recommendation X.25, Interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) for terminals operating in the packet mode on public data networks.

Draft Recommendation X.200, Reference model on open systems interconnection for CCITT applications

(3) ANSI

X3.66-1979, American National Standard for Advanced Data Communications Control Procedures (ADCCP)

2. LLC SUBLAYER INTERFACE SERVICE SPECIFICATIONS

This section covers the services required of or by the Logical Link Control (LLC) Sublayer at the logical interfaces with the Network Layer, the MAC Sublayer, and the LLC Sublayer Management function.

In general, the services of a layer (or sublayer) are the capabilities which it offers to a user in the next higher layer (or sublayer). In order to provide its service, a layer (or sublayer) builds its functions on the services which it requires from the next lower layer (or sublayer). Figure 2.a. illustrates this notion of service hierarchy and shows the relationship of the two correspondent n-users and their associated n-layer (or sublayer) peer protocol entities.

Services are specified by describing the information flow at the interface between the n-user and the n-layer (or sublayer). This information flow is modeled by discrete, instantaneous interface events, which characterize the provision of a service. Each event consists of passing a service primitive from one layer (or sublayer) to the other through an n-layer (or sublayer) service access point associated with an n-user. Service primitives convey the information required in providing a particular service. These service primitives are an abstraction in that they specify only the service provided rather than the means by which the service is provided. This definition of service is independent of any particular interface implementation.

Services are specified by describing the service primitives and parameters which characterize each service. A service may have one or more related primitives which constitute the interface activity which is related to the particular service. Each service primitive may have zero or more parameters which convey the information required to provide the service.

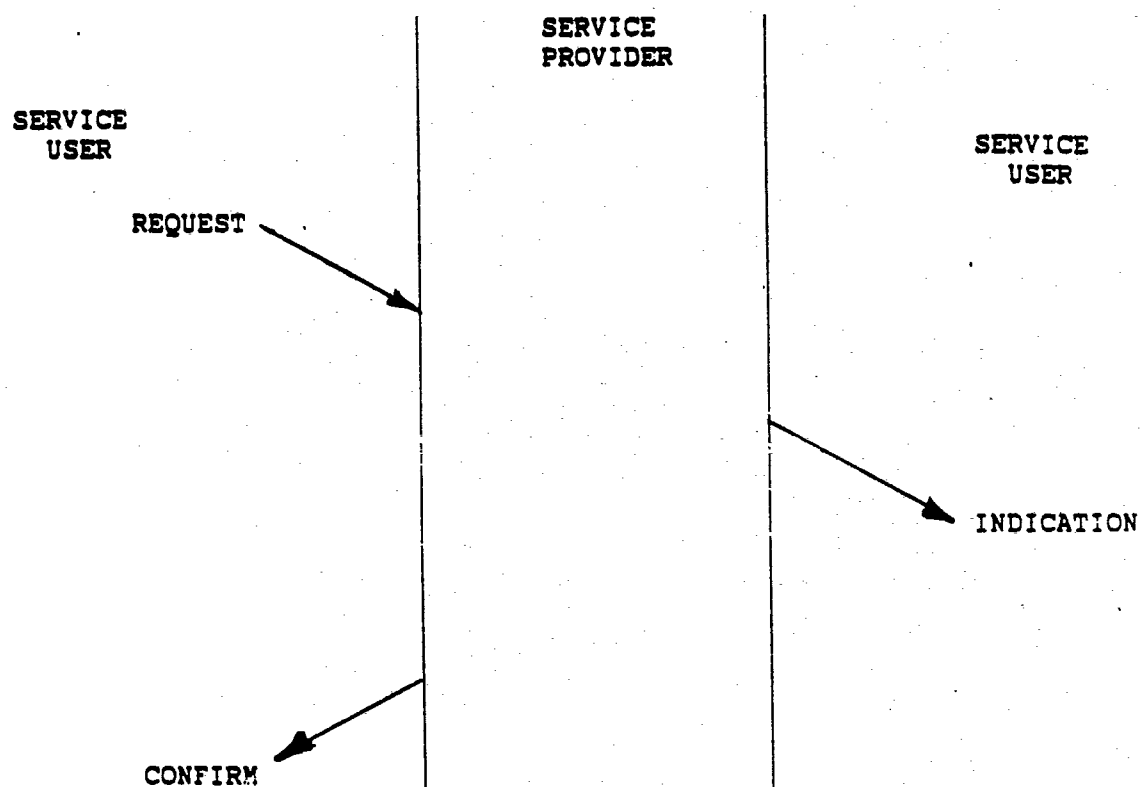


Figure 2.a. Service Primitives

Primitives are of three generic types:

- REQUEST** The request primitive is passed from the n-user to the n-layer (or sublayer) to request that a service be initiated.
- INDICATION** The indication primitive is passed from the n-layer (or sublayer) to the n-user to indicate an internal n-layer (or sublayer) event which is significant to the n-user. This event may be logically related to a remote service request, or may be caused by an event internal to the n-layer (or sublayer).
- CONFIRM** The confirm primitive is passed from the n-layer (or sublayer) to the n-user to convey the results of one or more associated previous service request(s). This primitive may indicate either failure to comply or some level of compliance. It does not necessarily indicate any activity at the remote peer interface.

Possible relationships among primitive types are illustrated by the time sequence diagrams shown in Figure 2.b. The figure also indicates the logical relationship of the primitive types. Primitive types which occur earlier in time and are connected by dotted lines in the diagrams are the logical antecedents of subsequent primitive types. Note that the logical and time relationship of the indication and the confirm primitive types are specified by the semantics of a particular service.

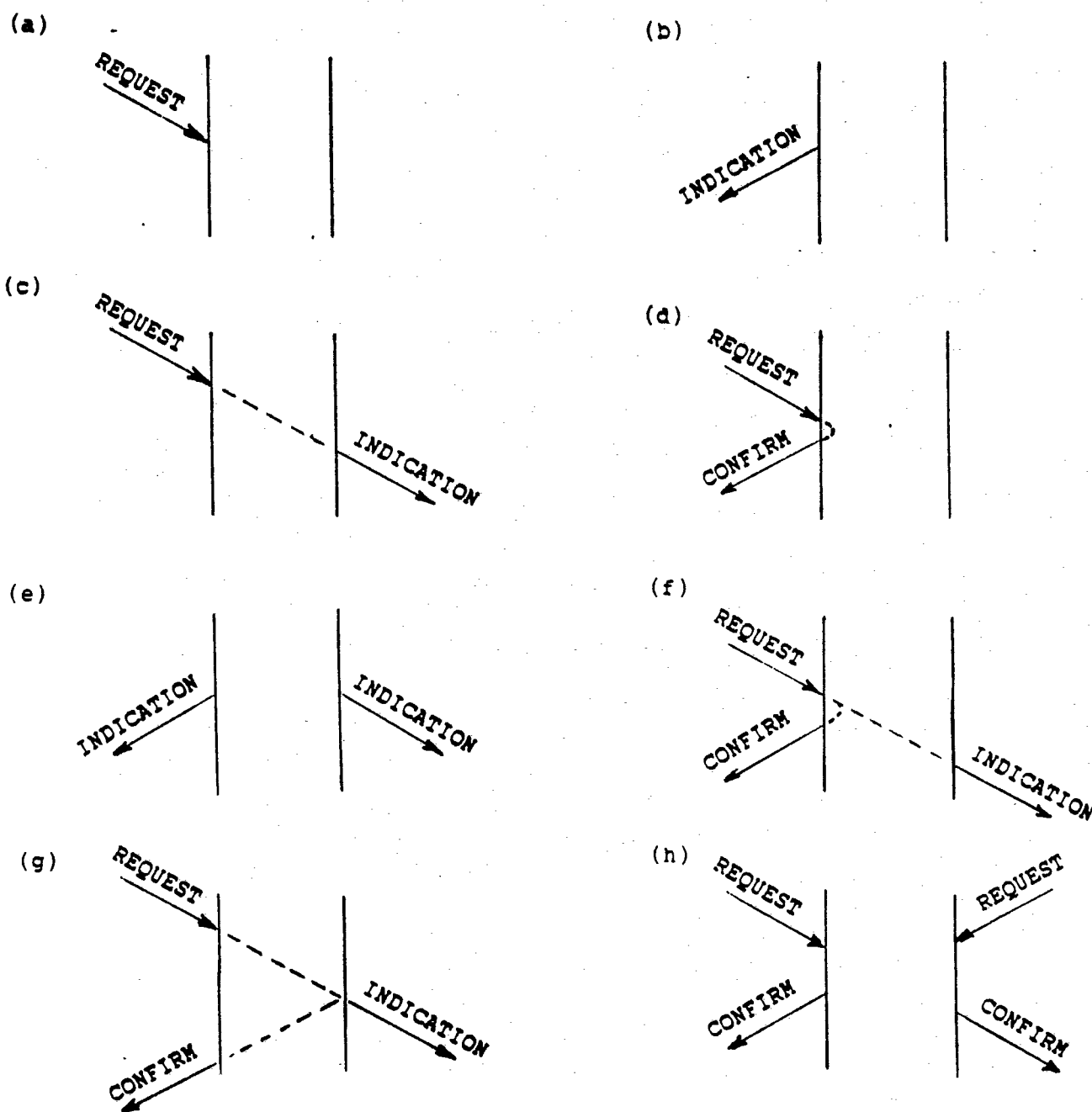


Figure 2.b. Time Sequence Diagram

2.1. Network Layer/LLC Sublayer Interface Service Specification. This section specifies the services required of the Logical Link Control (LLC) Sublayer by the Network Layer, as viewed from the Network Layer, to allow a local Network Layer entity to exchange packets with remote peer Network Layer entities. The services are described in an abstract way and do not imply any particular implementation or any exposed interface.

Two forms of service are provided - unacknowledged connectionless and connection-oriented:

- **Unacknowledged Connectionless Services.** The unacknowledged connectionless data transfer service provides the means by which network entities can exchange link service data units (LSDUs) without the establishment of a data link level connection. The data transfer can be point-to-point, multi-cast, or broadcast.

- **Connection-Oriented Services.** This set of services provides the means for establishing, using, resetting, and terminating data link layer connections. These connections are point-to-point connections between LSAPs.

The connections establishment service provides the means by which a network entity can request, or be notified of, the establishment of data link layer connections.

The connection-oriented data transfer service provides the means by which a network entity can send or receive LSDUs over a data link layer connection. This service also provides data link layer sequencing, flow control, and error recovery.

The connection reset service provides the means by which established connections can be returned to the initial state.

The connection termination service provides the means by which a network entity can request or be notified of the termination of data link layer connections.

The connection flow control service provides the means to control the flow of data associated with a specified connection, across the network layer/data link layer interface.

2.1.1. Overview of Interactions.

2.1.1.1. Unacknowledged Connectionless Services.

2.1.1.1.1. **Unacknowledged Connectionless Data Transfer.** The primitives associated with unacknowledged connectionless data transfer are:

L_DATA.request

L_DATA.indication

The L_DATA.request primitive is passed to the LLC sublayer to request that an LSDU be sent using unacknowledged connectionless procedures. The L_DATA.indication primitive is passed from the LLC sublayer to indicate the arrival of an LSDU.

2.1.1.2. Connection-Oriented Services.

2.1.1.2.1. **Connection Establishment.** The service primitives associated with connection establishment are:

- L_CONNECT.request
- L_CONNECT.indication
- L_CONNECT.confirm

The L_CONNECT.request primitive is passed to the LLC sublayer to request that a logical link connection be established between a local LSAP and a remote LSAP. The L_CONNECT.indication primitive is passed from the LLC sublayer to indicate the results of an attempt by a remote entity to establish a connection to a local LSAP. The L_CONNECT.confirm primitive is passed from the LLC sublayer to convey the results of the previous associated L_CONNECT.request primitive.

2.1.1.2.2. *Connection-Oriented Data Transfer.* The primitives associated with connection-oriented data transfer are:

- L_DATA_CONNECT.request
- L_DATA_CONNECT.indication
- L_DATA_CONNECT.confirm

The L_DATA_CONNECT.request primitive is passed to the LLC sublayer to request that an LSDU be sent using connection-oriented procedures. The L_DATA_CONNECT.indication primitive is passed from the LLC sublayer to indicate the arrival of an LSDU. The L_DATA_CONNECT.confirm primitive is passed by the LLC sublayer to convey the results of the previously associated L_DATA_CONNECT.request primitive.

2.1.1.2.3. *Connection Termination.* The primitives associated with connection termination are:

- L_DISCONNECT.request
- L_DISCONNECT.indication
- L_DISCONNECT.confirm

The L_DISCONNECT.request primitive is passed to the LLC sublayer to request the immediate termination of a link connection. The L_DISCONNECT.indication primitive is passed from the LLC sublayer to indicate to the network layer that a connection has been terminated. The L_DISCONNECT.confirm primitive is passed by the LLC sublayer to convey the results of the previous associated L_DISCONNECT.request primitive.

2.1.1.2.4. *Connection Reset.* The primitives associated with connection resetting are:

- L_RESET.request
- L_RESET.indication
- L_RESET.confirm

The `L_RESET.request` primitive is passed to the LLC sublayer to request that a connection be immediately reset to the initial state. The `L_RESET.indication` primitive is passed from the LLC sublayer to indicate that a connection has been reset. The `L_RESET.confirm` primitive is passed from the LLC sublayer to convey the results of the previous associated `L_RESET.request` primitive.

2.1.1.2.5. Connection Flow Control. The primitives associated with connection flow control are:

`L_CONNECTION_FLOWCONTROL.request`

`L_CONNECTION_FLOWCONTROL.indication`

The `L_CONNECTION_FLOWCONTROL.request` primitive is passed to the LLC sublayer to control the flow from the LLC sublayer of `L_DATA_CONNECT.indication` primitives related to a connection. The `L_CONNECTION_FLOWCONTROL.indication` primitive is passed from the LLC sublayer to control the flow from the network layer of `L_DATA_CONNECT.request` primitives related to a connection.

2.1.2. Detailed Service Specifications. This section describes in detail the primitives and parameters associated with the identified services. Note that the parameters are specified in an abstract sense. The parameters specify the information that must be available to the receiving entity. A specific implementation is not constrained in the method of making this information available. The LLC sublayer may provide local confirm mechanisms for request type primitives.

The "local_address" and "remote_address" parameters provide at a minimum the logical concatenation of the MAC address field (SA and/or DA) and the LLC address field (SSAP and/or DSAP). An implementation of connection-oriented services may make use of a locally significant connection identifier to imply local and remote address parameters. The "l_sdu" parameter may be provided by actually passing the link service data unit, by passing a pointer, or by other means. The "service_class" parameter provides the priority associated with the data unit transfer. The "service_class" parameter is passed transparently to the underlying MAC sublayer via the appropriate LLC/MAC primitives, see 2.2. The "status" parameter indicates the degree of success associated with the data unit transfer. The "reason" parameter provides an explanation of the disconnection, including a request by the remote entity, or an error internal to the LLC sublayer. The "amount" parameter provides information regarding the amount of data that the LLC entity is allowed to pass.

2.1.2.1. *L_DATA.request.*

2.1.2.1.1. Function. This primitive is the service request primitive for the unacknowledged connectionless data transfer service.

2.1.2.1.2. Semantics of the Service Primitive. The primitive shall provide parameters as follows:

```

L_DATA.request(
    local_address,
    remote_address,
    l_sdu,
    service_class
)

```

The `local_address` and `remote_address` parameters specify the local and remote LSAPs involved in the data unit transfer. The `remote_address` may specify either an individual or group address. The `l_sdu` parameter specifies the link service data unit to be transferred by the link layer entity. The `service_class` parameter specifies the priority desired for the data unit transfer.

2.1.2.1.3. When Generated. This primitive is passed from the network layer to the LLC sublayer to request that a LSDU be sent to one or more remote LSAP(s) using unacknowledged connectionless procedures.

2.1.2.1.4. Effect on Receipt. Receipt of this primitive causes the LLC sublayer to attempt to send the LSDU using unacknowledged connectionless procedures.

2.1.2.1.5. Additional Comments. This primitive is independent of any connection with the remote LSAP.

A possible logical sequence of primitives associated with successful unacknowledged connectionless data unit transfer is illustrated in Figure 2.b.(c).

2.1.2.2. *L_DATA.indication.*

2.1.2.2.1. Function. This primitive is the service indication primitive for the unacknowledged connectionless data unit transfer service.

2.1.2.2.2. Semantics of the Service Primitive. The primitive shall provide parameters as follows:

```

L_DATA.indication(
    local_address,
    remote_address,
    l_sdu,
    service_class
)

```

The `local_address` and `remote_address` parameters specify the local and remote LSAPs involved in the data unit transfer. The `local_address` may be the address of a local LSAP, or may be a group address specifying multiple LSAPs, including a local LSAP. The `l_sdu` parameter specifies the link service data unit which has been received by the LLC sublayer entity. The `service_class` parameter specifies the priority desired for the data unit transfer.

2.1.2.2.3. When Generated. This primitive is passed from the LLC sublayer to the network layer to indicate the arrival of an LSDU from the specified remote entity.

2.1.2.2.4. Effect on Receipt. The effect of receipt of this primitive by the network layer is unspecified.

2.1.2.2.5. Additional Comments. This primitive is independent of any connection with the remote LSAP.

In the absence of errors, the contents of the `l_sdu` parameter are logically complete and unchanged relative to the `l_sdu` parameter in the associated `L_DATA.request` primitive.

2.1.2.3. `L_CONNECT.request`.

2.1.2.3.1. Function. This primitive is the service request primitive for the connection establishment service.

2.1.2.3.2. Semantics of the Service Primitive. The primitive shall provide parameters as follows:

```
L_CONNECT.request(
    local_address,
    remote_address,
    service_class
)
```

The `local_address` and `remote_address` parameters specify the local and remote LSAPs which are to be connected. The `service_class` parameter specifies the priority desired for the connection.

2.1.2.3.3. When Generated. This primitive is passed from the network layer to the LLC sublayer when the network layer entity wishes to establish a logical link connection, of a given service class, to a remote LSAP.

2.1.2.3.4. Effect on Receipt. The receipt of this primitive by the LLC sublayer causes the local LLC entity to initiate the establishment of a connection with the remote LLC entity.

2.1.2.3.5. Additional Comments. A possible logical sequence of primitives associated with successful connection establishment is illustrated in Figure 2.b.(g).

2.1.2.4. `L_CONNECT.indication`.

2.1.2.4.1. Function. This primitive is the service indication primitive for the connection establishment service.

2.1.2.4.2. Semantics of the Service Primitive. The primitive shall provide parameters as follows:

```
L_CONNECT.indication(
    local_address,
    remote_address,
    status,
    service_class
)
```

The `local_address` and `remote_address` parameters specify the local and remote LSAPs which are to be connected. One status code must indicate a successful connection attempt. Other status codes indicate the reason for failure. The `service_class` parameter indicates the priority desired for the connection, if the attempt was successful.

2.1.2.4.3. When Generated. This primitive is passed from the LLC sublayer to the network layer to indicate either that a connection of a certain service class has been established, or that a connection attempt has been refused.

2.1.2.4.4. Effect on Receipt. The network entity may use this connection for data unit transfer.

2.1.2.4.5. Additional Comments. The basis for connection refusal by the LLC sublayer is implementation dependent but may include such reasons as access permission, or unacceptable connection service class.

If the network layer entity does not wish to maintain a connection, then a `L_DISCONNECT.request` primitive is issued.

2.1.2.5. `L_CONNECT.confirm`.

2.1.2.5.1. Function. This primitive is the service confirm primitive for the connection establishment service.

2.1.2.5.2. Semantics of the Service Primitive. The primitive shall provide parameters as follows:

```
L_CONNECT.confirm(
    local_address,
    remote_address,
    status,
    service_class
)
```

The `local_address` and `remote_address` parameters specify the local and remote LSAPs which are to be connected. One status code must indicate a successful connection attempt. Other status codes indicate the reason for failure. The `service_class` parameter indicates the priority provided for the connection, if the attempt was successful.

2.1.2.5.3. When Generated. This primitive is passed by the LLC sublayer to the network layer to convey the results of the previous associated `L_CONNECT.request` primitive. The results indicate either that the connection attempt was successful, or that a connection of the specified service class could not be achieved.

2.1.2.5.4. Effect on Receipt. The network entity may use this connection for data unit transfer.

2.1.2.5.5. Additional Comments. If the connection attempt was successful, this primitive indicates that the remote LLC sublayer entity received and positively acknowledged the connection attempt, and that it passed an `L_CONNECT.indication` to the remote user network entity.

2.1.2.6. `L_DATA_CONNECT.request`.

2.1.2.6.1. Function. This primitive is the service request primitive for the connection-oriented data unit transfer service.

2.1.2.6.2. Semantics of the Service Primitive. The primitive shall provide parameters as follows:

```

L_DATA_CONNECT.request(
    local_address,
    remote_address,
    l_sdu
)

```

The `local_address` and `remote_address` parameters specify the local and remote LSAPs of the connection. The `l_sdu` parameter specifies the link service data unit to be transferred by the LLC sublayer entity.

2.1.2.6.3. When Generated. This primitive is passed from the network layer to the LLC sublayer to request that a LSDU be transferred to a remote LSAP over an existing connection.

2.1.2.6.4. Effect on Receipt. The receipt of this primitive by the LLC sublayer causes the LLC sublayer to transfer the LSDU over the specified connection using connection-oriented procedures.

2.1.2.6.5. Additional Comments. A possible logical sequence of primitives associated with successful connection-oriented data unit transfer is illustrated in Figure 2.b.(g).

2.1.2.7. L_DATA_CONNECT.indication.

2.1.2.7.1. Function. This primitive is the service indication primitive for the connection-oriented data unit transfer service.

2.1.2.7.2. Semantics of the Service Primitive. The primitive shall provide parameters, as follows:

```

L_DATA_CONNECT.indication(
    local_address,
    remote_address,
    l_sdu
)

```

The `local_address` and `remote_address` parameters specify the local and remote LSAPs of the connection. The `l_sdu` parameter specifies the link service data unit which has been received by the LLC sublayer entity.

2.1.2.7.3. When Generated. This primitive is passed by the LLC sublayer to the network layer to indicate the arrival of an LSDU from the specified remote network layer entity over a particular connection.

2.1.2.7.4. Effect on Receipt. The effect of receipt of this primitive by the network layer is unspecified.

2.1.2.7.5. Additional Comments. The `L_DATA_CONNECT.request` primitive does not contain a `service_class` parameter because service class must be uniform for all `L_DATA_CONNECT.requests` in a particular connection.

In the absence of errors, the contents of the `l_sdu` parameter are logically complete and unchanged relative to the `l_sdu` parameter in the associated `L_DATA.request` primitive.

2.1.2.8. L_DATA_CONNECT.confirm.

2.1.2.8.1. *Function.* This primitive is the service confirm primitive for the connection-oriented data unit transfer service.

2.1.2.8.2. *Semantics of the Service Primitive.* The primitive shall provide parameters as follows:

```
L_DATA_CONNECT.confirm(  
    local_address,  
    remote_address,  
    status  
)
```

The local_address and remote_address parameters specify the local and remote LSAPs of the connection. The status parameter indicates the success or failure of the one or more previous associated connection-oriented data unit transfer request(s).

2.1.2.8.3. *When Generated.* This primitive is passed from the LLC sublayer to the network layer to indicate the success or failure of the one or more previous associated connection-oriented data unit transfer request(s).

2.1.2.8.4. *Effect on Receipt.* The effect of receipt of this primitive by the network layer is unspecified.

2.1.2.8.5. *Additional Comments.* If the transfer was successful this primitive indicates that the remote LLC sublayer entity received and positively acknowledged the LSDU.

2.1.2.9. L_DISCONNECT.request.

2.1.2.9.1. *Function.* This primitive is the service request primitive for the connection termination service.

2.1.2.9.2. *Semantics of the Service Primitive.* The primitive shall provide parameters as follows:

```
L_DISCONNECT.request(  
    local_address,  
    remote_address  
)
```

The local_address and remote_address parameters specify the local and remote LSAPs of the connection to be terminated.

2.1.2.9.3. *When Generated.* This primitive is passed from the network layer to the LLC sublayer when the network layer entity wishes to terminate a connection.

2.1.2.9.4. *Effect on Receipt.* Receipt of this primitive causes the LLC sublayer to immediately terminate the connection.

2.1.2.9.5. *Additional Comments.* All unacknowledged LSDUs are discarded. The connection termination service is an abortive service. That is, no guarantee of delivery can be assumed about data that is not yet acknowledged at a higher layer. Thus, graceful disconnection (i.e. without loss of data) is the responsibility of a higher layer protocol.

A possible logical sequence of primitives associated with successful connection termination is illustrated in Figure 2.b.(g).

2.1.2.10. *L_DISCONNECT.indication.*

2.1.2.10.1. *Function.* This primitive is the service indication primitive for the connection termination service.

2.1.2.10.2. *Semantics of the Service Primitive.* The primitive shall provide parameters as follows:

```
L_DISCONNECT.indication(
    local_address,
    remote_address,
    reason
)
```

The *local_address* and *remote_address* parameters specify the local and remote LSAPs of the terminated connection. The *reason* parameter specifies the reason for the disconnection. The reasons for disconnection may include a request by the remote entity, or an error internal to the LLC sublayer.

2.1.2.10.3. *When Generated.* This primitive is passed from the LLC sublayer to the network layer to inform the network layer that a connection has been terminated.

2.1.2.10.4. *Effect on Receipt.* The network entity may no longer use this connection for data unit transfer.

2.1.2.10.5. *Additional Comments.* All unacknowledged LSDUs are discarded. The connection termination service is an abortive service. That is, no guarantee of delivery can be assumed about data that is not yet acknowledged at a higher layer. Thus, graceful disconnection (i.e., without loss of data) is the responsibility of a higher layer protocol.

2.1.2.11. *L_DISCONNECT.confirm.*

2.1.2.11.1. *Function.* This primitive is the service confirm primitive for the connection termination service.

2.1.2.11.2. *Semantics of the Service Primitive.* The primitive shall provide parameters as follows:

```
L_DISCONNECT._confirm(
    local_address,
    remote_address,
    status
)
```

The *local_address* and *remote_address* parameters specify the local and remote LSAPs of the terminated connection. The *status* parameter indicates whether or not the remote LLC sublayer entity acknowledged the disconnection attempt.

2.1.2.11.3. *When Generated.* This primitive is passed from the LLC sublayer to the network layer to inform the network layer entity when connection termination is complete.

2.1.2.11.4. *Effect on Receipt.* The network entity may no longer use this connection for data unit transfer.

2.1.2.11.5. *Additional Comments.* The L_DISCONNECT.confirm primitive indicates that the remote LLC sublayer has acknowledged the disconnect. The remote LLC sublayer entity is expected to pass an L_DISCONNECT.indication to the remote user network layer.

2.1.2.12. L_RESET.request.

2.1.2.12.1. *Function.* This primitive is the service request primitive for the connection reset service.

2.1.2.12.2. *Semantics of the Service Primitive.* The primitive shall provide parameters as follows:

```
L_RESET.request(
    local_address,
    remote_address
)
```

The local_address and remote_address parameters specify the local and remote LSAPs of the connection to be reset.

2.1.2.12.3. *When Generated.* This primitive is passed from the network layer to the LLC sublayer to request that a connection be reset to the initial state.

2.1.2.12.4. *Effect on Receipt.* Receipt of this primitive causes immediate resetting of the connection.

2.1.2.12.5. *Additional Comments.* All unacknowledged LSDUs are discarded. The connection reset service is an abortive service. That is, no guarantee of delivery can be assumed about data that is not yet acknowledged at a higher layer. Thus, graceful reset (i.e. without loss of data) is the responsibility of a higher layer protocol.

A possible logical sequence of primitives associated with successful connection reset is illustrated in Figure 2.b.(g).

2.1.2.13. L_RESET.indication.

2.1.2.13.1. *Function.* This primitive is the service indication primitive for the connection reset service.

2.1.2.13.2. *Semantics of the Service Primitive.* The primitive shall provide parameters as follows:

```
L_RESET.indication(
    local_address,
    remote_address,
    reason
)
```

The local_address and remote_address parameters specify the local and remote LSAPs of the reset connection. The reason parameter specifies the reason for the connection reset. One of the reason codes indicates that the reset was requested by a remote network entity. All other codes indicate that the reset originated within the LLC sublayer, (as shown in Figure 2.b.(e)) and may optionally provide implementation dependent information about the cause of the reset.

2.1.2.13.3. *When Generated.* This primitive is passed from the LLC sublayer to the network layer to indicate that the connection has been reset.

2.1.2.13.4. *Effect on Receipt.* The effect of receipt of this primitive by the network layer is unspecified.

2.1.2.13.5. *Additional Comments.* The reasons for the reset may include a request by the remote entity, or an error internal to the LLC sublayer. All unacknowledged LSDUs are discarded. The connection reset service is an abortive service. That is, no guarantee of delivery can be assumed about data that is not yet acknowledged at a higher layer. Thus, graceful reset (i.e. without loss of data) is the responsibility of a higher layer protocol.

2.1.2.14. *L_RESET.confirm.*

2.1.2.14.1. *Function.* This primitive is the service confirm primitive for the connection reset service.

2.1.2.14.2. *Semantics of the Service Primitive.* The primitive shall provide parameters as follows:

```
L_RESET.confirm(
    local_address,
    remote_address,
    status
)
```

The local_address and remote_address parameters specify the local and remote LSAPs of the connection. The status parameter indicates whether or not the remote LLC sublayer entity acknowledged the reset attempt.

2.1.2.14.3. *When Generated.* This primitive is passed from the LLC sublayer to the network layer to inform the network layer that a connection reset has been completed.

2.1.2.14.4. *Effect on Receipt.* The network entity may use this connection for data unit transfer.

2.1.2.14.5. *Additional Comments.* This primitive indicates that the remote LLC sublayer entity has acknowledged the reset. The remote LLC sublayer entity is expected to pass an L_RESET.indication to the remote user network entity.

2.1.2.15. *L_CONNECTION_FLOWCONTROL.request.*

2.1.2.15.1. *Function.* This primitive is the service request primitive for the connection flow control service.

2.1.2.15.2. *Semantics of the Service Primitive.* The primitive shall provide parameters as follows:

```
L_CONNECTION-FLOWCONTROL.request(
    local_address,
    remote_address,
    amount
)
```

The `local_address` and `remote_address` parameters specify the local and remote LSAPs of the connection to be flow controlled. The `amount` parameter specifies the amount of data the LLC sublayer entity is permitted to pass.

2.1.2.15.3. When Generated. This primitive is passed from the network layer to the LLC sublayer to request control of the flow of `L_DATA_CONNECT.indication` primitives associated with a connection from the LLC sublayer.

2.1.2.15.4. Effect on Receipt. Receipt of this primitive causes the LLC sublayer to adjust the amount of data that may be passed to the network layer.

2.1.2.15.5. Additional Comments. Control of the flow of data on a connection is independent of control of the flow on other connections. The amount of data permitted to be passed is dynamically updated by each request. If amount is specified as zero, then the associated flow is stopped. Specific implementations may allow amount to be specified in implementation specific units, and may allow amount to be specified as "infinite".

A possible logical sequence of primitives associated with an `L_CONNECTION_FLOWCONTROL.request` is illustrated in Figure 2.b.(a).

2.1.2.16. `L_CONNECTION_FLOWCONTROL.indication`.

2.1.2.16.1. Function. This primitive is the service indication primitive for the connection flow control service.

2.1.2.16.2. Semantics of the Service Primitive. The primitive shall provide parameters as follows:

```
L_CONNECTION_FLOWCONTROL.indication(
                                local_address,
                                remote_address,
                                amount
                                )
```

The `local_address` and `remote_address` parameters specify the local and remote LSAPs of the connection to be controlled. The `amount` parameter specifies the amount of data which the network entity is permitted to pass to avoid data loss.

2.1.2.16.3. When Generated. This primitive is passed from the LLC sublayer to the network layer to request control of the flow of `L_DATA_CONNECT.request` primitives associated with a connection from the network layer.

2.1.2.16.4. Effect on Receipt. Receipt of the primitive causes the network layer to adjust the amount of data that it is allowed to pass without data loss.

2.1.2.16.5. Additional Comments. Control of the flow of data on a connection is independent of control of the flow on other connections. The amount of data permitted to be passed is dynamically updated by each request. If amount is specified as zero, then the associated flow is stopped. Specific implementations may allow amount to be specified in implementation specific units, and may allow amount to be specified as "infinite".

A possible logical sequence of primitives associated with an `L_CONNECTION_FLOWCONTROL.indication` is illustrated in Figure 2.b.(b).

2.2. LLC Sublayer/MAC Sublayer Interface Service Specification. This section specifies the services required of the Medium Access Control (MAC) sublayer by the Logical Link Control (LLC) sublayer to allow the local LLC sublayer entity to exchange LLC data units with peer LLC sublayer entities. The services are described in an abstract way and do not imply any particular implementation or any exposed interface.

2.2.1. Overview of Interactions.

- `MA-DATA.request`
- `MA_DATA.indication`
- `MA_DATA.confirm`

2.2.2. Detailed Service Specification.

2.2.2.1. `MA_DATA.request`.

2.2.2.1.1. Function. This primitive defines the transfer of a MSDU from a local LLC sublayer entity to a single peer LLC entity, or multiple peer LLC entities in the case of group addresses.

2.2.2.1.2. Semantics of the Service Primitive. The semantics of the primitive are as follows:

```
MA_DATA.request(
    destination_address,
    m_sdu,
    requested_service_class
)
```

The `destination_address` parameter shall specify either an individual or a group MAC entity address. It must contain sufficient information to create the DA field that is appended to the frame by the local MAC sublayer entity as well as any Physical Layer address information (e.g., transmit frequency in broadband applications). The `m_sdu` parameter specifies the MAC service data unit to be transmitted by the MAC sublayer entity, which includes the DSAP, SSAP, C, and information (if present) fields as specified in Section 3, as well as sufficient information for the MAC sublayer entity to determine the length of the data unit. The `requested_service_class` parameter specifies the priority desired for the data unit transfer.

2.2.2.1.3. When Generated. This primitive is generated by the LLC sublayer entity whenever a MSDU must be transferred to a peer LLC entity or entities. This can be as a result of a request from higher layers or protocol, or from a MSDU generated internally to the LLC sublayer, such as required by Type 2 operation.

2.2.2.1.4. Effect of Receipt. The receipt of this primitive shall cause the MAC entity to append all MAC specified fields, including DA, SA, and any fields that are unique to the particular medium access method, and pass the properly formatted frame to the lower layers of protocol for transfer to the peer MAC sublayer entity or entities.

2.2.2.1.5. Additional Comments. None.**2.2.2.2. MA_DATA.indication.**

2.2.2.2.1. Function. This primitive defines the transfer of a MSDU from the MAC sublayer entity to the LLC sublayer entity, or entities in the case of group addresses. In the absence of errors, the contents of the *m_sdu* parameter are logically complete and unchanged relative to the *m_sdu* parameter in the associated *MA_DATA.request* primitive.

2.2.2.2.2. Semantics of the Service Primitive. The semantics of the primitive are as follows:

```
MA_DATA.indication(
    destination_address,
    source_address,
    m_sdu,
    reception_status,
    requested_service_class
)
```

The *destination_address* parameter shall be either an individual or a group address as specified by the DA field of the incoming frame. The *source_address* parameter must be an individual address as specified by the SA field of the incoming frame. The *m_sdu* parameter specifies the MAC service data unit as received by the local MAC entity. The *reception_status* parameter indicates the success or failure of the incoming frame. The *requested_service_class* parameter specifies the service class desired for this data unit transfer.

2.2.2.2.3. When Generated. The *MA_DATA.indication* primitive is passed from the MAC sublayer entity to the LLC sublayer entity or entities to indicate the arrival of a frame at the local MAC sublayer entity. Frames are reported only if at the MAC sublayer they are validly formatted, received without error, and their destination address designates the local MAC entity.

2.2.2.2.4. Effect of Receipt. The effect of receipt of this primitive by the LLC sublayer is dependent on the validity and content of the frame.

2.2.2.2.5. Additional Comments. If the local MAC sublayer entity is designated by the *destination_address* parameter of an *MA_DATA.request* primitive, the indication primitive will also be invoked by the MAC entity to the local LLC entity. This full duplex characteristic of the MAC sublayer may be due to unique functionality within the MAC sublayer or full duplex characteristics of the lower layers, (e.g., all frames transmitted to the broadcast address will invoke *MA_DATA.indication* primitives at all stations in the network including the station that generated the request.)

2.2.2.3. MA_DATA.confirm.

2.2.2.3.1. Function. This primitive has local significance and shall provide an appropriate reply to the LLC sublayer *MA_DATA.request* primitive signifying the success or failure of the request.

2.2.2.3.2. Semantics of the Service Primitive. The semantics of this primitive are as follows:

```
MA_DATA.confirm(
    transmission_status,
    provided_service_class
)
```

The `transmission_status` parameter is used to pass status information back to the local requesting LLC sublayer entity. It is used to indicate the success or failure of the previous associated `MA_DATA.request` primitive. The types of failures that can be associated with this primitive are dependent on the particular implementation as well as the type of Medium Access Control sublayer that is used, (e.g., excessive collisions may be a failure returned by a CSMA/CD MAC sublayer entity). The `provided_service_class` parameter specifies the service class that was provided for the data unit transfer.

2.2.2.3.3. When Generated. This primitive is generated in reply to an `MA_DATA.request` primitive from the local LLC sublayer entity.

2.2.2.3.4. Effect of Receipt. The effect of receipt of this primitive by the LLC sublayer is unspecified.

2.2.2.3.5. Additional Comments. It is assumed that sufficient information is available to the LLC sublayer to associate the confirm with the appropriate request.

2.3. LLC Sublayer/LLC Sublayer Management Function Interface Service Specification.

(This matter is the subject of further on-going study and resolution.)

3. LLC PROTOCOL DATA UNIT (PDU) STRUCTURE

3.1. General. This section defines in detail the Logical Link Control (LLC) Protocol Data Unit (PDU) structure for data communication systems using bit-oriented procedures. It defines the relative positions of the various components of the Protocol Data Unit (PDU). It defines the method for representing Data Link Layer service access point addresses (to or from Network Layer entities). It defines a partition of these addresses into individual and group addresses. Details of the control and information field allocation are specified in Section 5.

3.2. LLC PDU Format. All LLC PDUs shall conform to the format shown in Figure 3.2:

DSAP Address	SSAP Address	Control	Information
8 bits	8 bits	y bits	8*M*bits

where

DSAP address = destination service access point address field,
 SSAP address = source service access point address field,
 Control = control field,
 16 bits for formats that include sequence numbering, and 8 bits for
 formats that do not (see 5.2).
 Information = information field,
 * = multiplication, and
 M = an integer value equal to or greater than 0. (Upper bound of M
 is a function of the medium access control methodology used.)

Figure 3.2. LLC PDU Format

3.3. Elements of the LLC PDU.

3.3.1. *Address Fields.* Each LLC PDU shall contain two address fields: the Destination Service Access Point (DSAP) Address field and the Source Service Access Point (SSAP) Address field, in that order. Each address field shall contain only a single address. The DSAP Address field shall identify the one or more service access points for which the LLC information field is intended. The SSAP Address field shall identify the specific service access point from which the LLC information field was initiated.

3.3.1.1. *Address Representation.* The representation of each address field shall be as shown in Figures 3.3.a and 3.3.b:

- (1) Each address field shall contain one octet.
- (2) Each address field shall contain 7 bits of actual address, and one bit that shall be used in the DSAP Address field to identify the DSAP address as either an individual or a group address (called the address type designation bit) and in the SSAP Address field to identify that the LLC PDU is a command or a response (called the command/response identifier bit).
- (3) The address type designation bit shall be located in the least significant bit position of the DSAP address field. If this bit is "0", it shall indicate that the address is an individual DSAP address. If this bit is "1", it shall indicate that the address is a group DSAP address that identifies none, one or more, or all of the service access points that are serviced by the LLC entity.
- (4) The command/response identifier bit shall be located in the least significant bit position of the SSAP address field. If this bit is "0", it shall indicate that the LLC PDU is a command. If this bit is "1", it shall indicate that the LLC PDU is a response.